Cisco IOS IP Application Services Configuration Guide

Release 12.4

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Cisco IOS IP Application Services Configuration Guide

About Cisco IOS and Cisco IOS XE Software Documentation

Last updated: August 6, 2008

This document describes the objectives, audience, conventions, and organization used in Cisco IOS and Cisco IOS XE software documentation, collectively referred to in this document as Cisco IOS documentation. Also included are resources for obtaining technical assistance, additional documentation, and other information from Cisco. This document is organized into the following sections:

- Documentation Objectives, page i
- Audience, page i
- Documentation Conventions, page ii
- Documentation Organization, page iii
- Additional Resources and Documentation Feedback, page xi

Documentation Objectives

Cisco IOS documentation describes the tasks and commands available to configure and maintain Cisco networking devices.

Audience

The Cisco IOS documentation set is i ntended for users who configure and maintain Cisco networking devices (such as routers and switches) but who may not be familiar with the configuration and maintenance tasks, the relationship among tasks, or the Cisco IOS commands necessary to perform particular tasks. The Cisco IOS documentation set is also intended for those users experienced with Cisco IOS who need to know about new features, new configuration options, and new software characteristics in the current Cisco IOS release.

Documentation Conventions

In Cisco IOS documentation, the term *router* may be used to refer to various Cisco products; for example, routers, access servers, and switches. These and other networking devices that support Cisco IOS software are shown interchangeably in examples and are used only for illustrative purposes. An example that shows one product does not necessarily mean that other products are not supported.

This section includes the following topics:

- Typographic Conventions, page ii
- Command Syntax Conventions, page ii
- Software Conventions, page iii
- Reader Alert Conventions, page iii

Typographic Conventions

Cisco IOS documentation uses the following typographic conventions:

Convention	Description
^ or Ctrl	Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination **^D** or **Ctrl-D** means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
string	A string is a nonquoted set of characters shown in italics. For example, when setting a Simple Network Management Protocol (SNMP) community string to *public*, do not use quotation marks around the string; otherwise, the string will include the quotation marks.

Command Syntax Conventions

Cisco IOS documentation uses the following command syntax conventions:

Convention	Description
bold	Bold text indicates commands and keywords that you enter as shown.
italic	Italic text indicates arguments for which you supply values.
[x]	Square brackets enclose an optional keyword or argument.
|	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
[x | y]	Square brackets enclosing keywords or arguments separated by a pipe indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a pipe indicate a required choice.
[x {y | z}]	Braces and a pipe within square brackets indicate a required choice within an optional element.

Software Conventions

Cisco IOS uses the following program code conventions:

Convention	Description
`Courier font`	Courier font is used for information that is displayed on a PC or terminal screen.
`Bold Courier font`	Bold Courier font indicates text that the user must enter.
< >	Angle brackets enclose text that is not displayed, such as a password. Angle brackets also are used in contexts in which the italic font style is not supported; for example, ASCII text.
!	An exclamation point at the beginning of a line indicates that the text that follows is a comment, not a line of code. An exclamation point is also displayed by Cisco IOS software for certain processes.
[ ]	Square brackets enclose default responses to system prompts.

Reader Alert Conventions

The Cisco IOS documentation set uses the following conventions for reader alerts:

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Documentation Organization

This section describes the Cisco IOS documentation set, how it is organized, and how to access it on Cisco.com. Included are lists of configuration guides, command references, and supplementary references and resources that make up the documentation set. The following topics are included:

- Cisco IOS Documentation Set, page iv
- Cisco IOS Documentation on Cisco.com, page iv
- Configuration Guides, Command References, and Supplementary Resources, page v

Cisco IOS Documentation Set

Cisco IOS documentation consists of the following:

- Release notes and caveats provide information about platform, technology, and feature support for a release and describe severity 1 (catastrophic), severity 2 (severe), and severity 3 (moderate) defects in released Cisco IOS code. Review release notes before other documents to learn whether or not updates have been made to a feature.
- Sets of configuration guides and command references organized by technology and published for each standard Cisco IOS release.
  - Configuration guides—Compilations of documents that provide informational and task-oriented descriptions of Cisco IOS features.
  - Command references—Compilations of command pages that provide detailed information about the commands used in the Cisco IOS features and processes that make up the related configuration guides. For each technology, there is a single command reference that covers all Cisco IOS releases and that is updated at each standard release.
- Lists of all the commands in a specific release and all commands that are new, modified, removed, or replaced in the release.
- Command reference book for **debug** commands. Command pages are listed in alphabetical order.
- Reference book for system messages for all Cisco IOS releases.

Cisco IOS Documentation on Cisco.com

The following sections describe the documentation organization and how to access various document types.

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

New Features List

The New Features List for each release provides a list of all features in the release with hyperlinks to the feature guides in which they are documented.

Feature Guides

Cisco IOS features are documented in feature guides. Feature guides describe one feature or a group of related features that are supported on many different software releases and platforms. Your Cisco IOS software release or platform may not support all the features documented in a feature guide. See the Feature Information table at the end of the feature guide for information about which features in that guide are supported in your software release.

Configuration Guides

Configuration guides are provided by technology and release and comprise a set of individual feature guides relevant to the release and technology.

Command References

Command reference books describe Cisco IOS commands that are supported in many different software releases and on many different platforms. The books are provided by technology. For information about all Cisco IOS commands, use the Command Lookup Tool at http://tools.cisco.com/Support/CLILookup or the *Cisco IOS Master Command List, All Releases*, at http://www.cisco.com/en/US/docs/ios/mcl/all_release/all_mcl.html.

Cisco IOS Supplementary Documents and Resources

Supplementary documents and resources are listed in Table 2 on page xi.

Configuration Guides, Command References, and Supplementary Resources

Table 1 lists, in alphabetical order, Cisco IOS and Cisco IOS XE software configuration guides and command references, including brief descriptions of the contents of the documents. The Cisco IOS command references are comprehensive, meaning that they include commands for both Cisco IOS software and Cisco IOS XE software, for all releases. The configuration guides and command references support many different software releases and platforms. Your Cisco IOS software release or platform may not support all these technologies.

For additional information about configuring and operating specific networking devices, go to the Product Support area of Cisco.com at http://www.cisco.com/web/psa/products/index.html.

Table 2 lists documents and resources that supplement the Cisco IOS software configuration guides and command references. These supplementary resources include release notes and caveats; master command lists; new, modified, removed, and replaced command lists; system messages; and the debug command reference.

Table 1 Cisco IOS and Cisco IOS XE Configuration Guides and Command References

Configuration Guide and Command Reference Titles	Features/Protocols/Technologies
Cisco IOS AppleTalk Configuration Guide *Cisco IOS XE AppleTalk Configuration Guide* *Cisco IOS AppleTalk Command Reference*	AppleTalk protocol.
Cisco IOS Asynchronous Transfer Mode Configuration Guide *Cisco IOS Asynchronous Transfer Mode Command Reference*	LAN ATM, multiprotocol over ATM (MPoA), and WAN ATM.

Table 1 Cisco IOS and Cisco IOS XE Configuration Guides and Command References (continued)

Configuration Guide and Command Reference Titles	Features/Protocols/Technologies
Cisco IOS Bridging and IBM Networking Configuration Guide *Cisco IOS Bridging Command Reference* *Cisco IOS IBM Networking Command Reference*	• Transparent and source-route transparent (SRT) bridging, source-route bridging (SRB), Token Ring Inter-Switch Link (TRISL), and token ring route switch module (TRRSM). • Data-link switching plus (DLSw+), serial tunnel (STUN), block serial tunnel (BSTUN); logical link control, type 2 (LLC2), synchronous data link control (SDLC); IBM Network Media Translation, including Synchronous Data Logical Link Control (SDLLC) and qualified LLC (QLLC); downstream physical unit (DSPU), Systems Network Architecture (SNA) service point, SNA frame relay access, advanced peer-to-peer networking (APPN), native client interface architecture (NCIA) client/server topologies, and IBM Channel Attach.
Cisco IOS Broadband and DSL Configuration Guide *Cisco IOS XE Broadband and DSL Configuration Guide* *Cisco IOS Broadband and DSL Command Reference*	Point-to-Point Protocol (PPP) over ATM (PPPoA) and PPP over Ethernet (PPPoE).
Cisco IOS Carrier Ethernet Configuration Guide *Cisco IOS Carrier Ethernet Command Reference*	Connectivity fault management (CFM), Ethernet Local Management Interface (ELMI), IEEE 802.3ad link bundling, Link Layer Discovery Protocol (LLDP), media endpoint discovery (MED), and operations, administration, and maintenance (OAM).
Cisco IOS Configuration Fundamentals Configuration Guide *Cisco IOS XE Configuration Fundamentals Configuration Guide* *Cisco IOS Configuration Fundamentals Command Reference*	Autoinstall, Setup, Cisco IOS command-line interface (CLI), Cisco IOS file system (IFS), Cisco IOS web browser user interface (UI), basic file transfer services, and file management.
Cisco IOS DECnet Configuration Guide *Cisco IOS XE DECnet Configuration Guide* *Cisco IOS DECnet Command Reference*	DECnet protocol.
Cisco IOS Dial Technologies Configuration Guide *Cisco IOS XE Dial Technologies Configuration Guide* *Cisco IOS Dial Technologies Command Reference*	Asynchronous communications, dial backup, dialer technology, dial-in terminal services and AppleTalk remote access (ARA), large scale dialout, dial-on-demand routing, dialout, modem and resource pooling, ISDN, multilink PPP (MLP), PPP, virtual private dialup network (VPDN).
Cisco IOS Flexible NetFlow Configuration Guide *Cisco IOS Flexible NetFlow Command Reference*	Flexible NetFlow.

Table 1 Cisco IOS and Cisco IOS XE Configuration Guides and Command References (continued)

Configuration Guide and Command Reference Titles	Features/Protocols/Technologies
Cisco IOS H.323 Configuration Guide	Gatekeeper enhancements for managed voice services, Gatekeeper Transaction Message Protocol, gateway codec order preservation and shutdown control, H.323 dual tone multifrequency relay, H.323 version 2 enhancements, Network Address Translation (NAT) support of H.323 v2 Registration, Admission, and Status (RAS) protocol, tokenless call authorization, and VoIP gateway trunk and carrier-based routing.
Cisco IOS High Availability Configuration Guide *Cisco IOS XE High Availability Configuration Guide* *Cisco IOS High Availability Command Reference*	A variety of High Availability (HA) features and technologies that are available for different network segments (from enterprise access to service provider core) to facilitate creation of end-to-end highly available networks. Cisco IOS HA features and technologies can be categorized in three key areas: system-level resiliency, network-level resiliency, and embedded management for resiliency.
Cisco IOS Integrated Session Border Controller Command Reference	A VoIP-enabled device that is deployed at the edge of networks. An SBC is a toolkit of functions, such as signaling interworking, network hiding, security, and quality of service (QoS).
Cisco IOS Intelligent Service Gateway Configuration Guide *Cisco IOS Intelligent Service Gateway Command Reference*	Subscriber identification, service and policy determination, session creation, session policy enforcement, session life-cycle management, accounting for access and service usage, session state monitoring.
Cisco IOS Interface and Hardware Component Configuration Guide *Cisco IOS XE Interface and Hardware Component Configuration Guide* *Cisco IOS Interface and Hardware Component Command Reference*	LAN interfaces, logical interfaces, serial interfaces, virtual interfaces, and interface configuration.
Cisco IOS IP Addressing Services Configuration Guide *Cisco IOS XE Addressing Services Configuration Guide* *Cisco IOS IP Addressing Services Command Reference*	Address Resolution Protocol (ARP), Network Address Translation (NAT), Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), and Next Hop Address Resolution Protocol (NHRP).
Cisco IOS IP Application Services Configuration Guide *Cisco IOS XE IP Application Services Configuration Guide* *Cisco IOS IP Application Services Command Reference*	Enhanced Object Tracking (EOT), Gateway Load Balancing Protocol (GLBP), Hot Standby Router Protocol (HSRP), IP Services, Server Load Balancing (SLB), Stream Control Transmission Protocol (SCTP), TCP, Web Cache Communication Protocol (WCCP), User Datagram Protocol (UDP), and Virtual Router Redundancy Protocol (VRRP).
Cisco IOS IP Mobility Configuration Guide *Cisco IOS IP Mobility Command Reference*	Mobile ad hoc networks (MANet) and Cisco mobile networks.
Cisco IOS IP Multicast Configuration Guide *Cisco IOS XE IP Multicast Configuration Guide* *Cisco IOS IP Multicast Command Reference*	Protocol Independent Multicast (PIM) sparse mode (PIM-SM), bidirectional PIM (bidir-PIM), Source Specific Multicast (SSM), Multicast Source Discovery Protocol (MSDP), Internet Group Management Protocol (IGMP), and Multicast VPN (MVPN).

Table 1 Cisco IOS and Cisco IOS XE Configuration Guides and Command References (continued)

Configuration Guide and Command Reference Titles	Features/Protocols/Technologies
Cisco IOS IP Routing Protocols Configuration Guide *Cisco IOS XE IP Routing Protocols Configuration Guide* *Cisco IOS IP Routing Protocols Command Reference*	Border Gateway Protocol (BGP), multiprotocol BGP, multiprotocol BGP extensions for IP multicast, bidirectional forwarding detection (BFD), Enhanced Interior Gateway Routing Protocol (EIGRP), Interior Gateway Routing Protocol (IGRP), Intermediate System-to-Intermediate System (IS-IS), on-demand routing (ODR), Open Shortest Path First (OSPF), and Routing Information Protocol (RIP).
Cisco IOS IP SLAs Configuration Guide *Cisco IOS XE IP SLAs Configuration Guide* *Cisco IOS IP SLAs Command Reference*	Cisco IOS IP Service Level Agreements (IP SLAs).
Cisco IOS IP Switching Configuration Guide *Cisco IOS XE IP Switching Configuration Guide* *Cisco IOS IP Switching Command Reference*	Cisco Express Forwarding, fast switching, and Multicast Distributed Switching (MDS).
Cisco IOS IPv6 Configuration Guide *Cisco IOS XE IPv6 Configuration Guide* *Cisco IOS IPv6 Command Reference*	For IPv6 features, protocols, and technologies, go to the IPv6 “Start Here” document at the following URL: http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-roadmap.html
Cisco IOS ISO CLNS Configuration Guide *Cisco IOS XE ISO CLNS Configuration Guide* *Cisco IOS ISO CLNS Command Reference*	ISO connectionless network service (CLNS).
Cisco IOS LAN Switching Configuration Guide *Cisco IOS XE LAN Switching Configuration Guide* *Cisco IOS LAN Switching Command Reference*	VLANs, Inter-Switch Link (ISL) encapsulation, IEEE 802.10 encapsulation, IEEE 802.1Q encapsulation, and multilayer switching (MLS).
Cisco IOS Mobile Wireless Gateway GPRS Support Node Configuration Guide *Cisco IOS Mobile Wireless Gateway GPRS Support Node Command Reference*	Cisco IOS Gateway GPRS Support Node (GGSN) in a 2.5-generation general packet radio service (GPRS) and 3-generation universal mobile telecommunication system (UMTS) network.
Cisco IOS Mobile Wireless Home Agent Configuration Guide *Cisco IOS Mobile Wireless Home Agent Command Reference*	Cisco Mobile Wireless Home Agent, an anchor point for mobile terminals for which mobile IP or proxy mobile IP services are provided.
Cisco IOS Mobile Wireless Packet Data Serving Node Configuration Guide *Cisco IOS Mobile Wireless Packet Data Serving Node Command Reference*	Cisco Packet Data Serving Node (PDSN), a wireless gateway that is between the mobile infrastructure and standard IP networks and that enables packet data services in a code division multiple access (CDMA) environment.
Cisco IOS Mobile Wireless Radio Access Networking Configuration Guide *Cisco IOS Mobile Wireless Radio Access Networking Command Reference*	Cisco IOS radio access network products.

Table 1 Cisco IOS and Cisco IOS XE Configuration Guides and Command References (continued)

Configuration Guide and Command Reference Titles	Features/Protocols/Technologies
Cisco IOS Multiprotocol Label Switching Configuration Guide *Cisco IOS XE Multiprotocol Label Switching Configuration Guide* *Cisco IOS Multiprotocol Label Switching Command Reference*	MPLS Label Distribution Protocol (LDP), MPLS Layer 2 VPNs, MPLS Layer 3 VPNs, MPLS Traffic Engineering (TE), and MPLS Embedded Management (EM) and MIBs.
Cisco IOS Multi-Topology Routing Configuration Guide *Cisco IOS Multi-Topology Routing Command Reference*	Unicast and multicast topology configurations, traffic classification, routing protocol support, and network management support.
Cisco IOS NetFlow Configuration Guide *Cisco IOS XE NetFlow Configuration Guide* *Cisco IOS NetFlow Command Reference*	Network traffic data analysis, aggregation caches, export features.
Cisco IOS Network Management Configuration Guide *Cisco IOS XE Network Management Configuration Guide* *Cisco IOS Network Management Command Reference*	Basic system management; system monitoring and logging; troubleshooting, logging, and fault management; Cisco Discovery Protocol; Cisco IOS Scripting with Tool Control Language (Tcl); Cisco networking services (CNS); DistributedDirector; Embedded Event Manager (EEM); Embedded Resource Manager (ERM); Embedded Syslog Manager (ESM); HTTP; Remote Monitoring (RMON); SNMP; and VPN Device Manager Client for Cisco IOS Software (XSM Configuration).
Cisco IOS Novell IPX Configuration Guide *Cisco IOS XE Novell IPX Configuration Guide* *Cisco IOS Novell IPX Command Reference*	Novell Internetwork Packet Exchange (IPX) protocol.
Cisco IOS Optimized Edge Routing Configuration Guide *Cisco IOS Optimized Edge Routing Command Reference*	Optimized edge routing (OER) monitoring, policy configuration, routing control, logging and reporting, and VPN IPsec/generic routing encapsulation (GRE) tunnel interface optimization.
Cisco IOS Quality of Service Solutions Configuration Guide *Cisco IOS XE Quality of Service Solutions Configuration Guide* *Cisco IOS Quality of Service Solutions Command Reference*	Class-based weighted fair queuing (CBWFQ), custom queuing, distributed traffic shaping (DTS), generic traffic shaping (GTS), IP- to-ATM class of service (CoS), low latency queuing (LLQ), modular QoS CLI (MQC), Network-Based Application Recognition (NBAR), priority queuing, Security Device Manager (SDM), Multilink PPP (MLPPP) for QoS, header compression, AutoQoS, QoS features for voice, Resource Reservation Protocol (RSVP), weighted fair queuing (WFQ), and weighted random early detection (WRED).
Cisco IOS Security Configuration Guide *Cisco IOS XE Security Configuration Guide* *Cisco IOS Security Command Reference*	Access control lists (ACLs), authentication, authorization, and accounting (AAA), firewalls, IP security and encryption, neighbor router authentication, network access security, network data encryption with router authentication, public key infrastructure (PKI), RADIUS, TACACS+, terminal access security, and traffic filters.

Table 1 Cisco IOS and Cisco IOS XE Configuration Guides and Command References (continued)

Configuration Guide and Command Reference Titles	Features/Protocols/Technologies
Cisco IOS Service Selection Gateway Configuration Guide *Cisco IOS Service Selection Gateway Command Reference*	Subscriber authentication, service access, and accounting.
Cisco IOS Software Activation Configuration Guide *Cisco IOS Software Activation Command Reference*	An orchestrated collection of processes and components to activate Cisco IOS software feature sets by obtaining and validating Cisco software licenses.
Cisco IOS Software Modularity Installation and Configuration Guide *Cisco IOS Software Modularity Command Reference*	Installation and basic configuration of software modularity images, including installations on single and dual route processors, installation rollbacks, software modularity binding, software modularity processes and patches.
Cisco IOS Terminal Services Configuration Guide *Cisco IOS Terminal Services Command Reference* *Cisco IOS XE Terminal Services Command Reference*	DEC, local-area transport (LAT), and X.25 packet assembler/disassembler (PAD).
Cisco IOS Virtual Switch Command Reference	Virtual switch redundancy, high availability, and packet handling; converting between standalone and virtual switch modes; virtual switch link (VSL); Virtual Switch Link Protocol (VSLP). **Note** For information about virtual switch configuration, refer to the product-specific software configuration information for the Cisco Catalyst 6500 series switch or for the Metro Ethernet 6500 series switch.
Cisco IOS Voice Configuration Library *Cisco IOS Voice Command Reference*	Cisco IOS support for voice call control protocols, interoperability, physical and virtual interface management, and troubleshooting. The library includes documentation for IP telephony applications.
Cisco IOS VPDN Configuration Guide *Cisco IOS XE VPDN Configuration Guide* *Cisco IOS VPDN Command Reference*	Layer 2 Tunneling Protocol (L2TP) dial-out load balancing and redundancy, L2TP extended failover, L2TP security VPDN, multihop by Dialed Number Identification Service (DNIS), timer and retry enhancements for L2TP and Layer 2 Forwarding (L2F), RADIUS Attribute 82: tunnel assignment ID, shell-based authentication of VPDN users, tunnel authentication via RADIUS on tunnel terminator.
Cisco IOS Wide-Area Networking Configuration Guide *Cisco IOS XE Wide-Area Networking Configuration Guide* *Cisco IOS Wide-Area Networking Command Reference*	Frame Relay, Layer 2 Tunneling Protocol Version 3 (L2TPv3), Link Access Procedure, Balanced (LAPB), Switched Multimegabit Data Service (SMDS), and X.25.
Cisco IOS Wireless LAN Configuration Guide *Cisco IOS Wireless LAN Command Reference*	Broadcast key rotation, IEEE 802.11x support, IEEE 802.1x authenticator, IEEE 802.1x local authentication service for Extensible Authentication Protocol-Flexible Authentication via Secure Tunneling (EAP-FAST), Multiple Basic Service Set ID (BSSID), Wi-Fi Multimedia (WMM) required elements, and Wi-Fi Protected Access (WPA).

Table 2 Cisco IOS Supplementary Documents and Resources

Document Title	Description
Cisco IOS Master Command List, All Releases	Alphabetical list of all the commands documented in all Cisco IOS releases.
Cisco IOS New, Modified, Removed, and Replaced Commands	List of all the new, modified, removed, and replaced commands for a Cisco IOS release.
Cisco IOS Software System Messages	List of Cisco IOS system messages and descriptions. System messages may indicate problems with your system; be informational only; or may help diagnose problems with communications lines, internal hardware, or the system software.
Cisco IOS Debug Command Reference	Alphabetical list of **debug** commands including brief descriptions of use, command syntax, and usage guidelines.
Release Notes and Caveats	Information about new and changed features, system requirements, and other useful information about specific software releases; information about defects in specific Cisco IOS software releases.
MIBs	Files used for network monitoring. To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator at the following URL: http://www.cisco.com/go/mibs
RFCs	Standards documents maintained by the Internet Engineering Task Force (IETF) that Cisco IOS documentation references where applicable. The full text of referenced RFCs may be obtained at the following URL: http://www.rfc-editor.org/

Additional Resources and Documentation Feedback

What's New in Cisco Product Documentation is published monthly and describes all new and revised Cisco technical documentation. The *What's New in Cisco Product Documentation* publication also provides information about obtaining the following resources:

- Technical documentation
- Cisco product security overview
- Product alerts and field notices
- Technical assistance

Cisco IOS technical documentation includes embedded feedback forms where you can rate documents and provide suggestions for improvement. Your feedback helps us improve our documentation.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Using the Command-Line Interface in Cisco IOS and Cisco IOS XE Software

Last updated: August 6, 2008

This document provides basic information about the command-line interface (CLI) in Cisco IOS and Cisco IOS XE software and how you can use some of the CLI features. This document contains the following sections:

- Initially Configuring a Device, page i
- Using the CLI, page ii
- Saving Changes to a Configuration, page xii
- Additional Information, page xii

For more information about using the CLI, see the "Using the Cisco IOS Command-Line Interface" section of the *Cisco IOS Configuration Fundamentals Configuration Guide*.

For information about the software documentation set, see the "About Cisco IOS and Cisco IOS XE Software Documentation" document.

Initially Configuring a Device

Initially configuring a device varies by platform. For information about performing an initial configuration, see the hardware installation documentation that is provided with the original packaging of the product or go to the Product Support area of Cisco.com at http://www.cisco.com/web/psa/products/index.html.

After you have performed the initial configuration and connected the device to your network, you can configure the device by using the console port or a remote access method, such as Telnet or Secure Shell (SSH), to access the CLI or by using the configuration method provided on the device, such as Security Device Manager.

Changing the Default Settings for a Console or AUX Port

There are only two changes that you can make to a console port and an AUX port:

- Change the port speed with the **config-register 0x** command. Changing the port speed is not recommended. The well-known default speed is 9600.
- Change the behavior of the port; for example, by adding a password or changing the timeout value.

Note The AUX port on the Route Processor (RP) installed in a Cisco ASR1000 series router does not serve any useful customer purpose and should be accessed only under the advisement of a customer support representative.

Using the CLI

This section describes the following topics:

Understanding Command Modes

The CLI command mode structure is hierarchical, and each mode supports a set of specific commands. This section describes the most common of the many modes that exist.

Table 1 lists common command modes with associated CLI prompts, access and exit methods, and a brief description of how each mode is used.

Table 1 CLI Command Modes

Command Mode	Access Method	Prompt	Exit Method	Mode Usage
User EXEC	Log in.	`Router>`	Issue the **logout** or **exit** command.	• Change terminal settings. • Perform basic tests. • Display device status.
Privileged EXEC	From user EXEC mode, issue the **enable** command.	`Router#`	Issue the **disable** command or the **exit** command to return to user EXEC mode.	• Issue **show** and **debug** commands. • Copy images to the device. • Reload the device. • Manage device configuration files. • Manage device file systems.
Global configuration	From privileged EXEC mode, issue the **configure terminal** command.	`Router(config)#`	Issue the **exit** command or the **end** command to return to privileged EXEC mode.	Configure the device.
Interface configuration	From global configuration mode, issue the **interface** command.	`Router(config-if)#`	Issue the **exit** command to return to global configuration mode or the **end** command to return to privileged EXEC mode.	Configure individual interfaces.
Line configuration	From global configuration mode, issue the **line vty** or **line console** command.	`Router(config-line)#`	Issue the **exit** command to return to global configuration mode or the **end** command to return to privileged EXEC mode.	Configure individual terminal lines.

Table 1 CLI Command Modes (continued)

Command Mode	Access Method	Prompt	Exit Method	Mode Usage
ROM monitor	From privileged EXEC mode, issue the **reload** command. Press the **Break** key during the first 60 seconds while the system is booting.	`rommon # >` The # symbol represents the line number and increments at each prompt.	Issue the **continue** command.	• Run as the default operating mode when a valid image cannot be loaded. • Access the fall-back procedure for loading an image when the device lacks a valid image and cannot be booted. • Perform password recovery when a CTRL-Break sequence is issued within 60 seconds of a power-on or reload event.
Diagnostic (available only on the Cisco ASR1000 series router)	The router boots or enters diagnostic mode in the following scenarios. When a Cisco IOS process or processes fail, in most scenarios the router will reload. • A user-configured access policy was configured using the **transport-map** command, which directed the user into diagnostic mode. • The router was accessed using an RP auxiliary port. • A break signal (**Ctrl-C**, **Ctrl-Shift-6**, or the **send break** command) was entered, and the router was configured to enter diagnostic mode when the break signal was received.	`Router(diag)#`	If a Cisco IOS process failure is the reason for entering diagnostic mode, the failure must be resolved and the router must be rebooted to exit diagnostic mode. If the router is in diagnostic mode because of a transport-map configuration, access the router through another port or using a method that is configured to connect to the Cisco IOS CLI. If the RP auxiliary port was used to access the router, use another port for access. Accessing the router through the auxiliary port is not useful for customer purposes.	• Inspect various states on the router, including the Cisco IOS state. • Replace or roll back the configuration. • Provide methods of restarting the Cisco IOS software or other processes. • Reboot hardware, such as the entire router, an RP, an ESP, a SIP, a SPA, or possibly other hardware components. • Transfer files into or off of the router using remote access methods such as FTP, TFTP, and SCP.

EXEC commands are not saved when the software reboots. Commands that you issue in a configuration mode can be saved to the startup configuration. If you save the running configuration to the startup configuration, these commands will execute when the software is rebooted. Global configuration mode is the highest level of configuration mode. From global configuration mode, you can enter a variety of other configuration modes, including protocol-specific modes.

ROM monitor mode is a separate mode that is used when the software cannot load properly. If a valid software image is not found when the software boots or if the configuration file is corrupted at startup, the software might enter ROM monitor mode. Use the question symbol (?) to view the commands that you can use while the device is in ROM monitor mode.

```
rommon 1 > ?
alias                set and display aliases command
boot                 boot up an external process
confreg              configuration register utility
cont                 continue executing a downloaded image
context              display the context of a loaded image
cookie               display contents of cookie PROM in hex
.
.
.
rommon 2 >
```

The following example shows how the command prompt changes to indicate a different command mode:

```
Router> enable
Router# configure terminal
Router(config)# interface ethernet 1/1
Router(config-if)# ethernet
Router(config-line)# exit
Router(config)# end
Router#
```

Note A keyboard alternative to the **end** command is Ctrl-Z.

Using the Interactive Help Feature

The CLI includes an interactive Help feature. Table 2 describes how to use the Help feature.

Table 2 CLI Interactive Help Commands

Command	Purpose
help	Provides a brief description of the help feature in any command mode.
?	Lists all commands available for a particular command mode.
*partial command***?**	Provides a list of commands that begin with the character string (no space between the command and the question mark).
partial command<**Tab**>	Completes a partial command name (no space between the command and <Tab>).
command **?**	Lists the keywords, arguments, or both associated with the command (space between the command and the question mark).
command keyword **?**	Lists the arguments that are associated with the keyword (space between the keyword and the question mark).

The following examples show how to use the help commands:

help

```
Router> help
```

```
Help may be requested at any point in a command by entering a question mark '?'.  If
nothing matches, the help list will be empty and you must backup until entering a '?'
shows the available options.
```

```
Two styles of help are provided:
```

```
1. Full help is available when you are ready to enter a command argument (e.g. 'show ?')
and describes each possible argument.
```

```
2. Partial help is provided when an abbreviated argument is entered and you want to know
what arguments match the input (e.g. 'show pr?'.)
```

?

```
Router# ?
Exec commands:
  access-enable        Create a temporary access-List entry
  access-profile       Apply user-profile to interface
  access-template      Create a temporary access-List entry
  alps                 ALPS exec commands
  archive              manage archive files
<snip>
```

partial command?

```
Router(config)# zo?
zone  zone-pair
```

partial command<Tab>

```
Router(config)# we<Tab> webvpn
```

command ?

```
Router(config-if)# pppoe ?
  enable        Enable pppoe
  max-sessions  Maximum PPPOE sessions
```

command keyword ?

```
Router(config-if)# pppoe enable ?
  group  attach a BBA group
  <cr>
```

Understanding Command Syntax

Command syntax is the format in which a command should be entered in the CLI. Commands include the name of the command, keywords, and arguments. Keywords are alphanumeric strings that are used literally. Arguments are placeholders for values that a user must supply. Keywords and arguments may be required or optional.

Specific conventions convey information about syntax and command elements. Table 3 describes these conventions.

Table 3 CLI Syntax Conventions

Symbol/Text	Function	Notes
< > (angle brackets)	Indicate that the option is an argument.	Sometimes arguments are displayed without angle brackets.
A.B.C.D.	Indicates that you must enter a dotted decimal IP address.	Angle brackets (< >) are not always used to indicate that an IP address is an argument.
WORD (all capital letters)	Indicates that you must enter one word.	Angle brackets (< >) are not always used to indicate that a WORD is an argument.
LINE (all capital letters)	Indicates that you must enter more than one word.	Angle brackets (< >) are not always used to indicate that a LINE is an argument.
<cr> (carriage return)	Indicates the end of the list of available keywords and arguments, and also indicates when keywords and arguments are optional. When <cr> is the only option, you have reached the end of the branch or the end of the command if the command has only one branch.	—

The following examples show syntax conventions:

```
Router(config)# ethernet cfm domain ?
  WORD  domain name
Router(config)# ethernet cfm domain dname ?
  level
Router(config)# ethernet cfm domain dname level ?
  <0-7>  maintenance level number
Router(config)# ethernet cfm domain dname level 7 ?
  <cr>
Router(config)# snmp-server file-transfer access-group 10 ?
  protocol  protocol options
  <cr>
Router(config)# logging host ?
  Hostname or A.B.C.D  IP address of the syslog server
  ipv6                 Configure IPv6 syslog server
Router(config)# snmp-server file-transfer access-group 10 ?
  protocol  protocol options
  <cr>
```

Understanding Enable and Enable Secret Passwords

Some privileged EXEC commands are used for actions that impact the system, and it is recommended that you set a password for these commands to prevent unauthorized use. Two types of passwords, enable (not encrypted) and enable secret (encrypted), can be set. The following commands set these passwords and are issued in global configuration mode:

- **enable** *password*
- **enable secret** *password*

Using an enable secret password is recommended because it is encrypted and more secure than the enable password. When you use an enable secret password, text is encrypted (unreadable) before it is written to the config.text file. When you use an enable password, the text is written as entered (readable) to the config.text file.

Each type of password is case sensitive, can contain from 1 to 25 uppercase and lowercase alphanumeric characters, and can start with a number. Spaces are also valid password characters; for example, “two words” is a valid password. Leading spaces are ignored, but trailing spaces are recognized.

Note Both password commands have numeric keywords that are single integer values. If you choose a number for the first character of your password followed by a space, the system will read the number as if it were the numeric keyword and not as part of your password.

When both passwords are set, the enable secret password takes precedence over the enable password.

To remove a password, use the **no** form of the commands: **no enable** *password* or **no enable secret** *password*.

For more information about password recovery procedures for Cisco products, see http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_tech_note09186a00801746e6.shtml.

Using the Command History Feature

The CLI command history feature saves the commands you enter during a session in a command history buffer. The default number of commands saved is 10, but the number is configurable within the range of 0 to 256. This command history feature is particularly useful for recalling long or complex commands.

To change the number of commands saved in the history buffer for a terminal session, issue the **terminal history size** command:

```
Router# terminal history size num
```

A command history buffer is also available in line configuration mode with the same default and configuration options. To set the command history buffer size for a terminal session in line configuration mode, issue the **history** command:

```
Router(config-line)# history [size num]
```

To recall commands from the history buffer, use the following methods:

- Press Ctrl-P or the up arrow key—Recalls commands beginning with the most recent command. Repeat the key sequence to recall successively older commands.

- Press Ctrl-N or the down arrow key—Recalls the most recent commands in the history buffer after they have been recalled using Ctrl-P or the up arrow key. Repeat the key sequence to recall successively more recent commands.

Note The arrow keys function only on ANSI-compatible terminals such as the VT100.

- Issue the **show history** command in user EXEC or privileged EXEC mode—Lists the most recent commands that you entered. The number of commands that are displayed is determined by the setting of the **terminal history size** and **history** commands.

  The CLI command history feature is enabled by default. To disable this feature for a terminal session, issue the **terminal no history** command in user EXEC or privileged EXEC mode or the **no history** command in line configuration mode.

Abbreviating Commands

Typing a complete command name is not always required for the command to execute. The CLI recognizes an abbreviated command when the abbreviation contains enough characters to uniquely identify the command. For example, the **show version** command can be abbreviated as **sh ver**. It cannot be abbreviated as **s ver** because **s** could mean **show**, **set**, or **systat**. The **sh v** abbreviation also is not valid because the **show** command has **vrrp** as a keyword in addition to **version**. (Command and keyword examples from Cisco IOS Release 12.4(13)T.)

Using Aliases for CLI Commands

To save time and the repetition of entering the same command multiple times, you can use a command alias. An alias can be configured to do anything that can be done at the command line, but an alias cannot move between modes, type in passwords, or perform any interactive functions.

Table 4 shows the default command aliases.

Table 4 Default Command Aliases

Command Alias	Original Command
h	help
lo	logout
p	ping
s	show
u or **un**	undebug
w	where

To create a command alias, issue the **alias** command in global configuration mode. The syntax of the command is **alias** *mode command-alias original-command*. Following are some examples:

- Router(config)# **alias exec prt partition**—privileged EXEC mode
- Router(config)# **alias configure sb source-bridge**—global configuration mode
- Router(config)# **alias interface rl rate-limit**—interface configuration mode

To view both default and user-created aliases, issue the **show alias** command.

For more information about the **alias** command, see
http://www.cisco.com/en/US/docs/ios/fundamentals/command/reference/cf_book.html.

Using the no and default Forms of Commands

Most configuration commands have a **no** form that is used to reset a command to its default value or disable a feature or function. For example, the **ip routing** command is enabled by default. To disable this command, you would issue the **no ip routing** command. To re-enable IP routing, you would issue the **ip routing** command.

Configuration commands may also have a **default** form, which returns the command settings to their default values. For commands that are disabled by default, using the **default** form has the same effect as using the **no** form of the command. For commands that are enabled by default and have default settings, the **default** form enables the command and returns the settings to their default values.

The **no** and **default** forms of commands are described in the command pages of command references.

Using the debug Command

A **debug** command produces extensive output that helps you troubleshoot problems in your network. These commands are available for many features and functions within Cisco IOS and Cisco IOS XE software. Some **debug** commands are **debug all**, **debug aaa accounting**, and **debug mpls packets**. To use **debug** commands during a Telnet session with a device, you must first enter the **terminal monitor** command. To turn off debugging completely, you must enter the **undebug all** command.

For more information about **debug** commands, see the *Cisco IOS Debug Command Reference* at
http://www.cisco.com/en/US/docs/ios/debug/command/reference/db_book.html.

⚠ **Caution** Debugging is a high priority and high CPU utilization process that can render your device unusable. Use **debug** commands only to troubleshoot specific problems. The best times to run debugging are during periods of low network traffic and when few users are interacting with the network. Debugging during these periods decreases the likelihood that the **debug** command processing overhead will affect network performance or user access or response times.

Filtering Output Using Output Modifiers

Many commands produce lengthy output that may use several screens to display. Using output modifiers, you can filter this output to show only the information that you want to see.

Three output modifiers are available and are described as follows:

- **begin** *regular expression*—Displays the first line in which a match of the regular expression is found and all lines that follow.
- **include** *regular expression*—Displays all lines in which a match of the regular expression is found.
- **exclude** *regular expression*—Displays all lines except those in which a match of the regular expression is found.

To use one of these output modifiers, type the command followed by the pipe symbol (|), the modifier, and the regular expression that you want to search for or filter. A regular expression is a case-sensitive alphanumeric pattern. It can be a single character or number, a phrase, or a more complex string.

The following example illustrates how to filter output of the **show interface** command to display only lines that include the expression "protocol."

```
Router# show interface | include protocol

FastEthernet0/0 is up, line protocol is up
Serial4/0 is up, line protocol is up
Serial4/1 is up, line protocol is up
Serial4/2 is administratively down, line protocol is down
Serial4/3 is administratively down, line protocol is down
```

Understanding CLI Error Messages

You may encounter some error messages while using the CLI. Table 5 shows the common CLI error messages.

Table 5 Common CLI Error Messages

Error Message	Meaning	How to Get Help
% Ambiguous command: "show con"	You did not enter enough characters for the command to be recognized.	Reenter the command followed by a space and a question mark (?). The keywords that you are allowed to enter for the command appear.
% Incomplete command.	You did not enter all the keywords or values required by the command.	Reenter the command followed by a space and a question mark (?). The keywords that you are allowed to enter for the command appear.
% Invalid input detected at "^" marker.	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The keywords that you are allowed to enter for the command appear.

For more system error messages, see the following documents:

- *Cisco IOS Release 12.2SR System Message Guide*
- *Cisco IOS System Messages, Volume 1 of 2* (Cisco IOS Release 12.4)
- *Cisco IOS System Messages, Volume 2 of 2* (Cisco IOS Release 12.4)

Saving Changes to a Configuration

To save changes that you made to the configuration of a device, you must issue the **copy running-config startup-config** command or the **copy system:running-config nvram:startup-config** command. When you issue these commands, the configuration changes that you made are saved to the startup configuration and saved when the software reloads or power to the device is turned off or interrupted. The following example shows the syntax of the **copy running-config startup-config** command:

```
Router# copy running-config startup-config
Destination filename [startup-config]?
```

You press Enter to accept the startup-config filename (the default), or type a new filename and then press Enter to accept that name. The following output is displayed indicating that the configuration was saved:

```
Building configuration...
[OK]
Router#
```

On most platforms, the configuration is saved to NVRAM. On platforms with a Class A flash file system, the configuration is saved to the location specified by the CONFIG_FILE environment variable. The CONFIG_FILE variable defaults to NVRAM.

Additional Information

- "Using the Cisco IOS Command-Line Interface" section of the *Cisco IOS Configuration Fundamentals Configuration Guide*:

  http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/cf_cli-basics.html

  or

  "Using Cisco IOS XE Software" chapter of the *Cisco ASR1000 Series Aggregation Services Routers Software Configuration Guide*:

  http://www.cisco.com/en/US/docs/routers/asr1000/configuration/guide/chassis/using_cli.html

- Cisco Product Support Resources

  http://www.cisco.com/web/psa/products/index.html

- Support area on Cisco.com (also search for documentation by task or product)

  http://www.cisco.com/en/US/support/index.html

- *White Paper: Cisco IOS Reference Guide*

  http://www.cisco.com/en/US/products/sw/iosswrel/ps1828/products_white_paper09186a008018305e.shtml

- Software Download Center (downloads; tools; licensing, registration, advisory, and general information) (requires Cisco.com User ID and password)

  http://www.cisco.com/kobayashi/sw-center/

- Error Message Decoder, a tool to help you research and resolve error messages for Cisco IOS software

  http://www.cisco.com/pcgi-bin/Support/Errordecoder/index.cgi

- Command Lookup Tool, a tool to help you find detailed descriptions of Cisco IOS commands (requires Cisco.com user ID and password)

  http://tools.cisco.com/Support/CLILookup

- Output Interpreter, a troubleshooting tool that analyzes command output of supported **show** commands

  https://www.cisco.com/pcgi-bin/Support/OutputInterpreter/home.pl\

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Cisco IOS IP Application Services Features Roadmap

First Published: May 5, 2008
Last Updated: July 11, 2008

This feature roadmap lists the Cisco IOS features documented in the *Cisco IOS IP Application Services Configuration Guide* and maps them to the documents in which they appear. The roadmap is organized so that you can select your release train and see the features in that release. Find the feature name you are searching for and click the URL in the "Where Documented" column to access the document containing that feature.

Note This feature road map does not contain the features documented in the First Hop Redundancy Protocol (FHRP) modules. For FHRP features, see the *FHRP Features Roadmap*.

Feature and Release Support

Table 1 lists IP Application Services feature support for the following Cisco IOS software release trains:

- Cisco IOS Release 12.2S
- Cisco IOS Release 12.2SB
- Cisco IOS Release 12.2SR
- Cisco IOS Release 12.2SX
- Cisco IOS Releases 12.2T, 12.3, 12.3T, 12.4 and 12.4T
- Cisco IOS Releases 12.2
- Cisco IOS XE Release 2
- Other Cisco IOS Releases

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 lists the most recent release of each software train first and the features in alphabetical order within the release.

Table 1 Supported IP Application Services Features

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Release 12.2S			
12.2(25)S	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
	WCCP Bypass Counters	The WCCP Bypass Counters feature allows you to display a count of packets that have been bypassed by a web cache and returned to the originating router to be forwarded normally.	*Configuring WCCP*
	WCCP Outbound ACL Check	The WCCP Outbound ACL Check feature enables you to ensure that traffic redirected by WCCP at an input interface is subjected to the outbound ACL checks that may be configured on the output interface prior to redirection. This feature is supported by WCCP Version 1 and Version 2.	*Configuring WCCP*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(14)S	AAA Load Balancing	IOS SLB provides RADIUS load-balancing capabilities for RADIUS authentication, authorization, and accounting (AAA) servers.	*Cisco IOS Server Load Balancing*
	Backup Server Farms	A backup server farm is a server farm that can be used when none of the real servers defined in a primary server farm is available to accept new connections.	*Cisco IOS Server Load Balancing*
	DFP Agent Subsystem Support	IOS SLB supports the Dynamic Feedback Protocol (DFP) Agent Subsystem feature, also called global load balancing, which enables client subsystems other than IOS SLB to act as DFP agents. With the DFP Agent Subsystem, you can use multiple DFP agents from different client subsystems at the same time.	*Cisco IOS Server Load Balancing*
	GPRS Load Balancing: Support for GPRS Tunneling Protocol (GTP) v0	IOS SLB supports both GTP Version 0 (GTP v0) and GTP Version 1 (GTP v1). Support for GTP enables IOS SLB to become "GTP aware," extending IOS SLB's knowledge into Layer 5.	*Cisco IOS Server Load Balancing*
	Multiple Firewall Farm Support	The Multiple Firewall Farm Support feature enables you to configure more than one firewall farm in each load-balancing device.	*Cisco IOS Server Load Balancing*
	Probes: DNS, Routed, and TCP Probes	IOS SLB probes determine the status of each real server in a server farm and of each firewall in a firewall farm.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(14)S	RADIUS Load Balancing: CDMA2000	IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). IOS SLB supports RADIUS load balancing for Simple IP CDMA2000 networks and Mobile IP CDMA2000 networks.	*Cisco IOS Server Load Balancing*
	RADIUS Load Balancing: General packet radio service (GPRS) networks	IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). IOS SLB supports RADIUS load balancing for GPRS networks. In a GPRS mobile wireless network, the RADIUS client is typically a gateway general packet radio service (GPRS) support node (GGSN).	*Cisco IOS Server Load Balancing*
	RADIUS Load Balancing: Multiple Service Gateway Server Farms	IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). IOS SLB supports RADIUS load balancing for multiple service gateway server farms (for example, one farm of SSGs and another of CSGs).	*Cisco IOS Server Load Balancing*
	Route Health Injection	By default, a virtual server's IP address is advertised (added to the routing table) when you bring the virtual server into service (using the **inservice** command). If you have a preferred host route to a website's virtual IP address, you can advertise that host route, but you have no guarantee that the IP address is available. However, you can use the **advertise** command to configure IOS SLB to advertise the host route only when IOS SLB has verified that the IP address is available. IOS SLB withdraws the advertisement when the IP address is no longer available. This function is known as route health injection.	*Cisco IOS Server Load Balancing*
	Static NAT	With static NAT, address translations exist in the NAT translation table as soon as you configure static NAT commands, and they remain in the translation table until you delete the static NAT commands.	*Cisco IOS Server Load Balancing*
	VPN Server Load Balancing	IOS SLB can balance Virtual Private Network (VPN) flows.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Release 12.2SB			
12.2(31)SB2	Clear IP Traffic CLI	The Clear IP Traffic CLI feature introduced the **clear ip traffic** command to clear all IP traffic statistics on a router instead of reloading the router. For added safety, you will see a confirmation prompt when entering this command.	*Configuring IP Services*
	ICMP Unreachable Rate Limiting User Feedback	The ICMP Unreachable Rate Limiting User Feedback feature enables you to clear and display packets that have been discarded because of an unreachable destination, and to configure a threshold interval for triggering error messages. When message logging is generated, it displays on your console.	*Configuring IP Services*
	TCP Application Flags Enhancement	The TCP Applications Flags Enhancement feature enables you to display additional flags with reference to TCP applications. There are two types of flags: status and option. The status flags indicate the status of TCP connections, such as retransmission timeouts, application closed, and synchronized (SYNC) handshakes for listen. The additional flags indicate the state of options such as whether or not a virtual private network (VPN) routing and forwarding (VRF) identification is set, whether or not a user is idle, and whether or not a keepalive timer is running.	*Configuring TCP*
	TCP Explicit Congestion Notification	The TCP Explicit Congestion Notification (ECN) feature provides a method for an intermediate router to notify the end hosts of impending network congestion. It also provides enhanced support for TCP sessions associated with applications that are sensitive to delay or packet loss including Telnet, web browsing, and transfer of audio and video data. The benefit of this feature is the reduction of delay and packet loss in data transmissions.	*Configuring TCP*
	TCP Show Extension	The TCP Show Extension feature introduces the capability to display addresses in IP format instead of hostname format and to display the virtual private network (VPN) routing and forwarding (VRF) table associated with the connection.	*Configuring TCP*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(31)SB2	TCP Window Scaling	The TCP Window Scaling feature adds support for the Window Scaling option in RFC 1323. A larger window size is recommended to improve TCP performance in network paths with large bandwidth, long-delay characteristics that are called Long Fat Networks (LFNs). This TCP Window Scaling enhancement provides that support.	*Configuring TCP*
Cisco IOS Release 12.2SR			
12.2(33)SRC1	Access Service Network (ASN) R6 Load Balancing	IOS SLB provides load balancing across a set of ASN gateways. The cluster of gateways appears to the base station as a single ASN gateway.	*Cisco IOS Server Load Balancing*
12.2(33)SRC	Connection Rate Limiting	IOS SLB enables you to specify the maximum connection rate allowed for a real server in a server farm.	*Cisco IOS Server Load Balancing*
	INOP_REAL State for Virtual Servers	The INOP_REAL State for Virtual Servers feature enables you to configure a virtual server such that, if all of the real servers that are associated with the virtual server are inactive, the following actions occur: • The virtual server is placed in the INOP_REAL state. • An SNMP trap is generated for the virtual server's state transition. • The virtual server stops answering ICMP requests.	*Cisco IOS Server Load Balancing*
	KeepAlive Application Protocol (KAL-AP) Agent Support	KAL-AP agent support enables IOS SLB to perform load balancing in a global server load balancing (GSLB) environment. KAL-AP provides load information along with its keepalive response message to the KAL-AP manager or GSLB device, such as the Global Site Selector (GSS), and helps the GSLB device load-balance client requests to the least-loaded IOS SLB devices.	*Cisco IOS Server Load Balancing*
	RADIUS Load Balancing Accelerated Data Plane Forwarding	RADIUS load balancing accelerated data plane forwarding, also known as Turbo RADIUS load balancing, is a high-performance solution that uses basic policy-based routing (PBR) route maps to handle subscriber data-plane traffic in a CSG environment. When Turbo RADIUS load balancing receives a RADIUS payload, it inspects the payload, extracts the framed-IP attribute, applies a route map to the IP address, and then determines which CSG is to handle the subscriber.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(33)SRB	GPRS Load Balancing: GPRS Load Balancing Maps	GPRS load balancing maps enable IOS SLB to categorize and route user traffic based on access point names (APNs).	*Cisco IOS Server Load Balancing*
	RADIUS Load Balancing: RADIUS Load Balancing Maps	RADIUS load balancing maps enable IOS SLB to categorize and route user traffic based on RADIUS calling station IDs and usernames. RADIUS load balancing maps is mutually exclusive with Turbo RADIUS load balancing and RADIUS load balancing accounting local acknowledgement.	*Cisco IOS Server Load Balancing*
12.2(33)SRA	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set.	*Configuring TCP*
	WCCP Increased Services	The WCCP Increased Services feature increases the number of services supported by WCCP to a maximum of 256.	*Configuring WCCP*
Cisco IOS Release 12.2SX			
12.2(33) SXH1	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
12.2(33) SXH	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set.	*Configuring TCP*
	WCCP Increased Services	The WCCP Increased Services feature increases the number of services supported by WCCP to a maximum of 256.	*Configuring WCCP*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(18)SXF 13	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
12.2(17d) SXE	GTP IMSI Sticky Database	IOS SLB can select a gateway general packet radio service (GPRS) support node (GGSN) for a given International Mobile Subscriber ID (IMSI), and forward all subsequent Packet Data Protocol (PDP) create requests from the same IMSI to the selected GGSN.	*Cisco IOS Server Load Balancing*
	Interface Awareness	Some environments require IOS SLB on both sides of a farm of CSGs, SSGs, or firewalls. For example, you might want IOS SLB to perform RADIUS load balancing on one side of a farm and firewall load balancing on the other, or firewall load balancing on both sides of a firewall farm.	*Cisco IOS Server Load Balancing*
	RADIUS Load Balancing: RADIUS Load Balancing IMSI Sticky Database	The IOS SLB RADIUS International Mobile Subscriber ID (IMSI) sticky database maps the IMSI address for each user to the corresponding gateway. This function enables IOS SLB to forward all subsequent flows for the same user to the same gateway.	*Cisco IOS Server Load Balancing*
12.2(17d) SXD	DFP and the Home Agent Director	For the Home Agent Director, you can define IOS SLB as a DFP manager and define a DFP agent on each home agent in the server farm, and the DFP agent can report the weights of the home agents. The DFP agents calculate the weight of each home agent based on CPU utilization, processor memory, and the maximum number of bindings that can be activated for each home agent.	*Cisco IOS Server Load Balancing*
12.2(17d) SXB1	GGSN-IOS SLB Messaging	This feature enables a GGSN to notify IOS SLB when certain conditions occur. The notifications enable IOS SLB to make intelligent decisions, which in turn improves GPRS load balancing and failure detection.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Releases 12.2T, 12.3, 12.3T, 12.4 and 12.4T			
12.4(20)T	FHRP - EOT Deprecation of **rtr** Keyword	Effective with Cisco IOS Release 12.4(20)T, the **track rtr** command is replaced by the **track ip sla** command.	*Configuring Enhanced Object Tracking*
	SCTP Release 4, Phase 2	Phase 2 of the SCTP Release 4 introduced the SCTP Add-IP feature. The SCTP Add-IP feature enables the ability to add or delete an IP address for an endpoint of an existing SCTP association and to communicate this change to the remote end.	*Stream Control Transmission Protocol*
	WCCP Layer 2 Redirection / Forwarding	The WCCP Layer 2 Redirection/Forwarding feature allows directly connected Cisco Content Engines to use Layer 2 redirection, which is more efficient than Layer 3 redirection via GRE encapsulation.	*Configuring WCCP*
	WCCP L2 Return	The WCCP L2 Return feature allows content engines to return packets to WCCP routers directly connected at Layer 2 by swapping the source and destination MAC addresses rather than tunnelling packets back to the router inside a Layer 3 GRE tunnel.	*Configuring WCCP*
	WCCP Mask Assignment	The WCCP Mask Assignment feature introduces support for ACNS/WAAS devices using mask assignment as a cache engine assignment method.	*Configuring WCCP*
12.4(15)T	SCTP Release 4	SCTP Release 4 introduced the SCTP Stream Reset and Authentication features.	*Stream Control Transmission Protocol*
12.4(11)T	SCTP Show/Clear CLI Enhancements	The Stream Control Transmission Protocol (SCTP) Show/Clear CLI Enhancements feature provides access to additional SCTP information that can help with troubleshooting potential problems. These enhancements also make the updated SCTP **show** and **clear** commands consistent with the CLI of other transport protocols.	*Stream Control Transmission Protocol*
	Show and Clear Commands for IOS Sockets	The Show and Clear Commands for IOS Sockets feature introduces the **show udp**, **show sockets**, and **clear sockets** commands. These new commands are useful for monitoring and managing the Cisco IOS Socket library.	*Configuring IP Services*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.4(2)T	Clear IP Traffic CLI	The Clear IP Traffic CLI feature introduced the **clear ip traffic** command to clear all IP traffic statistics on a router instead of reloading the router. For added safety, you will see a confirmation prompt when entering this command.	*Configuring IP Services*
	ICMP Unreachable Rate Limiting User Feedback	The ICMP Unreachable Rate Limiting User Feedback feature enables you to clear and display packets that have been discarded because of an unreachable destination, and to configure a threshold interval for triggering error messages. When message logging is generated, it displays on your console.	*Configuring IP Services*
	TCP Application Flags Enhancement	The TCP Applications Flags Enhancement feature enables you to display additional flags with reference to TCP applications. There are two types of flags: status and option. The status flags indicate the status of TCP connections, such as retransmission timeouts, application closed, and synchronized (SYNC) handshakes for listen. The additional flags indicate the state of set options, such as whether or not a virtual private network (VPN) routing and forwarding (VRF) identification is set, whether or not a user is idle, and whether or not a keepalive timer is running.	*Configuring TCP*
	TCP Show Extension	The TCP Show Extension feature introduces the capability to display addresses in IP format instead of hostname format and to display the virtual private network (VPN) routing and forwarding (VRF) table associated with the connection.	*Configuring TCP*
12.3(14)T	WCCP Increased Services	The WCCP Increased Services feature increases the number of services supported by WCCP to a maximum of 256.	*Configuring WCCP*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.3(7)T	TCP Congestion Avoidance	The TCP Congestion Avoidance feature enables the monitoring of acknowledgment packets to the TCP sender when multiple packets are lost in a single window of data. Previously the sender would exit Fast-Recovery mode, wait for three or more duplicate acknowledgment packets before retransmitting the next unacknowledged packet, or wait for the retransmission timer to slow start. This could lead to performance issues.	*Configuring TCP*
	TCP Explicit Congestion Notification	The TCP Explicit Congestion Notification (ECN) feature provides a method for an intermediate router to notify the end hosts of impending network congestion. It also provides enhanced support for TCP sessions associated with applications that are sensitive to delay or packet loss including Telnet, web browsing, and transfer of audio and video data. The benefit of this feature is the reduction of delay and packet loss in data transmissions.	*Configuring TCP*
	WCCP Bypass Counters	The WCCP Bypass Counters feature allows you to display a count of packets that have been bypassed by a web cache and returned to the originating router to be forwarded normally.	*Configuring WCCP*
	WCCP Outbound ACL Check	The WCCP Outbound ACL Check feature enables you to ensure that traffic redirected by WCCP at an input interface is subjected to the outbound ACL checks that may be configured on the output interface prior to redirection. This feature is supported by WCCP Version 1 and Version 2.	*Configuring WCCP*
12.2(8)T	SCTP Release 2	SCTP Release 2 introduced updated output for SCTP commands.	*Stream Control Transmission Protocol*
	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set. In 12.2(8)T, the command that was introduced by this feature was changed from **ip adjust-mss** to **ip tcp adjust-mss**.	*Configuring TCP*
	TCP Window Scaling	The TCP Window Scaling feature adds support for the Window Scaling option in RFC 1323. A larger window size is recommended to improve TCP performance in network paths with large bandwidth, long-delay characteristics that are called Long Fat Networks (LFNs). This TCP Window Scaling enhancement provides that support.	*Configuring TCP*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(4)T	SCTP, Release 1	Stream Control Transmission Protocol (SCTP) is a reliable datagram-oriented IP transport protocol specified by RFC 2960.	*Stream Control Transmission Protocol*
	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set.	*Configuring TCP*
	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set.	*Configuring TCP*
Cisco IOS Releases 12.2			
12.2(21)	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
12.2(15)	UDP Forwarding Support for IP Redundancy Virtual Router Group	User Datagram Protocol (UDP) forwarding is a feature used in Cisco IOS software to forward broadcast and multicast packets received for a specific IP address. Virtual Router Group (VRG) support is currently implemented with the Hot Standby Routing Protocol (HSRP) and it allows a set of routers to be grouped as a logical router that answers to a well known well-known IP address. The UDP Forwarding Support for IP Redundancy Virtual Router Groups feature enables UDP forwarding to be VRG aware, resulting in forwarding only to the active router in the VRG.	*Configuring UDP Forwarding Support for IP Redundancy Virtual Router Groups*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(1)	Active Standby	Active standby enables two IOS SLBs to load-balance the same virtual IP address while at the same time acting as backups for each other.	*Cisco IOS Server Load Balancing*
	Algorithms for Server Load Balancing	IOS SLB provides Weighted Round Robin, Weighted Least Connections and Route Map load-balancing algorithms	*Cisco IOS Server Load Balancing*
	Alternate IP Addresses	IOS SLB enables you to telnet to the load-balancing device using an alternate IP address.	*Cisco IOS Server Load Balancing*
	Audio and Video Load Balancing	IOS SLB can balance RealAudio and RealVideo streams via Real-Time Streaming Protocol (RTSP), for servers running RealNetworks applications.	*Cisco IOS Server Load Balancing*
	Automatic Server Failure Detection	IOS SLB automatically detects each failed TCP connection attempt to a real server, and increments a failure counter for that server. If a server's failure counter exceeds a configurable failure threshold, the server is considered out of service and is removed from the list of active real servers.	*Cisco IOS Server Load Balancing*
	Automatic Unfail	When a real server fails and is removed from the list of active servers, it is assigned no new connections for a length of time specified by a configurable retry timer. After that timer expires, the server is again eligible for new virtual server connections and IOS SLB sends the server the next qualifying connection. If the connection is successful, the failed server is placed back on the list of active real servers. If the connection is unsuccessful, the server remains out of service and the retry timer is reset. The unsuccessful connection must have experienced at least one retry, otherwise the next qualifying connection would also be sent to that failed server.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
	Avoiding Attacks on Server Farms and Firewall Farms	A highly secure site can take certain steps to protect its server farms and firewall farms from attacks.	*Cisco IOS Server Load Balancing*
	Bind ID Support	The bind ID allows a single physical server to be bound to multiple virtual servers and report a different weight for each one. Thus, the single real server is represented as multiple instances of itself, each having a different bind ID. Dynamic Feedback Protocol (DFP) uses the bind ID to identify for which instance of the real server a given weight is specified. The bind ID is needed only if you are using DFP.	*Cisco IOS Server Load Balancing*
	Client-Assigned Load Balancing	Client-assigned load balancing allows you to limit access to a virtual server by specifying the list of client IP subnets that are permitted to use that virtual server. With this feature, you can assign a set of client IP subnets (such as internal subnets) connecting to a virtual IP address to one server farm or firewall farm, and assign another set of clients (such as external clients) to a different server farm or firewall farm.	*Cisco IOS Server Load Balancing*
	Client NAT	If you use more than one load-balancing device in your network, replacing the client IP address with an IP address associated with one of the devices results in proper routing of outbound flows to the correct device. Client NAT also requires that the ephemeral client port be modified since many clients can use the same ephemeral port. Even in cases where multiple load-balancing devices are not used, client NAT can be useful to ensure that packets from load-balanced connections are not routed around the device.	*Cisco IOS Server Load Balancing*
	Content Flow Monitor Support	IOS SLB supports the Cisco Content Flow Monitor (CFM), a web-based status monitoring application within the CiscoWorks2000 product family. You can use CFM to manage Cisco server load-balancing devices. CFM runs on Windows NT and Solaris workstations, and is accessed using a web browser.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
	Delayed Removal of TCP Connection Context	Because of IP packet ordering anomalies, IOS SLB might "see" the termination of a TCP connection (a finish [FIN] or reset [RST]) followed by other packets for the connection. This problem usually occurs when there are multiple paths that the TCP connection packets can follow. To correctly redirect the packets that arrive after the connection is terminated, IOS SLB retains the TCP connection information, or context, for a specified length of time. The length of time the context is retained after the connection is terminated is controlled by a configurable delay timer.	*Cisco IOS Server Load Balancing*
	Dynamic Feedback Protocol for IOS SLB	IOS SLB supports the DFP Agent Subsystem feature, also called global load balancing, which enables client subsystems other than IOS SLB to act as DFP agents. With the DFP Agent Subsystem, you can use multiple DFP agents from different client subsystems at the same time.	*Cisco IOS Server Load Balancing*
	Firewall Load Balancing	As its name implies, firewall load balancing enables IOS SLB to balance flows to firewalls. Firewall load balancing uses a load-balancing device on each side of a group of firewalls (called a firewall farm) to ensure that the traffic for each flow travels to the same firewall, ensuring that the security policy is not compromised.	*Cisco IOS Server Load Balancing*
	IOS SLB, First Release on 12.2	The IOS SLB feature is an IOS-based solution that provides load balancing for a variety of networked devices and services.	*Cisco IOS Server Load Balancing*
	Maximum Connections	IOS SLB allows you to configure maximum connections for server and firewall load balancing.	*Cisco IOS Server Load Balancing*
	Port-Bound Servers	When you define a virtual server, you must specify the TCP or UDP port handled by that virtual server. However, if you configure NAT on the server farm, you can also configure port-bound servers. Port-bound servers allow one virtual server IP address to represent one set of real servers for one service, such as HTTP, and a different set of real servers for another service, such as Telnet.	*Cisco IOS Server Load Balancing*
	Probes: HTTP, Ping, and WSP Probes	IOS SLB probes determine the status of each real server in a server farm and of each firewall in a firewall farm.	*Cisco IOS Server Load Balancing*
	Protocol Support	IOS SLB supports a fixed set of protocols.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
	Server NAT	Server NAT involves replacing the virtual server IP address with the real server IP address (and vice versa).	*Cisco IOS Server Load Balancing*
	Slow Start	In an environment that uses weighted least connections load balancing, a real server that is placed in service initially has no connections, and could therefore be assigned so many new connections that it becomes overloaded. To prevent such an overload, the Slow Start feature controls the number of new connections that are directed to a real server that has just been placed in service.	*Cisco IOS Server Load Balancing*
	Stateful Backup	Stateful backup enables IOS SLB to incrementally backup its load-balancing decisions, or “keep state,” between primary and backup switches. The backup switch keeps its virtual servers in a dormant state until HSRP detects failover; then the backup (now primary) switch begins advertising virtual addresses and processing flows.	*Cisco IOS Server Load Balancing*
	Stateless Backup	Stateless backup provides high network availability by routing IP flows from hosts on Ethernet networks without relying on the availability of a single Layer 3 switch. Stateless backup is particularly useful for hosts that do not support a router discovery protocol (such as the Intermediate System-to-Intermediate System [IS-IS] Interdomain Routing Protocol [IDRP]) and do not have the functionality to shift to a new Layer 3 switch when their selected Layer 3 switch reloads or loses power.	*Cisco IOS Server Load Balancing*
	Sticky Connections	A client transaction can sometimes require multiple consecutive connections, which means new connections from the same client IP address or subnet must be assigned to the same real server. You can use the optional **sticky** command to enable IOS SLB to force connections from the same client to the same load-balanced server within a server farm. For firewall load balancing, the connections between the same client-server pair are assigned to the same firewall.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
	SynGuard	SynGuard limits the rate of TCP start-of-connection packets (SYNchronize sequence numbers, or SYNs) handled by a virtual server to prevent a type of network problem known as a SYN flood denial-of-service attack. A user might send a large number of SYNs to a server, which could overwhelm or crash the server, denying service to other users. SynGuard prevents such an attack from bringing down IOS SLB or a real server. SynGuard monitors the number of SYNs handled by a virtual server at specific intervals and does not allow the number to exceed a configured SYN threshold. If the threshold is reached, any new SYNs are dropped.	*Cisco IOS Server Load Balancing*
	TCP Session Reassignment	IOS SLB tracks each TCP SYN sent to a real server by a client attempting to open a new connection. If several consecutive SYNs are not answered, or if a SYN is replied to with an RST, the TCP session is reassigned to a new real server. The number of SYN attempts is controlled by a configurable reassign threshold.	*Cisco IOS Server Load Balancing*
	Transparent Web Cache Load Balancing	IOS SLB can load-balance HTTP flows across a cluster of transparent web caches. To set up this function, configure the subnet IP addresses served by the transparent web caches, or some common subset of them, as virtual servers. Virtual servers used for transparent web cache load balancing do not answer pings on behalf of the subnet IP addresses, and they do not affect traceroute.	*Cisco IOS Server Load Balancing*
	WAP Load Balancing	The Wireless Application Protocol (WAP) Load Balancing feature allows you to use IOS SLB to load-balance Wireless Session Protocol (WSP) sessions among a group of WAP gateways or servers on an IP bearer network.	*Cisco IOS Server Load Balancing*
12.1(5)T15	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
Cisco IOS XE Release 2			
Cisco IOS XE Release 2.1	Clear IP Traffic CLI	The Clear IP Traffic CLI feature introduced the **clear ip traffic** command to clear all IP traffic statistics on a router instead of reloading the router. For added safety, the user will see a confirmation prompt when entering this command.	*Configuring IP Services*
	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
	TCP Application Flags Enhancement	The TCP Applications Flags Enhancement feature enables the user to display additional flags with reference to TCP applications. There are two types of flags: status and option. The status flags indicate the status of TCP connections; for example, retransmission timeouts, application closed, and synchronized (SYNC) handshakes for listen. The additional flags indicate the state of set options; for example, whether or not a virtual private network (VPN) routing and forwarding (VRF) identification is set, whether or not a user is idle, and whether or not a keepalive timer is running.	*Configuring TCP*
	TCP MIB for RFC 4022 Support	The TCP MIB for RFC 4022 Support feature introduces support for RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*. RFC 4022 is an incremental change of the TCP MIB to improve the manageability of TCP.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs
	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set.	*Configuring TCP*
	TCP Show Extension	The TCP Show Extension feature introduces the capability to display addresses in IP format instead of hostname format and to display the virtual private network (VPN) routing and forwarding (VRF) table associated with the connection.	*Configuring TCP*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
Other Cisco IOS Releases			
12.2(18)ZU2	TCP MSS Adjust	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set.	*Configuring TCP*
12.2(14)ZA5	Exchange Director Features	IOS SLB supports the Exchange Director for the mobile Service Exchange Framework (mSEF) for Cisco 7600 series routers.	*Cisco IOS Server Load Balancing*
	Flow Persistence	Flow persistence provides intelligent return routing of load-balanced IP flows to the appropriate node, without the need for coordinated hash mechanisms on both sides of the load-balanced data path, and without using Network Address Translation (NAT) or proxies to change client or server IP addresses.	*Cisco IOS Server Load Balancing*
	Stateful Backup of Redundant Route Processors	When used with RPR+, IOS SLB supports the stateful backup of redundant route processors for mSEF for Cisco 7600 series routers. This feature enables you to deploy Cisco Multiprocessor WAN Application Modules (MWAMs) in the same chassis as IOS SLB, while maintaining high availability of load-balancing assignments.	*Cisco IOS Server Load Balancing*
12.2(14)ZA4	Automatic Server Failure Detection: Disabling Automatic Server Failure Detection	IOS SLB automatically detects each failed TCP connection attempt to a real server, and increments a failure counter for that server. If a server's failure counter exceeds a configurable failure threshold, the server is considered out of service and is removed from the list of active real servers.	*Cisco IOS Server Load Balancing*

Table 1 Supported IP Application Services Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(14)ZA2	GPRS Load Balancing: Support for GTP v0 and GTP v1	IOS SLB supports both GTP Version 0 (GTP v0) and GTP Version 1 (GTP v1). Support for GTP enables IOS SLB to become "GTP aware," extending IOS SLB's knowledge into Layer 5.	*Cisco IOS Server Load Balancing*
	GPRS Load Balancing with GTP Cause Code Inspection	GPRS load balancing with GTP cause code inspection enabled allows IOS SLB to monitor all PDP context signaling flows to and from GGSN server farms. This feature enables IOS SLB to monitor GTP failure cause codes, detecting system-level problems in both Cisco and non-Cisco GGSNs.	*Cisco IOS Server Load Balancing*
	Home Agent Director	The Home Agent Director load balances Mobile IP Registration Requests (RRQs) among a set of home agents (configured as real servers in a server farm). Home agents are the anchoring points for mobile nodes. Home agents route flows for a mobile node to its current foreign agent (point of attachment).	*Cisco IOS Server Load Balancing*
	Probes: Custom UDP Probes	IOS SLB probes determine the status of each real server in a server farm and of each firewall in a firewall farm.	*Cisco IOS Server Load Balancing*
12.1(27b)E1	IP Precedence Accounting	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.	*Configuring IP Services*
10.0	Flooding Packets Using Spanning-Tree	Enables the forwarding of UDP broadcast packets using the spanning-tree forwarding table.	*Configuring IPv4 Broadcast Packet Handling*
	IP Directed Broadcasts	Enables the translation of a directed broadcast to physical broadcasts.	*Configuring IPv4 Broadcast Packet Handling*
	Specifying an IP Broadcast Address	Specifies the IP broadcast address for an interface.	*Configuring IPv4 Broadcast Packet Handling*
	UDP Broadcast Packet Forwarding	Enables the forwarding of UDP broadcast packets.	*Configuring IPv4 Broadcast Packet Handling*

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring Server Load Balancing

First Published: January 14, 2008
Last Updated: June 27, 2008

This document describes how to configure the IOS Server Load Balancing (IOS SLB) feature. For a complete description of the IOS SLB commands in this chapter, refer to the "Server Load Balancing Commands" chapter of the *Cisco IOS IP Application Services Command Reference.* To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

The SLB feature is a Cisco IOS-based solution that provides IP server load balancing. Using the IOS SLB feature, the network administrator defines a *virtual* server that represents a group of *real* servers in a cluster of network servers known as a *server farm*. In this environment the clients are configured to connect to the IP address of the virtual server. The virtual server IP address is configured as a loopback address, or secondary IP address, on each of the real servers. When a client initiates a connection to the virtual server, the IOS SLB function chooses a real server for the connection based on a configured load-balancing algorithm.

The IOS SLB feature provides load balancing for a variety of networked devices and services, including:

- Application servers, such as Hypertext Transfer Protocol (HTTP), Telnet, File Transfer Protocol (FTP), and so on
- Firewalls
- Service nodes, such as authentication, authorization, and accounting (AAA) servers, web caches, and so on

In addition, the IOS SLB Exchange Director enables advanced load-balancing routing capabilities for the following additional service nodes:

- mobile Service Exchange Framework (mSEF) components:
  - Cisco Content Services Gateways (CSGs)

    If you are running with Supervisor Engine 32 (SUP32-MSFC2A), CSG Release 3.1(3)C7(1) or later is required.
  - Cisco gateway GPRS support nodes (GGSNs)
  - Cisco Service Selection Gateways (SSGs)
  - Cisco Home Agents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

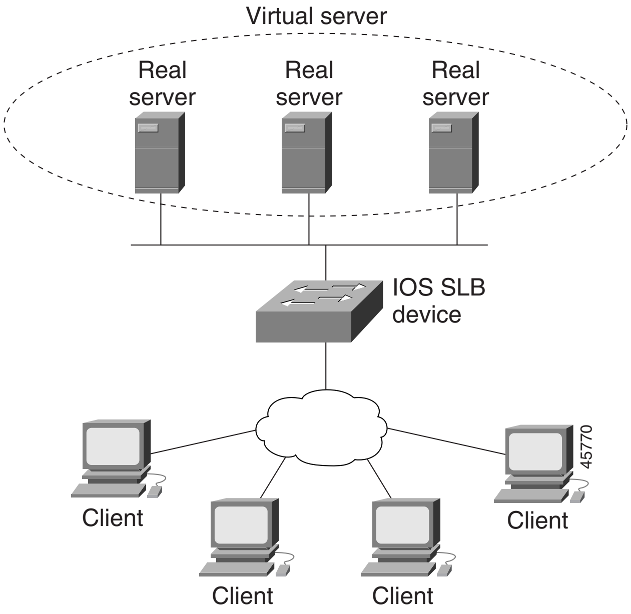
- Other components for mobile, Public Wireless LAN (PWLAN), and Service Provider networks:
  - Wireless Application Protocol (WAP) gateways
  - Protocol optimization gateways
  - Non-Cisco GGSNs and Home Agents
  - Other RADIUS-aware flow gateways. These gateways are proxies or routing nodes that receive RADIUS Authorization and Accounting requests for users that route flows through the gateways. The Exchange Director binds the RADIUS and data flows to the same gateway, ensuring that the gateway receives a complete and consistent view of the network activity for the user.

The Exchange Director also adds the following features:

- Enhanced failover capabilities for single-chassis failover within the mSEF for Catalyst 6500 family switches and Cisco 7600 series routers. When used with Route Processor Redundancy Plus (RPR+), IOS SLB stateful backup for redundant route processors provides full IOS SLB stateful failover for these platforms.
- Flow persistence, which provides intelligent return routing of load-balanced IP flows.

Figure 1 illustrates a logical view of a simple IOS SLB network.

Figure 1 Logical View of IOS SLB



Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for IOS SLB" section on page 176.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Restrictions for IOS SLB

General Restrictions

- **Does not support load balancing of flows between clients and real servers that are on the same local-area network (LAN) or virtual LAN (VLAN). The packets being load-balanced cannot enter and leave the load-balancing device on the same interface.**
- You cannot configure IOS SLB from different user sessions at the same time.
- Do not configure an IOS SLB virtual IP address on the same subnet as any real server IP address, unless all server farms that include the real server IP address are configured with **nat server**.
- Operates in a standalone mode and currently does not operate as a MultiNode Load Balancing (MNLB) Services Manager. Does not support IOS SLB and MNLB configured with the same virtual IP address, even if they are for different services. The presence of IOS SLB does not preclude the use of the existing MNLB Forwarding Agent with an external Services Manager (such as the LocalDirector) in an MNLB environment. (MNLB is sometimes called Cisco Application Services Architecture, or CASA.)
- Does not support coordinating server load-balancing statistics among different IOS SLB instances for backup capability.
- Supports FTP and firewall load balancing only in dispatched mode.
- Does not support Dynamic Host Configuration Protocol (DHCP) load balancing.
- Does not support Internet Protocol version 6 (IPv6).
- When operating in dispatched mode, real servers must be Layer 2-adjacent, tag-switched, or via GRE tunnel.

  When operating in directed mode with server NAT, real servers need not be Layer 2-adjacent to IOS SLB. This function allows for more flexible network design, since servers can be placed several Layer 3 hops away from the IOS SLB switch.
- When operating in directed mode as a member of a multicast group, IOS SLB can receive multicast flows but cannot send multicast flows. This is not a restriction when operating in dispatched mode.
- Supports client NAT and server port translation for TCP and UDP virtual servers only.

- When balancing streams to a virtual IP address that is the same as one of the IOS interface IP addresses (loopback, Ethernet, and so on), IOS SLB treats all UDP packets to that address as traceroute packets and replies with “host unreachable” ICMP packets. This occurs even if the IOS listens to the target UDP port. To avoid this issue, configure the virtual server as a network (address/31), not as a host (address/32).
- Do not use the virtual IP address configured in the IOS SLB virtual server for UDP-based router management applications such as SNMP. Doing so can result in high CPU usage. (This is not a problem for a UDP virtual server that is configured with destination port number 0.)
- The DFP agent requires a delay between hello messages of at least 3 seconds. Therefore, if your DFP manager provides a timeout specification, you must set the timeout to at least 3 seconds.
- When both IOS SLB and the Web Cache Communication Protocol (WCCP) are configured on a Catalyst 6500 family switch, and WCCP Input Redirection is configured with IOS SLB, Layer 2 WCCP forwarding must be used between the router and the cache. In this case, WCCP and IOS SLB both run in hardware and are processed in the correct order. If Generic Routing Encapsulation (GRE) forwarding is used, then IOS SLB takes precedence over WCCP and there is no redirection, because GRE forwarding is done on the MSFC. Note that the WCCP forwarding method, either Layer 2 or GRE, is configured on the cache engine and not on the switch.
- Do not configure IOS SLB and a Cisco Service Selection Gateway (SSG) on the same device.
- For “sandwich” configurations, if a flow is to be directed through two IOS SLB instances (virtual servers or firewall farms), the IOS SLB instances must reside in different Virtual Private Network (VPN) routing and forwardings (VRFs).
- Policy-based routing (PBR) and Virtual Private Network (VPN) routing and forwarding (VRF) are mutually exclusive and cannot be configured on the same interface.

Static NAT

- Does not work with client NAT server farms. That is, if a real server is using a given virtual IP address for server NAT, and a server farm is associated with that same virtual IP address, then you cannot configure the server farm to use client NAT.
- Requires that each real server be associated with only one virtual server, to ensure that IOS SLB can create connections correctly.
- Requires a 0-port virtual server.
- Does not support virtual service FTP.

Backup Server Farm Support

- Does not support defining the same real server in both primary and backup server farms.
- Requires the same NAT configuration (none, client, server, or both) for both primary and backup server farms. In addition, if NAT is specified, both server farms must use the same NAT pool.
- Does not support HTTP redirect load balancing. If a primary server farm specifies a redirect virtual server, you cannot define that primary as a backup, nor can you define a backup for that primary.

Firewall Load Balancing

- Is no longer limited to a single firewall farm in each load-balancing device.
- Requires that each firewall must have its own unique MAC address and must be Layer 2-adjacent to each device. The firewalls can be connected to individual interfaces on the device, or they can all share a VLAN and connect using a single interface.
- Requires Ethernet between each firewall load-balancing device and each firewall.

- On each firewall load-balancing device, requires that each Layer 2 firewall be connected to a single Layer 3 (IP) interface.
- Flows with a destination IP address on the same subnet as the configured firewall IP addresses are not load-balanced. (Such flows could be a firewall console session or other flows on the firewall LAN.)
- Does not support the following IOS SLB functions:
  - Network Address Translation (NAT)
  - Port-bound servers
  - SynGuard
  - TCP session reassignment
  - Transparent web cache load balancing

GPRS Load Balancing *Without* GTP Cause Code Inspection Enabled

- If a real server is defined in two or more server farms, each server farm must be associated with a different virtual server.
- Operates in either dispatched or directed server NAT mode only.
- Supports stateful backup only if sticky connections are enabled.
- Does not load-balance network-initiated PDP context requests.
- Does not support the following IOS SLB functions:
  - Bind IDs
  - Client-assigned load balancing
  - Slow start
  - Weighted least connections load-balancing algorithm

GPRS Load Balancing *With* GTP Cause Code Inspection Enabled

- If a real server is defined in two or more server farms, each server farm must be associated with a different virtual server.
- Operates in directed server NAT mode only.
- Cannot load-balance network-initiated PDP context requests.
- Requires inbound and outbound signaling to flow through IOS SLB.
- Requires either the SGSN or the GGSN to echo its peer.
- Does not support the following IOS SLB functions:
  - Bind IDs
  - Client-assigned load balancing
  - Slow start

VPN Server Load Balancing

- Does not support Internet Control Message Protocol (ICMP) and wildcard (0-protocol) virtual servers.

RADIUS Load Balancing Accelerated Data Plane Forwarding

- Requires the route map algorithm.
- Requires redundant CSGs for best results.
- Requires static provisioning of load distribution by subscriber address range.
- Supports only simple IP access control lists (ACLs).
- When VSA correlation used, IOS SLB maintains the correlation information only in the active RADIUS load-balancing device, not in the backup RADIUS load-balancing device. The backup RADIUS load-balancing device does not receive VSA correlation information from the active RADIUS load-balancing device.
- All Accounting-Request and Access-Accept messages must include the RADIUS-assigned Framed-ip-address attribute. The source IP address for each subscriber flow must also match the value of the Framed-ip-address attribute in the Access-Accept message.
- RADIUS accounting must be enabled on the RADIUS client, which is typically a Network Access Server (NAS).
- When you specify the **predictor route-map** command in SLB server farm configuration mode, no further commands in SLB server farm configuration mode or real server configuration mode are allowed.

VSA Correlation

- VSA correlation might result in a degradation of performance.
- IOS SLB maintains the correlation information only in the active RADIUS load-balancing device, not in the backup RADIUS load-balancing device. The backup RADIUS load-balancing device does not receive VSA correlation information from the active RADIUS load-balancing device.
- The Cisco VSA is injected into the RADIUS Accounting-Start packet. The Cisco VSA is not injected into any other RADIUS messages or packets, such as interim RADIUS Accounting ON or OFF messages or RADIUS Accounting-Stop packets.
- You cannot configure **radius inject acct** commands and **radius inject auth** commands on the same virtual server.

RADIUS Load Balancing for GPRS

- Requires the weighted round robin algorithm.
- Does not support fragmented RADIUS packets.
- All Accounting-Request and Access-Accept messages must include the RADIUS-assigned Framed-ip-address attribute. The source IP address for each subscriber flow must also match the value of the Framed-ip-address attribute in the Access-Accept message.
- RADIUS accounting must be enabled on the RADIUS client, which is typically a Network Access Server (NAS).

RADIUS Load Balancing for CDMA2000

- Requires the weighted round robin algorithm.
- Does not support fragmented RADIUS packets.
- All subscribers on the mobile network must be assigned a unique IP address (that is, no overlapping IP addresses) which can be routed in the mobile wireless network.
- Each User-Name attribute must correspond to a single subscriber, or at most to a very small number of subscribers. Otherwise, a single SSG might be burdened with an unexpectedly large load.

- For simple IP networks, the following additional restrictions apply:
  - The PDSN must include the User-Name attribute in all RADIUS Access-Request and Accounting-Start packets. The value of the User-Name attribute for a given subscriber must be the same in all the packets (except for Cisco PDSNs that provide MSID-based access).
  - The PDSN must include the Framed-ip-address attribute and the NAS-ip-address in all RADIUS Accounting-Start and Accounting-Stop packets. The value of the Framed-ip-address attribute must equal the source IP address in subscriber data packets routed by RADIUS load balancing for SSG service.
  - The PDSN must include the NAS-ip-address in all Accounting-Requests. For BSC/PCF hand-offs, the Accounting-Stop must include the 3GPP2-Session-Continue VSA with a value of **1**, to prevent the destruction of RADIUS load balancing sticky database objects for the subscriber.
- For Mobile IP networks, the following additional restrictions apply:
  - For a given subscriber session, the PDSN and HA must send the RADIUS Access-Request and Accounting-Start packets with the User-Name attribute. The value of the User-Name attribute in all PDSN and HA RADIUS packets must be the same for the session.
  - For a given subscriber session, the PDSN and HA must send RADIUS Accounting-Request packets with a Framed-ip-address attribute equal to the source IP address in subscriber data packets routed by RADIUS load balancing for SSG service. All RADIUS Accounting-Requests sent by the PDSN and HA must also include the NAS-ip-address attribute.
  - The PDSN must include the 3GPP2-Correlation-Identifier attribute in all Accounting-Requests.

Home Agent Director

- A Registration Request (RRQ) must include the network access identifier (NAI) in order to be load-balanced.
- An RRQ must include a home agent IP address of either 0.0.0.0 or 255.255.255.255 in order to be load-balanced.
- For fast switching, the NAI in the RRQ cannot be more than 96 bytes deep in the packet. If the NAI is deeper than 96 bytes, IOS SLB handles the packet at the process level.
- Operates in either dispatched or directed server NAT mode only.
- Does not support the following IOS SLB functions:
  - Bind IDs
  - Client-assigned load balancing
  - Slow start
  - Stateful backup
  - Sticky connections
  - Weighted least connections load-balancing algorithm

HTTP Probes

- HTTP probes do not support HTTP over Secure Socket Layer (HTTPS). That is, you cannot send an HTTP probe to an SSL server.

UDP Probes

- UDP probes do not support fragmented Response packets.
- UDP probes do not support hosts that require a particular source port value in probe packets. UDP probes select an ephemeral port for each probe.
- Protocols and applications that have Message Digest Algorithm Version 5 (MD5) checksums generated from payload must be captured by a “sniffer” to obtain a correct checksum.
- For Cisco IOS Multiprotocol Label Switching (MPLS):
  - Clients can connect to IOS SLB via the MPLS cloud in a Supervisor Engine 720 environment.
  - The MPLS client interface must be configured with Tunnel Engineering. No other MPLS configuration is supported.
  - The MPLS client interface must receive packets as IP packets.
  - The MPLS client interface must be behind a Penultimate Hop Popping (PHP) router.
- For Catalyst 6500 family switches and Cisco 7600 series routers:
  - Supports Native IOS only (c6sup images). Native IOS requires the MSFC and the Policy Feature Card (PFC). When running redundant MSFCs in the same Catalyst 6500 family switch, stateful backup between the two MSFCs is not supported, but stateless backup between the two MSFCs is supported.

    The term “MSFC” refers to an MSFC1, MSFC2, or MSFC3, except when specifically differentiated.

    The term “PFC” refers to a PFC1, PFC2, or PFC3, except when specifically differentiated.
  - Requires that the Multilayer Switching (MLS) flow mode operate in full-flow mode or in interface full-flow mode. IOS SLB automatically sets the flow mode for its own use. For more information about how to set the MLS flow, refer to the *Catalyst 6000 Family IOS Software Configuration Guide*.
  - When operating in dispatched mode, real servers must be Layer 2-adjacent to IOS SLB (that is, not beyond an additional router), with hardware data packet acceleration performed by the PFC. All real servers in the same server farm must be on the same VLAN. The loopback address must be configured in the real servers.
  - Requires that all real servers in a firewall farm be on the same VLAN. Real servers in different firewall farms can be on different VLANs.
  - Provides no hardware data packet acceleration in directed mode. (Hardware data packet acceleration is performed by the PFC, and in directed mode the packets are handled by the MSFC, not the PFC.)
  - For Supervisor Engine 2, “sandwich” configurations that require firewall load balancing are not supported, because such configurations require VRF, and VRF is not supported for Supervisor Engine 2.

Access Service Network (ASN) R6 Load Balancing

- Operates in either dispatched or directed server NAT mode only. In directed mode, IOS SLB changes the destination IP address of the Mobile Station (MS) Pre-Attachment request to that of the selected ASN gateway real server.
- Requires DFP

- Does not support the following features:
  - Client NAT
  - Stateful redundancy
  - Sticky connections
  - Weighted least connections algorithm (for MS Pre-Attachment requests)
- When the base station is configured to send MS Pre-Attachment ACK packets directly to an ASN gateway, bypassing IOS SLB, you must ensure that the session can time out without failing the real server. To do so, configure the **no faildetect inband** command real server configuration mode.

Information About IOS SLB

To configure IOS SLB, you should understand the following concepts:

- Benefits of IOS SLB, page 9
- General IOS SLB Features, page 10—This section describes the general features provided by IOS SLB.
- Exchange Director Features, page 28—This section describes the specific features provided by the Exchange Director for mobile Service Exchange Framework (mSEF).

Note Some IOS SLB features are specific to a single platform and are not described in this feature document. For information about those features, refer to the appropriate platform-specific documentation.

Benefits of IOS SLB

IOS SLB shares the same software code base as Cisco IOS and has all the software features sets of Cisco IOS software. IOS SLB is recommended for customers desiring complete integration of SLB technology into traditional Cisco switches and routers.

On Cisco Catalyst 6500 family switches, IOS SLB takes advantage of hardware acceleration to forward packets at very high speed when running in dispatched mode.

IOS SLB assures continuous, high availability of content and applications with proven techniques for actively managing servers and connections in a distributed environment. By distributing user requests across a cluster of servers, IOS SLB optimizes responsiveness and system capacity, and dramatically reduces the cost of providing Internet, database, and application services for large-, medium-, and small-scale sites.

IOS SLB facilitates scalability, availability, and ease of maintenance:

- The addition of new physical (real) servers, and the removal or failure of existing servers, can occur at any time, transparently, without affecting the availability of the virtual server.
- IOS SLB's slow start capability allows a new server to increase its load gradually, preventing failures caused by assigning the server too many new connections too quickly.
- IOS SLB supports fragmented packets and packets with IP options, buffering your servers from client or network vagaries that are beyond your control.
- IOS SLB firewall load balancing enables you to scale access to your Internet site. You can add firewalls without affecting existing connections, enabling your site to grow without impacting customers.

Using DFP enables IOS SLB to provide weights to another load-balancing system. IOS SLB can act as a DFP manager, receiving weights from host servers, and it can act as a DFP agent, sending weights to a DFP manager. The functions are enabled independently—you can implement either one, or both, at the same time.

Administration of server applications is easier. Clients know only about virtual servers; no administration is required for real server changes.

Security of the real server is provided because its address is never announced to the external network. Users are familiar only with the virtual IP address. You can filter unwanted flows based on both IP address and TCP or UDP port numbers. Additionally, though it does not eliminate the need for a firewall, IOS SLB can help protect against some denial-of-service attacks.

In a branch office, IOS SLB allows balancing of multiple sites and disaster recovery in the event of full-site failure, and distributes the work of load balancing.

General IOS SLB Features

IOS SLB provides the following features:

- Routing Features, page 10
- Security Features, page 20
- Server Failure Detection and Recovery Features, page 21
- Protocol Support Features, page 26
- Redundancy Features, page 27

Routing Features

IOS SLB provides the following routing features:

- Algorithms for Server Load Balancing, page 11
- Bind ID Support, page 13
- Client-Assigned Load Balancing, page 13
- Connection Rate Limiting, page 13
- Content Flow Monitor Support, page 13
- Delayed Removal of TCP Connection Context, page 13
- Firewall Load Balancing, page 13
- GTP IMSI Sticky Database, page 14
- Home Agent Director, page 14
- Interface Awareness, page 15
- Maximum Connections, page 15
- Multiple Firewall Farm Support, page 15
- Network Address Translation (NAT), page 15
- Port-Bound Servers, page 19
- Route Health Injection, page 19
- Sticky Connections, page 19

- TCP Session Reassignment, page 20
- Transparent Web Cache Load Balancing, page 20

Algorithms for Server Load Balancing

IOS SLB provides the following load-balancing algorithms:

- Weighted Round Robin, page 11
- Weighted Least Connections, page 12
- Route Map, page 12

You can specify one of these algorithms as the basis for choosing a real server for each new connection request that arrives at the virtual server.

For each algorithm, connections in closing state continue to be counted against the number of connections assigned to a real server. This impacts the least connections algorithm more than the other algorithms, because the least connections algorithm is influenced by the number of connections impacts the least connections more. IOS SLB adjusts the number of connections per real server, and the algorithm metrics, each time a connection is assigned.

Weighted Round Robin

The weighted round robin algorithm specifies that the real server used for a new connection to the virtual server is chosen from the server farm in a circular fashion. Each real server is assigned a weight, *n*, that represents its capacity to handle connections, as compared to the other real servers associated with the virtual server. That is, new connections are assigned to a given real server *n* times before the next real server in the server farm is chosen.

For example, assume a server farm comprised of real server ServerA with *n* = 3, ServerB with *n* = 1, and ServerC with *n* = 2. The first three connections to the virtual server are assigned to ServerA, the fourth connection to ServerB, and the fifth and sixth connections to ServerC.

Note Assigning a weight of *n*=1 to all of the servers in the server farm configures the IOS SLB device to use a simple round robin algorithm.

General packet radio service (GPRS) load balancing *without* GPRS Tunneling Protocol (GTP) cause code inspection enabled requires the weighted round robin algorithm. A server farm that uses weighted least connections can be bound to a virtual server providing GPRS load balancing without GTP cause code inspection enabled, but you cannot place the virtual server INSERVICE. If you try to do so, IOS SLB issues an error message.

The Home Agent Director requires the weighted round robin algorithm. A server farm that uses weighted least connections can be bound to a Home Agent Director virtual server, but you cannot place the virtual server INSERVICE. If you try to do so, IOS SLB issues an error message.

RADIUS load balancing requires the weighted round robin algorithm.

RADIUS load balancing accelerated data plane forwarding *does not* support the weighted round robin algorithm.

Weighted Least Connections

The weighted least connections algorithm specifies that the next real server chosen from a server farm for a new connection to the virtual server is the server with the fewest active connections. Each real server is assigned a weight for this algorithm, also. When weights are assigned, the server with the fewest connections is based on the number of active connections on each server, and on the relative capacity of each server. The capacity of a given real server is calculated as the assigned weight of that server divided by the sum of the assigned weights of all of the real servers associated with that virtual server, or $n_1/(n_1+n_2+n_3...)$.

For example, assume a server farm comprised of real server ServerA with *n* = 3, ServerB with *n* = 1, and ServerC with *n* = 2. ServerA would have a calculated capacity of 3/(3+1+2), or half of all active connections on the virtual server, ServerB one-sixth of all active connections, and ServerC one-third of all active connections. At any point in time, the next connection to the virtual server would be assigned to the real server whose number of active connections is farthest below its calculated capacity.

Note Assigning a weight of *n*=1 to all of the servers in the server farm configures the IOS SLB device to use a simple least-connection algorithm.

GPRS load balancing *without* GTP cause code inspection enabled *does not* support the weighted least connections algorithm.

GPRS load balancing *with* GTP cause code inspection enabled *does* support the weighted least connections algorithm.

Access Service Network (ASN) R6 load balancing (for MS Pre-Attachment requests), the Home Agent Director, RADIUS load balancing, and RADIUS load balancing accelerated data plane forwarding *do not* support the weighted least connections algorithm.

Route Map

The route map algorithm is valid only with IOS SLB RADIUS load balancing accelerated data plane forwarding, also known as Turbo RADIUS load balancing. Turbo RADIUS load balancing is a high-performance solution that uses policy-based routing (PBR) route maps to handle subscriber data-plane traffic in a Cisco Content Services Gateway (CSG) environment. When Turbo RADIUS load balancing receives a RADIUS payload, it inspects the payload, extracts the framed-IP attribute, applies a route map to the IP address, and then determines which CSG is to handle the subscriber.

For more information about policy-based routing, including how it works, when to use it, how to configure it, and how to enable it, see the “Policy-Based Routing” and “Configuring Policy-Based Routing” sections of the *Cisco IOS IP Routing Configuration Guide*.

Note RADIUS load balancing accelerated data plane forwarding requires the route map algorithm.

Policy-based routing (PBR) and Virtual Private Network (VPN) routing and forwarding (VRF) are mutually exclusive and cannot be configured on the same interface.

Bind ID Support

The bind ID allows a single physical server to be bound to multiple virtual servers and report a different weight for each one. Thus, the single real server is represented as multiple instances of itself, each having a different bind ID. Dynamic Feedback Protocol (DFP) uses the bind ID to identify for which instance of the real server a given weight is specified. The bind ID is needed only if you are using DFP.

GPRS load balancing and the Home Agent Director do not support bind IDs.

Client-Assigned Load Balancing

Client-assigned load balancing allows you to limit access to a virtual server by specifying the list of client IP subnets that are permitted to use that virtual server. With this feature, you can assign a set of client IP subnets (such as internal subnets) connecting to a virtual IP address to one server farm or firewall farm, and assign another set of clients (such as external clients) to a different server farm or firewall farm.

GPRS load balancing and the Home Agent Director do not support client-assigned load balancing.

Connection Rate Limiting

IOS SLB enables you to specify the maximum connection rate allowed for a real server in a server farm. For more information, see the description of the **rate** command in real server configuration mode.

Content Flow Monitor Support

IOS SLB supports the Cisco Content Flow Monitor (CFM), a web-based status monitoring application within the CiscoWorks2000 product family. You can use CFM to manage Cisco server load-balancing devices. CFM runs on Windows NT and Solaris workstations, and is accessed using a web browser.

Delayed Removal of TCP Connection Context

Because of IP packet ordering anomalies, IOS SLB might “see” the termination of a TCP connection (a finish [FIN] or reset [RST]) followed by other packets for the connection. This problem usually occurs when there are multiple paths that the TCP connection packets can follow. To correctly redirect the packets that arrive after the connection is terminated, IOS SLB retains the TCP connection information, or context, for a specified length of time. The length of time the context is retained after the connection is terminated is controlled by a configurable delay timer.

Firewall Load Balancing

As its name implies, firewall load balancing enables IOS SLB to balance flows to firewalls. Firewall load balancing uses a load-balancing device on each side of a group of firewalls (called a firewall farm) to ensure that the traffic for each flow travels to the same firewall, ensuring that the security policy is not compromised.

You can configure more than one firewall farm in each load-balancing device.

Layer 3 firewalls, which have ip-addressable interfaces, are supported by IOS SLB firewall load balancing if they are subnet-adjacent to the firewall load-balancing device and have unique MAC addresses. The device does not modify the IP addresses in the user packet. To send the packet to the chosen firewall, the device determines which interface to use and changes the Layer 2 headers accordingly. This type of routing is the standard dispatched routing used by IOS SLB.

Layer 2 firewalls, which do not have IP addresses, are transparent to IOS SLB firewall load balancing. IOS SLB supports Layer 2 firewalls by placing them between two ip-addressable interfaces.

Whereas many Layer 3 firewalls might exist off a single Layer 3 interface on the load-balancing device (for example, a single LAN), only one Layer 2 firewall can exist off each interface.

When configuring the load-balancing device, you configure a Layer 3 firewall using its IP address, and a Layer 2 firewall using the IP address of the interface of the device on the "other side" of the firewall.

To balance flows across the firewalls in a firewall farm, IOS SLB firewall load balancing performs a route lookup on each incoming flow, examining the source and destination IP addresses (and optionally the source and destination TCP or User Datagram Protocol [UDP] port numbers). Firewall load balancing applies a hash algorithm to the results of the route lookup and selects the best firewall to handle the connection request.

Note IOS SLB firewall load balancing *must* examine incoming packets and perform route lookup. On Catalyst 6500 Family Switches, some additional packets might need to be examined. Firewall load balancing impacts internal (secure) side routing performance and must be considered in the complete design.

To maximize availability and resilience in a network with multiple firewalls, configure a separate equal-weight route to each firewall, rather than a single route to only one of the firewalls.

IOS SLB firewall load balancing provides the following capabilities:

- Connections initiated from either side of the firewall farm are load-balanced.
- The load is balanced among a set of firewalls—the firewall farm.
- All packets for a connection travel through the same firewall. Subsequent connections can be "sticky," ensuring that they are assigned to the same firewall.
- Source-IP, destination-IP, and source-destination-IP sticky connections are supported.
- Probes are used to detect and recover from firewall failures.
- Redundancy is provided. Hot Standby Router Protocol (HSRP), stateless backup, and stateful backup are all supported.
- Multiple interface types and routing protocols are supported, enabling the external (Internet side) load-balancing device to act as an access router.
- Proxy firewalls are supported.

GTP IMSI Sticky Database

IOS SLB can select a gateway general packet radio service (GPRS) support node (GGSN) for a given International Mobile Subscriber ID (IMSI), and forward all subsequent Packet Data Protocol (PDP) create requests from the same IMSI to the selected GGSN.

To enable this feature, IOS SLB uses a GPRS Tunneling Protocol (GTP) IMSI sticky database, which maps each IMSI to its corresponding real server, in addition to its session database.

IOS SLB creates a sticky database object when it processes the first GTP PDP create request for a given IMSI. IOS SLB removes the sticky object when it receives a notification to do so from the real server, or as a result of inactivity. When the last PDP belonging to an IMSI is deleted on the GGSN, the GGSN notifies IOS SLB to remove the sticky object.

Home Agent Director

The Home Agent Director load balances Mobile IP Registration Requests (RRQs) among a set of home agents (configured as real servers in a server farm). Home agents are the anchoring points for mobile nodes. Home agents route flows for a mobile node to its current foreign agent (point of attachment).

The Home Agent Director has the following characteristics:

- Can operate in dispatched mode or in directed server NAT mode, but not in directed client NAT mode. In dispatched mode, the home agents must be Layer 2-adjacent to the IOS SLB device.
- Does not support stateful backup. See the "Stateful Backup" section on page 28 for more information.
- Delivers RRQs destined to the virtual Home Agent Director IP address to one of the real home agents, using the weighted round robin load-balancing algorithm. See the "Weighted Round Robin" section on page 11 for more information about this algorithm.
- Requires DFP in order to allocate RRQs based on capacity.

For more information about Mobile IP, home agents, and related topics, refer to the *Cisco IOS IP Mobility Configuration Guide*.

Interface Awareness

Some environments require IOS SLB on both sides of a farm of CSGs, SSGs, or firewalls. For example, you might want IOS SLB to perform RADIUS load balancing on one side of a farm and firewall load balancing on the other, or firewall load balancing on both sides of a firewall farm.

Such "sandwich" environments require IOS SLB to take into account the input interface when mapping packets to virtual servers, firewall farms, connections, and sessions. In IOS SLB, this function is called interface awareness. When interface awareness is configured, IOS SLB processes only traffic arriving on configured access interfaces. (An access interface is any Layer 3 interface.)

Maximum Connections

IOS SLB allows you to configure maximum connections for server and firewall load balancing.

- For server load balancing, you can configure a limit on the number of active connections that a real server is assigned. If the maximum number of connections is reached for a real server, IOS SLB automatically switches all further connection requests to other servers until the connection number drops below the specified limit.
- For firewall load balancing, you can configure a limit on the number of active TCP or UDP connections that a firewall farm is assigned. If the maximum number of connections is reached for the firewall farm, new connections are dropped until the connection number drops below the specified limit.

Multiple Firewall Farm Support

You can configure more than one firewall farm in each load-balancing device.

Network Address Translation (NAT)

Cisco IOS NAT, RFC 1631, allows unregistered "private" IP addresses to connect to the Internet by translating them into globally registered IP addresses. As part of this functionality, Cisco IOS NAT can be configured to advertise only one address for the entire network to the outside world. This configuration provides additional security and network privacy, effectively hiding the entire internal network from the world behind that address. NAT has the dual functionality of security and address conservation, and is typically implemented in remote access environments.

This section includes information about the following topics:

- Session Redirection, page 16
- Dispatched Mode, page 16

- Directed Mode, page 16
- Server NAT, page 17
- Client NAT, page 17
- Static NAT, page 17
- Server Port Translation, page 19

Session Redirection

Session redirection involves redirecting packets to real servers. IOS SLB can operate in one of two session redirection modes, dispatched mode or directed mode.

Note In both dispatched and directed modes, IOS SLB must track connections. Therefore, you must design your network so that there is no alternate network path from the real servers to the client that bypasses the load-balancing device.

Dispatched Mode

In dispatched mode, the virtual server address is known to the real servers; you must configure the virtual server IP address as a loopback address, or secondary IP address, on each of the real servers. IOS SLB redirects packets to the real servers at the media access control (MAC) layer. Since the virtual server IP address is not modified in dispatched mode, the real servers must be Layer 2-adjacent to IOS SLB, or intervening routers might not be able to route to the chosen real server.

For Catalyst 6500 family switches, dispatched mode with hardware data packet acceleration generally yields better performance than directed mode.

Refer to the "Configuring Virtual Interfaces" chapter of the *Cisco IOS Interface Configuration Guide* for more information about configuring the loopback address.

Note Some UDP applications cannot respond from the loopback interface. If that situation occurs, you must use directed mode.

Directed Mode

In directed mode, the virtual server can be assigned an IP address that is not known to any of the real servers. IOS SLB translates packets exchanged between a client and a real server, using NAT to translate the virtual server IP address to a real server IP address.

IOS SLB supports the following types of NAT:

- "Server NAT" section on page 17
- "Client NAT" section on page 17
- "Static NAT" section on page 17
- "Server Port Translation" section on page 19

Note You can use both server NAT and client NAT for the same connection.

IOS SLB does not support FTP or firewall load balancing in directed mode. Therefore, FTP and firewall load balancing cannot use NAT.

IOS SLB supports only client NAT for TCP and UDP virtual servers.

IOS SLB supports only server NAT (but not server port translation) for Encapsulation Security Payload (ESP) virtual servers or Generic Routing Encapsulation (GRE) virtual servers.

Server NAT

Server NAT involves replacing the virtual server IP address with the real server IP address (and vice versa). Server NAT provides the following benefits:

- Servers can be many hops away from the load-balancing device.
- Intervening routers can route to them without requiring tunnelling.
- Loopback and secondary interfaces are not required on the real server.
- The real server need not be Layer 2-adjacent to IOS SLB.
- The real server can initiate a connection to a virtual server on the same IOS SLB device.

Client NAT

If you use more than one load-balancing device in your network, replacing the client IP address with an IP address associated with one of the devices results in proper routing of outbound flows to the correct device. Client NAT also requires that the ephemeral client port be modified since many clients can use the same ephemeral port. Even in cases where multiple load-balancing devices are not used, client NAT can be useful to ensure that packets from load-balanced connections are not routed around the device.

Static NAT

With static NAT, address translations exist in the NAT translation table as soon as you configure static NAT commands, and they remain in the translation table until you delete the static NAT commands.

You can use static NAT to allow some users to utilize NAT and allow other users on the same Ethernet interface to continue with their own IP addresses. This option enables you to provide a default NAT behavior for real servers, differentiating between responses from a real server, and connection requests initiated by the real server.

For example, you can use server NAT to redirect Domain Name System (DNS) inbound request packets and outbound response packets for a real server, and static NAT to process connection requests from that real server.

Note Static NAT is not required for DNS, but it is recommended, because it hides your real server IP addresses from the outside world.

IOS SLB supports the following static NAT options, configured using the **ip slb static** command:

- Static NAT with dropped connections—The real server is configured to have its packets dropped by IOS SLB, if the packets do not correspond to existing connections. This option is usually used in conjunction with the subnet mask or port number option on the **real** command in static NAT configuration mode, such that IOS SLB builds connections to the specified subnet or port, and drops all other connections from the real server.
- Static NAT with a specified address—The real server is configured to use a user-specified virtual IP address when translating addresses.

- Static NAT with per-packet server load balancing—The real server is configured such that IOS SLB is not to maintain connection state for packets originating from the real server. That is, IOS SLB is to use server NAT to redirect packets originating from the real server. Per-packet server load balancing is especially useful for DNS load balancing. IOS SLB uses DNS probes to detect failures in the per-packet server load-balancing environment.
- Static NAT with sticky connections—The real server is configured such that IOS SLB is not to maintain connection state for packets originating from the real server, unless those packets match a sticky object:
  - If IOS SLB finds a matching sticky object, it builds the connection.
  - If IOS SLB does not find a matching sticky object, it forwards the packets without building the connection.

IOS SLB uses the following logic when handling a packet from a real server:

Step 1 Does the packet match a real server?

- If no, IOS SLB has no interest in the packet.
- If yes, continue.

Step 2 Does the packet match an existing connection?

- If yes, IOS SLB uses NAT to redirect the packet, in accordance with the connection control block.
- If no, continue.

Step 3 Is the real server configured to use static NAT?

- If no, IOS SLB handles the packet as usual. This functionality is also called static NAT pass-through.
- If yes, continue.

Step 4 Is the real server configured to have its packets dropped by IOS SLB, if the packets do not correspond to existing connections?

- If yes, IOS SLB drops the packet.
- If no, continue.

Step 5 Is the real server configured for per-packet server load balancing?

- If yes, IOS SLB uses NAT to redirect the packet.
- If no, continue.

Step 6 Is the real server configured to maintain connection state for sticky connections?

- If no, IOS SLB builds the connection.
- If yes, IOS SLB searches for a matching sticky object. Continue.

Step 7 Can IOS SLB find a matching sticky object?

- If no, IOS SLB drops the packet.
- If yes, IOS SLB builds the connection.

Server Port Translation

Server port translation, also known as port address translation, or PAT, is a form of server NAT that involves the translation of virtual server ports instead of virtual server IP addresses. Virtual server port translation does not require translation of the virtual server IP address, but you can use the two types of translation together.

IOS SLB supports server port translation for TCP and UDP only.

Port-Bound Servers

When you define a virtual server, you must specify the TCP or UDP port handled by that virtual server. However, if you configure NAT on the server farm, you can also configure port-bound servers. Port-bound servers allow one virtual server IP address to represent one set of real servers for one service, such as HTTP, and a different set of real servers for another service, such as Telnet.

Packets destined for a virtual server address for a port that is not specified in the virtual server definition are not redirected.

IOS SLB supports both port-bound and non-port-bound servers, but port-bound servers are recommended.

IOS SLB firewall load balancing does not support port-bound servers.

Route Health Injection

By default, a virtual server's IP address is advertised (added to the routing table) when you bring the virtual server into service (using the **inservice** command). If you have a preferred host route to a website's virtual IP address, you can advertise that host route, but you have no guarantee that the IP address is available. However, you can use the **advertise** command to configure IOS SLB to advertise the host route only when IOS SLB has verified that the IP address is available. IOS SLB withdraws the advertisement when the IP address is no longer available. This function is known as route health injection.

Sticky Connections

Sometimes, a client transaction can require multiple consecutive connections, which means new connections from the same client IP address or subnet must be assigned to the same real server. These connections are especially important in firewall load balancing, because the firewall might need to profile the multiple connections in order to detect certain attacks.

- IOS SLB supports source-IP sticky connections.
- Firewall load balancing supports source-IP, destination-IP, and source-destination-IP sticky connections.
- RADIUS load balancing supports calling-station-IP, framed-IP, and username sticky connections.

You can use the optional **sticky** command to enable IOS SLB to force connections from the same client to the same load-balanced server within a server farm. For firewall load balancing, the connections between the same client-server pair are assigned to the same firewall. New connections are considered to be sticky as long as the following conditions are met:

- The real server is in either OPERATIONAL or MAXCONNS_THROTTLED state.
- The sticky timer is defined on a virtual server or on a firewall farm.

This binding of new connections to the same server or firewall is continued for a user-defined period after the last sticky connection ends.

To get the client-server address sticky behavior needed for “sandwich” firewall load balancing, you must enable sticky on both sides of the firewall farm. In this configuration, client-server sticky associations are created when an initial connection is opened between a client-server address pair. After this initial connection is established, IOS SLB maintains the sticky association in the firewall load-balancing devices on either side of the farm, and applies the sticky association to connections initiated from either the client or server IP address, by both firewall load-balancing devices.

Client subnet sticky is enabled when you specify a subnet mask on the **sticky** command. Subnet sticky is useful when the client IP address might change from one connection to the next. For example, before reaching IOS SLB, the client connections might pass through a set of NAT or proxy firewalls that have no sticky management of their own. Such a situation can result in failed client transactions if the servers do not have the logic to cope with it. In cases where such firewalls assign addresses from the same set of subnets, IOS SLB's sticky subnet mask can overcome the problems that they might cause.

Sticky connections also permit the coupling of services that are handled by more than one virtual server or firewall farm. This option allows connection requests for related services to use the same real server. For example, web server (HTTP) typically uses TCP port 80, and HTTPS uses port 443. If HTTP virtual servers and HTTPS virtual servers are coupled, connections for ports 80 and 443 from the same client IP address or subnet are assigned to the same real server.

Virtual servers that are in the same sticky group are sometimes called buddied virtual servers.

Access Service Network (ASN) R6 load balancing and the Home Agent Director do not support sticky connections.

TCP Session Reassignment

IOS SLB tracks each TCP SYN sent to a real server by a client attempting to open a new connection. If several consecutive SYNs are not answered, or if a SYN is replied to with an RST, the TCP session is reassigned to a new real server. The number of SYN attempts is controlled by a configurable reassign threshold.

IOS SLB firewall load balancing does not support TCP session reassignment.

Transparent Web Cache Load Balancing

IOS SLB can load-balance HTTP flows across a cluster of transparent web caches. To set up this function, configure the subnet IP addresses served by the transparent web caches, or some common subset of them, as virtual servers. Virtual servers used for transparent web cache load balancing do not answer pings on behalf of the subnet IP addresses, and they do not affect traceroute.

In some cases, such as when its cache does not contain needed pages, a web cache might need to initiate its own connections to the Internet. Those connections should not be load-balanced back to the same set of web caches. To address this need, IOS SLB allows you to configure **client exclude** statements, which exclude connections initiated by the web caches from the load-balancing scheme.

IOS SLB firewall load balancing does not support transparent web cache load balancing.

Security Features

IOS SLB provides the following security features:

- Alternate IP Addresses, page 21
- Avoiding Attacks on Server Farms and Firewall Farms, page 21
- Slow Start, page 21
- SynGuard, page 21

Alternate IP Addresses

IOS SLB enables you to telnet to the load-balancing device using an alternate IP address. To do so, use either of the following methods:

- Use any of the interface IP addresses to telnet to the load-balancing device.
- Define a secondary IP address to telnet to the load-balancing device.

This function is similar to that provided by the LocalDirector (LD) Alias command.

Avoiding Attacks on Server Farms and Firewall Farms

IOS SLB relies on a site's firewalls to protect the site from attacks. In general, IOS SLB is no more susceptible to direct attack than is any switch or router. However, a highly secure site can take the following steps to enhance its security:

- Configure real servers on a private network to keep clients from connecting directly to them. This configuration ensures that the clients must go through IOS SLB to get to the real servers.
- Configure input access lists on the access router or on the IOS SLB device to deny flows from the outside network aimed directly at the interfaces on the IOS SLB device. That is, deny *all* direct flows from unexpected addresses.
- To protect against attackers trying to direct flows to real or nonexistent IP addresses in the firewall subnet, configure the firewalls in a private network.
- Configure firewalls to deny *all* unexpected flows targeted at the firewalls, especially flows originating from the external network.

Slow Start

In an environment that uses weighted least connections load balancing, a real server that is placed in service initially has no connections, and could therefore be assigned so many new connections that it becomes overloaded. To prevent such an overload, slow start controls the number of new connections that are directed to a real server that has just been placed in service.

GPRS load balancing and the Home Agent Director do not support slow start.

SynGuard

SynGuard limits the rate of TCP start-of-connection packets (SYNchronize sequence numbers, or SYNs) handled by a virtual server to prevent a type of network problem known as a SYN flood denial-of-service attack. A user might send a large number of SYNs to a server, which could overwhelm or crash the server, denying service to other users. SynGuard prevents such an attack from bringing down IOS SLB or a real server. SynGuard monitors the number of SYNs handled by a virtual server at specific intervals and does not allow the number to exceed a configured SYN threshold. If the threshold is reached, any new SYNs are dropped.

IOS SLB firewall load balancing and the Home Agent Director do not support SynGuard.

Server Failure Detection and Recovery Features

IOS SLB provides the following server failure detection and recovery features:

- Automatic Server Failure Detection, page 22
- Automatic Unfail, page 22
- Backup Server Farms, page 22

- DFP Agent Subsystem Support, page 23
- Dynamic Feedback Protocol for IOS SLB, page 23
- GGSN-IOS SLB Messaging, page 24
- INOP_REAL State for Virtual Servers, page 24
- Probes, page 25

Automatic Server Failure Detection

IOS SLB automatically detects each failed Transmission Control Protocol (TCP) connection attempt to a real server, and increments a failure counter for that server. (The failure counter is not incremented if a failed TCP connection from the same client has already been counted.) If a server's failure counter exceeds a configurable failure threshold, the server is considered out of service and is removed from the list of active real servers.

For RADIUS load balancing, the IOS SLB performs automatic server failure detection when a RADIUS request is not answered by the real server.

If you have configured all-port virtual servers (that is, virtual servers that accept flows destined for all ports except GTP ports), flows can be passed to servers for which no application port exists. When the servers reject these flows, IOS SLB might fail the servers and remove them from load balancing. This situation can also occur in slow-to-respond AAA servers in RADIUS load-balancing environments. To prevent this situation, you can disable automatic server failure detection.

Note If you disable automatic server failure detection using the **no faildetect inband** command, Cisco strongly recommends that you configure one or more probes.

If you specify the **no faildetect inband** command, the **faildetect numconns** command is ignored, if specified.

Automatic Unfail

When a real server fails and is removed from the list of active servers, it is assigned no new connections for a length of time specified by a configurable retry timer. After that timer expires, the server is again eligible for new virtual server connections and IOS SLB sends the server the next qualifying connection. If the connection is successful, the failed server is placed back on the list of active real servers. If the connection is unsuccessful, the server remains out of service and the retry timer is reset. The unsuccessful connection must have experienced at least one retry, otherwise the next qualifying connection would also be sent to that failed server.

Backup Server Farms

A backup server farm is a server farm that can be used when none of the real servers defined in a primary server farm is available to accept new connections. When configuring backup server farms, keep in mind the following considerations:

- A server farm can act as both primary and backup at the same time.
- The same real server cannot be defined in both primary and backup at the same time.
- Both primary and backup require the same NAT configuration (none, client, server, or both). In addition, if NAT is specified, both server farms must use the same NAT pool.

DFP Agent Subsystem Support

IOS SLB supports the DFP Agent Subsystem feature, also called global load balancing, which enables client subsystems other than IOS SLB to act as DFP agents. With the DFP Agent Subsystem, you can use multiple DFP agents from different client subsystems at the same time.

For more information about the DFP Agent Subsystem, refer to the *DFP Agent Subsystem* feature document for Cisco IOS Release 12.2(18)SXD.

Dynamic Feedback Protocol for IOS SLB

With IOS SLB Dynamic Feedback Protocol (DFP) support, a DFP manager in a load-balancing environment can initiate a TCP connection with a DFP agent. Thereafter, the DFP agent collects status information from one or more real host servers, converts the information to relative weights, and reports the weights to the DFP manager. The DFP manager factors in the weights when load balancing the real servers. In addition to reporting at user-defined intervals, the DFP agent sends an early report if there is a sudden change in a real server’s status.

The weights calculated by DFP override the static weights you define using the **weight** command in server farm configuration mode. If DFP is removed from the network, IOS SLB reverts to the static weights.

You can define IOS SLB as a DFP manager, as a DFP agent for another DFP manager, or as both at the same time. In such a configuration, IOS SLB sends periodic reports to the other DFP manager, which uses the information to choose the best server farm for each new connection request. IOS SLB then uses the same information to choose the best real server within the chosen server farm.

DFP also supports the use of multiple DFP agents from different client subsystems (such as IOS SLB and GPRS) at the same time.

See the following sections for more information:

- DFP and GPRS Load Balancing, page 23
- DFP and the Home Agent Director, page 24

DFP and GPRS Load Balancing

In GPRS load balancing, you can define IOS SLB as a DFP manager and define a DFP agent on each GGSN in the server farm. Thereafter, the DFP agent can report the weights of the GGSNs. The DFP agents calculate the weight of each GGSN based on CPU utilization, processor memory, and the maximum number of Packet Data Protocol (PDP) contexts (mobile sessions) that can be activated for each GGSN. As a first approximation, DFP calculates the weight as the number of existing PDP contexts divided by the maximum allowed PDP contexts:

(existing PDP contexts)/(maximum PDP contexts)

Maximum PDP contexts are specified using the **gprs maximum-pdp-context-allowed** command, which defaults to 10,000 PDP contexts. If you accept the default value, DFP might calculate a very low weight for the GGSN:

(existing PDP contexts)/10000 = Low GGSN weight

Keep this calculation in mind when specifying maximum PDP contexts using the **gprs maximum-pdp-context-allowed** command. For example, Cisco 7200 series routers acting as GGSNs are often configured with a maximum of 45,000 PDP contexts.

DFP and the Home Agent Director

For the Home Agent Director, you can define IOS SLB as a DFP manager and define a DFP agent on each home agent in the server farm, and the DFP agent can report the weights of the home agents. The DFP agents calculate the weight of each home agent based on CPU utilization, processor memory, and the maximum number of bindings that can be activated for each home agent:

> (maximum-number-of-bindings - current-number-of-bindings)/maximum-number-of-bindings *
> (cpu-utilization + memory-utilization)/32 * maximum-DFP-weight = reported-weight

To set the *maximum-number-of-bindings*, use the **ip mobile home-agent max-binding** command. To set the *maximum-DFP-weight* sent by the home agent to IOS SLB, use the **ip mobile home-agent dfp-max-weight** command. For detailed information about these Mobile IP commands, refer to the *Cisco Mobile Wireless Home Agent Release 2.0* document.

GGSN-IOS SLB Messaging

This feature enables a GGSN to notify IOS SLB when certain conditions occur. The notifications enable IOS SLB to make intelligent decisions, which in turn improves GPRS load balancing and failure detection.

The notifications sent by the GGSN use GTP with message types from the unused space (reserved for future use) and the following information elements (IEs):

- Notification type, which indicates the notification condition. For example, this could be a notification to IOS SLB to reassign the session to an alternate GGSN, when the current GGSN fails on Call Admission Control (CAC).
- Identifier of the relevant session (session key).
- Other IEs specific to the notification type. For example, for a notification to reassign, GGSN includes the create response, which it would otherwise have sent to the SGSN. This enables IOS SLB to relay this response back to SGSN when the maximum number of reassignments due to notification reach the configured limit.

GGSN-IOS SLB messaging is supported in both dispatched mode and directed modes.

INOP_REAL State for Virtual Servers

You can configure a virtual server such that, if all of the real servers that are associated with the virtual server are inactive, the following actions occur:

- The virtual server is placed in the INOP_REAL state.
- An SNMP trap is generated for the virtual server’s state transition.
- The virtual server stops answering ICMP requests.

For more information, see the description of the **inservice (server farm virtual server)** command in SLB server farm virtual server configuration mode.

Probes

IOS SLB supports DNS, HTTP, ping, TCP, custom UDP, and WSP probes:

- A DNS probe sends domain name resolve requests to real servers, and verifies the returned IP addresses.
- An HTTP probe establishes HTTP connections to real servers, sends HTTP requests to the real servers, and verifies the responses. HTTP probes are a simple way to verify connectivity for devices being server load-balanced, and for firewalls being firewall load-balanced (even devices on the other side of a firewall).

  HTTP probes also enable you to monitor applications being server load-balanced. With frequent probes, the operation of each application is verified, not just connectivity to the application.

  HTTP probes do not support HTTP over Secure Socket Layer (HTTPS). That is, you cannot send an HTTP probe to an SSL server.
- A ping probe pings real servers. Like HTTP probes, ping probes are a simple way to verify connectivity for devices and firewalls being load-balanced.
- A TCP probe establishes and removes TCP connections. Use TCP probes to detect failures on TCP port 443 (HTTPS).
- A custom UDP probe can to support a variety of applications and protocols, including:
  - RADIUS Accounting/Authorization probes
  - GTP Echo probes
  - Connectionless WSP probes
  - XML-over-UDP probes for CSG user-database load-balancing
  - Mobile IP RRQ/RRP
- A WSP probe simulates requests for wireless content and verifies the retrieved content. Use WSP probes to detect failures in the Wireless Application Protocol (WAP) stack on port 9201.

You can configure more than one probe, in any combination of supported types, for each server farm, or for each firewall in a firewall farm.

You can also flag a probe as a routed probe, with the following considerations:

- Only one instance of a routed probe per server farm can run at any given time.
- Outbound packets for a routed probe are routed directly to a specified IP address.

IOS SLB probes use the SA Agent. You might want to specify the amount of memory that the SA Agent can use, using the **rtr low-memory** command. If the amount of available free memory falls below the value specified in the **rtr low-memory** command, then the SA Agent does not allow new operations to be configured. Refer to the description of the **rtr low-memory** command in the *Cisco IOS IP SLAs Command Reference* for more details.

Probes in Server Load Balancing

Probes determine the status of each real server in a server farm. All real servers associated with all virtual servers tied to that server farm are probed.

If a real server fails for one probe, it is failed for all probes. After the real server recovers, all probes must acknowledge its recovery before it is restored to service.

Probes in Firewall Load Balancing

Probes detect firewall failures. All firewalls associated with the firewall farm are probed.

If a firewall fails for one probe, it is failed for all probes. After the firewall recovers, all probes must acknowledge its recovery before it is restored to service.

Make sure you configure the HTTP probe to expect status code 401, to eliminate password problems. Refer to the description of the **expect** command for more details.

Use the **ip http server** command to configure an HTTP server on the device. Refer to the description of the **ip http server** command in the *Cisco IOS Configuration Fundamentals Command Reference* for more details.

In a transparent web cache load-balancing environment, an HTTP probe uses the real IP address of the web cache, since there is no virtual IP address configured.

Protocol Support Features

IOS SLB provides the following protocol support features:

- Protocol Support, page 26
- AAA Load Balancing, page 27
- Audio and Video Load Balancing, page 27
- VPN Server Load Balancing, page 27

Protocol Support

IOS SLB supports the following protocols:

- Access Service Network (ASN) R6
- Domain Name System (DNS)
- Encapsulation Security Payload (ESP)
- File Transfer Protocol (FTP)
- Generic Routing Encapsulation (GRE)
- GPRS Tunneling Protocol v0 (GTPv0)
- GPRS Tunneling Protocol v1 (GTPv1)
- Hypertext Transfer Protocol (HTTP)
- Hypertext Transfer Protocol over Secure Socket Layer (HTTPS)
- Internet Message Access Protocol (IMAP)
- Internet Key Exchange (IKE, was ISAKMP)
- IP in IP Encapsulation (IPinIP)
- Mapping of Airline Traffic over IP, Type A (MATIP-A)
- Network News Transport Protocol (NNTP)
- Post Office Protocol, version 2 (POP2)
- Post Office Protocol, version 3 (POP3)
- RealAudio/RealVideo via RTSP
- Remote Authentication Dial-In User Service (RADIUS)

- Simple Mail Transport Protocol (SMTP)
- Telnet
- Transmission Control Protocol (TCP) and standard TCP protocols
- User Datagram Protocol (UDP) and standard UDP protocols
- X.25 over TCP (XOT)
- Wireless Application Protocol (WAP), including:
  - Connectionless Secure WSP
  - Connectionless WSP
  - Connection-Oriented Secure WSP
  - Connection-Oriented WSP

AAA Load Balancing

IOS SLB provides RADIUS load-balancing capabilities for RADIUS authentication, authorization, and accounting (AAA) servers.

IOS SLB provides the following RADIUS load-balancing functions:

- Balances RADIUS requests among available RADIUS servers and proxy servers.
- Routes RADIUS request retransmissions (such as retransmissions of unanswered requests) to the same RADIUS server or proxy server as the original request.
- Provides session-based automatic failure detection.
- Supports both stateless backup and stateful backup.

In addition, IOS SLB can load-balance devices that proxy the RADIUS Authorization and Accounting flows in both traditional and mobile wireless networks. For more information, see the "RADIUS Load Balancing" section on page 32.

Audio and Video Load Balancing

IOS SLB can balance RealAudio and RealVideo streams via Real-Time Streaming Protocol (RTSP), for servers running RealNetworks applications.

VPN Server Load Balancing

IOS SLB can balance Virtual Private Network (VPN) flows, including the following flows:

- IP Security (IPSec) flows. An IPSec flow consists of a UDP control session and an ESP tunnel.
- Point-to-Point Tunneling Protocol (PPTP) flows. A PPTP flow consists of a TCP control session and a GRE tunnel.

Redundancy Features

An IOS SLB device can represent a single point of failure, and the servers can lose their connections to the backbone, if either of the following occurs:

- The IOS SLB device fails.
- A link from a switch to the distribution-layer switch becomes disconnected.

To reduce that risk, IOS SLB supports the following redundancy enhancements, based on HSRP:

- Stateless Backup, page 28
- Stateful Backup, page 28
- Active Standby, page 28

Stateless Backup

Stateless backup provides high network availability by routing IP flows from hosts on Ethernet networks without relying on the availability of a single Layer 3 switch. Stateless backup is particularly useful for hosts that do not support a router discovery protocol (such as the Intermediate System-to-Intermediate System [IS-IS] Interdomain Routing Protocol [IDRP]) and do not have the functionality to shift to a new Layer 3 switch when their selected Layer 3 switch reloads or loses power.

Stateful Backup

Stateful backup enables IOS SLB to incrementally backup its load-balancing decisions, or "keep state," between primary and backup switches. The backup switch keeps its virtual servers in a dormant state until HSRP detects failover; then the backup (now primary) switch begins advertising virtual addresses and processing flows. You can use HSRP to configure how quickly the failover is detected.

Stateful backup provides IOS SLB with a one-to-one stateful or idle backup scheme. This means that only one instance of IOS SLB is handling client or server flows at a given time, and that there is at most one backup platform for each active IOS SLB switch.

Access Service Network (ASN) R6 load balancing and the Home Agent Director do not support stateful backup.

Active Standby

Active standby enables two IOS SLBs to load-balance the same virtual IP address while at the same time acting as backups for each other. If a site has only one virtual IP address to load-balance, an access router is used to direct a subset of the flows to each IOS SLB using policy-based routing.

IOS SLB firewall load balancing supports active standby. That is, you can configure two pairs of firewall load balancing devices (one pair on each side of the firewalls), with each device in each pair handling traffic and backing up its partner.

Exchange Director Features

IOS SLB supports the Exchange Director for the mobile Service Exchange Framework (mSEF) for Catalyst 6500 family switches and Cisco 7600 series routers. The Exchange Director provides the following features:

- ASN R6 Load Balancing, page 29
- GPRS Load Balancing, page 29
- GPRS Load Balancing without GTP Cause Code Inspection, page 30
- GPRS Load Balancing with GTP Cause Code Inspection, page 30
- Home Agent Director, page 31
- KeepAlive Application Protocol (KAL-AP) Agent Support, page 31
- RADIUS Load Balancing, page 32

ASN R6 Load Balancing

IOS SLB can provide load balancing across a set of Access Service Network (ASN) gateways. The gateway server farm appears to the base station as a single ASN gateway.

When a Mobile Subscriber Station (MSS) wants to enter the network, the base station sends a Mobile Station (MS) Pre-Attachment request to the virtual IP address of the IOS SLB. IOS SLB selects an ASN gateway and forwards the request to that gateway. The gateway responds directly to the base station with an MS Pre-Attachment response. If configured to do so, the base station then returns an MS Pre-Attachment ACK to IOS SLB, which forwards the ACK to the selected gateway. Thereafter, all subsequent transactions flow between the base station and the gateway.

GPRS Load Balancing

General packet radio service (GPRS) is the packet network infrastructure based on the European Telecommunications Standards Institute (ETSI) Global System for Mobile Communication (GSM) phase 2+ standards for transferring packet data from the GSM mobile user to the packet data network (PDN). The Cisco gateway GPRS support node (GGSN) interfaces with the serving GPRS support node (SGSN) using the GPRS Tunneling Protocol (GTP), which in turn uses UDP for transport. IOS SLB provides GPRS load balancing and increased reliability and availability for the GGSN.

When configuring the network shared by IOS SLB and the GGSNs, keep the following considerations in mind:

- Specify static routes (using **ip route** commands) and real server IP addresses (using **real** commands) such that the Layer 2 information is correct and unambiguous.
- Choose subnets carefully, using one of the following methods:
  - Do not overlap virtual template address subnets.
  - Specify next hop addresses to real servers, not to interfaces on those servers.
- IOS SLB assigns all PDP context creates from a specific IMSI to the same GGSN.
- IOS SLB supports both GTP Version 0 (GTP v0) and GTP Version 1 (GTP v1). Support for GTP enables IOS SLB to become “GTP aware,” extending IOS SLB’s knowledge into Layer 5.
- GPRS load balancing maps enable IOS SLB to categorize and route user traffic based on access point names (APNs).

IOS SLB supports two types of GPRS load balancing:

- GPRS Load Balancing without GTP Cause Code Inspection, page 30
- GPRS Load Balancing with GTP Cause Code Inspection, page 30

GPRS Load Balancing without GTP Cause Code Inspection

GPRS load balancing *without* GTP cause code inspection enabled is recommended for Cisco GGSNs. It has the following characteristics:

- Can operate in dispatched mode or in directed server NAT mode, but not in directed client NAT mode. In dispatched mode, the GGSNs must be Layer 2-adjacent to the IOS SLB device.
- Supports stateful backup only if sticky connections are enabled. See the "Stateful Backup" section on page 28 for more information.
- Delivers tunnel creation messages destined to the virtual GGSN IP address to one of the real GGSNs, using the weighted round robin load-balancing algorithm. See the "Weighted Round Robin" section on page 11 for more information about this algorithm.
- Requires DFP in order to account for secondary PDP contexts in GTP v1.

GPRS Load Balancing with GTP Cause Code Inspection

GPRS load balancing *with* GTP cause code inspection enabled allows IOS SLB to monitor all PDP context signaling flows to and from GGSN server farms. This enables IOS SLB to monitor GTP failure cause codes, detecting system-level problems in both Cisco and non-Cisco GGSNs.

Table 1 lists the PDP create response cause codes and the corresponding actions taken by IOS SLB.

Table 1 PDP Create Response Cause Codes and Corresponding IOS SLB Actions

Cause Code	IOS SLB Action
Request Accepted	Establish session
No Resource Available	Fail current real, reassign session, drop the response
All dynamic addresses are occupied	Fail current real, reassign session, drop the response
No memory is available	Fail current real, reassign session, drop the response
System Failure	Fail current real, reassign session, drop the response
Missing or Unknown APN	Forward the response
Unknown PDP Address or PDP type	Forward the response
User Authentication Failed	Forward the response
Semantic error in TFT operation	Forward the response
Syntactic error in TFT operation	Forward the response
Semantic error in packet filter	Forward the response
Syntactic error in packet filter	Forward the response
Mandatory IE incorrect	Forward the response
Mandatory IE missing	Forward the response
Optional IE incorrect	Forward the response
Invalid message format	Forward the response
Version not supported	Forward the response

GPRS load balancing *with* GTP cause code inspection enabled has the following characteristics:

- Must operate in directed server NAT mode.
- Supports stateful backup. See the "Stateful Backup" section on page 28 for more information.

- Tracks the number of open PDP contexts for each GGSN, which enables GGSN server farms to use the weighted least connections (**leastconns**) algorithm for GPRS load balancing. See the "Weighted Least Connections" section on page 12 for more information about this algorithm.
- Enables IOS SLB to deny access to a virtual GGSN if the carrier code of the requesting International Mobile Subscriber ID (IMSI) does not match a specified value.
- Enables IOS SLB to account for secondary PDP contexts even without DFP.

Home Agent Director

The Home Agent Director load balances Mobile IP Registration Requests (RRQs) among a set of home agents (configured as real servers in a server farm). Home agents are the anchoring points for mobile nodes. Home agents route flows for a mobile node to its current foreign agent (point of attachment).

The Home Agent Director has the following characteristics:

- Can operate in dispatched mode or in directed server NAT mode, but not in directed client NAT mode. In dispatched mode, the home agents must be Layer 2-adjacent to the IOS SLB device.
- Does not support stateful backup. See the "Stateful Backup" section on page 28 for more information.
- Delivers RRQs destined to the virtual Home Agent Director IP address to one of the real home agents, using the weighted round robin load-balancing algorithm. See the "Weighted Round Robin" section on page 11 for more information about this algorithm.
- Requires DFP in order to allocate RRQs based on capacity.

For more information about Mobile IP, home agents, and related topics, refer to the *Cisco IOS IP Configuration Guide*, Release 12.2.

KeepAlive Application Protocol (KAL-AP) Agent Support

KAL-AP agent support enables IOS SLB to perform load balancing in a global server load balancing (GSLB) environment. KAL-AP provides load information along with its keepalive response message to the KAL-AP manager or GSLB device, such as the Global Site Selector (GSS), and helps the GSLB device load-balance client requests to the least-loaded IOS SLB devices.

When configuring KAL-AP agent support for IOS SLB, keep the following considerations in mind:

- KAL-AP agent support automatically detects the Virtual Private Network (VPN) routing and forwarding (VRF) ID of an incoming request packet, and uses the same VRF ID when sending a response.
- A client that uses DNS caching might contact IOS SLB directly, instead of sending requests through the GSS. Therefore, configure the DNS setting in the client to avoid such a situation.

KAL-AP calculates the load value in one of two ways: relatively or absolutely. (IOS SLB CPU/memory load might affect the final KAL-AP load value.)

Relative KAL-AP Load Value

If the **farm-weight** command is not configured in server farm configuration mode, or if DFP is not enabled for the IOS SLB, KAL-AP calculates a relative load value, using the following formula:

KAL-AP Load = 256 - (number-*of-active-real-servers* * **256** / *number-of-inservice-real-servers*)

For example, if a site is provisioned with two real servers, and both real servers are inservice but only one is currently active, the resulting KAL-AP load value for that site is:

KAL-AP Load = 256 - (**1** * **256 / 2**) = **256** - **128 = 128**

Absolute KAL-AP Load Value

If the **farm-weight** command is configured in server farm configuration mode, and DFP is enabled for the IOS SLB, KAL-AP calculates an absolute load value, using the following formula:

KAL-AP Load = 256 - (*sum-of-max-dfp-weights-of-real-servers* * **256** / *farm-weight*)

Note The maximum DFP weight for a real server is configured using the **gprs dfp max-weight** command in global configuration mode. However, the actual maximum DFP weight reported to KAL-AP is proportional to the load on the GGSN. For example, if a GGSN is configured with a maximum DFP weight of 100, but the GGSN is 50% loaded, it reports a maximum DFP weight of 50 to KAL-AP.

If the DFP connection to the real server is down, KAL-AP uses the setting of the **weight** command in SLB real server configuration mode. If no **weight** command is configured for the real server, KAL-AP uses the default weight of 8.

For example, consider a site with the following settings:

- A server farm configured with a farm weight of 200.
- GGSN-1 configured with a maximum DFP weight of 100, 0% loaded (so it reports a DFP weight of 100).
- GGSN-2 configured with a maximum DFP weight of 100, 50% loaded (so it reports a DFP weight of only 50).

The resulting KAL-AP load value for that site is:

KAL-AP Load = 256 - [(**100 + 50**) * **256 / 200**] = **256** - **192 = 64**

For best results, configure a **farm-weight** that is equal to the sum of the maximum DFP weights for the real servers in the server farm. For example, if there are three real servers in a server farm, configured with maximum DFP weights of 100, 50, and 50, then configure a **farm-weight** of 200 (that is, 100 + 50 + 50). If a real server is added to or removed from the server farm, you must adjust the **farm-weight** accordingly.

RADIUS Load Balancing

IOS SLB provides RADIUS load-balancing capabilities for RADIUS servers. In addition, IOS SLB can load-balance devices that proxy the RADIUS Authorization and Accounting flows in both traditional and mobile wireless networks, if desired. IOS SLB does this by correlating data flows to the same proxy that processed the RADIUS for that subscriber flow.

IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). The following mobile wireless networks are supported:

- GPRS networks. In a GPRS mobile wireless network, the RADIUS client is typically a GGSN.
- Simple IP CDMA2000 networks. CDMA2000 is a third-generation (3-G) version of Code Division Multiple Access (CDMA). In a simple IP CDMA2000 mobile wireless network, the RADIUS client is a Packet Data Service Node (PDSN).
- Mobile IP CDMA2000 networks. In a Mobile IP CDMA2000 mobile wireless network, both the Home Agent (HA) and the PDSN/Foreign Agent (PDSN/FA) are RADIUS clients.

IOS SLB provides the following RADIUS load-balancing functions:

- Balances RADIUS requests among available RADIUS servers and proxy servers.
- Routes RADIUS request retransmissions (such as retransmissions of unanswered requests) to the same RADIUS server or proxy server as the original request.
- Routes all of a subscriber's RADIUS flows, as well as all non-RADIUS data flows for the same subscriber, to the same service gateway.
- Supports multiple service gateway server farms (for example, one farm of SSGs and another of CSGs). IOS SLB examines the input interface in a packet to route it to the correct service gateway server farm.
- Supports multiple WAP gateway server farms behind a RADIUS load balancing virtual server, using RADIUS calling station IDs and usernames to select specific server farms. This enhancement enables RADIUS load balancing on both the control plane and the data plane. RADIUS load balancing on the control plane enables RADIUS messages to be load-balanced to AAA servers for subscriber authorization, authentication and accounting. RADIUS load balancing on the data plane enables data flows for a given subscriber to maintain a consistent network path to the destination network device. In addition, the RADIUS virtual server can acknowledge RADIUS accounting messages and build or delete sticky objects, rather than having to forward the messages to the specified server.
- Can route data packets to a real server in the CSG farm, then to a real server in the SSG farm.
- Routes RADIUS Accounting-Request messages from a RADIUS client to the service gateway that processed the RADIUS Access-Request message for the subscriber. The service gateway can then clean up the host entry it has created for the subscriber.
- Uses the weighted round robin load-balancing algorithm. See the "Weighted Round Robin" section on page 11 for more information about this algorithm.
- Facilitates SSG single sign-on via the RADIUS protocol.
- Provides session-based automatic failure detection.
- Supports both stateless backup and stateful backup.

To perform RADIUS load balancing, IOS SLB uses the following RADIUS sticky databases:

- The IOS SLB RADIUS framed-IP sticky database associates each subscriber's IP address with a specific service gateway. In a GPRS mobile wireless network, IOS SLB uses the RADIUS framed-IP sticky database to route packets correctly.

Note Subscriber IP addresses are assigned by service gateways or by RADIUS clients. If subscriber IP addresses are assigned from disjoint per-service gateway pools (so that the next-hop service gateway can be chosen based on the source IP address), IOS SLB can use policy routing to route subscriber flows.

- The IOS SLB RADIUS calling-station-ID sticky database associates each subscriber's calling station ID with a specific service gateway.
- The IOS SLB RADIUS username sticky database associates each subscriber's username with a specific service gateway.
- RADIUS load balancing maps enable IOS SLB to categorize and route user traffic based on RADIUS calling station IDs and user names. RADIUS load balancing maps is mutually exclusive with Turbo RADIUS load balancing and RADIUS load balancing accounting local acknowledgement.

- RADIUS load balancing accounting local acknowledgement enables IOS SLB to respond to RADIUS accounting packets with an ACK response while maintaining sticky objects for those sessions. RADIUS load balancing accounting local acknowledgement is mutually exclusive with RADIUS load balancing maps and Turbo RADIUS load balancing.
- In a CDMA2000 mobile wireless network, to route packets correctly, IOS SLB requires both the RADIUS framed-IP sticky database and either the RADIUS username sticky database or the RADIUS calling-station-ID sticky database.
- The IOS SLB RADIUS International Mobile Subscriber ID (IMSI) sticky database maps the IMSI address for each user to the corresponding gateway. This enables IOS SLB to forward all subsequent flows for the same user to the same gateway.

RADIUS Load Balancing Accelerated Data Plane Forwarding

RADIUS load balancing accelerated data plane forwarding, also known as Turbo RADIUS load balancing, is a high-performance solution that uses basic policy-based routing (PBR) route maps to handle subscriber data-plane traffic in a CSG environment.

When Turbo RADIUS load balancing receives a RADIUS payload, it inspects the payload, extracts the framed-IP attribute, applies a route map to the IP address, and then determines which CSG is to handle the subscriber.

If vendor-specific attribute (VSA) correlation is configured, and if the Cisco VSA is buffered, then the Cisco VSA is injected into the RADIUS Accounting-Start packet.

Turbo RADIUS load balancing does not require VSA correlation, but it does require a server farm configured with **predictor route-map** on the accounting virtual server.

Note When you specify the **predictor route-map** command in SLB server farm configuration mode, no further commands in SLB server farm configuration mode or real server configuration mode are allowed.

For more information about policy-based routing, including how it works, when to use it, how to configure it, and how to enable it, see the "Policy-Based Routing" and "Configuring Policy-Based Routing" sections of the *Cisco IOS IP Routing Configuration Guide*.

In a mobile Service Exchange Framework (mSEF) environment, Turbo RADIUS load balancing does not require firewall load balancing on the network side of the CSG cluster. (Standard RADIUS load balancing does require firewall load balancing on the network side of the cluster.)

Turbo RADIUS load balancing is mutually exclusive with RADIUS load balancing maps and RADIUS load balancing accounting local acknowledgement.

Turbo RADIUS load balancing is mutually exclusive with Virtual Private Network (VPN) routing and forwarding (VRF).

Turbo RADIUS load balancing supports simple IP access control lists (ACLs) and match and set next-hop pairs.

WAP Load Balancing

You can use IOS SLB to load-balance Wireless Session Protocol (WSP) sessions among a group of WAP gateways or servers on an IP bearer network. WAP runs on top of UDP on a set of well known ports, with each port indicating a different WAP mode:

- Connectionless WSP mode (IP/UDP [9200]/WSP). In connectionless WSP mode, WSP is a simple one-request/one-response protocol in which a single server-bound packet results in a server response of one or more packets.

- Connection-oriented WSP mode (IP/UDP [9201]/WTP/WSP). In connection-oriented WSP mode, WTP handles retransmissions of WDP events, and WSP operates using a defined session bring-up/tear-down sequence. IOS SLB uses a WAP-aware finite state machine (FSM), driven by events in WSP sessions, to reassign sessions. This FSM operates only on port 9201, where the WSP sessions are not encrypted and WTP handles retransmissions.
- Connectionless secure WSP mode (IP/UDP [9202]/WTLS/WSP). This mode functions the same as connectionless WSP mode, but with security provided by WTLS.
- Connection-oriented secure WSP mode (IP/UDP [9203]/WTLS/WTP/WSP). This mode functions the same as connection-oriented WSP mode, but with security provided by WTLS.

IOS SLB uses WSP probes to detect failures in the WAP stack on port 9201.

Stateful Backup of Redundant Route Processors

When used with RPR+, IOS SLB supports the stateful backup of redundant route processors for mSEF for Catalyst 6500 family switches and Cisco 7600 series routers. This enables you to deploy Cisco Multiprocessor WAN Application Modules (MWAMs) in the same chassis as IOS SLB, while maintaining high availability of load-balancing assignments.

Flow Persistence

Flow persistence provides intelligent return routing of load-balanced IP flows to the appropriate node, without the need for coordinated hash mechanisms on both sides of the load-balanced data path, and without using Network Address Translation (NAT) or proxies to change client or server IP addresses.

How to Configure IOS SLB

Configuring IOS SLB involves identifying server farms, configuring groups of real servers in server farms, and configuring the virtual servers that represent the real servers to the clients.

For configuration examples associated with these tasks, see the "Configuration Examples for IOS SLB" section on page 105.

For a complete description of the IOS SLB commands in this section, refer to the "Server Load Balancing Commands" chapter of the *Cisco IOS IP Application Services Command Reference.* To locate documentation of other commands that appear in this section, search online using Cisco.com.

To configure IOS SLB, perform the tasks in the following sections:

- Configuring Required and Optional IOS SLB Functions, page 36 (Required)
- Configuring Firewall Load Balancing, page 47 (Optional)
- Configuring a Probe, page 53 (Optional)
- Configuring DFP, page 63 (Optional)
- GPRS Load Balancing Configuration Task List, page 64 (Optional)
- GGSN-IOS SLB Messaging Task List, page 66 (Optional)
- Configuring GPRS Load Balancing Maps, page 67 (Optional)
- Configuring KeepAlive Application Protocol (KAL-AP) Agent Support, page 69 (Optional)
- RADIUS Load Balancing Configuration Task List, page 71 (Optional)
- Exchange Director for mSEF Configuration Task List, page 79 (Optional)

- VPN Server Load Balancing Configuration Task List, page 88 (Optional)
- ASN R6 Load Balancing Configuration Task List, page 89 (Optional)
- Home Agent Director Configuration Task List, page 89 (Optional)
- Configuring NAT, page 91 (Optional)
- Configuring Static NAT, page 91 (Optional)
- Stateless Backup Configuration Task List, page 92 (Optional)
- Stateful Backup of Redundant Route Processors Configuration Task List, page 94 (Optional)
- Configuring Database Entries, page 95 (Optional)
- Configuring Buffers for the Fragment Database, page 96 (Optional)
- Clearing Databases and Counters, page 96 (Optional)
- Configuring a Wildcard Search, page 97 (Optional)
- Purging and Reassigning Connections, page 98 (Optional)
- Disabling Automatic Server Failure Detection, page 99 (Optional)
- Monitoring and Maintaining IOS SLB, page 100 (Optional)

Configuring Required and Optional IOS SLB Functions

To configure IOS SLB functions, perform the tasks in the following sections. Required and optional tasks are indicated.

- Configuring a Server Farm and a Real Server, page 36 (Required)
- Configuring a Virtual Server, page 40 (Required)
- Verifying a Server Farm, page 45 (Optional)
- Verifying a Virtual Server, page 45 (Optional)
- Verifying the Clients, page 46 (Optional)
- Verifying IOS SLB Connectivity, page 46 (Optional)

Configuring a Server Farm and a Real Server

Perform this task to configure a server farm and a real server.

Note You cannot configure IOS SLB from different user sessions at the same time.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb serverfarm** *server-farm*
4. **bindid** [*bind-id*]
5. **nat** {**client** *pool* | **server**}
6. **predictor** [**roundrobin** | **leastconns** | **route-map** *mapname*]

7. **probe** *probe*
8. **real** *ip-address* [*port*]
9. **faildetect numconns** *number-of-conns* [**numclients** *number-of-clients*]
10. **maxclients** *number-of-conns*
11. **maxconns** *number-of-conns* [**sticky-override**]
12. **reassign** *threshold*
13. **retry** *retry-value*
14. **weight** *setting*
15. **inservice**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb serverfarm** *server-farm* **Example:** `Router(config)# ip slb serverfarm PUBLIC`	Adds a server farm definition to the IOS Server Load Balancing (IOS SLB) configuration and enters server farm configuration mode.
Step 4	**bindid** [*bind-id*] **Example:** `Router(config-slb-sfarm)# bindid 309`	(Optional) Specifies a bind ID on the server farm for use by Dynamic Feedback Protocol (DFP). **Note** GPRS load balancing and Home Agent Director do not support this command.

Step 5	**nat** {**client** *pool* | **server**}
	Example:
	`Router(config-slb-sfarm)# nat server`
		(Optional) Configures Network Address Translation (NAT) client translation mode or NAT server address translation mode on the server farm.

	Command	Purpose
Step 6	**predictor** [**roundrobin** | **leastconns** | **route-map** *mapname*] **Example:** `Router(config-slb-sfarm)# predictor leastconns`	(Optional) Specifies the algorithm to be used to determine how a real server is selected. **Note** RADIUS load balancing requires the default setting (the weighted round robin algorithm). In GPRS load balancing without GTP cause code inspection enabled, you must accept the default setting (the weighted round robin algorithm). The Home Agent Director requires the default setting (the weighted round robin algorithm). When you specify the **predictor route-map** command in SLB server farm configuration mode, no further commands in SLB server farm configuration mode or real server configuration mode are allowed. See the following sections for more details: • Weighted Round Robin, page 11 • Weighted Least Connections, page 12 • Route Map, page 12
Step 7	**probe** *probe* **Example:** `Router(config-slb-sfarm)# probe PROBE1`	(Optional) Associates a probe with the real server.
Step 8	**real** *ip-address* [*port*] **Example:** `Router(config-slb-sfarm)# real 10.1.1.1`	Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode. **Note** In GPRS load balancing, specify the IP addresses (virtual template addresses, for Cisco GGSNs) of the real servers performing the GGSN function. In VPN server load balancing, specify the IP addresses of the real servers acting as VPN terminators. For the Home Agent Director, specify the IP addresses of the real servers acting as home agents.

	Command	Purpose
Step 9	**faildetect numconns** *number-of-conns* [**numclients** *number-of-clients*] **Example:** `Router(config-slb-real)# faildetect numconns 10 numclients 3`	(Optional) Specifies the number of consecutive connection failures and, optionally, the number of unique client connection failures, that constitute failure of the real server. In GPRS load balancing, if there is only one SGSN in your environment, specify the **numclients** keyword with a value of 1. In RADIUS load balancing, for automatic session-based failure detection, specify the **numclients** keyword with a value of 1.
Step 10	**maxclients** *number-of-conns* **Example:** `Router(config-slb-real)# maxclients 10`	(Optional) Specifies the maximum number of IOS Server Load Balancing (IOS SLB) RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server.
Step 11	**maxconns** *number-of-conns* [**sticky-override**] **Example:** `Router(config-slb-real)# maxconns 1000`	(Optional) Specifies the maximum number of active connections allowed on the real server at one time.
Step 12	**reassign** *threshold* **Example:** `Router(config-slb-real)# reassign 2`	(Optional) Specifies the threshold of consecutive unacknowledged SYNchronize sequence numbers (SYNs) or Create Packet Data Protocol (PDP) requests that, if exceeded, result in an attempted connection to a different real server. **Note** In GPRS load balancing, you must specify a reassign threshold less than the SGSN's N3-REQUESTS counter value.
Step 13	**retry** *retry-value* **Example:** `Router(config-slb-real)# retry 120`	(Optional) Specifies the interval, in seconds, to wait between the detection of a server failure and the next attempt to connect to the failed server.
Step 14	**weight** *setting* **Example:** `Router(config-slb-real)# weight 24`	(Optional) Specifies the real server's workload capacity relative to other servers in the server farm. **Note** If you use Dynamic Feedback Protocol (DFP), the static weights you define using the **weight** command in server farm configuration mode are overridden by the weights calculated by DFP. If DFP is removed from the network, IOS SLB reverts to the static weights.
Step 15	**inservice** **Example:** `Router(config-slb-real)# inservice`	Enables the real server for use by IOS Server Load Balancing (IOS SLB).

Note When performing server load balancing and firewall load balancing together on a Catalyst 6500 Family Switch, use the **mls ip slb wildcard search rp** command to reduce the probability of exceeding the capacity of the TCAM on the PFC. See the“Configuring a Wildcard Search” section on page 97 for more details.

Configuring a Virtual Server

Perform this task to configure a virtual server.

IOS SLB supports up to 500 virtual servers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb vserver** *virtual-server*
4. **virtual** *ip-address* [*netmask* [**group**]] {**esp** | **gre** | *protocol*}

   or

   virtual *ip-address* [*netmask* [**group**]] {**tcp** | **udp**} [*port* | **any**] [**service** *service*]
5. **serverfarm** *primary-farm* [**backup** *backup-farm* [**sticky**]] [**map** *map-id* **priority** *priority*]
6. **access** *interface* [**route framed-ip**]
7. **advertise** [**active**]
8. **client** {*ip-address netmask* [**exclude**] | **gtp carrier-code** [*code*]}
9. **delay** {*duration* | **radius framed-ip** *duration*}
10. **gtp notification cac** [*reassign-count*]
11. **hand-off radius** *duration*
12. **idle** [**asn r6 request** | **gtp request** | **ipmobile request** | **radius** {**request** | **framed-ip**}] *duration*
13. **purge radius framed-ip acct on-off**
14. **purge radius framed-ip acct stop** {*attribute-number* | {**26** | *vsa*} {*vendor-ID* | **3gpp** | **3gpp2**} *sub-attribute-number*}
15. **radius acct local-ack key** [*encrypt*] *secret-string*
16. **radius inject auth** *group-number* {**calling-station-id** | **username**}
17. **radius inject auth timer** *seconds*
18. **radius inject auth vsa** *vendor-id*
19. **replicate casa** *listen-ip remote-ip port* [*interval*] [**password** [*encrypt*] *secret-string timeout*]
20. **replicate interval** *interval*
21. **replicate slave**
22. **sticky** {*duration* [**group** *group-id*] [**netmask** *netmask*] | **gtp imsi** [**group** *group-id*] |
    radius calling-station-id | **radius framed-ip** [**group** *group-id*] |
    radius username [**msid-cisco**] [**group** *group-id*]}

23. **synguard** *syn-count interval*
24. **inservice** [**standby** *group-name*] [**active**]

DETAILED STEPS

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**ip slb vserver** *virtual-server* **Example:** `Router(config)# ip slb vserver PUBLIC_HTTP`	Identifies a virtual server and enters virtual server configuration mode.
Step 4	**virtual** *ip-address* [*netmask* [**group**]] {**esp** | **gre** | *protocol*} or **virtual** *ip-address* [*netmask* [**group**]] {**tcp** | **udp**} [*port* | **any**] [**service** *service*] **Example:** `Router(config-slb-vserver)# virtual 10.0.0.1 tcp www`	Specifies the virtual server IP address, type of connection, and optional TCP or User Datagram Protocol (UDP) port number, Internet Key Exchange (IKE) or Wireless Session Protocol (WSP) setting, and service coupling. **Note** For RADIUS load balancing, specify the **service radius** keyword option. **Note** For ASN R6 load balancing, specify the **service asn r6** keyword option. **Note** For GPRS load balancing: – Specify a virtual GGSN IP address as the virtual server, and specify the **udp** keyword option. – To load-balance GTP v1 sessions, specify port number 2123, if the GGSNs and SGSNs are in compliance with the ETSI standard, or specify port number 0 or **any** to configure an all-port virtual server (that is, a virtual server that accepts flows destined for all ports). – To load-balance GTP v0 sessions, specify port number 3386, if the GGSNs and SGSNs are in compliance with the ETSI standard, or specify port number 0 or **any** to configure an all-port virtual server. – To enable GPRS load balancing *without* GTP cause code inspection, specify the **service gtp** keyword option. – To enable GPRS load balancing *with* GTP cause code inspection, specify the **service gtp-inspect** keyword option.

	Command	Purpose
Step 5	**serverfarm** *primary-farm* [**backup** *backup-farm* [**sticky**]] [**map** *map-id* **priority** *priority*] **Example:** `Router(config-slb-vserver)# serverfarm SF1 backup SF2 map 1 priority 1`	Associates a real server farm with a virtual server, and optionally configures a backup server farm and specifies that sticky connections are to be used in the backup server farm. **Note** RADIUS load balancing and the Home Agent Director do not support the **sticky** keyword. For GPRS load balancing, if a real server is defined in two or more server farms, each server farm must be associated with a different virtual server. You can associate more than one server farm with a given RADIUS virtual server by configuring more than one **serverfarm** command, each with a unique map ID and a unique priority. (That is, each map ID and each map priority must be unique across all server farms associated with the virtual server.)
Step 6	**access** *interface* [**route framed-ip**] **Example:** `Router(config-slb-vserver)# access Vlan20 route framed-ip`	(Optional) Enables framed-IP routing to inspect the ingress interface.
Step 7	**advertise** [**active**] **Example:** `Router(config-slb-vserver)# advertise`	(Optional) Controls the installation of a static route to the Null0 interface for a virtual server address.

Step 8	**client** {*ip-address netmask* [**exclude**] | **gtp carrier-code** [*code*]}
	Example:
	`Router(config-slb-vserver)# client 10.4.4.0 255.255.255.0`
		(Optional) Specifies which clients are allowed to use the virtual server.
		Note GPRS load balancing supports only the **gtp carrier-code** option, and only if GTP cause code inspection is enabled.

Step 9	**delay** {*duration* | **radius framed-ip** *duration*}
	Example:
	`Router(config-slb-vserver)# delay 30`
		(Optional) Specifies the time IOS Server Load Balancing (IOS SLB) maintains TCP connection context after a connection has terminated.

Step 10	**gtp notification cac** [*reassign-count*] **Example:** `Router(config-slb-vserver)# gtp notification cac 5`	(Optional) Limits the number of times IOS SLB can reassign a session to a new real server for GGSN-IOS SLB messaging.
Step 11	**hand-off radius** *duration* **Example:** `Router(config-slb-vserver)# hand-off radius 30`	(Optional) Changes the amount of time IOS Server Load Balancing (IOS SLB) waits for an ACCT-START message from a new Mobile IP foreign agent in the event of a foreign agent hand-off.

	Command	Purpose
Step 12	**idle** [**asn r6 request** | **gtp request** | **ipmobile request** | **radius** {**request** | **framed-ip**}] *duration* **Example:** `Router(config-slb-vserver)# idle 120`	(Optional) Specifies the minimum time IOS Server Load Balancing (IOS SLB) maintains connection context in the absence of packet activity. **Note** In GPRS load balancing *without* GTP cause code inspection enabled, specify an idle timer greater than the longest possible interval between PDP context requests on the SGSN.
Step 13	**purge radius framed-ip acct on-off** **Example:** `Router(config-slb-vserver)# purge radius framed-ip acct on-off`	(Optional) Enables IOS SLB to purge entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting ON or OFF message.

Step 14	**purge radius framed-ip acct stop** {*attribute-number* | {**26** | *vsa*} {*vendor-ID* | **3gpp** | **3gpp2**} *sub-attribute-number*}
	Example:
	`Router(config-slb-vserver)# purge radius framed-ip acct stop 44`
		(Optional) Enables IOS SLB to purge entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting-Stop message.

Step 15	**radius acct local-ack key** [*encrypt*] *secret-string* **Example:** `Router(config-slb-vserver)# radius acct local-ack key SECRET_PASSWORD`	(Optional) Enables a RADIUS virtual server to acknowledge RADIUS accounting messages.

Step 16	**radius inject auth** *group-number* {**calling-station-id** | **username**}
	Example:
	`Router(config-slb-vserver)# radius inject auth 1 calling-station-id`
		(Optional) Configures a vendor-specific attribute (VSA) correlation group for an IOS SLB RADIUS load balancing accelerated data plane forwarding authentication virtual server, and specifies whether IOS SLB is to create VSA correlation entries based on RADIUS calling station IDs or RADIUS usernames.

Step 17	**radius inject auth timer** *seconds* **Example:** `Router(config-slb-vserver)# radius inject auth timer 45`	(Optional) Configures a timer for vendor-specific attribute (VSA) correlation for an IOS SLB RADIUS load balancing accelerated data plane forwarding authentication virtual server.
Step 18	**radius inject auth vsa** *vendor-id* **Example:** `Router(config-slb-vserver)# radius inject auth vsa vendor1`	(Optional) Buffers vendor-specific attributes (VSAs) for VSA correlation for an IOS SLB RADIUS load balancing accelerated data plane forwarding authentication virtual server.

	Command	Purpose
Step 19	**replicate casa** *listen-ip remote-ip port* [*interval*] [**password** [*encrypt*] *secret-string timeout*] **Example:** `Router(config-slb-vserver)# replicate casa 10.10.10.11 10.10.11.12 4231`	(Optional) Configures a stateful backup of IOS Server Load Balancing (IOS SLB) decision tables to a backup switch. **Note** The Home Agent Director does not support this command. If you specify the **service gtp** keyword on the **virtual** command, and you do not specify the **gtp imsi** keyword on the **sticky** command, the **replicate casa** command is not supported (because sessions are not persistent, and there is nothing to replicate).
Step 20	**replicate interval** *interval* **Example:** `Router(config-slb-vserver)# replicate interval 20`	(Optional) Sets the replication delivery interval for an IOS Server Load Balancing (IOS SLB) virtual server. **Note** The Home Agent Director does not support this command. If you specify the **service gtp** keyword on the **virtual** command, and you do not specify the **gtp imsi** keyword on the **sticky** command, the **replicate casa** command is not supported (because sessions are not persistent, and there is nothing to replicate).
Step 21	**replicate slave** **Example:** `Router(config-slb-vserver)#` **`replicate slave`**	(Optional) Enables stateful backup of redundant route processors for an IOS Server Load Balancing (IOS SLB) virtual server. **Note** The Home Agent Director does not support this command. If you specify the **service gtp** keyword on the **virtual** command, and you do not specify the **gtp imsi** keyword on the **sticky** command, the **replicate casa** command is not supported (because sessions are not persistent, and there is nothing to replicate). If you are using a single Supervisor with **replicate slave** configured, you might receive out-of-sync messages on the Supervisor.

Step 22	**sticky** {*duration* [**group** *group-id*] [**netmask** *netmask*] | **gtp imsi** [**group** *group-id*] | **radius calling-station-id** | **radius framed-ip** [**group** *group-id*] | **radius username** [**msid-cisco**] [**group** *group-id*]}
	Example:
	`Router(config-slb-vserver)# sticky 60 group 10`
		(Optional) Specifies that connections from the same client use the same real server, as long as the interval between client connections does not exceed the specified duration.
		Note In VPN server load balancing, specify a *duration* of at least 15 seconds.
		GPRS load balancing and the Home Agent Director do not support this command.

	Command	Purpose
Step 23	**synguard** *syn-count interval* **Example:** `Router(config-slb-vserver)# synguard 50`	(Optional) Specifies the rate of TCP SYNchronize sequence numbers (SYNs) handled by a virtual server in order to prevent a SYN flood denial-of-service attack. **Note** GPRS load balancing and the Home Agent Director do not support this command.
Step 24	**inservice** [**standby** *group-name*] [**active**] **Example:** `Router(config-slb-vserver)# inservice`	Enables the virtual server for use by IOS Server Load Balancing (IOS SLB).

Verifying a Virtual Server

Perform the following task to verify a virtual server.

SUMMARY STEPS

1. **show ip slb vservers**

DETAILED STEPS

The following **show ip slb vservers** command verifies the configuration of the virtual servers PUBLIC_HTTP and RESTRICTED_HTTP:

```
Router# show ip slb vservers

slb vserver      prot  virtual               state         conns
-------------------------------------------------------------------
PUBLIC_HTTP      TCP   10.0.0.1:80           OPERATIONAL     0
RESTRICTED_HTTP  TCP   10.0.0.2:80           OPERATIONAL     0
Router#
```

Verifying a Server Farm

Perform the following task to verify a server farm.

SUMMARY STEPS

1. **show ip slb reals**
2. **show ip slb serverfarm**

DETAILED STEPS

The following **show ip slb reals** command displays the status of server farms PUBLIC and RESTRICTED, the associated real servers, and their status:

```
Router# show ip slb real

real                    farm name        weight   state          conns
---------------------------------------------------------------------
10.1.1.1                 PUBLIC           8       OPERATIONAL      0
10.1.1.2                 PUBLIC           8       OPERATIONAL      0
10.1.1.3                 PUBLIC           8       OPERATIONAL      0
```

```
10.1.1.20                RESTRICTED       8       OPERATIONAL      0
10.1.1.21                RESTRICTED       8       OPERATIONAL      0
Router#
```

The following **show ip slb serverfarm** command displays the configuration and status of server farms PUBLIC and RESTRICTED:

```
Router# show ip slb serverfarm

server farm      predictor    nat   reals   bind id
---------------------------------------------------
PUBLIC           ROUNDROBIN   none  3       0
RESTRICTED       ROUNDROBIN   none  2       0
Router#
```

Verifying the Clients

Perform the following task to verify the clients.

SUMMARY STEPS

1. **show ip slb conns**

DETAILED STEPS

The following **show ip slb conns** command verifies the restricted client access and status:

```
Router# show ip slb conns

vserver         prot client                real                  state     nat
-------------------------------------------------------------------------------
RESTRICTED_HTTP TCP  10.4.4.0:80           10.1.1.20             CLOSING   none
Router#
```

The following **show ip slb conns** command displays detailed information about the restricted client access status:

```
Router# show ip slb conns client 10.4.4.0 detail
VSTEST_UDP, client = 10.4.4.0:80
  state = CLOSING, real = 10.1.1.20, nat = none
  v_ip = 10.0.0.2:80, TCP, service = NONE
  client_syns = 0, sticky = FALSE, flows attached = 0
Router#
```

Verifying IOS SLB Connectivity

Perform the following task to verify IOS SLB connectivity.

SUMMARY STEPS

1. **show ip slb stats**

DETAILED STEPS

To verify that the IOS SLB feature has been installed and is operating correctly, ping the real servers from the IOS SLB switch, then ping the virtual servers from the clients.

The following **show ip slb stats** command displays detailed information about the IOS SLB network status:

```
Router# show ip slb stats

 Pkts via normal switching:  0
 Pkts via special switching: 6
 Pkts dropped:               0
 Connections Created:        1
 Connections Established:    1
 Connections Destroyed:      0
 Connections Reassigned:     0
 Zombie Count:               0
 Connections Reused:         0
```

Normal switching is when IOS SLB packets are handled on normal IOS switching paths (CEF, fast switching, and process level switching). Special switching is when IOS SLB packets are handled on hardware-assisted switching paths.

See the "Monitoring and Maintaining IOS SLB" section on page 100 for additional commands used to verify IOS SLB networks and connections.

Configuring Firewall Load Balancing

Perform the following task to configure a basic IOS SLB firewall load-balancing network.

IOS SLB firewall load balancing uses probes to detect and recover from failures. You must configure a probe on each real server in the firewall farm. Ping probes are recommended; see the "Configuring a Ping Probe" section on page 58 for more details. If a firewall does not allow ping probes to be forwarded, use HTTP probes instead. See the "Configuring an HTTP Probe" section on page 56 for more details. You can configure more than one probe, in any combination of supported types (DNS, HTTP, TCP, or ping), for each firewall in a firewall farm.

When performing server load balancing and firewall load balancing together on a Catalyst 6500 Family Switch, use the **mls ip slb wildcard search rp** command to reduce the probability of exceeding the capacity of the TCAM on the PFC. See the"Configuring a Wildcard Search" section on page 97 for more details.

This section describes the following IOS SLB firewall load-balancing configuration tasks. Required and optional tasks are indicated.

- Configuring a Firewall Farm, page 47 (Required)
- Verifying a Firewall Farm, page 51 (Optional)
- Verifying Firewall Connectivity, page 52 (Optional)

Configuring a Firewall Farm

Perform the following task to configure a firewall farm.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb firewallfarm** *firewall-farm*
4. **real** *ip-address*

5. **probe** *probe*
6. **weight** *service*
7. **inservice**
8. **access** [**source** *source-ip netmask*] [**destination** *destination-ip netmask*]
9. **predictor hash address** [**port**]
10. **replicate casa** *listen-ip remote-ip port* [*interval*] [**password** [[*encrypt*] *secret-string* [*timeout*]]
11. **replicate interval** *interval*
12. **replicate slave**
13. **protocol tcp**
14. **delay** *duration*
15. **idle** *duration*
16. **maxconns** *maximum-number*
17. **sticky** *duration* [**netmask** *netmask*] [**source** | **destination**]
18. **protocol datagram**
19. **idle** *duration*
20. **maxconns** *maximum-number*
21. **sticky** *duration* [**netmask** *netmask*] [**source** | **destination**]
22. **inservice**

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip slb firewallfarm`** *`firewall-farm`*
	Example:
	`Router(config)# ip slb firewallfarm FIRE1`
		Adds a firewall farm definition to the IOS Server Load Balancing (IOS SLB) configuration and enters firewall farm configuration mode.

Step 4	**`real`** *`ip-address`*
	Example:
	`Router(config-slb-fw)# real 10.1.1.1`
		Identifies a firewall by IP address as a member of a firewall farm and enters real server configuration mode.

	Command	Purpose
Step 5	**probe** *probe* **Example:** `Router(config-slb-fw-real)# probe FireProbe`	Associates a probe with the firewall.
Step 6	**weight** *setting* **Example:** `Router(config-slb-fw-real)# weight 24`	(Optional) Specifies the firewall's workload capacity relative to other firewalls in the firewall farm.
Step 7	**inservice** **Example:** `Router(config-slb-fw-real)# inservice`	Enables the firewall for use by the firewall farm and by IOS Server Load Balancing (IOS SLB).
Step 8	**access** [**source** *source-ip netmask*] [**destination** *destination-ip netmask*] **Example:** `Router(config-slb-fw)# access destination 10.1.6.0 255.255.255.0`	(Optional) Routes specific flows to a firewall farm.
Step 9	**predictor hash address** [**port**] **Example:** `Router(config-slb-fw)# predictor hash address`	(Optional) Specifies whether the source and destination TCP or User Datagram Protocol (UDP) port numbers, in addition to the source and destination IP addresses, are to be used when selecting a firewall.
Step 10	**replicate casa** *listen-ip remote-ip port* [*interval*] [**password** [[*encrypt*] *secret-string* [*timeout*]] **Example:** `Router(config-slb-fw)# replicate casa 10.10.10.11 10.10.11.12 4231`	(Optional) Configures a stateful backup of IOS Server Load Balancing (IOS SLB) firewall load balancing decision tables to a backup switch.
Step 11	**replicate interval** *interval* **Example:** `Router(config-slb-fw)# replicate interval 20`	(Optional) Sets the replication delivery interval for an IOS Server Load Balancing (IOS SLB) firewall farm. **Note** The Home Agent Director does not support this command. If you specify the **service gtp** keyword on the **virtual** command, and you do not specify the **gtp imsi** keyword on the **sticky** command, the **replicate casa** command is not supported (because sessions are not persistent, and there is nothing to replicate).

	Command	Purpose
Step 12	**replicate slave** **Example:** Router(config-slb-fw)# replicate slave	(Optional) Enables stateful backup of redundant route processors for an IOS Server Load Balancing (IOS SLB) firewall farm. **Note** The Home Agent Director does not support this command. If you specify the **service gtp** keyword on the **virtual** command, and you do not specify the **gtp imsi** keyword on the **sticky** command, the **replicate casa** command is not supported (because sessions are not persistent, and there is nothing to replicate). If you are using a single Supervisor with **replicate slave** configured, you might receive out-of-sync messages on the Supervisor.
Step 13	**protocol tcp** **Example:** Router(config-slb-fw)# protocol tcp	(Optional) Enters firewall farm TCP protocol configuration mode.
Step 14	**delay** *duration* **Example:** Router(config-slb-fw-tcp)# delay 30	(Optional) For firewall farm TCP protocol configuration mode, specifies the time IOS Server Load Balancing (IOS SLB) firewall load balancing maintains TCP connection context after a connection has terminated.
Step 15	**idle** *duration* **Example:** Router(config-slb-fw-tcp)# idle 120	(Optional) For firewall farm TCP protocol configuration mode, specifies the minimum time IOS Server Load Balancing (IOS SLB) firewall load balancing maintains connection context in the absence of packet activity.
Step 16	**maxconns** *maximum-number* **Example:** Router(config-slb-fw-tcp)# maxconns 1000	(Optional) For firewall farm TCP protocol configuration mode, specifies the maximum number of active TCP connections allowed on the firewall farm at one time.
Step 17	**sticky** *duration* [**netmask** *netmask*] [**source** | **destination**] **Example:** Router(config-slb-fw-tcp)# sticky 60	(Optional) For firewall farm TCP protocol configuration mode, specifies that connections from the same IP address use the same firewall if either of the following conditions is met: • As long as any connection between the same pair of IP addresses exists (source/destination sticky). • For a period, defined by *duration*, after the last connection is destroyed.

	Command	Purpose

Step 18	`protocol datagram`
	Example:
	`Router(config-slb-fw)# protocol datagram`
		(Optional) Enters firewall farm datagram protocol configuration mode.

Step 19	**idle** *duration* **Example:** `Router(config-slb-fw-udp)# idle 120`	(Optional) For firewall farm datagram protocol configuration mode, specifies the minimum time IOS Server Load Balancing (IOS SLB) firewall load balancing maintains connection context in the absence of packet activity.
Step 20	**maxconns** *maximum-number* **Example:** `Router(config-slb-fw-udp)# maxconns 1000`	(Optional) For firewall farm datagram protocol configuration mode, specifies the maximum number of active datagram connections allowed on the firewall farm at one time.
Step 21	**sticky** *duration* [**netmask** *netmask*] [**source** | **destination**] **Example:** `Router(config-slb-fw-udp)# sticky 60`	(Optional) For firewall farm datagram protocol configuration mode, specifies that connections from the same IP address use the same firewall if either of the following conditions is met: • As long as any connection between the same pair of IP addresses exists (source/destination sticky). • For a period, defined by *duration*, after the last connection is destroyed.
Step 22	**inservice** **Example:** `Router(config-slb-fw)# inservice`	Enables the firewall farm for use by IOS Server Load Balancing (IOS SLB).

Verifying a Firewall Farm

Perform the following task to verify a firewall farm.

SUMMARY STEPS

1. **show ip slb real**
2. **show ip slb firewallfarm**

DETAILED STEPS

The following **show ip slb reals** command displays the status of firewall farm FIRE1, the associated real servers, and their status:

```
Router# show ip slb real

real                  farm name        weight   state          conns
-------------------------------------------------------------------
10.1.1.2              FIRE1            8        OPERATIONAL    0
10.1.2.2              FIRE1            8        OPERATIONAL    0
```

The following **show ip slb firewallfarm** command displays the configuration and status of firewall farm FIRE1:

```
Router# show ip slb firewallfarm

firewall farm    hash        state         reals
------------------------------------------------
FIRE1            IPADDR      INSERVICE     2
```

Verifying Firewall Connectivity

Perform the following task to verify firewall connectivity.

SUMMARY STEPS

1. Ping the external real servers.
2. Ping the internal real servers.
3. **show ip slb stats**
4. **show ip slb real detail**
5. **show ip slb conns**

DETAILED STEPS

To verify that IOS SLB firewall load balancing has been configured and is operating correctly, perform the following steps:

Step 1 Ping the external real servers (the ones outside the firewall) from the IOS SLB firewall load-balancing switch.

Step 2 Ping the internal real servers (the ones inside the firewall) from the clients.

Step 3 Use the **show ip slb stats** command to display detailed information about the IOS SLB firewall load-balancing network status:

```
Router# show ip slb stats

 Pkts via normal switching:  0
 Pkts via special switching: 0
 Pkts dropped:               0
 Connections Created:        1911871
 Connections Established:    1967754
 Connections Destroyed:      1313251
 Connections Reassigned:     0
 Zombie Count:               0
 Connections Reused:         59752
 Connection Flowcache Purges:1776582
 Failed Connection Allocs:   17945
 Failed Real Assignments:    0
```

Normal switching is when IOS SLB packets are handled on normal IOS switching paths (CEF, fast switching, and process level switching). Special switching is when IOS SLB packets are handled on hardware-assisted switching paths.

Step 4 Use the **show ip slb real detail** command to display detailed information about the IOS SLB firewall load-balancing real server status:

```
Router# show ip slb real detail

10.1.1.3, FIRE1, state = OPERATIONAL, type = firewall
  conns = 299310, dummy_conns = 0, maxconns = 4294967295
  weight = 10, weight(admin) = 10, metric = 104, remainder = 2
  total conns established = 1074779, hash count = 4646
  server failures = 0
  interface FastEthernet1/0, MAC 0010.f68f.7020
```

Step 5 Use the **show ip slb conns** command to display detailed information about the active IOS SLB firewall load-balancing connections:

```
Router# show ip slb conns

vserver         prot client               real                  state     nat
-------------------------------------------------------------------------------
FirewallTCP     TCP  80.80.50.187:40000   10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000   10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000   10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000   10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000   10.1.1.4              ESTAB     none
```

See the “Monitoring and Maintaining IOS SLB” section on page 100 for additional commands used to verify IOS SLB networks and connections.

Configuring a Probe

Perform the following task to configure a probe.

IOS SLB uses probes to verify connectivity and detect failures. For a detailed description of each type of probe, see the “Probes” section on page 25.

By default, no probes are configured in IOS SLB. The following sections describe how to configure and verify probes. Required and optional tasks are indicated.

- Configuring a Custom UDP Probe, page 53 (Required)
- Configuring a DNS Probe, page 55 (Required)
- Configuring an HTTP Probe, page 56 (Required)
- Configuring a Ping Probe, page 58 (Required)
- Configuring a TCP Probe, page 59 (Required)
- Configuring a WSP Probe, page 60 (Required)
- Associating a Probe, page 61 (Required)
- Verifying a Probe, page 62 (Optional)

Configuring a Custom UDP Probe

Perform the following task to configure a custom UDP probe.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb probe** *probe* **custom udp**
4. **address** [*ip-address*] [**routed**]
5. **faildetect** *number-of-probes*
6. **interval** *seconds*
7. **port** *port*
8. **request data** {*start-byte* | **continue**} *hex-data-string*
9. **response** *clause-number* **data** *start-byte hex-data-string*
10. **timeout** *seconds*

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb probe** *probe* **custom udp** **Example:** `Router(config)# ip slb probe PROBE6 custom udp`	Configures the IOS Server Load Balancing (IOS SLB) probe name and enters custom User Datagram Protocol (UDP) probe configuration mode.
Step 4	**address** [*ip-address*] [**routed**] **Example:** `Router(config-slb-probe)# address 10.1.1.1`	(Optional) Configures an IP address to which to send the custom User Datagram Protocol (UDP) probe.
Step 5	**faildetect** *number-of-probes* **Example:** `Router(config-slb-probe)# faildetect 16`	(Optional) Specifies the number of consecutive unacknowledged custom User Datagram Protocol (UDP) probes that constitute failure of the real server.
Step 6	**interval** *seconds* **Example:** `Router(config-slb-probe)# interval 11`	(Optional) Configures the custom User Datagram Protocol (UDP) probe transmit timers.

	Command	Description
Step 7	**port** *port* **Example:** `Router(config-slb-probe)# port 8`	Configures the port to which the custom User Datagram Protocol (UDP) probe is to connect.
Step 8	**request data** {*start-byte* | **continue**} *hex-data-string* **Example:** `Router(config-slb-probe)# request data 0 05 04 00 77 18 2A D6 CD 0A AD 53 4D F1 29 29 CF C1 96 59 CB`	Defines the payload of the User Datagram Protocol (UDP) request packet to be sent by a custom UDP probe.
Step 9	**response** *clause-number* **data** *start-byte hex-data-string* **Example:** `Router(config-slb-probe)# response 2 data 44 DD DD`	Defines the data string to match against custom User Datagram Protocol (UDP) probe response packets.
Step 10	**timeout** *seconds* **Example:** `Router(config-slb-probe)# timeout 20`	(Optional) Sets a timeout for custom User Datagram Protocol (UDP) probes.

Configuring a DNS Probe

Perform the following task to configure a DNS probe.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb probe** *probe* **dns**
4. **address** [*ip-address* [**routed**]]
5. **faildetect** *number-of-probes*
6. **interval** *seconds*
7. **lookup** *ip-address*

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb probe** *probe* **dns** **Example:** `Router(config)# ip slb probe PROBE4 dns`	Configures the IOS Server Load Balancing (IOS SLB) probe name and enters Domain Name System (DNS) probe configuration mode.
Step 4	**address** [*ip-address* [**routed**]] **Example:** `Router(config-slb-probe)# address 10.1.10.1`	(Optional) Configures an IP address to which to send the Domain Name System (DNS) probe.
Step 5	**faildetect** *number-of-probes* **Example:** `Router(config-slb-probe)# faildetect 16`	(Optional) Specifies the number of consecutive unacknowledged Domain Name System (DNS) probes that constitute failure of the real server or firewall.
Step 6	**interval** *seconds* **Example:** `Router(config-slb-probe)# interval 11`	(Optional) Configures the Domain Name System (DNS) probe transmit timers.
Step 7	**lookup** *ip-address* **Example:** `Router(config-slb-probe)# lookup 10.1.10.1`	(Optional) Configures an IP address of a real server that a Domain Name System (DNS) server should supply in response to a domain name resolve request.

Configuring an HTTP Probe

Perform the following task to configure an HTTP probe.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb probe** *probe* **http**
4. **address** [*ip-address* [**routed**]]
5. **credentials** {*username* [*password*]}
6. **expect** [**status** *status-code*] [**regex** *expression*]

7. **header** *field-name* [*field-value*]
8. **interval** *seconds*
9. **port** *port*
10. **request** [**method** {**get** | **post** | **head** | **name** *name*}] [**url** *path*]
11. Configure a route to the virtual server.

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb probe** *probe* **http** **Example:** `Router(config)# ip slb probe PROBE2 http`	Configures the IOS Server Load Balancing (IOS SLB) probe name and enters HTTP probe configuration mode.
Step 4	**address** [*ip-address* [**routed**]] **Example:** `Router(config-slb-probe)# address 10.1.10.1`	(Optional) Configures an IP address to which to send the HTTP probe.

Step 5	**credentials** {*username* [*password*]}
	Example:
	`Router(config-slb-probe)# credentials Username1 password`
		(Optional) Configures header values for the HTTP probe.

Step 6	**expect** [**status** *status-code*] [**regex** *expression*] **Example:** `Router(config-slb-probe)# expect status 401 regex Copyright`	(Optional) Configures the expected HTTP status code or regular expression.
Step 7	**header** *field-name* [*field-value*] **Example:** `Router(config-slb-probe)# header HeaderName HeaderValue`	(Optional) Configures header values for the HTTP probe.
Step 8	**interval** *seconds* **Example:** `Router(config-slb-probe)# interval 11`	(Optional) Configures the HTTP probe transmit timers.

	Command	Description
Step 9	**port** *port* **Example:** `Router(config-slb-probe)# port 8`	(Optional) Configures the port to which the HTTP probe is to connect.
Step 10	**request** [**method** {**get** | **post** | **head** | **name** *name*}] [**url** *path*] **Example:** `Router(config-slb-probe)# request method post url /probe.cgi?all`	(Optional) Configures the URL path to request from the server, and the method used to perform the request to the server.

In addition, HTTP probes require a route to the virtual server. The route is not used, but it must exist to enable the sockets code to verify that the destination can be reached, which in turn is essential for HTTP probes to function correctly. The route can be either a host route (advertised by the virtual server) or a default route (specified using the **ip route 0.0.0.0 0.0.0.0** command, for example).

Configuring a Ping Probe

Perform the following task to configure a ping probe.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb probe** *probe* **ping**
4. **address** [*ip-address* [**routed**]]
5. **faildetect** *number-of-pings*
6. **interval** *seconds*

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Description
Step 3	**ip slb probe** *probe* **ping** **Example:** `Router(config)# ip slb probe PROBE1 ping`	Configures the IOS Server Load Balancing (IOS SLB) probe name and enters ping probe configuration mode.
Step 4	**address** [*ip-address* [**routed**]] **Example:** `Router(config-slb-probe)# address 10.1.10.1`	(Optional) Configures an IP address to which to send the ping probe.
Step 5	**faildetect** *number-of-pings* **Example:** `Router(config-slb-probe)# faildetect 16`	(Optional) Specifies the number of consecutive unacknowledged pings that constitute failure of the real server or firewall.
Step 6	**interval** *seconds* **Example:** `Router(config-slb-probe)# interval 11`	(Optional) Configures the ping probe transmit timers.

Configuring a TCP Probe

Perform the following task to configure a TCP probe.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb probe** *probe* **tcp**
4. **address** [*ip-address* [**routed**]]
5. **interval** *seconds*
6. **port** *port*

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Description
Step 3	**ip slb probe** *probe* **tcp** **Example:** `Router(config)# ip slb probe PROBE5 tcp`	Configures the IOS Server Load Balancing (IOS SLB) probe name and enters TCP probe configuration mode.
Step 4	**address** [*ip-address* [**routed**]] **Example:** `Router(config-slb-probe)# address 10.1.10.1`	(Optional) Configures an IP address to which to send the TCP probe.
Step 5	**interval** *seconds* **Example:** `Router(config-slb-probe)# interval 5`	(Optional) Configures the TCP probe transmit timers.
Step 6	**port** *port* **Example:** `Router(config-slb-probe)# port 8`	Configures the port to which the TCP probe is to connect.

Configuring a WSP Probe

Perform the following task to configure a WSP probe.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb probe** *probe* **wsp**
4. **address** [*ip-address* [**routed**]]
5. **interval** *seconds*
6. **url** [*path*]

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Description

Step 3	**`ip slb probe`** *`probe`* **`wsp`**
	Example:
	`Router(config)# ip slb probe PROBE3 wsp`
		Configures the IOS Server Load Balancing (IOS SLB) probe name and enters Wireless Session Protocol (WSP) probe configuration mode.

Step 4	**`address`** `[`*`ip-address`* `[`**`routed`**`]]`
	Example:
	`Router(config-slb-probe)# address 10.1.10.1`
		(Optional) Configures an IP address to which to send the Wireless Session Protocol (WSP) probe.

Step 5	**`interval`** *`seconds`*
	Example:
	`Router(config-slb-probe)# interval 11`
		(Optional) Configures the Wireless Session Protocol (WSP) probe transmit timers.

Step 6	**`url`** `[`*`path`*`]`
	Example:
	`Router(config-slb-probe)# url http://localhost/test.txt`
		(Optional) Configures the Wireless Session Protocol (WSP) probe URL path.

Associating a Probe

Perform the following task to associate a probe with a real server or firewall.

After configuring a probe, you must associate it with a real server or firewall, using the **probe** command. See the "Configuring a Server Farm and a Real Server" section on page 36 and the "Configuring Firewall Load Balancing" section on page 47 for more details.

Note You cannot associate a WSP probe with a firewall.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb firewallfarm** *firewall-farm*

   or

   ip slb serverfarm *server-farm*
4. **probe** *probe*

DETAILED STEPS

	Command	Description

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip slb firewallfarm`** *`firewall-farm`*
	or
	`ip slb serverfarm` *`server-farm`*
	Example:
	`Router(config)# ip slb serverfarm PUBLIC`
	or
	`Router(config)# ip slb firewallfarm FIRE1`
		Identifies a firewall farm and enters firewall farm configuration mode.
		or
		Identifies a server farm and enters SLB server farm configuration mode.

Step 4	**`probe`** *`probe`*
	Example:
	`Router(config-slb-sfarm)# probe PROBE1`
	or
	`Router(config-slb-fw-real)# probe FireProbe`
		Associates a probe with a firewall farm or a server farm.

Verifying a Probe

Perform the following task to verify a probe.

SUMMARY STEPS

1. **show ip slb probe**

DETAILED STEPS

To verify that a probe is configured correctly, use the **show ip slb probe** command:

```
Router# show ip slb probe

Server:Port            State        Outages  Current  Cumulative
----------------------------------------------------------------
10.1.1.1:80            OPERATIONAL        0  never    00:00:00
10.1.1.2:80            OPERATIONAL        0  never    00:00:00
10.1.1.3:80            OPERATIONAL        0  never    00:00:00
```

Configuring DFP

Perform the following task to configure IOS SLB as a DFP manager, and to identify a DFP agent with which IOS SLB can initiate connections.

You can define IOS SLB as a DFP manager, as a DFP agent for another DFP manager, or as both at the same time. Depending on your network configuration, you might enter the commands for configuring IOS SLB as a DFP manager and the commands for configuring IOS SLB as a DFP agent on the same device or on different devices.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb dfp** [**password** [[*encrypt*] *secret-string* [*timeout*]]
4. **agent** *ip-address port* [*timeout* [*retry-count* [*retry-interval*]]]

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb dfp** [**password** [[*encrypt*] *secret-string* [*timeout*]] **Example:** `Router(config)# ip slb dfp password Password1 360`	Configures Dynamic Feedback Protocol (DFP), supplies an optional password, and enters DFP configuration mode.
Step 4	**agent** *ip-address port* [*timeout* [*retry-count* [*retry-interval*]]] **Example:** `Router(config-slb-dfp)# agent 10.1.1.1 2221 30 0 10`	Identifies a Dynamic Feedback Protocol (DFP) agent to which IOS Server Load Balancing (IOS SLB) can connect.

What to Do Next

To configure IOS SLB as a DFP agent, refer to the *DFP Agent Subsystem* feature document for Cisco IOS Release 12.2(18)SXB.

GPRS Load Balancing Configuration Task List

Perform the following tasks to configure GPRS load balancing.

SUMMARY STEPS

1. Configure a server farm and a real server.
2. Configure a virtual server.
3. Configure the virtual IP address as a loopback on each of the GGSNs in the servers.
4. Route each GGSN to each associated SGSN.
5. Route each SGSN to the virtual templates on each associated Cisco GGSN, and to the GPRS load-balancing virtual server.
6. Configure a GSN idle timer.

DETAILED STEPS

	Task	Description
Step 1	Configure a server farm and a real server.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for GPRS load balancing, keep the following considerations in mind: • If GTP cause code inspection is not enabled, accept the default setting (the weighted round robin algorithm) for the **predictor** command. • If GTP cause code inspection is enabled, you can specify either the weighted round robin (**roundrobin**) or the weighted least connections (**leastconns**) algorithm. • Specify the IP addresses (virtual template addresses, for Cisco GGSNs) of the real servers performing the GGSN function, using the **real** command. • Specify a reassign threshold less than the SGSN's N3-REQUESTS counter value, using the **reassign** command.

	Task	Description
Step 2	Configure a virtual server.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the **virtual** command, keep the following considerations in mind: • Specify a virtual GGSN IP address as the virtual server, and specify the **udp** keyword option. • To load-balance GTP v1 sessions, specify port number 2123, if the GGSNs and SGSNs are in compliance with the ETSI standard, or specify port number 0 or **any** to configure an all-port virtual server (that is, a virtual server that accepts flows destined for all ports). • To load-balance GTP v0 sessions, specify port number 3386, if the GGSNs and SGSNs are in compliance with the ETSI standard, or specify port number 0 or **any** to configure an all-port virtual server. • To enable GPRS load balancing *without* GTP cause code inspection, specify the **service gtp** keyword option. • To enable GPRS load balancing *with* GTP cause code inspection, specify the **service gtp-inspect** keyword option. In GPRS load balancing *without* GTP cause code inspection enabled, when you configure the idle timer using the **idle** command, specify an idle timer greater than the longest possible interval between PDP context requests on the SGSN.
Step 3	Configure the virtual IP address as a loopback on each of the GGSNs in the servers.	(Required for dispatched mode) This step is required only if you are using dispatched mode *without* GTP cause code inspection enabled. Refer to the "Configuring Virtual Interfaces" section in the *Cisco IOS Interface Configuration Guide* for more information.
Step 4	Route each GGSN to each associated SGSN.	The route can be static or dynamic, but the GGSN needs to be able to reach the SGSN. Refer to the "Configuring Network Access to the GGSN" section of the *Cisco IOS Mobile Wireless Configuration Guide* for more details.
Step 5	Route each SGSN to the virtual templates on each associated Cisco GGSN, and to the GPRS load-balancing virtual server.	(Required) Refer to the configuration guide for your SGSN for more details.
Step 6	Configure a GSN idle timer.	(Optional) This step is applicable only if GTP cause code inspection is enabled. See the "Configuring a GSN Idle Timer" section on page 66 for more information.

Configuring a GSN Idle Timer

Perform this task to configure a GSN idle timer.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb timers gtp gsn** *duration*

DETAILED STEPS

	Command	Description

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip slb timers gtp gsn`** *`duration`*
	Example:
	`Router(config)# ip slb timers gtp gsn 45`
		Change the amount of time IOS Server Load Balancing (IOS SLB) maintains sessions to and from an idle gateway GPRS support node (GGSN) or serving GPRS support node (SGSN).

GGSN-IOS SLB Messaging Task List

Perform this task to configure GGSN-IOS SLB messaging.

SUMMARY STEPS

1. Configure the GGSN to support GGSN-IOS SLB messaging.
2. Configure a server farm and a real server.
3. Configure a virtual server.

DETAILED STEPS

	Task	Description
Step 1	Configure the GGSN to support GGSN-IOS SLB messaging.	When you configure GGSN-IOS SLB messaging support, configure all IOS SLB virtual servers that share the same GGSN to use the same NAT mode, either dispatched mode or directed mode, using the **gprs slb mode** command. The virtual servers cannot use a mix of dispatched mode and directed mode, because you can configure only one NAT mode on a given GGSN. For more information, refer to the *Cisco IOS Mobile Wireless Configuration Guide* for GGSN 5.0 for Cisco IOS Release 12.3(2)XU or later.
Step 2	Configure a server farm and a real server.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for GGSN-IOS SLB messaging, to prevent IOS SLB from failing the current real server when reassigning the session to a new real server, disable automatic server failure detection by specifying the **no faildetect inband** command.
Step 3	Configure a virtual server.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for GGSN-IOS SLB messaging, specify the **gtp notification cac** command to limit the number of times IOS SLB can reassign a session to a new real server.

Configuring GPRS Load Balancing Maps

Perform this task to configure GPRS load balancing maps.

GPRS load balancing maps enable IOS SLB to categorize and route user traffic based on APNs. To enable maps for GPRS load balancing, you must define a GTP map, then associate the map with a server farm.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb map** *map-id* **gtp** | **radius**}
4. **apn** *string*
5. **exit**
6. **ip slb vserver** *virtual-server*
7. **virtual** *ip-address* [*netmask* [**group**]] {**tcp** | **udp**} [*port* | **any**] [**service** *service*]
8. **serverfarm** *primary-farm* [**backup** *backup-farm* [**sticky**]] [**map** *map-id* **priority** *priority*]

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ip slb map** *map-id* **gtp** | **radius**}
	Example:
	`Router(config)# ip slb map 1 radius`
		Configures an IOS SLB GTP map and enters SLB GTP map configuration mode.

Step 4	**apn** *string* **Example:** `Router(config-slb-map-gtp)# apn abc`	Configures an ASCII regular expression string to be matched against the access point name (APN) for general packet radio service (GPRS) load balancing.
Step 5	**exit** **Example:** `Router(config-slb-map-gtp)# exit`	Exits SLB GTP map configuration mode.
Step 6	**ip slb vserver** *virtual-server* **Example:** `Router(config)# ip slb vserver GGSN_SERVER`	Identifies a virtual server and enters virtual server configuration mode.

Step 7	**`virtual`** *`ip-address`* `[`*`netmask`* `[`**`group`**`]]`
	`{`**`tcp`** `|` **`udp`**`}` `[`*`port`* `|` **`any`**`]` `[`**`service`** *`service`*`]`
	Example:
	`Router(config-slb-vserver)# virtual 10.10.10.10 udp 0 service gtp`
		Specifies the virtual server IP address, type of connection, and optional TCP or User Datagram Protocol (UDP) port number, Internet Key Exchange (IKE) or Wireless Session Protocol (WSP) setting, and service coupling.
		Note For GPRS load balancing:
		– Specify a virtual GGSN IP address as the virtual server, and specify the **udp** keyword option.
		– To load-balance GTP v1 sessions, specify port number 2123, if the GGSNs and SGSNs are in compliance with the ETSI standard, or specify port number 0 or **any** to configure an all-port virtual server (that is, a virtual server that accepts flows destined for all ports).
		– To load-balance GTP v0 sessions, specify port number 3386, if the GGSNs and SGSNs are in compliance with the ETSI standard, or specify port number 0 or **any** to configure an all-port virtual server.
		– To enable GPRS load balancing *without* GTP cause code inspection, specify the **service gtp** keyword option.
		– To enable GPRS load balancing *with* GTP cause code inspection, specify the **service gtp-inspect** keyword option.

Step 8	**`serverfarm`** *`primary-farm`*
	`[`**`backup`** *`backup-farm`* `[`**`sticky`**`]]`
	`[`**`map`** *`map-id`* **`priority`** *`priority`*`]`
	Example:
	`Router(config-slb-vserver)# serverfarm farm1 backup farm2 map 1 priority 3`
		Associates a GTP map with a server farm. Associates a real server farm with a virtual server, and optionally configures a backup server farm and specifies that sticky connections are to be used in the backup server farm.
		Note For GPRS load balancing, if a real server is defined in two or more server farms, each server farm must be associated with a different virtual server.
		You can associate more than one server farm with a given virtual server by configuring more than one **serverfarm** command, each with a unique map ID and a unique priority. (That is, each map ID and each map priority must be unique across all server farms associated with the virtual server.
		If you are using GTP maps, and you have configured a given real server in more than one server farm, you must associate a different virtual server with each server farm.

Configuring KeepAlive Application Protocol (KAL-AP) Agent Support

Perform this task to configure KAL-AP agent support.

KAL-AP agent support enables IOS SLB to perform load balancing in a global server load balancing (GSLB) environment.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb capp udp**
4. **peer** [*ip-address*] **port** *port*
5. **peer** [*ip-address*] **secret** [*encrypt*] *secret-string*
6. **exit**
7. **ip slb serverfarm** *server-farm*
8. **kal-ap domain** *tag*
9. **farm-weight** *setting*

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb capp udp** **Example:** `Router(config)# ip slb capp udp`	Enables the KAL-AP agent and enters SLB Content Application Peering Protocol (CAPP) configuration mode.
Step 4	**peer** [*ip-address*] **port** *port* **Example:** `Router(config-slb-capp)# peer port 6000`	(Optional) Specifies the port to which the KAL-AP agent is to connect.
Step 5	**peer** [*ip-address*] **secret** [*encrypt*] *secret-string* **Example:** `Router(config-slb-capp)# peer secret SECRET_STRING`	(Optional) Enables Message Digest Algorithm Version 5 (MD5) authentication for the KAL-AP agent.
Step 6	**exit** **Example:** `Router(config-slb-map-gtp)# exit`	Exits SLB CAPP configuration mode.

Step 7	**`ip slb serverfarm`** *`server-farm`*
	Example:
	`Router(config)# ip slb serverfarm PUBLIC`
		Identifies a server farm and enters SLB server farm configuration mode.

Step 8	**`kal-ap domain`** *`tag`*
	Example:
	`Router(config-slb-sfarm)# kal-ap domain chicago-com`
		(Optional) Enables the KAL-AP agent to look for a domain tag when reporting the load for a virtual server.

Step 9	**`farm-weight`** *`setting`*
	Example:
	`Router(config-slb-sfarm)# farm-weight 16`
		(Optional) Specifies a weight to be used by the KAL-AP agent when calculating the load value for a server farm.

RADIUS Load Balancing Configuration Task List

Perform this task to configure RADIUS load balancing.

SUMMARY STEPS

1. Configure the GGSN to support GGSN-IOS SLB messaging.
2. Configure a server farm and a real server.
3. Configure a virtual server.
4. Enable IOS SLB to inspect packets for RADIUS framed-IP sticky routing.
5. Configure RADIUS load balancing maps.
6. Configure RADIUS load balancing accelerated data plane forwarding.
7. Increase the number of available MLS entries.
8. Configure a probe.

DETAILED STEPS

	Task	Description
Step 1	Configure a server farm and a real server.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for RADIUS load balancing, keep the following considerations in mind: • Accept the default setting (the weighted round robin algorithm) for the **predictor** command. • (Optional) Specify a value of 1 for the **numclients** keyword on the **faildetect numconns** command, if you want to enable session-based failure detection. • (Optional) To specify the maximum number of IOS SLB RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server, use the **maxclients** command.

	Task	Description
Step 2	Configure a virtual server.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for RADIUS load balancing, keep the following considerations in mind: • Specify the **service radius** keyword option, using the **virtual** command. • (Optional) To enable framed-IP routing to inspect the ingress interface, specify the **access** *interface* **route framed-ip** command. If you configure the **access** *interface* **route framed-ip** command, you must also configure the **virtual** command with the **service radius** keywords specified. • (Optional) To change the amount of time IOS SLB waits for an ACCT-START message from a new Mobile IP foreign agent in the event of a foreign agent hand-off, configure a **hand-off radius** command. • (Optional) To set a duration for RADIUS entries in the IOS SLB session database, configure an **idle** command with the **radius request** keywords specified. • (Optional) To set a duration for entries in the IOS SLB RADIUS framed-IP sticky database, configure an **idle** command with the **radius framed-ip** keywords specified.
	Configure a virtual server. (continued)	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for RADIUS load balancing, keep the following considerations in mind: • (Optional) To enable IOS SLB to create the IOS SLB RADIUS framed-IP sticky database and direct RADIUS requests and non-RADIUS flows from a given subscriber to the same service gateway, specify the **radius framed-ip** keywords on the **sticky** command. If you configure the **sticky radius framed-ip** command, you must also configure the **virtual** command with the **service radius** keywords specified. • (Optional) To enable IOS SLB to purge entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting ON or OFF message, specify the **purge radius framed-ip acct on-off** virtual server configuration command. To prevent IOS SLB from purging entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting ON or OFF message, specify the **no purge radius framed-ip acct on-off** virtual server configuration command. • (Optional) To enable IOS SLB to purge entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting-Stop message, specify the **purge radius framed-ip acct stop** virtual server configuration command. To prevent IOS SLB from purging entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting-Stop message, specify the **no purge radius framed-ip acct stop** virtual server configuration command.

	Task	Description
	Configure a virtual server. (continued)	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for RADIUS load balancing, keep the following considerations in mind: • (Optional—for CDMA2000 networks only) To enable IOS SLB to create the IOS SLB RADIUS calling-station-ID sticky database and direct RADIUS requests from a given subscriber to the same service gateway based on the calling station ID, specify the **radius calling-station-id** keywords on the **sticky** command. To enable IOS SLB to create the IOS SLB RADIUS username sticky database and direct RADIUS requests from a given subscriber to the same service gateway based on the username, specify the **radius username** keywords on the **sticky** command. If you configure the **sticky radius calling-station-id** command or the **sticky radius username** command, you must also configure the **virtual** command with the **service radius** keywords specified, and you must configure the **sticky radius framed-ip** command. You cannot configure both the **sticky radius calling-station-id** command and the **sticky radius username** command on the same virtual server. • (Optional—for RADIUS load balancing accelerated data plane forwarding only) To configure a VSA correlation group for an authentication virtual server, and to specify whether IOS SLB is to create VSA correlation entries based on RADIUS calling station IDs or RADIUS usernames, configure the **radius inject auth** command. To configure a timer for VSA correlation for an authentication virtual server, configure the **radius inject auth timer** command. To buffer VSAs for VSA correlation for an authentication virtual server, configure the **radius inject auth vsa** command. To configure a VSA correlation group for an accounting virtual server, and to enable Message Digest Algorithm Version 5 (MD5) authentication for VSA correlation, configure the **radius inject acct** command.
Step 3	Enable IOS SLB to inspect packets for RADIUS framed-IP sticky routing.	(Optional) See the "Enabling IOS SLB to Inspect Packets for RADIUS Framed-IP Sticky Routing" section on page 74.
Step 4	Configure RADIUS load balancing maps.	(Optional) See the "Configuring RADIUS Load Balancing Maps" section on page 75.
Step 5	Configure RADIUS load balancing accelerated data plane forwarding.	(Optional) See the "Configuring RADIUS Load Balancing Accelerated Data Plane Forwarding" section on page 76.

	Task	Description
Step 6	Increase the number of available MLS entries.	(Optional) If you are running IOS SLB in dispatched mode on a Catalyst 6500 Family Switch with Supervisor Engine 2, you can improve performance by configuring the **no mls netflow** command. This command increases the number of MLS entries available for hardware switching of end-user flows. **Note** If you are using IOS features that use the hardware NetFlow table, such as micro-flow QoS, reflexive ACLs, TCP intercept, or Web Cache Redirect, do not configure the **no mls netflow** command. For more information about configuring MLS NetFlow, refer to the *Catalyst 6000 Family IOS Software Configuration Guide*.
Step 7	Configure a probe.	See the "Configuring a Probe" section on page 53. To verify the health of the server, configure a ping probe.

Enabling IOS SLB to Inspect Packets for RADIUS Framed-IP Sticky Routing

Perform this task to enable IOS SLB to inspect packets for RADIUS framed-IP sticky routing.

You can enable IOS SLB to inspect packets whose source IP addresses match a configured IP address and subnet mask. If the source IP address of an inspected packet matches an entry in the IOS SLB RADIUS framed-IP sticky database, IOS SLB uses that entry to route the packet. Otherwise, IOS routes the packet.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb route** {**framed-ip deny** | *ip-address netmask* **framed-ip** | **inter-firewall**}

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ip slb route** {**framed-ip deny** | *ip-address netmask* **framed-ip** | **inter-firewall**}
	Example:
	`Router(config)# ip slb route 10.10.10.1 255.255.255.255 framed-ip`
		Enables IOS Server Load Balancing (IOS SLB) to route packets using the RADIUS framed-IP sticky database, or to route packets from one firewall real server back through another firewall real server.

Configuring RADIUS Load Balancing Maps

Perform this task to configure RADIUS load balancing maps.

RADIUS load balancing maps enable IOS SLB to categorize and route user traffic based on RADIUS calling station IDs and user names. To enable maps for RADIUS load balancing, you must define a RADIUS map, then associate the map with a server farm.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb map** *map-id* **radius**
4. **calling-station-id** *string*
5. **username** *string*
6. **exit**
7. **ip slb vserver** *virtual-server*
8. **virtual** *ip-address* [*netmask* [**group**]] {**tcp** | **udp**} [*port* | **any**] [**service** *service*]
9. **serverfarm** *primary-farm* [**backup** *backup-farm* [**sticky**]] [**map** *map-id* **priority** *priority*]

DETAILED STEPS

	Command	Description

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip slb map`** *`map-id`* **`radius`**
	Example:
	`Router(config)# ip slb map 1 radius`
		Configures an IOS SLB RADIUS map and enters SLB RADIUS map configuration mode.

Step 4	**`calling-station-id`** *`string`*
	Example:
	`Router(config-slb-radius-map)# calling-station-id .919*`
		Configures an ASCII regular expression string to be matched against the calling station ID attribute for RADIUS load balancing.

Step 5	**`username`** *`string`*
	Example:
	`Router(config-slb-map-radius)# )# username ...?525*`
		Configures an ASCII regular expression string to be matched against the username attribute for RADIUS load balancing.

Step 6	**`exit`**
	Example:
	`Router(config-slb-map-gtp)# exit`
		Exits SLB RADIUS map configuration mode.

Step 7	**`ip slb vserver`** *`virtual-server`*
	Example:
	`Router(config)# ip slb vserver GGSN_SERVER`
		Identifies a virtual server and enters virtual server configuration mode.

Step 8	**`virtual`** *`ip-address`* `[`*`netmask`* `[`**`group`**`]] {`**`tcp`** `|` **`udp`**`} [`*`port`* `|` **`any`**`] [`**`service`** *`service`*`]`
	Example:
	`Router(config-slb-vserver)# virtual 10.0.0.1 udp 0 service radius`
		Specifies the virtual server IP address, type of connection, and optional TCP or User Datagram Protocol (UDP) port number, Internet Key Exchange (IKE) or Wireless Session Protocol (WSP) setting, and service coupling.
		Note For RADIUS load balancing, specify the **service radius** keyword option.

Step 9	**`serverfarm`** *`primary-farm`* `[`**`backup`** *`backup-farm`* `[`**`sticky`**`]] [`**`map`** *`map-id`* **`priority`** *`priority`*`]`
	Example:
	`Router(config-slb-vserver)# serverfarm SF1 backup SF2 map 1 priority 1`
		Associates a RADIUS map with a server farm. Associates a real server farm with a virtual server, and optionally configures a backup server farm and specifies that sticky connections are to be used in the backup server farm.
		Note RADIUS load balancing does not support the **sticky** keyword.
		You can associate more than one server farm with a given virtual server by configuring more than one **serverfarm** command, each with a unique map ID and a unique priority. (That is, each map ID and each map priority must be unique across all server farms associated with the virtual server.)

Configuring RADIUS Load Balancing Accelerated Data Plane Forwarding

Perform this task to configure RADIUS load balancing accelerated data plane forwarding.

RADIUS load balancing accelerated data plane forwarding, also known as Turbo RADIUS load balancing, is a high-performance solution that uses basic policy-based routing (PBR) route maps to handle subscriber data-plane traffic in a CSG environment.

Prerequisites

Turbo RADIUS load balancing requires a server farm configured with **predictor route-map** on the accounting virtual server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb serverfarm** *server-farm*
4. **predictor** [**roundrobin** | **leastconns** | **route-map** *mapname*]

5. **ip slb vserver** *virtual-server*
6. **virtual** *ip-address* [*netmask* [**group**]] {**tcp** | **udp**} [*port* | **any**] [**service** *service*]
7. **serverfarm** *primary-farm* [**backup** *backup-farm* [**sticky**]] [**map** *map-id* **priority** *priority*]
8. **radius acct local-ack key** [*encrypt*] *secret-string*
9. **radius inject auth** *group-number* {**calling-station-id** | **username**}
10. **radius inject auth timer** *seconds*
11. **radius inject auth vsa** *vendor-id*

DETAILED STEPS

	Command	Description
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb serverfarm** *server-farm* **Example:** `Router(config)# ip slb serverfarm PUBLIC`	Identifies a server farm and enters SLB server farm configuration mode.
Step 4	**predictor** [**roundrobin** | **leastconns** | **route-map** *mapname*] **Example:** `Router(config-slb-sfarm)# predictor route-map map1`	(Optional) Specifies the algorithm to be used to determine how a real server is selected. Turbo RADIUS load balancing requires the **route-map** keyword and *mapname* argument. When you specify the **predictor route-map** command, no further commands in SLB server farm configuration mode or real server configuration mode are allowed.
Step 5	**exit** **Example:** `Router(config-slb-sfarm)# exit`	Exits SLB server farm configuration mode.
Step 6	**ip slb vserver** *virtual-server* **Example:** `Router(config)# ip slb vserver RADIUS_AUTH`	Identifies a virtual server and enters virtual server configuration mode.

Step 7	**virtual** *ip-address* [*netmask* [**group**]] {**tcp** | **udp**} [*port* | **any**] [**service** *service*] **Example:** `Router(config-slb-vserver)# virtual 10.10.10.10 udp 1813 service radius`	Specifies the virtual server IP address, type of connection, and optional TCP or User Datagram Protocol (UDP) port number, Internet Key Exchange (IKE) or Wireless Session Protocol (WSP) setting, and service coupling and enters SLB virtual server configuration mode. **Note** For RADIUS load balancing, specify the **service radius** keyword option.
Step 8	**serverfarm** *primary-farm* [**backup** *backup-farm* [**sticky**]] [**map** *map-id* **priority** *priority*] **Example:** `Router(config-slb-vserver)# serverfarm AAAFARM`	Associates a RADIUS map with a server farm. Associates a real server farm with a virtual server, and optionally configures a backup server farm and specifies that sticky connections are to be used in the backup server farm. **Note** RADIUS load balancing does not support the **sticky** keyword. You can associate more than one server farm with a given virtual server by configuring more than one **serverfarm** command, each with a unique map ID and a unique priority. (That is, each map ID and each map priority must be unique across all server farms associated with the virtual server.)
Step 9	**radius acct local-ack key** [*encrypt*] *secret-string* **Example:** `Router(config-slb-vserver)# radius acct local-ack key SECRET_PASSWORD`	(Optional) Configures VSA correlation and enables a RADIUS virtual server to acknowledge RADIUS accounting messages **Note** If vendor-specific attribute (VSA) correlation is configured, and if the Cisco VSA is buffered, then the Cisco VSA is injected into the RADIUS Accounting-Start packet. Turbo RADIUS load balancing does not require VSA correlation. This command is valid only for VSA correlation accounting virtual servers.

Step 10	**radius inject auth** *group-number* {**calling-station-id** | **username**}
	Example:
	`Router(config-slb-vserver)# radius inject auth 1 calling-station-id`
		(Optional) Configures a vendor-specific attribute (VSA) correlation group for an IOS SLB RADIUS load balancing accelerated data plane forwarding authentication virtual server, and specifies whether IOS SLB is to create VSA correlation entries based on RADIUS calling station IDs or RADIUS usernames.
		For a given authentication virtual server, you can configure a single **radius inject auth** *group-number* **calling-station-id** command or a single **radius inject auth** *group-number* **username** command, but not both.
		This command is valid only for VSA correlation authentication virtual servers.

Step 11	**radius inject auth timer** *seconds* **Example:** `Router(config-slb-vserver)# radius inject auth timer 45`	(Optional) Configures a timer for vendor-specific attribute (VSA) correlation for an IOS SLB RADIUS load balancing accelerated data plane forwarding authentication virtual server. This command is valid only for VSA correlation authentication virtual servers.
Step 12	**radius inject auth vsa** *vendor-id* **Example:** `Router(config-slb-vserver)# radius inject auth vsa vendor1`	(Optional) Buffers vendor-specific attributes (VSAs) for VSA correlation for an IOS SLB RADIUS load balancing accelerated data plane forwarding authentication virtual server. This command is valid only for VSA correlation authentication virtual servers.

Exchange Director for mSEF Configuration Task List

Perform this task to configure Exchange Director for mSEF.

This section contains the following information:

- RADIUS Configuration for the Exchange Director, page 79
- Firewall Configuration for the Exchange Director, page 82

RADIUS Configuration for the Exchange Director

Perform this task to configure RADIUS load balancing for the Exchange Director.

SUMMARY STEPS

1. Configure a server farm and a real server.
2. Configure a virtual server.
3. Enable IOS SLB to inspect packets for RADIUS framed-IP sticky routing.
4. Configure RADIUS load balancing maps.
5. Increase the number of available MLS entries.
6. Configure a probe.

DETAILED STEPS

	Task	Description
Step 1	Configure a server farm and a real server.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for RADIUS for the Exchange Director, keep the following considerations in mind: • (Optional) Specify a value of 1 for the **numclients** keyword on the **faildetect numconns** command, if you want to enable session-based failure detection. • (Optional) To specify the maximum number of IOS SLB RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server, use the **maxclients** command.
Step 2	Configure a virtual server.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for RADIUS for the Exchange Director, keep the following considerations in mind: • Specify the **service radius** keyword option, using the **virtual** command. • (Optional) To enable framed-IP routing to inspect the ingress interface, specify the **access** *interface* **route framed-ip** command. If you configure the **access** *interface* **route framed-ip** command, you must also configure the **virtual** command with the **service radius** keywords specified. • (Optional) To change the amount of time IOS SLB waits for an ACCT-START message from a new Mobile IP foreign agent in the event of a foreign agent hand-off, configure a **hand-off radius** command. • (Optional) To set a duration for RADIUS entries in the IOS SLB session database, configure an **idle** command with the **radius request** keywords specified. • (Optional) To set a duration for entries in the IOS SLB RADIUS framed-IP sticky database, configure an **idle** command with the **radius framed-ip** keywords specified.

	Task	Description
Step 3	Configure a virtual server. (continued)	See the “Configuring a Virtual Server” procedure on page -40. When you configure the virtual server for RADIUS load balancing, keep the following considerations in mind: • (Optional) To enable IOS SLB to create the IOS SLB RADIUS framed-IP sticky database and direct RADIUS requests and non-RADIUS flows from a given subscriber to the same service gateway, specify the **radius framed-ip** keywords on the **sticky** command. If you configure the **sticky radius framed-ip** command, you must also configure the **virtual** command with the **service radius** keywords specified. • (Optional—for CDMA2000 networks only) To enable IOS SLB to create the IOS SLB RADIUS calling-station-ID sticky database and direct RADIUS requests from a given subscriber to the same service gateway based on the calling station ID, specify the **radius calling-station-id** keywords on the **sticky** command. To enable IOS SLB to create the IOS SLB RADIUS username sticky database and direct RADIUS requests from a given subscriber to the same service gateway based on the username, specify the **radius username** keywords on the **sticky** command. If you configure the **sticky radius calling-station-id** command or the **sticky radius username** command, you must also configure the **virtual** command with the **service radius** keywords specified, and you must configure the **sticky radius framed-ip** command. You cannot configure both the **sticky radius calling-station-id** command and the **sticky radius username** command on the same virtual server.
Step 4	Enable IOS SLB to inspect packets for RADIUS framed-IP sticky routing.	(Optional) See the “Enabling IOS SLB to Inspect Packets for RADIUS Framed-IP Sticky Routing” section on page 74.
Step 5	Configure RADIUS load balancing maps.	(Optional) See the “Configuring RADIUS Load Balancing Maps” section on page 75.
Step 6	Increase the number of available MLS entries.	(Optional) If you are running IOS SLB in dispatched mode on a Catalyst 6500 Family Switch with Supervisor Engine 2, you can improve performance by configuring the **no mls netflow** command. This command increases the number of MLS entries available for hardware switching of end-user flows. **Note** If you are using IOS features that use the hardware NetFlow table, such as micro-flow QoS, reflexive ACLs, TCP intercept, or Web Cache Redirect, do not configure the **no mls netflow** command. For more information about configuring MLS NetFlow, refer to the *Catalyst 6000 Family IOS Software Configuration Guide*.
Step 7	Configure a probe.	See the “Configuring a Probe” section on page 53. To verify the health of the server, configure a ping probe.

Firewall Configuration for the Exchange Director

Perform this task to configure firewall load balancing for the Exchange Director.

This section lists the tasks used to configure firewalls for the Exchange Director. Detailed configuration information is contained in the referenced sections of this or other documents. Required and optional tasks are indicated.

- Configuring a Firewall Farm, page 82 (Required)
- Verifying a Firewall Farm, page 85 (Optional)
- Verifying Firewall Connectivity, page 86 (Optional)
- Configuring a Probe, page 87 (Required)
- Configuring a Wildcard Search, page 87 (Optional)

Configuring a Firewall Farm

Perform the following task to configure a firewall farm.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb firewallfarm** *firewall-farm*
4. **real** *ip-address*
5. **probe** *probe*
6. **weight** *setting*
7. **inservice**
8. **exit**
9. **access** [**source** *source-ip netmask*] [**destination** *destination-ip netmask*]| **inbound** *inbound-interface* | **outbound** *outbound-interface*]
10. **predictor hash address** [**port**]
11. **replicate casa** *listen-ip remote-ip port* [*interval*] [**password** [[*encrypt*] *secret-string* [*timeout*]]
12. protocol tcp
13. **delay** *duration*
14. **idle** *duration*
15. **maxconns** *maximum-number*
16. **sticky** *seconds* [**netmask** *netmask*] [**source** | **destination**]
17. **exit**
18. **protocol datagram**
19. **idle** *duration*
20. **maxconns** *maximum-number*
21. **sticky** *seconds* [**netmask** *netmask*] [**source** | **destination**]

22. **exit**
23. **inservice**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb firewallfarm** *firewall-farm* **Example:** `Router(config)# ip slb firewallfarm FIRE1`	Adds a firewall farm definition to the IOS Server Load Balancing (IOS SLB) configuration and enters firewall farm configuration mode.
Step 4	**real** *ip-address* **Example:** `Router(config-slb-fw)# real 10.1.1.1`	Identifies a firewall by IP address as a member of a firewall farm and enters real server configuration mode.
Step 5	**probe** *probe* **Example:** `Router(config-slb-fw-real)# probe FireProbe`	Associates a probe with the firewall.
Step 6	**weight** *setting* **Example:** `Router(config-slb-fw-real)# weight 16`	(Optional) Specifies the firewall's workload capacity relative to other firewalls in the firewall farm.
Step 7	**inservice** **Example:** `Router(config-slb-fw-real)# inservice`	Enables the firewall for use by the firewall farm and by IOS Server Load Balancing (IOS SLB).
Step 8	**exit** **Example:** `Router(config-slb-fw-real)# exit`	Exits real server configuration mode.
Step 9	**access** [**source** *source-ip netmask*] [**destination** *destination-ip netmask*]| **inbound** *inbound-interface* | **outbound** *outbound-interface*] **Example:** `Router(config-slb-fw)# access destination 10.1.6.0 255.255.255.0`	(Optional) Routes specific flows to a firewall farm.

	Command	Purpose
Step 10	**predictor hash address** [**port**] **Example:** `Router(config-slb-fw)# predictor hash address`	(Optional) Specifies whether the source and destination TCP or User Datagram Protocol (UDP) port numbers, in addition to the source and destination IP addresses, are to be used when selecting a firewall.
Step 11	**replicate casa** *listen-ip remote-ip port* [*interval*] [**password** [[*encrypt*] *secret-string* [*timeout*]] **Example:** `Router(config-slb-fw)# replicate casa 10.10.10.11 10.10.11.12 4231`	(Optional) Configures a stateful backup of IOS Server Load Balancing (IOS SLB) firewall load balancing decision tables to a backup switch.
Step 12	**protocol tcp** **Example:** `Router(config-slb-fw)# protocol tcp`	(Optional) Enters firewall farm TCP protocol configuration mode.
Step 13	**delay** *duration* **Example:** `Router(config-slb-fw-tcp)# delay 30`	(Optional) For firewall farm TCP protocol configuration mode, specifies the time IOS Server Load Balancing (IOS SLB) firewall load balancing maintains TCP connection context after a connection has terminated.
Step 14	**idle** *duration* **Example:** `Router(config-slb-fw-tcp)# idle 120`	(Optional) For firewall farm TCP protocol configuration mode, specifies the minimum time IOS Server Load Balancing (IOS SLB) firewall load balancing maintains connection context in the absence of packet activity.
Step 15	**maxconns** *maximum-number* **Example:** `Router(config-slb-fw-tcp)# maxconns 1000`	(Optional) For firewall farm TCP protocol configuration mode, specifies the maximum number of active TCP connections allowed on the firewall farm at one time.
Step 16	**sticky** *seconds* [**netmask** *netmask*] [**source** | **destination**] **Example:** `Router(config-slb-fw-tcp)# sticky 60`	(Optional) For firewall farm TCP protocol configuration mode, specifies that connections from the same IP address use the same firewall if either of the following conditions is met: • As long as any connection between the same pair of IP addresses exists (source/destination sticky). • For a period, defined by *duration*, after the last connection is destroyed.
Step 17	**exit** **Example:** `Router(config-slb-fw-tcp)# exit`	Exits firewall farm TCP protocol configuration mode.
Step 18	**protocol datagram** **Example:** `Router(config-slb-fw)# protocol datagram`	(Optional) Enters firewall farm datagram protocol configuration mode.

	Command	Purpose

Step 19	**`idle`** *`duration`*
	Example:
	`Router(config-slb-fw-udp)# idle 120`
		(Optional) For firewall farm datagram protocol configuration mode, specifies the minimum time IOS Server Load Balancing (IOS SLB) firewall load balancing maintains connection context in the absence of packet activity.

Step 20	**`maxconns`** *`maximum-number`*
	Example:
	`Router(config-slb-fw-udp)# maxconns 1000`
		(Optional) For firewall farm datagram protocol configuration mode, specifies the maximum number of active datagram connections allowed on the firewall farm at one time.

Step 21	**`sticky`** *`seconds`* `[`**`netmask`** *`netmask`*`] [`**`source`** `|` **`destination`**`]`
	Example:
	`Router(config-slb-fw-udp)# sticky 60`
		(Optional) For firewall farm datagram protocol configuration mode, specifies that connections from the same IP address use the same firewall if either of the following conditions is met:
		• As long as any connection between the same pair of IP addresses exists (source/destination sticky).
		• For a period, defined by *duration*, after the last connection is destroyed.

Step 22	**`exit`**
	Example:
	`Router(config-slb-fw-udp)# exit`
		Exits firewall farm datagram protocol configuration mode.

Step 23	**`inservice`**
	Example:
	`Router(config-slb-fw)# inservice`
		Enables the firewall farm for use by IOS Server Load Balancing (IOS SLB).

Verifying a Firewall Farm

Perform the following task to verify a firewall farm.

SUMMARY STEPS

1. **show ip slb real**
2. **show ip slb firewallfarm**

DETAILED STEPS

The following **show ip slb reals** command displays the status of firewall farm FIRE1, the associated real servers, and their status:

```
Router# show ip slb real

real                  farm name        weight   state          conns
---------------------------------------------------------------------
10.1.1.2              FIRE1            8        OPERATIONAL    0
10.1.2.2              FIRE1            8        OPERATIONAL    0
```

The following **show ip slb firewallfarm** command displays the configuration and status of firewall farm FIRE1:

```
Router# show ip slb firewallfarm

firewall farm    hash        state          reals
------------------------------------------------
FIRE1            IPADDR      INSERVICE      2
```

Verifying Firewall Connectivity

Perform the following task to verify firewall connectivity.

SUMMARY STEPS

1. Ping the external real servers.
2. Ping the internal real servers.
3. **show ip slb stats**
4. **show ip slb real detail**
5. **show ip slb conns**

DETAILED STEPS

To verify that IOS SLB firewall load balancing has been configured and is operating correctly, perform the following steps:

Step 1 Ping the external real servers (the ones outside the firewall) from the IOS SLB firewall load-balancing device.

Step 2 Ping the internal real servers (the ones inside the firewall) from the clients.

Step 3 Use the **show ip slb stats** command to display detailed information about the IOS SLB firewall load-balancing network status:

```
Router# show ip slb stats

 Pkts via normal switching:  0
 Pkts via special switching: 0
 Pkts dropped:               0
 Connections Created:        1911871
 Connections Established:    1967754
 Connections Destroyed:      1313251
 Connections Reassigned:     0
 Zombie Count:               0
 Connections Reused:         59752
 Connection Flowcache Purges:1776582
 Failed Connection Allocs:   17945
 Failed Real Assignments:    0
```

Normal switching is when IOS SLB packets are handled on normal IOS switching paths (CEF, fast switching, and process level switching). Special switching is when IOS SLB packets are handled on hardware-assisted switching paths.

Step 4 Use the **show ip slb real detail** command to display detailed information about the IOS SLB firewall load-balancing real server status:

```
Router# show ip slb real detail

10.1.1.3, FIRE1, state = OPERATIONAL, type = firewall
  conns = 299310, dummy_conns = 0, maxconns = 4294967295
  weight = 10, weight(admin) = 10, metric = 104, remainder = 2
  total conns established = 1074779, hash count = 4646
  server failures = 0
  interface FastEthernet1/0, MAC 0010.f68f.7020
```

Step 5 Use the **show ip slb conns** command to display detailed information about the active IOS SLB firewall load-balancing connections:

```
Router# show ip slb conns

vserver         prot client                real                  state     nat
-------------------------------------------------------------------------------
FirewallTCP     TCP  80.80.50.187:40000    10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000    10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000    10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000    10.1.1.4              ESTAB     none
FirewallTCP     TCP  80.80.50.187:40000    10.1.1.4              ESTAB     none
```

See the "Monitoring and Maintaining IOS SLB" section on page 100 for additional commands used to verify IOS SLB networks and connections.

Configuring a Probe

Perform the following task to configure a probe.

SUMMARY STEPS

1. Configure a probe on each real server in the firewall farm.

DETAILED STEPS

The Exchange Director uses probes to detect and recover from failures. You must configure a probe on each real server in the firewall farm.

- Ping probes are recommended; see the "Configuring a Ping Probe" section on page 58 for more details.
- If a firewall does not allow ping probes to be forwarded, use HTTP probes instead. See the "Configuring an HTTP Probe" section on page 56 for more details.
- You can configure more than one probe, in any combination of supported types (DNS, HTTP, TCP, or ping), for each firewall in a firewall farm.

Configuring a Wildcard Search

Perform the following task to configure a wildcard search.

SUMMARY STEPS

1. **mls ip slb wildcard search rp**

DETAILED STEPS

Use the **mls ip slb wildcard search rp** command to reduce the probability of exceeding the capacity of the TCAM on the PFC.

VPN Server Load Balancing Configuration Task List

Perform the following task to configure VPN server load balancing.

SUMMARY STEPS

1. Configure a server farm and a real server.
2. Configure a virtual server.
3. Configure a probe.

DETAILED STEPS

	Task	Description
Step 1	Configure a server farm and a real server.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for VPN server load balancing, specify the IP addresses of the real servers acting as VPN terminators, using the **real** command.
Step 2	Configure a virtual server.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for VPN server load balancing of IPSec flows, keep the following considerations in mind: • Configure a UDP virtual server, using the **virtual** command with the protocol set to **udp** and the port set to **isakmp**. The **isakmp** keyword enables the cryptographic key exchange to occur via IKE (port 500). • Configure an ESP virtual server, using the **virtual** command with the protocol set to **esp**. • Specify a sticky connection from the UDP virtual server to the ESP virtual server, and vice versa, using the **sticky** command with a *duration* of at least 15 seconds. When you configure the virtual server for VPN server load balancing of PPTP flows, keep the following considerations in mind: • Configure a TCP virtual server, using the **virtual** command with the **tcp** keyword and port number **1723** specified. • Configure a GRE virtual server, using the **virtual** command with the **gre** keyword specified. • Specify a sticky connection from the TCP virtual server to the GRE virtual server, and vice versa, using the **sticky** command with a *duration* of at least 15 seconds.
Step 3	Configure a probe.	See the "Configuring a Probe" section on page 53. To verify the health of the server, configure a ping probe.

ASN R6 Load Balancing Configuration Task List

Perform the following task to configure load balancing across a set of Access Service Network (ASN) gateways.

SUMMARY STEPS

1. Configure the base station.
2. Configure a server farm and a real server.
3. Configure a virtual server.
4. Configure a probe.

DETAILED STEPS

	Task	Description
Step 1	Configure the base station.	To enable IOS SLB to handle requests from the Mobile Subscriber Station (MSS), configure the base station with the virtual IP address of the IOS SLB device.
Step 2	Configure a probe.	See the "Configuring a Probe" section on page 53. To verify the health of the server, configure a ping probe.
Step 3	Associate a server farm and a real server with the probe.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for ASN R6 load balancing, specify the IP addresses of the ASN gateways, using the **real** command.
Step 4	Associate a virtual server with the server farm.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for ASN R6 load balancing, keep the following considerations in mind: • Configure a virtual server, using the **virtual** command with the service set to **asn r6**. • Configure an idle connection timer for ASN R6 load balancing, using the **idle** command with the service set to **asn r6 request** keywords specified.

Home Agent Director Configuration Task List

Perform the following task to configure the Home Agent Director.

SUMMARY STEPS

1. Configure a server farm and a real server.
2. Configure a virtual server.
3. Configure the virtual IP address as a loopback on each of the home agents in the servers.
4. Configure DFP.

DETAILED STEPS

	Task	Description
Step 1	Configure a server farm and a real server.	See the "Configuring a Server Farm and a Real Server" procedure on page -36. When you configure the server farm and real server for the Home Agent Director, keep the following considerations in mind: • Accept the default setting (the weighted round robin algorithm) for the **predictor** command. • Specify the IP addresses of the real servers acting as home agents, using the **real** command.
Step 2	Configure a virtual server.	See the "Configuring a Virtual Server" procedure on page -40. When you configure the virtual server for the Home Agent Director using the **virtual** command, keep the following considerations in mind: • Specify the Home Agent Director's IP address as the virtual server. • Specify the **udp** keyword option. • Specify port number 434 if the home agents are in compliance with the IP Mobility Support, RFC 2002, or specify port number 0 or **any** to configure an all-port virtual server (that is, a virtual server that accepts flows destined for all ports). • Specify the **service ipmobile** keyword option.
Step 3	Configure the virtual IP address as a loopback on each of the home agents in the servers.	(Required for dispatched mode) This step is required only if you are using dispatched mode. Refer to the "Configuring a Loopback Interface" section in the *Cisco IOS Interface Configuration Guide*, Release 12.2 for more information.
Step 4	Configure DFP.	(Optional) See the "Configuring DFP" section on page 63. When you configure DFP for the Home Agent Director, keep the following considerations in mind: • To control the maximum DFP weight sent by the home agent to IOS SLB, use the **ip mobile home-agent dfp-max-weight** command. • To set the source address and home agent address field in the Registration Reply (RRP) as the real home agent's address, use the **ip mobile home-agent dynamic-address** command. • To set the maximum number of bindings, use the **ip mobile home-agent max-binding** command. For detailed information about these Mobile IP commands, refer to the *Cisco Mobile Wireless Home Agent Release 2.0* feature module.

Configuring NAT

Perform the following task to configure the IOS SLB NAT client address pool for client NAT.

SUMMARY STEPS

1. **ip slb natpool**
2. **nat**

DETAILED STEPS

Command	Purpose
ip slb natpool pool *start-ip end-ip* [**netmask** *netmask* | **prefix-length** *leading-1-bits*] [**entries** *init-address* [*max-address*]] **Example:** `Router(config)# ip slb natpool web-clients 10.1.10.1 10.1.10.5 netmask 255.255.0.0`	Configures the client address pool. **Note** GPRS load balancing does not support this command. You do not need to configure the client address pool for server NAT.

You must also specify either NAT client translation mode or NAT server address translation mode on the server farm, using the **nat** command. See the "Configuring a Server Farm and a Real Server" section on page 36 for more details. When you configure the virtual server for NAT, remember that you cannot configure client NAT for an ESP or GRE virtual server.

Configuring Static NAT

Perform the following task to configure static NAT.

Static NAT enables you to allow some users to utilize NAT and allow other users on the same Ethernet interface to continue with their own IP addresses. This option enables you to provide a default NAT behavior for real servers, differentiating between responses from a real server, and connection requests initiated by the real server.

Note To avoid unexpected results, make sure your static NAT configuration mirrors your virtual server configuration.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb static** {**drop** | **nat** {**virtual** | *virtual-ip* [**per-packet** | **sticky**]}}
4. **real** *ip-address* [*port*]

DETAILED STEPS

	Command	Description

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**ip slb static** {**drop** | **nat** {**virtual** | *virtual-ip* [**per-packet** | **sticky**]}}
	Example:
	`Router(config)# ip slb static nat 10.1.10.1 per-packet`
		Configures the real server's Network Address Translation (NAT) behavior and enters static NAT configuration mode.
		Note If you specify the *virtual-ip* argument and you do not specify the **per-packet** option, IOS Server Load Balancing (IOS SLB) uses server port translation to distinguish between connection requests initiated by different real servers.

Step 4	**real** *ip-address* [*port*] **Example:** `Router(config-slb-static)# real 10.1.1.3`	Configures one or more real servers to use static Network Address Translation (NAT).

Stateless Backup Configuration Task List

Perform the following task to configure stateless backup over VLANs between IOS SLB devices.

Note For active standby, in which multiple IOS SLB devices share a virtual IP address, you must use exclusive client ranges and you must use policy routing to forward flows to the correct IOS SLB device.

SUMMARY STEPS

1. Configure required and optional IOS SLB functions.
2. Configure firewall load balancing.
3. Configure the IP routing protocol.
1. Configure the VLAN between the IOS SLB devices.
2. Verify the stateless backup configuration.

DETAILED STEPS

	Task	Description
Step 1	Configure required and optional IOS SLB functions.	(Required for server load balancing) See the "Configuring Required and Optional IOS SLB Functions" section on page 36.
Step 2	Configure firewall load balancing.	(Required for firewall load balancing) See the "Configuring Firewall Load Balancing" section on page 47.
Step 3	Configure the IP routing protocol.	Refer to the "IP Routing Protocols" chapter of the *Cisco IOS IP Configuration Guide*, Release 12.2 for more details.
Step 4	Configure the VLAN between the IOS SLB devices.	Refer to the "Virtual LANs" chapter of the *Cisco IOS Switching Services Configuration Guide*, Release 12.2 for more details.
Step 5	Verify the stateless backup configuration.	(Optional) See the "Verifying the Stateless Backup Configuration" section on page 93.

Verifying the Stateless Backup Configuration

Perform the following task to verify the stateless backup configuration.

SUMMARY STEPS

1. **show ip slb vservers**
2. **show ip slb vservers detail**
3. **show ip slb firewallfarm**
4. **show ip slb firewallfarm details**

DETAILED STEPS

For server load balancing, to verify that stateless backup has been configured and is operating correctly, use the following **show ip slb vservers** commands to display information about the IOS SLB virtual server status:

```
Router# show ip slb vservers

slb vserver      prot  virtual              state         conns
-------------------------------------------------------------------
VS1              TCP   10.10.10.12:23       OPERATIONAL     2
VS2              TCP   10.10.10.18:23       OPERATIONAL     2

Router# show ip slb vservers detail

VS1, state = OPERATIONAL, v_index = 10
  virtual = 10.10.10.12:23, TCP, service = NONE, advertise = TRUE
  server farm = SERVERGROUP1, delay = 10, idle = 3600
  sticky timer = 0, sticky subnet = 255.255.255.255
  sticky group id = 0
  synguard counter = 0, synguard period = 0
  conns = 0, total conns = 0, syns = 0, syn drops = 0
  standby group = None
VS2, state = INSERVICE, v_index = 11
  virtual = 10.10.10.18:23, TCP, service = NONE, advertise = TRUE
  server farm = SERVERGROUP2, delay = 10, idle = 3600
  sticky timer = 0, sticky subnet = 255.255.255.255
  sticky group id = 0
  synguard counter = 0, synguard period = 0
  conns = 0, total conns = 0, syns = 0, syn drops = 0
  standby group = None
```

For firewall load balancing, to verify that stateless backup has been configured and is operating correctly, use the following **show ip slb firewallfarm** commands to display information about the IOS SLB firewall farm status:

```
Router# show ip slb firewallfarm

firewall farm    hash        state         reals
------------------------------------------------
FIRE1            IPADDR      INSERVICE     2

Router# show ip slb firewallfarm details

FIRE1, hash = IPADDRPORT, state = INSERVICE, reals = 2
  FirewallTCP:
   sticky timer = 0, sticky subnet = 255.255.255.255
   idle = 3600, delay = 10, syns = 1965732, syn drop = 0
   maxconns = 4294967295, conns = 597445, total conns = 1909512
  FirewallUDP:
   sticky timer = 0, sticky subnet = 255.255.255.255
   idle = 3600
   maxconns = 1, conns = 0, total conns = 1
  Real firewalls:
    10.1.1.3, weight = 10, OPERATIONAL, conns = 298823
    10.1.1.4, weight = 10, OPERATIONAL, conns = 298622
  Total connections = 597445
```

Stateful Backup of Redundant Route Processors Configuration Task List

Perform the following task to configure stateful backup of redundant route processors.

SUMMARY STEPS

1. Configure the replication message rate for slave replication.
2. Configure required and optional IOS SLB functions.
3. Configure firewall load balancing.

DETAILED STEPS

	Task	Description
Step 1	Configure the replication message rate for slave replication.	Specify the **ip slb replicate slave rate** command in global configuration mode.

	Task	Description
Step 2	Configure required and optional IOS SLB functions.	(Required for server load balancing) See the "Configuring Required and Optional IOS SLB Functions" section on page 36. When you configure the virtual server for stateful backup of redundant route processors, keep the following considerations in mind: • Specify the **replicate slave** command. • (Optional) To set the replication delivery interval for the virtual server, configure a **replicate interval** command.
Step 3	Configure firewall load balancing.	(Required for firewall load balancing) See the "Configuring Firewall Load Balancing" section on page 47. When you configure the firewall farm for stateful backup of redundant route processors, keep the following considerations in mind: • Specify the **replicate slave** command. • (Optional) To set the replication delivery interval for the firewall farm, configure a **replicate interval** command.

Configuring Database Entries

Perform the following task to configure database entries.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb entries** [**conn** [*init-conn* [*max-conn*]] | **frag** [*init-frag* [*max-frag*] | **lifetime** *timeout*] | **gtp** {**gsn** [*init-gsn* [*max-gsn*] | **nsapi** [*init-nsapi* [*max-nsapi*]} | **sticky** [*init-sticky* [*max-sticky*]]]

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ip slb entries** [**conn** [*init-conn* [*max-conn*]] | **frag** [*init-frag* [*max-frag*] | **lifetime** *timeout*] | **gtp** {**gsn** [*init-gsn* [*max-gsn*] | **nsapi** [*init-nsapi* [*max-nsapi*]} | **sticky** [*init-sticky* [*max-sticky*]]] **Example:** `Router(config)# ip slb entries conn 128000 512000`	Specifies an initial allocation and a maximum value for IOS Server Load Balancing (IOS SLB) database entries. **Note** Enter this command *before* entering the rest of your IOS SLB configuration. If your IOS SLB configuration already exists, you must reload ISO SLB after entering this command.

Configuring Buffers for the Fragment Database

Perform the following task to configure buffers for the fragment database.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb maxbuffers frag** *buffers*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb maxbuffers frag** *buffers* **Example:** `Router(config)# ip slb maxbuffers frag 300`	Configures the maximum number of buffers for the IOS Server Load Balancing (IOS SLB) fragment database.

Clearing Databases and Counters

Perform the following task to clear databases and counters.

SUMMARY STEPS

1. **clear ip slb connections** [**firewallfarm** *firewall-farm* | **serverfarm** *server-farm* | **vserver** *virtual-server*]
2. **clear ip slb counters** [**kal-ap**]

3. **clear ip slb sessions** [**firewallfarm** *firewall-farm* | **serverfarm** *server-farm* | **vserver** *virtual-server*]
4. **clear ip slb sticky gtp imsi** [**id** *imsi*]
5. **clear ip slb sticky radius** {**calling-station-id** [**id** *string*] | framed-ip [*framed-ip* [*netmask*]]}

DETAILED STEPS

	Command or Action	Purpose
Step 1	`clear ip slb connections [firewallfarm firewall-farm	serverfarm server-farm	vserver virtual-server]` **Example:** `Router# clear ip slb connections vserver VSERVER1`	Clears the IOS Server Load Balancing (IOS SLB) connection database for one or more firewall farms, server farms, or virtual servers.

Step 2	`clear ip slb counters [kal-ap]`
	Example:
	`Router# clear ip slb counters`
		Clears the IOS Server Load Balancing (IOS SLB) counters.
		Use the **kal-ap** keyword to clear only IP IOS SLB KeepAlive Application Protocol (KAL-AP) counters.

Step 3	`clear ip slb sessions [firewallfarm firewall-farm	serverfarm server-farm	vserver virtual-server]` **Example:** `Router# clear ip slb sessions serverfarm FARM1`	Clears the IOS Server Load Balancing (IOS SLB) RADIUS session database for one or more firewall farms, server farms, or virtual servers.

Step 4	`clear ip slb sticky gtp imsi [id imsi]`
	Example:
	`Router# clear ip slb sticky gtp imsi`
		Clears entries from an IOS Server Load Balancing (IOS SLB) general packet radio service (GPRS) Tunneling Protocol (GTP) International Mobile Subscriber ID (IMSI) sticky database.

Step 5	`clear ip slb sticky radius {calling-station-id [id string]	framed-ip [framed-ip [netmask]]}` **Example:** `Router# clear ip slb sticky radius framed-ip`	Clears entries from an IOS Server Load Balancing (IOS SLB) RADIUS sticky database.

Configuring a Wildcard Search

Perform the following task to configure a wildcard search.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mls ip slb search**

DETAILED STEPS

Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	Router(config)# **mls ip slb search** {**wildcard** [**pfc** | **rp**] | **icmp**} **Example:** `Router(config)# mls ip slb search wildcard rp`	Specifies the behavior of IOS Server Load Balancing (IOS SLB) wildcard searches. This command is supported for Catalyst 6500 family switches only.

Purging and Reassigning Connections

Perform the following task to purge an reassign connections.

You can enable IOS SLB to automatically remove connections to failed real servers and firewalls from the connection database even if the idle timers have not expired. This function is useful for applications that do not rotate the source port (such as IKE), and for protocols that do not have ports to differentiate flows (such as ESP).

You can also enable IOS SLB to automatically reassign to a new real server or firewall RADIUS sticky objects that are destined for a failed real server or firewall.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb serverfarm** *server-farm*
4. **failaction** [**purge** | **gtp purge** | **radius reassign**]
5. **exit**
6. **ip slb firewallfarm** *firewall-farm*
7. **failaction purge**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb serverfarm** *server-farm* **Example:** `Router(config)# ip slb serverfarm PUBLIC`	Enters server farm configuration mode.
Step 4	**failaction** [**purge** | **gtp purge** | **radius reassign**] **Example:** `Router(config-slb-sfarm)# failaction purge`	Configures IOS Server Load Balancing (IOS SLB)'s behavior when a real server fails.
Step 5	**exit** **Example:** `Router(config-slb-sfarm)# exit`	Exits server farm configuration mode.
Step 6	**ip slb firewallfarm** *firewall-farm* **Example:** `Router(config)# ip slb firewallfarm fire1`	Enters firewall farm configuration mode.
Step 7	**failaction purge** **Example:** `Router(config-slb-fw)# failaction purge`	Configures IOS Server Load Balancing (IOS SLB)'s behavior when a firewall fails.

Disabling Automatic Server Failure Detection

Perform the following task to disable automatic server failure detection.

If you have configured all-port virtual servers (that is, virtual servers that accept flows destined for all ports except GTP ports), flows can be passed to servers for which no application port exists. When the servers reject these flows, IOS SLB might fail the servers and remove them from load balancing. This situation can also occur in slow-to-respond AAA servers in RADIUS load-balancing environments. To prevent this situation, you can disable automatic server failure detection.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip slb serverfarm** *server-farm*
4. **real ip-address** [**port**]
5. **no faildetect inband**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip slb serverfarm** *server-farm* **Example:** `Router(config)# ip slb serverfarm PUBLIC`	Enters server farm configuration mode.
Step 4	**real ip-address** [**port**] **Example:** `Router(config-slb-sfarm)# real 10.1.1.1`	Identifies a real server as a member of a server farm and enters real server configuration mode.
Step 5	**no faildetect inband** **Example:** `Router(config-slb-real)# no faildetect inband`	Disables automatic server failure detection. **Note** If you disable automatic server failure detection using the **no faildetect inband** command, Cisco strongly recommends that you configure one or more probes. If you specify the **no faildetect inband** command, the **faildetect numconns** command is ignored, if specified.

Monitoring and Maintaining IOS SLB

Perform the following task to obtain and display runtime information about IOS SLB.

SUMMARY STEPS

1. **show ip slb conns**
2. **show ip slb dfp**
3. **show ip slb firewallfarm**
4. **show ip slb fragments**
5. **show ip slb gtp**
6. **show ip slb map**
7. **show ip slb natpool**
8. **show ip slb probe**
9. **show ip slb reals**

10. **show ip slb replicate**
11. **show ip slb serverfarms**
12. **show ip slb sessions**
13. **show ip slb static**
14. **show ip slb stats**
15. **show ip slb sticky**
16. **show ip slb vservers**

DETAILED STEPS

Step 1 **show ip slb conns** [**vserver** *virtual-server* | **client** *ip-address* | **firewall** *firewall-farm*] [**detail**]

Displays all connections handled by IOS SLB, or, optionally, only those connections associated with a particular virtual server or client. The following is sample output from this command:

```
Router# show ip slb conns

vserver          prot   client                real                  state
----------------------------------------------------------------------------
TEST             TCP    10.150.72.183:328     10.80.90.25:80        INIT
TEST             TCP    10.250.167.226:423    10.80.90.26:80        INIT
TEST             TCP    10.234.60.239:317     10.80.90.26:80        ESTAB
TEST             TCP    10.110.233.96:747      10.80.90.26:80        ESTAB
TEST             TCP    10.162.0.201:770       10.80.90.30:80        CLOSING
TEST             TCP    10.22.225.219:995      10.80.90.26:80        CLOSING
TEST             TCP    10.2.170.148:169       10.80.90.30:80
```

Step 2 **show ip slb dfp** [**agent** *agent-ip port* | **manager** *manager-ip* | **detail** | **weights**]

Displays information about Dynamic Feedback Protocol (DFP) and DFP agents, and about the weights assigned to real servers. The following is sample output from this command:

```
Router# show ip slb dfp

DFP Manager:
Current passwd:NONE Pending passwd:NONE
Passwd timeout:0 sec
Agent IP          Port    Timeout   Retry Count   Interval
--------------------------------------------------------------
172.16.2.34       61936   0         0             180 (Default)
```

Step 3 **show ip slb firewallfarm** [**detail**]

Displays information about firewall farms. The following is sample output from this command:

```
Router# show ip slb firewallfarm

firewall farm    hash        state         reals
------------------------------------------------
FIRE1            IPADDR      OPERATIONAL     2
```

Step 4 **show ip slb fragments**

Displays information from the IOS SLB fragment database. The following is sample output from this command:

```
Router# show ip slb fragments

ip src          id    forward         src nat         dst nat
---------------------------------------------------------------------
10.11.2.128     12    10.11.2.128     10.11.11.11     10.11.2.128
10.11.2.128     13    10.11.2.128     10.11.11.11     10.11.2.128
10.11.2.128     14    10.11.2.128     10.11.11.11     10.11.2.128
10.11.2.128     15    10.11.2.128     10.11.11.11     10.11.2.128
10.11.2.128     16    10.11.2.128     10.11.11.11     10.11.2.128
```

Step 5 **show ip slb gtp** {**gsn** [*gsn-ip-address*] | **nsapi** [*nsapi-key*] [**detail**]

Displays IOS Server Load Balancing (IOS SLB) GTP information. The following is sample output from this command:

```
Router# show ip slb gtp gsn 10.0.0.0
type ip              recovery-ie  purging
------------------------------------------
SGSN 10.0.0.0 UNKNOWN      N
```

Step 6 **show ip slb map** [*map-id*]

Displays information about IOS SLB protocol maps. The following is sample output from this command:

```
Router# show ip slb map

ID: 1, Service: GTP
 APN: Cisco.com, yahoo.com
 PLMN ID(s): 11122, 444353
 SGSN access list: 100
ID: 2, Service: GTP
 PLMN ID(s): 67523, 345222
 PDP Type: IPv4, PPP
ID: 3, Service: GTP
 PDP Type: IPv6
ID: 4, Service: RADIUS
 Calling-station-id: “?919*”
ID: 5, Service: RADIUS
 Username: “..778cisco.*”
```

Step 7 **show ip slb natpool** [**name** *pool*] [**detail**]

Displays information about the IOS Server Load Balancing (IOS SLB) Network Address Translation (NAT) configuration. The following is sample output from this command:

```
Router# show ip slb natpool

nat client B 209.165.200.225 1.1.1.6 1.1.1.8 Netmask 255.255.255.0
nat client A 10.1.1.1 1.1.1.5 Netmask 255.255.255.0
```

Step 8 **show ip slb probe** [**name** *probe*] [**detail**]

Displays information about probes defined to IOS Server Load Balancing (IOS SLB). The following is sample output from this command:

```
Router# show ip slb probe

Server:Port            State        Outages  Current  Cumulative
----------------------------------------------------------------
10.10.4.1:0            OPERATIONAL        0  never    00:00:00
10.10.5.1:0            FAILED             1  00:00:06 00:00:06
```

Step 9 **show ip slb reals** [**sfarm** *server-farm*] [**detail**]

Displays information about the real servers defined to IOS Server Load Balancing (IOS SLB). The following is sample output from this command:

```
Router# show ip slb reals

real             farm name        weight   state           conns
--------------------------------------------------------------------
10.80.2.112      FRAG             8        OUTOFSERVICE    0
10.80.5.232      FRAG             8        OPERATIONAL     0
10.80.15.124     FRAG             8        OUTOFSERVICE    0
10.254.2.2       FRAG             8        OUTOFSERVICE    0
10.80.15.124     LINUX            8        OPERATIONAL     0
10.80.15.125     LINUX            8        OPERATIONAL     0
10.80.15.126     LINUX            8        OPERATIONAL     0
10.80.90.25      SRE              8        OPERATIONAL     220
10.80.90.26      SRE              8        OPERATIONAL     216
10.80.90.27      SRE              8        OPERATIONAL     216
10.80.90.28      SRE              8        TESTING         1
10.80.90.29      SRE              8        OPERATIONAL     221
10.80.90.30      SRE              8        OPERATIONAL     224
10.80.30.3       TEST             100      READY_TO_TEST   0
10.80.30.4       TEST             100      READY_TO_TEST   0
10.80.30.5       TEST             100      READY_TO_TEST   0
10.80.30.6       TEST             100      READY_TO_TEST   0
```

Step 10 **show ip slb replicate**

Displays information about the IOS Server Load Balancing (IOS SLB) replication configuration. The following is sample output from this command:

```
Router# show ip slb replicate

VS1, state = NORMAL, interval = 10
 Slave Replication: Enabled
 Slave Replication statistics:
  unsent conn updates:         0
  conn updates received:       0
  conn updates transmitted:    0
  update messages received:    0
  update messages transmitted: 0
 Casa Replication:
  local = 10.1.1.1 remote = 10.2.2.2 port = 1024
  current password = <none> pending password = <none>
  password timeout = 180 sec (Default)
 Casa Replication statistics:
  unsent conn updates:        0
  conn updates received:      0
  conn updates transmitted:   0
  update packets received:    0
  update packets transmitted: 0
  failovers:                  0
```

Step 11 **show ip slb serverfarms** [**name** *server-farm*] [**detail**]

Displays information about the server farms defined to IOS Server Load Balancing (IOS SLB). The following is sample output from this command:

```
Router# show ip slb serverfarms

server farm      predictor     reals   bind id
-------------------------------------------------
FRAG             ROUNDROBIN    4       0
LINUX            ROUNDROBIN    3       0
SRE              ROUNDROBIN    6       0
TEST             ROUNDROBIN    4       0
```

Step 12 **show ip slb sessions** [**asn r6** | **gtp** | **gtp-inspect** | **ipmobile** | **radius**] [**vserver** *virtual-server*] [**client** *ip-address netmask*] [**detail**]

Displays information about sessions handled by IOS Server Load Balancing (IOS SLB). The following is sample output from this command:

```
Router# show ip slb sessions radius

Source               Dest                   Retry
Addr/Port            Addr/Port           Id Count  Real            Vserver
------------------------------------------------------------------------------
10.10.11.1/1645      10.10.11.2/1812     15     1  10.10.10.1  RADIUS_ACCT
```

Step 13 **show ip slb static**

Displays information about the IOS Server Load Balancing (IOS SLB) server Network Address Translation (NAT) configuration. The following is sample output from this command:

```
Router# show ip slb static

real                     action         address         counter
---------------------------------------------------------------
10.11.3.4                drop           0.0.0.0         0
10.11.3.1                NAT            10.11.11.11     3
10.11.3.2                NAT sticky     10.11.11.12     0
10.11.3.3                NAT per-packet 10.11.11.13     0
```

Step 14 **show ip slb stats**

Displays IOS Server Load Balancing (IOS SLB) statistics. The following is sample output from this command:

```
Router# show ip slb stats

Pkts via normal switching:     779
Pkts via special switching:    0
Pkts via slb routing:          0
Pkts Dropped:                  4
Connections Created:           4
Connections Established:       4
Connections Destroyed:         4
Connections Reassigned:        5
Zombie Count:                  0
Connections Reused:            0
Connection Flowcache Purges:   0
Failed Connection Allocs:      0
Failed Real Assignments:       0
RADIUS framed-ip Sticky Count: 0
RADIUS username Sticky Count:            0
RADIUS calling-station-id Sticky Count:  0
```

```
GTP IMSI Sticky Count:         0
Failed Correlation Injects:    0
Pkt fragments drops in ssv:    0
```

Step 15 **show ip slb sticky** [**client** *ip-address netmask* | **radius calling-station-id** [**id** *string*] | **radius framed-ip** [**client** *ip-address netmask*] | **radius username** [**name** *string*]]

Displays information about the sticky connections defined to IOS Server Load Balancing (IOS SLB). The following is sample output from this command:

```
Router# show ip slb sticky
client           netmask          group  real                 conns
-----------------------------------------------------------------------
10.10.2.12       255.255.0.0      4097   10.10.3.2            1
```

Step 16 **show ip slb vservers** [**name** *virtual-server*] [**redirect**] [**detail**]

Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB). The following is sample output from this command:

```
Router# show ip slb vservers

slb vserver      prot   virtual              state          conns
---------------------------------------------------------------------
TEST             TCP     10.80.254.3:80        OPERATIONAL    1013
TEST21           TCP     10.80.254.3:21        OUTOFSERVICE   0
TEST23           TCP     10.80.254.3:23        OUTOFSERVICE   0
```

Configuration Examples for IOS SLB

This section provides real-world examples of IOS SLB configurations. For a complete description of the IOS SLB commands in this section, refer to the *Cisco IOS IP Application Services Command Reference*. To locate documentation of other commands that appear in this section, search online using Cisco.com.

This section includes the following examples:

- Configuring IOS SLB with GTP IMSI Sticky Database: Example, page 169
- Configuring IOS SLB with Transparent Web Cache Load Balancing: Example, page 169
- Configuring IOS SLB with KAL-AP Agent: Example, page 170

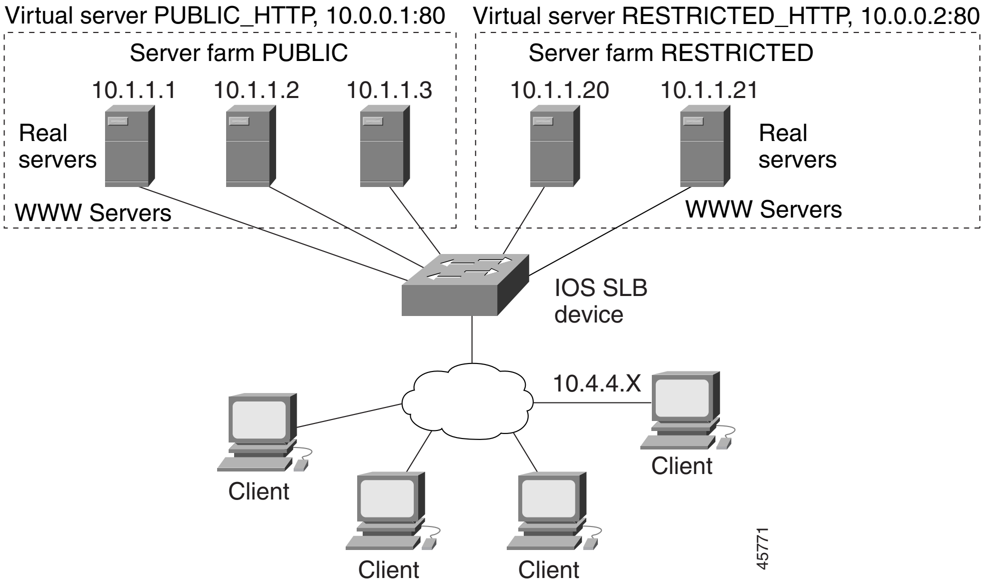
Note The IP and network addresses in these examples are generic; you must replace them with the actual addresses for your network.

Configuring a Basic IOS SLB Network: Example

Figure 2 shows a sample IOS SLB network with the following components:

- Two server farms—one configured to allow access by the public and named PUBLIC, one configured to allow limited access and named RESTRICTED.
- Five real servers configured as follows:
  - Three real servers in the PUBLIC server farm with IP addresses 10.1.1.1, 10.1.1.2, and 10.1.1.3
  - Two real servers in the restricted server farm with IP addresses 10.1.1.20 and 10.1.1.21
- Two virtual servers—one configured to allow access by the public and named PUBLIC_HTTP and one configured to allow limited access and named RESTRICTED_HTTP.
  - Virtual server PUBLIC_HTTP is configured with IP address 10.0.0.1 load balancing TCP connections on the WWW port (80).
  - Virtual server RESTRICTED_HTTP is configured with IP address 10.0.0.2 load balancing TCP connections on the WWW port (80) and allows access only from clients from network 10.4.4.0 255.255.255.0.

Figure 2 Example IOS SLB Network



The following sections include examples of the configuration commands used to configure and verify the IOS SLB network shown in Figure 2:

- Server Farm Configuration, page 107
- Virtual Server Configuration, page 107
- Restricted Client Configuration, page 108

Server Farm Configuration

The following example shows the configuration for the server farm PUBLIC, associated with three real servers:

```
ip slb serverfarm PUBLIC
  real 10.1.1.1
    reassign 2
    faildetect numconns 4 numclients 2
    retry 20
    inservice
    exit
  real 10.1.1.2
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
    exit
  real 10.1.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
    end
```

The following example shows the configuration for the server farm RESTRICTED, associated with two real servers:

```
ip slb serverfarm RESTRICTED
  real 10.1.1.20
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
    exit
  real 10.1.1.21
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
    end
```

Virtual Server Configuration

The following example shows the configuration for the virtual servers PUBLIC_HTTP and RESTRICTED_HTTP:

```
ip slb vserver PUBLIC_HTTP
  virtual 10.0.0.1 tcp www
  serverfarm PUBLIC
  idle 120
  delay 5
  inservice
  exit
ip slb vserver RESTRICTED_HTTP
  virtual 10.0.0.2 tcp www
  serverfarm RESTRICTED
  idle 120
  delay 5
  inservice
  end
```

Restricted Client Configuration

The following example shows the configuration for the virtual server RESTRICTED_HTTP:

```
ip slb vserver RESTRICTED_HTTP
  no inservice
  client 10.4.4.0 255.255.255.0
  inservice
  end
```

Configuring a Complete IOS SLB Network: Example

The following example provides a complete configuration using many of the commands described in this feature document:

```
ip slb probe PROBE2 http
 request method POST url /probe.cgi?all
 header HeaderName HeaderValue
!
ip slb serverfarm PUBLIC
 nat server
 real 10.1.1.1
  reassign 4
  faildetect numconns 16
  retry 120
  inservice
 real 10.1.1.2
  reassign 4
  faildetect numconns 16
  retry 120
  inservice
probe PROBE2
!
ip slb serverfarm RESTRICTED
 predictor leastconns
 bindid 309
 real 10.1.1.1
  weight 32
  maxconns 1000
  reassign 4
  faildetect numconns 16
  retry 120
  inservice
 real 10.1.1.20
  reassign 4
  faildetect numconns 16
  retry 120
  inservice
 real 10.1.1.21
  reassign 4
  faildetect numconns 16
  retry 120
  inservice
```

```
!
ip slb vserver PUBLIC_HTTP
 virtual 10.0.0.1 tcp www
 serverfarm PUBLIC
!
ip slb vserver RESTRICTED_HTTP
 virtual 10.0.0.2 tcp www
 serverfarm RESTRICTED
 no advertise
 sticky 60 group 1
 idle 120
 delay 5
 client 10.4.4.0 255.255.255.0
 synguard 3600000
 inservice
```

Configuring IOS SLB with Firewall Load Balancing: Examples

This section contains the following examples, illustrating several different IOS SLB firewall load-balancing configurations:

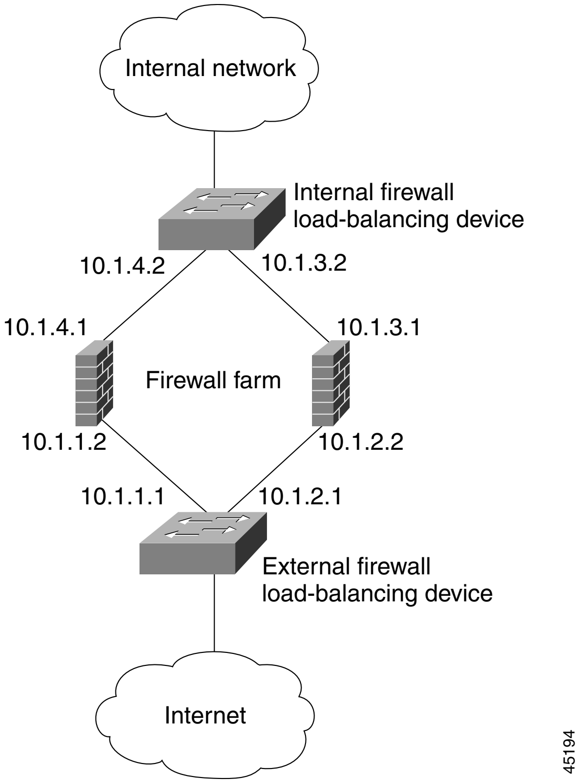
- Configuring IOS SLB with Basic Firewall Load Balancing: Example, page 109
- Configuring IOS SLB with Server Load Balancing and Firewall Load Balancing: Example, page 111
- Configuring IOS SLB with Multiple Firewall Farms: Example, page 113
- Configuring IOS SLB with Dual Firewall Load Balancing “Sandwich”: Example, page 115
- Configuring IOS SLB with RADIUS Load Balancing/Firewall Load Balancing “Sandwich”: Example, page 163

Configuring IOS SLB with Basic Firewall Load Balancing: Example

Figure 3 shows a sample IOS SLB firewall load-balancing network with the following components:

- Two firewalls with IP addresses as shown
- An internal firewall load-balancing device on the secure side of the firewalls
- An external firewall load-balancing device on the Internet side of the firewalls
- One firewall farm named FIRE1, containing both firewalls

Figure 3 IOS SLB with Layer 3 Firewalls in Different Subnets



When you configure IOS SLB firewall load balancing, the load-balancing devices use route lookup to recognize flows destined for the firewalls. To enable route lookup, you must configure each device with the IP address of each firewall that will route flows to that device.

In the following firewall farm configuration samples:

- The internal (secure side) firewall load-balancing device is configured with firewall IP addresses 10.1.3.1 and 10.1.4.1.
- The external (Internet side) firewall load-balancing device is configured with firewall IP addresses 10.1.1.2 and 10.1.2.2.

Internal Firewall Load-Balancing Device

The following example shows the configuration for ping probe PROBE1, HTTP probe PROBE2, and firewall farm FIRE1, associated with the two real servers for the load-balancing device on the internal (secure) side of the firewall:

```
!-----Ping probe
ip slb probe PROBE1 ping
!-----IP address of other load-balancing device
  address 10.1.1.1
  faildetect 4
!-----HTTP probe
  ip slb probe PROBE2 http
!-----IP address of other load-balancing device
  address 10.1.2.1
  expect status 401
!-----Firewall farm FIRE1
ip slb firewallfarm FIRE1
!-----First firewall
```

```
  real 10.1.4.1
    probe PROBE1
!-----Enable first firewall
    inservice
!-----Second firewall
    real 10.1.3.1
    probe PROBE2
!-----Enable second firewall
    inservice
```

External Firewall Load-Balancing Device

The following example shows the configuration for ping probe PROBE1, HTTP probe PROBE2, and firewall farm FIRE1, associated with the two real servers for the load-balancing device on the external (Internet) side of the firewall:

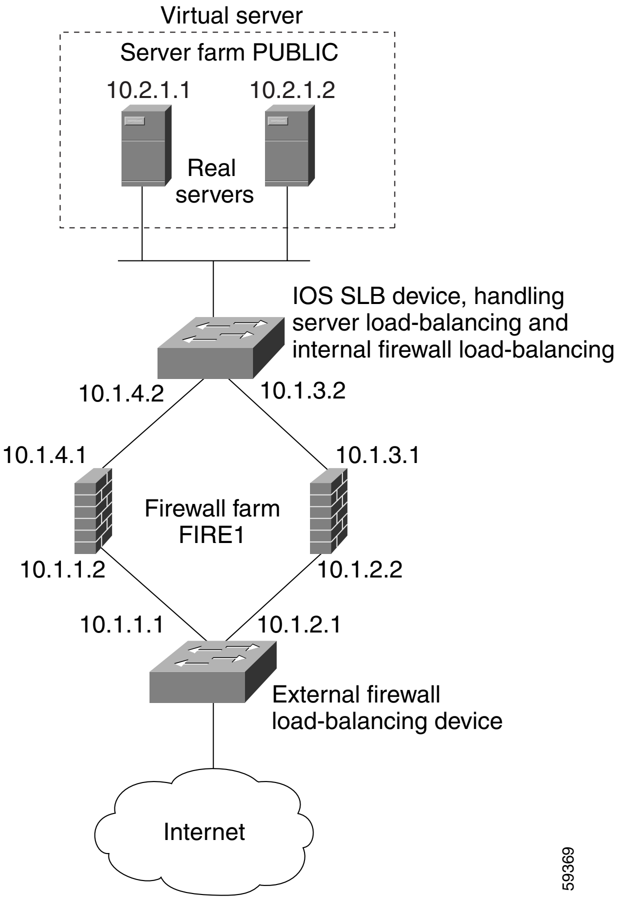
```
!-----Ping probe
ip slb probe PROBE1 ping
!-----IP address of other load-balancing device
  address 10.1.4.2
  faildetect 4
!-----HTTP probe
ip slb probe PROBE2 http
!-----IP address of other load-balancing device
  address 10.1.3.2
  expect status 401
!-----Firewall farm FIRE1
ip slb firewallfarm FIRE1
!-----First firewall
  real 10.1.1.2
    probe PROBE1
!-----Enable first firewall
    inservice

!-----Second firewall
  real 10.1.2.2
    probe PROBE2
!-----Enable second firewall
    inservice
    exit
  inservice
```

Configuring IOS SLB with Server Load Balancing and Firewall Load Balancing: Example

Figure 4 shows a sample IOS SLB load-balancing network with server load balancing and firewall load balancing running together, and the following components:

- Two real servers with IP addresses as shown
- One server farm named PUBLIC, containing both real servers
- Two firewalls with IP addresses as shown
- One firewall farm named FIRE1, containing both firewalls
- An internal IOS SLB device on the secure side of the firewalls, performing server load balancing and firewall load balancing
- An external firewall load-balancing device on the Internet side of the firewalls

Figure 4 IOS SLB with Server Load Balancing and Firewall Load Balancing



In the following firewall farm configuration samples:

- The internal (secure side) firewall load-balancing device is configured with firewall IP addresses 10.1.3.1 and 10.1.4.1.
- The external (Internet side) firewall load-balancing device is configured with firewall IP addresses 10.1.1.2 and 10.1.2.2.

Internal Server and Firewall Load-Balancing Device

The following example shows the configuration for ping probes ABCPROBE and XYZPROBE, firewall farm FIRE1, and server farm PUBLIC for the load-balancing device on the internal (secure) side of the firewalls:

```
ip slb probe ABCPROBE ping
  address 10.1.1.1
ip slb probe XYZPROBE ping
  address 10.1.2.1
!
ip slb firewallfarm FIRE1
  real 10.1.4.1
    probe ABCPROBE
    inservice
  real 10.1.3.1
    probe XYZPROBE
    inservice
  inservice
!
ip slb serverfarm PUBLIC
  nat server
  real 10.2.1.1
    inservice
    real 10.2.1.2
    inservice
!
ip slb vserver HTTP1
  virtual 128.1.0.1 tcp www
  serverfarm PUBLIC
  idle 120
  delay 5
  inservice
```

Note On Catalyst 6500 family switches, you can also specify that IOS SLB wildcard searches are to be performed by the route processor, using the **mls ip slb search wildcard rp** command in global configuration mode.

External Firewall Load-Balancing Device

The following example shows the configuration for ping probes ABCPROBE and XYZPROBE and firewall farm FIRE1 for the load-balancing device on the external (Internet) side of the firewalls:

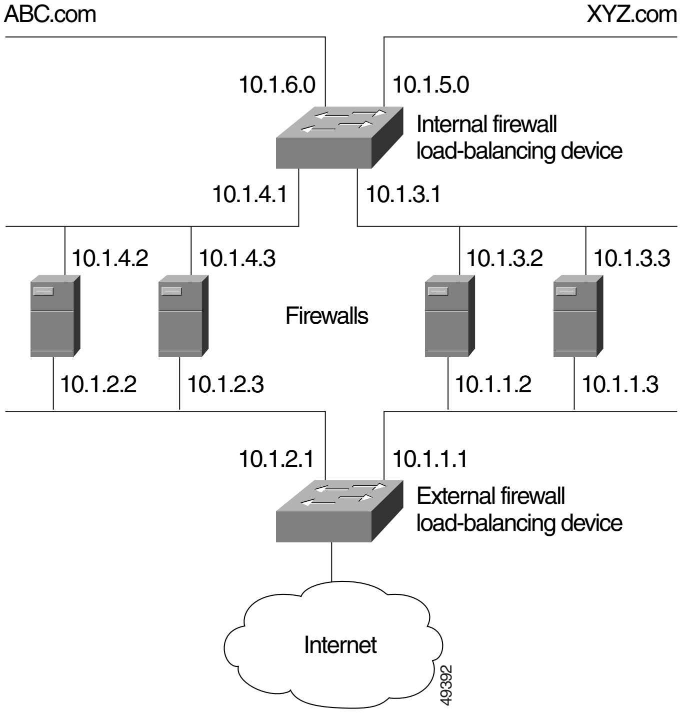
```
ip slb probe ABCPROBE ping
  address 10.1.4.2
  ip slb probe XYZPROBE ping
  address 10.1.3.2
  ip slb firewallfarm FIRE1
  real 10.1.1.2
    probe ABCPROBE
    inservice
    probe XYZPROBE
    inservice
```

Configuring IOS SLB with Multiple Firewall Farms: Example

Figure 5 shows a sample IOS SLB load-balancing network with multiple firewall farms and the following components:

- Four firewalls with IP addresses as shown
- An internal firewall load-balancing device on the secure side of the firewalls
- An external firewall load-balancing device on the Internet side of the firewalls
- One firewall farm named ABCFARM, containing the two firewalls on the left.
- One firewall farm named XYZFARM, containing the two firewalls on the right.

Figure 5 IOS SLB with Multiple Firewall Farms



In the following firewall farm configuration samples:

- The internal (secure side) firewall load-balancing device is configured with firewall IP addresses 10.1.3.1 and 10.1.4.1.
- The external (Internet side) firewall load-balancing device is configured with firewall IP addresses 10.1.1.2 and 10.1.2.2.

Internal Firewall Load-Balancing Device

The following example shows the configuration for ping probes ABCPROBE and XYZPROBE and firewall farms ABCFARM and XYZFARM for the load-balancing device on the internal (secure) side of the firewalls:

```
ip slb probe ABCPROBE ping
  address 10.1.2.1
ip slb probe XYZPROBE ping
  address 10.1.1.1
ip slb firewallfarm ABCFARM
  access source 10.1.6.0 255.255.255.0
  inservice
  real 10.1.4.2
    probe ABCPROBE
    inservice
  real 10.1.4.3
    probe ABCPROBE
    inservice
ip slb firewallfarm XYZFARM
  access source 10.1.5.0 255.255.255.0
  inservice
  real 10.1.3.2
    probe XYZPROBE
    inservice
  real 10.1.3.3
    probe XYZPROBE
    inservice
```

External Firewall Load-Balancing Device

The following example shows the configuration for ping probes ABCPROBE and XYZPROBE and firewall farms ABCFARM and XYZFARM for the load-balancing device on the external (Internet) side of the firewalls:

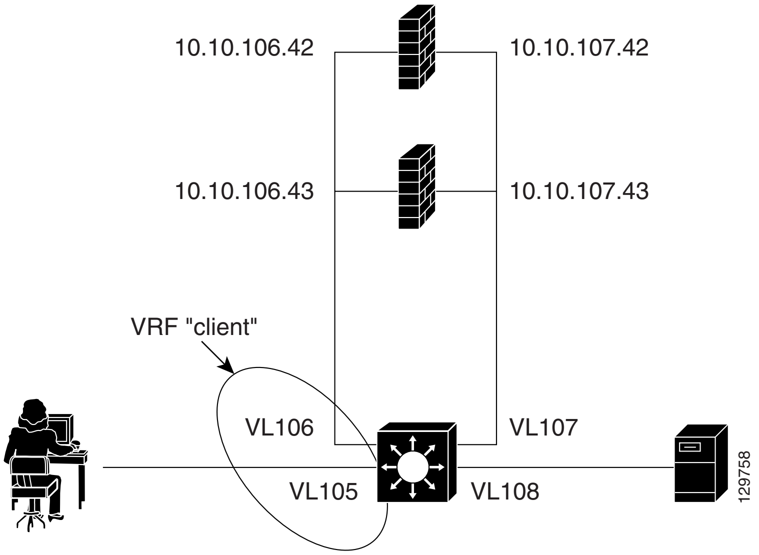
```
ip slb probe ABCPROBE ping
  address 10.1.4.1
ip slb probe XYZPROBE ping
  address 10.1.3.1
ip slb firewallfarm ABCFARM
  access destination 10.1.6.0 255.255.255.0
  inservice
  real 10.1.2.2
    probe ABCPROBE
    inservice
  real 10.1.2.3
    probe ABCPROBE
    inservice
ip slb firewallfarm XYZFARM
  access destination 10.1.5.0 255.255.255.0
  inservice
  real 10.1.1.2
    probe XYZPROBE
    inservice
  real 10.1.1.3
    probe XYZPROBE
    inservice
```

Configuring IOS SLB with Dual Firewall Load Balancing “Sandwich”: Example

Figure 6 illustrates a basic dual firewall load balancing “sandwich” configuration hosted on a single IOS SLB device, including Virtual Private Network (VPN) routing and forwarding (VRF) and access interface configuration. VL105, VL106, VL107, and VL108 are VLANs.

Note The client and server in this configuration are directly connected; in a more typical deployment, additional routes would be needed inside and outside the VRF.

Figure 6 IOS SLB with Dual Firewall Load Balancing "Sandwich"



Following are the IOS SLB configuration statements for the configuration shown in Figure 6:

```
ip vrf client
 rd 0:1
!
ip slb probe P642 ping
 address 10.10.106.42
 interval 120
ip slb probe P643 ping
 address 10.10.106.43
 interval 120
ip slb probe P742 ping
 address 10.10.107.42
 interval 120
ip slb probe P743 ping
 address 10.10.107.43
 interval 120
!
ip slb firewallfarm CLIENT
 access inbound Vlan105
 access outbound Vlan106
 no inservice
!
 real 10.10.106.42
  probe P642
  inservice
 real 10.10.106.43
  probe P643
  inservice
 protocol tcp
  sticky 180 source
 protocol datagram
  sticky 180 source
 predictor hash address port
!
ip slb firewallfarm SERVER
 access inbound Vlan108
 access outbound Vlan107
 inservice
!
```

```
 real 10.10.107.42
  probe P742
  inservice
 real 10.10.107.43
  probe P743
  inservice
 protocol tcp
  sticky 180 destination
 protocol datagram
  sticky 180 destination
 predictor hash address port
!
mls flow ip interface-full
!
!*************************************************
!* Switchports, port channels and trunks         *
!* added to vlans 105-108 (left out for brevity) *
!*************************************************
!
interface Vlan105
 ip vrf forwarding client
 ip address 10.10.105.2 255.255.255.0
!
interface Vlan106
 ip vrf forwarding client
 ip address 10.10.106.2 255.255.255.0
!
interface Vlan107
 ip address 10.10.107.2 255.255.255.0
!
interface Vlan108
 ip address 10.10.108.2 255.255.255.0
!
ip route 10.10.105.0 255.255.255.0 10.10.107.42
ip route vrf client 10.10.108.0 255.255.255.0 10.10.106.42
```

Configuring IOS SLB with Probes: Examples

This section contains the following examples, illustrating several different IOS SLB probe configurations:

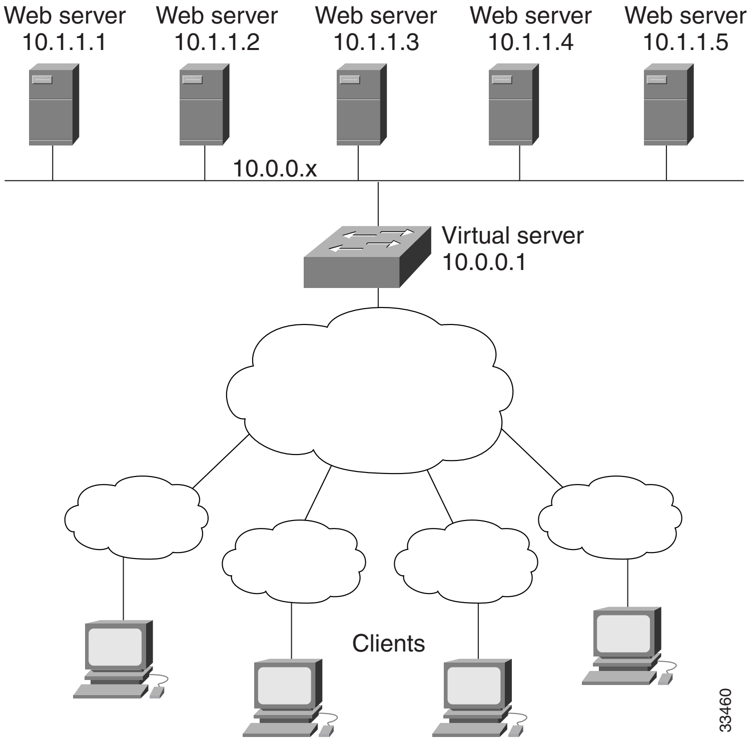
- Configuring IOS SLB with Ping and HTTP Probes: Example, page 117
- Configuring IOS SLB with Routed Probe: Example, page 119

Configuring IOS SLB with Ping and HTTP Probes: Example

Figure 7 shows a sample configuration with IOS SLB real server connections configured as part of a server farm, focusing on using ping and HTTP probes to monitor applications being server load-balanced.

Figure 7 Sample Ping and HTTP Probe Topology

:



The topology shown in Figure 7 is a heterogeneous server farm servicing a single virtual server. Following are the configuration statements for this topology, including a ping probe named PROBE1 and an HTTP probe named PROBE2:

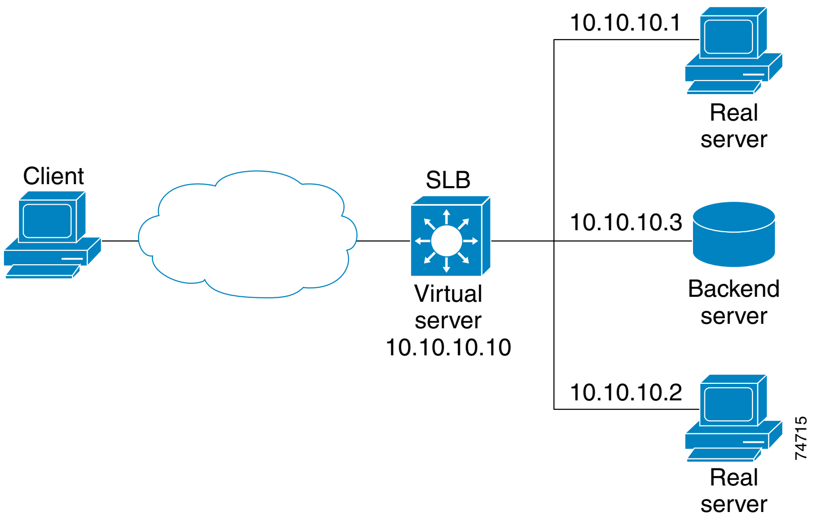
```
! Configure ping probe PROBE1, change CLI to IOS SLB probe configuration mode
ip slb probe PROBE1 ping
! Configure probe to receive responses from IP address 13.13.13.13
  address 13.13.13.13
! Configure unacknowledged ping threshold to 16
  faildetect 16
! Configure ping probe timer interval to send every 11 seconds
  interval 11
! Configure HTTP probe PROBE2
  ip slb probe PROBE2 http
! Configure request method as POST, set URL as /probe.cgi?all
  request method post url /probe.cgi?all
! Configure header HeaderName
  header HeaderName HeaderValue
! Configure basic authentication username and password
  credentials Semisweet chips
! Exit to global configuration mode
  exit
! Enter server farm configuration mode for server farm PUBLIC
ip slb serverfarm PUBLIC
! Configure NAT server and real servers on the server farm
  nat server
  real 10.1.1.1
   inservice
  real 10.1.1.2
   inservice
  real 10.1.1.3
   inservice
  real 10.1.1.4
```

```
   inservice
  real 10.1.1.5
   inservice
! Configure ping probe on the server farm
  probe PROBE1
! Configure HTTP probe on the server farm
  probe PROBE2
  end
```

Configuring IOS SLB with Routed Probe: Example

Figure 8 shows a typical datacenter and IOS SLB configuration. Virtual server ACME_VSERVER is configured with two real servers (10.10.10.1 and 10.10.10.2) in server farm ACME_FARM. The user wants the real servers to fail based on the health of the backend server (10.10.10.3). To accomplish this configuration without sending health checks via the real servers, the user defines BACKEND, a routed ping probe to the backend server's IP address.

Figure 8 IOS SLB with a Routed Ping Probe



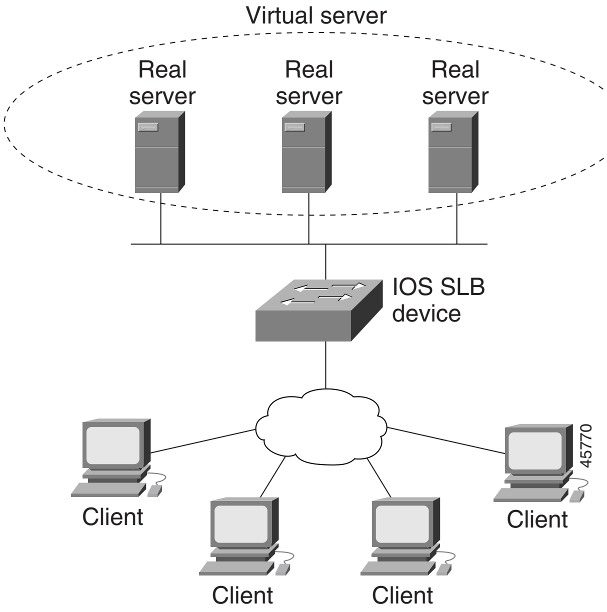
Following are the IOS SLB configuration statements for the configuration shown in Figure 8:

```
ip slb probe BACKEND ping
  address 10.10.10.3 routed

ip slb serverfarm ACME_SFARM
  nat server
  probe BACKEND
  real 10.10.10.1
   inservice
  real 10.10.10.2
   inservice
ip slb vserver ACME_VSERVER
  virtual 10.10.10.10 tcp 80
  serverfarm ACME_SFARM
  inservice
```

Configuring a Layer 3 Switch with IOS SLB: Example

Figure 9 shows a sample configuration with IOS SLB server connections configured as part of a server farm.

Figure 9 Network Configuration for IOS SLB



As shown in the following sample configuration, the example topology has three public web servers and two restricted web servers for privileged clients in subnet 10.4.4.0. The public web servers are weighted according to their capacity, with server 10.1.1.2 having the lowest capacity and having a connection limit imposed on it. The restricted web servers are configured as members of the same sticky group, so that HTTP connections and Secure Socket Layer (SSL) connections from the same client use the same real server.

The network configuration to provide the previously described IOS SLB functionality follows:

```
ip slb probe PROBE2 http
  request method POST url /probe.cgi?all
  header HeaderName HeaderValue
  header Authorization Basic U2VtaXN3ZWV0OmNoaXBz
!
ip slb serverfarm PUBLIC
  nat server
  predictor leastconns
! First real server
  real 10.1.1.1
    reassign 4
    faildetect numconns 16
    retry 120
    inservice
! Second real server
  real 10.1.1.2
    reassign 4
    faildetect numconns 16
    retry 120
    inservice
! Third real server
```

```
  real 10.1.1.3
    reassign 4
    faildetect numconns 16
    retry 120
    inservice
! Probe
  probe PROBE2
! Restricted web server farm
ip slb serverfarm RESTRICTED
  predictor leastconns
! First real server
  real 10.1.1.20
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
! Second real server
  real 10.1.1.21
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
!
! Unrestricted web virtual server
ip slb vserver PUBLIC_HTTP
  virtual 10.0.0.1 tcp www
  serverfarm PUBLIC
  idle 120
  delay 5
  inservice
!
! Restricted HTTP virtual server
ip slb vserver RESTRICTED_HTTP
  virtual 10.0.0.1 tcp www
  serverfarm RESTRICTED
  client 10.4.4.0 255.255.255.0
  sticky 60 group 1
  idle 120
  delay 5
  inservice
!
! Restricted SSL virtual server
ip slb vserver RESTRICTED_SSL
  virtual 10.0.0.1 tcp https
  serverfarm RESTRICTED
  client 10.4.4.0 255.255.255.0
  sticky 60 group 1
  idle 120
  delay 5
  inservice
!
interface GigabitEthernet1/1
  switchport
  switchport access vlan 3
  switchport mode access
  no ip address
!
interface FastEthernet2/1
  switchport
  switchport access vlan 2
  switchport mode access
  no ip address
!
interface FastEthernet2/2
```

```
  switchport
  switchport access vlan 2
  switchport mode access
  no ip address
!
interface FastEthernet2/3
  switchport
  switchport access vlan 2
  switchport mode access
  no ip address
!
interface Vlan2
  ip address 10.1.1.100 255.255.255.0
!
interface Vlan3
  ip address 40.40.40.1 255.255.255.0
```

Configuring IOS SLB with NAT and Static NAT: Examples

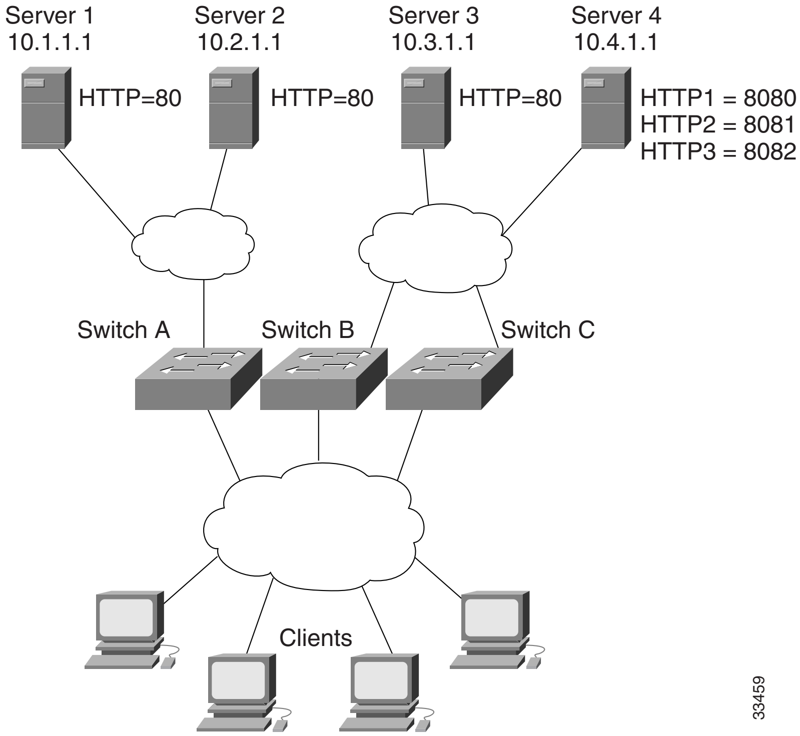
This section contains the following examples, illustrating several different IOS SLB NAT configurations:

- Configuring IOS SLB with NAT: Example, page 122
- Configuring IOS SLB with Static NAT: Example, page 125
- Configuring IOS SLB with GPRS Load Balancing and NAT: Example, page 153
- Configuring IOS SLB with GPRS Load Balancing, NAT, and GTP Cause Code Inspection: Example, page 156

Configuring IOS SLB with NAT: Example

Figure 10 shows a sample configuration with IOS SLB real server connections configured as part of a server farm, focusing on the configuration of the NAT server and address pool of clients.

Figure 10 Sample IOS SLB NAT Topology



The topology in Figure 10 has four web servers, configured as follows:

- Servers 1, 2, and 3 are running single HTTP server applications listening on port 80.
- Server 4 has multiple HTTP server applications listening on ports 8080, 8081, and 8082.

Server 1 and Server 2 are load-balanced using Switch A, which is performing server address translation.

Server 3 and Server 4 are load-balanced using Switch B and Switch C. These two switches are performing both server and client address translation since there are multiple paths between the clients and the servers. These switches also must perform server port translation for HTTP packets to and from Server 4.

Switch A Configuration Statements

```
ip slb serverfarm FARM1
! Translate server addresses
  nat server
! Server 1 port 80
  real 10.1.1.1
    reassign 2
    faildetect numconns 4 numclients 2
    retry 20
    inservice
! Server 2 port 80
  real 10.2.1.1
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
!
ip slb vserver HTTP1
! Handle HTTP (port 80) requests
```

```
  virtual 128.1.0.1 tcp www
  serverfarm FARM1
  idle 120
  delay 5
  inservice
```

Switch B Configuration Statements

```
ip slb natpool web-clients 128.3.0.1 128.3.0.254
! NAT address pool for clients
ip slb serverfarm FARM2
! Translate server addresses
  nat server
! Translate client addresses
  nat client web-clients
! Server 3 port 80
  real 10.3.1.1
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
! Server 4 port 8080
  real 10.4.1.1 port 8080
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
! Server 4 port 8081
  real 10.4.1.1 port 8081
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
! Server 4 port 8082
  real 10.4.1.1 port 8082
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
!
ip slb vserver HTTP2
! Handle HTTP (port 80) requests
  virtual 128.2.0.1 tcp www
  serverfarm FARM2
  idle 120
  delay 5
  inservice
```

Switch C Configuration Statements

```
ip slb natpool web-clients 128.5.0.1 128.5.0.254
! NAT address pool for clients
ip slb serverfarm FARM2
! Translate server addresses
  nat server
! Translate client addresses
  nat client web-clients
! Server 3 port 80
  real 10.3.1.1
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
```

```
! Server 4 port 8080
  real 10.4.1.1 port 8080
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
! Server 4 port 8081
  real 10.4.1.1 port 8081
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
! Server 4 port 8082
  real 10.4.1.1 port 8082
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
!
ip slb vserver HTTP2
! Handle HTTP (port 80) requests
  virtual 128.4.0.1 tcp www
  serverfarm FARM2
  idle 120
  delay 5
  inservice
```

Configuring IOS SLB with Static NAT: Example

The following example shows configuration statements for the following items:

- A DNS probe, PROBE4, configured to supply real server IP address 13.13.13.13 in response to domain name resolve requests.
- A server farm, DNS, that is configured to use server NAT and PROBE4.
- An all-port virtual server, 10.11.11.11, associated with server farm DNS, that performs per-packet server load balancing for UDP connections.
- A real server, 10.1.1.3, associated with server farm DNS, configured for static NAT and per-packet server load balancing.

```
ip slb probe PROBE4 dns
 lookup 13.13.13.13
!
ip slb serverfarm DNS
 nat server
 probe PROBE4
 real 10.1.1.3
  inservice
!
ip slb vserver DNS
 virtual 10.11.11.11 UDP 0 service per-packet
 serverfarm DNS
!
ip slb static nat 10.11.11.11 per-packet
 real 10.1.1.3
```

Configuring IOS SLB with Redundancy: Examples

This section contains the following examples, illustrating several different IOS SLB configurations with redundancy:

- Configuring IOS SLB with Stateless Backup: Examples, page 126
- Configuring IOS SLB with Stateful Backup: Example, page 135
- Configuring IOS SLB with Stateful Backup of Redundant Route Processors: Example, page 137
- Configuring IOS SLB with Active Standby: Example, page 138

Configuring IOS SLB with Stateless Backup: Examples

There are several different ways in which you can configure IOS SLB stateless backup. The differences between the configurations depend on the networking capabilities of your load balancing devices, and on the capabilities of the distribution devices that direct client traffic to those load balancing devices.

- If a load balancing device is capable of Layer 2 switching and VLAN trunking (such as the Catalyst 6500 family switch), you can wire the device directly to its real servers, and it can handle outbound flows from the real servers while acting as a standby for IOS SLB. HSRP is used on the server-side VLANs of the load balancing device, with the real servers routing to the HSRP address.
- If a load balancing device is *not* capable of both Layer 2 switching and VLAN trunking, you must connect it and its real servers to a Layer 2 switch. This configuration is required in order to use HSRP on the server-side VLANs.
- If a distribution device is capable of Layer 3 switching, it can use route redistribution to direct flows to the active load balancing device.
- If a distribution device is capable of Layer 2 switching, it can use client-side HSRP on the load balancing device to direct flows to the active load balancing device.
- While HSRP offers faster failover times, routing converges quickly enough for most configurations. If you use both client-side and server-side HSRP on the load balancing devices, you must use HSRP interface tracking and priorities to synchronize the client-side and server-side HSRP groups.

This section contains the following examples, illustrating several different IOS SLB stateless backup configurations:

- Configuring Dynamic Routing and Trunking: Example, page 126
- Configuring Dynamic Routing and No Trunking: Example, page 128
- Configuring Static Routing and Trunking: Example, page 130
- Configuring Static Routing and No Trunking: Example, page 132

Note Stateful backup is omitted from these examples in the interest of simplicity. To see an example that uses stateful backup, see the "Configuring IOS SLB with Stateful Backup: Example" section on page 135.

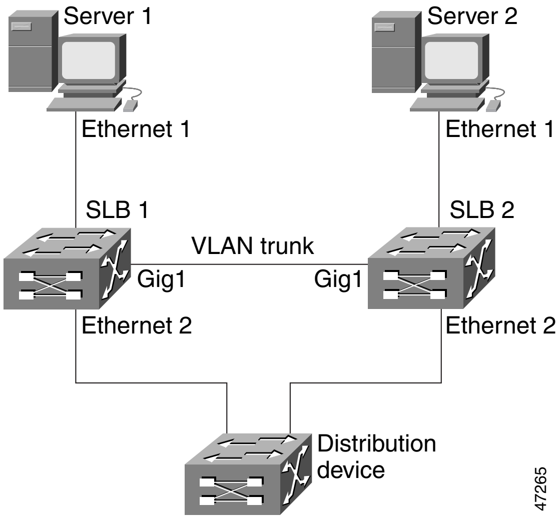
Configuring Dynamic Routing and Trunking: Example

Figure 11 shows a sample IOS SLB stateless backup configuration with the following characteristics:

- The IP address for real server 1 is 10.10.1.3, and for real server 2 is 10.10.1.4, routed to clients through 10.10.1.100.
- The IP address for the virtual server is 10.10.14.1.

- The IP address for VLAN 1 is 10.10.1.0, with a subnet mask of 255.255.255.0.
- The IP address for Subnet 2 is 10.10.2.0, with a subnet mask of 255.255.255.0.
- The IP address for Subnet 3 is 10.10.3.0, with a subnet mask of 255.255.255.0.
- The distribution device uses EIGRP to learn the route to 10.10.14.1 via either 10.10.2.1 or 10.10.3.1, depending on which IOS SLB is active.

Figure 11 Stateless Backup with Layer 3 and Trunking



SLB 1 Configuration Statements

```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4 numclients 2
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface Ethernet1
  switchport
  switchport vlan 1
interface Ethernet2
  ip address 10.10.2.1 255.255.255.0
interface vlan 1
  ip address 10.10.1.1 255.255.255.0
  standby ip 10.10.1.100
  standby priority 10 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
router eigrp 666
  redistribute static
  network 10.0.0.0
```

SLB 2 Configuration Statements

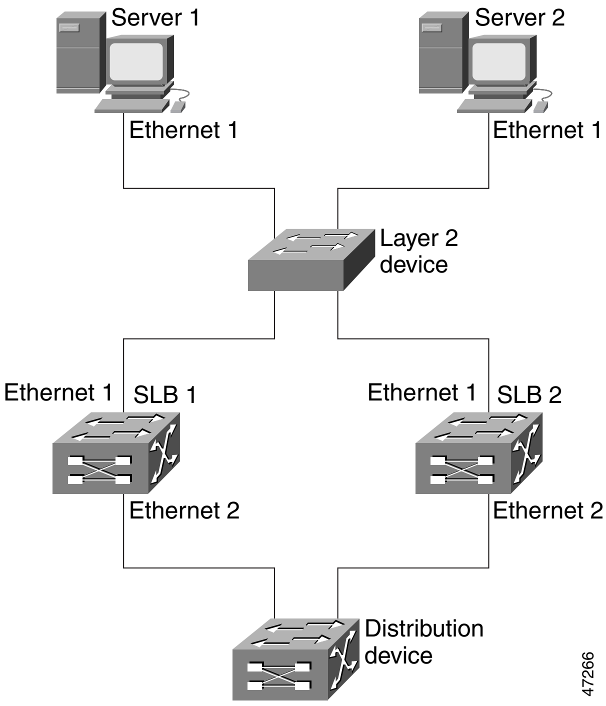
```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface GigabitEthernet1
  no ip address
  switchport
  switchport trunk encapsulation isl
interface Ethernet1
  switchport
  switchport vlan 1
interface Ethernet2
  ip address 10.10.3.1 255.255.255.0
interface vlan 1
  ip address 10.10.1.2 255.255.255.0
  standby ip 10.10.1.100
  standby priority 5 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
router eigrp 666
  redistribute static
  network 10.0.0.0
```

Configuring Dynamic Routing and No Trunking: Example

Figure 12 shows a sample IOS SLB stateless backup configuration with the following characteristics:

- The IP address for real server 1 is 10.10.1.3, and for real server 2 is 10.10.1.4, routed to clients through 10.10.1.100.
- The IP address for the virtual server is 10.10.14.1.
- The IP address for Subnet 2 is 10.10.2.0, with a subnet mask of 255.255.255.0.
- The IP address for Subnet 3 is 10.10.3.0, with a subnet mask of 255.255.255.0.
- The distribution device uses EIGRP to learn the route to 10.10.14.1 via either 10.10.2.2 or 10.10.3.2, depending on which IOS SLB is active.

Figure 12 Stateless Backup with Layer 3 and No Trunking



SLB 1 Configuration Statements

```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface Ethernet1
  ip address 10.10.1.1 255.255.255.0
  standby ip 10.10.1.100
  standby priority 10 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
interface Ethernet2
  ip address 10.10.2.1 255.255.255.0
router eigrp 666
  redistribute static
  network 10.0.0.0
```

SLB 2 Configuration Statements

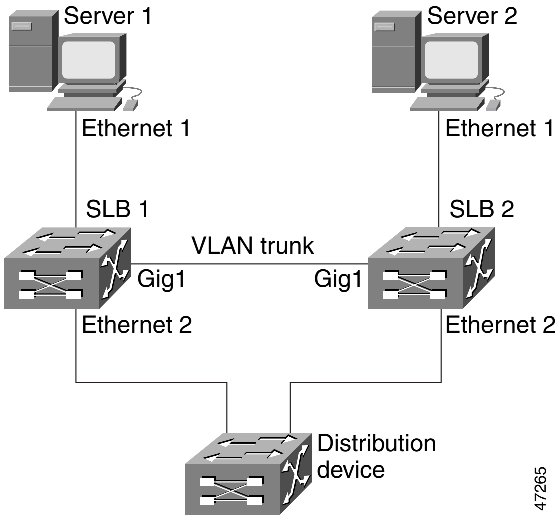
```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface Ethernet1
  ip address 10.10.1.2 255.255.255.0
  standby ip 10.10.1.100
  standby priority 5 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
interface Ethernet2
  ip address 10.10.3.1 255.255.255.0
router eigrp 666
  redistribute static
  network 10.0.0.0
```

Configuring Static Routing and Trunking: Example

Figure 13 shows a sample IOS SLB stateless backup configuration with the following characteristics:

- The IP address for real server 1 is 10.10.1.3, and for real server 2 is 10.10.1.4, routed to clients through 10.10.1.100.
- The IP address for the virtual server is 10.10.14.1.
- The IP address for VLAN 1 is 10.10.1.0, with a subnet mask of 255.255.255.0.
- The IP address for Subnet 2 is 10.10.2.0, with a subnet mask of 255.255.255.0.
- The IP address for Subnet 3 is 10.10.3.0, with a subnet mask of 255.255.255.0.
- The configuration uses static routing to the HSRP route on the distribution device.

Figure 13 Stateless Backup with Layer 2 and Trunking



SLB 1 Configuration Statements

```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface Ethernet1
  switchport
  switchport vlan 1
interface Ethernet2
  ip address 10.10.2.1 255.255.255.0
  standby ip 10.10.2.100
  standby priority 10 preempt delay sync 20
  standby track vlan1
  standby timers 1 3
interface vlan 1
  ip address 10.10.1.1 255.255.255.0
  standby ip 10.10.1.100
  standby priority 10 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
```

SLB 2 Configuration Statements

```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface GigabitEthernet1
  no ip address
  switchport
  switchport trunk encapsulation isl
interface Ethernet1
  switchport
  switchport vlan 1
interface Ethernet2
  ip address 10.10.2.2 255.255.255.0
  standby ip 10.10.2.100
  standby priority 5 preempt delay sync 20
  standby track vlan 1
  standby timers 1 3
interface vlan 1
  ip address 10.10.1.2 255.255.255.0
  standby ip 10.10.1.100
  standby priority 5 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
```

Distribution Device Configuration Statements

```
interface Ethernet1
  switchport
  switchport distribution vlan 2
interface Ethernet2
  switchport
  switchport distribution vlan 2
interface vlan2
  ip address 10.10.2.3 255.255.255.0
  no shut
ip route 10.10.14.1 255.255.255.255 10.10.2.100
```

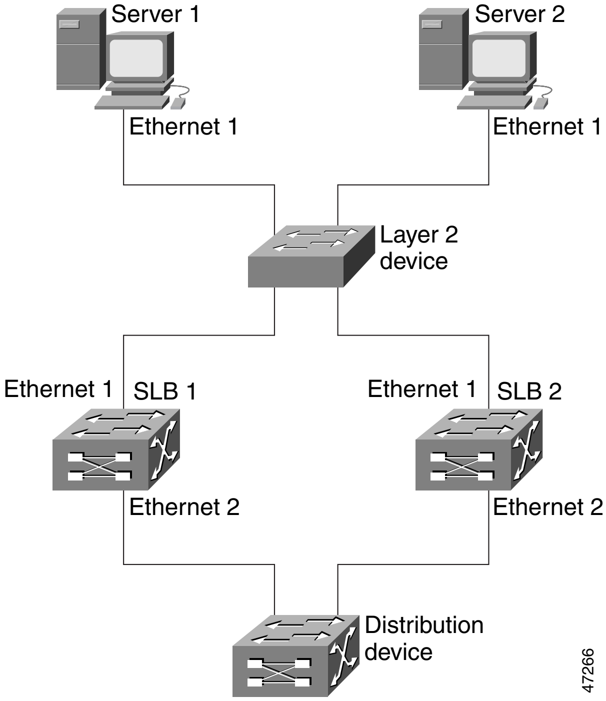
Configuring Static Routing and No Trunking: Example

Figure 14 shows a sample IOS SLB stateless backup configuration with the following characteristics:

- The IP address for real server 1 is 10.10.1.3, and for real server 2 is 10.10.1.4, routed to clients through 10.10.1.100.
- The IP address for the virtual server is 10.10.14.1.
- The IP address for Subnet 2 is 10.10.2.0, with a subnet mask of 255.255.255.0.

- The IP address for Subnet 3 is 10.10.3.0, with a subnet mask of 255.255.255.0.
- The configuration uses static routing to the HSRP route on the distribution device.

Figure 14 Stateless Backup with Layer 2 and No Trunking



SLB 1 Configuration Statements

```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface Ethernet1
  ip address 10.10.1.1 255.255.255.0
  standby ip 10.10.1.100
  standby priority 10 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
interface Ethernet2
  ip address 10.10.2.1 255.255.255.0
  standby ip 10.10.2.100
  standby priority 10 preempt delay sync 20
  standby track Ethernet1
  standby timers 1 3
```

SLB 2 Configuration Statements

```
ip slb serverfarm SF1
  real 10.10.1.3
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
  real 10.10.1.4
    reassign 2
    faildetect numconns 4
    retry 20
    inservice
ip slb vserver VS1
  virtual 10.10.14.1 tcp www
  serverfarm SF1
  idle 120
  delay 5
  inservice standby SERVER
!
interface Ethernet1
  ip address 10.10.1.2 255.255.255.0
  standby ip 10.10.1.100
  standby priority 5 preempt delay sync 20
  standby name SERVER
  standby track Ethernet2
  standby timers 1 3
!
interface Ethernet2
  ip address 10.10.2.2 255.255.255.0
  standby ip 10.10.2.100
  standby priority 5 preempt delay sync 20
  standby track Ethernet1
  standby timers 1 3
```

Distribution Device Configuration Statements

```
interface Ethernet1
  switchport
  switchport distribution vlan 2
interface Ethernet2
  switchport
  switchport distribution vlan 2
interface vlan2
  ip address 10.10.2.3 255.255.255.0
  no shut
ip route 10.10.14.1 255.255.255.255 10.10.2.100
```

Configuring IOS SLB with Stateful Backup: Example

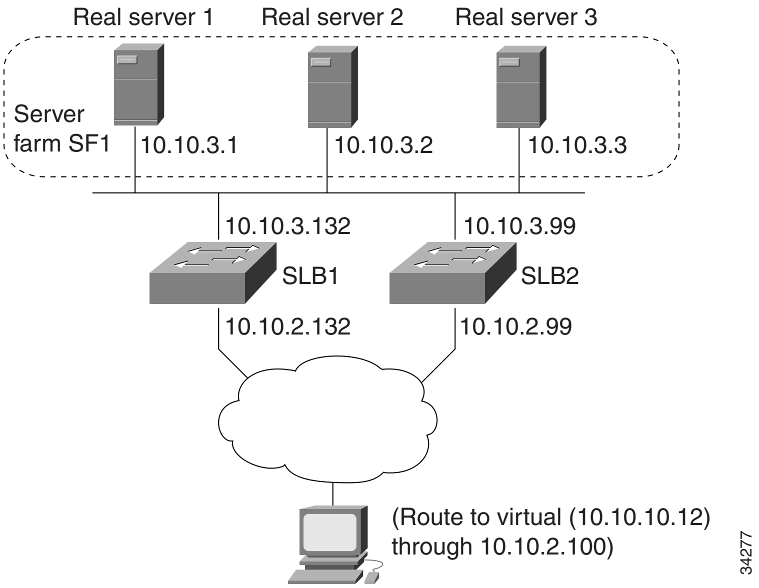
This sample configuration focuses on the IOS SLB real server connections configured as part of a server farm, with real and virtual servers over Fast Ethernet interfaces configured with stateful backup standby connections.

Figure 15 is an example of a stateful backup configuration, using HSRP on both the client and server sides to handle failover. The real servers route outbound flows to 10.10.3.100, which is the HSRP address on the server side interfaces. The client (or access router), routes to the virtual IP address (10.10.10.12) through 10.10.2.100, HSRP address on the client side.

Notice the loopback interfaces configured on both devices for the exchange of these messages. Each IOS SLB should also be given duplicate routes to the other switch loopback address. This configuration allows replication messages to flow despite an interface failure.

Note To allow HSRP to function properly, the **set spantree portfast** command must be configured on any Layer 2 device between the IOS SLB switches.

Figure 15 IOS SLB Stateful Environment



Switch SLB1 Configuration Statements

```
ip slb serverfarm SF1
  nat server
  real 10.10.3.1
   inservice
  real 10.10.3.2
   inservice
  real 10.10.3.3
   inservice
!
ip slb vserver VS1
  virtual 10.10.10.12 tcp telnet
  serverfarm SF1
  replicate casa 10.10.99.132 10.10.99.99 1024 password PASS
  inservice standby virt
!
interface loopback 1
  ip address 10.10.99.132 255.255.255.255
!
interface FastEthernet1
  ip address 10.10.3.132 255.255.255.0
  no ip redirects
  no ip mroute-cache
  standby priority 5 preempt
  standby name out
  standby ip 10.10.3.100
  standby track FastEthernet2
  standby timers 1 3
interface FastEthernet2
  ip address 10.10.2.132 255.255.255.0
  no ip redirects
  standby priority 5
  standby name virt
  standby ip 10.10.2.100
  standby timers 1 3
```

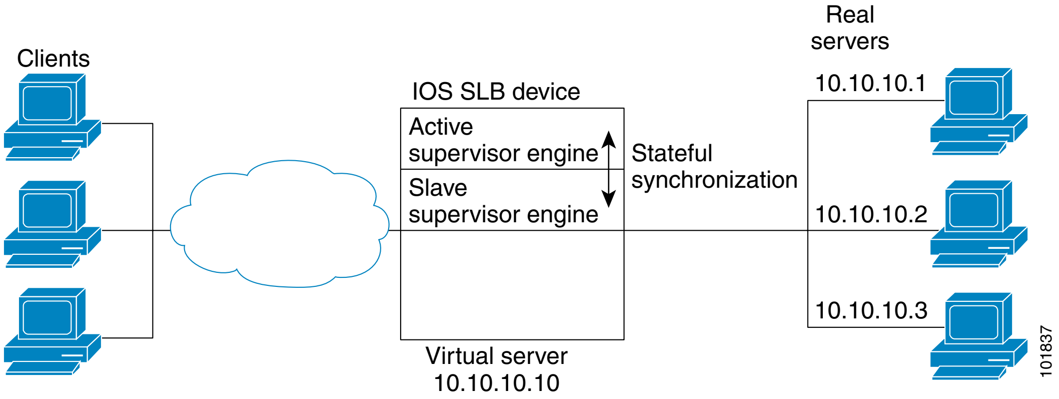
Switch SLB2 Configuration Statements

```
ip slb serverfarm SF1
  nat server
  real 10.10.3.1
   inservice
  real 10.10.3.2
   inservice
  real 10.10.3.3
   inservice
!
ip slb vserver VS1
  virtual 10.10.10.12 tcp telnet
  serverfarm SF1
  replicate casa 10.10.99.99 10.10.99.132 1024 password PASS
  inservice standby virt
!
interface loopback 1
  ip address 10.10.99.99 255.255.255.255
!
interface FastEthernet2
  ip address 10.10.2.99 255.255.255.0
  no ip redirects
  no ip route-cache
  no ip mroute-cache
  standby priority 10 preempt delay sync 20
  standby name virt
  standby ip 10.10.2.100
  standby track FastEthernet3
  standby timers 1 3
!
interface FastEthernet3
  ip address 10.10.3.99 255.255.255.0
  no ip redirects
  no ip route-cache
  no ip mroute-cache
  standby priority 10 preempt delay 20
  standby name out
  standby ip 10.10.3.100
  standby track FastEthernet2
  standby timers 1 3
```

Configuring IOS SLB with Stateful Backup of Redundant Route Processors: Example

In Figure 16, the IOS SLB device includes two Supervisor engines configured for stateful backup. If the active Supervisor engine fails, the backup Supervisor engine takes over via RPR+ with IOS SLB synchronization information already populated. IOS SLB replicates state information for virtual server ACME_VSERVER (10.10.10.10) from the active Supervisor engine to the backup every 20 seconds. The real servers (10.10.10.1, 10.10.10.2, and 10.10.10.3) are configured in server farm ACME_SFARM.

Figure 16 IOS SLB with Redundant Route Processors



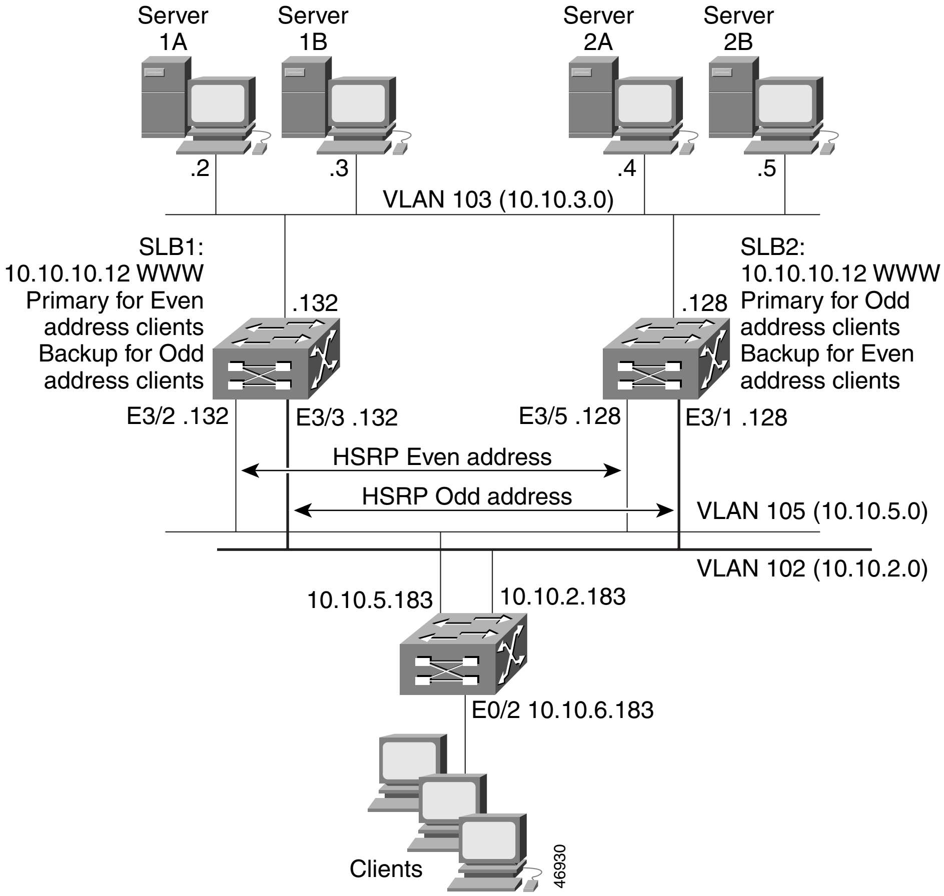
Following are the IOS SLB configuration statements for the configuration shown in Figure 16:

```
ip slb replicate slave rate 300

ip slb serverfarm ACME_SFARM
  nat server
  real 10.10.10.1
   inservice
  real 10.10.10.2
   inservice
  real 10.10.10.3
   inservice

ip slb vserver ACME_VSERVER
 virtual 10.10.10.10 tcp 80
 replicate interval 20
 replicate slave
 serverfarm ACME_SFARM
 inservice
```

Configuring IOS SLB with Active Standby: Example

Figure 17 shows an IOS SLB network configured for active standby, with two IOS SLB devices load-balancing the same virtual IP address while backing up each other. If either device fails, the other takes over its load via normal HSRP failover and IOS SLB stateless redundancy.

Figure 17 IOS SLB Active Standby



The sample network configuration in Figure 17 has the following characteristics:

- SLB 1 balances servers 1A and 1B and SLB 2 balances 2A and 2B.
- A single virtual IP address (10.10.10.12 for web) is supported across the two IOS SLB devices.
- Client traffic is divided in an access router, sending clients with even IP addresses to HSRP1 (10.10.5.100) and clients with odd IP addresses to HSRP2 (10.10.2.100). SLB 1 is configured as primary for clients with odd IP addresses, and SLB 2 is primary for clients with even IP addresses.
- The IOS SLB devices balance the traffic to disjoint sets of real servers. (If client NAT was used in this example, this characteristic would not be a requirement).
- Each set of real servers has a default gateway configured to its IOS SLB device.
- The HSRP address on VLAN 105 is 10.10.5.100. The HSRP address on VLAN 102 is 10.10.2.100.

SLB 1 Configuration Statements

```
ip slb serverfarm EVEN
 nat server
 real 10.10.3.2
  reassign 2
  faildetect numconns 4 numclients 2
  retry 20
  inservice
 real 10.10.3.3
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
!
ip slb serverfarm ODD
 nat server
 real 10.10.3.2
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
 real 10.10.3.3
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
!-----Same EVEN virtual server as in SLB 2
ip slb vserver EVEN
 virtual 10.10.10.12 tcp www
 serverfarm EVEN
 client 0.0.0.0 0.0.0.1
 idle 120
 delay 5
!-----See standby name in Ethernet 3/3 below
 inservice standby STANDBY_EVEN
!-----Same ODD virtual server as in SLB 2
ip slb vserver ODD
 virtual 10.10.10.12 tcp www
 serverfarm ODD
 client 0.0.0.1 0.0.0.1
 idle 120
 delay 5
!-----See standby name in Ethernet 3/2 below
 inservice standby STANDBY_ODD
!
interface Ethernet3/2
 ip address 10.10.5.132 255.255.255.0
 standby priority 20 preempt delay sync 20
!-----See standby name in SLB 2, Ethernet 3/5
 standby name STANDBY_ODD
 standby ip 10.10.5.100
 standby track Ethernet3/3
 standby timers 1 3
!
interface Ethernet3/3
 ip address 10.10.2.132 255.255.255.0
 standby priority 10
!-----See standby name in SLB 2, Ethernet 3/1
 standby name STANDBY_EVEN
 standby ip 10.10.2.100
 standby track Ethernet3/2
 standby timers 1 3
```

SLB 2 Configuration Statements

```
ip slb serverfarm EVEN
 nat server
 real 10.10.3.4
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
 real 10.10.3.5
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
!
ip slb serverfarm ODD
 nat server
 real 10.10.3.4
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
 real 10.10.3.5
  reassign 2
  faildetect numconns 4
  retry 20
  inservice
!-----Same EVEN virtual server as in SLB 1
ip slb vserver EVEN
 virtual 10.10.10.12 tcp www
 serverfarm EVEN
 client 0.0.0.0 0.0.0.1
 idle 120
 delay 5
!-----See standby name in Ethernet 3/1 below
 inservice standby STANDBY_EVEN
!-----Same ODD virtual server as in SLB 1
ip slb vserver ODD
 virtual 10.10.10.12 tcp www
 serverfarm ODD
 client 0.0.0.1 0.0.0.1
 idle 120
 delay 5
!-----See standby name in Ethernet 3/5 below
 inservice standby STANDBY_ODD
!
interface Ethernet3/1
 ip address 10.10.2.128 255.255.255.0
 standby priority 20 preempt delay sync 20
!-----See standby name in SLB 1, Ethernet 3/3
 standby name STANDBY_EVEN
 standby ip 10.10.2.100
 standby track Ethernet3/5
 standby timers 1 3
!
interface Ethernet3/5
 ip address 10.10.5.128 255.255.255.0
 standby priority 10 preempt delay sync 20
!-----See standby name in SLB 1, Ethernet 3/2
 standby name STANDBY_ODD
 standby ip 10.10.5.100
 standby track Ethernet3/1
 standby timers 1 3
```

Access Router Configuration Statements

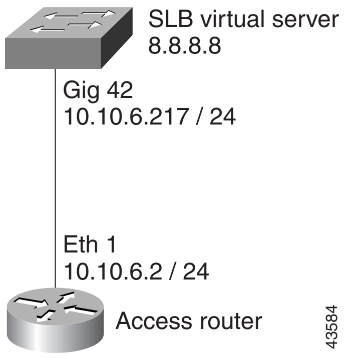
```
interface Ethernet0/0
 ip address 10.10.5.183 255.255.255.0
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet0/1
 ip address 10.10.2.183 255.255.255.0
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet0/2
 ip address 10.10.6.183 255.255.255.0
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
 ip policy route-map virts
!
access-list 100 permit ip 0.0.0.1 255.255.255.254 host 10.10.10.12
access-list 101 permit ip 0.0.0.0 255.255.255.254 host 10.10.10.12
route-map virts permit 10
match ip address 100
set ip next-hop 10.10.5.100
!
route-map virts permit 15
match ip address 101
set ip next-hop 10.10.2.100
```

Configuring IOS SLB with Redistribution of Static Routes: Example

Figure 18 shows an IOS SLB network configured to distribute static routes to a virtual server's IP address. The route to the address is added to the routing table as **static** if you advertise the address when you bring the virtual server into service (using the **inservice** command). See the description of the **advertise** command in the *Cisco IOS IP Application Services Command Reference* for more details about advertising virtual server IP addresses.

Because the routing configuration varies from protocol to protocol, sample configurations for several different routing protocols are given.

Figure 18 IOS SLB Redistribution of Static Routes



Routing Information Protocol (RIP)

Following is the RIP static route redistribution configuration for the IOS SLB switch shown in Figure 18:

```
router rip
 redistribute static
 network 10.0.0.0
 network 8.0.0.0
```

Following is the RIP static route redistribution configuration for the access router that is listening for routing updates shown in Figure 18:

```
router rip
 network 10.0.0.0
 network 8.0.0.0
```

Open Shortest Path First (OSPF)

Following is the OSPF static route redistribution configuration for the IOS SLB switch shown in Figure 18:

```
router ospf 1
 redistribute static subnets
 network 10.10.6.217 0.0.0.0 area 0
 network 8.8.8.0 0.0.0.255 area 0
```

Following is the OSPF static route redistribution configuration for the access router that is listening for routing updates shown in Figure 18:

```
router ospf 1
 network 10.10.6.2 0.0.0.0 area 0
 network 8.8.8.0 0.0.0.255 area 0
```

Interior Gateway Routing Protocol (IGRP)

Following is the IGRP static route redistribution configuration for the IOS SLB switch shown in Figure 18:

```
router igrp 1
 redistribute connected
 redistribute static
 network 8.0.0.0
 network 10.0.0.0
```

Following is the IGRP static route redistribution configuration for the access router that is listening for routing updates shown in Figure 18:

```
router igrp 1
 network 8.0.0.0
 network 10.0.0.0
```

Enhanced Interior Gateway Routing Protocol (Enhanced IGRP)

Following is the Enhanced IGRP static route redistribution configuration for the IOS SLB switch shown in Figure 18:

```
router eigrp 666
 redistribute static
 network 10.0.0.0
 network 8.0.0.0
```

Following is the Enhanced IGRP static route redistribution configuration for the access router that is listening for routing updates shown in Figure 18:

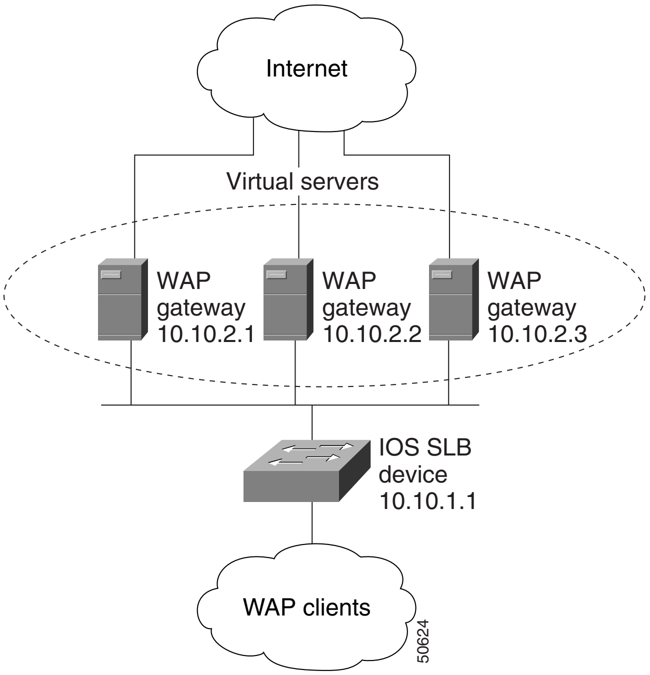
```
router eigrp 666
 network 10.0.0.0
 network 8.0.0.0
```

Configuring IOS SLB with WAP and UDP Load Balancing: Example

Figure 19 shows an IOS SLB network configured to balance WAP flows. In this example:

- WAP flows are balanced between WAP gateways 10.10.2.1, 10.10.2.2, and 10.10.2.3.
- The clients connect to 10.10.1.1, the IOS SLB virtual server address.
- For a given session, load-balancing decisions change if the connection idles longer than the virtual server’s idle connection timer (3000 seconds in this example).

Figure 19 IOS SLB with WAP Load Balancing



There are two ways to configure IOS SLB load balancing for WAP:

- To load-balance sessions running in connection-oriented WSP mode, define a WSP probe and use WAP load balancing. WAP load balancing requires a WAP virtual server configured on one of the WAP ports.
- To load-balance sessions running in connectionless WSP, connectionless secure WSP, and connection-oriented secure WSP modes, define a ping or WSP probe and use standard UDP load balancing with a low idle timer.

Balancing WAP Flows on UDP Port 9201

The following example shows the configuration for the IOS SLB device shown in Figure 19, which balances WAP flows on UDP port 9201 (WSP/WTP/UDP):

```
ip slb probe PROBE3 wsp
  url http://localhost/test.txt
!
ip slb serverfarm WAPFARM
  nat server
  real 10.10.2.1
  inservice
  real 10.10.2.2
  inservice
  real 10.10.2.3
  inservice
  probe PROBE3
!
ip slb vserver VSERVER
  virtual 10.10.1.1 udp 9201
  serverfarm WAPFARM
  idle 3000
  inservice
```

Balancing WAP Flows on UDP Port 9203

The following example shows the configuration for the IOS SLB device shown in Figure 19, which balances WAP flows on UDP port 9203 (WSP/WTP/WTLS/UDP):

```
ip slb probe PROBE1 ping
!
ip slb serverfarm WAPFARM
  nat server
  real 10.10.2.1
  inservice
  real 10.10.2.2
  inservice
  real 10.10.2.3
  inservice
  probe PROBE1
!
ip slb vserver VSERVER
  virtual 10.10.1.1 udp 9203
  serverfarm WAPFARM
  idle 3000
  inservice
```

Configuring IOS SLB with Route Health Injection: Examples

This section contains the following examples, illustrating several different IOS SLB route health injection configurations:

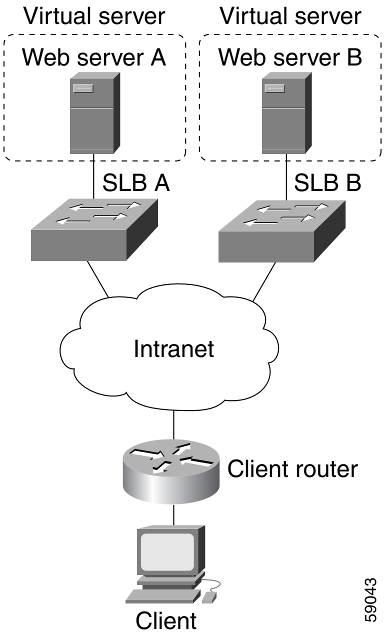
- Configuring Two Distributed Sites with One Web Server Each: Example, page 145
- Configuring Two Distributed Sites with Two Web Servers Each: Example, page 146
- Configuring Two Distributed Sites with One Web Server and a Backup IOS SLB Switch Each: Example, page 147

Configuring Two Distributed Sites with One Web Server Each: Example

Figure 20 shows an IOS SLB network configured with route health injection with the following characteristics:

- Both IOS SLB devices are configured with the same virtual IP address.
- Each IOS SLB device has a server farm containing only the locally attached web server as a real server.
- The path to SLB A has the lower weight.

Figure 20 Two Distributed Sites with One Web Server Each



When both web servers in Figure 20 are operational, the client router receives the host route from both IOS SLB devices.

If Web Server A fails, the virtual server for the virtual IP address on SLB A enters FAILED state and stops advertising the host route for the virtual IP address. The client router then begins using the route to SLB B.

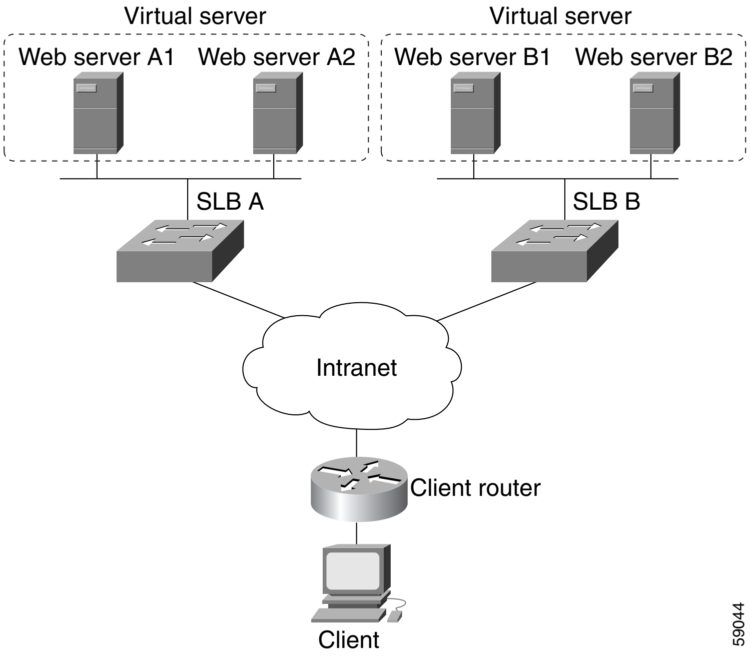
When Web Server A is again available, the virtual server again advertises the host route for the virtual IP address, and the client router begins using SLB A.

Configuring Two Distributed Sites with Two Web Servers Each: Example

Figure 21 shows an IOS SLB network configured with route health injection with the following characteristics:

- Both IOS SLB devices are configured with the same virtual IP address.
- Each IOS SLB device has a server farm containing two locally attached web servers as real servers.
- The path to SLB A has the lower weight.

Figure 21 Two Distributed Sites with Two Web Servers Each



When all web servers in Figure 21 are operational, the client router receives the host route from both IOS SLB devices.

If one web server in either server farm fails, the route continues to be advertised by the given IOS SLB device.

If both Web Server A1 and Web Server A2 fail, the virtual server for the virtual IP address on SLB A enters FAILED state and stops advertising the host route for the virtual IP address. The client router then begins using the route to SLB B.

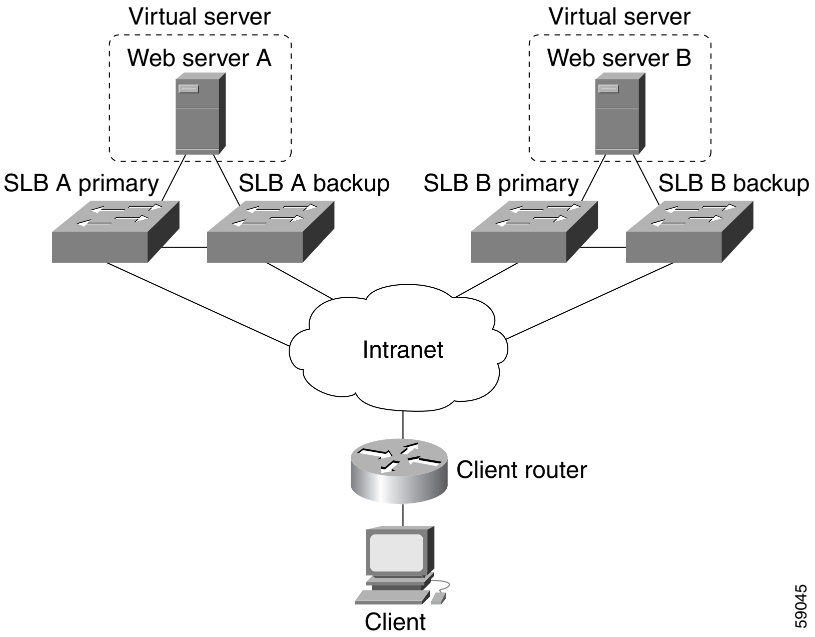
When either Web Server A1 or Web Server A2 is again available, the virtual server again advertises the host route for the virtual IP address, and the client router begins using SLB A.

Configuring Two Distributed Sites with One Web Server and a Backup IOS SLB Switch Each: Example

Figure 22 shows an IOS SLB network configured with route health injection with the following characteristics:

- Both IOS SLB devices are configured with the same virtual IP address.
- Each IOS SLB device has a server farm containing only the locally attached web server as a real server.
- Each site has a primary IOS SLB device and a backup IOS SLB device.
- The path to SLB A has the lower weight.

Figure 22 Two Distributed Sites with One Web Server and a Backup IOS SLB Switch Each



When both web servers in Figure 22 are operational, the client router receives the host route from both SLB A Primary and SLB B Primary.

If SLB A Primary fails, SLB A Backup begins advertising the host route to the virtual IP address. If SLB A Backup also fails, the virtual server for the virtual IP address on SLB A Primary and SLB A Backup enters FAILED state and stops advertising the host route for the virtual IP address. The client router then begins using the route to SLB B Primary (or to SLB B Backup, if SLB B Primary is not available).

When either SLB A Primary or SLB A Backup is again available, the virtual server again advertises the host route for the virtual IP address, and the client router begins using SLB A Primary or SLB A Backup.

Configuring IOS SLB with GPRS Load Balancing: Examples

This section contains the following examples, illustrating several different IOS SLB configurations with redundancy:

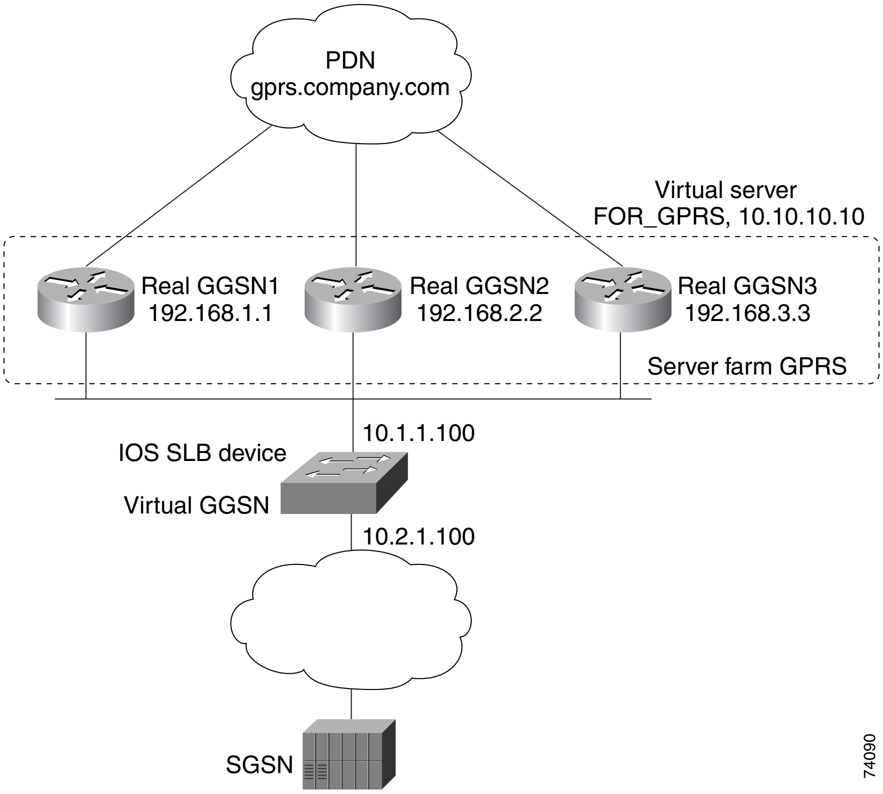
- Configuring IOS SLB with GPRS Load Balancing Without GTP Cause Code Inspection: Example, page 148
- Configuring IOS SLB with GPRS Load Balancing and NAT: Example, page 153
- Configuring IOS SLB with GPRS Load Balancing, NAT, and GTP Cause Code Inspection: Example, page 156
- Configuring IOS SLB with GPRS Load Balancing Maps: Example, page 157
- Configuring IOS SLB with RADIUS Load Balancing for a GPRS Network: Example, page 159

Configuring IOS SLB with GPRS Load Balancing Without GTP Cause Code Inspection: Example

Figure 23 shows a typical GPRS load-balancing configuration *without* GTP cause code inspection enabled. In this configuration:

- IOS SLB can balance GPRS flows across multiple real GGSNs. The SGSN “sees” the real GGSNs as a single virtual GGSN. This configuration increases the flow-handling capability of the real GGSNs and increases the reliability and availability.
- The virtual template address of the SGSN is 10.111.111.111.
- The virtual template address of GGSN1 is 192.168.1.1.
- The virtual template address of GGSN2 is 192.168.2.2.
- The virtual template address of GGSN3 is 192.168.3.3.

Figure 23 IOS SLB with GPRS Load Balancing



Following are the configuration statements for the configuration shown in Figure 23:

- IOS SLB Configuration Statements, page 149
- GGSN1 Configuration Statements, page 150
- GGSN2 Configuration Statements, page 151
- GGSN3 Configuration Statements, page 152

For more detailed GGSN configuration examples, refer to the *Cisco IOS Mobile Wireless Configuration Guide*.

IOS SLB Configuration Statements

```
hostname GTP_SLB
!
ip domain-name gprs.com
!
ip slb serverfarm GPRS
 real 192.168.1.1
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
 real 192.168.2.2
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
```

```
 real 192.168.3.3
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
ip slb vserver FOR_GPRS
 virtual 10.10.10.10 udp 3386 service gtp
 serverfarm GPRS
 inservice
!
ip slb dfp password Password1 0
 agent 10.1.1.201 1111 30 0 10
 agent 10.1.1.202 1111 30 0 10
 agent 10.1.1.203 1111 30 0 10
!
interface FastEthernet1/0
 description TO SERVERFARM GPRS
 ip address 10.1.1.100 255.255.255.0
 no ip redirects
 duplex half
!
interface FastEthernet3/0
 description TO SGSN
 ip address 10.2.1.100 255.255.255.0
 no ip mroute-cache
 duplex half
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
ip route 192.168.1.1 255.255.255.255 10.1.1.201
ip route 192.168.2.2 255.255.255.255 10.1.1.202
ip route 192.168.3.3 255.255.255.255 10.1.1.203
```

GGSN1 Configuration Statements

```
service gprs ggsn
!
hostname GGSN1
!
ip dfp agent gprs
 port 1111
 password Password1 0
 inservice
!
ip domain-name gprs.com
!
interface loopback 1
 description LOOPBACK SAME AS IOS SLB VSERVER ADDRESS
 ip address 10.10.10.10 255.255.255.255
 no ip route-cache
 no ip mroute-cache
!
interface FastEthernet1/0
 description TO SLB
 ip address 10.1.1.201 255.255.255.0
 ip directed-broadcast
 no ip mroute-cache
 duplex half
!
interface Virtual-Template1
 description GTP VIRTUAL TEMPLATE
 ip address 192.168.1.1 255.255.255.0
 encapsulation gtp
 gprs access-point-list gprs1
```

```
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
!
gprs access-point-list gprs1
  access-point 1
   access-point-name gprs.company.com
   access-mode non-transparent
   ip-address-pool dhcp-proxy-client
   dhcp-server 10.100.0.5 10.100.0.6
   dhcp-gateway-address 10.27.3.1
   exit
!
gprs maximum-pdp-context-allowed 45000
gprs qos map canonical-qos
gprs gtp path-echo-interval 0
gprs dfp max-weight 32
gprs slb cef 10.10.10.10
```

GGSN2 Configuration Statements

```
service gprs ggsn
!
hostname GGSN2
!
ip dfp agent gprs
 port 1111
 password Password1 0
 inservice
!
ip domain-name gprs.com
!
interface loopback 1
 description LOOPBACK SAME AS IOS SLB VSERVER ADDRESS
 ip address 10.10.10.10 255.255.255.255
 no ip route-cache
 no ip mroute-cache
!
interface FastEthernet1/0
 description TO SLB
 ip address 10.1.1.202 255.255.255.0
 ip directed-broadcast
 no ip mroute-cache
 duplex half
!
interface Virtual-Template1
 description GTP VIRTUAL TEMPLATE
 ip address 192.168.2.2 255.255.255.0
 encapsulation gtp
 gprs access-point-list gprs1
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
!
gprs access-point-list gprs1
  access-point 1
   access-point-name gprs.company.com
   access-mode non-transparent
   ip-address-pool dhcp-proxy-client
   dhcp-server 10.100.0.5 10.100.0.6
   dhcp-gateway-address 10.27.3.1
   exit
!
gprs maximum-pdp-context-allowed 45000
gprs qos map canonical-qos
gprs gtp path-echo-interval 0
gprs dfp max-weight 32
gprs slb cef 10.10.10.10
```

GGSN3 Configuration Statements

```
service gprs ggsn
!
hostname GGSN3
!
ip dfp agent gprs
 port 1111
 password Password1 0
 inservice
!
ip domain-name gprs.com
!
interface loopback 1
 description LOOPBACK SAME AS IOS SLB VSERVER ADDRESS
 ip address 10.10.10.10 255.255.255.255
 no ip route-cache
 no ip mroute-cache
!
interface FastEthernet1/0
 description TO SLB
 ip address 10.1.1.203 255.255.255.0
 ip directed-broadcast
 no ip mroute-cache
 duplex half
!
interface Virtual-Template1
 description GTP VIRTUAL TEMPLATE
 ip address 192.168.3.3 255.255.255.0
 encapsulation gtp
 gprs access-point-list gprs1
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
!
gprs access-point-list gprs1
  access-point 1
   access-point-name gprs.company.com
   access-mode non-transparent
   ip-address-pool dhcp-proxy-client
   dhcp-server 10.100.0.5 10.100.0.6
   dhcp-gateway-address 10.27.3.1
   exit
!
gprs maximum-pdp-context-allowed 45000
gprs qos map canonical-qos
gprs gtp path-echo-interval 0
gprs dfp max-weight 32
gprs slb cef 10.10.10.10
```

Configuring IOS SLB with GPRS Load Balancing and NAT: Example

The following example uses the same basic configuration as in the "Configuring IOS SLB with GPRS Load Balancing Without GTP Cause Code Inspection: Example" section on page 148, including the network shown in Figure 23, but with the addition of NAT:

- IOS SLB Configuration Statements, page 153
- GGSN1 Configuration Statements, page 154
- GGSN2 Configuration Statements, page 154
- GGSN3 Configuration Statements, page 155

For more detailed GGSN configuration examples, refer to the *Cisco IOS Mobile Wireless Configuration Guide*.

IOS SLB Configuration Statements

```
hostname GTP_SLB
!
ip domain-name gprs.com
!
ip slb serverfarm GPRS
 nat server
 real 192.168.1.1
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
 real 192.168.2.2
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
 real 192.168.3.3
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
ip slb vserver FOR_GPRS
 virtual 10.10.10.10 udp 3386 service gtp
 serverfarm GPRS
 inservice
!
ip slb dfp password Password1 0
 agent 10.1.1.201 1111 30 0 10
 agent 10.1.1.202 1111 30 0 10
 agent 10.1.1.203 1111 30 0 10
!
interface FastEthernet1/0
 description TO SERVERFARM GPRS
 ip address 10.1.1.100 255.255.255.0
 no ip redirects
 duplex half
!
interface FastEthernet3/0
 description TO SGSN
 ip address 10.2.1.100 255.255.255.0
 no ip mroute-cache
 duplex half
!
```

```
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
ip route 192.168.1.1 255.255.255.255 10.1.1.201
ip route 192.168.2.2 255.255.255.255 10.1.1.202
ip route 192.168.3.3 255.255.255.255 10.1.1.203
```

GGSN1 Configuration Statements

```
service gprs ggsn
!
hostname GGSN1
!
ip dfp agent gprs
 port 1111
 password Password1 0
 inservice
!
ip domain-name gprs.com
!
interface FastEthernet1/0
 description TO SLB
 ip address 10.1.1.201 255.255.255.0
 ip directed-broadcast
 no ip mroute-cache
 duplex half
!
interface Virtual-Template1
 description GTP VIRTUAL TEMPLATE
 ip address 192.168.1.1 255.255.255.0
 encapsulation gtp
 gprs access-point-list gprs1
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
!
gprs access-point-list gprs1
  access-point 1
   access-point-name gprs.company.com
   access-mode non-transparent
   ip-address-pool dhcp-proxy-client
   dhcp-server 10.100.0.5 10.100.0.6
   dhcp-gateway-address 10.27.3.1
   exit
!
gprs maximum-pdp-context-allowed 45000
gprs qos map canonical-qos
gprs gtp path-echo-interval 0
gprs dfp max-weight 32
```

GGSN2 Configuration Statements

```
service gprs ggsn
!
hostname GGSN2
!
ip dfp agent gprs
 port 1111
 password Password1 0
 inservice
!
ip domain-name gprs.com
!
interface FastEthernet1/0
 description TO SLB
 ip address 10.1.1.202 255.255.255.0
 ip directed-broadcast
 no ip mroute-cache
 duplex half
interface Virtual-Template1
 description GTP VIRTUAL TEMPLATE
 ip address 192.168.2.2 255.255.255.0
 encapsulation gtp
 gprs access-point-list gprs1
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
!
gprs access-point-list gprs1
  access-point 1
   access-point-name gprs.company.com
   access-mode non-transparent
   ip-address-pool dhcp-proxy-client
   dhcp-server 10.100.0.5 10.100.0.6
   dhcp-gateway-address 10.27.3.1
   exit
!
gprs maximum-pdp-context-allowed 45000
gprs qos map canonical-qos
gprs gtp path-echo-interval 0
gprs dfp max-weight 32
```

GGSN3 Configuration Statements

```
service gprs ggsn
!
hostname GGSN3
!
ip dfp agent gprs
 port 1111
 password Password1 0
 inservice
!
ip domain-name gprs.com
!
interface FastEthernet1/0
 description TO SLB
 ip address 10.1.1.203 255.255.255.0
 ip directed-broadcast
 no ip mroute-cache
 duplex half
!

interface Virtual-Template1
 description GTP VIRTUAL TEMPLATE
 ip address 192.168.3.3 255.255.255.0
 encapsulation gtp
 gprs access-point-list gprs1
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
!
gprs access-point-list gprs1
  access-point 1
   access-point-name gprs.company.com
   access-mode non-transparent
   ip-address-pool dhcp-proxy-client
   dhcp-server 10.100.0.5 10.100.0.6
   dhcp-gateway-address 10.27.3.1
   exit
!
```

```
gprs maximum-pdp-context-allowed 45000
gprs qos map canonical-qos
gprs gtp path-echo-interval 0
gprs dfp max-weight 32
```

Configuring IOS SLB with GPRS Load Balancing, NAT, and GTP Cause Code Inspection: Example

The following example uses the same basic configuration as in the "Configuring IOS SLB with GPRS Load Balancing and NAT: Example" section on page 153, including the network shown in Figure 23, but with the GTP cause code inspection enabled. In this configuration:

- The GSN idle timer is set to 20 seconds.
- The GTP request idle timer is set to 15 seconds.
- The virtual server accepts PDP context creates only from International Mobile Subscriber IDs (IMSIs) with carrier code **mcc 222 mnc 22**.

Following are the configuration statements for the configuration shown in Figure 23, with the addition of NAT and GTP cause code inspection support:

- IOS SLB Configuration Statements, page 156
- GGSN1 Configuration Statements, page 154 (no change for GTP cause code inspection)
- GGSN2 Configuration Statements, page 154 (no change for GTP cause code inspection)
- GGSN3 Configuration Statements, page 155 (no change for GTP cause code inspection)

For more detailed GGSN configuration examples, refer to the *Cisco IOS Mobile Wireless Configuration Guide*.

IOS SLB Configuration Statements

```
hostname GTP_SLB
!
ip domain-name gprs.com
!
ip slb timers gtp gsn 20
!
ip slb serverfarm GPRS
 nat server
 real 192.168.1.1
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
 real 192.168.2.2
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
 real 192.168.3.3
  weight 1
  faildetect numconns 1 numclients 1
  inservice
!
ip slb vserver FOR_GPRS
 virtual 10.10.10.10 udp 0 service gtp-inspect
 idle gtp request 15
 client gtp carrier-code mcc 222 mnc 22
 serverfarm GPRS
 inservice
```

```
!
ip slb dfp password Password1 0
 agent 10.1.1.201 1111 30 0 10
 agent 10.1.1.202 1111 30 0 10
 agent 10.1.1.203 1111 30 0 10
!
interface FastEthernet1/0
 description TO SERVERFARM GPRS
 ip address 10.1.1.100 255.255.255.0
 no ip redirects
 duplex half
!
interface FastEthernet3/0
 description TO SGSN
 ip address 10.2.1.100 255.255.255.0
 no ip mroute-cache
 duplex half
!
ip route 10.111.111.111 255.255.255.255 FastEthernet1/0
ip route 192.168.1.1 255.255.255.255 10.1.1.201
ip route 192.168.2.2 255.255.255.255 10.1.1.202
ip route 192.168.3.3 255.255.255.255 10.1.1.203
```

Configuring IOS SLB with GPRS Load Balancing Maps: Example

The following sample configuration enables IOS SLB to support multiple server farms behind a GPRS load balancing virtual server, using access point names (APNs) to select server farms. Server farm **farm6** is configured without an associated map, and therefore acts as a default server farm. If IOS SLB cannot match any of the other server farm maps, and a default server farm is configured, IOS SLB sends the GPRS request to the default serverfarm.

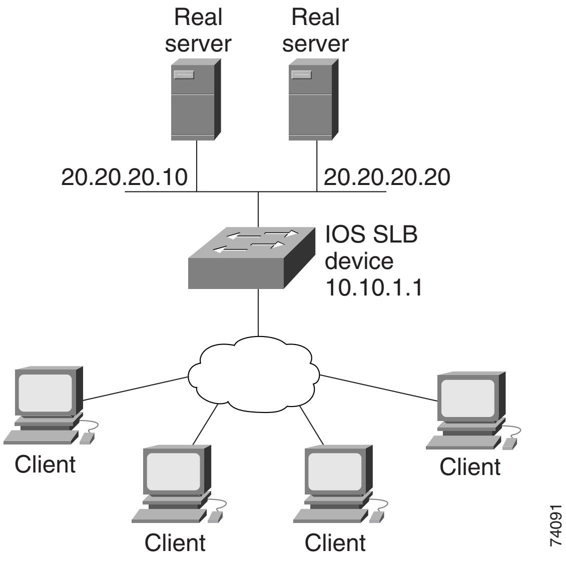
```
ip slb map 1 gtp
 apn cisco*
ip slb map 4 gtp
 apn abc.microsoft.com
 apn xyz.intel.com
ip slb map 5 gtp
 apn yahoo.com
!
ip slb serverfarm farm1
 real 10.0.0.1
 inservice
 real 10.0.0.2
 inservice
ip slb serverfarm farm2
 real 10.0.0.3
 inservice
 real 10.0.0.4
 inservice
ip slb serverfarm farm3
 real 10.0.0.5
 inservice
 real 10.0.0.6
 inservice
ip slb serverfarm farm4
 real 10.0.0.7
 inservice
 real 10.0.0.8
 inservice
ip slb serverfarm farm5
 real 10.0.0.9
 inservice
 real 10.0.0.10
 inservice
ip slb serverfarm farm6
 real 10.0.0.11
 inservice
!
ip slb map 1 gtp
 apn cisco*
ip slb map 4 gtp
 apn abc.microsoft.com
 apn xyz.intel.com
ip slb map 5 gtp
 apn yahoo.com
!
ip slb vserver GGSN_SERVER
 virtual 10.10.10.10 udp 0 service gtp
 serverfarm farm1 backup farm2 map 1 priority 3
 serverfarm farm4 map 4 priority 1
 serverfarm farm5 map 5 priority 4
 serverfarm farm6
 inservice
```

Configuring IOS SLB with VPN Server Load Balancing: Example

Figure 24 shows a typical VPN server load-balancing configuration. In this configuration:

- VPN flows are balanced between real servers 20.20.20.10 and 20.20.20.20.
- Clients connect to 10.10.1.1, the IOS SLB virtual server address.
- There is a sticky connection between the ESP virtual server and the UDP virtual server.
- The cryptographic key exchange occurs via IKE (ISAKMP; port 500).

Figure 24 IOS SLB with VPN Server Load Balancing



Following are the IOS SLB configuration statements for the configuration shown in Figure 24:

```
ip slb serverfarm VPN
 nat server
 real 20.20.20.10
  inservice
 real 20.20.20.20
  inservice
 failaction purge
!
ip slb vserver ESP
 virtual 10.10.1.1 ESP
 serverfarm VPN
 sticky 3600 group 69
 inservice
!
ip slb vserver UDP
 virtual 10.10.1.1 UDP isakmp
 serverfarm VPN
 sticky 3600 group 69
 inservice
```

Configuring IOS SLB with RADIUS Load Balancing: Examples

This section contains the following examples, illustrating several different IOS SLB RADIUS load-balancing configurations:

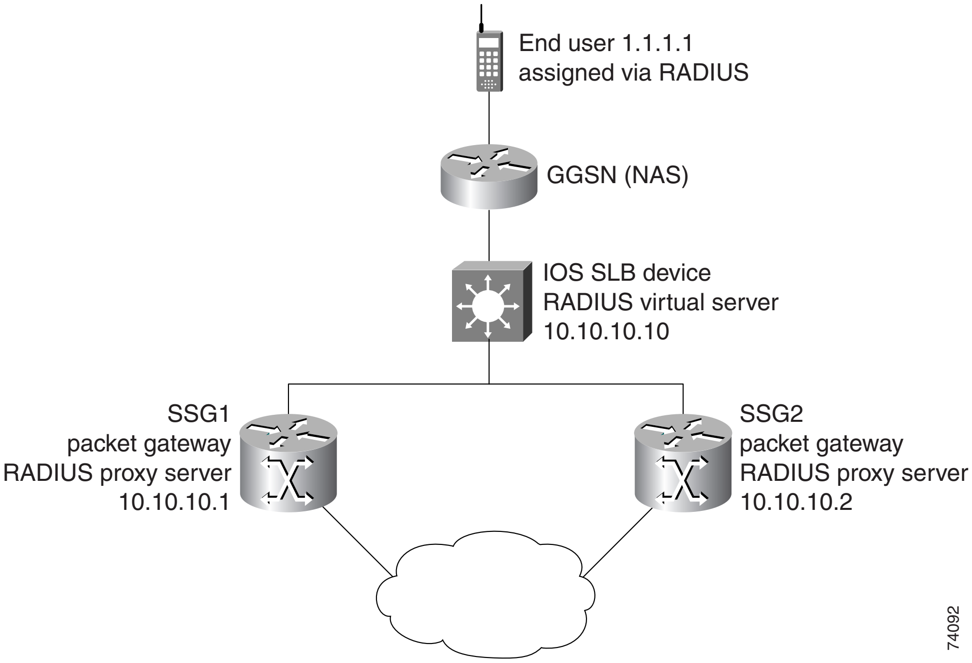
- Configuring IOS SLB with RADIUS Load Balancing for a GPRS Network: Example, page 159
- Configuring IOS SLB with RADIUS Load Balancing for a Simple IP CDMA2000 Network: Example, page 161
- Configuring IOS SLB with RADIUS Load Balancing for a Mobile IP CDMA2000 Network: Example, page 162
- Configuring IOS SLB with RADIUS Load Balancing for Multiple Service Gateway Farms: Example, page 163
- Configuring IOS SLB with RADIUS Load Balancing/Firewall Load Balancing “Sandwich”: Example, page 163
- Configuring IOS SLB with RADIUS Load Balancing Maps: Example, page 165
- Configuring IOS SLB with RADIUS Load Balancing Accelerated Data Plane Forwarding: Example, page 166

Configuring IOS SLB with RADIUS Load Balancing for a GPRS Network: Example

Figure 25 shows a typical IOS SLB RADIUS load-balancing configuration for a GPRS network. In this configuration:

- RADIUS requests are load-balanced between SSG RADIUS proxy servers 10.10.10.1 and 10.10.10.2.
- End-user data packets are routed to the IOS SLB device.
- End-user data packets from the 1.1.1.0 subnet are directed by IOS SLB to SSG1.
- End-user data packets from the 1.1.2.0 subnet are directed by IOS SLB to SSG2.

Figure 25 IOS SLB with RADIUS Load Balancing for a GPRS Network



Following are the IOS SLB configuration statements for the configuration shown in Figure 25:

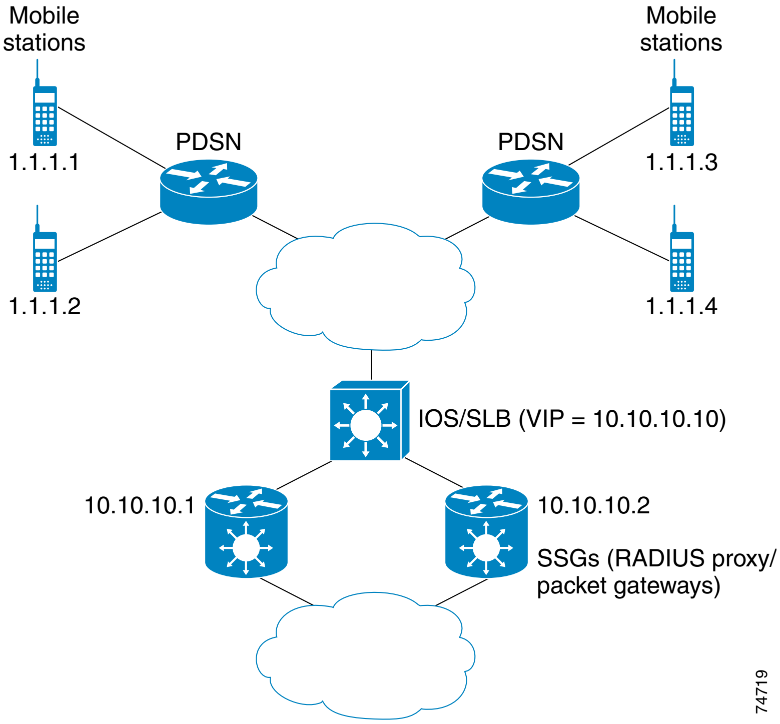
```
ip slb route 1.1.0.0 255.255.0.0 framed-ip
!
ip slb serverfarm SSGFARM
 nat server
 real 10.10.10.1
  inservice
 real 10.10.10.2
  inservice
!
ip slb vserver RADIUS_ACCT
 virtual 10.10.10.10 udp 1813 service radius
 serverfarm SSGFARM
 idle radius request 20
 idle radius framed-ip 7200
 sticky radius framed-ip group 1
 inservice
!
ip slb vserver RADIUS_AUTH
 virtual 10.10.10.10 udp 1812 service radius
 serverfarm SSGFARM
 idle radius request 20
 idle radius framed-ip 7200
 sticky radius framed-ip group 1
 inservice
```

Configuring IOS SLB with RADIUS Load Balancing for a Simple IP CDMA2000 Network: Example

Figure 26 shows a typical IOS SLB RADIUS load-balancing configuration for a CDMA2000 network with simple IP service. In this configuration:

- The RADIUS virtual server IP address for the PDSNs is 10.10.10.10.
- RADIUS requests are load-balanced between SSG RADIUS proxy servers 10.10.10.1 and 10.10.10.2.
- End-user data packets are routed to the IOS SLB device.
- End-user data packets from the 1.1.0.0 network are routed to the SSGs.

Figure 26 IOS SLB with RADIUS Load Balancing for a Simple IP CDMA2000 Network



Following are the IOS SLB configuration statements for the configuration shown in Figure 26:

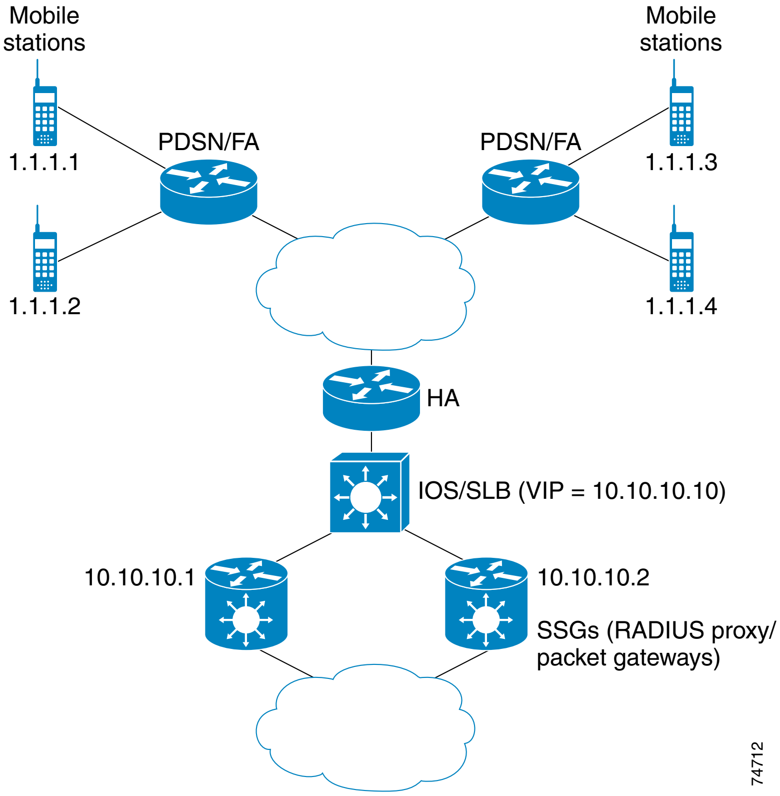
```
ip slb route 1.1.0.0 255.255.0.0 framed-ip
!
ip slb serverfarm SSGFARM
  nat server
  real 10.10.10.1
    inservice
  real 10.10.10.2
    inservice
!
ip slb vserver RADIUS_SIP
  virtual 10.10.10.10 udp 0 service radius
  serverfarm SSGFARM
  idle radius framed-ip 3600
  sticky radius username
  sticky radius framed-ip
  inservice
```

Configuring IOS SLB with RADIUS Load Balancing for a Mobile IP CDMA2000 Network: Example

Figure 27 shows a typical IOS SLB RADIUS load-balancing configuration for a CDMA2000 network with Mobile IP service. In this configuration:

- The RADIUS virtual server IP address for the PDSNs and the HA is 10.10.10.10.
- RADIUS requests are load-balanced between SSG RADIUS proxy servers 10.10.10.1 and 10.10.10.2.
- End-user data packets are routed to the IOS SLB device.
- End-user data packets from the 1.1.0.0 network are routed to the SSGs.

Figure 27 IOS SLB with RADIUS Load Balancing for a Mobile IP CDMA2000 Network



Following are the IOS SLB configuration statements for the configuration shown in Figure 27:

```
ip slb route 1.1.0.0 255.255.0.0 framed-ip
!
ip slb serverfarm SSGFARM
  nat server
  real 10.10.10.1
    inservice
  real 10.10.10.2
    inservice
!
ip slb vserver RADIUS_SIP
  virtual 10.10.10.10 udp 0 service radius
  serverfarm SSGFARM
  idle radius framed-ip 3600
  sticky radius username
  sticky radius framed-ip
  inservice
```

Configuring IOS SLB with RADIUS Load Balancing for Multiple Service Gateway Farms: Example

The following sample configuration enables IOS SLB to balance packet flows for a set of subscribers among multiple service gateway server farms (in this sample, a server farm of SSGs and a server farm of CSGs):

```
ip slb route 1.1.0.0 255.255.0.0 framed-ip
!
ip slb serverfarm SSGFARM
 nat server
 real 10.10.10.1
  inservice
 real 10.10.10.2
  inservice
!
ip slb serverfarm CSGFARM
 nat server
 real 20.20.20.1
  inservice
 real 20.20.20.2
  inservice
!
ip slb vserver SSG_AUTH
 virtual 10.10.10.10 udp 1812 service radius
 serverfarm SSGFARM
 idle radius request 20
 idle radius framed-ip 7200
 sticky radius framed-ip group 1
 access Vlan20 route framed-ip
 inservice
!
ip slb vserver SSG_ACCT
 virtual 10.10.10.10 udp 1813 service radius
 serverfarm SSGFARM
 idle radius request 20
 idle radius framed-ip 7200
 sticky radius framed-ip group 1
 access Vlan20 route framed-ip
 inservice
!
ip slb vserver CSG_ACCT
 virtual 20.20.20.20 udp 1813 service radius
 serverfarm CSGFARM
 idle radius request 25
 idle radius framed-ip 0
 sticky radius framed-ip
 access Vlan30 route framed-ip
 inservice
```

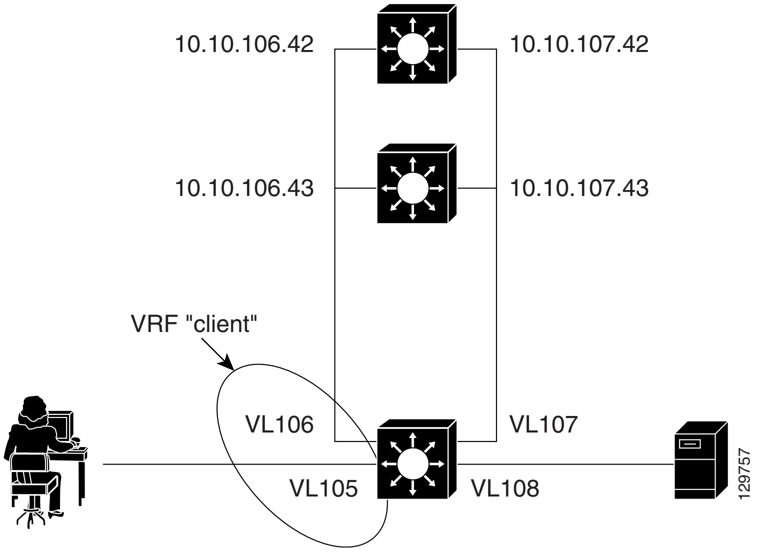
Configuring IOS SLB with RADIUS Load Balancing/Firewall Load Balancing “Sandwich”: Example

Figure 28 shows a RADIUS load balancing/firewall load balancing “sandwich” on a single IOS SLB device. In this sample configuration:

- The RADIUS load balancing virtual IP address is 5.5.5.5.
- The subscriber framed-IP network is 1.0.0.0/255.0.0.0.
- VL105, VL106, VL107, and VL108 are VLANs.

- RADIUS requests arriving on VLAN VL105 are balanced to 10.10.106.42 and 10.10.106.43.
- User traffic is stickied based on framed-IP address assignments in the 1.0.0.0 subnet.
- Firewall load balancing on the other side (10.10.107.42/43) ensures that return path traffic to the subscriber is delivered to the correct gateway.

Figure 28 IOS SLB with RADIUS Load Balancing/Firewall Load Balancing "Sandwich"



Following are the IOS SLB configuration statements for the configuration shown in Figure 28:

```
ip vrf client
 rd 0:1
!
ip slb probe P742 ping
 address 10.10.107.42
 interval 120
!
ip slb probe P743 ping
 address 10.10.107.43
 interval 120
!
ip slb route 1.0.0.0 255.0.0.0 framed-ip
ip slb route framed-ip deny
!
ip slb firewallfarm SERVER
 access inbound Vlan108
 access outbound Vlan107
 inservice
 real 10.10.107.42
  probe P742
  inservice
 real 10.10.107.43
  probe P743
  inservice
 protocol tcp
  sticky 180 destination
 protocol datagram
  sticky 180 destination
 predictor hash address port
!
```

```
ip slb serverfarm SF1
 nat server
 access Vlan106
!
 real 10.10.106.42
  inservice
 real 10.10.106.43
  inservice
!
ip slb vserver VS1
 virtual 5.5.5.5 udp 0 service radius
 serverfarm SF1
 sticky radius framed-ip
 access Vlan105 route framed-ip
 access Vlan105
 inservice
!
mls flow ip interface-full
!
!************************************************
!* Switchports, port channels and trunks         *
!* added to vlans 105-108 (left out for brevity) *
!************************************************
!
interface Vlan105
 ip vrf forwarding client
 ip address 10.10.105.2 255.255.255.0
!
interface Vlan106
 ip vrf forwarding client
 ip address 10.10.106.2 255.255.255.0
!
interface Vlan107
 ip address 10.10.107.2 255.255.255.0
!
interface Vlan108
 ip address 10.10.108.2 255.255.255.0
!
ip route 10.10.105.0 255.255.255.0 10.10.107.42
ip route vrf client 10.10.108.0 255.255.255.0 10.10.106.42
```

Configuring IOS SLB with RADIUS Load Balancing Maps: Example

The following sample configuration enables IOS SLB to support multiple server farms behind a RADIUS load balancing virtual server, using RADIUS calling station IDs and usernames to select server farms. Server farm **farm3** is configured without an associated map, and therefore acts as a default server farm. If IOS SLB cannot match any of the other server farm maps, and a default server farm is configured, IOS SLB sends the RADIUS request to the default serverfarm.

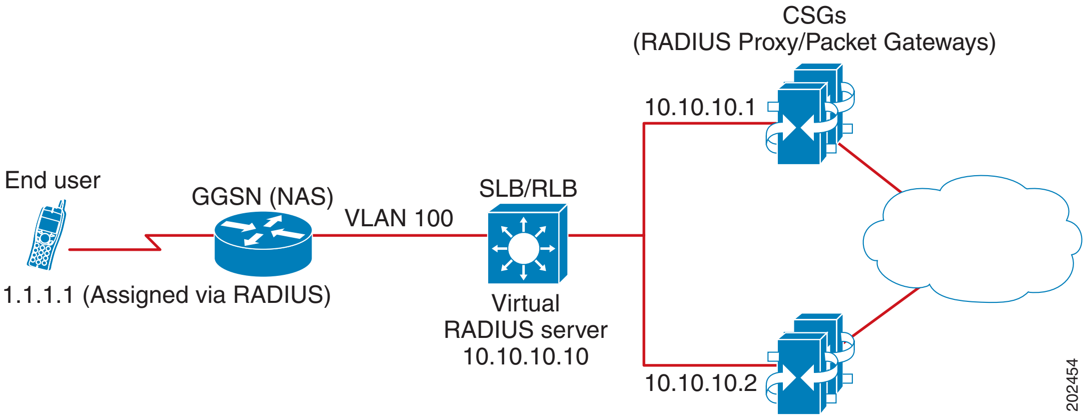
```
ip slb serverfarm CSGFARM
 predictor route-map rlb-pbr
ip slb serverfarm AAAFARM
 nat server
 real 10.10.10.1
  inservice
 real 10.10.10.2
  inservice

ip slb vserver RADIUS_ACCT
 virtual 10.10.10.10 udp 1813 service radius
 serverfarm CSGFARM
 radius inject acct 1 key 0 cisco
 inservice

ip slb vserver RADIUS_AUTH
 virtual 10.10.10.10 udp 1812 service radius
 serverfarm AAAFARM
 radius inject auth 1 calling-station-id
 radius inject auth timer 45
 radius inject auth vsa cisco
 inservice

!
interface vlan 100
 ip policy route-map rlb-pbr
!
access-list 1 permit 0.0.0.1 255.255.255.254
access-list 2 permit 0.0.0.0 255.255.255.254
!
route-map rlb-pbr permit 10
 match ip address 1
 set ip next-hop 10.10.10.1
!
route-map rlb-pbr permit 20
 match ip address 2
 set ip next-hop 10.10.10.2
```

Configuring IOS SLB with RADIUS Load Balancing Accelerated Data Plane Forwarding: Example

The following IOS SLB configuration has the following characteristics:

- There is a virtual RADIUS server with IP address 10.10.10.10 that handles Network Access Server (NAS) devices.
- There are two packet gateways with IP addresses 10.10.10.1 and 10.10.10.2.
- RADIUS traffic destined for the virtual RADIUS server is distributed between the packet gateways, based on mapped framed-IP addresses, according to route map **rlb-pbr**.
- Server farm CSGFARM is configured with real IP addresses that match the possible results for route map **rlb-pbr**.
- End user traffic arriving on VLAN 100 is routed to the correct Cisco Content Services Gateway (CSG) based on access control lists (ACLs):
  - ACL 1 sends IP addresses ending in odd numbers to the CSGs behind packet gateway 10.10.10.1.
  - ACL 2 sends IP addresses ending in even numbers to the CSGs behind packet gateway 10.10.10.2.

Figure 29 IOS SLB with RADIUS Load Balancing Accelerated Data Plane Forwarding



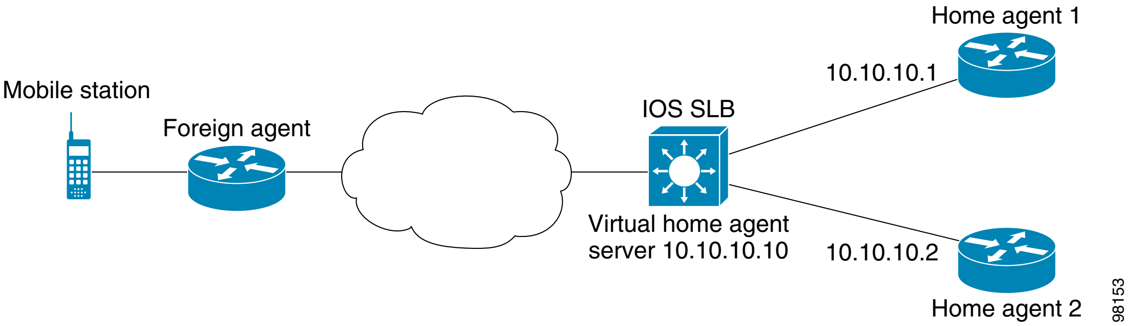
Following are the IOS SLB configuration statements for the configuration shown in Figure 29:

```
ip slb serverfarm CSGFARM
 predictor route-map rlb-pbr
ip slb serverfarm AAAFARM
 nat server
 real 10.10.10.1
  inservice
 real 10.10.10.2
  inservice
!
ip slb vserver RADIUS_ACCT
 virtual 10.10.10.10 udp 1813 service radius
 serverfarm CSGFARM
 radius inject acct 1 key 0 cisco
 inservice
!
ip slb vserver RADIUS_AUTH
 virtual 10.10.10.10 udp 1812 service radius
 serverfarm AAAFARM
 radius inject auth 1 calling-station-id
 radius inject auth timer 45
 radius inject auth vsa cisco
 inservice
!
interface vlan 100
 ip policy route-map rlb-pbr
!
access-list 1 permit 0.0.0.1 255.255.255.254
access-list 2 permit 0.0.0.0 255.255.255.254
!
route-map rlb-pbr permit 10
 match ip address 1
 set ip next-hop 10.10.10.1
!
route-map rlb-pbr permit 20
 match ip address 2
 set ip next-hop 10.10.10.2
```

Configuring IOS SLB with Home Agent Director: Example

The following sample configuration enables IOS SLB to balance Mobile IP RRQs among multiple home agents.

Figure 30 IOS SLB with Home Agent Director



Following are the IOS SLB configuration statements for the configuration shown in Figure 30:

```
ip slb serverfarm HA_FARM
 nat server
 real 10.10.10.1
  inservice
 real 10.10.10.2
  inservice

ip slb vserver VIRTUAL_HA
 virtual 10.10.10.10 udp 434 service ipmobile
 serverfarm HA_FARM
 inservice
```

Configuring IOS SLB with Sticky Connections: Example

The following sample configuration assigns all HTTP connections from a subnet to the same real server in server farm PUBLIC:

```
ip slb vserver http
  serverfarm PUBLIC
  sticky 30 group 1 netmask 255.255.255.248
  virtual 20.20.20.20 tcp 80
  inservice
```

The following sample configuration adds HTTPS connections to the above configuration, using the same sticky information but with a different virtual server:

```
ip slb vserver https
  serverfarm PUBLIC
  sticky 30 group 1 netmask 255.255.255.248
  virtual 20.20.20.20 tcp 443
  inservice
```

Now, all HTTP *and* HTTPS connections from the subnet are assigned to the same real server. For example, if a user connects to HTTP, then a second user connects to HTTPS, both connections are assigned to the same real server.

Configuring IOS SLB with GTP IMSI Sticky Database: Example

The following sample configuration shows how to enable the IOS SLB GTP IMSI sticky database:

```
ip slb serverfarm GGSN_FARM
 failaction gtp purge
 real 10.20.10.1
  weight 1
  faildetect numconns 255 numclients 8
  inservice
!
 real 10.20.10.2
  weight 1
  faildetect numconns 255 numclients 8
  inservice
!
 real 10.20.10.3
  weight 1
  faildetect numconns 255 numclients 8
  inservice
!
ip slb vserver GGSN_SERVER1
 virtual 10.10.10.10 udp 3386 service gtp
 serverfarm GGSN_FARM backup GGSN_FARM
 idle gtp request 90
 idle gtp imsi 10000000
 sticky gtp imsi group 1
 gtp notification cac 3
 inservice
!
ip slb vserver GGSN_SERVER2
 virtual 10.10.10.10 udp 2123 service gtp
 serverfarm GGSN_FARM backup GGSN_FARM
 idle gtp request 90
 idle gtp imsi 10000000
 sticky gtp imsi group 1
 gtp notification cac 3
 inservice
```

Configuring IOS SLB with Transparent Web Cache Load Balancing: Example

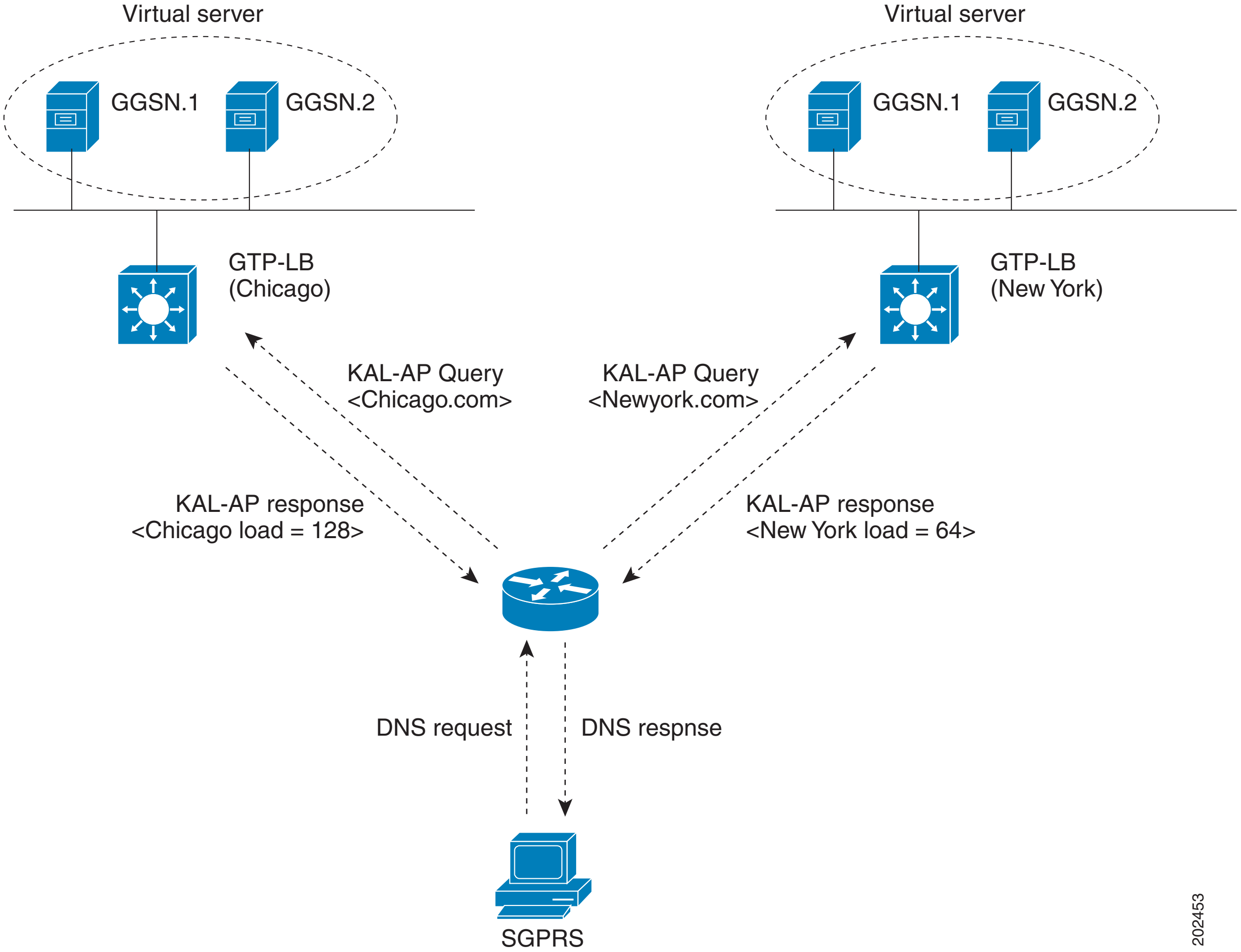
In the following sample configuration, virtual server WEBCACHE examines all web flows passing through the load-balancing device and dispatches them to server farm WEBCACHE-FARM. The **client exclude** statement ignores flows originating from subnet 80.80.7.0, enabling the real servers 80.80.7.188 and 80.80.7.189 to communicate with the Internet as needed.

```
ip slb serverfarm WEBCACHE-FARM
  real 80.80.7.188
   inservice
  real 80.80.7.189
   inservice
ip slb vserver WEBCACHE
  virtual 0.0.0.0 0.0.0.0 tcp www
  serverfarm WEBCACHE-FARM
  client 80.80.7.0 255.255.255.0 exclude
  inservice
```

Configuring IOS SLB with KAL-AP Agent: Example

In the following sample configuration, the client is configured to send a Domain Name System (DNS) query **abcd.com** to the GSS. The Global Site Selector (GSS) in the DUBLIN site receives the requests from the client. The GSS answers the DNS query with the virtual IP address of either CHICAGO (10.0.0.100) or NEWYORK (10.0.0.200), based on the load reported by those virtual servers.

Figure 31 IOS SLB with KAL-AP Agent



Following are the IOS SLB configuration statements for the configuration shown in Figure 31:

GSS

```
shared-keepalive kalap 192.168.1.1 capp-secure enable key kap
shared-keepalive kalap 192.168.2.1 capp-secure enable key kap
!
answer vip 10.0.0.100 name CHICAGO activate
 keepalive type kalap tag 192.168.1.1 chicao.com
answer vip 10.0.0.200 name NEWYORK activate
 keepalive type kalap tag 192.168.2.1 newyork.com
!
```

```
answer-group ABCD owner System type vip
answer-add 10.0.0.100 name CHICAGO weight 1 order 0 load-threshold 254 activate
answer-add 10.0.0.200 name NEWYORK weight 1 order 0 load-threshold 254 activate
dns rule ABCDGPRS owner System source-address-list Anywhere domain-list abcd.com query a
 clause 1 vip-group method least-loaded ttl 20 count 1 sticky disable
```

Site-1: IOS SLB - CHICAGO

```
ip slb capp udp
 peer port 6000 secret 0 kap
!
ip slb serverfarm SF
 kal-ap domain chicago.com
 farm-weight 200
 real 10.10.10.1
  inservice
 real 10.10.10.2
  inservice
!
ip slb vserver chicago
 virtual 10.0.0.100 udp 0
 serverfarm SF
 inservice
!
ip slb dfp
 agent 10.10.10.1 5000 30 0 10
 agent 10.10.10.2 5000 30 0 10
!
int vlan100
 ip address 192.168.1.1 255.255.255.0
```

GGSN-1

```
gprs dfp max-weight 100
gprs maximum-pdp-context-allowed 20000
!
ip dfp agent gprs
 port 5000
 inservice
```

GGSN-2

```
gprs dfp max-weight 100
gprs maximum-pdp-context-allowed 20000
!
ip dfp agent gprs
 port 5000
 inservice
```

Site-2: IOS SLB - NEWYORK

```
ip slb capp udp
 peer port 6000
 peer 192.1.1.1 secret 0 test
 peer 10.100.100.100 port 1234
!
ip slb serverfarm SF
 kal-ap domain newyork.com
 farm-weight 6200
 real 10.20.20.1
```

```
  inservice
 real 10.20.20.2
  inservice
 real 10.20.20.3
  inservice
 real 10.20.20.4
  inservice
!
ip slb vserver chicago
 virtual 10.0.0.200 udp 0
 serverfarm SF
 inservice
!
ip slb dfp
 agent 10.10.10.1 5000 30 0 10
 agent 10.10.10.2 5000 30 0 10
!
int vlan200
 ip address 192.168.2.1 255.255.255.0
```

GGSN-1

```
gprs dfp max-weight 100
gprs maximum-pdp-context-allowed 20000
!
ip dfp agent gprs
 port 5000
 inservice
```

GGSN-2

```
gprs dfp max-weight 100
gprs maximum-pdp-context-allowed 20000
!
ip dfp agent gprs
 port 5000
 inservice
```

Additional References

The following sections provide references related to IOS SLB.

- Troubleshooting, page 173
- Supported Platforms, page 175
- Supported Platforms, page 175
- Standards, page 175
- MIBs, page 176
- RFCs, page 176
- Technical Assistance, page 176

Troubleshooting

Question	Answer
Can I use IOS SLB to load-balance clients and real servers that are on the same LAN or VLAN?	**NO!** **IOS SLB does not support load balancing of flows between clients and real servers that are on the same LAN or VLAN. The packets being load-balanced cannot enter and leave the load-balancing device on the same interface.**
Why is IOS SLB not marking my connections as ESTABLISHED even though I'm transferring data?	If you are using dispatched mode, make sure there are no alternate paths that allow outbound flows to bypass IOS SLB. Also, make sure the clients and real servers are not on the same IP subnet (that is, they are not on the same LAN or VLAN).
Why am I able to connect to real servers directly, but unable to connect to the virtual server?	Make sure that the virtual IP address is configured as a loopback in each of the real servers (if you are running in dispatched mode).
Why is IOS SLB not marking my real server as failed when I disconnect it from the network?	Tune the values for the **numclients**, **numconns**, and **delay** keywords. If you have a very small client population (for example, in a test environment), the **numclients** keyword could be causing the problem. This parameter prevents IOS SLB from mistaking the failure of a small number of clients for the failure of a real server.
Why does IOS SLB show my real server as INSERVICE even though I have taken it down or physically disconnected it?	The INSERVICE and OUTOFSERVICE states indicate whether the network administrator *intends* for that real server to be used when it is operational. A real server that was INSERVICE but was removed from the selection list dynamically by IOS SLB as a result of automatic failure detection, is marked as FAILED. Use the **show ip slb reals detail** command to display these real server states. Beginning with release 12.1(1)E, INSERVICE is changed to OPERATIONAL, to better reflect what is actually occurring.
How can I verify that IOS SLB sticky connections are working properly?	Use the following procedure: 1. Configure the sticky connections. 2. Start a client connection. 3. Enter the **show ip slb reals detail** and **show ip slb conns** commands. 4. Examine the real server connection counts. The real server whose count increased is the one to which the client connection is assigned. 5. Enter the **show ip slb sticky** command to display the sticky relationships stored by IOS SLB. 6. End the connection. 7. Ensure that the real server's connection count decreased. 8. Restart the connection, after waiting no longer than the sticky timeout value. 9. Enter the **show ip slb conns** command again. 10. Examine the real server connection counts again, and verify that the sticky connection is assigned to the same real server as before.

Question	Answer
How can I verify that server failures are being detected correctly?	Use the following procedure: 1. Use a large client population. If the number of clients is very small, tune the **numclients** keyword on the **faildetect numconns (real server)** command so that the servers are not displayed as FAILED. 2. Enter the **show ip slb reals detail** command to show the status of the real servers. 3. Examine the status and connection counts of the real servers. – Servers that failed show a status of FAILED, TESTING, or READY_TO_TEST, based on whether IOS SLB is checking that the server came back up when the command was sent. – When a real server fails, connections that are assigned but not established (no SYN or ACK is received) are reassigned to another real server on the first inbound SYN after the **reassign** threshold is met. However, any connections that were already established are forwarded to the same real server because, while it might not be accepting new connections, it might be servicing existing ones. – For weighted least connections, a real server that has just been placed in service starts slowly so that it is not overloaded with new connections. (See the "Slow Start" section on page 21 for more information.) Therefore, the connection counts displayed for a new real server show connections going to other real servers (despite the new real server's lower count). The connection counts also show "dummy connections" to the new real server, which IOS SLB uses to artificially inflate the connection counts for the real server during the slow start period.
Does the **no inservice** command take a resource out of service immediately?	When you use the **no** form of the **inservice** command to remove a firewall, firewall farm, real server, or virtual server from service, the resource acquiesces gracefully. No new connections are assigned, and existing connections are allowed to complete. To stop all existing connections for an entire firewall farm or virtual server immediately, use the **clear ip slb connections** command.
I configured both IOS SLB and input ACLs on the same Catalyst 6500 Family Switch, and now I see TCAM Capacity Exceeded messages. Why?	If you configure IOS SLB and either input ACLs or firewall load balancing on the same Catalyst 6500 Family Switch, you can exceed the capacity of the TCAM on the Policy Feature Card (PFC). To correct the problem, use the **mls ip slb search wildcard rp** command to reduce the amount of TCAM space used by IOS SLB, but be aware that this command can result in a slight increase in route processor utilization.
Which IOS releases and platforms support IOS SLB VRF?	Virtual Private Network (VPN) routing and forwarding (VRF) for IOS SLB is supported in IOS release 12.2(18)SXE or later on the Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for the Cisco 7600 series routers.

Question	Answer
What can cause IOS SLB out-of-sync messages on the Supervisor?	If you are using a single Supervisor with **replicate slave** configured, you might receive out-of-sync messages on the Supervisor.
Can IOS SLB provide both firewall load balancing and RADIUS load balancing on the same Supervisor?	IOS SLB can provide both firewall load balancing and RADIUS load balancing on the same Supervisor Engine 720 (SUP720-MSFC3).

Related Documents

Related Topic	Document Title
Cisco IOS configuration fundamentals	*Cisco IOS Configuration Fundamentals Configuration Guide*
Cisco IOS IP configuration information	*Cisco IOS IP Addressing Configuration Guide* *Cisco IOS IP Addressing Command Reference* *Cisco IOS IP Application Services Configuration Guide* *Cisco IOS IP Application Services Command Reference*
Cisco IOS mobile wireless configuration information	*Cisco IOS IP Mobility Configuration Guide* *Cisco IOS IP Mobility Command Reference*
DFP configuration information	*Dynamic Feedback Protocol Support in Distributed Director*
CFM configuration information	*Using Content Flow Monitor*

Supported Platforms

Switch or Router	Supported Platforms
Cisco 7600 Series Routers	• Supervisor Engine 32 with an MSFC2A (SUP32-MSFC2A) • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) • Cisco Route Switch Processor 720 with Distributed Forwarding Card DFC3CXL with two Gigabit Ethernet ports (RSP720-3CXL-GE)

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
CISCO-SLB-MIB CISCO-SLB-CAPABILITY **Note** Although the objects in these MIBs are defined as *read-create*, you cannot use the SNMP SET command to modify them. Instead, you must use the command line to set the associated command line keywords, after which the new values are reflected in SNMP.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 1631	*The IP Network Address Translator (NAT)*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for IOS SLB

Table 2 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 12.2(1) or a later release appear in the table.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, refer to the *Cisco IOS IP Application Services Command Reference.*

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 2 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 2 Feature Information for IOS SLB

Feature Name	Releases	Feature Information
IOS SLB, First Release on 12.2	12.2(1)	The IOS SLB feature is an IOS-based solution that provides load balancing for a variety of networked devices and services.
Active Standby	12.2(1)	Active standby enables two IOS SLBs to load-balance the same virtual IP address while at the same time acting as backups for each other. The following sections provide information about this feature: • Active Standby, page 28 • Stateless Backup Configuration Task List, page 92 • Configuring IOS SLB with Active Standby: Example, page 138
Algorithms for Server Load Balancing	12.2(1)	IOS SLB provides the following load-balancing algorithms: • Weighted Round Robin, page 11 • Weighted Least Connections, page 12 • Route Map, page 12 The following sections provide information about this feature: • Algorithms for Server Load Balancing, page 11 • Configuring a Server Farm and a Real Server, page 36
Alternate IP Addresses	12.2(1)	IOS SLB enables you to telnet to the load-balancing device using an alternate IP address. The following section provides information about this feature: • Alternate IP Addresses, page 21
Audio and Video Load Balancing	12.2(1)	IOS SLB can balance RealAudio and RealVideo streams via Real-Time Streaming Protocol (RTSP), for servers running RealNetworks applications. The following section provides information about this feature: • Audio and Video Load Balancing, page 27

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Automatic Server Failure Detection	12.2(1)	IOS SLB automatically detects each failed TCP connection attempt to a real server, and increments a failure counter for that server. If a server's failure counter exceeds a configurable failure threshold, the server is considered out of service and is removed from the list of active real servers. The following sections provide information about this feature: • Automatic Server Failure Detection, page 22 • Disabling Automatic Server Failure Detection, page 99
Automatic Unfail	12.2(1)	When a real server fails and is removed from the list of active servers, it is assigned no new connections for a length of time specified by a configurable retry timer. After that timer expires, the server is again eligible for new virtual server connections and IOS SLB sends the server the next qualifying connection. If the connection is successful, the failed server is placed back on the list of active real servers. If the connection is unsuccessful, the server remains out of service and the retry timer is reset. The unsuccessful connection must have experienced at least one retry, otherwise the next qualifying connection would also be sent to that failed server. The following section provides information about this feature: • Automatic Unfail, page 22
Avoiding Attacks on Server Farms and Firewall Farms	12.2(1)	A highly secure site can take certain steps to protect its server farms and firewall farms from attacks. The following section provides information about this feature: • Avoiding Attacks on Server Farms and Firewall Farms, page 21
Bind ID Support	12.2(1)	The bind ID allows a single physical server to be bound to multiple virtual servers and report a different weight for each one. Thus, the single real server is represented as multiple instances of itself, each having a different bind ID. Dynamic Feedback Protocol (DFP) uses the bind ID to identify for which instance of the real server a given weight is specified. The bind ID is needed only if you are using DFP. The following sections provide information about this feature: • Bind ID Support, page 13 • Dynamic Feedback Protocol for IOS SLB, page 23 • Configuring a Server Farm and a Real Server, page 36
Client-Assigned Load Balancing	12.2(1)	Client-assigned load balancing allows you to limit access to a virtual server by specifying the list of client IP subnets that are permitted to use that virtual server. With this feature, you can assign a set of client IP subnets (such as internal subnets) connecting to a virtual IP address to one server farm or firewall farm, and assign another set of clients (such as external clients) to a different server farm or firewall farm. The following section provides information about this feature: • Client-Assigned Load Balancing, page 13

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Client NAT	12.2(1)	If you use more than one load-balancing device in your network, replacing the client IP address with an IP address associated with one of the devices results in proper routing of outbound flows to the correct device. Client NAT also requires that the ephemeral client port be modified since many clients can use the same ephemeral port. Even in cases where multiple load-balancing devices are not used, client NAT can be useful to ensure that packets from load-balanced connections are not routed around the device. The following sections provide information about this feature: • Client NAT, page 17 • Configuring NAT, page 91 • Configuring IOS SLB with NAT and Static NAT: Examples, page 122
Content Flow Monitor Support	12.2(1)	IOS SLB supports the Cisco Content Flow Monitor (CFM), a web-based status monitoring application within the CiscoWorks2000 product family. You can use CFM to manage Cisco server load-balancing devices. CFM runs on Windows NT and Solaris workstations, and is accessed using a web browser. The following section provides information about this feature: • Content Flow Monitor Support, page 13
Delayed Removal of TCP Connection Context	12.2(1)	Because of IP packet ordering anomalies, IOS SLB might “see” the termination of a TCP connection (a finish [FIN] or reset [RST]) followed by other packets for the connection. This problem usually occurs when there are multiple paths that the TCP connection packets can follow. To correctly redirect the packets that arrive after the connection is terminated, IOS SLB retains the TCP connection information, or context, for a specified length of time. The length of time the context is retained after the connection is terminated is controlled by a configurable delay timer. The following sections provide information about this feature: • Delayed Removal of TCP Connection Context, page 13 • Configuring a Server Farm and a Real Server, page 36
Dynamic Feedback Protocol for IOS SLB	12.2(1)	IOS SLB supports the DFP Agent Subsystem feature, also called global load balancing, which enables client subsystems other than IOS SLB to act as DFP agents. With the DFP Agent Subsystem, you can use multiple DFP agents from different client subsystems at the same time. The following sections provide information about this feature: • Dynamic Feedback Protocol for IOS SLB, page 23 • Configuring DFP, page 63 • Configuring IOS SLB with GPRS Load Balancing: Examples, page 148 • Configuring IOS SLB with KAL-AP Agent: Example, page 170

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Firewall Load Balancing	12.2(1)	As its name implies, firewall load balancing enables IOS SLB to balance flows to firewalls. Firewall load balancing uses a load-balancing device on each side of a group of firewalls (called a firewall farm) to ensure that the traffic for each flow travels to the same firewall, ensuring that the security policy is not compromised. The following sections provide information about this feature: • Firewall Load Balancing, page 13 • Configuring Firewall Load Balancing, page 47 • Configuring IOS SLB with Firewall Load Balancing: Examples, page 109
Maximum Connections	12.2(1)	IOS SLB allows you to configure maximum connections for server and firewall load balancing.The following sections provide information about this feature: • Maximum Connections, page 15 • Configuring a Server Farm and a Real Server, page 36 • Configuring a Firewall Farm, page 47 • Configuring a Complete IOS SLB Network: Example, page 108
Port-Bound Servers	12.2(1)	When you define a virtual server, you must specify the TCP or UDP port handled by that virtual server. However, if you configure NAT on the server farm, you can also configure port-bound servers. Port-bound servers allow one virtual server IP address to represent one set of real servers for one service, such as HTTP, and a different set of real servers for another service, such as Telnet. The following sections provide information about this feature: • Port-Bound Servers, page 19 • Configuring a Virtual Server, page 40
Probes: HTTP, Ping, and WSP Probes	12.2(1)	IOS SLB probes determine the status of each real server in a server farm and of each firewall in a firewall farm. The following sections provide information about this feature: • Probes, page 25 • Configuring a Probe, page 53 • Configuring IOS SLB with Probes: Examples, page 117
Protocol Support	12.2(1)	IOS SLB supports a fixed set of protocols. The following section provides information about this feature: • Protocol Support, page 26

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Server NAT	12.2(1)	Server NAT involves replacing the virtual server IP address with the real server IP address (and vice versa). The following sections provide information about this feature: • Server NAT, page 17 • Configuring NAT, page 91 • Configuring IOS SLB with NAT and Static NAT: Examples, page 122
Slow Start	12.2(1)	In an environment that uses weighted least connections load balancing, a real server that is placed in service initially has no connections, and could therefore be assigned so many new connections that it becomes overloaded. To prevent such an overload, slow start controls the number of new connections that are directed to a real server that has just been placed in service. The following section provides information about this feature: • Slow Start, page 21
Stateful Backup	12.2(1)	Stateful backup enables IOS SLB to incrementally backup its load-balancing decisions, or "keep state," between primary and backup switches. The backup switch keeps its virtual servers in a dormant state until HSRP detects failover; then the backup (now primary) switch begins advertising virtual addresses and processing flows. The following sections provide information about this feature: • Stateful Backup, page 28 • Stateful Backup of Redundant Route Processors Configuration Task List, page 94 • Configuring IOS SLB with Stateful Backup: Example, page 135 • Configuring IOS SLB with Stateful Backup of Redundant Route Processors: Example, page 137
Stateless Backup	12.2(1)	Stateless backup provides high network availability by routing IP flows from hosts on Ethernet networks without relying on the availability of a single Layer 3 switch. Stateless backup is particularly useful for hosts that do not support a router discovery protocol (such as the Intermediate System-to-Intermediate System [IS-IS] Interdomain Routing Protocol [IDRP]) and do not have the functionality to shift to a new Layer 3 switch when their selected Layer 3 switch reloads or loses power. The following sections provide information about this feature: • Stateless Backup, page 28 • Stateless Backup Configuration Task List, page 92 • Configuring IOS SLB with Stateless Backup: Examples, page 126

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Sticky Connections	12.2(1)	Sometimes, a client transaction can require multiple consecutive connections, which means new connections from the same client IP address or subnet must be assigned to the same real server. You can use the optional **sticky** command to enable IOS SLB to force connections from the same client to the same load-balanced server within a server farm. For firewall load balancing, the connections between the same client-server pair are assigned to the same firewall. The following sections provide information about this feature: • Sticky Connections, page 19 • Configuring a Server Farm and a Real Server, page 36 • Configuring a Virtual Server, page 40 • Configuring a Firewall Farm, page 47 • Configuring IOS SLB with Sticky Connections: Example, page 168
Supported Platforms	12.2(1)	IOS SLB for 12.2(1) included support for only the following platform: • Cisco 7200 series routers
SynGuard	12.2(1)	SynGuard limits the rate of TCP start-of-connection packets (SYNchronize sequence numbers, or SYNs) handled by a virtual server to prevent a type of network problem known as a SYN flood denial-of-service attack. A user might send a large number of SYNs to a server, which could overwhelm or crash the server, denying service to other users. SynGuard prevents such an attack from bringing down IOS SLB or a real server. SynGuard monitors the number of SYNs handled by a virtual server at specific intervals and does not allow the number to exceed a configured SYN threshold. If the threshold is reached, any new SYNs are dropped. The following sections provide information about this feature: • SynGuard, page 21 • Configuring a Virtual Server, page 40 • Configuring a Complete IOS SLB Network: Example, page 108
TCP Session Reassignment	12.2(1)	IOS SLB tracks each TCP SYN sent to a real server by a client attempting to open a new connection. If several consecutive SYNs are not answered, or if a SYN is replied to with an RST, the TCP session is reassigned to a new real server. The number of SYN attempts is controlled by a configurable reassign threshold. The following sections provide information about this feature: • TCP Session Reassignment, page 20 • Configuring a Server Farm and a Real Server, page 36 • GPRS Load Balancing Configuration Task List, page 64

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Transparent Web Cache Load Balancing	12.2(1)	IOS SLB can load-balance HTTP flows across a cluster of transparent web caches. To set up this function, configure the subnet IP addresses served by the transparent web caches, or some common subset of them, as virtual servers. Virtual servers used for transparent web cache load balancing do not answer pings on behalf of the subnet IP addresses, and they do not affect traceroute. The following sections provide information about this feature: • Transparent Web Cache Load Balancing, page 20 • Configuring a Virtual Server, page 40 • Configuring IOS SLB with Transparent Web Cache Load Balancing: Example, page 169
WAP Load Balancing	12.2(1)	You can use IOS SLB to load-balance Wireless Session Protocol (WSP) sessions among a group of WAP gateways or servers on an IP bearer network. The following sections provide information about this feature: • WAP Load Balancing, page 34 • Configuring a Virtual Server, page 40 • Configuring a WSP Probe, page 60 • Configuring IOS SLB with WAP and UDP Load Balancing: Example, page 143
AAA Load Balancing	12.2(14)S	IOS SLB provides RADIUS load-balancing capabilities for RADIUS authentication, authorization, and accounting (AAA) servers. The following section provides information about this feature: • AAA Load Balancing, page 27
Backup Server Farms	12.2(14)S	A backup server farm is a server farm that can be used when none of the real servers defined in a primary server farm is available to accept new connections. The following sections provide information about this feature: • Backup Server Farms, page 22 • Configuring a Virtual Server, page 40

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
DFP Agent Subsystem Support	12.2(14)S	IOS SLB supports the DFP Agent Subsystem feature, also called global load balancing, which enables client subsystems other than IOS SLB to act as DFP agents. With the DFP Agent Subsystem, you can use multiple DFP agents from different client subsystems at the same time. The following sections provide information about this feature: • DFP Agent Subsystem Support, page 23 • Configuring DFP, page 63 • Configuring IOS SLB with GPRS Load Balancing Without GTP Cause Code Inspection: Example, page 148 • Configuring IOS SLB with GPRS Load Balancing and NAT: Example, page 153 • Configuring IOS SLB with KAL-AP Agent: Example, page 170
GPRS Load Balancing: Support for GPRS Tunneling Protocol (GTP) v0	12.2(14)S	IOS SLB supports both GTP Version 0 (GTP v0) and GTP Version 1 (GTP v1). Support for GTP enables IOS SLB to become "GTP aware," extending IOS SLB's knowledge into Layer 5. The following sections provide information about this feature: • GPRS Load Balancing, page 29 • Configuring a Virtual Server, page 40 • GPRS Load Balancing Configuration Task List, page 64 • Configuring IOS SLB with GPRS Load Balancing: Examples, page 148
Multiple Firewall Farm Support	12.2(14)S	You can configure more than one firewall farm in each load-balancing device. The following sections provide information about this feature: • Multiple Firewall Farm Support, page 15 • Configuring Firewall Load Balancing, page 47 • Configuring IOS SLB with Multiple Firewall Farms: Example, page 113
Probes: DNS, Routed, and TCP Probes	12.2(14)S	IOS SLB probes determine the status of each real server in a server farm and of each firewall in a firewall farm. The following sections provide information about this feature: • Probes, page 25 • Configuring a Probe, page 53 • Configuring IOS SLB with Probes: Examples, page 117

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
RADIUS Load Balancing: CDMA2000	12.2(14)S	IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). IOS SLB supports RADIUS load balancing for the following CDMA2000 mobile wireless networks: • Simple IP CDMA2000 networks. CDMA2000 is a third-generation (3-G) version of Code Division Multiple Access (CDMA). In a simple IP CDMA2000 mobile wireless network, the RADIUS client is a Packet Data Service Node (PDSN). • Mobile IP CDMA2000 networks. In a Mobile IP CDMA2000 mobile wireless network, both the Home Agent (HA) and the PDSN/Foreign Agent (PDSN/FA) are RADIUS clients. The following sections provide information about this feature: • RADIUS Load Balancing, page 32 • RADIUS Load Balancing Configuration Task List, page 71 • Configuring IOS SLB with RADIUS Load Balancing for a Simple IP CDMA2000 Network: Example, page 161 • Configuring IOS SLB with RADIUS Load Balancing for a Mobile IP CDMA2000 Network: Example, page 162
RADIUS Load Balancing: General packet radio service (GPRS) networks	12.2(14)S	IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). IOS SLB supports RADIUS load balancing for GPRS networks. In a GPRS mobile wireless network, the RADIUS client is typically a GGSN. The following sections provide information about this feature: • RADIUS Load Balancing, page 32 • RADIUS Load Balancing Configuration Task List, page 71 • Configuring IOS SLB with RADIUS Load Balancing for a GPRS Network: Example, page 159
RADIUS Load Balancing: Multiple Service Gateway Server Farms	12.2(14)S	IOS SLB provides RADIUS load balancing in mobile wireless networks that use service gateways, such as the Cisco Service Selection Gateway (SSG) or the Cisco Content Services Gateway (CSG). IOS SLB supports RADIUS load balancing for multiple service gateway server farms (for example, one farm of SSGs and another of CSGs). The following sections provide information about this feature: • RADIUS Load Balancing, page 32 • RADIUS Load Balancing Configuration Task List, page 71 • Configuring IOS SLB with RADIUS Load Balancing for Multiple Service Gateway Farms: Example, page 163

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Route Health Injection	12.2(14)S	By default, a virtual server's IP address is advertised (added to the routing table) when you bring the virtual server into service (using the **inservice** command). If you have a preferred host route to a website's virtual IP address, you can advertise that host route, but you have no guarantee that the IP address is available. However, you can use the **advertise** command to configure IOS SLB to advertise the host route only when IOS SLB has verified that the IP address is available. IOS SLB withdraws the advertisement when the IP address is no longer available. This function is known as route health injection. The following sections provide information about this feature: • Route Health Injection, page 19 • Configuring a Virtual Server, page 40 • Configuring IOS SLB with Route Health Injection: Examples, page 145
Static NAT	12.2(14)S	With static NAT, address translations exist in the NAT translation table as soon as you configure static NAT commands, and they remain in the translation table until you delete the static NAT commands. The following sections provide information about this feature: • Static NAT, page 17 • Configuring NAT, page 91 • Configuring IOS SLB with Static NAT: Example, page 125
VPN Server Load Balancing	12.2(14)S	IOS SLB can balance Virtual Private Network (VPN) flows. The following sections provide information about this feature: • VPN Server Load Balancing, page 27 • VPN Server Load Balancing Configuration Task List, page 88 • Configuring IOS SLB with VPN Server Load Balancing: Example, page 158
GPRS Load Balancing: Support for GTP v0 and GTP v1	12.2(14)ZA2	IOS SLB supports both GTP Version 0 (GTP v0) and GTP Version 1 (GTP v1). Support for GTP enables IOS SLB to become "GTP aware," extending IOS SLB's knowledge into Layer 5. The following sections provide information about this feature: • GPRS Load Balancing, page 29 • Configuring a Virtual Server, page 40 • GPRS Load Balancing Configuration Task List, page 64 • Configuring IOS SLB with GPRS Load Balancing: Examples, page 148

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
GPRS Load Balancing with GTP Cause Code Inspection	12.2(14)ZA2	GPRS load balancing *with* GTP cause code inspection enabled allows IOS SLB to monitor all PDP context signaling flows to and from GGSN server farms. This enables IOS SLB to monitor GTP failure cause codes, detecting system-level problems in both Cisco and non-Cisco GGSNs. The following sections provide information about this feature: • GPRS Load Balancing with GTP Cause Code Inspection, page 30 • GPRS Load Balancing Configuration Task List, page 64 • Configuring IOS SLB with GPRS Load Balancing, NAT, and GTP Cause Code Inspection: Example, page 156
Home Agent Director	12.2(14)ZA2	The Home Agent Director load balances Mobile IP Registration Requests (RRQs) among a set of home agents (configured as real servers in a server farm). Home agents are the anchoring points for mobile nodes. Home agents route flows for a mobile node to its current foreign agent (point of attachment). The following sections provide information about this feature: • Home Agent Director, page 31 • Home Agent Director Configuration Task List, page 89 • Configuring IOS SLB with Home Agent Director: Example, page 168
Probes: Custom UDP Probes	12.2(14)ZA2	IOS SLB probes determine the status of each real server in a server farm and of each firewall in a firewall farm. The following sections provide information about this feature: • Probes, page 25 • Configuring a Probe, page 53 • Configuring IOS SLB with Probes: Examples, page 117
Supported Platforms	12.2(14)ZA2	IOS SLB for 12.2(14)ZA2 included support for only the following platforms: • Cisco 7100 series routers • Cisco 7200 series routers • Supervisor Engine 1 with an MSFC2 for Cisco Catalyst 6500 family switches • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for Cisco Catalyst 6500 family switches • Supervisor Engine 1 with an MSFC2 for the Cisco 7600 series routers • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for the Cisco 7600 series routers

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Automatic Server Failure Detection: Disabling Automatic Server Failure Detection	12.2(14)ZA4	IOS SLB automatically detects each failed TCP connection attempt to a real server, and increments a failure counter for that server. If a server's failure counter exceeds a configurable failure threshold, the server is considered out of service and is removed from the list of active real servers. The following sections provide information about this feature: • Automatic Server Failure Detection, page 22 • Disabling Automatic Server Failure Detection, page 99
Exchange Director Features	12.2(14)ZA5	IOS SLB supports the Exchange Director for the mobile Service Exchange Framework (mSEF) for Cisco 7600 series routers. The following section provides information about this feature: • Exchange Director Features, page 28
Flow Persistence	12.2(14)ZA5	Flow persistence provides intelligent return routing of load-balanced IP flows to the appropriate node, without the need for coordinated hash mechanisms on both sides of the load-balanced data path, and without using Network Address Translation (NAT) or proxies to change client or server IP addresses. The following section provides information about this feature: • Flow Persistence, page 35
Stateful Backup of Redundant Route Processors	12.2(14)ZA5	When used with RPR+, IOS SLB supports the stateful backup of redundant route processors for mSEF for Cisco 7600 series routers. This enables you to deploy Cisco Multiprocessor WAN Application Modules (MWAMs) in the same chassis as IOS SLB, while maintaining high availability of load-balancing assignments. The following sections provide information about this feature: • Stateful Backup of Redundant Route Processors, page 35 • Stateful Backup of Redundant Route Processors Configuration Task List, page 94 • Configuring IOS SLB with Stateful Backup of Redundant Route Processors: Example, page 137
—	12.2(14)ZA6	This release incorporated only minor corrections and clarifications.
Supported Platforms	12.2(17d)SXB	IOS SLB for 12.2(17d)SXB included support for only the following platforms: • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for Cisco Catalyst 6500 family switches • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for the Cisco 7600 series routers

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
GGSN-IOS SLB Messaging	12.2(17d)SXB1	This feature enables a GGSN to notify IOS SLB when certain conditions occur. The notifications enable IOS SLB to make intelligent decisions, which in turn improves GPRS load balancing and failure detection. The following sections provide information about this feature: • GGSN-IOS SLB Messaging, page 24 • GGSN-IOS SLB Messaging Task List, page 66
DFP and the Home Agent Director	12.2(17d)SXD	For the Home Agent Director, you can define IOS SLB as a DFP manager and define a DFP agent on each home agent in the server farm, and the DFP agent can report the weights of the home agents. The DFP agents calculate the weight of each home agent based on CPU utilization, processor memory, and the maximum number of bindings that can be activated for each home agent. The following sections provide information about this feature: • Dynamic Feedback Protocol for IOS SLB, page 23 • DFP and the Home Agent Director, page 24 • Home Agent Director, page 31 • Configuring DFP, page 63 • Home Agent Director Configuration Task List, page 89 • Configuring IOS SLB with GPRS Load Balancing: Examples, page 148 • Configuring IOS SLB with Home Agent Director: Example, page 168 • Configuring IOS SLB with KAL-AP Agent: Example, page 170
Supported Platforms	12.2(17d)SXD	IOS SLB for 12.2(17d)SXD included support for only the following platforms: • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for Cisco Catalyst 6500 family switches • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for Cisco Catalyst 6500 family switches • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for the Cisco 7600 series routers • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for the Cisco 7600 series routers

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
GTP IMSI Sticky Database	12.2(17d)SXE	IOS SLB can select a gateway general packet radio service (GPRS) support node (GGSN) for a given International Mobile Subscriber ID (IMSI), and forward all subsequent Packet Data Protocol (PDP) create requests from the same IMSI to the selected GGSN. The following sections provide information about this feature: • GTP IMSI Sticky Database, page 14 • Configuring IOS SLB with GTP IMSI Sticky Database: Example, page 169
Interface Awareness	12.2(17d)SXE	Some environments require IOS SLB on both sides of a farm of CSGs, SSGs, or firewalls. For example, you might want IOS SLB to perform RADIUS load balancing on one side of a farm and firewall load balancing on the other, or firewall load balancing on both sides of a firewall farm. Such “sandwich” environments require IOS SLB to take into account the input interface when mapping packets to virtual servers, firewall farms, connections, and sessions. In IOS SLB, this function is called interface awareness. When interface awareness is configured, IOS SLB processes only traffic arriving on configured access interfaces. (An access interface is any Layer 3 interface.) The following sections provide information about this feature: • Interface Awareness, page 15 • Configuring IOS SLB with Dual Firewall Load Balancing “Sandwich”: Example, page 115 • Configuring IOS SLB with RADIUS Load Balancing/Firewall Load Balancing “Sandwich”: Example, page 163
RADIUS Load Balancing: RADIUS Load Balancing IMSI Sticky Database	12.2(17d)SXE	The IOS SLB RADIUS International Mobile Subscriber ID (IMSI) sticky database maps the IMSI address for each user to the corresponding gateway. This enables IOS SLB to forward all subsequent flows for the same user to the same gateway. The following sections provide information about this feature: • RADIUS Load Balancing, page 32 • RADIUS Load Balancing Configuration Task List, page 71
—	12.2(18)SXF	This release incorporated only minor corrections and clarifications.

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Supported Platforms	12.2(17d)SXF5	IOS SLB for 12.2(17d)SXF5 included support for only the following platforms: • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for Cisco Catalyst 6500 family switches • Supervisor Engine 32 with an MSFC2A (SUP32-MSFC2A) for Cisco Catalyst 6500 family switches • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for Cisco Catalyst 6500 family switches • Supervisor Engine 2 with an MSFC2 (SUP2-MSFC2) for the Cisco 7600 series routers • Supervisor Engine 32 with an MSFC2A (SUP32-MSFC2A) for the Cisco 7600 series routers • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for the Cisco 7600 series routers
—	12.2(18)SXF7	This release incorporated only minor corrections and clarifications.
Supported Platforms	12.2(33)SRB	IOS SLB for 12.2(33)SRB included support for only the following platforms: • Supervisor Engine 32 with an MSFC2A (SUP32-MSFC2A) for the Cisco 7600 series routers • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for the Cisco 7600 series routers
GPRS Load Balancing: GPRS Load Balancing Maps	12.2(33)SRB	GPRS load balancing maps enable IOS SLB to categorize and route user traffic based on access point names (APNs). The following sections provide information about this feature: • GPRS Load Balancing, page 29 • Configuring GPRS Load Balancing Maps, page 67 • Configuring IOS SLB with GPRS Load Balancing Maps: Example, page 157
RADIUS Load Balancing: RADIUS Load Balancing Maps	12.2(33)SRB	RADIUS load balancing maps enable IOS SLB to categorize and route user traffic based on RADIUS calling station IDs and user names. RADIUS load balancing maps is mutually exclusive with Turbo RADIUS load balancing and RADIUS load balancing accounting local acknowledgement. The following sections provide information about this feature: • RADIUS Load Balancing, page 32 • Configuring RADIUS Load Balancing Maps, page 75 • Configuring IOS SLB with RADIUS Load Balancing Maps: Example, page 165

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
Supported Platforms	12.2(33)SRC	IOS SLB for 12.2(33)SRC included support for only the following platforms: • Supervisor Engine 32 with an MSFC2A (SUP32-MSFC2A) for the Cisco 7600 series routers • Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for the Cisco 7600 series routers • Cisco Route Switch Processor 720 with Distributed Forwarding Card DFC3CXL with two Gigabit Ethernet ports (RSP720-3CXL-GE)
Connection Rate Limiting	12.2(33)SRC	IOS SLB enables you to specify the maximum connection rate allowed for a real server in a server farm. The following sections provide information about this feature: • Connection Rate Limiting, page 13 • Configuring a Server Farm and a Real Server, page 36
INOP_REAL State for Virtual Servers	12.2(33)SRC	You can configure a virtual server such that, if all of the real servers that are associated with the virtual server are inactive, the following actions occur: • The virtual server is placed in the INOP_REAL state. • An SNMP trap is generated for the virtual server's state transition. • The virtual server stops answering ICMP requests. The following sections provide information about this feature: • INOP_REAL State for Virtual Servers, page 24 • Configuring a Virtual Server, page 40
KeepAlive Application Protocol (KAL-AP) Agent Support	12.2(33)SRC	KAL-AP agent support enables IOS SLB to perform load balancing in a global server load balancing (GSLB) environment. KAL-AP provides load information along with its keepalive response message to the KAL-AP manager or GSLB device, such as the Global Site Selector (GSS), and helps the GSLB device load-balance client requests to the least-loaded IOS SLB devices. The following sections provide information about this feature: • KeepAlive Application Protocol (KAL-AP) Agent Support, page 31 • Configuring KeepAlive Application Protocol (KAL-AP) Agent Support, page 69 • Configuring IOS SLB with KAL-AP Agent: Example, page 170

Table 2 Feature Information for IOS SLB (continued)

Feature Name	Releases	Feature Information
RADIUS Load Balancing Accelerated Data Plane Forwarding	12.2(33)SRC	RADIUS load balancing accelerated data plane forwarding, also known as Turbo RADIUS load balancing, is a high-performance solution that uses basic policy-based routing (PBR) route maps to handle subscriber data-plane traffic in a CSG environment. When Turbo RADIUS load balancing receives a RADIUS payload, it inspects the payload, extracts the framed-IP attribute, applies a route map to the IP address, and then determines which CSG is to handle the subscriber. The following sections provide information about this feature: • RADIUS Load Balancing Accelerated Data Plane Forwarding, page 34 • Configuring RADIUS Load Balancing Accelerated Data Plane Forwarding, page 76 • Configuring IOS SLB with RADIUS Load Balancing Accelerated Data Plane Forwarding: Example, page 166
Access Service Network (ASN) R6 Load Balancing	12.2(33)SRC1	IOS SLB provides load balancing across a set of ASN gateways. The cluster of gateways appears to the base station as a single ASN gateway. The following sections provide information about this feature: • Restrictions for IOS SLB, page 3 • ASN R6 Load Balancing, page 29 • ASN R6 Load Balancing Configuration Task List, page 89 The following commands were modified by this feature: **debug ip slb, idle (virtual server), show ip slb sessions, show ip slb stats, show ip slb vservers, virtual**

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring Enhanced Object Tracking

First Published: May 2, 2005
Last Updated: July 11, 2008

Before the introduction of the Enhanced Object Tracking feature, the Hot Standby Router Protocol (HSRP) had a simple tracking mechanism that allowed you to track the interface line-protocol state only. If the line-protocol state of the interface went down, the HSRP priority of the router was reduced, allowing another HSRP router with a higher priority to become active.

The Enhanced Object Tracking feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.

A client process, such as HSRP, Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP), can now register its interest in tracking objects and then be notified when the tracked object changes state.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for Enhanced Object Tracking" section on page 31.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Information About Enhanced Object Tracking

Before you configure the Enhanced Object Tracking feature, you should understand the following concepts:

Feature Design of Enhanced Object Tracking

Enhanced Object Tracking provides complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes. Thus, several clients such as HSRP, VRRP, or GLPB can register their interest with the tracking process, track the same object, and each take different action when the object changes.

Each tracked object is identified by a unique number that is specified on the tracking command-line interface (CLI). Client processes use this number to track a specific object.

The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to interested client processes, either immediately or after a specified delay. The object values are reported as either up or down.

You can also configure a combination of tracked objects in a list and a flexible method for combining objects using Boolean logic. This functionality includes the following capabilities:

- Threshold—The tracked list can be configured to use a weight or percentage threshold to measure the state of the list. Each object in a tracked list can be assigned a threshold weight. The state of the tracked list is determined by whether or not the threshold has been met.
- Boolean “and” function—When a tracked list has been assigned a Boolean “and” function, it means that each object defined within a subset must be in an up state so that the tracked object can become up.
- Boolean “or” function—When the tracked list has been assigned a Boolean “or” function, it means that at least one object defined within a subset must be in an up state so that the tracked object can become up.

Enhanced Object Tracking and Embedded Event Manager

Beginning with Cisco IOS Release 12.4(2)T, Enhanced Object Tracking (EOT) is now integrated with Embedded Event Manager (EEM) to allow EEM to report on status change of a tracked object and to allow EOT to track EEM objects. A new type of tracking object—a stub object—is created. The stub object can be modified by an external process through a defined Application Programming Interface (API). See the “Embedded Event Manager Overview” document in the *Cisco IOS Network Management Configuration Guide* for more information on how EOT works with EEM.

EOT Support for Carrier Delay

The EOT Support for Carrier Delay feature enables Enhanced Object Tracking (EOT) to consider the carrier-delay timer when tracking the status of an interface.

If a link fails, by default there is a two-second timer that must expire before an interface and the associated routes are declared as being down. If a link goes down and comes back up before the carrier delay timer expires, the down state is effectively filtered, and the rest of the software on the switch is not aware that a link-down event occurred. You can configure the carrier-delay seconds command in interface configuration mode to extend the timer up to 60 seconds.

When EOT is configured on an interface, the tracking may detect the interface is down before a configured carrier-delay timer has expired. This is because EOT looks at the interface state and does not consider the carrier delay timer. Use the **carrier-delay** command in tracking configuration mode to enable tracking to consider the carrier-delay timer configured on an interface.

Enhanced Object Tracking for Mobile IP Applications

The Enhanced Object Tracking Support for Mobile IP feature enables EOT to monitor the presence of Home Agent, Packet Data Serving Node (PDSN), or Gateway GPRS Support Node (GGSN) traffic on a router for mobile wireless applications.

When a redundant pair of Home Agents running HSRP between them loses connectivity, both HSRP nodes become active. Once the connectivity is restored between the two nodes, a graceful way is needed to restore proper HSRP states without losing Home Agent bindings. During the time of no connectivity, one of the nodes will continue to process Home Agent, GGSN, or PDSN traffic while the other will not. The node that continues to process traffic needs to remain active once connectivity is restored. To ensure that the active node remains in the active state, the priority of the HSRP group member that does not process Home Agent traffic is reduced. Reducing the priority of the node that is not processing Home Agent traffic ensures that this node will become the standby after connectivity is restored. When connectivity is restored, the normal Home Agent state synchronization will get all bindings back into the inactive node and, depending on the preempt configuration, it may switch over again. This state synchronization ensures that no Mobile IP, GGSN, or PDSN bindings are lost.

For more information on configuring Mobile IP services, see the following Cisco IOS configuration guides:

Cisco IOS Mobile Wireless Home Agent Configuration Guide

Cisco IOS Mobile Wireless Gateway GPRS Support Node Configuration Guide

Cisco IOS Mobile Wireless Packet Data Serving Node Configuration Guide

Cisco IOS IP Mobility Configuration Guide

Benefits of Enhanced Object Tracking

- Increases the availability and speed of recovery of a network.
- Decreases network outages and their duration.
- Provides a scalable solution that allows other client processes such as VRRP and GLBP the ability to track objects individually or as a list of objects. Prior to the introduction of this functionality, the tracking process was embedded within HSRP.

How to Configure Enhanced Object Tracking

The following sections describe configuration tasks for Enhanced Object Tracking:

- Tracking the Line-Protocol State of an Interface, page 4 (optional)
- Tracking the IP-Routing State of an Interface, page 6 (optional)
- Tracking IP-Route Reachability, page 8 (optional)
- Tracking the Threshold of IP-Route Metrics, page 9 (optional)
- Tracking IP SLAs Operations, page 12 (optional)
- Configuring a Tracked List and Boolean Expression, page 16 (optional)
- Configuring a Tracked List and Threshold Weight, page 17 (optional)
- Configuring a Tracked List and Threshold Percentage, page 19 (optional)
- Configuring the Track List Defaults, page 20 (optional)
- Configuring Tracking for Mobile IP Applications, page 21

Tracking the Line-Protocol State of an Interface

Perform this task to track the line-protocol state of an interface.

Tracking the IP-routing state of an interface using the **track interface ip routing** command can be more useful in some situations than just tracking the line-protocol state using the **track interface line-protocol** command, especially on interfaces for which IP addresses are negotiated. See the "Tracking the IP-Routing State of an Interface" section for more information.

You can optionally configure EOT to consider the carrier-delay timer when tracking the line-protocol state of an interface by using the **carrier-delay** command in tracking configuration mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer interface** *seconds*
4. **track** *object-number* **interface** *type number* **line-protocol**
5. **carrier-delay**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **end**
8. **show track** *object-number*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**track timer interface** *seconds* **Example:** `Router(config)# track timer interface 5`	(Optional) Specifies the interval in which the tracking process polls the tracked object. - The default interval that the tracking process polls interface objects is 1 second.

Step 4	**track** *object-number* **interface** *type number* **line-protocol**
	
	Example:
	`Router(config)# track 3 interface ethernet 0/1`
	`line-protocol`
		Tracks the line-protocol state of an interface and enters tracking configuration mode.

Step 5	**`carrier-delay`**
	
	Example:
	`Router(config-track)# carrier-delay`
		(Optional) Enables EOT to consider the carrier-delay timer when tracking the status of an interface.

Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	
	Example:
	`Router(config-track)# delay up 30`
		(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

Step 7	**`end`**
	
	Example:
	`Router(config-track)# end`
		Exits to privileged EXEC mode.

Step 8	**show track** *object-number* **Example:** `Router# show track 3`	(Optional) Displays tracking information. - Use this command to verify the configuration. See the display output in the "Examples" section.

Examples

The following example shows the state of the line protocol on an interface when it is tracked:

```
Router# show track 3

Track 3
   Interface Ethernet0/1 line-protocol
   Line protocol is Up
     1 change, last change 00:00:05
```

```
Tracked by:
  HSRP Ethernet0/3 1
```

Tracking the IP-Routing State of an Interface

Perform this task to track the IP-routing state of an interface. An IP-routing object is considered up when the following criteria exist:

- IP routing is enabled and active on the interface.
- The interface line-protocol state is up.
- The interface IP address is known. The IP address is configured or received through the Dynamic Host Configuration Protocol (DHCP) or IP Control Protocol (IPCP) negotiation.

Interface IP routing will go down when one of the following criteria exist:

- IP routing is disabled globally.
- The interface line-protocol state is down.
- The interface IP address is unknown. The IP address is not configured or received through DHCP or IPCP negotiation.

Tracking the IP-routing state of an interface using the **track interface ip routing** command can be more useful in some situations than just tracking the line-protocol state using the **track interface line-protocol** command, especially on interfaces for which IP addresses are negotiated. For example, on a serial interface that uses the Point-to-Point Protocol (PPP), the line protocol could be up (link control protocol [LCP] negotiated successfully), but IP could be down (IPCP negotiation failed).

The **track interface ip routing** command supports the tracking of an interface with an IP address acquired through any of the following methods:

- Conventional IP address configuration
- PPP/IPCP
- DHCP
- Unnumbered interface

You can optionally configure EOT to consider the carrier-delay timer when tracking the IP-routing state of an interface by using the **carrier-delay** command in tracking configuration mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer interface** *seconds*
4. **track** *object-number* **interface** *type number* **ip routing**
5. **carrier-delay**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **end**
8. **show track** *object-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**track timer interface** *seconds* **Example:** `Router(config)# track timer interface 5`	(Optional) Specifies the interval in which the tracking process polls the tracked object. • The default interval that the tracking process polls interface objects is 1 second.
Step 4	**track** *object-number* **interface** *type number* **ip routing** **Example:** `Router(config)# track 1 interface ethernet 0/1 ip routing`	Tracks the IP-routing state of an interface and enters tracking configuration mode. • IP-route tracking tracks an IP route in the routing table and the ability of an interface to route IP packets.
Step 5	**carrier-delay** **Example:** `Router(config-track)# carrier-delay`	(Optional) Enables EOT to consider the carrier-delay timer when tracking the status of an interface.

Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	Example:
	`Router(config-track)# delay up 30`
		(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

Step 7	**end**
	Example:
	`Router(config-track)# end`
		Returns to privileged EXEC mode.

Step 8	**show track** *object-number* **Example:** `Router# show track 1`	Displays tracking information. • Use this command to verify the configuration. See the display output in the "Examples" section.

Examples

The following example shows the state of IP routing on an interface when it is tracked:

```
Router# show track 1

Track 1
   Interface Ethernet0/1 ip routing
   IP routing is Up
     1 change, last change 00:01:08
```

```
Tracked by:
  HSRP Ethernet0/3 1
```

Tracking IP-Route Reachability

Perform this task to track the reachability of an IP route. A tracked object is considered up when a routing table entry exists for the route and the route is accessible.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer ip route** *seconds*
4. **track** *object-number* **ip route** *ip-address/prefix-length* **reachability**
5. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
6. **ip vrf** *vrf-name*
7. **end**
8. **show track** *object-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**track timer ip route** *seconds* **Example:** `Router(config)# track timer ip route 20`	(Optional) Specifies the interval in which the tracking process polls the tracked object. • The default interval that the tracking process polls IP-route objects is 15 seconds.
Step 4	**track** *object-number* **ip route** *ip-address***/***prefix-length* **reachability** **Example:** `Router(config)# track 4 ip route 10.16.0.0/16 reachability`	Tracks the reachability of an IP route and enters tracking configuration mode.

	Command or Action	Purpose

Step 5	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	Example:
	`Router(config-track)# delay up 30`
		(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

Step 6	**ip vrf** *vrf-name* **Example:** `Router(config-track)# ip vrf VRF2`	(Optional) Configures a VPN routing and forwarding (VRF) table.

Step 7	**end**
	Example:
	`Router(config-track)# end`
		Returns to privileged EXEC mode.

Step 8	**show track** *object-number* **Example:** `Router# show track 4`	(Optional) Displays tracking information. • Use this command to verify the configuration. See the display output in the "Examples" section.

Examples

The following example shows the state of the reachability of an IP route when it is tracked:

```
Router# show track 4

Track 4
   IP route 10.16.0.0 255.255.0.0 reachability
   Reachability is Up (RIP)
     1 change, last change 00:02:04
   First-hop interface is Ethernet0/1
   Tracked by:
     HSRP Ethernet0/3 1
```

Tracking the Threshold of IP-Route Metrics

Perform this task to track the threshold of IP route metrics.

Scaled Route Metrics

The **track ip route** command enables tracking of a route in the routing table. If a route exists in the table, the metric value is converted into a number. To provide a common interface to tracking clients, route metric values are normalized to the range from 0 to 255, where 0 is connected and 255 is inaccessible. Scaled metrics can be tracked by setting thresholds. Up and down state notification occurs when the thresholds are crossed. The resulting value is compared against threshold values to determine the tracking state as follows:

- State is up if the scaled metric for that route is less than or equal to the up threshold.
- State is down if the scaled metric for that route is greater than or equal to the down threshold.

Tracking uses a per-protocol configurable resolution value to convert the real metric to the scaled metric. Table 1 shows the default values used for the conversion. You can use the **track resolution** command to change the metric resolution default values.

Table 1 Metric Conversion

Route Type[1]	Metric Resolution
Static	10
Enhanced Interior Gateway Routing Protocol (EIGRP)	2560
Open Shortest Path First (OSPF)	1
Intermediate System-to-Intermediate System (IS-IS)	10

[1] RIP is scaled directly to the range from 0 to 255 because its maximum metric is less than 255.

For example, a change in 10 in an IS-IS metric results in a change of 1 in the scaled metric. The default resolutions are designed so that approximately one 2-Mbps link in the path will give a scaled metric of 255.

Scaling the very large metric ranges of EIGRP and IS-IS to a 0 to 255 range is a compromise. The default resolutions will cause the scaled metric to go above the maximum limit with a 2-Mbps link. However, this scaling allows a distinction between a route consisting of three Fast-Ethernet links and a route consisting of four Fast-Ethernet links.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track timer ip route** *seconds*
4. **track resolution ip route** {**eigrp** *resolution-value* | **isis** *resolution-value* | **ospf** *resolution-value* | **static** *resolution-value*}
5. **track** *object-number* **ip route** *ip-address/prefix-length* **metric threshold**
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **ip vrf** *vrf-name*
8. **threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
9. **end**
10. **show track** *object-number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**track timer ip route** *seconds* **Example:** `Router(config)# track timer ip route 20`	(Optional) Specifies the interval in which the tracking process polls the tracked object. - The default interval that the tracking process polls IP-route objects is 15 seconds.
Step 4	**track resolution ip route** {**eigrp** *resolution-value* | **isis** *resolution-value* | **ospf** *resolution-value* | **static** *resolution-value*} **Example:** `Router(config)# track resolution ip route eigrp 300`	(Optional) Specifies resolution parameters for a tracked object. - Use this command to change the default metric resolution values.
Step 5	**track** *object-number* **ip route** *ip-address/prefix-length* **metric threshold** **Example:** `Router(config)# track 6 ip route 10.16.0.0/16 metric threshold`	Tracks the scaled metric value of an IP route to determine if it is above or below a threshold. - The default down value is 255, which equates to an inaccessible route. - The default up value is 254.

Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	Example:
	`Router(config-track)# delay up 30`
		(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

Step 7	**ip vrf** *vrf-name* **Example:** `Router(config-track)# ip vrf VRF1`	(Optional) Configures a VRF table.

Step 8	**threshold metric** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
	`Router(config-track)# threshold metric up 254 down 255`
		(Optional) Sets a metric threshold other than the default value.

	Command or Action	Purpose

Step 9	**end**
	Example:
	`Router(config-track)# end`
		Exits to privileged EXEC mode.

Step 10	**show track** *object-number* **Example:** `Router# show track 6`	(Optional) Displays tracking information. • Use this command to verify the configuration. See the display output in the "Examples" section.

Examples

The following example shows the metric threshold of an IP route when it is tracked:

```
Router# show track 6

Track 6
   IP route 10.16.0.0 255.255.0.0 metric threshold
   Metric threshold is Up (RIP/6/102)
     1 change, last change 00:00:08
   Metric threshold down 255 up 254
   First-hop interface is Ethernet0/1
   Tracked by:
     HSRP Ethernet0/3 1
```

Tracking IP SLAs Operations

Perform the following tasks to track Cisco IOS IP Service Level Agreements (SLAs) operations:

- Tracking the State of an IP SLAs Operation, page 13
- Tracking the Reachability of an IP SLAs IP Host, page 14

Object tracking of IP SLAs operations allows tracking clients to track the output from IP SLAs objects and use the provided information to trigger an action.

Cisco IOS IP SLAs is a network performance measurement and diagnostics tool that uses active monitoring. Active monitoring is the generation of traffic in a reliable and predictable manner to measure network performance. Cisco IOS software uses IP SLAs to collect real-time metrics such as response time, network resource availability, application performance, jitter (interpacket delay variance), connect time, throughput, and packet loss.

These metrics can be used for troubleshooting, for proactive analysis before problems occur, and for designing network topologies.

Every IP SLAs operation maintains an operation return-code value. This return code is interpreted by the tracking process. The return code can return OK, OverThreshold, and several other return codes. Different operations can have different return-code values, so only values common to all operation types are used.

Two aspects of an IP SLAs operation can be tracked: state and reachability. The difference between these aspects relates to the acceptance of the OverThreshold return code. Table 2 shows the state and reachability aspects of IP SLAs operations that can be tracked.

Table 2 Comparison of State and Reachability Operations

Tracking	Return Code	Track State
State	OK (all other return codes)	Up Down
Reachability	OK or OverThreshold (all other return codes)	Up Down

Tracking the State of an IP SLAs Operation

Perform this task to track the state of an IP SLAs operation.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* **rtr** *operation-number* **state**
   or
   track *object-number* **ip sla** *operation-number* **state**
4. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
5. **end**
6. **show track** *object-number*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command or Action	Purpose

Step 3	**Cisco IOS Releases Prior to 12.4(20)T**
	track *object-number* **rtr** *operation-number* **state**
	Cisco IOS Release 12.4(20)T or Later Releases
	track *object-number* **ip sla** *operation-number* **state**
	Example: Cisco IOS Releases Prior to 12.4(20)T
	`Router(config)# track 2 rtr 4 state`
	Example: Cisco IOS Release 12.4(20)T or Later Releases
	`Router(config)# track 2 ip sla 4 state`
		Tracks the state of an IP SLAs object and enters tracking configuration mode.
		Note Effective with Cisco IOS Release 12.4(20)T the **track rtr** command was replaced by the **track ip sla** command.

Step 4	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	Example:
	`Router(config-track)# delay up 60 down 30`
		(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

Step 5	**end**
	Example:
	`Router(config-track)# end`
		Exits to privileged EXEC mode.

Step 6	**show track** *object-number* **Example:** `Router# show track 2`	(Optional) Displays tracking information. • Use this command to verify the configuration. See the display output in the "Examples" section of this task.

Examples

The following example shows the state of the IP SLAs tracking:

```
Router# show track 2

Track 2
   IP SLA 1 state
   State is Down
     1 change, last change 00:00:47
   Latest operation return code: over threshold
   Latest RTT (millisecs) 4
   Tracked by:
     HSRP Ethernet0/1 3
```

Tracking the Reachability of an IP SLAs IP Host

Perform this task to track the reachability of an IP host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **track** *object-number* **rtr** *operation-number* **reachability**

   or

   track *object-number* **ip sla** *operation-number* **reachability**
4. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
5. **end**
6. **show track** *object-number*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**Cisco IOS Releases Prior to 12.4(20)T**
	`track` *`object-number`* **`rtr`** *`operation-number`* **`reachability`**
	Cisco IOS Release 12.4(20)T or Later Releases
	`track` *`object-number`* **`ip sla`** *`operation-number`* **`reachability`**
	Example: Cisco IOS Releases Prior to 12.4(20)T
	`Router(config)# track 2 rtr 4 reachability`
	Example: Cisco IOS Release 12.4(20)T or Later Releases
	`Router(config)# track 2 ip sla 4 reachability`
		Tracks the reachability of an IP SLAs IP host and enters tracking configuration mode.
		Note Effective with Cisco IOS Release 12.4(20)T the **track rtr** command was replaced by the **track ip sla** command.

Step 4	**`delay`** `{`**`up`** *`seconds`* `[`**`down`** *`seconds`*`] | [`**`up`** *`seconds`*`]` **`down`** *`seconds`*`}`
	Example:
	`Router(config-track)# delay up 30 down 10`
		(Optional) Specifies a period of time (in seconds) to delay communicating state changes of a tracked object.

Step 5	**`end`**
	Example:
	`Router(config-track)# end`
		Exits to privileged EXEC mode.

Step 6	**`show track`** *`object-number`*
	Example:
	`Router# show track 3`
		(Optional) Displays tracking information.
		• Use this command to verify the configuration. See the display output in the "Examples" section of this task.

Examples

The following example shows whether the route is reachable:

```
Router# show track 3

Track 3
   IP SLA 1 reachability
   Reachability is Up
     1 change, last change 00:00:47
   Latest operation return code: over threshold
   Latest RTT (millisecs) 4
   Tracked by:
     HSRP Ethernet0/1 3
```

Configuring a Tracked List and Boolean Expression

Perform this task to configure a tracked list of objects and a Boolean expression to determine the state of the list. A tracked list contains one or more objects. The Boolean expression enables two types of calculations by using either "and" or "or" operators. For example, when tracking two interfaces using the "and" operator, up means that *both* interfaces are up, and down means that *either* interface is down.

You may also configure a tracked list state to be measured using a weight or percentage threshold. See "Configuring a Tracked List and Threshold Weight" section on page 17 and "Configuring a Tracked List and Threshold Percentage" section on page 19.

Note The "not" operator is specified for one or more objects and negates the state of the object.

Prerequisites

An object must exist before it can be added to a tracked list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number* **list boolean** {**and** | **or**}
4. **object** *object-number* [**not**]
5. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**track** *track-number* **list boolean** {**and** | **or**}
	Example:
	`Router(config-track)# track 100 list boolean and`
		Configures a tracked list object and enters tracking configuration mode. The keywords are as follows:
		• **boolean**—Specifies that the state of the tracked list is based on a Boolean calculation. The keywords are as follows:
		– **and**—Specifies that the list is up if all objects are up, or down if one or more objects are down. For example when tracking two interfaces, up means that both interfaces are up, and down means that either interface is down.
		– **or**—Specifies that the list is up if at least one object is up. For example, when tracking two interfaces, up means that either interface is up, and down means that both interfaces are down.

Step 4	**object** *object-number* [**not**] **Example:** `Router(config-track)# object 3 not`	Specifies the object to be tracked. The *object-number* argument has a valid range from 1 to 500. There is no default. The optional **not** keyword negates the state of the object. **Note** The example means that when object 3 is up, the tracked list detects object 3 as down.

Step 5	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	Example:
	`Router(config-track)# delay up 3`
		(Optional) Specifies a tracking delay in seconds between up and down states.

Step 6	**end**
	Example:
	`Router(config-track)# end`
		Returns to privileged EXEC mode.

Configuring a Tracked List and Threshold Weight

Perform this task to configure a list of tracked objects, to specify that weight be used as the threshold, and to configure a weight for each of its objects. A tracked list contains one or more objects. Using a threshold weight, the state of each object is determined by comparing the total weight of all objects that are up against a threshold weight for each object.

You can also configure a tracked list state to be measured using a Boolean calculation or threshold percentage. See the "Configuring a Tracked List and Boolean Expression" section on page 16 and the "Configuring a Tracked List and Threshold Percentage" section on page 19.

Prerequisites

An object must exist before it can be added to a tracked list.

Restrictions

You cannot use the Boolean "not" operator in a weight or percentage threshold list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number* **list threshold weight**
4. **object** *object-number* [**weight** *weight-number*]
5. **threshold weight** {**up** *number* **down** *number* | **up** *number* | **down** *number*}
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**track** *track-number* **list threshold weight** **Example:** `Router(config-track)# track 100 list threshold weight`	Configures a tracked list object and enters tracking configuration mode. The keywords are as follows: • **threshold**—Specifies that the state of the tracked list is based on a threshold. • **weight**—Specifies that the threshold is based on a specified weight.
Step 4	**object** *object-number* [**weight** *weight-number*] **Example:** `Router(config-track)# object 3 weight 30`	Specifies the object to be tracked. The *object-number* argument has a valid range from 1 to 500. There is no default. The optional **weight** keyword specifies a threshold weight for each object.

	Command or Action	Purpose

Step 5	`threshold weight {up number down number | up number | down number}`
	Example:
	`Router(config-track)# threshold weight up 30`
		Specifies the threshold weight. The keywords and arguments are as follows:
		• **up** *number*—Valid range is from 1 to 255.
		• **down** *number*—Range depends upon what you select for the **up** keyword. For example, if you configure 25 for up, you will see a range from 0 to 24 for down.

Step 6	`delay {up seconds [down seconds] | [up seconds] down seconds}`
	Example:
	`Router(config-track)# delay up 3`
		(Optional) Specifies a tracking delay in seconds between up and down states.

Step 7	`end`
	Example:
	`Router(config-track)# end`
		Returns to privileged EXEC mode.

Configuring a Tracked List and Threshold Percentage

Perform this task to configure a tracked list of objects, to specify that a percentage will be used as the threshold, and to specify a percentage for each object in the list. A tracked list contains one or more objects. Using the threshold percentage, the state of the list is determined by comparing the assigned percentage of each object to the list.

You may also configure a tracked list state to be measured using a Boolean calculation or threshold weight. See "Configuring a Tracked List and Boolean Expression" section on page 16 and "Configuring a Tracked List and Threshold Weight" section on page 17.

Prerequisites

An object must exist before it can be added to a tracked list.

Restrictions

You cannot use the Boolean "not" operator in a weight or percentage threshold list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number* **list threshold percentage**
4. **object** *object-number*
5. **threshold percentage** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
6. **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
7. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**track** *track-number* **list threshold percentage** **Example:** `Router(config-track)# track 100 list threshold percentage`	Configures a tracked list object and enters tracking configuration mode. The keywords are as follows: • **threshold**—Specifies that the state of the tracked list is based on a threshold. • **percentage**—Specifies that the threshold is based on a percentage.
Step 4	**object** *object-number* **Example:** `Router(config-track)# object 3`	Specifies the object to be tracked. The *object-number* argument has a valid range from 1 to 500. There is no default.

Step 5	**threshold percentage** {**up** *number* [**down** *number*] | **down** *number* [**up** *number*]}
	Example:
	`Router(config-track)# threshold percentage up 30`
		Specifies the threshold percentage. The keywords and arguments are as follows:
		• **up** *number*—Valid range is from 1 to 100.
		• **down** *number*—Range depends upon what you have selected for the **up** keyword. For example, if you specify 25 as up, a range from 26 to 100 is displayed for the **down** keyword.

Step 6	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}
	Example:
	`Router(config-track)# delay up 3`
		(Optional) Specifies a tracking delay in seconds between up and down states.

Step 7	**`end`**
	Example:
	`Router(config-track)# end`
		Returns to privileged EXEC mode.

Configuring the Track List Defaults

Perform this task to configure a default delay value for a tracked list, a default object, and default threshold parameters for a tracked list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number*
4. **default** {**delay** | **object** *object-number* | **threshold percentage**}
5. *end*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**track** *track-number* **Example:** `Router(config)# track 3`	Enters tracking configuration mode.

Step 4	**default** {**delay** | **object** *object-number* | **threshold percentage**}
	Example:
	`Router(config-track)# default delay`
		Specifies a default delay value for a tracked list, a default object, and default threshold parameters for a tracked list. The keywords and arguments are as follows:
		• **delay**—Reverts to the default delay.
		• **object** *object-number*—Specifies a default object for the track list. The valid range is from 1 to 500.
		• **threshold percentage**—Specifies a default threshold percentage.

Step 5	**end**
	Example:
	`Router(config-track)# end`
		Returns to privileged EXEC mode.

Configuring Tracking for Mobile IP Applications

Perform this task to configure a tracked list of Mobile IP application objects.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *track-number* **application home-agent**

4. **exit**
5. **track** *track-number* **application pdsn**
6. **exit**
7. **track** *track-number* **application ggsn**
8. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`track`** *`track-number`* **`application home-agent`**
	Example:
	`Router(config)# track 100 application home-agent`
		(Optional) Tracks the presence of Home Agent traffic on a router.

Step 4	**`exit`**
	Example:
	`Router(config-track)# exit`
		Returns to global configuration mode.

Step 5	**`track`** *`track-number`* **`application pdsn`**
	Example:
	`Router(config)# track 100 application pdsn`
		(Optional) Tracks the presence of PDSN traffic on a router.

Step 6	**`exit`**
	Example:
	`Router(config-track)# exit`
		Returns to global configuration mode.

Step 7	**`track`** *`track-number`* **`application ggsn`**
	Example:
	`Router(config)# track 100 application ggsn`
		(Optional) Tracks the presence of GGSN traffic on a router.

Step 8	**`end`**
	Example:
	`Router(config)# end`
		Returns to privileged EXEC mode.

Configuration Examples for Enhanced Object Tracking

This section provides the following configuration examples:

- Interface Line Protocol: Example, page 23
- Interface IP Routing: Example, page 23
- IP-Route Reachability: Example, page 24
- IP-Route Threshold Metric: Example, page 25
- IP SLAs IP Host Tracking in Releases Prior to Cisco IOS Release 12.4(20)T: Example, page 25
- IP SLAs IP Host Tracking in Cisco IOS Release 12.4(20)T or Later Releases: Example, page 26
- Boolean Expression for a Tracked List: Example, page 26
- Threshold Weight for a Tracked List: Example, page 27
- Threshold Percentage for a Tracked List: Example, page 27
- Mobile IP Application Tracking: Example, page 28

Interface Line Protocol: Example

The following example is very similar to the IP-routing example. Instead, the tracking process is configured to track the line-protocol state of serial interface 1/0. HSRP on Ethernet interface 0/0 then registers with the tracking process to be informed of any changes to the line-protocol state of serial interface 1/0. If the line protocol on serial interface 1/0 goes down, the priority of the HSRP group is reduced by 10.

Router A Configuration

```
track 100 interface serial1/0 line-protocol
!
interface Ethernet0/0
 ip address 10.1.0.21 255.255.0.0
 standby 1 preempt
 standby 1 ip 10.1.0.1
 standby 1 priority 110
 standby 1 track 100 decrement 10
```

Router B Configuration

```
track 100 interface serial1/0 line-protocol
!
 interface Ethernet0/0
 ip address 10.1.0.22 255.255.0.0
 standby 1 preempt
 standby 1 ip 10.1.0.1
 standby 1 priority 105
 standby 1 track 100 decrement 10
```

Interface IP Routing: Example

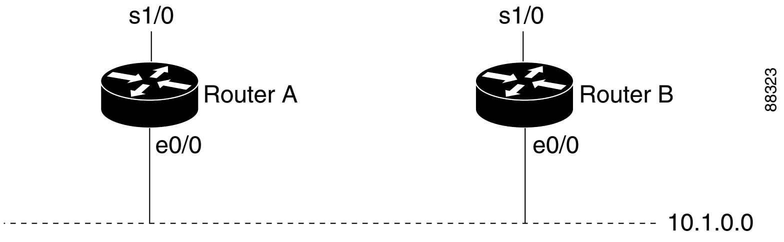
In the following example, the tracking process is configured to track the IP-routing capability of serial interface 1/0. HSRP on Ethernet interface 0/0 then registers with the tracking process to be informed of any changes to the IP-routing state of serial interface 1/0. If the IP-routing state on serial interface 1/0 goes down, the priority of the HSRP group is reduced by 10.

In the following example, EOT is configured to take the carrier-delay timer into consideration when tracking the state of serial interface 1/0.

If both serial interfaces are operational, Router A will be the HSRP active router because it has the higher priority. However, if IP on serial interface 1/0 in Router A fails, the HSRP group priority will be reduced and Router B will take over as the active router, thus maintaining a default virtual gateway service to hosts on the 10.1.0.0 subnet.

See Figure 1 for a sample topology.

Figure 1 Topology for IP-Routing Support



Router A Configuration

```
track 100 interface serial1/0 ip routing
 carrier-delay
!
interface Ethernet0/0
 ip address 10.1.0.21 255.255.0.0
 standby 1 preempt
 standby 1 ip 10.1.0.1
 standby 1 priority 110
 standby 1 track 100 decrement 10
```

Router B Configuration

```
track 100 interface serial1/0 ip routing
 carrier-delay
!
 interface Ethernet0/0
 ip address 10.1.0.22 255.255.0.0
 standby 1 preempt
 standby 1 ip 10.1.0.1
 standby 1 priority 105
 standby 1 track 100 decrement 10
```

IP-Route Reachability: Example

In the following example, the tracking process is configured to track the reachability of IP route 10.2.2.0/24:

Router A Configuration

```
track 100 ip route 10.2.2.0/24 reachability
!
interface Ethernet0/0
 ip address 10.1.1.21 255.255.255.0
 standby 1 preempt
 standby 1 ip 10.1.1.1
 standby 1 priority 110
 standby 1 track 100 decrement 10
```

Router B Configuration

```
track 100 ip route 10.2.2.0/24 reachability
!
interface Ethernet0/0
 ip address 10.1.1.22 255.255.255.0
 standby 1 preempt
 standby 1 ip 10.1.1.1
 standby 1 priority 105
 standby 1 track 100 decrement 10
```

IP-Route Threshold Metric: Example

In the following example, the tracking process is configured to track the threshold metric of IP route 10.2.2.0/24:

Router A Configuration

```
track 100 ip route 10.2.2.0/24 metric threshold
!
interface Ethernet0/0
 ip address 10.1.1.21 255.255.255.0
 standby 1 preempt
 standby 1 ip 10.1.1.1
 standby 1 priority 110
 standby 1 track 100 decrement 10
```

Router B Configuration

```
track 100 ip route 10.2.2.0/24 metric threshold
!
interface Ethernet0/0
 ip address 10.1.1.22 255.255.255.0
 standby 1 preempt
 standby 1 ip 10.1.1.1
 standby 1 priority 105
 standby 1 track 100 decrement 10
```

IP SLAs IP Host Tracking in Releases Prior to Cisco IOS Release 12.4(20)T: Example

The following example shows how to configure IP host tracking for IP SLAs operation 1 in Cisco IOS releases prior to Cisco IOS Release 12.4(20)T:

```
ip sla 1
 icmp-echo 10.51.12.4
 timeout 1000
 frequency 3
 threshold 2
 request-data-size 1400
 exit
ip sla schedule 1 start-time now life forever
 exit
track 2 rtr 1 state
track 3 rtr 1 reachability
 exit
interface ethernet0/1
```

```
 ip address 10.21.0.4 255.255.0.0
 no shutdown
 standby 3 ip 10.21.0.10d
 standby 3 priority 120
 standby 3 preempt
 standby 3 track 2 decrement 10
 standby 3 track 3 decrement 10
```

IP SLAs IP Host Tracking in Cisco IOS Release 12.4(20)T or Later Releases: Example

The following example shows how to configure IP host tracking for IP SLAs operation 1 in Cisco IOS Release 12.4(20)T and later releases:

```
ip sla 1
 icmp-echo 10.51.12.4
 timeout 1000
 frequency 3
 threshold 2
 request-data-size 1400
 exit
ip sla schedule 1 start-time now life forever
 exit
track 2 ip sla 1 state
track 3 ip sla 1 reachability
 exit
interface ethernet0/1
 ip address 10.21.0.4 255.255.0.0
 no shutdown
 standby 3 ip 10.21.0.10d
 standby 3 priority 120
 standby 3 preempt
 standby 3 track 2 decrement 10
 standby 3 track 3 decrement 10
```

Boolean Expression for a Tracked List: Example

In the following example, a track list object is configured to track two serial interfaces when both serial interfaces are up and when either serial interface is down:

```
track 1 interface serial2/0 line-protocol
track 2 interface serial2/1 line-protocol
 exit
track 100 list boolean and
 object 1
 object 2
```

In the following example, a track list object is configured to track two serial interfaces when either serial interface is up and when both serial interfaces are down:

```
track 1 interface serial2/0 line-protocol
track 2 interface serial2/1 line-protocol
 exit
track 101 list boolean or
 object 1
 object 2
```

The following configuration example shows that tracked list 4 has two objects and one object state is negated (if the list is up, the list detects that object 2 is down):

```
track 4 list boolean and
 object 1
 object 2 not
```

Threshold Weight for a Tracked List: Example

In the following example, three serial interfaces in tracked list 100 are configured with a threshold weight of 20 each. The down threshold is configured to 0 and the up threshold is configured to 40:

```
track 1 interface serial2/0 line-protocol
track 2 interface serial2/1 line-protocol
track 3 interface serial2/2 line-protocol
 exit
track 100 list threshold weight
 object 1 weight 20
 object 2 weight 20
 object 3 weight 20
 threshold weight down 0 up 40
```

The above example means that the track-list object goes down only when all three serial interfaces go down, and only comes up again when at least two serial interfaces are up (since 20+20 >= 40). The advantage of this configuration is that it prevents the track-list object from coming up if two interfaces are down and the third interface is flapping.

The following configuration example shows that if object 1 and object 2 are down, then track list 4 is up, because object 3 satisfies the up threshold value of up 30. But, if object 3 is down, both objects 1 and 2 need to be up in order to satisfy the threshold weight.

```
track 4 list threshold weight
 object 1 weight 15
 object 2 weight 20
 object 3 weight 30
 threshold weight up 30 down 10
```

This configuration may be useful to you if you have two small bandwidth connections (represented by object 1 and 2) and one large bandwidth connection (represented by object 3). Also the down 10 value means that once the tracked object is up, it will not go down until the threshold value is lower or equal to 10, which in this example means that all connections are down.

Threshold Percentage for a Tracked List: Example

In the following example, four serial interfaces in track list 100 are configured for an up threshold percentage of 75. The track list is up when 75 percent of the serial interfaces are up and down when fewer than 75 percent of the serial interfaces are up.

```
track 1 interface serial2/0 line-protocol
track 2 interface serial2/1 line-protocol
track 3 interface serial2/2 line-protocol
track 4 interface serial2/3 line-protocol
 exit
track 100 list threshold percentage
 object 1
 object 2
 object 3
 object 4
 threshold percentage up 75
```

Mobile IP Application Tracking: Example

The following example shows how to configure EOT to track Mobile IP, GGSN, and PDSN traffic on a router:

```
track 1 application home-agent
 exit
track 2 application ggsn
 exit
track 3 application pdsn
```

Additional References

The following sections provide references related to Enhanced Object Tracking.

Related Documents

Related Topic	Document Title
Embedded Event Manager	*Embedded Event Manager Overview*
HSRP concepts and configuration tasks	*Configuring HSRP*
GLBP concepts and configuration tasks	*Configuring GLBP*
VRRP concepts and configuration tasks	*Configuring VRRP*
GLBP, HSRP, and VRRP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference.*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for Enhanced Object Tracking

Table 3 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Releases 12.2(1) or 12.0(3)S or later appear in the table.

Not all commands may be available in your Cisco IOS software release. For details on when support for specific commands was introduced, see the command reference documents.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 3 Feature Information for Enhanced Object Tracking

Feature Name	Releases	Feature Configuration Information
Enhanced Tracking Support	12.2(15)T 12.2(25)S 12.2(28)SB 12.2(33)SRA 12.2(33)SXH Cisco IOS XE Release 2.1	The Enhanced Tracking Support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state. The following sections provide information about this feature: • Tracking the Line-Protocol State of an Interface, page 4 • Tracking the IP-Routing State of an Interface, page 6 • Tracking IP-Route Reachability, page 8 • Tracking the Threshold of IP-Route Metrics, page 9 The following commands were introduced or modified by this feature: **debug track**, **delay tracking**, **ip vrf**, **show track**, **standby track**, **threshold metric**, **track interface**, **track ip route**, **track timer**.
FHRP—Enhanced Object Tracking of IP SLAs Operations	12.3(4)T 12.2(25)S 12.2(27)SBC 12.2(33)SRA 12.2(33)SXH Cisco IOS XE Release 2.1 12.4(20)T	This feature enables First Hop Redundancy Protocols (FHRPs) and other Enhanced Object Tracking (EOT) clients to track the output from IP SLAs objects and use the provided information to trigger an action. The following section provides information about this feature: • Tracking IP SLAs Operations, page 12 The following command was introduced by this feature: **track rtr**. Effective with Cisco IOS Release 12.4(20)T, the **track rtr** command is replaced by the **track ip sla** command.

Table 3 Feature Information for Enhanced Object Tracking (continued)

Feature Name	Releases	Feature Configuration Information
FHRP—Object Tracking List	12.3(8)T 12.2(30)S 12.2(33)SRA 12.2(31)SB2 12.2(33)SXH Cisco IOS XE Release 2.1	This feature enhances the tracking capabilities to enable the configuration of a combination of tracked objects in a list, and a flexible method of combining objects using Boolean logic. The following sections provide information about this feature: • Configuring a Tracked List and Boolean Expression, page 16 • Configuring a Tracked List and Threshold Weight, page 17 • Configuring a Tracked List and Threshold Percentage, page 19 • Configuring the Track List Defaults, page 20 The following commands were introduced or modified by this feature: **show track**, **threshold percentage**, **threshold weight**, **track list**, **track resolution**.
FHRP—Enhanced Object Tracking Integration with Embedded Event Manager	12.4(2)T 12.2(33)SRB Cisco IOS XE Release 2.1	EOT is now integrated with EEM to allow EEM to report on a status change of a tracked object and to allow EOT to track EEM objects. The following section provides information about this feature: • Enhanced Object Tracking and Embedded Event Manager, page 2 The following commands were introduced or modified by this feature: **action track read**, **action track set**, **default-state**, **event resource**, **event rf**, **event track**, **show track**, **track stub**.

Table 3 Feature Information for Enhanced Object Tracking (continued)

Feature Name	Releases	Feature Configuration Information
FHRP—Enhanced Object Tracking Support for Mobile IP	12.4(11)T	The FHRP—Enhanced Object Tracking Support for Mobile IP feature provides new tracking objects needed by mobile wireless applications to track the presence of Home Agent, GGSN, or PDSN traffic on a router. The following sections provide information about this feature: • Enhanced Object Tracking for Mobile IP Applications, page 3 • Configuring Tracking for Mobile IP Applications, page 21 • Mobile IP Application Tracking: Example, page 28 The following command was introduced by this feature: **track application**.
EOT Support for Carrier Delay	12.4(9)T	The EOT Support for Carrier Delay feature enables Enhanced Object Tracking (EOT) to consider the carrier-delay timer when tracking the status of an interface. The following sections provide information about this feature: • EOT Support for Carrier Delay, page 3 • Tracking the Line-Protocol State of an Interface, page 4 • Tracking the IP-Routing State of an Interface, page 6 • Interface IP Routing: Example, page 23 The following commands were introduced or modified by this feature: **carrier-delay (tracking)**, **show track**.

Glossary

DHCP—Dynamic Host Configuration Protocol. DHCP is a protocol that delivers IP addresses and configuration information to network clients.

GLBP—Gateway Load Balancing Protocol. Provides automatic router backup for IP hosts that are configured with a single default gateway on an IEEE 802.3 LAN. Multiple first-hop routers on the LAN combine to offer a single virtual first-hop IP router while sharing the IP packet forwarding load. Other routers on the LAN may act as redundant (GLBP) routers that will become active if any of the existing forwarding routers fail.

GGSN—Gateway GPRS Support Node. A wireless gateway that allows mobile cell phone users to access the public data network (PDN) or specified private IP networks. The GGSN function is implemented on the Cisco routers.

GPRS—General Packet Radio Service. A 2.5G mobile communications technology that enables mobile wireless service providers to offer their mobile subscribers with packet-based data services over GSM networks.

GSM network—Global System for Mobile Communications network. A digital cellular technology that is used worldwide, predominantly in Europe and Asia. GSM is the world's leading standard in digital wireless communications.

Home Agent—A Home Agent is a router on the home network of the Mobile Node (MN) that maintains an association between the home IP address of the MN and its care-of address, which is the current location of the MN on a foreign or visited network. The HA redirects packets by tunneling them to the MN while it is away from the home network.

HSRP—Hot Standby Router Protocol. Provides high network availability and transparent network topology changes. HSRP creates a Hot Standby router group with a lead router that services all packets sent to the Hot Standby address. The lead router is monitored by other routers in the group, and if it fails, one of these standby routers inherits the lead position and the Hot Standby group address.

IPCP—IP Control Protocol. The protocol used to establish and configure IP over PPP.

LCP—Link Control Protocol. The protocol used to establish, configure, and test data-link connections for use by PPP.

PDSN—Packet Data Serving Node. The Cisco PDSN is a standards-compliant, wireless gateway that enables packet data services in a Code Division Multiplex Access (CDMA) environment. Acting as an access gateway, the Cisco PDSN provides simple IP and Mobile IP access, foreign-agent support, and packet transport for Virtual Private Networks (VPN).

PPP—Point-to-Point Protocol. Provides router-to-router and host-to-network connections over synchronous and asynchronous circuits. PPP is most commonly used for dial-up Internet access. Its features include address notification, authentication via CHAP or PAP, support for multiple protocols, and link monitoring.

VRF—VPN routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a provider edge router.

VRRP—Virtual Router Redundancy Protocol. Eliminates the single point of failure inherent in the static default routed environment. VRRP specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VRRP routers on a LAN. The VRRP router that controls the IP addresses associated with a virtual router is called the master, and forwards packets sent to these IP addresses. The election process provides dynamic failover in the forwarding responsibility should the master become unavailable. Any of the virtual router IP addresses on a LAN can then be used as the default first-hop router by end hosts.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring IP Services

First Published: October 23, 2006
Last Updated: May 5, 2008

This module describes how to configure optional IP services. For a complete description of the IP services commands in this chapter, refer to the *Cisco IOS IP Application Services Command Reference.* To locate documentation of other commands that appear in this module, use the command reference master index, or search online.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for IP Services" section on page 20.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Information About IP Services

To configure the IP services described in this module, you should understand the following concepts:

IP Source Routing

The Cisco IOS software examines IP header options on every packet. It supports the IP header options Strict Source Route, Loose Source Route, Record Route, and Time Stamp, which are defined in RFC 791. If the software finds a packet with one of these options enabled, it performs the appropriate action. If it finds a packet with an invalid option, it sends an Internet Control Message Protocol (ICMP) parameter problem message to the source of the packet and discards the packet.

IP provides a provision known as source routing that allows the source IP host to specify a route through the IP network. Source routing is specified as an option in the IP header. If source routing is specified, the software forwards the packet according to the specified source route. IP source routing is employed when you want to force a packet to take a certain route through the network. The default is to perform source routing. IP source routing is rarely used for legitimate purposes in networks. Some older IP implementations do not process source-routed packets properly, and it may be possible to crash devices running these implementations by sending them datagrams with source routing options. Disable IP source routing whenever possible. Disabling IP source routing will cause a Cisco router to never forward an IP packet that carries a source routing option.

ICMP Overview

Originally created for the TCP/IP suite in RFC 792, the Internet Control Message Protocol (ICMP) was designed to report a small set of error conditions. ICMP also can report a wide variety of error conditions and provide feedback and testing capabilities. Each message uses a common format and is sent and received by using the same protocol rules.

ICMP enables IP to perform addressing, datagram packaging, and routing by allowing encapsulated messages to be sent and received between IP devices. These messages are encapsulated in IP datagrams just like any other IP message. When the message is generated, the original IP header is encapsulated in the ICMP message and these two pieces are encapsulated within a new IP header to be returned as an error report to the sending device.

ICMP messages are sent in several situations: when a datagram cannot reach its destination, when the gateway does not have the buffering capacity to forward a datagram, and when the gateway can direct the host to send traffic on a shorter route. To avoid the infinite regress of messages about messages, no ICMP messages are sent about ICMP messages.

ICMP does not make IP reliable or ensure the delivery of datagrams or the return of a control message. Some datagrams may be dropped without any report of their loss. The higher-level protocols that use IP must implement their own reliability procedures if reliable communication is required.

For information about IPv6 and ICMP, refer to the "Implementing IPv6 Addressing and Basic Connectivity" document in the *Cisco IOS IPv6 Configuration Guide* at the following URL:

http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-addrg_bsc_con.html

ICMP Unreachable Error Messages

Type 3 error messages are sent when a message cannot be delivered completely to the application at a destination host. Six codes contained in the ICMP header describe the unreachable condition as follows:

- 0—Network unreachable
- 1—Host unreachable
- 2—Protocol unreachable
- 3—Port unreachable
- 4—Fragmentation needed and the "don't fragment" (DF) bit is set
- 5—Source route failed

Cisco IOS software can suppress the generation of ICMP unreachable destination error messages, which is called rate-limiting. The default is no unreachable messages more often than once every half second. Separate intervals can be configured for code 4 and all other unreachable destination error messages. However, there is no method of displaying how many ICMP messages have not been sent.

The ICMP Unreachable Destination Counters feature provides a method to count and display the unsent Type 3 messages. This feature also provides console logging with error messages when there are periods of excessive rate limiting that would indicate a Denial of Service (DoS) attack against the router.

If the Cisco IOS software receives a nonbroadcast packet destined for itself that uses an unknown protocol, it sends an ICMP protocol unreachable message back to the source. Similarly, if the software receives a packet that it is unable to deliver to the final destination because it knows of no route to the destination address, it sends an ICMP host unreachable message to the source. This functionality is enabled by default.

Disable Internet Message Control Protocol (ICMP) host unreachable messages whenever possible. ICMP supports IP traffic by relaying information about paths, routes, and network conditions. These messages can be used by an attacker to gain network mapping information.

Because the null interface is a packet sink, packets forwarded there will always be discarded and, unless disabled, will generate host unreachable messages. In that case, if the null interface is being used to block a Denial-of-Service attack, these messages flood the local network with these messages. Disabling these messages prevents this situation. In addition, because all blocked packets are forwarded to the null interface, an attacker receiving host unreachable messages could use those messages to determine Access Control List (ACL) configuration. If the "null 0" interface is configured on your router, disable ICMP host unreachable messages for discarded packets or packets routed to the null interface.

ICMP Mask Reply Messages

Occasionally, network devices must know the subnet mask for a particular subnetwork in the internetwork. To obtain this information, such devices can send ICMP mask request messages. ICMP mask reply messages are sent in reply from devices that have the requested information. The Cisco IOS software can respond to ICMP mask request messages if this function is enabled.

These messages can be used by an attacker to gain network mapping information.

ICMP Redirect Messages

Routes are sometimes less than optimal. For example, it is possible for the router to be forced to resend a packet through the same interface on which it was received. If the router resends a packet through the same interface on which it was received, the Cisco IOS software sends an ICMP redirect message to the originator of the packet telling the originator that the router is on a subnet directly connected to the receiving device, and that it must forward the packet to another system on the same subnet. The software sends an ICMP redirect message to the originator of the packet because the originating host presumably could have sent that packet to the next hop without involving this device at all. The redirect message instructs the sender to remove the receiving device from the route and substitute a specified device representing a more direct path. This functionality is enabled by default.

In a properly functioning IP network, a router will send redirects only to hosts on its own local subnets, no end node will ever send a redirect, and no redirect will ever be traversed more than one network hop. However, an attacker may violate these rules; some attacks are based on this. Disabling ICMP redirects will cause no operational impact to the network, and it eliminates this possible method of attack.

Denial of Service Attack

Denial of service has become a growing concern, especially when considering the associated costs of such an attack. DoS attacks can decrease the performance of networked devices, disconnect the devices from the network, and cause system crashes. When network services are unavailable, enterprises and service providers suffer the loss of productivity and sales.

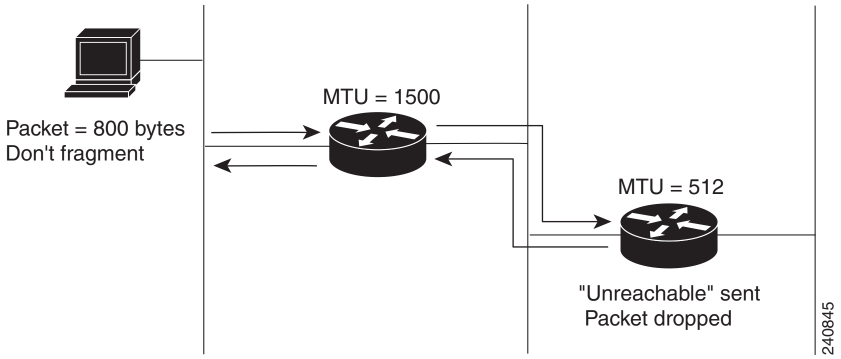
The objective of a DoS attack is to deprive a user or organization access to services or resources. If a Website is compromised by a DoS attack, millions of users could be denied access to the site. DoS attacks do not typically result in intrusion or the illegal theft of information. Instead of providing access to unauthorized users, DoS attacks can cause much aggravation and cost to the target customer by preventing authorized access. Distributed DoS (DDoS) attacks amplify DoS attacks in that a multitude of compromised systems coordinate to flood targets with attack packets, thereby causing denial of service for users of the targeted systems.

A DoS attack occurs when a stream of ICMP echo requests (pings) are broadcast to a destination subnet. The source addresses of these requests are falsified to be the source address of the target. For each request sent by the attacker, many hosts on the subnet will respond flooding the target and wasting bandwidth. The most common DoS attack is called a “smurf” attack, named after an executable program and is in the category of network-level attacks against hosts. DoS attacks can be easily detected when error-message logging of the ICMP Unreachable Destination Counters feature is enabled.

Path MTU Discovery

The Cisco IOS software supports the IP Path MTU Discovery mechanism, as defined in RFC 1191. IP Path MTU Discovery allows a host to dynamically discover and cope with differences in the maximum allowable maximum transmission unit (MTU) size of the various links along the path. Sometimes a router is unable to forward a datagram because it requires fragmentation (the packet is larger than the MTU you set for the interface with the **ip mtu** interface configuration command), but the “don’t fragment” (DF) bit is set. The Cisco IOS software sends a message to the sending host, alerting it to the problem. The host will need to fragment packets for the destination so that they fit the smallest packet size of all the links along the path. This technique is shown in Figure 1.

Figure 1 IP Path MTU Discovery



IP Path MTU Discovery is useful when a link in a network goes down, forcing the use of another, different MTU-sized link (and different routers). As shown in Figure 1, suppose a router is sending IP packets over a network where the MTU in the first router is set to 1500 bytes, but the second router is set to 512 bytes. If the “don’t fragment” bit of the datagram is set, the datagram would be dropped because the 512-byte router is unable to forward it. All packets larger than 512 bytes are dropped in this case. The second router returns an ICMP destination unreachable message to the source of the datagram with its Code field indicating “Fragmentation needed and DF set.” To support IP Path MTU Discovery, it would also include the MTU of the next hop network link in the low-order bits of an unused header field.

IP Path MTU Discovery is also useful when a connection is being established and the sender has no information at all about the intervening links. It is always advisable to use the largest MTU that the links will bear; the larger the MTU, the fewer packets the host must send.

Note IP Path MTU Discovery is a process initiated by end hosts. If an end host does not support IP Path MTU Discovery, the receiving device will have no mechanism available to avoid fragmenting datagrams generated by the end host.

If a router that is configured with a small MTU on an outbound interface receives packets from a host that is configured with a large MTU (for example, receiving packets from a Token Ring interface and forwarding them to an outbound Ethernet interface), the router fragments received packets that are larger than the MTU of the outbound interface. Fragmenting packets slows the performance of the router. To keep routers in your network from fragmenting received packets, run IP Path MTU Discovery on all hosts and routers in your network, and always configure the largest possible MTU for each router interface type.

IP MAC and Precedence Accounting

Cisco IP accounting support provides basic IP accounting functions. By enabling IP accounting, users can see the number of bytes and packets switched through the Cisco IOS software on a source and destination IP address basis. Only transit IP traffic is measured and only on an outbound basis; traffic generated by the software or terminating in the software is not included in the accounting statistics. To maintain accurate accounting totals, the software maintains two accounting databases: an active and a checkpointed database.

Cisco IP accounting support also provides information identifying IP traffic that fails IP access lists. Identifying IP source addresses that violate IP access lists alerts you to possible attempts to breach security. The data also indicates that you should verify IP access list configurations. To make this functionality available to users, you must enable IP accounting of access list violations using the **ip accounting access-violations** interface configuration command. Users can then display the number of bytes and packets from a single source that attempted to breach security against the access list for the source destination pair. By default, IP accounting displays the number of packets that have passed access lists and were routed.

The MAC address accounting functionality provides accounting information for IP traffic based on the source and destination MAC addresses on LAN interfaces. MAC accounting calculates the total packet and byte counts for a LAN interface that receives or sends IP packets to or from a unique MAC address. It also records a timestamp for the last packet received or sent. For example, with IP MAC accounting, you can determine how much traffic is being sent to and/or received from various peers at Network Access Profiles (NAPS)/peering points. IP MAC accounting is supported on Ethernet, Fast Ethernet, and FDDI interfaces and supports Cisco Express Forwarding (CEF), distributed CEF (dCEF), flow, and optimum switching.

The Precedence Accounting feature provides accounting information for IP traffic based on the precedence on any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching.

Show and Clear Commands for IOS Sockets

The Show and Clear Commands for IOS Sockets feature introduces the **show udp**, **show sockets**, and **clear sockets** commands. These new commands are useful for monitoring and managing the Cisco IOS Socket library.

In Cisco IOS software, sockets are a per process entity. This means that the maximum number of sockets is per process and all sockets are managed on a per process basis. For example, each Cisco IOS process could have a socket with file descriptor number 1. This is unlike UNIX or other operating systems that have per system file descriptor allocations.

The **show** and **clear** commands operate on a per process basis to be consistent with the current functionality. Thus, any action taken by the commands will be applicable only to a particular process at a time as selected by the process ID entered on the CLI.

Many applications have a need for **show** and **clear** commands, which primarily aid in debugging. The following scenarios provide examples of when these commands might be useful:

- The application H.323 is using sockets for voice calls. According to the current number of calls, there is still space for more sockets. However, no more sockets can be opened. You can now use the the **show sockets** command to find out if the socket space is indeed exhausted or if there are unused sockets available.

- An application is waiting for a particular socket event to happen. A UDP segment was seen, but the application never became active. You can use the **show udp** command to display the list of events being monitored to determine if a UDP socket event is being monitored or if the socket library failed to activate the application.
- An application wants to forcibly close all the sockets for a particular process. You can use the **clear sockets** command to close both the sockets and the underlying TCP or UDP connection or Stream Control Transmission Protocol (SCTP) association.

How to Configure IP Services

This section contains the following procedures:

- Protecting Your Network from DOS Attacks, page 7
- Configuring ICMP Unreachable Rate Limiting User Feedback, page 8
- Setting the MTU Packet Size, page 10
- Configuring IP Accounting, page 11
- Monitoring and Maintaining the IP Network, page 13

Protecting Your Network from DOS Attacks

ICMP supports IP traffic by relaying information about paths, routes, and network conditions. ICMP messages can be used by an attacker to gain network mapping information. IP source routing allows the source IP host to specify a route through the IP network and is rarely used for legitimate purposes in networks. Some older IP implementations do not process source-routed packets properly, and it may be possible to crash devices running these implementations by sending them datagrams with source routing options.

Whenever possible, ICMP messages and IP source routing should be disabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no ip source-route**
4. **interface** *type*/*number*
5. **no ip unreachables**
6. **no ip redirects**
7. **no ip mask-reply**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**no ip source-route** **Example:** `Router(config)# no ip source-route`	Disables IP source routing.
Step 4	**interface** *type***/***number* **Example:** `Router(config)# interface null 0`	Specifies the interface to configure and enters interface configuration mode.
Step 5	**no ip unreachables** **Example:** `Router(config-if)# no ip unreachables`	Disables the sending of ICMP protocol unreachable and host unreachable messages. This command is enabled by default. **Note** Disabling the unreachable messages also disables IP Path MTU Discovery because path discovery works by having the Cisco IOS software send unreachable messages.
Step 6	**no ip redirects** **Example:** `Router(config-if)# no ip redirects`	Disables the sending of ICMP redirect messages to learn routes. This command is enabled by default.
Step 7	**no ip mask-reply** **Example:** `Router(config-if)# no ip mask-reply`	Disables the sending of ICMP mask reply messages.

Configuring ICMP Unreachable Rate Limiting User Feedback

Perform this task to clear all of the unreachable destination packet statistics and to specify an interval number for unreachable destination messages. This task also configures a packet counter (threshold) and interval to trigger a logging message to a console. This task is beneficial to begin a new log after the thresholds have been set.

SUMMARY STEPS

1. **enable**
2. **clear ip icmp rate-limit** [*interface-type interface-number*]

3. **configure terminal**
4. **ip icmp rate-limit unreachable** [**df**] [*ms*] [**log** [*packets*] [*interval-ms*]]
5. **exit**
6. **show ip icmp rate-limit** [*interface-type interface-number*]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`clear ip icmp rate-limit`** [*`interface-type interface-number`*] **Example:** `Router# clear ip icmp rate-limit ethernet 2/3`	Clears all current ICMP unreachable statistics for all configured interfaces. The optional *interface-type* and *interface-number* arguments clear the statistics for only one interface.

Step 3	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 4	**`ip icmp rate-limit unreachable`** [**`df`**] [*`ms`*] [**`log`** [*`packets`*] [*`interval-ms`*]] **Example:** `Router(config)# ip icmp rate-limit unreachable df log 1100 12000`	Specifies the rate limitation of ICMP unreachable destination messages and the error message log threshold for generating a message. The default is no unreachable messages are sent more often than once every half second. The arguments and keywords are as follows: • **df**—(Optional) When “don’t fragment” (DF) bit is set in the ICMP header, a datagram cannot be fragmented. If the **df** keyword is not specified, all other types of destination unreachable messages are sent. • *ms*—(Optional) Interval at which unreachable messages are generated. The valid range is from 1 to 4294967295. • **log**—(Optional) List of error messages. The arguments are as follows: – *packets*—(Optional) Number of packets that determine a threshold for generating a log. The default is 1000. – *interval-ms*—(Optional) Time limit for an interval for which a logging message is triggered. The default is 60000, which is 1 minute. **Note** Counting begins as soon as this command is configured.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Router# exit`	Exits to privileged EXEC mode.
Step 6	**show ip icmp rate-limit** [*interface-type interface-number*] **Example:** `Router# show ip icmp rate-limit ethernet 2/3`	(Optional) Displays all current ICMP unreachable statistics for all configured interfaces. The optional *interface-type* and *interface-number* arguments display the statistics for only one interface.

Examples

The following output using the **show ip icmp rate-limit** command displays the unreachable destinations by interface:

```
Router# show ip icmp rate-limit

                           DF bit unreachables       All other unreachables
Interval (millisecond)     500                       500

Interface                  # DF bit unreachables     # All other unreachables
---------                  ---------------------     ------------------------
Ethernet0/0                0                         0
Ethernet0/2                0                         0
Serial3/0/3                0                         19

The greatest number of unreachables is on serial interface 3/0/3.
```

Setting the MTU Packet Size

All interfaces have a default MTU packet size. You can adjust the IP MTU size so that the Cisco IOS software will fragment any IP packet that exceeds the MTU set for an interface.

Changing the MTU value (with the **mtu** interface configuration command) can affect the IP MTU value. If the current IP MTU value is the same as the MTU value and you change the MTU value, the IP MTU value will be modified automatically to match the new MTU. However, the reverse is not true; changing the IP MTU value has no effect on the value for the **mtu** interface configuration command.

All devices on a physical medium must have the same protocol MTU in order to operate.

Perform this task to set the MTU packet size for a specified interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type/number*
4. **ip mtu** *bytes*
5. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type`***`/`***`number`*
	Example:
	`Router(config)# interface ethernet1/1`
		Specifies the interface to configure and enters interface configuration mode.

Step 4	**`ip mtu`** *`bytes`*
	Example:
	`Router(config-if)# ip mtu 300`
		Sets the IP MTU packet size for an interface.

Step 5	**`exit`**
	Example:
	`Router(config-if)# exit`
		Exits to privileged EXEC mode.

Configuring IP Accounting

To enable IP accounting, perform this task for each interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip accounting-threshold** *threshold*
4. **ip accounting-list** *ip-address wildcard*
5. **ip accounting-transits** *count*
6. **interface** *type/number*
7. **ip accounting** [**access-violations**] [**output-packets**]
8. **ip accounting mac-address** {**input** | **output**}
   or
   ip accounting precedence {**input** | **output**}

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip accounting-threshold** *threshold* **Example:** `Router(config)# ip accounting-threshold 500`	(Optional) Sets the maximum number of accounting entries to be created.
Step 4	**ip accounting-list** *ip-address wildcard* **Example:** `Router(config)# ip accounting-list 192.31.0.0 0.0.255.255`	(Optional) Filters accounting information for hosts.
Step 5	**ip accounting-transits** *count* **Example:** `Router(config)# ip accounting-transits 100`	(Optional) Controls the number of transit records that will be stored in the IP accounting database.
Step 6	**interface** *type***/***number* **Example:** `Router(config)# interface ethernet1/1`	Specifies the interface and enters interface configuration mode.
Step 7	**ip accounting** [**access-violations**] [**output-packets**] **Example:** `Router(config-if)# ip accounting access-violations`	Enables basic IP accounting. • Use the optional **access-violations** keyword to enable IP accounting with the ability to identify IP traffic that fails IP access lists. • Use the optional **output-packets** keyword to enable IP accounting based on the IP packets output on the interface.

Step 8	**ip accounting mac-address** {**input** | **output**}
	or
	ip accounting precedence {**input** | **output**}
	Example:
	`Router(config-if)# ip accounting mac-address output`
	or
	Example:
	`Router(config-if)# ip accounting precedence output`
		(Optional) Configures IP accounting based on the MAC address of received (input) or transmitted (output) packets.
		or
		(Optional) Configures IP accounting based on the precedence of received (input) or transmitted (output) packets.

Monitoring and Maintaining the IP Network

You can display specific statistics such as the contents of IP routing tables, caches, databases and socket processes. The resulting information can be used to determine resource utilization and to solve network problems.

To monitor and maintain your IP network, perform any of the optional steps in this task.

SUMMARY STEPS

1. **clear ip traffic**
2. **clear ip accounting** [**checkpoint**]
3. **clear sockets** *process-id*
4. **show ip accounting** [**checkpoint**] [**output-packets** | **access-violations**]
5. **show interface** [*type number*] **mac**
6. **show interface** [*type number*] **precedence**
7. **show ip redirects**
8. **show ip sockets**
9. **show sockets** *process-id* [**detail**] [**events**]
10. **show udp** [**detail**]
11. **show ip traffic**

Note In Cisco IOS Release 12.4(11)T and later releases, the **show ip sockets** command was replaced by the **show udp**, **show sockets**, and **show ip sctp** commands. See the *Cisco IOS Voice Command Reference* for information about the **show ip sctp** command.

Step 1 **clear ip traffic**

To clear all IP traffic statistical counters on all interfaces, use the following command:

```
Router# clear ip traffic
```

Step 2 **clear ip accounting** [**checkpoint**]

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database can become necessary when the contents of the particular structure have become or are suspected to be invalid. To clear the active IP accounting database when IP accounting is enabled, use the following command:

```
Router# clear ip accounting
```

To clear the checkpointed IP accounting database when IP accounting is enabled, use the following command:

```
Router# clear ip accounting checkpoint
```

Step 3 **clear sockets** *process-id*

To close all IP sockets and clear the underlying transport connections and data structures for the specified process, use the following command:

```
Router# clear sockets 35
```

```
All sockets (TCP, UDP and SCTP) for this process will be cleared.
Do you want to proceed? [yes/no]: y
Cleared sockets for PID 35
```

Step 4 **show ip accounting** [**checkpoint**] [**output-packets** | **access-violations**]

To display access list violations, use the **show ip accounting** command. To use this command, you must first enable IP accounting on a per-interface basis.

Use the **checkpoint** keyword to display the checkpointed database. Use the **output-packets** keyword to indicate that information pertaining to packets that passed access control and were routed should be displayed. Use the **access-violations** keyword to display the number of the access list failed by the last packet for the source and destination pair. The number of packets reveals how aggressive the attack is upon a specific destination. If you do not specify the **access-violations** keyword, the command defaults to displaying the number of packets that have passed access lists and were routed.

If neither the **output-packets** nor **access-violations** keyword is specified, output-packets is the default.

The following is sample output from the **show ip accounting** command:

```
Router# show ip accounting

   Source           Destination              Packets               Bytes
 172.16.19.40    192.168.67.20                  7                 306
 172.16.13.55    192.168.67.20                 67                 2749
 172.16.2.50     192.168.33.51                  17                1111
 172.16.2.50     172.31.2.1                     5                 319
 172.16.2.50     172.31.1.2                    463                30991
 172.16.19.40    172.16.2.1                      4                 262
 172.16.19.40    172.16.1.2                      28                2552
 172.16.20.2     172.16.6.100                    39                2184
 172.16.13.55    172.16.1.2                      35                3020
 172.16.19.40    192.168.33.51                  1986               95091
 172.16.2.50     192.168.67.20                   233               14908
 172.16.13.28    192.168.67.53                   390               24817
 172.16.13.55    192.168.33.51                214669             9806659
 172.16.13.111   172.16.6.23                  27739             1126607
 172.16.13.44    192.168.33.51                 35412             1523980
 192.168.7.21    172.163.1.2                       11                 824
 172.16.13.28    192.168.33.2                      21                1762
 172.16.2.166    192.168.7.130                   797              141054
 172.16.3.11     192.168.67.53                       4                 246
 192.168.7.21    192.168.33.51                 15696              695635
 192.168.7.24    192.168.67.20                     21                 916
 172.16.13.111   172.16.10.1                     16                1137
 accounting threshold exceeded for 7 packets and 433 bytes
```

The following is sample output from the **show ip accounting access-violations** command. The output pertains to packets that failed access lists and were not routed:

```
Router# show ip accounting access-violations

   Source           Destination      Packets        Bytes        ACL
172.16.19.40    192.168.67.20              7          306         77
172.16.13.55    192.168.67.20             67         2749        185
172.16.2.50     192.168.33.51             17         1111        140
172.16.2.50     172.16.2.1                5          319        140
172.16.19.40    172.16.2.1                4          262         77
Accounting data age is 41
```

Step 5 **show interface** [*type number*] **mac**

To display information for interfaces configured for MAC accounting, use the **show interface mac** command. The following is sample output from the **show interface mac** command:

```
Router# show interface ethernet 0/1 mac

Ethernet0/1
Input  (511 free)
0007.f618.4449(228):  4 packets, 456 bytes, last: 2684ms ago
Total:  4 packets, 456 bytes
Output  (511 free)
0007.f618.4449(228):  4 packets, 456 bytes, last: 2692ms ago
Total:  4 packets, 456 bytes
```

Step 6 **show interface** [*type number*] **precedence**

To display information for interfaces configured for precedence accounting, use the **show interface precedence** command.

The following is sample output from the **show interface precedence** command. In this example, the total packet and byte counts are calculated for the interface that receives (input) or sends (output) IP packets and sorts the results based on IP precedence.

```
Router# show interface ethernet 0/1 precedence

Ethernet0/1
Input
Precedence 0:  4 packets, 456 bytes
Output
Precedence 0:  4 packets, 456 bytes
```

Step 7 **show ip redirects**

To display the address of the default router and the address of hosts for which an ICMP redirect message has been received, use the **show ip redirects** command.

The following is sample output from the **show ip redirects** command:

```
Router# show ip redirects

Default gateway is 172.16.80.29

Host               Gateway           Last Use    Total Uses  Interface
172.16.1.111      172.16.80.240         0:00             9  Ethernet0
172.16.1.4        172.16.80.240         0:00             4  Ethernet0
```

Step 8 **show ip sockets**

To display IP socket information, and to verify that the socket being used is opening correctly, use the **show ip sockets** command. If there is a local and remote endpoint, a connection is established with the ports indicated.

The following is sample output from the **show ip sockets** command:

```
Router# show ip sockets

Proto    Remote         Port      Local           Port  In Out Stat TTY OutputIF
 17      10.0.0.0         0       172.16.186.193  67    0   0    1   0
 17      172.16.191.135   514     172.16.191.129  1811  0   0    0   0
 17      172.16.135.20    514     172.16.191.1    4125  0   0    0   0
 17      172.16.207.163   49      172.16.186.193  49    0   0    9   0
 17      10.0.0.0         123     172.16.186.193  123   0   0    1   0
 88      10.0.0.0         0       172.16.186.193  202   0   0    0   0
 17      172.16.96.59     32856   172.16.191.1    161   0   0    1   0
 17     --listen--             --any--        496  0    0   1    0
```

Step 9 **show sockets** *process-id* [**detail**] [**events**]

To display the number of sockets currently open and their distribution with respect to the transport protocol process specified by the *process-id* argument, use the **show sockets** command. The following sample output from the **show sockets** command displays the total number of open sockets for the specified process:

```
Router# show sockets 35

Total open sockets - TCP:7, UDP:0, SCTP:0
```

The following sample output shows information about the same open processes with the **detail** keyword specified:

```
Router# show sockets 35 detail

   FD LPort FPort Proto Type    TransID

   0  5000  0     TCP   STREAM  0x6654DEBC
State: SS_ISBOUND
Options: SO_ACCEPTCONN

   1  5001  0     TCP   STREAM  0x6654E494
State: SS_ISBOUND
Options: SO_ACCEPTCONN

   2  5002  0     TCP   STREAM  0x656710B0
State: SS_ISBOUND
Options: SO_ACCEPTCONN

   3  5003  0     TCP   STREAM  0x65671688
State: SS_ISBOUND
Options: SO_ACCEPTCONN

   4  5004  0     TCP   STREAM  0x65671C60
State: SS_ISBOUND
Options: SO_ACCEPTCONN

   5  5005  0     TCP   STREAM  0x65672238
State: SS_ISBOUND
Options: SO_ACCEPTCONN

   6  5006  0     TCP   STREAM  0x64C7840C
State: SS_ISBOUND
Options: SO_ACCEPTCONN
```

Total open sockets - TCP:7, UDP:0, SCTP:0

The following example displays IP socket event information:

```
Router# show sockets 35 events

Events watched for this process: READ
FD Watched Present Select Present

0 --- --- R-- R--
```

Step 10 **show udp** [**detail**]

To display IP socket information about UDP processes, use the **show udp** command. The following example shows how to display detailed information about UDP sockets:

```
Router# show udp detail

 Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
```

```
17       10.0.0.0    0         10.0.21.70  67    0  0   2211 0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
17       10.0.0.0    0         10.0.21.70  2517  0   0  11   0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
17       10.0.0.0    0         10.0.21.70  5000  0  0   211  0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
17       10.0.0.0    0         10.0.21.70  5001  0  0   211  0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
17       10.0.0.0    0         10.0.21.70  5002  0  0   211  0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
17       10.0.0.0    0         10.0.21.70  5003  0  0   211  0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
Proto    Remote      Port      Local       Port  In Out Stat TTY OutputIF
17       10.0.0.0    0         10.0.21.70  5004  0  0   211  0
Queues: output 0
        input  0 (drops 0, max 50, highwater 0)
```

Note In Cisco IOS Release 12.4(11)T and later releases, the **show ip sockets** command was replaced by the **show udp**, **show sockets**, and **show ip sctp** commands. See the *Cisco IOS Voice Command Reference* for information about the **show ip sctp** command.

Step 11 **show ip traffic**

To display IP protocol statistics, use the **show ip traffic** command. The following example shows that the IP traffic statistics have been cleared by the **clear ip traffic** command:

```
Router# clear ip traffic
Router# show ip traffic

IP statistics:
 Rcvd:  0 total, 0 local destination
        0 format errors, 0 checksum errors, 0 bad hop count
        0 unknown protocol, 0 not a gateway
        0 security failures, 0 bad options, 0 with options
 Opts:  0 end, 0 nop, 0 basic security, 0 loose source route
        0 timestamp, 0 extended security, 0 record route
        0 stream ID, 0 strict source route, 0 alert, 0 cipso
        0 other
 Frags: 0 reassembled, 0 timeouts, 0 couldn't reassemble
        0 fragmented, 0 couldn't fragment
 Bcast: 0 received, 0 sent
 Mcast: 0 received, 0 sent
 Sent: 0 generated, 0 forwarded
 Drop: 0 encapsulation failed, 0 unresolved, 0 no adjacency
       0 no route, 0 unicast RPF, 0 forced drop

ICMP statistics:
 Rcvd: 0 format errors, 0 checksum errors, 0 redirects, 0 unreachable
       0 echo, 0 echo reply, 0 mask requests, 0 mask replies, 0 quench
       0 parameter, 0 timestamp, 0 info request, 0 other
       0 irdp solicitations, 0 irdp advertisements
 Sent: 0 redirects, 0 unreachable, 0 echo, 0 echo reply
       0 mask requests, 0 mask replies, 0 quench, 0 timestamp
       0 info reply, 0 time exceeded, 0 parameter problem
       0 irdp solicitations, 0 irdp advertisements

UDP statistics:
 Rcvd: 0 total, 0 checksum errors, 0 no port
 Sent: 0 total, 0 forwarded broadcasts

TCP statistics:
 Rcvd: 0 total, 0 checksum errors, 0 no port
 Sent: 0 total

Probe statistics:
 Rcvd: 0 address requests, 0 address replies
       0 proxy name requests, 0 where-is requests, 0 other
 Sent: 0 address requests, 0 address replies (0 proxy)
       0 proxy name replies, 0 where-is replies

EGP statistics:
 Rcvd: 0 total, 0 format errors, 0 checksum errors, 0 no listener
 Sent: 0 total

IGRP statistics:
 Rcvd: 0 total, 0 checksum errors
 Sent: 0 total

OSPF statistics:
 Rcvd: 0 total, 0 checksum errors
       0 hello, 0 database desc, 0 link state req
       0 link state updates, 0 link state acks

 Sent: 0 total

IP-IGRP2 statistics:
 Rcvd: 0 total
 Sent: 0 total

PIMv2 statistics: Sent/Received
 Total: 0/0, 0 checksum errors, 0 format errors
 Registers: 0/0, Register Stops: 0/0, Hellos: 0/0
 Join/Prunes: 0/0, Asserts: 0/0, grafts: 0/0
 Bootstraps: 0/0, Candidate_RP_Advertisements: 0/0

IGMP statistics: Sent/Received
 Total: 0/0, Format errors: 0/0, Checksum errors: 0/0
 Host Queries: 0/0, Host Reports: 0/0, Host Leaves: 0/0
 DVMRP: 0/0, PIM: 0/0
```

Configuration Examples for IP Services

This section provides the following IP configuration examples:

- Protecting Your Network from DOS Attacks: Example, page 19
- Configuring ICMP Unreachable Destination Counters: Example, page 19
- Setting the MTU Packet Size: Example, page 19
- Configuring IP Accounting: Example, page 19

Protecting Your Network from DOS Attacks: Example

The following example shows how to change some of the ICMP defaults for Ethernet interface 0/0 to prevent ICMP from relaying information about paths, routes, and network conditions, which can be used by an attacker to gain network mapping information.

Disabling the unreachable messages will have a secondary effect: it also will disable IP Path MTU Discovery, because path discovery works by having the Cisco IOS software send Unreachable messages. If you have a network segment with a small number of devices and an absolutely reliable traffic pattern—which could easily happen on a segment with a small number of rarely used user devices—you would be disabling options that your device would be unlikely to use anyway.

```
configure terminal
no ip source-route
interface ethernet 0/0
 no ip unreachables
 no ip redirects
 no ip mask-reply
```

Configuring ICMP Unreachable Destination Counters: Example

The following example shows how to clear all of the unreachable destination packet statistics and to specify an interval number for unreachable destination messages. This example also shows how to configure a packet counter threshold and interval to trigger a logging message to a console.

```
clear ip icmp rate-limit ethernet 0/0
configure terminal
 ip icmp rate-limit unreachable df log 1100 12000
```

Setting the MTU Packet Size: Example

The following example shows how to change the default MTU packet size for Ethernet interface 0/0:

```
configure terminal
interface ethernet 0/0
 ip mtu 300
```

Configuring IP Accounting: Example

The following example shows how to enable IP accounting based on the source and destination MAC address and based on IP precedence for received and transmitted packets:

```
configure terminal
 interface Ethernet0/5
 ip accounting mac-address input
 ip accounting mac-address output
 ip accounting precedence input
 ip accounting precedence output
```

Additional References

The following sections provide references related to IP services.

Related Documents

Related Topic	Document Title
IP addressing and services configuration tasks	*Cisco IOS IP Addressing Services Configuration Guide*
IP application services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference.*

RFCs

RFC	Title
RFC 791	*Internet Protocol*
RFC 792	*Internet Control Message Protocol*
RFC 1191	*Path MTU discovery*

Technical Assistance

Description	Link
The Cisco Technical Support & Documentation website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, tools, and technical documentation. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Feature Information for IP Services

Table 1 lists the features in this module and provides links to specific configuration information.

For information on a feature in this technology that is not documented here, see the "Cisco IOS IP Application Services Features Roadmap" or the "FHRP Features Roadmap."

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for IP Services

Feature Name	Releases	Feature Information
Clear IP Traffic CLI	12.4(2)T 12.2(31)SB2 Cisco IOS XE Release 2.1	The Clear IP Traffic CLI feature introduced the **clear ip traffic** command to clear all IP traffic statistics on a router instead of reloading the router. For added safety, the user will see a confirmation prompt when entering this command. In Cisco IOS Release 12.4(2)T, this feature was introduced. The following sections provide information about this feature: • Monitoring and Maintaining the IP Network, page 13 The following command was introduced by this feature: **clear ip traffic**.
ICMP Unreachable Rate Limiting User Feedback	12.4(2)T 12.2(31)SB2	The ICMP Unreachable Rate Limiting User Feedback feature enables you to clear and display packets that have been discarded because of an unreachable destination, and to configure a threshold interval for triggering error messages. When message logging is generated, it displays on your console. In Cisco IOS Release 12.4(2)T, this feature was introduced. The following sections provide information about this feature: • ICMP Overview, page 2 • Denial of Service Attack, page 4 • Configuring ICMP Unreachable Rate Limiting User Feedback, page 8 • Protecting Your Network from DOS Attacks: Example, page 19 The following commands were introduced or modified by this feature: **clear ip icmp rate-limit**, **ip icmp rate-limit unreachable**, **show ip icmp rate-limit**.

Table 1 Feature Information for IP Services (continued)

Feature Name	Releases	Feature Information
IP Precedence Accounting	12.2(21) 12.1(27b)E1 12.1(5)T15 12.2(25)S 12.2(33)SRA 12.2(18)SXF13 12.2(33)SXH1 Cisco IOS XE Release 2.1	The IP Precedence Accounting feature provides accounting information for IP traffic based on the precedence of any interface. This feature calculates the total packet and byte counts for an interface that receives or sends IP packets and sorts the results based on the IP precedence. This feature is supported on all interfaces and subinterfaces and supports CEF, dCEF, flow, and optimum switching. The following sections provide information about this feature: • IP MAC and Precedence Accounting, page 6 • Configuring IP Accounting: Example, page 19 The following command was introduced by this feature: **show interface precedence**, **ip accounting precedence**.
Show and Clear Commands for IOS Sockets	12.4(11)T	The Show and Clear Commands for IOS Sockets feature introduces the **show udp**, **show sockets**, and **clear sockets** commands. These new commands are useful for monitoring and managing the Cisco IOS Socket library. The following sections provide information about this feature: • Show and Clear Commands for IOS Sockets, page 6 • Monitoring and Maintaining the IP Network, page 13 The following commands were introduced or modified by this feature: **clear sockets**, **show sockets**, **show udp**. The following command was replaced by this feature: **show ip sockets**.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring TCP

First Published: October 23, 2006
Last Updated: May 5, 2008

The Transmission Control Protocol (TCP) is a protocol that specifies the format of data and acknowledgments used in data transfer. TCP is a connection-oriented protocol because participants must establish a connection before data can be transferred. By performing flow control and error correction, TCP guarantees reliable, in-sequence delivery of packets. It is considered a reliable protocol because if an IP packet is dropped or received out of order, TCP will request the correct packet until it receives it.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for TCP" section on page 19.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for TCP

TCP Time Stamp, TCP Selective Acknowledgment, and TCP Header Compression

Because TCP time stamps are always sent and echoed in both directions and the time-stamp value in the header is always changing, TCP header compression will not compress the outgoing packet. To allow TCP header compression over a serial link, the TCP time-stamp option is disabled. If you want to use TCP header compression over a serial line, TCP time stamp and TCP selective acknowledgment must be disabled. Both features are disabled by default. Use the **no ip tcp selective-ack** command to disable TCP selective acknowledgment once it is enabled.

Information About TCP

To configure TCP, you should understand the following concepts:

- TCP Services, page 2
- TCP Connection Establishment, page 3
- TCP Connection Attempt Time, page 3
- TCP Selective Acknowledgment, page 3
- TCP Time Stamp, page 4
- TCP Maximum Read Size, page 4
- TCP Path MTU Discovery, page 4
- TCP Window Scaling, page 5
- TCP Sliding Window, page 5
- TCP Outgoing Queue Size, page 6
- TCP Congestion Avoidance, page 6
- TCP Explicit Congestion Notification, page 6
- TCP MSS Adjustment, page 6
- TCP Applications Flags Enhancement, page 7
- TCP Show Extension, page 7

TCP Services

TCP provides reliable transmission of data in an IP environment. TCP corresponds to the transport layer (Layer 4) of the OSI reference model. Among the services TCP provides are stream data transfer, reliability, efficient flow control, full-duplex operation, and multiplexing.

With stream data transfer, TCP delivers an unstructured stream of bytes identified by sequence numbers. This service benefits applications because they do not have to chop data into blocks before handing it off to TCP. Instead, TCP groups bytes into segments and passes them to IP for delivery.

TCP offers reliability by providing connection-oriented, end-to-end reliable packet delivery through an internetwork. It does this by sequencing bytes with a forwarding acknowledgment number that indicates to the destination the next byte the source expects to receive. Bytes not acknowledged within a specified time period are retransmitted. The reliability mechanism of TCP allows devices to deal with lost, delayed, duplicate, or misread packets. A time-out mechanism allows devices to detect lost packets and request retransmission.

TCP offers efficient flow control, which means that, when sending acknowledgments back to the source, the receiving TCP process indicates the highest sequence number it can receive without overflowing its internal buffers.

TCP offers full-duplex operation and TCP processes can both send and receive at the same time.

TCP multiplexing allows numerous simultaneous upper-layer conversations to be multiplexed over a single connection.

TCP Connection Establishment

To use reliable transport services, TCP hosts must establish a connection-oriented session with one another. Connection establishment is performed by using a “three-way handshake” mechanism.

A three-way handshake synchronizes both ends of a connection by allowing both sides to agree upon initial sequence numbers. This mechanism also guarantees that both sides are ready to transmit data and know that the other side is ready to transmit as well. The three-way handshake is necessary so that packets are not transmitted or retransmitted during session establishment or after session termination.

Each host randomly chooses a sequence number used to track bytes within the stream it is sending. Then, the three-way handshake proceeds in the following manner:

- The first host (Host A) initiates a connection by sending a packet with the initial sequence number (X) and synchronize/start (SYN) bit set to indicate a connection request.
- The second host (Host B) receives the SYN, records the sequence number X, and replies by acknowledging the SYN (with an ACK = X + 1). Host B includes its own initial sequence number (SEQ = Y). An ACK = 20 means the host has received bytes 0 through 19 and expects byte 20 next. This technique is called forward acknowledgment.
- Host A acknowledges all bytes Host B sent with a forward acknowledgment indicating the next byte Host A expects to receive (ACK = Y + 1). Data transfer then can begin.

TCP Connection Attempt Time

You can set the amount of time the Cisco IOS software will wait to attempt to establish a TCP connection. Because the connection attempt time is a host parameter, it does not pertain to traffic going through the device, just to traffic originated at the device. To set the TCP connection attempt time, use the **ip tcp synwait-time** command in global configuration mode. The default is 30 seconds.

TCP Selective Acknowledgment

The TCP Selective Acknowledgment feature improves performance in the event that multiple packets are lost from one TCP window of data.

Prior to this feature, with the limited information available from cumulative acknowledgments, a TCP sender could learn about only one lost packet per round-trip time. An aggressive sender could choose to resend packets early, but such resent segments might have already been successfully received.

The TCP selective acknowledgment mechanism helps improve performance. The receiving TCP host returns selective acknowledgment packets to the sender, informing the sender of data that have been received. In other words, the receiver can acknowledge packets received out of order. The sender can then resend only the missing data segments (instead of everything since the first missing packet).

Prior to selective acknowledgment, if TCP lost packets 4 and 7 out of an 8-packet window, TCP would receive acknowledgment of only packets 1, 2, and 3. Packets 4 through 8 would need to be resent. With selective acknowledgment, TCP receives acknowledgment of packets 1, 2, 3, 5, 6, and 8. Only packets 4 and 7 must be resent.

TCP selective acknowledgment is used only when multiple packets are dropped within one TCP window. There is no performance impact when the feature is enabled but not used. Use the **ip tcp selective-ack** command in global configuration mode to enable TCP selective acknowledgment.

Refer to RFC 2018 for more detailed information about TCP selective acknowledgment.

TCP Time Stamp

The TCP time-stamp option provides better TCP round-trip time measurements. Because the time stamps are always sent and echoed in both directions and the time-stamp value in the header is always changing, TCP header compression will not compress the outgoing packet. To allow TCP header compression over a serial link, the TCP time-stamp option is disabled. Use the **ip tcp timestamp** command to enable the TCP time-stamp option.

Refer to RFC 1323 for more detailed information on TCP time stamp. Refer to the "Configuring TCP Header Compression" chapter of the *Cisco IOS Quality of Service Solutions Configuration Guide* for more information about TCP header compression.

TCP Maximum Read Size

The maximum number of characters that TCP reads from the input queue for Telnet and rlogin at one time is a very large number (the largest possible 32-bit positive number) by default. To change the TCP maximum read size value, use the **ip tcp chunk-size** command in global configuration mode.

We do not recommend that you change this value.

TCP Path MTU Discovery

Path MTU Discovery is a method for maximizing the use of available bandwidth in the network between the endpoints of a TCP connection, which is described in RFC 1191. IP Path MTU Discovery allows a host to dynamically discover and cope with differences in the maximum allowable maximum transmission unit (MTU) size of the various links along the path. Sometimes a router is unable to forward a datagram because it requires fragmentation (the packet is larger than the MTU you set for the interface with the interface configuration command), but the "don't fragment" (DF) bit is set. The intermediate gateway sends a "Fragmentation needed and DF bit set" ICMP message to the sending host, alerting it to the problem. Upon receiving this ICMP message, the host reduces its assumed path MTU and consequently sends a smaller packet that will fit the smallest packet size of all the links along the path.

By default, TCP Path MTU Discovery is disabled. Existing connections are not affected when this feature is enabled or disabled.

Customers using TCP connections to move bulk data between systems on distinct subnets would benefit most by enabling this feature. Customers using remote source-route bridging (RSRB) with TCP encapsulation, serial tunnel (STUN), X.25 Remote Switching (also known as XOT or X.25 over TCP), and some protocol translation configurations might also benefit from enabling this feature.

Use the **ip tcp path-mtu-discovery** global configuration command to enable Path MTU Discovery for connections initiated by the router when it is acting as a host.

For more information about Path MTU Discovery, refer to the "Configuring IP Services" chapter of the *Cisco IOS IP Application Services Configuration Guide*.

TCP Window Scaling

The TCP Window Scaling feature adds support for the Window Scaling option in RFC 1323. A larger window size is recommended to improve TCP performance in network paths with large bandwidth-delay product characteristics that are called Long Fat Networks (LFNs). The TCP Window Scaling enhancement provides that support.

The window scaling extension in Cisco IOS software expands the definition of the TCP window to 32 bits and then uses a scale factor to carry this 32-bit value in the 16-bit window field of the TCP header. The window size can increase to a scale factor of 14. Typical applications use a scale factor of 3 when deployed in LFNs.

The TCP Window Scaling feature complies with RFC 1323, *TCP Extensions for High Performance*. The maximum window size has been increased to 1,073,741,823 bytes. The larger scalable window size will allow TCP to perform better over LFNs. Use the **ip tcp window-size** command in global configuration mode to configure the TCP window size.

TCP Sliding Window

A TCP sliding window provides more efficient use of network bandwidth because it enables hosts to send multiple bytes or packets before waiting for an acknowledgment.

In TCP, the receiver specifies the current window size in every packet. Because TCP provides a byte-stream connection, window sizes are expressed in bytes. A window is the number of data bytes that the sender is allowed to send before waiting for an acknowledgment. Initial window sizes are indicated at connection setup, but might vary throughout the data transfer to provide flow control. A window size of zero means "Send no data." The default TCP window size is 4128 bytes. We recommend you keep the default value unless you know your router is sending large packets (greater than 536 bytes). Use the **ip tcp window-size** command to change the default window size.

In a TCP sliding-window operation, for example, the sender might have a sequence of bytes to send (numbered 1 to 10) to a receiver who has a window size of five. The sender then places a window around the first five bytes and transmits them together. The sender then waits for an acknowledgment.

The receiver responds with an ACK = 6, indicating that it has received bytes 1 to 5 and is expecting byte 6 next. In the same packet, the receiver indicates that its window size is 5. The sender then moves the sliding window five bytes to the right and transmit bytes 6 to 10. The receiver responds with an ACK = 11, indicating that it is expecting sequenced byte 11 next. In this packet, the receiver might indicate that its window size is 0 (because, for example, its internal buffers are full). At this point, the sender cannot send any more bytes until the receiver sends another packet with a window size greater than 0.

TCP Outgoing Queue Size

The default TCP outgoing queue size per connection is 5 segments if the connection has a TTY associated with it (such as a Telnet connection). If no TTY connection is associated with a connection, the default queue size is 20 segments. Use the **ip tcp queuemax** command to change the 5-segment default value.

TCP Congestion Avoidance

The TCP Congestion Avoidance feature enables the monitoring of acknowledgment packets to the TCP sender when multiple packets are lost in a single window of data. Previously the sender would exit Fast-Recovery mode, wait for three or more duplicate acknowledgment packets before retransmitting the next unacknowledged packet, or wait for the retransmission timer to slow start. This could lead to performance issues.

Implementation of RFC 2581 and RFC 3782 addresses the modifications to the Fast-Recovery algorithm that incorporates a response to partial acknowledgments received during Fast Recovery, improving performance in situations where multiple packets are lost in a single window of data.

This feature is an enhancement to the existing Fast Recovery algorithm. There are no commands used to enable or disable this feature.

To monitor the acknowledgment packets, the output of the **debug ip tcp transactions** command has been enhanced to show the following conditions:

- TCP entering Fast Recovery mode.
- Duplicate acknowledgments being received during Fast Recovery mode.
- Partial acknowledgments being received.

TCP Explicit Congestion Notification

The TCP Explicit Congestion Notification (ECN) feature provides a method for an intermediate router to notify the end hosts of impending network congestion. It also provides enhanced support for TCP sessions associated with applications that are sensitive to delay or packet loss including Telnet, web browsing, and transfer of audio and video data. The benefit of this feature is the reduction of delay and packet loss in data transmissions. Use the **ip tcp ecn** command in global configuration mode to enable TCP ECN.

TCP MSS Adjustment

The TCP MSS Adjustment feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments with the SYN bit set. Use the **ip tcp adjust-mss** command in interface configuration mode to specify the MSS value on the intermediate router of the SYN packets to avoid truncation.

When a host (usually a PC) initiates a TCP session with a server, it negotiates the IP segment size by using the MSS option field in the TCP SYN packet. The value of the MSS field is determined by the maximum transmission unit (MTU) configuration on the host. The default MSS value for a PC is 1500 bytes.

The PPP over Ethernet (PPPoE) standard supports a MTU of only 1492 bytes. The disparity between the host and PPPoE MTU size can cause the router in between the host and the server to drop 1500-byte packets and terminate TCP sessions over the PPPoE network. Even if the path MTU (which detects the correct MTU across the path) is enabled on the host, sessions may be dropped because system administrators sometimes disable the ICMP error messages that must be relayed from the host in order for path MTU to work.

The **ip tcp adjust-mss** command helps prevent TCP sessions from being dropped by adjusting the MSS value of the TCP SYN packets.

The **ip tcp adjust-mss** command is effective only for TCP connections passing through the router.

In most cases, the optimum value for the *max-segment-size* argument of the **ip tcp adjust-mss** command is 1452 bytes. This value plus the 20-byte IP header, the 20-byte TCP header, and the 8-byte PPPoE header add up to a 1500-byte packet that matches the MTU size for the Ethernet link.

See the "Configuring the MSS Value and MTU for Transient TCP SYN Packets" section on page 9 for configuration instructions.

TCP Applications Flags Enhancement

The TCP Applications Flags Enhancement feature enables the user to display additional flags with reference to TCP applications. There are two types of flags: status and option. The status flags indicate the status of TCP connections such as retransmission timeouts, application closed, and synchronized (SYNC) handshakes for listen. The additional flags indicate the state of set options such as whether or not a virtual private network (VPN) routing and forwarding (VRF) instance is set, whether or not a user is idle, and whether or not a keepalive timer is running. Use the **show tcp** command to display TCP application flags.

TCP Show Extension

The TCP Show Extension feature introduces the capability to display addresses in IP format instead of hostname format and to display the virtual private network (VPN) routing and forwarding (VRF) table associated with the connection. To display the status for all endpoints with the addresses in IP format, use the **show tcp brief numeric** command.

How to Configure TCP

This section contains the following procedures:

- Configuring TCP Performance Parameters, page 7
- Configuring the MSS Value and MTU for Transient TCP SYN Packets, page 9
- Verifying TCP Performance Parameters, page 10

Configuring TCP Performance Parameters

Perform the following task to configure TCP performance parameters.

Prerequisites

- Both sides of the link must be configured to support window scaling or the default of 65,535 bytes will apply as the maximum window size.
- To support ECN, the remote peer must be ECN-enabled because the ECN capability is negotiated during a three-way handshake with the remote peer.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip tcp synwait-time** *seconds*
4. **ip tcp path-mtu-discovery** [**age-timer** {*minutes* | **infinite**}]
5. **ip tcp selective-ack**
6. **ip tcp timestamp**
7. **ip tcp chunk-size** *characters*
8. **ip tcp window-size** *bytes*
9. **ip tcp ecn**
10. **ip tcp queuemax** *packets*

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip tcp synwait-time`** *`seconds`*
	Example:
	`Router(config)# ip tcp synwait-time 60`
		(Optional) Sets the amount of time the Cisco IOS software will wait to attempt to establish a TCP connection. The default is 30 seconds.

Step 4	**`ip tcp path-mtu-discovery`** `[`**`age-timer`** `{`*`minutes`* `|` **`infinite`**`}]`
	Example:
	`Router(config)# ip tcp path-mtu-discovery age-timer 11`
		(Optional) Enables Path MTU Discovery.
		• **age-timer**—Time interval, in minutes, TCP reestimates the path MTU with a larger maximum segment size (MSS). The default is 10 minutes. The maximum is 30 minutes.
		• **infinite**—Disables the age timer.

Step 5	**`ip tcp selective-ack`**
	Example:
	`Router(config)# ip tcp selective-ack`
		(Optional) Enables TCP selective acknowledgment.

	Command or Action	Purpose

Step 6	**`ip tcp timestamp`**
	Example:
	`Router(config)# ip tcp timestamp`
		(Optional) Enables the TCP time stamp.

Step 7	**`ip tcp chunk-size`** *`characters`*
	Example:
	`Router(config)# ip tcp chunk-size 64000`
		(Optional) Sets the TCP maximum read size for Telnet or rlogin.
		Note We do not recommend that you change this value.

Step 8	**`ip tcp window-size`** *`bytes`*
	Example:
	`Router(config)# ip tcp window-size 75000`
		(Optional) Sets the TCP window size.
		The *bytes* argument can be set to an integer from 0 to 1073741823. To enable window scaling to support LFNs, the TCP window size must be more than 65535. The default window size is 4128 if window scaling is not configured.

Step 9	`ip tcp ecn`
	Example:
	`Router(config)# ip tcp ecn`
		(Optional) Enables ECN for TCP.

Step 10	**`ip tcp queuemax`** *`packets`*
	Example:
	`Router(config)# ip tcp queuemax 10`
		(Optional) Sets the TCP outgoing queue size.

Configuring the MSS Value and MTU for Transient TCP SYN Packets

Perform this task to configure the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments with the SYN bit set, and to configure the MTU size of IP packets.

If you are configuring the **ip mtu** command on the same interface as the **ip tcp adjust-mss** command, we recommend that you use the following commands and values:

- **ip tcp adjust-mss 1452**
- **ip mtu 1492**

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip tcp adjust-mss** *max-segment-size*
5. **ip mtu** *bytes*
6. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip tcp adjust-mss`** *`max-segment-size`*
	Example:
	`Router(config-if)# ip tcp adjust-mss 1452`
		Adjusts the MSS value of TCP SYN packets going through a router. The *max-segment-size* argument is the maximum segment size, in bytes. The range is from 500 to 1460.

Step 5	**`ip mtu`** *`bytes`*
	Example:
	`Router(config-if)# ip mtu 1492`
		Sets the MTU size of IP packets, in bytes, sent on an interface.

Step 6	**`end`**
	Example:
	`Router(config-if)# end`
		Exits to global configuration mode.

Verifying TCP Performance Parameters

This task shows you how to verify configured TCP performance parameters.

SUMMARY STEPS

1. **show tcp** [*line-number*] [**tcb** *address*]
2. **show tcp brief** [**all** | **numeric**]
3. **debug ip tcp transactions**

DETAILED STEPS

Step 1 **show tcp** [*line-number*] [**tcb** *address*]

Displays the status of TCP connections. The arguments and keyword are as follows:

- *line-number*—(Optional) Absolute line number of the Telnet connection status.
- **tcb**—(Optional) Transmission control block (TCB) of the ECN-enabled connection.

- *address*—(Optional) TCB hexadecimal address. The valid range is from 0x0 to 0xFFFFFFFF.

The following is sample output from the **show tcp tcb** command that displays detailed information by hexadecimal address about an ECN-enabled connection:

```
Router# show tcp tcb 0x62CD2BB8

Connection state is LISTEN, I/O status: 1, unread input bytes: 0
Connection is ECN enabled
Local host: 10.10.10.1, Local port: 179
Foreign host: 10.10.10.2, Foreign port: 12000

Enqueued packets for retransmit: 0, input: 0 mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x4F31940):
Timer          Starts    Wakeups            Next
Retrans             0          0             0x0
TimeWait            0          0             0x0
AckHold             0          0             0x0
SendWnd             0          0             0x0
KeepAlive           0          0             0x0
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0

iss:          0 snduna:          0 sndnxt:          0     sndwnd:      0
irs:          0 rcvnxt:          0 rcvwnd:       4128  delrcvwnd:      0

SRTT: 0 ms, RTTO: 2000 ms, RTV: 2000 ms, KRTT: 0 ms
minRTT: 60000 ms, maxRTT: 0 ms, ACK hold: 200 ms
Flags: passive open, higher precedence, retransmission timeout

TCB is waiting for TCP Process (67)

Datagrams (max data segment is 516 bytes):
Rcvd: 6 (out of order: 0), with data: 0, total data bytes: 0
Sent: 0 (retransmit: 0, fastretransmit: 0), with data: 0, total data
bytes: 0
```

Cisco IOS Software Modularity

The following is sample output from the **show tcp tcb** command from a Software Modularity image:

```
Router# show tcp tcb 0x1059C10

Connection state is ESTAB, I/O status: 0, unread input bytes: 0
Local host: 10.4.2.32, Local port: 23
Foreign host: 10.4.2.39, Foreign port: 11000
VRF table id is: 0

Current send queue size: 0 (max 65536)
Current receive queue size: 0 (max 32768)  mis-ordered: 0 bytes

Event Timers (current time is 0xB9ACB9):
Timer          Starts    Wakeups            Next(msec)
Retrans             6          0                0
SendWnd             0          0                0
TimeWait            0          0                0
AckHold             8          4                0
KeepAlive          11          0          7199992
PmtuAger            0          0                0
GiveUp              0          0                0
Throttle            0          0                0

irs:    1633857851  rcvnxt: 1633857890  rcvadv: 1633890620  rcvwnd:  32730
```

```
iss:    4231531315  snduna: 4231531392  sndnxt: 4231531392  sndwnd:   4052
sndmax: 4231531392  sndcwnd:     10220

SRTT: 84 ms,  RTTO: 650 ms,  RTV: 69 ms,  KRTT: 0 ms
minRTT: 0 ms,  maxRTT: 200 ms, ACK hold: 200 ms

Keepalive time: 7200 sec, SYN wait time: 75 sec
Giveup time: 0 ms, Retransmission retries: 0, Retransmit forever: FALSE

State flags: none

Feature flags: Nagle

Request flags: none
Window scales: rcv 0, snd 0, request rcv 0, request snd 0
Timestamp option: recent 0, recent age 0, last ACK sent          0

Datagrams (in bytes): MSS 1460, peer MSS 1460, min MSS 1460, max MSS 1460
Rcvd: 14 (out of order: 0), with data: 10, total data bytes: 38
Sent: 10 (retransmit: 0, fastretransmit: 0), with data: 5, total data bytes: 76

Header prediction hit rate: 72 %

Socket states: SS_ISCONNECTED, SS_PRIV

Read buffer flags: SB_WAIT, SB_SEL, SB_DEL_WAKEUP
Read notifications: 4

Write buffer flags: SB_DEL_WAKEUP
Write notifications: 0
Socket status: 0
```

Step 2 **show tcp brief** [**all** | **numeric**]

(Optional) Displays addresses in IP format.

Use the **show tcp brief** command to display a concise description of TCP connection endpoints. The keywords are as follows. Use the optional **all** keyword to display the status for all endpoints with the addresses in a Domain Name System (DNS) hostname format. Without this keyword, endpoints in the LISTEN state are not shown. Use the optional **numeric** keyword to display the status for all endpoints with the addresses in IP format.

The following is sample output from the **show tcp brief** command while a user is connected to the system by using Telnet:

```
Router# show tcp brief

TCB       Local Address           Foreign Address        (state)
609789AC  Router.cisco.com.23     cider.cisco.com.3733   ESTAB
```

The following example shows the IP activity by using the **numeric** keyword to display the addresses in IP format.

```
Router# show tcp brief numeric

TCB           Local Address          Foreign Address      (state)
6523A4FC      10.1.25.3.11000        10.1.25.3.23          ESTAB
65239A84      10.1.25.3.23           10.1.25.3.11000       ESTAB
653FCBBC      *.1723 *.* LISTEN
```

Step 3 **debug ip tcp transactions**

Use the **debug ip tcp transactions** command to display information about significant TCP transactions such as state changes, retransmissions, and duplicate packets. This command is particularly useful for debugging a performance problem on a TCP/IP network that you have isolated above the data-link layer.

The following is sample output from the **debug ip tcp transactions** command:

```
Router# debug ip tcp transactions

TCP: sending SYN, seq 168108, ack 88655553
TCP0: Connection to 10.9.0.13:22530, advertising MSS 966
TCP0: state was LISTEN -> SYNRCVD [23 -> 10.9.0.13(22530)]
TCP0: state was SYNSENT -> SYNRCVD [23 -> 10.9.0.13(22530)]
TCP0: Connection to 10.9.0.13:22530, received MSS 956
TCP0: restart retransmission in 5996
TCP0: state was SYNRCVD -> ESTAB [23 -> 10.9.0.13(22530)]
TCP2: restart retransmission in 10689
TCP2: restart retransmission in 10641
TCP2: restart retransmission in 10633
TCP2: restart retransmission in 13384 -> 10.0.0.13(16151)]
TCP0: restart retransmission in 5996 [23 -> 10.0.0.13(16151)]
```

The following line from the **debug ip tcp transactions** command output shows that TCP has entered Fast Recovery mode:

```
fast re-transmit - sndcwnd - 512, snd_last - 33884268765
```

The following lines from the **debug ip tcp transactions** command output show that a duplicate acknowledgment is received when in Fast Recovery mode (first line) and a partial acknowledgment has been received (second line):

```
TCP0:ignoring second congestion in same window sndcwn - 512, snd_1st - 33884268765
TCP0:partial ACK received sndcwnd:338842495
```

Configuration Examples for TCP

This section provides the following configuration examples:

- Verifying the Configuration of TCP ECN: Example, page 13
- Configuring the TCP MSS Adjustment: Examples, page 15
- Configuring the TCP Application Flags Enhancement: Example, page 17
- Displaying Addresses in IP Format: Example, page 17

Verifying the Configuration of TCP ECN: Example

The following example shows how to verify that ECN is configured:

```
Router# show running-config

Building configuration...
.
.
.
ip tcp ecn ! ECN is configured.
.
.
.
```

The following example shows how to verify that TCP is ECN enabled on a specific connection (local host):

```
Router# show tcp tcb 123456A

!Local host
!
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Connection is ECN Enabled
Local host: 10.1.25.31, Local port: 11002
Foreign host: 10.1.25.34, Foreign port: 23
```

The following example shows how to display concise information about one address:

```
Router# show tcp brief
!
TCB          Local address            Foreign Address        (state)
609789C      Router.cisco.com.23      cider.cisco.com.3733    ESTAB
```

The following example show how to enable IP TCP ECN debugging:

```
Router# debug ip tcp ecn
!
TCP ECN debugging is on
!
Router# telnet 10.1.25.31

Trying 10.1.25.31 ...
!
01:43:19: 10.1.25.35:11000 <---> 10.1.25.31:23   out ECN-setup SYN
01:43:21: 10.1.25.35:11000 <---> 10.1.25.31:23   congestion window changes
01:43:21: cwnd from 1460 to 1460, ssthresh from 65535 to 2920
01:43:21: 10.1.25.35:11000 <---> 10.1.25.31:23   in non-ECN-setup SYN-ACK
```

Before a TCP connection can use ECN, a host sends an ECN-setup SYN (synchronization) packet to a remote end that contains an Echo Congestion Experience (ECE) and Congestion window reduced (CWR) bit set in the header. Setting the ECE and CWR bits indicates to the remote end that the sending TCP is ECN capable, rather than an indication of congestion. The remote end sends an ECN-setup SYN-ACK (acknowledgment) packet to the sending host.

In the example above, the “out ECN-setup SYN” text means that a SYN packet with the ECE and CWR bit set was sent to the remote end. The “in non-ECN-setup SYN-ACK” text means that the remote end did not favorably acknowledge the ECN request and, therefore, the session is ECN capable.

The following debug output shows that ECN capabilities are enabled at both ends. In response to the ECN-setup SYN, the other end favorably replied with an ECN-setup SYN-ACK message. This connection is now ECN capable for the rest of the session.

```
Router# telnet 10.10.10.10

Trying 10.10.10.10 ... Open
Password required, but none set
!
1d20h: 10.1.25.34:11003 <---> 10.1.25.35:23   out ECN-setup SYN
1d20h: 10.1.25.34:11003 <---> 10.1.25.35:23   in ECN-setup SYN-ACK
```

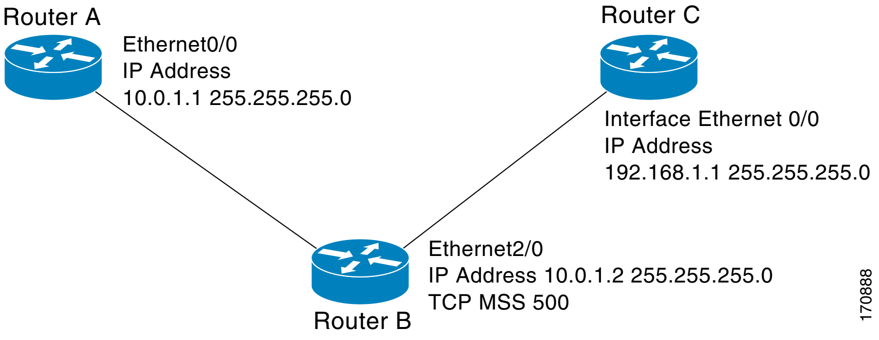
The following example shows how to verify that the hosts are connected:

```
Router# show debugging
!
TCP:
  TCP Packet debugging is on
  TCP ECN debugging is on
```

```
!
Router# telnet 10.1.25.234
!
Trying 10.1.25.234 ...
!
00:02:48: 10.1.25.31:11001 <---> 10.1.25.234:23   out ECN-setup SYN
00:02:48: tcp0: O CLOSED 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 ECE CWR SYN  WIN 4128
00:02:50: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:02:50: cwnd from 1460 to 1460, ssthresh from 65535 to 2920
00:02:50: tcp0: R SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 ECE CWR SYN  WIN 4128
00:02:54: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:02:54: cwnd from 1460 to 1460, ssthresh from 2920 to 2920
00:02:54: tcp0: R SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 ECE CWR SYN  WIN 4128
00:03:02: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:03:02: cwnd from 1460 to 1460, ssthresh from 2920 to 2920
00:03:02: tcp0: R SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 ECE CWR SYN  WIN 4128
00:03:18: 10.1.25.31:11001 <---> 10.1.25.234:23   SYN with ECN disabled
00:03:18: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:03:18: cwnd from 1460 to 1460, ssthresh from 2920 to 2920
00:03:18: tcp0: O SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 SYN  WIN 4128
00:03:20: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:03:20: cwnd from 1460 to 1460, ssthresh from 2920 to 2920
00:03:20: tcp0: R SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 SYN  WIN 4128
00:03:24: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:03:24: cwnd from 1460 to 1460, ssthresh from 2920 to 2920
00:03:24: tcp0: R SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 SYN  WIN 4128
00:03:32: 10.1.25.31:11001 <---> 10.1.25.234:23   congestion window changes
00:03:32: cwnd from 1460 to 1460, ssthresh from 2920 to 2920
00:03:32: tcp0: R SYNSENT 10.1.25.234:11001 10.1.25.31:23 seq 1922220018
        OPTS 4 SYN  WIN 4128
    !Connection timed out; remote host not responding
```

Configuring the TCP MSS Adjustment: Examples

Figure 1 Example Topology for TCP MSS Adjustment



The following example shows how to configure and verify the interface adjustment value. Configure the interface adjustment value on router B:

```
Router_B(config)# interface ethernet2/0
Router_B(config-if)# ip tcp adjust-mss 500
```

Telnet from router A to router C, with B having the MSS adjustment configured.

```
Router_A# telnet 192.168.1.1
Trying 192.168.1.1... Open
```

Observe the debug output from router C:

```
Router_C# debug ip tcp transactions

Sep 5 18:42:46.247: TCP0: state was LISTEN -> SYNRCVD [23 -> 10.0.1.1(38437)]
Sep 5 18:42:46.247: TCP: tcb 32290C0 connection to 10.0.1.1:38437, peer MSS 500, MSS is 
500
Sep 5 18:42:46.247: TCP: sending SYN, seq 580539401, ack 6015751
Sep 5 18:42:46.247: TCP0: Connection to 10.0.1.1:38437, advertising MSS 500
Sep 5 18:42:46.251: TCP0: state was SYNRCVD -> ESTAB [23 -> 10.0.1.1(38437)]
```

The MSS gets adjusted to 500 on Router_B as configured.

The following example shows the configuration of a PPPoE client with the MSS value set to 1452:

```
vpdn enable
no vpdn logging
!
vpdn-group 1
request-dialin
protocol pppoe
!
interface Ethernet0
 ip address 192.168.100.1.255.255.255.0
 ip tcp adjust-mss 1452
 ip nat inside
!
interface ATM0
 no ip address
 no atm ilmi-keepalive
 pvc 8/35
 pppoe client dial-pool-number 1
!
dsl equipment-type CPE
dsl operating-mode GSHDSL symmetric annex B
dsl linerate AUTO
!
interface Dialer1
 ip address negotiated
 ip mtu 1492
 ip nat outside
 encapsulation ppp
 dialer pool 1
 dialer-group 1
 ppp authentication pap callin
 ppp pap sent-username sohodyn password 7 141B1309000528
!
ip nat inside source list 101 Dialer1 overload
ip route 0.0.0.0.0.0.0.0 Dialer1
access-list permit ip 192.168.100.0.0.0.0.255 any
```

Configuring the TCP Application Flags Enhancement: Example

The following output shows the flags (status and option) displayed using the **show tcp** command.

```
Router# show tcp
.
.
.
Status Flags: passive open, active open, retransmission timeout
 App closed

Option Flags: vrf id set
IP Precedence value: 6
.
.
.
SRTT: 273 ms, RTTO: 490 ms, RTV: 217 ms, KRTT: 0 ms
minRTT: 0 ms, maxRTT: 300 ms, ACK hold: 200 ms
```

Displaying Addresses in IP Format: Example

The following example shows the IP activity by using the **numeric** keyword to display the addresses in IP format.

```
Router# show tcp brief numeric

TCB            Local Address          Foreign Address      (state)
6523A4FC       10.1.25.3.11000        10.1.25.3.23          ESTAB
65239A84       10.1.25.3.23           10.1.25.3.11000       ESTAB
653FCBBC       *.1723 *.* LISTEN
```

Additional References

The following sections provide references related to TCP.

Related Documents

Related Topic	Document Title
IP addressing and services configuration tasks	*Cisco IOS IP Addressing and Services Configuration Guide*
IP application services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference.*
Path MTU Discovery	*Configuring IP Services*
TCP security features	*TCP Out-of-Order Packet Support for Cisco IOS Firewall and Cisco IOS IPS* *Configuring TCP Intercept (Preventing Denial-of-Service Attacks)*
TCP Header Compression, Class-based TCP Header Compression	*Configuring Class-Based RTP and TCP Header Compression* *Configuring TCP Header Compression*
Troubleshooting TCP	*"Troubleshooting TCP/IP"* part of the *Internetwork Troubleshooting Handbook*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified.	—

MIBs

MIB	MIBs Link
CISCO-TCP-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 793	*Transmission Control Protocol*
RFC 1191	*Path MTU discovery*
RFC 1323	*TCP Extensions for High Performance*
RFC 2018	*TCP Selective Acknowledgment Options*
RFC 2581	*TCP Congestion Control*
RFC 3168	*The Addition of Explicit Congestion Notification (ECN) to IP*

RFC	Title
RFC 3782	*The NewReno Modification to TCP's Fast Recovery Algorithm*
RFC 4022	*Management Information Base for the Transmission Control Protocol (TCP)*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for TCP

Table 1 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 12.2(1) or a later release appear in the table.

For information on a feature in this technology that is not documented here, see the "Cisco IOS IP Application Services Features Roadmap" or the "FHRP Features Roadmap."

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for TCP

Feature Name	Releases	Feature Information
TCP Application Flags Enhancement	12.4(2)T, 12.2(31)SB2 Cisco IOS XE Release 2.1	The TCP Applications Flags Enhancement feature enables the user to display additional flags with reference to TCP applications. There are two types of flags: status and option. The status flags indicate the status of TCP connections; for example, retransmission timeouts, application closed, and synchronized (SYNC) handshakes for listen. The additional flags indicate the state of set options; for example, whether or not a virtual private network (VPN) routing and forwarding (VRF) identification is set, whether or not a user is idle, and whether or not a keepalive timer is running. The following sections contain information about this feature: • TCP Applications Flags Enhancement, page 7 • Verifying TCP Performance Parameters, page 10 • Configuring the TCP Application Flags Enhancement: Example, page 17 The following command was modified by this feature: **show tcp**.

Table 1 Feature Information for TCP (continued)

Feature Name	Releases	Feature Information
TCP Congestion Avoidance	12.3(7)T	The TCP Congestion Avoidance feature enables the monitoring of acknowledgment packets to the TCP sender when multiple packets are lost in a single window of data. Previously the sender would exit Fast-Recovery mode, wait for three or more duplicate acknowledgment packets before retransmitting the next unacknowledged packet, or wait for the retransmission timer to slow start. This could lead to performance issues. Implementation of RFC 2581 and RFC 3782 addresses the modifications to the Fast-Recovery algorithm that incorporates a response to partial acknowledgments received during Fast Recovery, improving performance in situations where multiple packets are lost in a single window of data. This feature is an enhancement to the existing Fast Recovery algorithm. There are no commands used to enable or disable this feature. To monitor the acknowledgment packets, the output of the **debug ip tcp transactions** command has been enhanced to show the following conditions: • TCP entering Fast Recovery mode. • Duplicate acknowledgments being received during Fast Recovery mode. • Partial acknowledgments being received. The following sections provide information about this feature: • TCP Congestion Avoidance, page 6 • Verifying TCP Performance Parameters, page 10 The following command was modified by this feature: **debug ip tcp transactions**.

Table 1 Feature Information for TCP (continued)

Feature Name	Releases	Feature Information
TCP Explicit Congestion Notification	12.3(7)T 12.2(31)SB2	The TCP Explicit Congestion Notification (ECN) feature provides a method for an intermediate router to notify the end hosts of impending network congestion. It also provides enhanced support for TCP sessions associated with applications that are sensitive to delay or packet loss including Telnet, web browsing, and transfer of audio and video data. The benefit of this feature is the reduction of delay and packet loss in data transmissions. The following sections provide information about this feature: • TCP Explicit Congestion Notification, page 6 • Configuring TCP Performance Parameters, page 7 • Verifying TCP Performance Parameters, page 10 • Verifying the Configuration of TCP ECN: Example, page 13 The following commands were introduced or modified by this feature: **debug ip tcp ecn**, **ip tcp ecn**, **show debugging**, **show tcp**.
TCP MIB for RFC 4022 Support	Cisco IOS XE Release 2.1	The TCP MIB for RFC 4022 Support feature introduces support for RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*. RFC 4022 is an incremental change of the TCP MIB to improve the manageability of TCP. To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs. There are no new or modified command for this feature.

Table 1 Feature Information for TCP (continued)

Feature Name	Releases	Feature Information
TCP MSS Adjust	12.2(4)T 12.2(8)T 12.2(28)SB 12.2(33)SRA 12.2(18)ZU2 12.2(33)SXH Cisco IOS XE Release 2.1	The TCP MSS Adjust feature enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments in the SYN bit set. In 12.2(4)T, this feature was introduced. In 12.2(8)T, the command that was introduced by this feature was changed from **ip adjust-mss** to **ip tcp adjust-mss**. In 12.2(28)SB and 12.2(33)SRA, this feature was enhanced to be configurable on subinterfaces. The following sections provide information about this feature: • TCP MSS Adjustment, page 6 • Configuring the MSS Value and MTU for Transient TCP SYN Packets, page 9 • Configuring the TCP MSS Adjustment: Examples, page 15 The following command was introduced by this feature: **ip tcp adjust-mss**.
TCP Show Extension	12.4(2)T 12.2(31)SB2 Cisco IOS XE Release 2.1	The TCP Show Extension feature introduces the capability to display addresses in IP format instead of hostname format and to display the virtual private network (VPN) routing and forwarding (VRF) table associated with the connection. The following sections contain information about this feature: • TCP Show Extension, page 7 • Verifying TCP Performance Parameters, page 10 • Displaying Addresses in IP Format: Example, page 17 The following command was modified by this feature: **show tcp brief**.

Table 1 Feature Information for TCP (continued)

Feature Name	Releases	Feature Information
TCP Window Scaling	12.2(8)T 12.2(31)SB2	The TCP Window Scaling feature adds support for the Window Scaling option in RFC 1323. A larger window size is recommended to improve TCP performance in network paths with large bandwidth, long-delay characteristics that are called Long Fat Networks (LFNs). This TCP Window Scaling enhancement provides that support. The following sections provide information about this feature: • TCP Window Scaling, page 5 • Configuring TCP Performance Parameters, page 7 • Verifying TCP Performance Parameters, page 10 The following commands were introduced or modified by this feature: **ip tcp window-size**.

Glossary

LFN—Long Fat Networks. Large bandwidth, long-delay networks where the throughput is high and the transmission distance is long. Networks with satellite connections are one example of an LFN. Satellite links always have high propagation delays and typically have high bandwidth.

TCP—Transmission Control Protocol. Connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring WCCP

First Published: August 21, 2007
Last Updated: July 11, 2008

The Web Cache Communication Protocol (WCCP) is a Cisco-developed content-routing technology that intercepts IP packets and redirects those packets to a destination other than that specified in the IP packet. Typically the packets are redirected from their destination web server on the Internet to a content engine that is local to the client. In some WCCP deployment scenarios, redirection of traffic may also be required from the web server to the client. WCCP enables you to integrate content engines into your network infrastructure.

Cisco IOS Release 12.1 and later releases allow the use of either WCCP Version 1 (WCCPv1) or Version 2 (WCCPv2).

The tasks in this document assume that you have already configured content engines on your network. For specific information on hardware and network planning associated with Cisco Content Engines and WCCP, see the Cisco Content Engines documentation at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/webscale/content/index.htm

Note Cisco Systems replaced the Cache Engine 500 series platforms with Content Engine Platforms in July 2001. Cache Engine Products were the Cache Engine 505, 550, 570, and 550-DS3. Content Engine Products are the Content Engine 507, 560, 590, and 7320.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for WCCP" section on page 29.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Contents

Prerequisites for WCCP

- To use WCCP, IP must be configured on the interface connected to the Internet and the interface must be connected to the Content Engine.
- The interface connected to the Content Engine must be a Fast Ethernet or Gigabit Ethernet interface.

Restrictions for WCCP

General

The following limitations apply to WCCPv1 and WCCPv2:

- WCCP works only with IPv4 networks.

WCCPv1

The following limitations apply to WCCPv1:

- WCCPv1 supports the redirection of HTTP (TCP port 80) traffic only.
- WCCPv1 does not allow multiple routers to be attached to a cluster of content engines.

WCCPv2

The following limitations apply to WCCPv2:

- WCCP works only with IPv4 networks.
- For routers servicing a multicast cluster, the Time To Live (TTL) value must be set at 15 or fewer.
- Service groups can comprise up to 32 content engines and 32 routers.
- All content engines in a cluster must be configured to communicate with all routers servicing the cluster.
- Multicast addresses must be from 224.0.0.0 to 239.255.255.255.

Layer 2 Forwarding and Return

The following limitations apply to WCCP Layer 2 Forwarding and Return:

- Layer 2 redirection requires that content engines be directly connected to an interface on each WCCP router. Unless multicast IP addresses are used, WCCP configuration of the content engine must reference the directly connected interface IP address of the WCCP router and not a loopback IP address or any other IP address configured on the WCCP router.

Cisco Catalyst 4500 Series Switches

The following limitations apply to Cisco Catalyst 4500 series switches:

- Catalyst 4500 series switches do not support WCCPv1.
- Up to eight service groups are supported at the same time on the same client interface.
- The Layer 2 (L2) rewrite forwarding method is supported, but generic route encapsulation (GRE) is not.
- Direct Layer 2 (L2) connectivity to content engines is required; Layer 3 (L3) connectivity of one or more hops away is not supported.
- Ternary content addressable memory (TCAM) friendly mask-based assignment is supported, but the hash bucket-based method is not.
- Redirect access control list (ACL) for WCCP on a client interface is not supported.
- Incoming traffic redirection on an interface is supported, but outgoing traffic re-direction is not.
- When TCAM space is exhausted, traffic is not redirected; it is forwarded normally.
- WCCP version 2 standard allows for support of up to 256 distinct masks. However, a Catalyst 4500 series switch only supports mask assignment table with a single mask.

Cisco Catalyst 6500 Series Switches

The following limitation apply to Cisco Catalyst 6500 series switches:

- With a Policy Feature Card 2 (PFC2), Release 12.2(17d)SXB and later releases support WCCP.
- With a PFC3, Release 12.2(18)SXD1 and later releases support WCCP.
- To use the WCCP Layer 2 PFC redirection feature, configure WCCP on the Catalyst 6500 series switch as described in this chapter and configure accelerated WCCP on the cache engine as described in the *Transparent Caching* document available at the following URL:

  http://www.cisco.com/en/US/docs/app_ntwk_services/waas/acns/v42/configuration/guide/transprt.html
- Cisco Application and Content Networking System (ACNS) software releases later than Release 4.2.2 support WCCP Layer 2 Policy Feature Card (PFC) redirection hardware acceleration.
- A content engine configured for mask assignment that tries to join a farm where the selected assignment method is hash remains out of the farm as long as the cache engine assignment method does not match that of the existing farm.
- With WCCP Layer 2 PFC redirection as the forwarding method for a service group, the packet counters in the **show ip wccp** *service-number* command output display flow counts instead of packet counts.

Catalyst 6500 Series Switches and Cisco 7600 Series Routers Access Control Lists

When WCCP is using the mask assignment, any redirect list is merged with the mask information from the appliance and the resulting merged access control list (ACL) is passed down to the Catalyst 6500 series switch or Cisco 7600 series router hardware.

The following restrictions apply to the redirect-list ACL:

- The ACL must be an IPV4 simple or extended ACL.

- The protocol must be IP, UDP, or TCP.
- Only individual source or destination port numbers may be specified; port ranges cannot be specified.
- The only valid matching criteria besides individual source or destination port numbers is **dscp** or **tos**.
- The use of **fragments**, **time-range**, **options** or any TCP flags is not permitted.

If the redirect ACL does not meet the above restrictions the system will log the following error message:

```
WCCP-3-BADACE: Service <service group>, invalid access-list entry (seq:<sequence>,
reason:<reason>)
```

WCCP continues to redirect packets, but the redirection is carried out in software (Netflow Switching) until the access list is adjusted.

Information About WCCP

To configure WCCP, you should understand the following concepts:

- Understanding WCCP, page 4
- Layer 2 Forwarding, Redirection and Return, page 5
- WCCP Mask Assignment, page 5
- Hardware Acceleration, page 6
- WCCPv1 Configuration, page 6
- WCCPv2 Configuration, page 7
- WCCP Bypass Packets, page 10
- WCCP Closed Services and Open Services, page 10
- WCCP Outbound ACL Check, page 10
- WCCP Service Groups, page 11

Understanding WCCP

WCCP uses Cisco Content Engines (or other content engines running WCCP) to localize web traffic patterns in the network, enabling content requests to be fulfilled locally. Traffic localization reduces transmission costs and download time.

WCCP enables Cisco IOS routing platforms to transparently redirect content requests. The main benefit of transparent redirection is that users need not configure their browsers to use a web proxy. Instead, they can use the target URL to request content, and have their requests automatically redirected to a content engine. The word “transparent” in this case means that the end user does not know that a requested file (such as a web page) came from the content engine instead of from the originally specified server.

When a content engine receives a request, it attempts to service it from its own local cache. If the requested information is not present, the content engine issues its own request to the originally targeted server to get the required information. When the content engine retrieves the requested information, it forwards it to the requesting client and caches it to fulfill future requests, thus maximizing download performance and substantially reducing transmission costs.

WCCP enables a series of content engines, called a content engine cluster, to provide content to a router or multiple routers. Network administrators can easily scale their content engines to handle heavy traffic loads through these clustering capabilities. Cisco clustering technology enables each cluster member to work in parallel, resulting in linear scalability. Clustering content engines greatly improves the scalability, redundancy, and availability of your caching solution. You can cluster up to 32 content engines to scale to your desired capacity.

Layer 2 Forwarding, Redirection and Return

WCCP uses either Generic Routing Encapsulation (GRE) or Layer 2 (L2) to redirect or return IP traffic. When WCCP forwards traffic via GRE, the redirected packets are encapsulated within a GRE header. The packets also have a WCCP redirect header. When WCCP forwards traffic using L2, the original MAC header of the IP packet is overwritten and replaced with the MAC header for the WCCP client.

Using L2 as a forwarding method allows direct forwarding to the content engine without further lookup. Layer 2 redirection requires that the router and content engines are directly connected, that is, on the same IP subnetwork.

When WCCP returns traffic via GRE, the returned packets are encapsulated within a GRE header. The destination IP address is the address of the router and the source address is the address of the WCCP client. When WCCP returns traffic via L2, the original IP packet is returned without any added header information. The router to which the packet is returned will recognize the source of the packet and prevent redirection.

The WCCP redirection method does not have to match the return method.

L2 forwarding, return, or redirection are typically used for hardware accelerated platforms. In Cisco IOS Release 12.4(20)T and later releases, L2 forwarding, return, and redirection can also be used for software switching platforms.

For content engines running Application and Content Networking System (ACNS) software, use the **wccp custom-web-cache** command with the **l2-redirect** keyword to configure L2 redirection. For content engines running Cisco Wide Area Application Services (WAAS) software, use the **wccp tcp-promiscuous** command with the **l2-redirect** keyword to configure L2 redirection.

For more information on Cisco ACNS commands used to configure Cisco Content Engines, see the *Cisco ACNS Software Command Reference*, Release 5.5, at the following URL:

http://www.cisco.com/en/US/docs/app_ntwk_services/waas/acns/v55/command/reference/55cref.html

For more information on WAAS commands used to configure Cisco Content Engines, see the *Cisco Wide Area Application Services Command Reference (Software Versions 4.0.1 and 4.0.3)* at the following URL:

http://www.cisco.com/en/US/docs/app_ntwk_services/waas/waas/v401_v403/command/reference/cmdref.html

WCCP Mask Assignment

The WCCP Mask Assignment feature enables mask assignment as the load-balancing method (instead of the default hash assignment method) for a WCCP service.

For content engines running Application and Content Networking System (ACNS) software, use the **wccp custom-web-cache** command with the **mask-assign** keywords to configure mask assignment. For content engines running Cisco Wide Area Application Services (WAAS) software, use the **wccp tcp-promiscuous** command with the **mask-assign** keyword to configure mask assignment.

For more information on Cisco ACNS commands used to configure Cisco Content Engines, see the *Cisco ACNS Software Command Reference*, Release 5.5, at the following URL:

http://www.cisco.com/en/US/docs/app_ntwk_services/waas/acns/v55/command/reference/55cref.html

For more information on WAAS commands used to configure Cisco Content Engines, see the *Cisco Wide Area Application Services Command Reference (Software Versions 4.0.1 and 4.0.3)* at the following URL:

http://www.cisco.com/univercd/cc/td/doc/product/webscale/waas/waas40/cmdref/index.htm

Hardware Acceleration

Catalyst 4500 series switches provide hardware acceleration for directly connected Cisco Content Engines.

Catalyst 6500 series switches and Cisco 7600 series routers provide WCCP Layer 2 Policy Feature Card (PFC) redirection hardware acceleration. Hardware acceleration allows Cisco Content Engines to perform a L2 MAC address rewrite redirection method when directly connected to a compatible switch or router.

Redirection processing is accelerated in the switching or routing hardware, which is more efficient than L3 redirection with Generic Routing Encapsulation (GRE). L2 redirection takes place on the switch or router, and is not visible to the Multilayer Switch Feature Card (MSFC). The WCCP L2 PFC redirection feature requires no configuration on the MSFC. The **show ip wccp** {*service-number* | **web-cache**} **detail** command displays which redirection method is in use for each content engine.

In order for the router or switch to make full use of hardware redirection, the content engine must be configured with L2 redirection and mask assignment as discussed in the "Layer 2 Forwarding, Redirection and Return" section on page 5.

Use the **ip wccp web-cache accelerated** command on hardware-based platforms to enforce the use of L2 redirection and mask assignment. Using this command configures the router to form a service group and redirect packets with an appliance only if the appliance is configured for L2 and mask assignment.

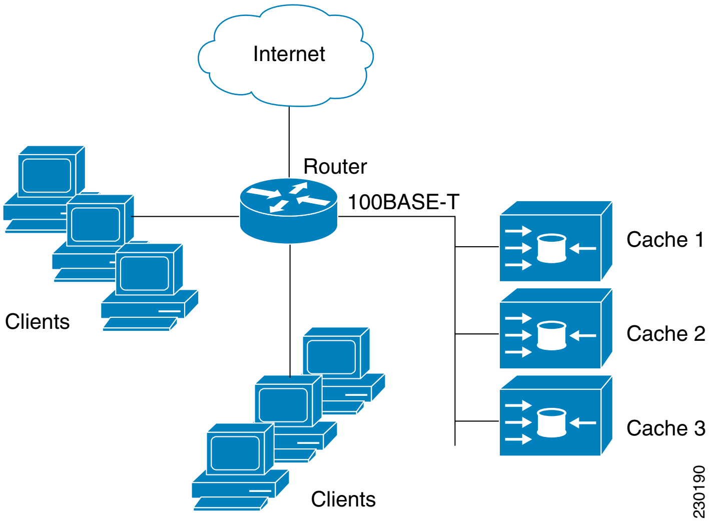
The following guidelines apply to WCCP Layer 2 PFC redirection:

- The WCCP Layer 2 PFC redirection feature sets the IP flow mask to full-flow mode.
- You can configure the Cisco Cache Engine software Release 2.2 or later releases to use the WCCP Layer 2 PFC redirection feature.
- L2 redirection takes place on the PFC and is not visible to the MSFC. The **show ip wccp** {*service-number* | **web-cache**} **detail** command on the MSFC displays statistics for only the first packet of a L2 redirected flow, which provides an indication of how many flows, rather than packets, are using L2 redirection. Entering the **show mls entries** command displays the other packets in the L2 redirected flows. The PFC3 provides hardware acceleration for GRE. If you use WCCP Layer 3 redirection with GRE, there is hardware support for encapsulation, but the PFC3 does not provide hardware support for decapsulation of WCCP GRE traffic.

WCCPv1 Configuration

With WCCPv1, only a single router services a cluster. In this scenario, this router is the device that performs all the IP packet redirection. Figure 1 illustrates the WCCPv1 configuration.

Figure 1 WCCPv1 Configuration



Content is not duplicated on the content engines. The benefit of using multiple content engines is that you can scale a caching solution by clustering multiple physical content engines to appear as one logical cache.

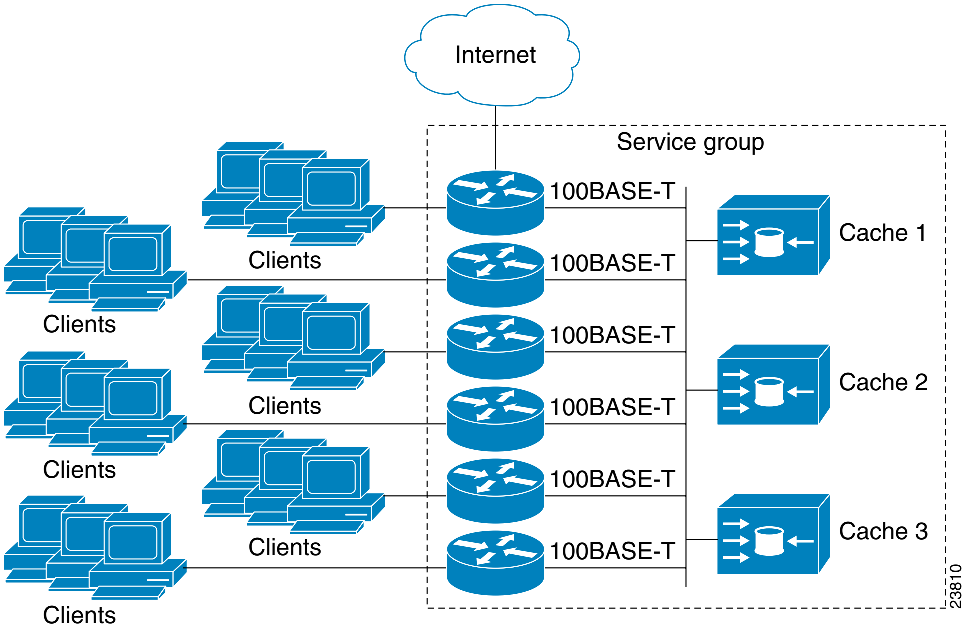
The following sequence of events details how WCCPv1 configuration works:

1. Each content engine is configured by the system administrator with the IP address of the control router. Up to 32 content engines can connect to a single control router.
2. The content engines send their IP addresses to the control router using WCCP, indicating their presence. Routers and content engines communicate to each other via a control channel; this channel is based on UDP port 2048.
3. This information is used by the control router to create a cluster view (a list of caches in the cluster). This view is sent to each content engine in the cluster, essentially making all the content engines aware of each other. A stable view is established after the membership of the cluster remains the same for a certain amount of time.
4. Once a stable view has been established, one content engine is elected as the lead content engine. (The lead is defined as the content engine seen by all the content engines in the cluster with the lowest IP address). This lead content engine uses WCCP to indicate to the control router how IP packet redirection should be performed. Specifically, the lead content engine designates how redirected traffic should be distributed across the content engines in the cluster.

WCCPv2 Configuration

Multiple routers can use WCCPv2 to service a content engine cluster. This configuration is in contrast to WCCPv1, in which only one router could redirect content requests to a cluster. Figure 2 illustrates a sample configuration using multiple routers.

Figure 2 Cisco Cache Engine Network Configuration Using WCCPv2



The subset of content engines within a cluster and routers connected to the cluster that are running the same service is known as a *service group*. Available services include TCP and User Datagram Protocol (UDP) redirection.

Using WCCPv1, the content engines were configured with the address of the single router. WCCPv2 requires that each content engine be aware of all the routers in the service group. To specify the addresses of all the routers in a service group, you must choose one of the following methods:

- Unicast—A list of router addresses for each of the routers in the group is configured on each content engine. In this case the address of each router in the group must be explicitly specified for each content engine during configuration.
- Multicast—A single multicast address is configured on each content engine. In the multicast address method, the content engine sends a single-address notification that provides coverage for all routers in the service group. For example, a content engine could indicate that packets should be sent to a multicast address of 224.0.0.100, which would send a multicast packet to all routers in the service group configured for group listening using WCCP (see the **ip wccp group-listen** interface configuration command for details).

The multicast option is easier to configure because you need only specify a single address on each content engine. This option also allows you to add and remove routers from a service group dynamically, without needing to reconfigure the content engines with a different list of addresses each time.

The following sequence of events details how WCCPv2 configuration works:

1. Each content engine is configured with a list of routers.
2. Each content engine announces its presence and a list of all routers with which it has established communications. The routers reply with their view (list) of content engines in the group.
3. Once the view is consistent across all content engines in the cluster, one content engine is designated as the lead and sets the policy that the routers need to deploy in redirecting packets.

Support for Services Other Than HTTP

WCCPv2 allows redirection of traffic other than HTTP (TCP port 80 traffic), including a variety of UDP and TCP traffic. WCCPv1 supported the redirection of HTTP (TCP port 80) traffic only. WCCPv2 supports the redirection of packets intended for other ports, including those used for proxy-web cache handling, File Transfer Protocol (FTP) caching, FTP proxy handling, web caching for ports other than 80, and Real Audio, video, and telephony applications.

To accommodate the various types of services available, WCCPv2 introduces the concept of multiple *service groups*. Service information is specified in the WCCP configuration commands using dynamic services identification numbers (such as 98) or a predefined service keyword (such as **web-cache**). This information is used to validate that service group members are all using or providing the same service.

The content engines in a service group specify traffic to be redirected by protocol (TCP or UDP) and up to eight source or destination ports. Each service group has a priority status assigned to it. The priority of a dynamic service is assigned by the content engine. The priority value is in the range of 0 to 255 where 0 is the lowest priority. The predefined web-cache service has an assigned priority of 240.

Support for Multiple Routers

WCCPv2 allows multiple routers to be attached to a cluster of cache engines. The use of multiple routers in a service group allows for redundancy, interface aggregation, and distribution of the redirection load. WCCPv2 supports up to 32 routers per service group. Each service group is established and maintained independently.

MD5 Security

WCCPv2 provides optional authentication that enables you to control which routers and content engines become part of the service group using passwords and the HMAC MD5 standard. Shared-secret MD5 one-time authentication (set using the **ip wccp** [**password** [**0-7**] *password*] global configuration command) enables messages to be protected against interception, inspection, and replay.

Web Cache Packet Return

If a content engine is unable to provide a requested object it has cached due to error or overload, the content engine will return the request to the router for onward transmission to the originally specified destination server. WCCPv2 provides a check on packets that determines which requests have been returned from the content engine unserviced. Using this information, the router can then forward the request to the originally targeted server (rather than attempting to resend the request to the content engine cluster). This process provides error handling transparency to clients.

Typical reasons why a content engine would reject packets and initiate the packet return feature include the following:

- Instances when the content engine is overloaded and has no room to service the packets
- Instances when the content engine is filtering for certain conditions that make caching packets counterproductive (for example, when IP authentication has been turned on)

Load Distribution

WCCPv2 can be used to adjust the load being offered to individual content engines to provide an effective use of the available resources while helping to ensure high quality of service (QoS) to the clients. WCCPv2 allows the designated content engine to adjust the load on a particular content engine and balance the load across the content engines in a cluster. WCCPv2 uses three techniques to perform load distribution:

- Hot Spot Handling—Allows an individual hash bucket to be distributed across all the content engines. Prior to WCCPv2, information from one hash bucket could only go to one content engine.

- Load Balancing—Allows the set of hash buckets assigned to a content engine to be adjusted so that the load can be shifted from an overwhelmed content engine to other members that have available capacity.
- Load Shedding—Enables the router to selectively redirect the load to avoid exceeding the capacity of a content engine.

The use of these hashing parameters prevents one content engine from being overloaded and reduces the potential for bottlenecking.

WCCP Bypass Packets

WCCP intercepts IP packets and redirects those packets to a destination other than the destination that is specified in the IP header. Typically the packets are redirected from a web server on the Internet to a web cache that is local to the destination.

Occasionally a web cache decides that it cannot deal with the redirected packets appropriately and returns the packets unchanged to the originating router. These packets are called bypass packets and are returned to the originating router using either Layer 2 forwarding without encapsulation (L2) or encapsulated in generic routing encapsulation (GRE). The router decapsulates and forwards the packets normally.

GRE is a tunnelling protocol developed by Cisco that encapsulates packet types from a variety of protocols inside IP tunnels, creating a virtual point-to-point link over an IP network.

WCCP Closed Services and Open Services

In applications where packet flows are intercepted and redirected by a Cisco IOS router to external WCCP client devices, it may be necessary to block the packet flows for the application when a WCCP client device is not available. This blocking is achieved by configuring a WCCP closed service. When a WCCP services is configured as closed, WCCP discards packets that do not have a WCCP client registered to receive the redirected traffic.

By default, WCCP operates as an open service, wherein communication between clients and servers proceeds normally in the absence of an intermediary device.

The **ip wccp service-list** command can only be used for closed-mode services. Use the **service-list** keyword and *service-access-list* argument to register an application protocol type or port number.

When there is a conflict in service list definitions, the configured definition takes precedence over the external definition received via WCCP protocol messages.

WCCP Outbound ACL Check

WCCP operates by intercepting IP packets and redirecting those packets to a destination other than the destination that is specified in the IP header. Typically the packets are redirected from a web server on the Internet to a web cache that is local to the redirecting router.

Access control lists (ACLs) filter network traffic by controlling whether routed packets are forwarded or blocked at the router interface. Each packet is examined to determine whether it will be forwarded or dropped, according to the specified criteria within the ACL. ACL criteria can be the source address of the traffic, the destination address of the traffic, or the upper-layer protocol. An IP ACL is a sequential collection of permit and deny conditions that apply to an IP address. The router tests addresses against the conditions in the ACL one at a time. The first match determines whether the address is accepted or rejected. Because Cisco IOS software stops testing conditions after the first match, the order of the conditions is critical. If no conditions match, the router rejects the address, by virtue of an implicit “deny all” clause.

If there is an outbound ACL configured on the interface at which redirection takes place, it is possible, under some circumstances, that hosts whose traffic is redirected will gain access to destinations to which they would otherwise be blocked.

The WCCP Outbound ACL Check feature ensures that the outbound ACL checking is performed at the original interface so that the checking is secure and consistent across all platforms and Cisco IOS switching paths.

WCCP Service Groups

WCCP is a component of Cisco IOS software that redirects traffic with defined characteristics from its original destination to an alternative destination. The typical application of WCCP is to redirect traffic bound for a remote web server to a local web cache to improve response time and optimize network resource usage.

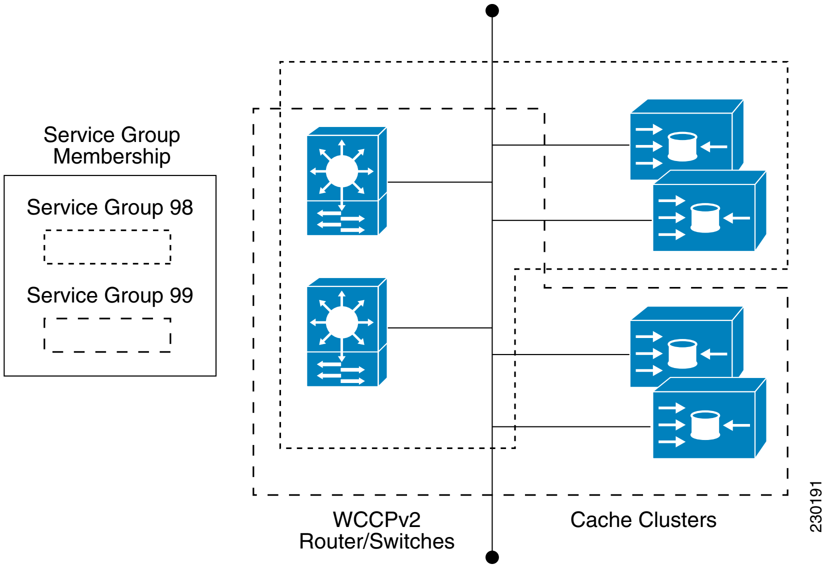
The nature of the selected traffic for redirection is defined by service groups specified on content engines and communicated to routers by using WCCP. The current implementation of WCCP in Cisco IOS releases prior to Cisco IOS Release 12.3(14)T allowed a maximum of eight service groups to be defined. This maximum restricted caching deployments. In Cisco IOS Release 12.3(14)T and later releases, the maximum number of service groups allowed is increased to 256.

WCCPv2 supports up to 32 routers per service group. Each service group is established and maintained independently.

WCCPv2 uses service groups based on logical redirection services, deployed for intercepting and redirecting traffic. The standard service is web cache, which intercepts TCP port 80 (HTTP) traffic and redirects that traffic to the content engines. This service is referred to as a *well-known service*, because the characteristics of the web cache service are known by both the router and content engines. A description of a well-known service is not required beyond a service identification. To specify the standard web cache service, use the **ip wccp** command with the **web-cache** keyword.

Note More than one service can run on a router at the same time, and routers and content engines can be part of multiple service groups at the same time.

Figure 3 WCCP Service Groups



The dynamic services are defined by the content engines; the content engine instructs the router which protocol or ports to intercept, and how to distribute the traffic. The router itself does not have information on the characteristics of the dynamic service group's traffic, because this information is provided by the first content engine to join the group. In a dynamic service, up to eight ports can be specified within a single protocol.

Cisco Content Engines, for example, use dynamic service 99 to specify a reverse-proxy service. However, other content engine devices may use this service number for some other service. The configuration information in this document deals with enabling general services on Cisco routers.

An interface may be configured with more than one WCCP service. When more than one WCCP service is configured on an interface, the precedence of a service depends on the relative priority of the service compared to the priority of the other configured services. Each WCCP service has a priority value as part of its definition. When an interface is configured with more than one WCCP service, the precedence of the packets are matched against service groups in priority order.

Note The priority of a WCCP service group cannot be configured via Cisco IOS software.

With the **ip wccp check services all** command, WCCP can be configured to check all configured services for a match and perform redirection for those services if appropriate. The caches to which packets are redirected can be controlled by a redirect ACL as well as by the service priority.

If no WCCP services are configured with a redirect ACL, the services are considered in priority order until a service is found that matches the IP packet. If no services match the packet, the packet is not redirected. If a service matches the packet and the service has a redirect ACL configured, then the IP packet will be checked against the ACL. If the packet is rejected by the ACL, the packet will not be passed down to lower priority services unless the **ip wccp check services all** command is configured. When the **ip wccp check services all** command is configured, WCCP will continue to attempt to match the packet against any remaining lower priority services configured on the interface.

How to Configure WCCP

The following configuration tasks assume that you have already installed and configured the content engines you want to include in your network. You must configure the content engines in the cluster before configuring WCCP functionality on your routers or switches. Refer to the *Cisco Cache Engine User Guide* for content engine configuration and setup tasks.

Perform these tasks to configure WCCP on a router or switch:

- Configuring WCCP, page 13 (required)
- Configuring Closed Services, page 14
- Registering a Router to a Multicast Address, page 16 (optional)
- Using Access Lists for a WCCP Service Group, page 17 (optional)
- Enabling the WCCP Outbound ACL Check, page 19 (optional)
- Verifying and Monitoring WCCP Configuration Settings, page 20 (optional)

Configuring WCCP

Perform this task to configure WCCP.

Until you configure a WCCP service using the **ip wccp** {**web-cache** | *service-number*} global configuration command, WCCP is disabled on the router. The first use of a form of the **ip wccp** command enables WCCP. By default WCCPv2 is used for services, but you can use WCCPv1 functionality instead. To change the running version of WCCP from Version 2 to Version 1, or to return to WCCPv2 after an initial change, use the **ip wccp version** command in global configuration mode.

If a function is not allowed in WCCPv1, an error prompt will be printed to the screen. For example, if WCCPv1 is running on the router and you try to configure a dynamic service, the following message will be displayed: “WCCP V1 only supports the web-cache service.” The **show ip wccp** EXEC command will display the WCCP protocol version number that is currently running on your router.

Using the **ip wccp web-cache password** command, you can set a password for a router and the content engines in a service group. MD5 password security requires that each router and content engine that wants to join a service group be configured with the service group password. The password can consist of up to eight characters. Each content engine or router in the service group will authenticate the security component in a received WCCP packet immediately after validating the WCCP message header. Packets failing authentication will be discarded.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp version** {**1** | **2**}
4. **ip wccp** {**web-cache** | *service-number*} [**group-address** *group-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password*]
5. **interface** *type number*
6. **ip wccp** {**web-cache** | *service-number*} **redirect** {**out** | **in**}
7. **ip wccp redirect exclude in**

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip wccp version`** {**1** | **2**}
	Example:
	`Router(config)# ip wccp version 2`
		Specifies which version of WCCP to configure on a router. WCCPv2 is the default running version.

Step 4	**`ip wccp`** {**web-cache** | *service-number*} [**group-address** *group-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password* [**0** | **7**]] **Example:** `Router(config)# ip wccp web-cache password password1`	Specifies a web-cache or dynamic service to enable on the router, specifies the IP multicast address used by the service group, specifies any access lists to use, specifies whether to use MD5 authentication, and enables the WCCP service.
Step 5	**`interface`** *type number* **Example:** `Router(config)# interface ethernet0/0`	Targets an interface number for which the web cache service will run, and enters interface configuration mode.

Step 6	**`ip wccp`** {**web-cache** | *service-number*} **redirect** {**out** | **in**}
	Example:
	`Router(config-if)# ip wccp web-cache redirect in`
		Enables packet redirection on an outbound or inbound interface using WCCP.
		As indicated by the **out** and **in** keyword options, redirection can be specified for outbound interfaces or inbound interfaces.

Step 7	**`ip wccp redirect exclude in`**
	Example:
	`Router(config-if)# ip wccp redirect exclude in`
		(Optional) Excludes traffic on the specified interface from redirection.

Configuring Closed Services

Perform this task to specify the number of service groups for WCCP, to configure a service group as a closed or open service, and to optionally specify a check of all services.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp** *service-number* [**service-list** *service-access-list*] **mode** {**open** | **closed**}

   or

ip wccp web-cache mode {**open** | **closed**}

4. **ip wccp check services all**
5. **ip wccp** {**web-cache** | *service-number*}
6. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip wccp`** *`service-number`* `[`**`service-list`** *`service-access-list`*`]` **`mode`** `{`**`open`** `|` **`closed`**`}`
	or
	`ip wccp web-cache mode` `{`**`open`** `|` **`closed`**`}`
	Example:
	`Router(config)# ip wccp 90 service-list 120 mode closed`
	or
	Example:
	`Router(config)# ip wccp web-cache mode closed`
		Configures a WCCP service as closed or open.
		Note When configuring the web-cache service as a closed service, you cannot specify a service access list.

Step 4	**`ip wccp check services all`**
	Example:
	`Router(config)# ip wccp check services all`
		(Optional) Enables a check of all WCCP services.
		With the **ip wccp check services all** command, WCCP can be configured to check the other configured services for a match and perform redirection for those services if appropriate. The caches to which packets are redirected can be controlled by the redirect ACL and not just the service description.
		Note The **ip wccp check services all** command is a global WCCP command that applies to all services and is not associated with a single service.

	Command or Action	Purpose

Step 5	**ip wccp** {**web-cache** | *service-number*}
	Example:
	`Router(config)# ip wccp 201`
		Specifies the WCCP service identifier. You can specify the standard web-cache service or a dynamic service number from 0 to 255.
		The maximum number of services that can be specified is 256.

Step 6	**exit** **Example:** `Router(config)# exit`	Exits to privileged EXEC mode.

Registering a Router to a Multicast Address

If you decide to use the multicast address option for your service group, you must configure the router to listen for the multicast broadcasts on an interface.

For network configurations where redirected traffic needs to traverse an intervening router, the router being traversed must be configured to perform IP multicast routing. You must configure the following two components to enable traversal over an intervening router:

- Enable IP multicast routing using the **ip multicast-routing** global configuration command.
- Enable the interfaces to which the cache engines will connect to receive multicast transmissions using the **ip wccp group-listen** interface configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing** [**vrf** *vrf-name*] [**distributed**]
4. **ip wccp** {**web-cache** | *service-number*} **group-address** *multicast-address*
5. **interface** *type number*
6. **ip pim** {**sparse-mode** | **sparse-dense-mode** | **dense-mode** [**proxy-register** {**list** *access-list* | **route-map** *map-name*}]}
7. **ip wccp** {**web-cache** | *service-number*} **group-listen**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command	Purpose
Step 3	**ip multicast-routing** [**vrf** *vrf-name*] [**distributed**] **Example:** `Router(config)# ip multicast-routing`	Enables IP multicast routing.
Step 4	**ip wccp** {**web-cache** | *service-number*} **group-address** *multicast-address* **Example:** `Router(config)# ip wccp 99 group-address 239.1.1.1`	Specifies the multicast address for the service group.
Step 5	**interface** *type number* **Example:** `Router(config)# interface ethernet0/0`	Enables the interfaces to which the content engines will connect to receive multicast transmissions for which the web cache service will run, and enters interface configuration mode.

Step 6	**ip pim** {**sparse-mode** | **sparse-dense-mode** | **dense-mode** [**proxy-register** {**list** *access-list* | **route-map** *map-name*}]}
	Example:
	`Router(config-if)# ip pim dense-mode`
		(Optional) Enables Protocol Independent Multicast (PIM) on an interface.
		Note To ensure correct operation of the **ip wccp group-listen** command on Catalyst 6500 series switches and Cisco 7600 series routers, you must enter the **ip pim** command in addition to the **ip wccp group-listen** command.

Step 7	**ip wccp** {**web-cache** | *service-number*} **group-listen** **Example:** `Router(config-if)# ip wccp 99 group-listen`	Configures an interface to enable or disable the reception of IP multicast packets for WCCP.

What to Do Next

For more information about configuring IP Multicast features, see the *Cisco IOS IP Multicast Configuration Guide*.

Using Access Lists for a WCCP Service Group

Perform this task to configure the router to use an access list to determine which traffic should be directed to which content engines.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **remark** *remark*
4. **access-list** *access-list-number* **permit** {*source* [*source-wildcard*] | **any**} [**log**]
5. **access-list** *access-list-number* **remark** *remark*
6. **access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} [**log**]

7. Repeat some combination of Steps 3 through 6 until you have specified the sources on which you want to base your access list.
8. **ip wccp web-cache group-list** *access-list*
9. **ip wccp web-cache redirect-list** *access-list*

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**access-list** *access-list-number* **remark** *remark* **Example:** `Router(config)# access-list 1 remark Give access to user1`	(Optional) Adds a user-friendly comment about an access list entry. • A remark of up to 100 characters can precede or follow an access list entry.
Step 4	**access-list** *access-list-number* **permit** {*source* [*source-wildcard*] | **any**} [**log**] **Example:** `Router(config)# access-list 1 permit 172.16.5.22 0.0.0.0`	Creates an access list that enables or disables traffic redirection to the cache engine. Permits the specified source based on a source address and wildcard mask. • Every access list needs at least one permit statement; it need not be the first entry. • Standard IP access lists are numbered 1 to 99 or 1300 to 1999. • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the keyword **any** as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255. • In this example, host 172.16.5.22 is allowed to pass the access list.
Step 5	**access-list** *access-list-number* **remark** *remark* **Example:** `Router(config)# access-list 1 remark Give access to user1`	(Optional) Adds a user-friendly comment about an access list entry. • A remark of up to 100 characters can precede or follow an access list entry.

	Command	Purpose
Step 6	**access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} [**log**] **Example:** `Router(config)# access-list 1 deny 172.16.7.34 0.0.0.0`	Denies the specified source based on a source address and wildcard mask. • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the abbreviation any as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255. • In this example, host 172.16.7.34 is denied passing the access list.
Step 7	Repeat some combination of Steps 3 through 6 until you have specified the sources on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit **deny** statement at the end of the access list.
Step 8	**ip wccp web-cache group-list** *access-list* **Example:** `Router(config) ip wccp web-cache group-list 1`	Indicates to the router from which IP addresses of content engines to accept packets.
Step 9	**ip wccp web-cache redirect-list** *access-list* **Example:** `Router(config)# ip wccp web-cache redirect-list 1`	(Optional) Disables caching for certain clients.

What to Do Next

For more information about configuring and using IP access lists, see "IP Access List Features Roadmap" in the *Cisco IOS Security Configuration Guide*.

Enabling the WCCP Outbound ACL Check

Perform this task to enable an outbound ACL check for WCCP.

Note When all redirection is performed in the hardware, the mode of redirection will change when outbound ACL checking is enabled. The first packet is switched in software to allow the extra ACL check to be performed before a shortcut is installed.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password*]
4. **ip wccp check acl outbound**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip wccp`** `{`**`web-cache`** `|` *`service-number`*`}` `[`**`group-address`** *`multicast-address`*`]` `[`**`redirect-list`** *`access-list`*`]` `[`**`group-list`** *`access-list`*`]` `[`**`password`** *`password`*`]`
	Example:
	`Router(config)# ip wccp web-cache`
		Enables the support for a Cisco content engine service group or any content engine service group and configures a redirect ACL list or group ACL.
		Note The **web-cache** keyword is for WCCP version 1 and version 2 and the *service-number* argument is for WCCP version 2 only.

Step 4	**`ip wccp check acl outbound`**
	Example:
	`Router(config)# ip wccp check acl outbound`
		Enables the ACL outbound check on the originating interface.

Step 5	**`exit`**
	Example:
	`Router(config)# exit`
		Exits global configuration.

Verifying and Monitoring WCCP Configuration Settings

Use the following commands in EXEC mode to verify and monitor the configuration settings for WCCP.

SUMMARY STEPS

1. **enable**
2. **show ip wccp** [*service-number* | **web-cache**] [**detail** | **view**]
3. **show ip interface**
4. **more system:running-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`show ip wccp`** [*`service-number`* | **`web-cache`**] [**`detail`** | **`view`**] **Example:** `Router# show ip wccp 24 detail`	Displays global information related to WCCP, including the protocol version currently running, the number of content engines in the router service group, which content engine group is allowed to connect to the router, and which access list is being used. The argument and keywords are as follows: • *service-number*—(Optional) Dynamic number of the web-cache service group being controlled by the content engine. The range is from 0 to 99. For web caches that use Cisco Content Engines, the reverse proxy service is indicated by a value of 99. • **web-cache**—(Optional) Statistics for the web-cache service. • **detail**—(Optional) Other members of a particular service group or web cache that have or have not been detected. • **view**—(Optional) Information about a router or all web caches.

Step 3	**`show ip interface`**
	Example:
	`Router# show ip interface`
		Displays status about whether any **ip wccp redirection** commands are configured on an interface. For example, "Web Cache Redirect is enabled / disabled."

Step 4	**`more system:running-config`**
	Example:
	`Router# more system:running-config`
		(Optional) Displays contents of the currently running configuration file (equivalent to the **show running-config** command.)

Troubleshooting Tips

Problems have been encountered because CPU usage is very high when WCCP is enabled. The counters enable a determination of the bypass traffic directly on the router and can indicate whether or not this is the cause. In some situations, 10 percent bypass traffic may be normal; in other situations, it may be high. However, any figure above 25 percent should prompt a closer investigation of what is occurring in the web cache.

If the counters suggest that the level of bypass traffic is high, the next step is to examine the bypass counters in the content engine and determine why the content engine is choosing to bypass the traffic. You can log in to the content engine console and use CLI to investigate further. The counters allow you to determine the percent of traffic being bypassed.

Configuration Examples for WCCP

This section provides the following configuration examples:

Changing the Version of WCCP on a Router: Example

The following example shows how to change the WCCP version from the default of WCCPv2 to WCCPv1, and enabling the web-cache service in WCCPv1:

```
show ip wccp
% WCCP version 2 is not enabled
configure terminal
 ip wccp version 1
 end
show ip wccp
% WCCP version 1 is not enabled
configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
 ip wccp web-cache
 end
show ip wccp
Global WCCP information:
    Router information:
        Router Identifier:                   10.4.9.8
        Protocol Version:                    1.0
.
.
.
```

Configuring a General WCCPv2 Session: Example

The following example shows how to configure a general WCCPv2 session:

```
configure terminal
 ip wccp web-cache group-address 224.1.1.100 password password1
 interface ethernet0
 ip wccp web-cache redirect out
 exit
 ip wccp check services all ! Configures a check of all WCCP services.
```

Setting a Password for a Router and Content Engines: Example

The following example shows how to configure a WCCPv2 password where the password is password1:

```
configure terminal
 ip wccp web-cache password password1
```

Configuring a Web Cache Service: Example

The following example shows how to configure a web cache service:

```
configure terminal
 ip wccp web-cache
 interface ethernet 0
 ip wccp web-cache redirect out
 exit
copy running-config startup-config
```

The following example shows how to configure a session in which redirection of HTTP traffic arriving on Ethernet interface 0/1 is enabled:

```
configure terminal
 interface ethernet 0/1
 ip wccp web-cache redirect in
 exit
show ip interface ethernet 0/1
.
.
.
WCCP Redirect inbound is enabled
WCCP Redirect exclude is disabled
.
.
.
```

Running a Reverse Proxy Service: Example

The following example assumes you are configuring a service group using Cisco Cache Engines, which use dynamic service 99 to run a reverse proxy service:

```
configure terminal
 ip wccp 99
 interface ethernet 0
 ip wccp 99 redirect out
```

Registering a Router to a Multicast Address: Example

The following example shows how to register a router to a multicast address of 224.1.1.100:

```
ip wccp web-cache group-address 224.1.1.100
interface ethernet 0
ip wccp web cache group-listen
```

The following example shows a router configured to run a reverse proxy service, using the multicast address of 224.1.1.1. Redirection applies to packets outgoing via interface Ethernet interface 0:

```
ip wccp 99 group-address 224.1.1.1
interface ethernet 0
ip wccp 99 redirect out
```

Using Access Lists: Example

To achieve better security, you can use a standard access list to notify the router which IP addresses are valid addresses for a content engine attempting to register with the current router. The following example shows a standard access list configuration session where the access list number is 10 for some sample hosts:

```
access-list 10 permit host 11.1.1.1
access-list 10 permit host 11.1.1.2
access-list 10 permit host 11.1.1.3
ip wccp web-cache group-list 10
```

To disable caching for certain clients, servers, or client/server pairs, you can use WCCP access lists. The following example shows that any requests coming from 10.1.1.1 to 12.1.1.1 will bypass the cache, and that all other requests will be serviced normally:

```
ip wccp web-cache redirect-list 120
access-list 120 deny tcp host 10.1.1.1 any
access-list 120 deny tcp any host 12.1.1.1
access-list 120 permit ip any any
```

The following example configures a router to redirect web-related packets received via interface ethernet 0/1, destined to any host except 209.165.200.224:

```
access-list 100 deny ip any host 209.165.200.224
access-list 100 permit ip any any
ip wccp web-cache redirect-list 100
interface Ethernet 0/1
ip wccp web-cache redirect in
```

WCCP Outbound ACL Check Configuration: Example

The following configuration example shows that the access list prevents traffic from network 10.0.0.0 leaving Fast Ethernet interface 0/0. Because the outbound ACL check is enabled, WCCP does not redirect that traffic. WCCP checks packets against the ACL before they are redirected.

```
ip wccp web-cache
ip wccp check acl outbound
interface fastethernet0/0
ip access-group 10 out
exit
ip wccp web-cache redirect-list redirect-out
access-list 10 deny 10.0.0.0 0.255.255.255
access-list 10 permit any
```

If the outbound ACL check is disabled, the HTTP packets from network 10.0.0.0 would be redirected to a web cache. Users with that network address could retrieve web pages even though the network administrator wanted to prevent it.

Verifying WCCP Settings: Examples

The following example shows how to verify your configuration changes by using the **more system:running-config** command in privileged EXEC mode. The following example shows that both the web cache service and dynamic service 99 are enabled on the router:

```
Router# more system:running-config

    Building configuration...
    Current configuration:
    !
    version 12.0
    service timestamps debug uptime
    service timestamps log uptime
    no service password-encryption
    service udp-small-servers
    service tcp-small-servers
    !
    hostname router4
    !
    enable secret 5 $1$nSVy$faliJsVQXVPW.KuCxZNTh1
    enable password password1
    !
    ip subnet-zero
    ip wccp web-cache
    ip wccp 99
    ip domain-name cisco.com
    ip name-server 10.1.1.1
    ip name-server 10.1.1.2
    ip name-server 10.1.1.3
    !
    !
    !
    interface Ethernet0
    ip address 10.3.1.2 255.255.255.0
    no ip directed-broadcast
    ip wccp web-cache redirect out
    ip wccp 99 redirect out
    no ip route-cache
    no ip mroute-cache
    !

    interface Ethernet1
    ip address 10.4.1.1 255.255.255.0
    no ip directed-broadcast
    ip wccp 99 redirect out
    no ip route-cache
    no ip mroute-cache
    !
    interface Serial0
    no ip address
    no ip directed-broadcast
    no ip route-cache
    no ip mroute-cache
    shutdown
    !
    interface Serial1
    no ip address
    no ip directed-broadcast
    no ip route-cache
    no ip mroute-cache
    shutdown
    !
    ip default-gateway 10.3.1.1
    ip classless
    ip route 0.0.0.0 0.0.0.0 10.3.1.1
    no ip http server
    !
    !
    !
    line con 0
    transport input none
    line aux 0
    transport input all
    line vty 0 4
    password alaska1
    login
    !
    end
```

The following example shows how to display information about bypassed packets for process, fast, and CEF that are switching paths in Cisco IOS.

```
Router# show ip wccp web-cache detail

WCCP Client information:
 Web Client ID:          10.10.10.1
 Protocol Version:      2.0
 State:                 Usable
 Initial Hash Info:     00000000000000000000000000000000
                        00000000000000000000000000000000
 Assigned Hash Info:    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
                        FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
 Hash Allotment:        256 (100.00%)
 Packets Redirected:    4320
 Connect Time:          00:04:53
 Bypassed Packets
 Process:               0
 Fast:                  0
 CEF:                   250
```

For more information about the **show ip wccp web-cache** command, see the *Cisco IOS IP Application Services Command Reference*.

Additional References

The following sections provide references related to WCCP.

Related Documents

Related Topic	Document Title
Cisco ACNS software configuration information	- *Cisco ACNS Software Caching Configuration Guide*, Release 4.2 - Cisco ACNS Software listing page on Cisco.com
IP Access List overview, configuration tasks, and commands	- *IP Access List Features Roadmap* - *Cisco IOS Security Command Reference*
IP addressing and services commands and configuration tasks	- *Cisco IOS IP Addressing Services Configuration Guide* - *Cisco IOS IP Addressing Services Command Reference*
WCCP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference*

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for WCCP

Table 1 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 12.2(1) or a later release appear in the table.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for WCCP

Feature Name	Releases	Feature Information
WCCP Bypass Counters	12.3(7)T 12.2(25)S	The WCCP Bypass Counters feature allows you to display a count of packets that have been bypassed by a web cache and returned to the originating router to be forwarded normally. The following sections provide information about this feature: • WCCP Bypass Packets, page 10 • Verifying WCCP Settings: Examples, page 25 The **show ip wccp** command was modified by this feature.
WCCP Closed Services	12.4(11)T	The WCCP Closed Services feature permits WCCP services to be configured so that WCCP always intercepts traffic for such services but, if no WCCP client (such as a content engine) has registered to receive this traffic, packets are discarded. This new behavior supports AONS (Application-Oriented Network Services) applications, which require traffic to be transparently intercepted using WCCP but do not want the packets to be forwarded to their destination if the WCCP client is unavailable to perform its processing. (This is contrary to the traditional use of WCCP to assist caches where the absence of a cache does not change the behavior as observed by the user.) • WCCP Closed Services and Open Services, page 10 • Configuring Closed Services, page 14 The **ip wccp** command was modified by this feature.

Table 1 Feature Information for WCCP (continued)

Feature Name	Releases	Feature Information
WCCP Increased Services	12.3(14)T 12.2(33)SRA 12.2(33)SXH	The WCCP Increased Services feature increases the number of services supported by WCCP to a maximum of 256. The following sections provide information about this feature: • WCCP Service Groups, page 11 • Configuring Closed Services, page 14 • Configuring WCCP, page 13 • Verifying WCCP Settings: Examples, page 25 The following commands were modified by this feature: **ip wccp**, **ip wccp check services all**, **show ip wccp**.
WCCP Layer 2 Redirection / Forwarding	12.4(20)T	The WCCP Layer 2 Redirection/Forwarding feature allows directly connected Cisco Content Engines to use Layer 2 redirection, which is more efficient than Layer 3 redirection via GRE encapsulation. You can configure a directly connected Cache Engine to negotiate use of the WCCP Layer 2 Redirection/Forwarding feature. The WCCP Layer 2 Redirection/Forwarding feature requires no configuration on the router or switch. The following sections provide information about this feature: • Restrictions for WCCP, page 2 • Layer 2 Forwarding, Redirection and Return, page 5 • WCCPv2 Configuration, page 7 There are no new or modified commands associated with this feature.
WCCP L2 Return	12.4(20)T	The WCCP L2 Return feature allows content engines to return packets to WCCP routers directly connected at Layer 2 by swapping the source and destination MAC addresses rather than tunnelling packets back to the router inside a Layer 3 GRE tunnel. The following sections provide information about this feature: • Layer 2 Forwarding, Redirection and Return, page 5 There are no new or modified commands associated with this feature.

Table 1 Feature Information for WCCP (continued)

Feature Name	Releases	Feature Information
WCCP Mask Assignment	12.4(20)T	The WCCP Mask Assignment feature introduces support for ACNS/WAAS devices using mask assignment as a cache engine assignment method. The following section provides information about this feature: • WCCP Mask Assignment, page 5 There are no new or modified commands associated with this feature.
WCCP Outbound ACL Check	12.3(7)T 12.2(25)S	The WCCP Outbound ACL Check feature enables you to ensure that traffic redirected by WCCP at an input interface is subjected to the outbound ACL checks that may be configured on the output interface prior to redirection. This feature is supported by Web Cache Communication Protocol (WCCP) Version 1 and Version 2. The following sections provide information about this feature: • WCCP Outbound ACL Check, page 10 • Enabling the WCCP Outbound ACL Check, page 19 • WCCP Outbound ACL Check Configuration: Example, page 24 The **ip wccp** command was modified by this feature.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

First Hop Redundancy Protocols

FHRP Features Roadmap

First Published: May 2, 2005
Last Updated: July 11, 2008

This feature roadmap lists the Cisco IOS features documented in the First Hop Redundancy Protocol (FHRP) modules in the *Cisco IOS IP Application Services Configuration Guide* and maps them to the documents in which they appear. The roadmap is organized so that you can select your release train and see the features in that release. Find the feature name you are searching for and click on the URL in the "Where Documented" column to access the document containing that feature.

Feature and Release Support

Table 1 lists FHRP feature support for the following Cisco IOS software release trains:

- Cisco IOS Release 12.2S
- Cisco IOS Release 12.2SB
- Cisco IOS Release 12.2SR
- Cisco IOS Release 12.2SX
- Cisco IOS Releases 12.2T, 12.3, and 12.3T
- Cisco IOS Release 12.4T
- Cisco IOS XE Release 2
- Other Cisco IOS Releases

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 lists the most recent release of each software train first and the features in alphabetical order within the release.

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Table 1 Supported FHRP Features

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Release 12.2S			
12.2(25)S	Enhanced Tracking Support	The Enhanced Tracking Support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.	*Configuring Enhanced Object Tracking*
	FHRP—Enhanced Object Tracking of IP SLAs Operations	This feature enables FHRPs and other Enhanced Object Tracking (EOT) clients to track the output from IP SLAs objects and use the provided information to trigger an action.	*Configuring Enhanced Object Tracking*
	HSRP MD5 Authentication	The HSRP MD5 Authentication feature is an enhancement to generate an MD5 digest for the HSRP portion of the multicast HSRP protocol packet. This feature provides added security and protects against the threat from HSRP-spoofing software.	*Configuring HSRP*
	HSRP Version 2	The HSRP Version 2 feature was introduced to prepare for further enhancements and to expand the capabilities beyond what is possible with HSRP version 1. HSRP version 2 has a different packet format than HSRP version 1.	*Configuring HSRP*
	SSO—HSRP	SSO HSRP alters the behavior of HSRP when a router with redundant Route Processors (RPs) is configured for Stateful Switchover (SSO). When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.	*Configuring HSRP*
12.2(18)S	GLBP MD5 Authentication	MD5 authentication provides greater security than the alternative plain text authentication scheme.	*Configuring GLBP*
12.2(14)S	Gateway Load Balancing Protocol	The Gateway Load Balancing Protocol (GLBP) protects data traffic from a failed router or circuit, like Hot Standby Router Protocol (HSRP) and Virtual Router Redundancy Protocol (VRRP), while allowing packet load sharing between a group of redundant routers.	*Configuring GLBP*
	Virtual Router Redundancy Protocol	VRRP enables a group of routers to form a single virtual router to provide redundancy. The LAN clients can then be configured with the virtual router as their default gateway. The virtual router, representing a group of routers, is also known as a VRRP group.	*Configuring VRRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Release 12.2SB			
12.2(31) SB2	FHRP—Object Tracking List	This feature enhances the tracking capabilities to enable the configuration of a combination of tracked objects in a list, and a flexible method of combining objects using Boolean logic.	*Configuring Enhanced Object Tracking*
	ISSU—GLBP	GLBP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in SSO mode even when different versions of Cisco IOS software are running on the active and standby RPs or line cards.	*Configuring GLBP*
	SSO—GLBP	GLBP is now SSO aware. GLBP can detect when a router is failing over to the secondary RP and continue in its current GLBP group state.	*Configuring GLBP*
12.2(28)SB	Enhanced Tracking Support	The Enhanced Tracking Support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.	*Configuring Enhanced Object Tracking*
	HSRP Support for MPLS VPNs	HSRP support for a Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) interface is useful when an Ethernet LAN is connected between two provider edge (PE) routers with either of the following conditions:	*Configuring HSRP*
Cisco IOS Release 12.2SR			
12.2(33) SRC	FHRP—HSRP Group Shutdown	The FHRP—HSRP Group Shutdown feature enables you to configure an HSRP group to become disabled (its state changed to Init) instead of having its priority decremented when a tracked object goes down.	*Configuring HSRP*
	ICMP Router Discovery Protocol	The ICMP Router Discovery Protocol (IRDP) allows IPv4 hosts to locate routers that provide IPv4 connectivity to other (non-local) IP networks.	*Configuring IRDP*
	ISSU—VRRP	VRRP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in Stateful Switchover (SSO) mode even when different versions of Cisco IOS software are running on the active and standby Route Processors (RPs) or line cards.	*Configuring VRRP*
	SSO—VRRP	VRRP is now SSO aware. VRRP can detect when a router is failing over to the secondary RP and continue in its current VRRP group state.	*Configuring VRRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(33) SRB1	ISSU—GLBP	GLBP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in SSO mode even when different versions of Cisco IOS software are running on the active and standby RPs or line cards.	*Configuring GLBP*
	HSRP—ISSU	The HSRP—ISSU feature enables support for ISSU in HSRP. The ISSU process allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues.	*Configuring HSRP*
12.2(33) SRB	FHRP—HSRP Multiple Group Optimization	HSRP Multiple Group Optimization improves the negotiation and maintenance of multiple HSRP groups configured on a subinterface. Only one HSRP group is required on a physical interface for the purposes of electing active and standby routers. This group is known as the *master* group. Other HSRP groups may be created on each subinterface and linked to the master group via the group name. These linked HSRP groups are known as *client* or *follow* groups.	*Configuring HSRP*
	FHRP—HSRP Support for IPv6	Support for IPv6 was added. For more information see the "Configuring First Hop Redundancy Protocols" section of the *Cisco IOS IPv6 Configuration Guide*, Release 12.4T.	*Configuring HSRP*
	FHRP—Integration of Embedded Event Manager with Enhanced Object Tracking	EOT is now integrated with Embedded Event Manager (EEM) to allow EEM to report on a status change of a tracked object and to allow EOT to track EEM objects.	*Configuring Enhanced Object Tracking*
	SSO—GLBP	GLBP is now SSO aware. GLBP can detect when a router is failing over to the secondary RP and continue in its current GLBP group state.	*Configuring GLBP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.2(33) SRA	Enhanced Tracking Support	The Enhanced Tracking Support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.	*Configuring Enhanced Object Tracking*
	FHRP—Enhanced Object Tracking of IP SLAs Operations	This feature enables FHRPs and other EOT clients to track the output from IP SLAs objects and use the provided information to trigger an action.	*Configuring Enhanced Object Tracking*
	FHRP—Object Tracking List	This feature enhances the tracking capabilities to enable the configuration of a combination of tracked objects in a list, and a flexible method of combining objects using Boolean logic.	*Configuring Enhanced Object Tracking*
	HSRP MD5 Authentication	The HSRP MD5 Authentication feature is an enhancement to generate an MD5 digest for the HSRP portion of the multicast HSRP protocol packet. This feature provides added security and protects against the threat from HSRP-spoofing software.	*Configuring HSRP*
	SSO—HSRP	SSO HSRP alters the behavior of HSRP when a router with redundant RPs is configured for SSO. When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.	*Configuring HSRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Release 12.2SX			
12.2(33) SXH	Enhanced Tracking Support	The Enhanced Tracking support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.	*Configuring Enhanced Object Tracking*
	FHRP—Enhanced Object Tracking of IP SLAs Operations	This feature enables FHRPs and other EOT clients to track the output from IP SLAs objects and use the provided information to trigger an action.	*Configuring Enhanced Object Tracking*
	FHRP—Object Tracking List	This feature enhances the tracking capabilities to enable the configuration of a combination of tracked objects in a list, and a flexible method of combining objects using Boolean logic.	*Configuring Enhanced Object Tracking*
	GLBP MD5 Authentication	MD5 authentication provides greater security than the alternative plain text authentication scheme.	*Configuring GLBP*
	HSRP MD5 Authentication	The HSRP MD5 Authentication feature is an enhancement to generate an MD5 digest for the HSRP portion of the multicast HSRP protocol packet. This feature provides added security and protects against the threat from HSRP-spoofing software.	*Configuring HSRP*
	SSO—GLBP	GLBP is now SSO aware. GLBP can detect when a router is failing over to the secondary RP and continue in its current GLBP group state.	*Configuring GLBP*
	SSO—HSRP	SSO HSRP alters the behavior of HSRP when a router with redundant Route Processors (RPs) is configured for Stateful Switchover (SSO). When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.	*Configuring HSRP*
Cisco IOS Releases 12.2T, 12.3, and 12.3T			
12.3(14)T	FHRP—VRRP Enhancements	The FHRP—VRRP Enhancements feature adds support for the following capabilities: • MD5 Authentication—Added to routers that are configured for VRRP, similar to HSRP, to provide a method of authenticating peers using a more simple method than the method in RFC 2338. • Bridged Virtual Interface (BVI)—Added the capability to configure VRRP on BVIs. This functionality is similar to the existing HSRP support for BVIs.	*Configuring VRRP*
12.3(11)T	VRRP MIB—RFC 2787	This feature enables an enhancement to the MIB for use with SNMP-based network management. The feature adds support for configuring, monitoring, and controlling routers that use VRRP.	*Configuring VRRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.3(8)T	FHRP—Object Tracking List	This feature enhances the tracking capabilities to enable the configuration of a combination of tracked objects in a list, and a flexible method of combining objects using Boolean logic.	*Configuring Enhanced Object Tracking*
12.3(4)T	FHRP—Enhanced Object Tracking of IP SLAs Operations	This feature enables FHRPs and other EOT clients to track the output from IP SLAs objects and use the provided information to trigger an action.	*Configuring Enhanced Object Tracking*
	HSRP Version 2	The HSRP Version 2 feature was introduced to prepare for further enhancements and to expand the capabilities beyond what is possible with HSRP version 1. HSRP version 2 has a different packet format than HSRP version 1.	*Configuring HSRP*
12.3(2)T	GLBP MD5 Authentication	MD5 authentication provides greater security than the alternative plain text authentication scheme.	*Configuring GLBP*
	HSRP MD5 Authentication	The HSRP MD5 Authentication feature is an enhancement to generate an MD5 digest for the HSRP portion of the multicast HSRP protocol packet. This feature provides added security and protects against the threat from HSRP-spoofing software.	*Configuring HSRP*
	VRRP Object Tracking	VRRP object tracking extends the capabilities of the VRRP to allow tracking of specific objects within the router that can alter the priority level of a virtual router for a VRRP group.	*Configuring VRRP*
12.2(15)T	Enhanced Tracking Support	The Enhanced Tracking support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.	*Configuring Enhanced Object Tracking*
	Gateway Load Balancing Protocol	GLBP protects data traffic from a failed router or circuit, like HSRP and VRRP, while allowing packet load sharing between a group of redundant routers.	*Configuring GLBP*
12.2(13)T	Virtual Router Redundancy Protocol	VRRP enables a group of routers to form a single virtual router to provide redundancy. The LAN clients can then be configured with the virtual router as their default gateway. The virtual router, representing a group of routers, is also known as a VRRP group.	*Configuring VRRP*
12.2(8)T	HSRP Support for MPLS VPNs	HSRP support for a Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) interface is useful when an Ethernet LAN is connected between two provider edge (PE) routers with either of the following conditions:	*Configuring HSRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
Cisco IOS Release 12.4T			
12.4(20)T	FHRP - EOT Deprecation of **rtr** Keyword	Effective with Cisco IOS Release 12.4(20)T, the **track rtr** command is replaced by the **track ip sla** command.	*Configuring Enhanced Object Tracking*
12.4(15)T	GLBP Client Cache	The GLBP client cache contains information about network hosts that are using a GLBP group as the default gateway. The GLBP client cache stores the MAC address of each host that is using a particular GLBP group, the number of the GLBP forwarder that each network host has been assigned to and the total number of network hosts currently assigned to each forwarder in a GLBP group. The GLBP client cache also stores the protocol address used by each network host and the time elapsed since the host-to-forwarder assignment was last updated.	*Configuring GLBP*
12.4(11)T	FHRP—HSRP BFD Peering	The HSRP BFD Peering feature introduces Bidirectional Forwarding Detection (BFD) in the HSRP group member health monitoring system. Previously, group member monitoring relied exclusively on HSRP multicast messages, which are relatively large and consume CPU memory to produce and check. In architectures where a single interface hosts a large number of groups, there is a need for a protocol with low CPU memory consumption and processing overhead. BFD addresses this issue and offers sub-second health monitoring (failure detection in milliseconds) at a relatively low CPU impact.	*Configuring HSRP*
	FHRP—Enhanced Object Tracking Support for Mobile IP	The FHRP—Enhanced Object Tracking Support for Mobile IP feature provides new tracking objects needed by mobile wireless applications to track the presence of Home Agent, Gateway GPRS Support Node (GGSN), or Packet Data Serving Node (PDSN) traffic on a router.	*Configuring Enhanced Object Tracking*
12.4(9)T	EOT Support for Carrier Delay	The EOT Support for Carrier Delay feature enables EOT to consider the carrier-delay timer when tracking the status of an interface.	*Configuring Enhanced Object Tracking*
	FHRP—HSRP Group Shutdown	The FHRP—HSRP Group Shutdown feature enables you to configure an HSRP group to become disabled (its state changed to Init) instead of having its priority decremented when a tracked object goes down.	*Configuring HSRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
12.4(6)T	HSRP Multiple Group Optimization	HSRP Multiple Group Optimization improves the negotiation and maintenance of multiple HSRP groups configured on a subinterface. Only one HSRP group is required on a physical interface for the purposes of electing active and standby routers. This group is known as the *master* group. Other HSRP groups may be created on each subinterface and linked to the master group via the group name. These linked HSRP groups are known as *client* or *follow* groups.	*Configuring HSRP*
12.4(2)T	FHRP—Enhanced Object Tracking Integration with Embedded Event Manager	EOT is now integrated with EEM to allow EEM to report on a status change of a tracked object and to allow EOT to track EEM objects.	*Configuring Enhanced Object Tracking*
Cisco IOS XE Release 2			
Cisco IOS XE Release 2.1	Enhanced Tracking Support	The Enhanced Tracking Support feature separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by other Cisco IOS processes as well as HSRP. This feature allows tracking of other objects in addition to the interface line-protocol state.	*Configuring Enhanced Object Tracking*
	FHRP - Enhanced Object Tracking of IP SLAs	This feature enables FHRPs and other EOT clients to track the output from IP SLAs objects and use the provided information to trigger an action.	*Configuring Enhanced Object Tracking*
	FHRP—HSRP-MIB	The FHRP—HSRP-MIB feature introduces support for the CISCO-HRSP-MIB.	*Configuring HSRP*
	FHRP—Object Tracking List	This feature enhances the tracking capabilities to enable the configuration of a combination of tracked objects in a list, and a flexible method of combining objects using Boolean logic.	*Configuring Enhanced Object Tracking*
	Gateway Load Balancing Protocol (GLBP)	Gateway Load Balancing Protocol (GLBP) protects data traffic from a failed router or circuit, like Hot Standby Router Protocol (HSRP) and Virtual Router Redundancy Protocol (VRRP), while allowing packet load sharing between a group of redundant routers.	*Configuring GLBP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
	HSRP—Hot Standby Router Protocol	The Hot Standby Router Protocol (HSRP) is a First Hop Redundancy Protocol (FHRP) designed to allow for transparent fail-over of the first-hop IP router. HSRP provides high network availability by providing first-hop routing redundancy for IP hosts on Ethernet, Fiber Distributed Data Interface (FDDI), Bridge-Group Virtual Interface (BVI), LAN Emulation (LANE), or Token Ring networks configured with a default gateway IP address.	*Configuring HSRP*
	HSRP—ISSU	The HSRP—ISSU feature enables support for ISSU in HSRP. The ISSU process allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues.	*Configuring HSRP*
	HSRP Support for MPLS VPNs	HSRP support for a Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) interface is useful when an Ethernet LAN is connected between two provider edge (PE) routers with either of the following conditions:	*Configuring HSRP*
	ISSU—GLBP	GLBP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in SSO mode even when different versions of Cisco IOS software are running on the active and standby RPs or line cards.	*Configuring GLBP*
	ISSU—VRRP	VRRP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in Stateful Switchover (SSO) mode even when different versions of Cisco IOS software are running on the active and standby Route Processors (RPs) or line cards.	*Configuring VRRP*
	SSO—HSRP	SSO HSRP alters the behavior of HSRP when a router with redundant Route Processors (RPs) is configured for Stateful Switchover (SSO). When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.	*Configuring HSRP*
	SSO—VRRP	VRRP is now SSO aware. VRRP can detect when a router is failing over to the secondary RP and continue in its current VRRP group state.	*Configuring VRRP*
	VRRP MIB—RFC 2787	This feature enables an enhancement to the MIB for use with SNMP-based network management. The feature adds support for configuring, monitoring, and controlling routers that use VRRP.	*Configuring VRRP*

Table 1 Supported FHRP Features (continued)

Release	Feature Name	Feature Description	Where Documented
Other Cisco IOS Releases			
12.2(27) SBC	FHRP—Enhanced Object Tracking of IP SLAs Operations	The FHRP—Enhanced Object Tracking of IP SLAs Operations feature enables FHRPs and other EOT clients to track the output from IP SLAs objects and use the provided information to trigger an action.	*Configuring Enhanced Object Tracking*
12.2(31) SGA	HSRP—ISSU	The HSRP—ISSU feature enables support for ISSU in HSRP. The ISSU process allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues.	*Configuring HSRP*
12.0(3)T 12.0(12)S	FHRP—HSRP-MIB	The FHRP—HSRP-MIB feature introduces support for the CISCO-HRSP-MIB.	*Configuring HSRP*

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring GLBP

First Published: May 2, 2005
Last Updated: May 5, 2008

Gateway Load Balancing Protocol (GLBP) protects data traffic from a failed router or circuit, like Hot Standby Router Protocol (HSRP) and Virtual Router Redundancy Protocol (VRRP), while allowing packet load sharing between a group of redundant routers.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for <Phrase Based on Module Title>" section on page 6.

Use Cisco Feature Navigator to find information about platform support and Cisco IOS, Catalyst OS, and Cisco IOS XE software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for GLBP

Before configuring GLBP, ensure that the routers can support multiple MAC addresses on the physical interfaces. For each GLBP forwarder to be configured, an additional MAC address is used.

Information About GLBP

To configure GLBP, you need to understand the following concepts:

GLBP Overview

GLBP provides automatic router backup for IP hosts configured with a single default gateway on an IEEE 802.3 LAN. Multiple first-hop routers on the LAN combine to offer a single virtual first-hop IP router while sharing the IP packet forwarding load. Other routers on the LAN may act as redundant GLBP routers that will become active if any of the existing forwarding routers fail.

GLBP performs a similar function for the user as HSRP and VRRP. HSRP and VRRP allow multiple routers to participate in a virtual router group configured with a virtual IP address. One member is elected to be the active router to forward packets sent to the virtual IP address for the group. The other routers in the group are redundant until the active router fails. These standby routers have unused bandwidth that the protocol is not using. Although multiple virtual router groups can be configured for the same set of routers, the hosts must be configured for different default gateways, which results in an extra administrative burden. The advantage of GLBP is that it additionally provides load balancing over multiple routers (gateways) using a single virtual IP address and multiple virtual MAC addresses. The forwarding load is shared among all routers in a GLBP group rather than being handled by a single router while the other routers stand idle. Each host is configured with the same virtual IP address, and all routers in the virtual router group participate in forwarding packets. GLBP members communicate between each other through hello messages sent every 3 seconds to the multicast address 224.0.0.102, User Datagram Protocol (UDP) port 3222 (source and destination).

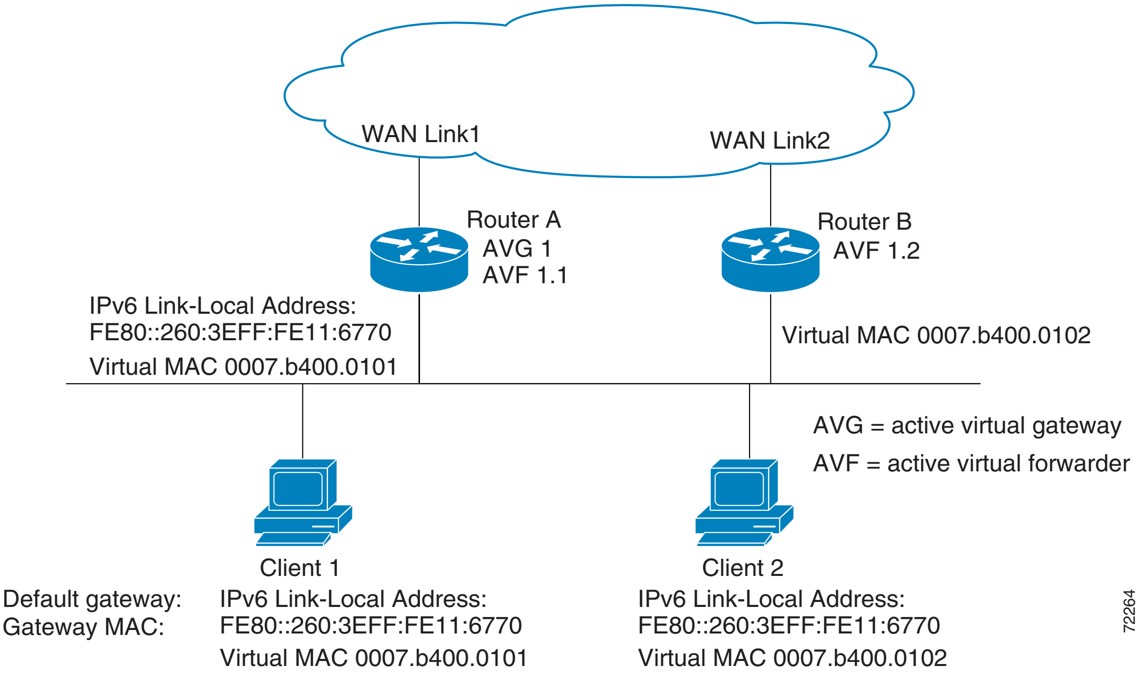
GLBP Active Virtual Gateway

Members of a GLBP group elect one gateway to be the active virtual gateway (AVG) for that group. Other group members provide backup for the AVG in the event that the AVG becomes unavailable. The function of the AVG is that it assigns a virtual MAC address to each member of the GLBP group. Each gateway assumes responsibility for forwarding packets sent to the virtual MAC address assigned to it by the AVG. These gateways are known as active virtual forwarders (AVFs) for their virtual MAC address.

The AVG is also responsible for answering Address Resolution Protocol (ARP) requests for the virtual IP address. Load sharing is achieved by the AVG replying to the ARP requests with different virtual MAC addresses.

In Figure 1, Router A is the AVG for a GLBP group, and is responsible for the virtual IP address 10.21.8.10. Router A is also an AVF for the virtual MAC address 0007.b400.0101. Router B is a member of the same GLBP group and is designated as the AVF for the virtual MAC address 0007.b400.0102. Client 1 has a default gateway IP address of 10.21.8.10 and a gateway MAC address of 0007.b400.0101. Client 2 shares the same default gateway IP address but receives the gateway MAC address 0007.b400.0102 because Router B is sharing the traffic load with Router A.

Figure 1 GLBP Topology



If Router A becomes unavailable, Client 1 will not lose access to the WAN because Router B will assume responsibility for forwarding packets sent to the virtual MAC address of Router A, and for responding to packets sent to its own virtual MAC address. Router B will also assume the role of the AVG for the entire GLBP group. Communication for the GLBP members continues despite the failure of a router in the GLBP group.

GLBP Virtual MAC Address Assignment

A GLBP group allows up to four virtual MAC addresses per group. The AVG is responsible for assigning the virtual MAC addresses to each member of the group. Other group members request a virtual MAC address after they discover the AVG through hello messages. Gateways are assigned the next MAC address in sequence. A virtual forwarder that is assigned a virtual MAC address by the AVG is known as a primary virtual forwarder. Other members of the GLBP group learn the virtual MAC addresses from hello messages. A virtual forwarder that has learned the virtual MAC address is referred to as a secondary virtual forwarder.

GLBP Virtual Gateway Redundancy

GLBP operates virtual gateway redundancy in the same way as HSRP. One gateway is elected as the AVG, another gateway is elected as the standby virtual gateway, and the remaining gateways are placed in a listen state.

If an AVG fails, the standby virtual gateway will assume responsibility for the virtual IP address. A new standby virtual gateway is then elected from the gateways in the listen state.

GLBP Virtual Forwarder Redundancy

Virtual forwarder redundancy is similar to virtual gateway redundancy with an AVF. If the AVF fails, one of the secondary virtual forwarders in the listen state assumes responsibility for the virtual MAC address.

The new AVF is also a primary virtual forwarder for a different forwarder number. GLBP migrates hosts away from the old forwarder number using two timers that start as soon as the gateway changes to the active virtual forwarder state. GLBP uses the hello messages to communicate the current state of the timers.

The redirect time is the interval during which the AVG continues to redirect hosts to the old virtual forwarder MAC address. When the redirect time expires, the AVG stops using the old virtual forwarder MAC address in ARP replies, although the virtual forwarder will continue to forward packets that were sent to the old virtual forwarder MAC address.

The secondary holdtime is the interval during which the virtual forwarder is valid. When the secondary holdtime expires, the virtual forwarder is removed from all gateways in the GLBP group. The expired virtual forwarder number becomes eligible for reassignment by the AVG.

GLBP Gateway Priority

GLBP gateway priority determines the role that each GLBP gateway plays and what happens if the AVG fails.

Priority also determines if a GLBP router functions as a backup virtual gateway and the order of ascendancy to becoming an AVG if the current AVG fails. You can configure the priority of each backup virtual gateway with a value of 1 through 255 using the **glbp priority** command.

In Figure 1, if Router A—the AVG in a LAN topology—fails, an election process takes place to determine which backup virtual gateway should take over. In this example, Router B is the only other member in the group so it will automatically become the new AVG. If another router existed in the same

GLBP group with a higher priority, then the router with the higher priority would be elected. If both routers have the same priority, the backup virtual gateway with the higher IP address would be elected to become the active virtual gateway.

By default, the GLBP virtual gateway preemptive scheme is disabled. A backup virtual gateway can become the AVG only if the current AVG fails, regardless of the priorities assigned to the virtual gateways. You can enable the GLBP virtual gateway preemptive scheme using the **glbp preempt** command. Preemption allows a backup virtual gateway to become the AVG, if the backup virtual gateway is assigned a higher priority than the current AVG.

GLBP Gateway Weighting and Tracking

GLBP uses a weighting scheme to determine the forwarding capacity of each router in the GLBP group. The weighting assigned to a router in the GLBP group can be used to determine whether it will forward packets and, if so, the proportion of hosts in the LAN for which it will forward packets. Thresholds can be set to disable forwarding when the weighting for a GLBP group falls below a certain value, and when it rises above another threshold, forwarding is automatically reenabled.

The GLBP group weighting can be automatically adjusted by tracking the state of an interface within the router. If a tracked interface goes down, the GLBP group weighting is reduced by a specified value. Different interfaces can be tracked to decrement the GLBP weighting by varying amounts.

By default, the GLBP virtual forwarder preemptive scheme is enabled with a delay of 30 seconds. A backup virtual forwarder can become the AVF if the current AVF weighting falls below the low weighting threshold for 30 seconds. You can disable the GLBP forwarder preemptive scheme using the **no glbp forwarder preempt** command or change the delay using the **glbp forwarder preempt delay minimum** command.

GLBP Client Cache

The GLBP client cache contains information about network hosts that are using a GLBP group as the default gateway.

When an IPv4 Address Resolution Protocol (ARP) request or an IPv6 Neighbor Discovery (ND) request for a GLBP virtual IP address is received from a network host by a GLBP group’s Active Virtual Gateway (AVG), a new entry is created in the GLBP client cache. The cache entry contains information about the host that sent the ARP or ND request and which forwarder the AVG has assigned to it.

The GLBP client cache stores the MAC address of each host that is using a particular GLBP group, the number of the GLBP forwarder that each network host has been assigned to and the total number of network hosts currently assigned to each forwarder in a GLBP group. The GLBP client cache also stores the protocol address used by each network host and the time elapsed since the host-to-forwarder assignment was last updated.

The GLBP client cache can store information on up to 2000 network hosts for a GLBP group. The expected normal maximum configuration is 1000 network hosts. You can configure a lower maximum number of network hosts that will be cached for each GLBP group independently based on the number of network hosts that are using each GLBP group by using the **glbp client-cache maximum** command. This command enables you to limit the amount of memory used by the cache per GLBP group. If the GLBP client cache has reached the maximum configured number of clients and a new client is added, the least recently updated client entry will be discarded. Reaching this condition indicates that the configured maximum limit is too small.

The amount of memory that is used by the GLBP client cache is dependent upon the number of network hosts using GLBP groups for which the client cache is enabled. For each host at least 20 bytes is required, with an additional 3200 bytes per GLBP group.

You can display the contents of the GLBP client cache using the **show glbp detail** command on the router that is currently the AVG for a GLBP group. If you issue the **show glbp detail** command on any other router in a GLBP group, you will be directed to reissue the command on the AVG to view client cache information. The **show glbp detail** command also displays statistics about the GLBP client cache usage and the distribution of clients among forwarders. These statistics are accurate as long as the cache timeout and client limit parameters have been set appropriately. Appropriate values would be where the number of end hosts on the network does not exceed the configured limit and where the maximum end host ARP cache timeout does not exceed the configured GLBP client cache timeout.

You can enable or disable the GLBP client cache independently for each GLBP group by using the **glbp client-cache** command. The GLBP client cache is disabled by default. There is no limit on the number of groups for which the GLBP client cache can be enabled.

You can configure GLBP cache entries to time out after a specified time by using the **timeout** keyword option with the **glbp client-cache maximum** command.

ISSU—GLBP

GLBP supports In Service Software Upgrade (ISSU). In Service Software Upgrade (ISSU) allows a high-availability (HA) system to run in Stateful Switchover (SSO) mode even when different versions of Cisco IOS software are running on the active and standby Route Processors (RPs) or line cards.

ISSU provides the ability to upgrade or downgrade from one supported Cisco IOS release to another while continuing to forward packets and maintain sessions, thereby reducing planned outage time. The ability to upgrade or downgrade is achieved by running different software versions on the active RP and standby RP for a short period of time to maintain state information between RPs. This feature allows the system to switch over to a secondary RP running upgraded (or downgraded) software and continue forwarding packets without session loss and with minimal or no packet loss. This feature is enabled by default.

For detailed information about ISSU, see the *Cisco IOS In Service Software Upgrade Process* document at the following URL:

http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-inserv_updg.html

For detailed information about ISSU on the 7600 series routers, see the *ISSU and eFSU on Cisco 7600 Series Routers* document at the following URL:

http://www.cisco.com/en/US/docs/routers/7600/ios/12.2SR/configuration/efsuovrw.html

GLBP SSO

With the introduction of the GLBP SSO feature, GLBP is Stateful Switchover (SSO) aware. GLBP can detect when a router is failing over to the secondary RP and continue in its current group state.

SSO functions in networking devices (usually edge devices) that support dual Route Processors (RPs). SSO provides RP redundancy by establishing one of the RPs as the active processor and the other RP as the standby processor. SSO also synchronizes critical state information between the RPs so that network state information is dynamically maintained between RPs.

Prior to being SSO aware, if GLBP was deployed on a router with redundant RPs, a switchover of roles between the active RP and the standby RP would result in the router relinquishing its activity as a GLBP group member and then rejoining the group as if it had been reloaded. The GLBP SSO feature enables GLBP to continue its activities as a group member during a switchover. GLBP state information between redundant RPs is maintained so that the standby RP can continue the router's activities within the GLBP during and after a switchover.

This feature is enabled by default. To disable this feature, use the **no glbp sso** command in global configuration mode.

For more information, see the *Stateful Switchover* document.

GLBP Benefits

Load Sharing

You can configure GLBP in such a way that traffic from LAN clients can be shared by multiple routers, thereby sharing the traffic load more equitably among available routers.

Multiple Virtual Routers

GLBP supports up to 1024 virtual routers (GLBP groups) on each physical interface of a router and up to four virtual forwarders per group.

Preemption

The redundancy scheme of GLBP enables you to preempt an active virtual gateway with a higher priority backup virtual gateway that has become available. Forwarder preemption works in a similar way, except that forwarder preemption uses weighting instead of priority and is enabled by default.

Authentication

You can also use the industry-standard message digest 5 (MD5) algorithm for improved reliability, security, and protection against GLBP-spoofing software. A router within a GLBP group with a different authentication string than other routers will be ignored by other group members. You can alternatively use a simple text password authentication scheme between GLBP group members to detect configuration errors.

How to Configure GLBP

This section contains the following procedures:

- Enabling and Verifying GLBP, page 8 (required)
- Customizing GLBP, page 10 (optional)
- Configuring GLBP Authentication, page 12 (optional)
- Configuring GLBP Weighting Values and Object Tracking, page 17 (optional)
- Troubleshooting GLBP, page 19 (optional)

Enabling and Verifying GLBP

Perform this task to enable GLBP on an interface and verify its configuration and operation. GLBP is designed to be easy to configure. Each gateway in a GLBP group must be configured with the same group number, and at least one gateway in the GLBP group must be configured with the virtual IP address to be used by the group. All other required parameters can be learned.

Prerequisites

If VLANs are in use on an interface, the GLBP group number must be different for each VLAN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **glbp** *group* **ip** [*ip-address* [**secondary**]]
6. **exit**
7. **show glbp** [*interface-type interface-number*] [*group*] [*state*] [**brief**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface fastethernet 0/0`
		Specifies an interface type and number, and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-if)# ip address 10.21.8.32 255.255.255.0`
		Specifies a primary or secondary IP address for an interface.

	Command or Action	Purpose
Step 5	**glbp** *group* **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# glbp 10 ip 10.21.8.10`	Enables GLBP on an interface and identifies the primary IP address of the virtual gateway. • After you identify a primary IP address, you can use the **glbp** *group* **ip** command again with the **secondary** keyword to indicate additional IP addresses supported by this group.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode, and returns the router to global configuration mode.
Step 7	**show glbp** [*interface-type interface-number*] [*group*] [*state*] [**brief**] **Example:** `Router(config)# show glbp 10`	(Optional) Displays information about GLBP groups on a router. • Use the optional **brief** keyword to display a single line of information about each virtual gateway or virtual forwarder. • See the display output for this command in the "Examples" section of this task.

Examples

In the following example, sample output is displayed about the status of the GLBP group, named 10, on the router:

```
Router# show glbp 10

FastEthernet0/0 - Group 10
  State is Active
    2 state changes, last state change 23:50:33
  Virtual IP address is 10.21.8.10
  Hello time 5 sec, hold time 18 sec
    Next hello sent in 4.300 secs
  Redirect time 600 sec, forwarder time-out 7200 sec
  Authentication text "stringabc"
  Preemption enabled, min delay 60 sec
  Active is local
  Standby is unknown
  Priority 254 (configured)
  Weighting 105 (configured 110), thresholds: lower 95, upper 105
    Track object 2 state Down decrement 5
  Load balancing: host-dependent
  There is 1 forwarder (1 active)
  Forwarder 1
    State is Active
      1 state change, last state change 23:50:15
    MAC address is 0007.b400.0101 (default)
    Owner ID is 0005.0050.6c08
    Redirection enabled
    Preemption enabled, min delay 60 sec
    Active is local, weighting 105
```

Customizing GLBP

Perform this task to customize your GLBP configuration.

Customizing the behavior of GLBP is optional. Be aware that as soon as you enable a GLBP group, that group is operating. It is possible that if you first enable a GLBP group before customizing GLBP, the router could take over control of the group and become the AVG before you have finished customizing the feature. Therefore, if you plan to customize GLBP, it is a good idea to do so before enabling GLBP.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **glbp** *group* **timers** [**msec**] *hellotime* [**msec**] *holdtime*
6. **glbp** *group* **timers redirect** *redirect timeout*
7. **glbp** *group* **load-balancing** [**host-dependent** | **round-robin** | **weighted**]
8. **glbp** *group* **priority** *level*
9. **glbp** *group* **preempt** [**delay minimum** *seconds*]
10. **glbp** *group* **client-cache maximum** *number* [**timeout** *minutes*]
11. **glbp** *group* **name** *redundancy-name*
12. **exit**
13. **no glbp sso**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface fastethernet 0/0`
		Specifies an interface type and number, and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-if)# ip address 10.21.8.32 255.255.255.0`
		Specifies a primary or secondary IP address for an interface.

	Command or Action	Purpose
Step 5	**glbp** *group* **timers** [**msec**] *hellotime* [**msec**] *holdtime* **Example:** `Router(config-if)# glbp 10 timers 5 18`	Configures the interval between successive hello packets sent by the AVG in a GLBP group. • The *holdtime* argument specifies the interval in seconds before the virtual gateway and virtual forwarder information in the hello packet is considered invalid. • The optional **msec** keyword specifies that the following argument will be expressed in milliseconds, instead of the default seconds.
Step 6	**glbp** *group* **timers redirect** *redirect timeout* **Example:** `Router(config-if)# glbp 10 timers redirect 600 7200`	Configures the time interval during which the AVG continues to redirect clients to an AVF. • The *timeout* argument specifies the interval in seconds before a secondary virtual forwarder becomes invalid.
Step 7	**glbp** *group* **load-balancing** [**host-dependent** | **round-robin** | **weighted**] **Example:** `Router(config-if)# glbp 10 load-balancing host-dependent`	Specifies the method of load balancing used by the GLBP AVG.
Step 8	**glbp** *group* **priority** *level* **Example:** `Router(config-if)# glbp 10 priority 254`	Sets the priority level of the gateway within a GLBP group. • The default value is 100.
Step 9	**glbp** *group* **preempt** [**delay minimum** *seconds*] **Example:** `Router(config-if)# glbp 10 preempt delay minimum 60`	Configures the router to take over as AVG for a GLBP group if it has a higher priority than the current AVG. • This command is disabled by default. • Use the optional **delay** and **minimum** keywords and the *seconds* argument to specify a minimum delay interval in seconds before preemption of the AVG takes place.
Step 10	**glbp** *group* **client-cache maximum** *number* [**timeout** *minutes*] **Example:** `Router(config-if)# glbp 10 client-cache maximum 1200 timeout 245`	(Optional) Enables the GLBP client cache. • This command is disabled by default. • Use the *number* argument to specify the maximum number of clients the cache will hold for this GLBP group. The range is from 8 to 2000. • Use the optional **timeout** *minutes* keyword and argument pair to configure the maximum amount of time a client entry can stay in the GLBP client cache after the client information was last updated. The range is from 1 to 1440 minutes (one day). **Note** For IPv4 networks, Cisco recommends setting a GLBP client cache timeout value that is slightly longer than the maximum expected end-host Address Resolution Protocol (ARP) cache timeout value.

	Command or Action	Purpose
Step 11	**glbp** *group* **name** *redundancy-name* **Example:** `Router(config-if)# glbp 10 name abcompany`	Enables IP redundancy by assigning a name to the GLBP group. • The GLBP redundancy client must be configured with the same GLBP group name so the redundancy client and the GLBP group can be connected.
Step 12	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode, and returns the router to global configuration mode.

Step 13	`no glbp sso`
	Example:
	`Router(config)# no glbp sso`
		(Optional) Disables GLBP support of SSO.

Configuring GLBP Authentication

The following sections describe configuration tasks for GLBP authentication. The task you perform depends on whether you want to use text authentication, a simple MD5 key string, or MD5 key chains for authentication.

- Configuring GLBP MD5 Authentication Using a Key String, page 13
- Configuring GLBP MD5 Authentication Using a Key Chain, page 14
- Configuring GLBP Text Authentication, page 16

How GLBP MD5 Authentication Works

MD5 authentication provides greater security than the alternative plain text authentication scheme. MD5 authentication allows each GLBP group member to use a secret key to generate a keyed MD5 hash that is part of the outgoing packet. A keyed hash of an incoming packet is generated and, if the hash within the incoming packet does not match the generated hash, the packet is ignored.

The key for the MD5 hash can either be given directly in the configuration using a key string or supplied indirectly through a key chain.

A router will ignore incoming GLBP packets from routers that do not have the same authentication configuration for a GLBP group. GLBP has three authentication schemes:

- No authentication
- Plain text authentication
- MD5 authentication

GLBP packets will be rejected in any of the following cases:

- The authentication schemes differ on the router and in the incoming packet.
- MD5 digests differ on the router and in the incoming packet.
- Text authentication strings differ on the router and in the incoming packet.

Benefits of GLBP MD5 Authentication

- Protects against spoofing software.
- Uses the industry-standard MD5 algorithm for improved reliability and security.

Configuring GLBP MD5 Authentication Using a Key String

Perform this task to configure GLBP MD5 authentication using a key string.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **glbp** *group-number* **authentication md5 key-string** [**0** | **7**] *key*
6. **glbp** *group-number* **ip** [*ip-address* [**secondary**]]
7. Repeat Steps 1 through 6 on each router that will communicate.
8. **end**
9. **show glbp**

DETAILED STEPS

	Command	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`interface` *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	`ip address` *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-if)# ip address 10.0.0.1 255.255.255.0`
		Specifies a primary or secondary IP address for an interface.

	Command	Purpose

Step 5	**`glbp`** *`group-number`* **`authentication md5 key-string`** `[`**`0`** `|`
	`7``]` *`key`*
	Example:
	`Router(config-if)# glbp 1 authentication md5 key-string d00b4r987654321a`
		Configures an authentication key for GLBP MD5 authentication.
		• The number of characters in the command plus the key string must not exceed 255 characters.
		• No prefix to the *key* argument or specifying **0** means the key is unencrypted.
		• Specifying **7** means the key is encrypted. The key-string authentication key will automatically be encrypted if the **service password-encryption** global configuration command is enabled.

Step 6	**`glbp`** *`group-number`* **`ip`** `[`*`ip-address`* `[`**`secondary`**`]]`
	Example:
	`Router(config-if)# glbp 1 ip 10.0.0.10`
		Enables GLBP on an interface and identifies the primary IP address of the virtual gateway.

Step 7	Repeat Steps 1 through 6 on each router that will communicate.	—

Step 8	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 9	**`show glbp`**
	Example:
	`Router# show glbp`
		(Optional) Displays GLBP information.
		• Use this command to verify your configuration. The key string and authentication type will be displayed if configured.

Configuring GLBP MD5 Authentication Using a Key Chain

Perform this task to configure GLBP MD5 authentication using a key chain. Key chains allow a different key string to be used at different times according to the key chain configuration. GLBP will query the appropriate key chain to obtain the current live key and key ID for the specified key chain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **key chain** *name-of-chain*
4. **key** *key-id*
5. **key-string** *string*
6. **exit**
7. exit
8. **interface** *type number*
9. **ip address** *ip-address mask* [**secondary**]

10. **glbp** *group-number* **authentication md5 key-chain** *name-of-chain*
11. **glbp** *group-number* **ip** [*ip-address* [**secondary**]]
12. Repeat Steps 1 through 10 on each router that will communicate.
13. **end**
14. **show glbp**
15. **show key chain**

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables higher privilege levels, such as privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`key chain`** *`name-of-chain`*
	Example:
	`Router(config)# key chain glbp2`
		Enables authentication for routing protocols and identifies a group of authentication keys.

Step 4	**`key`** *`key-id`*
	Example:
	`Router(config-keychain)# key 100`
		Identifies an authentication key on a key chain.
		• The *key-id* must be a number.

Step 5	**`key-string`** *`string`*
	Example:
	`Router(config-keychain-key)# key-string xmen382`
		Specifies the authentication string for a key.
		• The *string* can be 1 to 80 uppercase or lowercase alphanumeric characters; the first character cannot be a numeral.

Step 6	**`exit`**
	Example:
	`Router(config-keychain-key)# exit`
		Returns to keychain configuration mode.

Step 7	**`exit`**
	Example:
	`Router(config-keychain)# exit`
		Returns to global configuration mode.

Step 8	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

	Command	Purpose
Step 9	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.21.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 10	**glbp** *group-number* **authentication md5 key-chain** *name-of-chain* **Example:** `Router(config-if)# glbp 1 authentication md5 key-chain glbp2`	Configures an authentication MD5 key chain for GLBP MD5 authentication. • The key chain name must match the name specified in Step 3.
Step 11	**glbp** *group-number* **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# glbp 1 ip 10.21.0.12`	Enables GLBP on an interface and identifies the primary IP address of the virtual gateway.
Step 12	Repeat Steps 1 through 10 on each router that will communicate.	—

Step 13	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 14	**show glbp** **Example:** `Router# show glbp`	(Optional) Displays GLBP information. • Use this command to verify your configuration. The key chain and authentication type will be displayed if configured.
Step 15	**show key chain** **Example:** `Router# show key chain`	(Optional) Displays authentication key information.

Configuring GLBP Text Authentication

Perform this task to configure GLBP text authentication. This method of authentication provides minimal security. Use MD5 authentication if security is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **glbp** *group-number* **authentication text** *string*
6. **glbp** *group-number* **ip** [*ip-address* [**secondary**]]
7. Repeat Steps 1 through 6 on each router that will communicate.

8. **end**
9. **show glbp**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet0/1`	Configures an interface type and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 5	**glbp** *group-number* **authentication text** *string* **Example:** `Router(config-if)# glbp 10 authentication text stringxyz`	Authenticates GLBP packets received from other routers in the group. • If you configure authentication, all routers within the GLBP group must use the same authentication string.
Step 6	**glbp** *group-number* **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# glbp 1 ip 10.0.0.10`	Enables GLBP on an interface and identifies the primary IP address of the virtual gateway.
Step 7	Repeat Steps 1 through 6 on each router that will communicate.	—

Step 8	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 9	**show glbp** **Example:** `Router# show glbp`	(Optional) Displays GLBP information. • Use this command to verify your configuration.

Configuring GLBP Weighting Values and Object Tracking

Perform this task to configure GLBP weighting values and object tracking.

GLBP weighting is used to determine whether a GLBP group can act as a virtual forwarder. Initial weighting values can be set and optional thresholds specified. Interface states can be tracked and a decrement value set to reduce the weighting value if the interface goes down. When the GLBP group weighting drops below a specified value, the group will no longer be an active virtual forwarder. When the weighting rises above a specified value, the group can resume its role as an active virtual forwarder.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* **interface** *type number* {**line-protocol** | **ip routing**}
4. **exit**
5. **interface** *type number*
6. **glbp** *group* **weighting** *maximum* [**lower** *lower*] [**upper** *upper*]
7. **glbp** *group* **weighting track** *object-number* [**decrement** *value*]
8. **glbp** *group* **forwarder preempt** [**delay minimum** *seconds*]
9. **end**
10. **show track** [*object-number* | **brief**] [**interface** [**brief**] | **ip route** [**brief**] | **resolution** | **timers**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**track** *object-number* **interface** *type number* {**line-protocol** | **ip routing**}
	Example:
	`Router(config)# track 2 interface POS 6/0 ip routing`
		Configures an interface to be tracked where changes in the state of the interface affect the weighting of a GLBP gateway, and enters tracking configuration mode.
		• This command configures the interface and corresponding object number to be used with the **glbp weighting track** command.
		• The **line-protocol** keyword tracks whether the interface is up. The **ip routing** keywords also check that IP routing is enabled on the interface, and an IP address is configured.

Step 4	**exit** **Example:** `Router(config-track)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 5	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Enters interface configuration mode.
Step 6	**glbp** *group* **weighting** *maximum* [**lower** *lower*] [**upper** *upper*] **Example:** `Router(config-if)# glbp 10 weighting 110 lower 95 upper 105`	Specifies the initial weighting value, and the upper and lower thresholds, for a GLBP gateway.
Step 7	**glbp** *group* **weighting track** *object-number* [**decrement** *value*] **Example:** `Router(config-if)# glbp 10 weighting track 2 decrement 5`	Specifies an object to be tracked that affects the weighting of a GLBP gateway. • The *value* argument specifies a reduction in the weighting of a GLBP gateway when a tracked object fails.
Step 8	**glbp** *group* **forwarder preempt** [**delay minimum** *seconds*] **Example:** `Router(config-if)# glbp 10 forwarder preempt delay minimum 60`	Configures the router to take over as AVF for a GLBP group if the current AVF for a GLBP group falls below its low weighting threshold. • This command is enabled by default with a delay of 30 seconds. • Use the optional **delay** and **minimum** keywords and the *seconds* argument to specify a minimum delay interval in seconds before preemption of the AVF takes place.

Step 9	**end**
	Example:
	`Router(config-if)# exit`
		Returns to privileged EXEC mode.

Step 10	**show track** [*object-number* | **brief**] [**interface** [**brief**]| **ip route** [**brief**] | **resolution** | **timers**] **Example:** `Router# show track 2`	Displays tracking information.

Troubleshooting GLBP

GLBP introduces five privileged EXEC mode commands to enable diagnostic output concerning various events relating to the operation of GLBP to be displayed on a console. The **debug condition glbp**, **debug glbp errors**, **debug glbp events**, **debug glbp packets**, and **debug glbp terse** commands are intended only for troubleshooting purposes because the volume of output generated by the software can result in severe performance degradation on the router. Perform this task to minimize the impact of using the **debug glbp** commands.

This procedure will minimize the load on the router created by the **debug condition glbp** or **debug glbp** commands because the console port is no longer generating character-by-character processor interrupts. If you cannot connect to a console directly, you can run this procedure via a terminal server. If you must break the Telnet connection, however, you may not be able to reconnect because the router may be unable to respond due to the processor load of generating the debugging output.

Prerequisites

This task requires a router running GLBP to be attached directly to a console.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no logging console**
4. Use Telnet to access a router port and repeat Steps 1 and 2.
5. **end**
6. **terminal monitor**
7. **debug condition glbp** *interface-type interface-number group* [*forwarder*]
8. **terminal no monitor**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`no logging console`
	Example:
	`Router(config)# no logging console`
		Disables all logging to the console terminal.
		• To reenable logging to the console, use the **logging console** command in global configuration mode.

Step 4	Use Telnet to access a router port and repeat Steps 1 and 2.	Enters global configuration mode in a recursive Telnet session, which allows the output to be redirected away from the console port.

Step 5	`end`
	Example:
	`Router(config)# end`
		Exits to privileged EXEC mode.

	Command or Action	Purpose

Step 6	**`terminal monitor`**
	Example:
	`Router# terminal monitor`
		Enables logging output on the virtual terminal.

Step 7	**`debug condition glbp`** *`interface-type interface-number group`* `[`*`forwarder`*`]`
	Example:
	`Router# debug condition glbp fastethernet 0/0 10 1`
		Displays debugging messages about GLBP conditions.
		• Try to enter only specific **debug condition glbp** or **debug glbp** commands to isolate the output to a certain subcomponent and minimize the load on the processor. Use appropriate arguments and keywords to generate more detailed debug information on specified subcomponents.
		• Enter the specific **no debug condition glbp** or **no debug glbp** command when you are finished.

Step 8	**`terminal no monitor`**
	Example:
	`Router# terminal no monitor`
		Disables logging on the virtual terminal.

Configuration Examples for GLBP

This section contains the following configuration examples:

- Customizing GLBP Configuration: Example, page 21
- Configuring GLBP MD5 Authentication Using Key Strings: Example, page 22
- Configuring GLBP MD5 Authentication Using Key Chains: Example, page 22
- Configuring GLBP Text Authentication: Example, page 22
- Configuring GLBP Weighting: Example, page 22
- Enabling GLBP Configuration: Example, page 23

Customizing GLBP Configuration: Example

The following example shows how to configure Router A as shown in Figure 1:

```
interface fastethernet 0/0
 ip address 10.21.8.32 255.255.255.0
 glbp 10 timers 5 18
 glbp 10 timers redirect 600 7200
 glbp 10 load-balancing host-dependent
 glbp 10 priority 254
 glbp 10 preempt delay minimum 60
 glbp 10 client-cache maximum 1200 timeout 245
```

Configuring GLBP MD5 Authentication Using Key Strings: Example

The following example shows how to configure GLBP MD5 authentication using a key string:

```
!
interface Ethernet0/1
 ip address 10.0.0.1 255.255.255.0
 glbp 2 authentication md5 key-string ThisStringIsTheSecretKey
 glbp 2 ip 10.0.0.10
```

Configuring GLBP MD5 Authentication Using Key Chains: Example

In the following example, GLBP queries the key chain “AuthenticateGLBP” to obtain the current live key and key ID for the specified key chain:

```
key chain AuthenticateGLBP
 key 1
  key-string ThisIsASecretKey

interface Ethernet0/1
 ip address 10.0.0.1 255.255.255.0
 glbp 2 authentication md5 key-chain AuthenticateGLBP
 glbp 2 ip 10.0.0.10
```

Configuring GLBP Text Authentication: Example

The following example shows how to configure GLBP text authentication using a text string:

```
interface fastethernet 0/0
 ip address 10.21.8.32 255.255.255.0
 glbp 10 authentication text stringxyz
 glbp 10 ip 10.21.8.10
```

Configuring GLBP Weighting: Example

In the following example, Router A, shown in Figure 1, is configured to track the IP routing state of the POS interface 5/0 and 6/0, an initial GLBP weighting with upper and lower thresholds is set, and a weighting decrement value of 10 is set. If POS interface 5/0 and 6/0 goes down, the weighting value of the router is reduced.

```
track 1 interface POS 5/0 ip routing
track 2 interface POS 6/0 ip routing
interface fastethernet 0/0
 glbp 10 weighting 110 lower 95 upper 105
 glbp 10 weighting track 1 decrement 10
 glbp 10 weighting track 2 decrement 10
 glbp 10 forwarder preempt delay minimum 60
```

Enabling GLBP Configuration: Example

In the following example, Router A, shown in Figure 1, is configured to enable GLBP, and the virtual IP address of 10.21.8.10 is specified for GLBP group 10:

```
interface fastethernet 0/0
 ip address 10.21.8.32 255.255.255.0
 glbp 10 ip 10.21.8.10
```

Additional References

The following sections provide references related to GLBP.

Related Documents

Related Topic	Document Title
GLBP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference.*
In Service Software Upgrade (ISSU) configuration	*Cisco IOS In Service Software Upgrade Process*
Key chains and key management commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Routing Protocols Command Reference*
Object tracking	*Configuring Enhanced Object Tracking*
Stateful Switchover	*Stateful Switchover*
VRRP	*Configuring VRRP*
HSRP	*Configuring HSRP*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
No new MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for GLBP

Table 1 lists the features in this module and provides links to specific configuration information.

Not all commands may be available in your Cisco IOS software release. For details on when support for a specific command was introduced, see the command reference documentation.

For information on a feature in this technology that is not documented here, see the "Cisco IOS IP Application Services Features Roadmap" or the "FHRP Features Roadmap."

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for GLBP

Feature Name	Releases	Feature Configuration Information
Gateway Load Balancing Protocol	12.2(14)S 12.2(15)T Cisco IOS XE Release 2.1	GLBP protects data traffic from a failed router or circuit, like HSRP and VRRP, while allowing packet load sharing between a group of redundant routers. All sections in this configuration module provide information about this feature. The following commands were introduced or modified by this feature: **glbp forwarder preempt, glbp ip, glbp load-balancing, glbp name, glbp preempt, glbp priority, glbp sso, glbp timers, glbp timers redirect, glbp weighting, glbp weighting track, show glbp**.
GLBP Client Cache	12.4(15)T	The GLBP client cache contains information about network hosts that are using a GLBP group as the default gateway. The GLBP client cache stores the MAC address of each host that is using a particular GLBP group, the number of the GLBP forwarder that each network host has been assigned to and the total number of network hosts currently assigned to each forwarder in a GLBP group. The GLBP client cache also stores the protocol address used by each network host and the time elapsed since the host-to-forwarder assignment was last updated. The following sections provide information about this feature: • GLBP Client Cache, page 5 • Customizing GLBP, page 10 The following commands were introduced or modified by this feature: **glbp client-cache maximum** and **show glbp**.

Table 1 Feature Information for GLBP (continued)

Feature Name	Releases	Feature Configuration Information
GLBP MD5 Authentication	12.2(18)S 12.3(2)T 12.2(33)SXH	MD5 authentication provides greater security than the alternative plain text authentication scheme. MD5 authentication allows each GLBP group member to use a secret key to generate a keyed MD5 hash that is part of the outgoing packet. A keyed hash of an incoming packet is generated and, if the hash within the incoming packet does not match the generated hash, the packet is ignored. The following section provides information about this feature: • Configuring GLBP Authentication, page 12 The following commands were modified by this feature: **glbp authentication**, **show glbp**.
ISSU—GLBP	12.2(31)SB2 12.2(33)SRB1 Cisco IOS XE Release 2.1	GLBP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in Stateful Switchover (SSO) mode even when different versions of Cisco IOS software are running on the active and standby Route Processors (RPs) or line cards. This feature provides customers with the same level of HA functionality for planned outages due to software upgrades as is available with SSO for unplanned outages. That is, the system can switch over to a secondary RP and continue forwarding packets without session loss and with minimal or no packet loss. This feature is enabled by default. The following sections provide information about this feature: • ISSU—GLBP, page 6

Table 1 Feature Information for GLBP (continued)

Feature Name	Releases	Feature Configuration Information
SSO—GLBP	12.2(31)SB2 12.2(33)SRB 12.2(33)SXH	GLBP is now SSO aware. GLBP can detect when a router is failing over to the secondary RP and continue in its current GLBP group state. Prior to being SSO aware, GLBP was not able to detect that a second RP was installed and configured to take over in the event that the primary RP failed. When the primary failed, the GLBP device would stop participating in the GLBP group and, depending on its role, could trigger another router in the group to take over as the active router. With this enhancement, GLBP detects the failover to the secondary RP and no change occurs to the GLBP group. If the secondary RP fails and the primary is still not available, then the GLBP group detects this and re-elects a new active GLBP router. This feature is enabled by default. The following sections provide information about this feature: • GLBP SSO, page 6 • Customizing GLBP, page 10 The following commands were introduced or modified by this feature: **debug glbp events**, **glbp sso**, **show glbp**.

Glossary

active RP—The Route Processor (RP) controls the system, provides network services, runs routing protocols and presents the system management interface.

AVF—active virtual forwarder. One virtual forwarder within a GLBP group is elected as active virtual forwarder for a specified virtual MAC address, and it is responsible for forwarding packets sent to that MAC address. Multiple active virtual forwarders can exist for each GLBP group.

AVG—active virtual gateway. One virtual gateway within a GLBP group is elected as the active virtual gateway, and is responsible for the operation of the protocol.

checkpointing—The process of saving or synchronizing of client-specific state data that will be transferred to a peer client on a remote unit for redundancy switchover and to the local router for process restarts. Once a valid checkpointing session is established, the checkpointed state data is guaranteed to be delivered to the remote peer client in order and without corruption.

GLBP gateway—Gateway Load Balancing Protocol gateway. A router or gateway running GLBP. Each GLBP gateway may participate in one or more GLBP groups.

GLBP group—Gateway Load Balancing Protocol group. One or more GLBP gateways configured with the same GLBP group number on connected Ethernet interfaces.

ISSU—In Service Software Upgrade. A process that allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues. In most networks, planned software upgrades are a significant cause of downtime. ISSU allows Cisco IOS software to be modified while packet forwarding continues, which increases network availability and reduces downtime caused by planned software upgrades.

NSF—Nonstop Forwarding. The ability of a router to continue to forward traffic to a router that may be recovering from a failure. Also, the ability of a router recovering from a failure to continue to correctly forward traffic sent to it by a peer.

RP—Route Processor. A generic term for the centralized control unit in a chassis. Platforms usually use a platform-specific term, such as RSP on the Cisco 7500, the PRE on the Cisco 10000, or the SUP+MSFC on the Cisco 7600.

RPR—Route Processor Redundancy. RPR provides an alternative to the High System Availability (HSA) feature. HSA enables a system to reset and use a standby Route Processor (RP) if the active RP fails. Using RPR, you can reduce unplanned downtime because RPR enables a quicker switchover between an active and standby RP if the active RP experiences a fatal error.

RPR+—An enhancement to RPR in which the standby RP is fully initialized.

SSO—Stateful Switchover. Enables applications and features to maintain state information between an active and standby unit.

standby RP—An RP that has been fully initialized and is ready to assume control from the active RP should a manual or fault-induced switchover occur.

switchover—An event in which system control and routing protocol execution are transferred from the active RP to the standby RP. Switchover may be a manual operation or may be induced by a hardware or software fault. Switchover may include transfer of the packet forwarding function in systems that combine system control and packet forwarding in an indivisible unit.

vIP—virtual IP address. An IPv4 address. There must be only one virtual IP address for each configured GLBP group. The virtual IP address must be configured on at least one GLBP group member. Other GLBP group members can learn the virtual IP address from hello messages.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring HSRP

First Published: May 2, 2005
Last Updated: May 5, 2008

The Hot Standby Router Protocol (HSRP) is a First Hop Redundancy Protocol (FHRP) designed to allow for transparent fail-over of the first-hop IP router. HSRP provides high network availability by providing first-hop routing redundancy for IP hosts on Ethernet, Fiber Distributed Data Interface (FDDI), Bridge-Group Virtual Interface (BVI), LAN Emulation (LANE), or Token Ring networks configured with a default gateway IP address. HSRP is used in a group of routers for selecting an active router and a standby router. In a group of router interfaces, the active router is the router of choice for routing packets; the standby router is the router that takes over when the active router fails or when preset conditions are met.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for HSRP" section on page 59.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Restrictions for HSRP

- HSRP is designed for use over multiaccess, multicast, or broadcast capable Ethernet LANs. HSRP is not intended as a replacement for existing dynamic protocols.
- HSRP is configurable on Ethernet, FDDI, BVI, LANE, or Token Ring interfaces. Token Ring interfaces allow up to three Hot Standby groups each, the group numbers being 0, 1, and 2.
- The Cisco 2500 series, Cisco 3000 series, Cisco 4000 series, and Cisco 4500 routers that use Lance Ethernet hardware do not support multiple Hot Standby groups on a single Ethernet interface. The Cisco 800 series and Cisco 1600 series that use PQUICC Ethernet hardware do not support multiple Hot Standby groups on a single Ethernet interface. You can configure a workaround solution by using the **standby use-bia** interface configuration command, which uses the burned-in address of the interface as its virtual MAC address, instead of the preassigned MAC address.
- HSRP support for Bidirectional Forwarding Detection (BFD) is not available for all platforms and interfaces. In Cisco IOS Release 12.4(11)T, this feature was introduced on Cisco 7200 series, Cisco 7600 series, and Gigabit Switch Routers (GSRs).

Information About HSRP

To configure HSRP, you should understand the following concepts:

HSRP Operation

Most IP hosts have an IP address of a single router configured as the default gateway. When HSRP is used, the HSRP virtual IP address is configured as the host’s default gateway instead of the IP address of the router.

HSRP is useful for hosts that do not support a router discovery protocol (such as ICMP Router Discovery Protocol [IRDP]) and cannot switch to a new router when their selected router reloads or loses power. Because existing TCP sessions can survive the failover, this protocol also provides a more transparent recovery for hosts that dynamically choose a next hop for routing IP traffic.

When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among a group of routers running HSRP. The address of this HSRP group is referred to as the *virtual IP address*. One of these devices is selected by the protocol to be the active router. The active router receives and routes packets destined for the MAC address of the group. For *n* routers running HSRP, *n* + 1 IP and MAC addresses are assigned.

HSRP detects when the designated active router fails, at which point a selected standby router assumes control of the MAC and IP addresses of the Hot Standby group. A new standby router is also selected at that time.

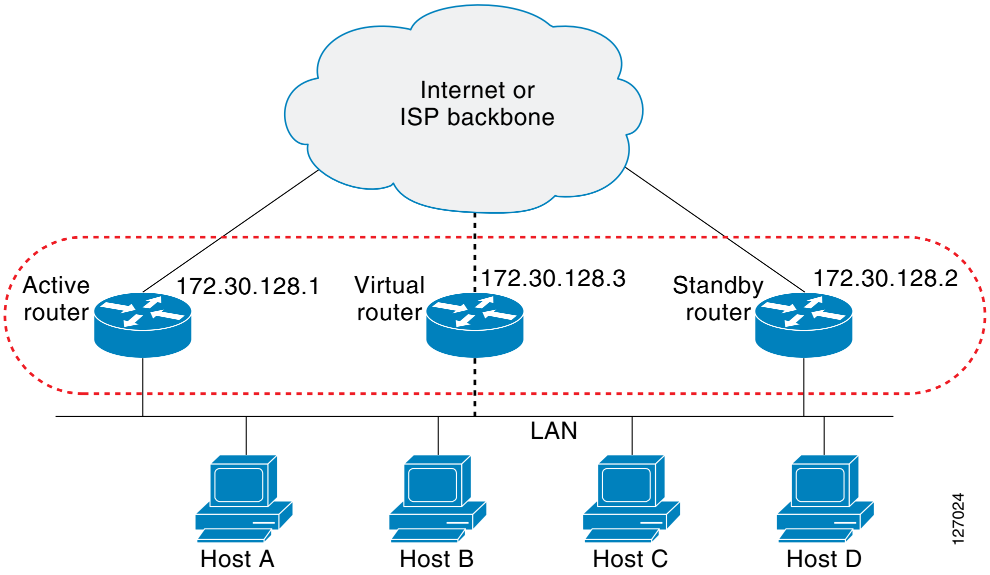
HSRP uses a priority mechanism to determine which HSRP configured router is to be the default active router. To configure a router as the active router, you assign it a priority that is higher than the priority of all the other HSRP-configured routers. The default priority is 100, so if you configure just one router to have a higher priority, that router will be the default active router.

Devices that are running HSRP send and receive multicast User Datagram Protocol (UDP)-based hello messages to detect router failure and to designate active and standby routers. When the active router fails to send a hello message within a configurable period of time, the standby router with the highest priority becomes the active router. The transition of packet forwarding functions between routers is completely transparent to all hosts on the network.

You can configure multiple Hot Standby groups on an interface, thereby making fuller use of redundant routers and load sharing.

Figure 1 shows a network configured for HSRP. By sharing a virtual MAC address and IP address, two or more routers can act as a single *virtual router*. The virtual router does not physically exist but represents the common default gateway for routers that are configured to provide backup to each other. You do not need to configure the hosts on the LAN with the IP address of the active router. Instead, you configure them with the IP address (virtual IP address) of the virtual router as their default gateway. If the active router fails to send a hello message within the configurable period of time, the standby router takes over and responds to the virtual addresses and becomes the active router, assuming the active router duties.

Figure 1 HSRP Topology



HSRP is supported over Inter-Switch Link (ISL) encapsulation. See the "Virtual LANs" section of the "Configuring Routing Between VLANs" chapter in the *Cisco IOS LAN Switching Configuration Guide,* Release 12.4.

HSRP Benefits

Redundancy

HSRP employs a redundancy scheme that is time proven and deployed extensively in large networks.

Fast Failover

HSRP provides transparent fast failover of the first-hop router.

Preemption

Preemption allows a standby router to delay becoming active for a configurable amount of time.

Authentication

HSRP message digest 5 (MD5) algorithm authentication protects against HSRP-spoofing software and uses the industry-standard MD5 algorithm for improved reliability and security.

HSRP Groups and Group Attributes

By using the command-line interface (CLI), group attributes can be applied to:

- A single HSRP group—performed in interface configuration mode and applies to a group.
- All groups on the interface—performed in interface configuration mode and applies to all groups on the interface.
- All groups on all interfaces—performed in global configuration mode and applies to all groups on all interfaces.

HSRP Addressing

HSRP routers communicate between each other by exchanging HSRP hello packets. These packets are sent to the destination IP multicast address 224.0.0.2 (reserved multicast address used to communicate to all routers) on UDP port 1985. The active router sources hello packets from its configured IP address and the HSRP virtual MAC address while the standby router sources hellos from its configured IP address and the interface MAC address, which may or may not be the Burned-In MAC address (BIA).

Because hosts are configured with their default gateway as the HSRP virtual IP address, hosts must communicate with the MAC address associated with the HSRP virtual IP address. This MAC address will be a virtual MAC address composed of 0000.0C07.ACxy, where *xy* is the HSRP group number in hexadecimal based on the respective interface. For example, HSRP group one will use the HSRP virtual MAC address of 0000.0C07.AC01. Hosts on the adjoining LAN segment use the normal Address Resolution Protocol (ARP) process to resolve the associated MAC addresses.

Token Ring interfaces use functional addresses for the HSRP MAC address. Functional addresses are the only general multicast mechanism available. There are a limited number of Token Ring functional addresses available, and many of them are reserved for other functions. The following are the only three addresses available for use with HSRP:

- c000.0001.0000 (group 0)
- c000.0002.0000 (group 1)
- c000.0004.0000 (group 2)

Thus, only three HSRP groups may be configured on Token Ring interfaces unless the **standby use-bia** interface configuration command is configured.

HSRP version 2 uses the new IP multicast address 224.0.0.102 to send hello packets instead of the multicast address of 224.0.0.2, which is used by version 1. This new multicast address allows Cisco Group Management Protocol (CGMP) leave processing to be enabled at the same time as HSRP.

HSRP version 2 permits an expanded group number range, 0 to 4095, and consequently uses a new MAC address range 0000.0C9F.F000 to 0000.0C9F.FFFF.

HSRP Support for IPv6

Most IPv4 hosts have a single router's IP address configured as the default gateway. When HSRP is used, then the HSRP virtual IP address is configured as the host's default gateway instead of the router's IP address. Simple load sharing may be achieved by using two HSRP groups and configuring half the hosts with one virtual IP address and half the hosts with the other virtual IP address.

In contrast, IPv6 hosts learn of available IPv6 routers through IPv6 neighbor discovery Router Advertisement (RA) messages. These are multicast periodically, or may be solicited by hosts. HSRP is designed to provide only a virtual first hop for IPv6 hosts.

An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number, and a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address. Periodic RAs are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active. These RAs stop after a final RA is sent when the group leaves the active state.

Periodic RAs for the interface link-local address stop after a final RA is sent while at least one virtual IPv6 link-local address is configured on the interface. No restrictions occur for the interface IPv6 link-local address other than that mentioned for the RAs. Other protocols continue to receive and send packets to this address.

HSRP uses a priority mechanism to determine which HSRP configured router is to be the default active router. To configure a router as the active router, you assign it a priority that is higher than the priority of all the other HSRP-configured routers. The default priority is 100, so if you configure just one router to have a higher priority, that router will be the default active router.

For more information see the "Configuring First Hop Redundancy Protocols in IPv6" chapter of the *Cisco IOS IPv6 Configuration Guide*.

HSRP Messages and States

Routers configured with HSRP exchange three types of multicast messages:

- Hello—The hello message conveys to other HSRP routers the HSRP priority and state information of the router.
- Coup—When a standby router wants to assume the function of the active router, it sends a coup message.
- Resign—A router that is the active router sends this message when it is about to shut down or when a router that has a higher priority sends a hello or coup message.

At any time, a router configured with HSRP is in one of the following states:

- Active—The router is performing packet-transfer functions.
- Standby—The router is prepared to assume packet-transfer functions if the active router fails.
- Speak—The router is sending and receiving hello messages.
- Listen—The router is receiving hello messages.
- Init or Disabled—The router is not yet ready or able to participate in HSRP, possibly because the associated interface is not up. HSRP groups configured on other routers on the network that are learned via snooping are displayed as being in the Init state. Locally configured groups with an interface that is down or groups without a specified interface IP address appear in the Init state.

HSRP and ARP

HSRP also works when the hosts are configured for proxy ARP. When the active HSRP router receives an ARP request for a host that is not on the local LAN, the router replies with the MAC address of the virtual router. If the active router becomes unavailable or its connection to the remote LAN goes down, the router that becomes the active router receives packets addressed to the virtual router and transfers them accordingly. If the Hot Standby state of the interface is not active, proxy ARP responses are suppressed.

HSRP Object Tracking

Object tracking separates the tracking mechanism from HSRP and creates a separate standalone tracking process that can be used by any other process as well as HSRP. The priority of a device can change dynamically when it has been configured for object tracking and the object that is being tracked goes down. Examples of objects that can be tracked are the line protocol state of an interface or the reachability of an IP route. If the specified object goes down, the HSRP priority is reduced.

A client process, such as HSRP, Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP), can now register its interest in tracking objects and then be notified when the tracked object changes state.

For more information about Object Tracking, see the *Configuring Enhanced Object Tracking* document.

HSRP Group Shutdown

The FHRP—HSRP Group Shutdown feature enables you to configure an HSRP group to become disabled (its state changed to Init) instead of having its priority decremented when a tracked object goes down. Use the **standby track** command with the **shutdown** keyword to configure HSRP group shutdown.

If an object is already being tracked by an HSRP group, you cannot change the configuration to use the HSRP Group Shutdown feature. You must first remove the tracking configuration using the **no standby track** command and then reconfigure it using the **standby track** command with the **shutdown** keyword.

The following example shows how to change the configuration of a tracked object to include the HSRP Group Shutdown feature:

```
no standby 1 track 101 decrement 10
standby 1 track 101 shutdown
```

HSRP Support for MPLS VPNs

HSRP support for a Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) interface is useful when an Ethernet LAN is connected between two provider edge (PE) routers with either of the following conditions:

- A customer edge (CE) router with a default route to the HSRP virtual IP address
- One or more hosts with the HSRP virtual IP address configured as the default gateway

Each VPN is associated with one or more VPN routing/forwarding (VRF) instances. A VRF consists of the following elements:

- IP routing table
- Cisco Express Forwarding (CEF) table
- Set of interfaces that use the CEF forwarding table
- Set of rules and routing protocol parameters to control the information in the routing tables

VPN routing information is stored in the IP routing table and the CEF table for each VRF. A separate set of routing and CEF tables is maintained for each VRF. These tables prevent information from being forwarded outside a VPN and also prevent packets that are outside a VPN from being forwarded to a router within the VPN.

HSRP adds ARP entries and IP hash table entries (aliases) using the default routing table instance. However, a different routing table instance is used when VRF forwarding is configured on an interface, causing ARP and ICMP echo requests for the HSRP virtual IP address to fail.

HSRP support for MPLS VPNs ensures that the HSRP virtual IP address is added to the correct IP routing table and not to the default routing table.

HSRP Multiple Group Optimization

Increasingly, many hundreds of subinterfaces are being configured on the same physical interface, with each subinterface having its own HSRP group. The negotiation and maintenance of multiple HSRP groups can have a detrimental impact on network traffic and CPU utilization.

Only one HSRP group is required on a physical interface for the purposes of electing active and standby routers. This group is known as the *master* group. Other HSRP groups may be created on each subinterface and linked to the master group via the group name. These linked HSRP groups are known as *client* or *follow* groups.

The HSRP group state of the client groups follows that of the master group. Client groups do not participate in any sort of router election mechanism.

Client groups send periodic messages in order to refresh their virtual MAC addresses in switches and learning bridges. The refresh message may be sent at a much lower frequency compared with the protocol election messages sent by the master group.

HSRP—ISSU

The In Service Software Upgrade (ISSU) process allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues. In most networks, planned software upgrades are a significant cause of downtime. ISSU allows Cisco IOS software to be modified while packet forwarding continues, which increases network availability and reduces downtime caused by planned software upgrades. This document provides information about ISSU concepts and describes the steps taken to perform ISSU in a system.

For detailed information about ISSU, see the *Cisco IOS In Service Software Upgrade Process* document at the following URL:

http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-inserv_updg.html

For detailed information about ISSU on the 7600 series routers, see the *ISSU and eFSU on Cisco 7600 Series Routers* document at the following URL:

http://www.cisco.com/en/US/docs/routers/7600/ios/12.2SR/configuration/efsuovrw.html

For detailed information about ISSU on Cisco Catalyst 4500 series switches, see the "Configuring the Cisco IOS In Service Software Upgrade Process" chapter of the *Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide*, Release 12.2(31)SGA at the following URL:

http://www.cisco.com/en/US/docs/switches/lan/catalyst4500/12.2/31sga/configuration/guide/issu.html

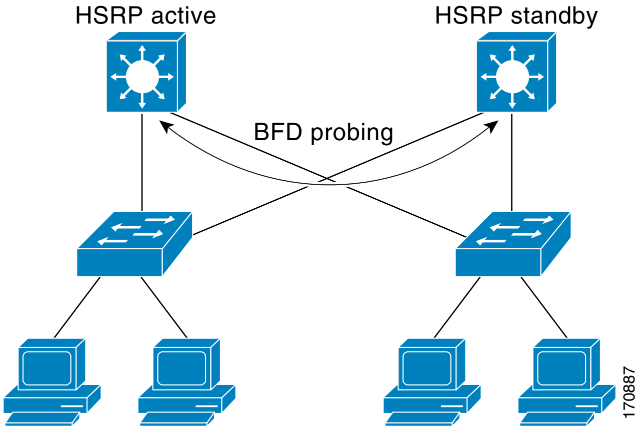
HSRP BFD Peering

The HSRP BFD Peering feature introduces BFD in the HSRP group member health monitoring system. Previously, group member monitoring relied exclusively on HSRP multicast messages, which are relatively large and consume CPU memory to produce and check. In architectures where a single interface hosts a large number of groups, there is a need for a protocol with low CPU memory consumption and processing overhead. BFD addresses this issue and offers sub-second health monitoring (failure detection in milliseconds) with relatively low CPU impact. This feature is enabled by default. The HSRP standby router learns the real IP address of the HSRP active router from the HSRP Hello messages. The standby router will register as a BFD client and ask to be notified if the active router becomes unavailable.

BFD provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes. BFD is a detection protocol that you enable at the interface and routing protocol levels. Cisco supports the BFD asynchronous mode, which depends on the sending of BFD control packets between two systems to activate and maintain BFD neighbor sessions between routers. Therefore, in order for a BFD session to be created, you must configure BFD on both systems (or BFD peers). Once BFD has been enabled on the interfaces and at the router level for HSRP, a BFD session is created, BFD timers are negotiated, and the BFD peers will begin to send BFD control packets to each other at the negotiated interval.

BFD provides fast BFD peer failure detection times independently of all media types, encapsulations, topologies, and routing protocols BGP, EIGRP, HSRP, IS-IS, and OSPF. By sending rapid failure detection notices to the routing protocols in the local router to initiate the routing table recalculation process, BFD contributes to greatly reduce overall network convergence time. Figure 2 shows a simple network with two routers running HSRP and BFD.

Figure 2 HSRP BFD Peering



For more information on BFD, see the “Bidirectional Forwarding Detection” chapter in the *Cisco IOS IP Routing Configuration Guide* at the following URL:

http://www.cisco.com/en/US/docs/ios/iproute/configuration/guide/irp_bfd.html

How to Configure HSRP

This section contains the following procedures:

- Enabling HSRP, page 10 (required)
- Delaying the Initialization of HSRP on an Interface, page 11 (optional)
- Configuring HSRP Priority and Preemption, page 13 (required)
- Configuring HSRP Object Tracking, page 15 (optional)
- Configuring HSRP Authentication, page 17 (optional)
- Customizing HSRP, page 25 (optional)
- Configuring Multiple HSRP Groups for Load Balancing, page 27 (optional)
- Improving CPU and Network Performance with HSRP Multiple Group Optimization, page 28 (optional)
- Enabling HSRP Support for ICMP Redirects, page 30 (optional)
- Configuring HSRP Virtual MAC Addresses or BIA MAC Addresses, page 34 (optional)
- Linking IP Redundancy Clients to HSRP Groups, page 35 (optional)

- Changing to HSRP Version 2, page 37 (optional)
- Configuring SSO HSRP, page 39 (optional)
- Enabling HSRP MIB Traps, page 42 (optional)
- Configuring HSRP BFD Peering, page 43

Enabling HSRP

Perform this task to enable HSRP.

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the virtual IP address for the Hot Standby group. For HSRP to elect a designated router, you must configure the virtual IP address for at least one of the routers in the group; it can be learned on the other routers in the group.

Prerequisites

You can configure many attributes in HSRP such as authentication, timers, priority, and preemption. It is best practice to configure the attributes first before enabling the HSRP group.

This practice avoids authentication error messages and unexpected state changes in other routers that can occur if the group is enabled first and then there is a long enough delay (one or two hold times) before the other configuration is entered.

We recommend that you always specify an HSRP IP address.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **end**
7. **show standby** [**all**] [**brief**]
8. **show standby** *type number* [*group-number* | **all**] [**brief**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *type number* **Example:** `Router(config)# interface ethernet 0`	Configures an interface type and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 172.16.6.5 255.255.255.0`	Configures an IP address for an interface.
Step 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 1 ip 172.16.6.100`	Activates HSRP. • If you do not configure a group number, it defaults to 0. The group number range is from 0 to 255 for HSRP version 1 and from 0 to 4095 for HSRP version 2. • The *ip-address* is the virtual IP address of the virtual router. For HSRP to elect a designated router, you must configure the virtual IP address for at least one of the routers in the group; it can be learned on the other routers in the group.

Step 6	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 7	**show standby** [**all**] [**brief**] **Example:** `Router# show standby`	(Optional) Displays HSRP information. • This command displays information for each group. The **all** option display groups that are learned or that do not have the **standby ip** command configured.
Step 8	**show standby** *type number* [*group-number* | **all**] [**brief**] **Example:** `Router# show standby ethernet 0`	(Optional) Displays HSRP information about specific groups or interfaces.

Delaying the Initialization of HSRP on an Interface

Perform this task to delay the initialization of HSRP on an interface.

The **standby delay** command is used to delay HSRP initialization either after a reload and/or after an interface comes up. This configuration allows the interface and router time to settle down after the interface up event and helps prevent HSRP state flapping.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **ip address** *ip-address mask*
5. **standby delay minimum** *min-delay* **reload** *reload-seconds*
6. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
7. **end**
8. **show standby delay** [*type number*]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`*
	Example:
	`Router(config-if)# ip address 10.0.0.1 255.255.255.0`
		Specifies an IP address for an interface.

Step 5	**`standby delay minimum`** *`min-delay`* **`reload`** *`reload-seconds`*
	Example:
	`Router(config-if)# standby delay minimum 20 reload 25`
		(Optional) Configures the delay period before the initialization of HSRP groups.
		• The *min-delay* value is the minimum time (in seconds) to delay HSRP group initialization after an interface comes up. This minimum delay period applies to all subsequent interface events.
		• The *reload-seconds* value is the time period to delay after the router has reloaded. This delay period applies only to the first interface-up event after the router has reloaded.

Step 6	**`standby`** `[`*`group-number`*`]` **`ip`** `[`*`ip-address`* `[`**`secondary`**`]]`
	Example:
	`Router(config-if)# standby 1 ip 10.0.0.3 255.255.255.0`
		Activates HSRP.

	Command or Action	Purpose

Step 7	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 8	**show standby delay** [*type number*] **Example:** `Router# show standby delay`	(Optional) Displays HSRP information about delay periods.

Troubleshooting Tips

We recommend that you use the **standby delay minimum reload** command if the **standby timers** command is configured in milliseconds or if HSRP is configured on a VLAN interface of a switch.

Configuring HSRP Priority and Preemption

Perform this task to configure HSRP priority and preemption.

HSRP Priority and Preemption

Preemption enables the HSRP router with the highest priority to immediately become the active router. Priority is determined first by the configured priority value, and then by the IP address. In case of ties, the primary IP addresses are compared, and the higher IP address has priority. In each case, a higher value is of greater priority. If you do not use the **standby preempt** interface configuration command in the configuration for a router, that router will not become the active router, even if its priority is higher than all other routers.

A standby router with equal priority but a higher IP address will not preempt the active router.

When a router first comes up, it does not have a complete routing table. You can set a preemption delay that allows preemption to be delayed for a configurable time period. This delay period allows the router to populate its routing table before becoming the active router.

How Object Tracking Affects the Priority of an HSRP Router

The priority of a device can change dynamically if it has been configured for object tracking and the object that is being tracked goes down. The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to HSRP, either immediately or after a specified delay. The object values are reported as either up or down. Examples of objects that can be tracked are the line protocol state of an interface or the reachability of an IP route. If the specified object goes down, the HSRP priority is reduced. The HSRP router with the higher priority can now become the active router if it has the **standby preempt** command configured. See the "Configuring HSRP Object Tracking" section on page 15 for more information on object tracking.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **priority** *priority*
6. **standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}]
7. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
8. **end**
9. **show standby** [**all**] [**brief**]
10. **show standby** *type number* [*group-number* | **all**] [**brief**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`*
	Example:
	`Router(config-if)# ip address 10.0.0.1 255.255.255.0`
		Specifies an IP address for an interface.

Step 5	**`standby`** [*`group-number`*] **`priority`** *`priority`*
	Example:
	`Router(config-if)# standby 1 priority 110`
		Configures HSRP priority.
		• The default priority is 100.

Step 6	**`standby`** [*`group-number`*] **`preempt`** [**`delay`** {**`minimum`** *`delay`* | **`reload`** *`delay`* | **`sync`** *`delay`*}] **Example:** `Router(config-if)# standby 1 preempt delay minimum 380`	Configures HSRP preemption and preemption delay. • The default delay period is 0 seconds; if the router wants to preempt, it will do so immediately. By default, the router that comes up later becomes the standby.
Step 7	**`standby`** [*`group-number`*] **`ip`** [*`ip-address`* [**`secondary`**]] **Example:** `Router(config-if)# standby 1 ip 10.0.0.3 255.255.255.0`	Activates HSRP.

	Command or Action	Purpose

Step 8	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 9	**show standby** [**all**] [**brief**] **Example:** `Router# show standby`	(Optional) Displays HSRP information. • This command displays information for each group. The **all** option display groups that are learned or that do not have the **standby ip** command configured.
Step 10	**show standby** *type number* [*group-number* | **all**] [**brief**] **Example:** `Router# show standby ethernet 0/1`	(Optional) Displays HSRP information about specific groups or interfaces.

Configuring HSRP Object Tracking

Perform this task to configure HSRP to track an object and change the HSRP priority based on the state of the object.

Each tracked object is identified by a unique number that is specified on the tracking CLI. Client processes use this number to track a specific object.

For more information on object tracking, see the "Configuring Enhanced Object Tracking" document.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* **interface** *type number* {**line-protocol** | **ip routing**}
4. **exit**
5. **interface** *type number*
6. **standby** [*group-number*] **track** *object-number* [**decrement** *priority-decrement*]
7. **standby** [*group-number*] **track** *object-number* **shutdown**
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **end**
10. **show track** [*object-number* | **brief**] [**interface** [**brief**] | **ip route** [**brief**] | **resolution** | **timers**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`track`** *`object-number`* **`interface`** *`type number`* {**`line-protocol`** | **`ip routing`**}
	Example:
	`Router(config)# track 100 interface serial2/0 line-protocol`
		Configures an interface to be tracked and enters tracking configuration mode.

Step 4	**`exit`**
	Example:
	`Router(config-track)# exit`
		Returns to global configuration mode.

Step 5	**`interface`** *`type number`*
	Example:
	`Router(config)# interface ethernet 2`
		Configures an interface type and enters interface configuration mode.

Step 6	**`standby`** [*`group-number`*] **`track`** *`object-number`* [**`decrement`** *`priority-decrement`*] **Example:** `Router(config-if)# standby 1 track 100 decrement 20`	Configures HSRP to track an object and change the Hot Standby priority on the basis of the state of the object. - By default, the priority of the router is decreased by 10 if a tracked object goes down. Use the **decrement** *priority-decrement* keyword and argument combination to change the default behavior. - When multiple tracked objects are down and *priority-decrement* values have been configured, these configured priority decrements are cumulative. If tracked objects are down, but none of them were configured with priority decrements, the default decrement is 10 and it is cumulative.

Step 7	**`standby`** [*`group-number`*] **`track`** *`object-number`* **`shutdown`**
	Example:
	`Router(config-if)# standby 1 track 100 shutdown`
		(Optional) Configures HSRP to track an object and disable the HSRP group when the tracked object goes down.
		- Use the **shutdown** keyword to disable the HRSP group on the router when the tracked object goes down.

Step 8	**`standby`** [*`group-number`*] **`ip`** [*`ip-address`* [**`secondary`**]] **Example:** `Router(config-if)# standby 1 ip 10.10.10.0`	Activates HSRP. - The default group number is 0. The group number range is from 0 to 255 for HSRP version 1 and from 0 to 4095 for HSRP version 2.

	Command or Action	Purpose

Step 9	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 10	**show track** [*object-number* | **brief**] [**interface** [**brief**]| **ip route** [**brief**]| **resolution** | **timers**] **Example:** `Router# show track 100 interface`	Displays tracking information.

Configuring HSRP Authentication

HSRP ignores unauthenticated HSRP protocol messages. The default authentication type is text authentication.

The following sections describe configuration tasks for HSRP authentication. The task you perform depends on whether you want to use text authentication, a simple MD5 key string, or MD5 key chains for authentication.

- Configuring HSRP MD5 Authentication Using a Key String, page 18
- Configuring HSRP MD5 Authentication Using a Key Chain, page 20
- Troubleshooting HSRP MD5 Authentication, page 22
- Configuring HSRP Text Authentication, page 23

How HSRP MD5 Authentication Works

Before the introduction of HSRP MD5 authentication, HSRP authenticated protocol packets with a simple plain text string. HSRP MD5 authentication is an enhancement to generate an MD5 digest for the HSRP portion of the multicast HSRP protocol packet. This functionality provides added security and protects against the threat from HSRP-spoofing software.

MD5 authentication provides greater security than the alternative plain text authentication scheme. MD5 authentication allows each HSRP group member to use a secret key to generate a keyed MD5 hash that is part of the outgoing packet. A keyed hash of an incoming packet is generated and if the hash within the incoming packet does not match the generated hash, the packet is ignored.

The key for the MD5 hash can be either given directly in the configuration using a key string or supplied indirectly through a key chain.

HSRP has two authentication schemes:

- Plain text authentication
- MD5 authentication

HSRP authentication protects against false HSRP hello packets causing a denial-of-service attack. For example, Router A has a priority of 120 and is the active router. If a host sends spoof HSRP hello packets with a priority of 130, then Router A stops being the active router. If Router A has authentication configured such that the spoof HSRP hello packets are ignored, Router A will remain the active router.

HSRP packets will be rejected in any of the following cases:

- The authentication schemes differ on the router and in the incoming packets.
- MD5 digests differ on the router and in the incoming packet.
- Text authentication strings differ on the router and in the incoming packet.

Benefits of HSRP MD5 Authentication

- Protects against HSRP-spoofing software
- Uses the industry-standard MD5 algorithm for improved reliability and security

Restrictions

Text authentication cannot be combined with MD5 authentication for an HSRP group at any one time. When MD5 authentication is configured, the text authentication field in HSRP hello messages is set to all zeroes on transmit and ignored on receipt, provided the receiving router also has MD5 authentication enabled.

Configuring HSRP MD5 Authentication Using a Key String

Perform this task to configure HSRP MD5 authentication using a key string.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **standby** [*group-number*] **priority** *priority*
6. **standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}]
7. **standby** [*group-number*] **authentication md5 key-string** [**0** | **7**] *key* [**timeout** *seconds*]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. Repeat Steps 1 through 8 on each router that will communicate.
10. **end**
11. **show standby**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet0/1`	Configures an interface type and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 5	**standby** [*group-number*] **priority** *priority* **Example:** `Router(config-if)# standby 1 priority 110`	Configures HSRP priority.
Step 6	**standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}] **Example:** `Router(config-if)# standby 1 preempt`	Configures HSRP preemption.
Step 7	**standby** [*group-number*] **authentication md5 key-string** [**0** | **7**] *key* [**timeout** *seconds*] **Example:** `Router(config-if)# standby 1 authentication md5 key-string d00b4r987654321a timeout 30`	Configures an authentication string for HSRP MD5 authentication. • The *key* argument can be up to 64 characters in length and it is recommended that at least 16 characters be used. • No prefix to the *key* argument or specifying **0** means the key will be unencrypted. • Specifying **7** means the key will be encrypted. The key-string authentication key will automatically be encrypted if the **service password-encryption** global configuration command is enabled. • The timeout value is the period of time that the old key string will be accepted to allow configuration of all routers in a group with a new key.

	Command	Purpose
Step 8	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 1 ip 10.0.0.3`	Activates HSRP.
Step 9	Repeat Steps 1 through 8 on each router that will communicate.	—

Step 10	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 11	**show standby** **Example:** `Router# show standby`	(Optional) Displays HSRP information. • Use this command to verify your configuration. The key string or key chain will be displayed if configured.

Troubleshooting Tips

If you are changing a key string in a group of routers, change the active router last to prevent any HSRP state change. The active router should have its key string changed no later than one holdtime period, specified by the **standby timers** interface configuration command, after the non-active routers. This procedure ensures that the non-active routers do not time out the active router.

Configuring HSRP MD5 Authentication Using a Key Chain

Perform this task to configure HSRP MD5 authentication using a key chain. Key chains allow a different key string to be used at different times according to the key chain configuration. HSRP will query the appropriate key chain to obtain the current live key and key ID for the specified key chain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **key chain** *name-of-chain*
4. **key** *key-id*
5. **key-string** *string*
6. **exit**
7. **interface** *type number*
8. **ip address** *ip-address mask* [**secondary**]
9. **standby** [*group-number*] **priority** *priority*
10. **standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}]
11. **standby** [*group-number*] **authentication md5 key-chain** *key-chain-name*
12. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
13. Repeat Steps 1 through 12 on each router that will communicate.

14. **end**
15. **show standby**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**key chain** *name-of-chain* **Example:** `Router(config)# key chain hsrp1`	Enables authentication for routing protocols and identifies a group of authentication keys.
Step 4	**key** *key-id* **Example:** `Router(config-keychain)# key 100`	Identifies an authentication key on a key chain. - The *key-id* must be a number.
Step 5	**key-string** *string* **Example:** `Router(config-keychain-key)# key-string mno172`	Specifies the authentication string for a key. - The *string* can be 1 to 80 uppercase or lowercase alphanumeric characters; the first character cannot be a number.
Step 6	**exit** **Example:** `Router(config-keychain-key)# exit`	Returns to global configuration mode.
Step 7	**interface** *type number* **Example:** `Router(config)# interface Ethernet0/1`	Configures an interface type and enters interface configuration mode.
Step 8	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.21.8.32 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 9	**standby** [*group-number*] **priority** *priority* **Example:** `Router(config-if)# standby 1 priority 110`	Configures HSRP priority.

	Command	Purpose
Step 10	**standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}] **Example:** `Router(config-if)# standby 1 preempt`	Configures HSRP preemption.
Step 11	**standby** [*group-number*] **authentication md5 key-chain** *key-chain-name* **Example:** `Router(config-if)# standby 1 authentication md5 key-chain hsrp1`	Configures an authentication MD5 key chain for HSRP MD5 authentication. • The key chain name must match the name specified in Step 3.
Step 12	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 1 ip 10.21.8.12`	Activates HSRP.
Step 13	Repeat Steps 1 through 12 on each router that will communicate.	—

Step 14	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 15	**show standby** **Example:** `Router# show standby`	(Optional) Displays HSRP information. • Use this command to verify your configuration. The key string or key chain will be displayed if configured.

Troubleshooting HSRP MD5 Authentication

Perform this task if HSRP MD5 authentication is not operating correctly.

SUMMARY STEPS

1. **enable**
2. **debug standby errors**

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`debug standby errors`**
	Example:
	`Router# debug standby errors`
		Displays error messages related to HSRP.
		• Error messages will be displayed for each packet that fails to authenticate, so use this command with care.
		• See the "Examples" section for an example of the type of error messages displayed when two routers are not authenticating.

Examples

In the following example, Router A has MD5 text string authentication configured, but Router B has the default text authentication:

```
Router# debug standby errors

A:Jun 16 12:14:50.337:HSRP:Et0/1 Grp 0 Auth failed for Hello pkt from 10.21.0.5, MD5
confgd but no tlv
B:Jun 16 12:16:34.287:HSRP:Et0/1 Grp 0 Auth failed for Hello pkt from 10.21.0.4, Text auth
failed
```

In the following example, both Router A and Router B have different MD5 authentication strings:

```
Router# debug standby errors

A:Jun 16 12:19:26.335:HSRP:Et0/1 Grp 0 Auth failed for Hello pkt from 10.21.0.5, MD5 auth
failed
B:Jun 16 12:18:46.280:HSRP:Et0/1 Grp 0 Auth failed for Hello pkt from 10.21.0.4, MD5 auth
failed
```

Configuring HSRP Text Authentication

Perform this task to configure HSRP text authentication.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **standby** [*group-number*] **priority** *priority*
6. **standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}]
7. **standby** [*group-number*] **authentication text** *string*
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]

9. Repeat Steps 1 through 8 on each router that will communicate.
10. **end**
11. **show standby**

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables higher privilege levels, such as privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-if)# ip address 10.0.0.1 255.255.255.0`
		Specifies a primary or secondary IP address for an interface.

Step 5	**`standby`** `[`*`group-number`*`]` **`priority`** *`priority`*
	Example:
	`Router(config-if)# standby 1 priority 110`
		Configures HSRP priority.

Step 6	**`standby`** `[`*`group-number`*`]` **`preempt`** `[`**`delay`** `{`**`minimum`** *`delay`* `|` **`reload`** *`delay`* `|` **`sync`** *`delay`*`}]`
	Example:
	`Router(config-if)# standby 1 preempt`
		Configures HSRP preemption.

Step 7	**`standby`** `[`*`group-number`*`]` **`authentication text`** *`string`*
	Example:
	`Router(config-if)# standby 1 authentication text authentication1`
		Configures an authentication string for HSRP text authentication.
		• The default string is cisco.

Step 8	**`standby`** `[`*`group-number`*`]` **`ip`** `[`*`ip-address`* `[`**`secondary`**`]]`
	Example:
	`Router(config-if)# standby 1 ip 10.0.0.3`
		Activates HSRP.

Step 9	Repeat Steps 1 through 8 on each router that will communicate.	—

	Command	Purpose

Step 10	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 11	**show standby** **Example:** `Router# show standby`	(Optional) Displays HSRP information. • Use this command to verify your configuration. The key string or key chain will be displayed if configured.

Customizing HSRP

Perform this task to customize HSRP parameters.

HSRP Timers

Each HSRP router maintains three timers that are used for timing hello messages: an active timer, a standby timer, and a hello timer. When a timer expires, the router changes to a new HSRP state. Routers or access servers for which timer values are not configured can learn timer values from the active or standby router. The timers configured on the active router always override any other timer settings. All routers in a Hot Standby group should use the same timer values.

For HSRP version 1, nonactive routers learn timer values from the active router, unless millisecond timer values are being used. If millisecond timer values are being used, all routers must be configured with the millisecond timer values. This rule applies if either the hello time or the hold time is specified in milliseconds. This configuration is necessary because the HSRP hello packets advertise the timer values in seconds. HSRP version 2 does not have this limitation; it advertises the timer values in milliseconds.

HSRP MAC Refresh Interval

When HSRP runs over FDDI, you can change the interval at which a packet is sent to refresh the MAC cache on learning bridges or switches. HSRP hello packets use the burned-in address (BIA) instead of the MAC virtual address. Refresh packets keep the MAC cache on switches and learning bridges current.

You can change the refresh interval on FDDI rings to a longer or shorter interval, thereby using bandwidth more efficiently. You can prevent the sending of any MAC refresh packets if you do not need them (if you have FDDI but do not have a learning bridge or switch).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*
6. **standby mac-refresh** *seconds*
7. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-if)# ip address 10.0.0.1 255.255.255.0`
		Specifies a primary or secondary IP address for an interface.

Step 5	**`standby`** `[`*`group-number`*`]` **`timers`** `[`**`msec`**`]` *`hellotime`* `[`**`msec`**`]` *`holdtime`*
	Example:
	`Router(config-if)# standby 1 timers 5 15`
		Configures the time between hello packets and the time before other routers declare the active Hot Standby router to be down.
		- Normally, the *holdtime* value is greater than or equal to three times the value of *hellotime*.
		- See the "HSRP Timers" concept in this section for more information.

Step 6	**`standby mac-refresh`** *`seconds`*
	Example:
	`Router(config-if)# standby mac-refresh 100`
		Changes the interval at which packets are sent to refresh the MAC cache when HSRP is running over FDDI.
		- This command applies to HSRP running over FDDI only.

Step 7	**`standby`** `[`*`group-number`*`]` **`ip`** `[`*`ip-address`* `[`**`secondary`**`]]`
	Example:
	`Router(config-if)# standby 1 ip 10.0.0.3`
		Activates HSRP.

Troubleshooting Tips

Some HSRP state flapping can occasionally occur if the holdtime is set to less than 250 milliseconds, and the processor is busy. It is recommended that holdtime values less than 250 milliseconds be used on Cisco 7200 platforms or better, and on Fast-Ethernet or FDDI interfaces or better. You can use the **standby delay** command to allow the interface to come up completely before HSRP initializes.

Configuring Multiple HSRP Groups for Load Balancing

Perform this task to configure multiple HSRP groups for load balancing.

Multiple HSRP groups enable redundancy and load-sharing within networks and allow redundant routers to be more fully utilized. While a router is actively forwarding traffic for one HSRP group, it can be in standby or in the listen state for another group.

If two routers are used, then Router A would be configured as active for group 1 and standby for group 2. Router B would be standby for group 1 and active for group 2. Fifty percent of the hosts on the LAN would be configured with the virtual IP address of group 1 and the remaining hosts would be configured with the virtual IP address of group 2. See the "Multiple HSRP for Load Balancing: Example" section on page 51 for a diagram and configuration example.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **standby** [*group-number*] **priority** *priority*
6. **standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}]
7. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
8. On the same router, repeat Steps 5 through 7 to configure the router attributes for different standby groups.
9. **exit**
10. Repeat Steps 3 through 9 to configure HSRP on another router.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 5	**standby** [*group-number*] **priority** *priority* **Example:** `Router(config-if)# standby 1 priority 110`	Configures HSRP priority.
Step 6	**standby** [*group-number*] **preempt** [**delay** {**minimum** *delay* | **reload** *delay* | **sync** *delay*}] **Example:** `Router(config-if)# standby 1 preempt`	Configures HSRP preemption.
Step 7	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 1 ip 10.0.0.3`	Activates HSRP.
Step 8	On the same router, repeat Steps 5 through 7 to configure the router attributes for different standby groups.	For example, Router A can be configured as an active router for group 1 and be configured for active or standby router for another HSRP group with different priority and preemption values.
Step 9	**exit** **Example:** `Router(config-if)# exit`	Exits to global configuration mode.
Step 10	Repeat Steps 3 through 9 on another router.	Configures multiple HSRP and enables load balancing on another router.

Improving CPU and Network Performance with HSRP Multiple Group Optimization

Configure the HSRP master group using the steps in the previous section, "Configuring Multiple HSRP Groups for Load Balancing."

Perform this task to configure multiple HSRP client groups.

The **standby follow** command configures an HSRP group to become an IP redundancy client of another HSRP group.

HSRP client groups follow the master HSRP with a slight, random delay so that all client groups do not change at the same time.

Active client groups use the existing FDDI MAC refresh mechanism to send hello packets at less frequent intervals than the master group. The default interval is 10 seconds and can be configured to as much as 255 seconds.

Restrictions

- Client or follow groups must be on the same physical interface as the master group.
- A client group takes its state from the group it is following. Therefore, the client group does not use its timer, priority, or preemption settings. A warning is displayed if these settings are configured on a client group:

```
Router(config-if)# standby 1 priority 110
%Warning: This setting has no effect while following another group.

Router(config-if)# standby 1 timers 5 15
% Warning: This setting has no effect while following another group.

Router(config-if)# standby 1 preempt delay minimum 300
% Warning: This setting has no effect while following another group.
```

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **standby** *group-number* **follow** *group-name*
6. **exit**
7. Repeat Steps 3 through 6 to configure additional HSRP client groups.

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet0/1`	Configures an interface type and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.

	Command or Action	Purpose

Step 5	**`standby`** *`group-number`* **`follow`** *`group-name`*
	Example:
	`Router(config-if)# standby 1 follow HSRP1`
		Configures an HSRP group as a client group.

Step 6	**`exit`**
	Example:
	`Router(config-if)# exit`
		Exits to global configuration mode.

Step 7	Repeat Steps 3 through 6.	Configures multiple HSRP client groups.

Enabling HSRP Support for ICMP Redirects

By default, HSRP filtering of ICMP redirect messages is enabled on routers running HSRP. Perform this task to reenable this feature on your router if it is disabled.

ICMP is a network layer Internet protocol that provides message packets to report errors and other information relevant to IP processing. ICMP can send error packets to a host and can send redirect packets to a host.

When running HSRP, it is important to prevent hosts from discovering the interface (or real) IP addresses of routers in the HSRP group. If a host is redirected by ICMP to the real IP address of a router, and that router later fails, then packets from the host will be lost.

ICMP redirect messages are automatically enabled on interfaces configured with HSRP. This functionality works by filtering outgoing ICMP redirect messages through HSRP, where the next hop IP address may be changed to an HSRP virtual IP address.

ICMP Redirects to Active HSRP Routers

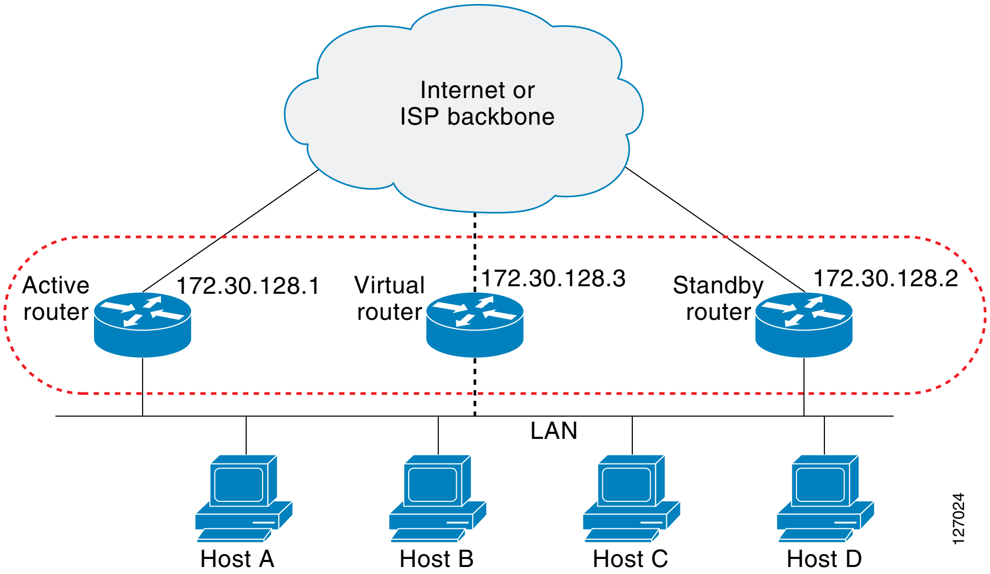
The next-hop IP address is compared to the list of active HSRP routers on that network; if a match is found, then the real next-hop IP address is replaced with a corresponding virtual IP address and the redirect message is allowed to continue.

If no match is found, then the ICMP redirect message is sent only if the router corresponding to the new next hop IP address is not running HSRP. Redirects to passive HSRP routers are not allowed (a passive HSRP router is a router running HSRP, but which contains no active HSRP groups on the interface).

For optimal operation, every router in a network that is running HSRP should contain at least one active HSRP group on an interface to that network. Every HSRP router need not be a member of the same group. Each HSRP router will snoop on all HSRP packets on the network to maintain a list of active routers (virtual IP addresses versus real IP addresses).

Consider the network shown in Figure 3, which supports the HSRP ICMP redirection filter.

Figure 3 Network Supporting the HSRP ICMP Redirection Filter



If the host wants to send a packet to another host on Net D, then it first sends it to its default gateway, the virtual IP address of HSRP group 1.

The following is the packet received from the host:

```
dest MAC         = HSRP group 1 virtual MAC
source MAC       = Host MAC
dest IP          = host-on-netD IP
source IP        = Host IP
```

Router R1 receives this packet and determines that router R4 can provide a better path to Net D, so it prepares to send a redirect message that will redirect the host to the real IP address of router R4 (because only real IP addresses are in its routing table).

The following is the initial ICMP redirect message sent by router R1:

```
dest MAC         = Host MAC
source MAC       = router R1 MAC
dest IP          = Host IP
source IP        = router R1 IP
gateway to use   = router R4 IP
```

Before this redirect occurs, the HSRP process of router R1 determines that router R4 is the active HSRP router for group 3, so it changes the next hop in the redirect message from the real IP address of router R4 to the virtual IP address of group 3. Furthermore, it determines from the destination MAC address of the packet that triggered the redirect message that the host used the virtual IP address of group 1 as its gateway, so it changes the source IP address of the redirect message to the virtual IP address of group 1.

The modified ICMP redirect message showing the two modified fields (*) is as follows:

```
dest MAC         = Host MAC
source MAC       = router R1 MAC
dest IP          = Host IP
source IP*       = HSRP group 1 virtual IP
gateway to use*  = HSRP group 3 virtual IP
```

This second modification is necessary because hosts compare the source IP address of the ICMP redirect message with their default gateway. If these addresses do not match, the ICMP redirect message is ignored. The routing table of the host now consists of the default gateway, virtual IP address of group 1, and a route to Net D through the virtual IP address of group 3.

ICMP Redirects to Passive HSRP Routers

Redirects to passive HSRP routers are not permitted. Redundancy may be lost if hosts learn the real IP addresses of HSRP routers.

In Figure 3, redirection to router R8 is not allowed because R8 is a passive HSRP router. In this case, packets from the host to Net D will first go to router R1 and then be forwarded to router R4; that is, they will traverse the network twice.

A network configuration with passive HSRP routers is considered a misconfiguration. For HSRP ICMP redirection to operate optimally, every router on the network that is running HSRP should contain at least one active HSRP group.

ICMP Redirects to Non-HSRP Routers

Redirects to routers not running HSRP on their local interface are permitted. No redundancy is lost if hosts learn the real IP address of non-HSRP routers.

In Figure 3, redirection to router R7 is allowed because R7 is not running HSRP. In this case, the next hop IP address is unchanged. The source IP address is changed dependent upon the destination MAC address of the original packet. You can specify the **no standby redirect unknown** command to stop these redirects from being sent.

Passive HSRP Router Advertisements

Passive HSRP routers send out HSRP advertisement messages both periodically and when entering or leaving the passive state. Thus, all HSRP routers can determine the HSRP group state of any HSRP router on the network. These advertisements inform other HSRP routers on the network of the HSRP interface state, as follows:

- Dormant—Interface has no HSRP groups. A single advertisement is sent once when the last group is removed.
- Passive—Interface has at least one non-active group and no active groups. Advertisements are sent out periodically.
- Active—Interface has at least one active group. A single advertisement is sent out when the first group becomes active.

You can adjust the advertisement interval and holddown time using the **standby redirect timers** command.

ICMP Redirects Not Sent

If the HSRP router cannot uniquely determine the IP address used by the host when it sends the packet that caused the redirect, the redirect message will not be sent. The router uses the destination MAC address in the original packet to make this determination. In certain configurations, such as the use of the **standby use-bia** interface configuration command specified on an interface, redirects cannot be sent.

In this case, the HSRP groups use the interface MAC address as their virtual MAC address. The router now cannot determine if the default gateway of the host is the real IP address or one of the HSRP virtual IP addresses that are active on the interface.

Using HSRP with ICMP redirects is not possible in the Cisco 800 series, Cisco 1000 series, Cisco 1600 series, Cisco 2500 series, Cisco 3000 series, and Cisco 4500 series routers because the Ethernet controller can only support one MAC address.

The IP source address of an ICMP packet must match the gateway address used by the host in the packet that triggered the ICMP packet, otherwise the host will reject the ICMP redirect packet. An HSRP router uses the destination MAC address to determine the gateway IP address of the host. If the HSRP router is using the same MAC address for multiple IP addresses then it is not possible to uniquely determine the gateway IP address of the host and the redirect message is not sent.

The following is sample output from the **debug standby events icmp** EXEC command if HSRP could not uniquely determine the gateway used by the host:

```
10:43:08: SB: ICMP redirect not sent to 20.0.0.4 for dest 30.0.0.2
10:43:08: SB: could not uniquely determine IP address for mac 00d0.bbd3.bc22
```

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **standby redirect** [**timers** *advertisement holddown*] [**unknown**]
5. **end**
6. **show standby redirect** [*ip-address*] [*interface-type interface-number*] [**active**] [**passive**] [**timers**]

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`interface` *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	`standby redirect [timers` *`advertisement holddown`*`] [unknown]`
	Example:
	`Router(config-if)# standby redirect`
		Enables HSRP filtering of ICMP redirect messages.
		• You can also use this command in global configuration mode, which enables HSRP filtering of ICMP redirect messages on all interfaces configured for HSRP.

	Command or Action	Purpose

Step 5	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 6	**show standby redirect** [*ip-address*] [*interface-type interface-number*] [**active**] [**passive**] [**timers**] **Example:** `Router# show standby redirect`	(Optional) Displays ICMP redirect information on interfaces configured with HSRP.

Configuring HSRP Virtual MAC Addresses or BIA MAC Addresses

Perform this task to configure an HSRP virtual MAC address or a burned-in address (BIA) MAC address.

A router automatically generates a virtual MAC address for each HSRP router. However, some network implementations, such as Advanced Peer-to-Peer Networking (APPN), use the MAC address to identify the first hop for routing purposes. In this case, it is often necessary to be able to specify the virtual MAC address by using the **standby mac-address** command; the virtual IP address is unimportant for these protocols.

The **standby use-bia** command was implemented to overcome the limitations of using a functional address for the HSRP MAC address on Token Ring interfaces. This command allows HSRP groups to use the BIA MAC address of an interface instead of the HSRP virtual MAC address. When HSRP runs on a multiple-ring, source-routed bridging environment and the HSRP routers reside on different rings, configuring the **standby use-bia** command can prevent confusion about the routing information field (RFI).

Restrictions

You cannot use the **standby use-bia** and **standby mac-address** commands in the same configuration; they are mutually exclusive.

The **standby use-bia** command has the following disadvantages:

- When a router becomes active the virtual IP address is moved to a different MAC address. The newly active router sends a gratuitous ARP response, but not all host implementations handle the gratuitous ARP correctly.
- Proxy ARP breaks when the **standby use-bia** command is configured. A standby router cannot cover for the lost proxy ARP database of the failed router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]

5. **standby** [*group-number*] **mac-address** *mac-address*
   or
   standby use-bia [**scope interface**]
6. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`* `[`**`secondary`**`]`
	Example:
	`Router(config-if)# ip address 172.16.6.5 255.255.255.0`
		Configures an IP address for an interface.

Step 5	**`standby`** `[`*`group-number`*`]` **`mac-address`** *`mac-address`*
	or
	`standby use-bia` `[`**`scope interface`**`]`
	Example:
	`Router(config-if)# standby 1 mac-address 5000.1000.1060`
	or
	Example:
	`Router(config-if)# standby use-bia`
		Specifies a virtual MAC address for HSRP.
		• This command cannot be used on a Token Ring interface.
		or
		Configures HSRP to use the burned-in address of the interface as its virtual MAC address.
		• The **scope interface** keywords specify that the command is configured just for the subinterface on which it was entered, instead of the major interface.

Step 6	**`standby`** `[`*`group-number`*`]` **`ip`** `[`*`ip-address`* `[`**`secondary`**`]]`
	Example:
	`Router(config-if)# standby 1 ip 172.16.6.100`
		Activates HSRP.

Linking IP Redundancy Clients to HSRP Groups

Perform this task to link IP redundancy clients to HSRP groups.

HSRP provides stateless redundancy for IP routing. HSRP by itself is limited to maintaining its own state. Linking an IP redundancy client to an HSRP group provides a mechanism that allows HSRP to provide a service to client applications so they can implement stateful failover.

IP redundancy clients are other Cisco IOS processes or applications that use HSRP to provide or withhold a service or resource dependent upon the state of the group.

Prerequisites

Within the client application, you must first specify the same name as configured in the **standby name** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **name** [*redundancy-name*]
6. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`*
	Example:
	`Router(config-if)# ip address 10.0.0.1 255.255.255.0`
		Specifies an IP address for an interface.

	Command or Action	Purpose
Step 5	**standby** [*group-number*] **name** [*redundancy-name*] **Example:** `Router(config-if)# standby 1 name HSRP-1`	Configures the name of the standby group. • HSRP groups have a default name so it is not a requirement to specify a name.
Step 6	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 1 ip 10.0.0.11`	Activates HSRP.

Changing to HSRP Version 2

HSRP version 2 was introduced to prepare for further enhancements and to expand the capabilities beyond what is possible with HSRP version 1. HSRP version 2 has a different packet format than HSRP version 1.

HSRP Version 2 Design

HSRP version 2 is designed to address the following issues relative to HSRP version 1:

- Previously, millisecond timer values are not advertised or learned. HSRP version 2 advertises and learns millisecond timer values. This change ensures stability of the HSRP groups in all cases.
- Group numbers are restricted to the range from 0 to 255. HSRP version 2 expands the group number range from 0 to 4095.
- HSRP version 2 provides improved management and troubleshooting. With HSRP version 1, there is no method to identify from HSRP active hello messages which physical router sent the message because the source MAC address is the HSRP virtual MAC address. The HSRP version 2 packet format includes a 6-byte identifier field that is used to uniquely identify the sender of the message. Typically, this field is populated with the interface MAC address.
- The multicast address 224.0.0.2 is used to send HSRP hello messages. This address can conflict with Cisco Group Management Protocol (CGMP) leave processing.

Version 1 is the default version of HSRP.

HSRP version 2 uses the new IP multicast address 224.0.0.102 to send hello packets instead of the multicast address of 224.0.0.2, which is used by version 1. This new multicast address allows CGMP leave processing to be enabled at the same time as HSRP.

HSRP version 2 permits an expanded group number range, 0 to 4095, and consequently uses a new MAC address range 0000.0C9F.F000 to 0000.0C9F.FFFF. The increased group number range does not imply that an interface can, or should, support that many HSRP groups. The expanded group number range was changed to allow the group number to match the VLAN number on subinterfaces.

When the HSRP version is changed, each group will reinitialize because it now has a new virtual MAC address.

HSRP version 2 has a different packet format than HSRP version 1. The packet format uses a type-length-value (TLV) format. HSRP version 2 packets received by an HSRP version 1 router will have the type field mapped to the version field by HSRP version 1 and subsequently ignored.

The Gateway Load Balancing Protocol (GLBP) also addresses the same issues relative to HSRP version 1 that HSRP version 2 does. See the *Configuring GLBP* document for more information on GLBP.

Restrictions

- HSRP version 2 is not available for ATM interfaces running LAN emulation.
- HSRP version 2 will not interoperate with HSRP version 1. An interface cannot operate both version 1 and version 2 because both versions are mutually exclusive. However, the different versions can be run on different physical interfaces of the same router. You cannot change from version 2 to version 1 if you have configured groups above the group number range allowed for version 1 (0 to 255).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **standby version** {**1** | **2**}
6. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
7. **end**
8. **show standby**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface vlan 400`
		Configures an interface type and enters interface configuration mode.

Step 4	**`ip address`** *`ip-address mask`*
	Example:
	`Router(config-if)# ip address 10.10.28.1 255.255.255.0`
		Sets an IP address for an interface.

	Command or Action	Purpose

Step 5	**standby version** {**1** | **2**}
	Example:
	`Router(config-if)# standby version 2`
		Changes the HSRP version.

Step 6	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 400 ip 10.10.28.5`	Activates HSRP. • The group number range for HSRP version 2 is expanded to 0 through 4095. The group number range for HSRP version 1 is 0 through 255.

Step 7	**end**
	Example:
	`Router(config-if)# end`
		Ends the current configuration session and returns to privileged EXEC mode.

Step 8	**show standby** **Example:** `Router# show standby`	(Optional) Displays HSRP information. • HSRP version 2 information will be displayed if configured.

Configuring SSO HSRP

This section contains the following tasks:

- Enabling SSO Aware HSRP, page 40 (required)
- Verifying SSO Aware HSRP, page 41 (optional)

SSO HSRP alters the behavior of HSRP when a router with redundant Route Processors (RPs) is configured for Stateful Switchover (SSO) redundancy mode. When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.

With this functionality, HSRP SSO information is synchronized to the standby RP, allowing traffic that is sent using the HSRP virtual IP address to be continuously forwarded during a switchover without a loss of data or a path change. Additionally, if both RPs fail on the active HSRP router, then the standby HSRP router takes over as the active HSRP router.

The feature is enabled by default when the redundancy mode of operation is set to SSO.

SSO Dual-Route Processors and Cisco Nonstop Forwarding

SSO functions in networking devices (usually edge devices) that support dual RPs. SSO provides RP redundancy by establishing one of the RPs as the active processor and the other RP as the standby processor. SSO also synchronizes critical state information between the RPs so that network state information is dynamically maintained between RPs.

SSO is generally used with Cisco Nonstop Forwarding (NSF). Cisco NSF enables forwarding of data packets to continue along known routes while the routing protocol information is being restored following a switchover. With NSF, users are less likely to experience service outages.

HSRP and SSO Working Together

SSO HSRP enables the Cisco IOS HSRP subsystem software to detect that a standby RP is installed and the system is configured in SSO redundancy mode. Further, if the active RP fails, no change occurs to the HSRP group itself and traffic continues to be forwarded through the current active gateway router.

Prior to this feature, when the primary RP of the active router failed, it would stop participating in the HSRP group and trigger another router in the group to take over as the active HSRP router.

SSO HSRP is required to preserve the forwarding path for traffic destined to the HSRP virtual IP address through an RP switchover.

Configuring SSO on the edge router enables the traffic on the Ethernet links to continue during an RP failover without the Ethernet traffic switching over to an HSRP standby router (and then back, if preemption is enabled).

Enabling SSO Aware HSRP

The functionality is enabled by default when the redundancy mode is set to SSO. Perform this task to reenable HSRP to be SSO aware if it has been disabled.

Note You may want to disable SSO HSRP by using the **no standby sso** command if you have LAN segments that should switch HSRP traffic to a redundant device while SSO maintains traffic flow for other connections.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **mode sso**
5. **exit**
6. **no standby sso**
7. **standby sso**
8. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`configure terminal`
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command or Action	Purpose
Step 3	**redundancy** **Example:** `Router(config)# redundancy`	Enters redundancy configuration mode.
Step 4	**mode sso** **Example:** `Router(config-red)# mode sso`	Enables the redundancy mode of operation to SSO. • After performing this step, HSRP is SSO aware on interfaces that are configured for HSRP and the standby RP is automatically reset.
Step 5	**exit** **Example:** `Router(config-red)# exit`	Exits redundancy configuration mode.
Step 6	**no standby sso** **Example:** `Router(config)# no standby sso`	Disables HSRP SSO mode for all HSRP groups.
Step 7	**standby sso** **Example:** `Router(config)# standby sso`	Enables the SSO HSRP feature if you have disabled the functionality.

Step 8	**end**
	Example:
	`Router(config)# end`
		Ends the current configuration session and returns to privileged EXEC mode.

Verifying SSO Aware HSRP

To verify or debug HSRP SSO operation, perform the following steps from the active RP console.

SUMMARY STEPS

1. **show standby**
2. **debug standby events ha**

DETAILED STEPS

Step 1 **show standby**

Use the **show standby** command to display the state of the standby RP, for example:

```
Router# show standby

Ethernet0/0/1 - Group 1
 State is Active (standby RP)
 Virtual IP address is 10.1.0.7
 Active virtual MAC address is unknown
  Local virtual MAC address is 000a.f3fd.5001 (bia)
 Hello time 1 sec, hold time 3 sec
 Authentication text “authword”
 Preemption enabled
 Active router is unknown
 Standby router is unknown
 Priority 110 (configured 120)
  Track object 1 state Down decrement 10
 Group name is “name1” (cfgd)
```

Step 2 **debug standby events ha**

Use the **debug standby events ha** command to display the active and standby RPs, for example:

```
Router# debug standby events ha

!Active RP

*Apr 27 04:13:47.755: HSRP: Et0/0/1 Grp 101 RF Encode state Listen into sync buffer
*Apr 27 04:13:47.855: HSRP: CF Sync send ok
*Apr 27 04:13:57.755: HSRP: Et0/0/1 Grp 101 RF Encode state Speak into sync buffer
*Apr 27 04:13:57.855: HSRP: CF Sync send ok
*Apr 27 04:14:07.755: HSRP: Et0/0/1 Grp 101 RF Encode state Standby into sync buffer
*Apr 27 04:14:07.755: HSRP: Et0/0/1 Grp 101 RF Encode state Active into sync buffer
*Apr 27 04:14:07.863: HSRP: CF Sync send ok
*Apr 27 04:14:07.867: HSRP: CF Sync send ok

!Standby RP

*Apr 27 04:11:21.011: HSRP: RF CF client 32, entity 0 got msg len 24
*Apr 27 04:11:21.011: HSRP: Et0/0/1 Grp 101 RF sync state Init -> Listen
*Apr 27 04:11:31.011: HSRP: RF CF client 32, entity 0 got msg len 24
*Apr 27 04:11:31.011: HSRP: Et0/0/1 Grp 101 RF sync state Listen -> Speak
*Apr 27 04:11:41.071: HSRP: RF CF client 32, entity 0 got msg len 24
*Apr 27 04:11:41.071: HSRP: RF CF client 32, entity 0 got msg len 24
*Apr 27 04:11:41.071: HSRP: Et0/0/1 Grp 101 RF sync state Speak -> Standby
*Apr 27 04:11:41.071: HSRP: Et0/0/1 Grp 101 RF sync state Standby -> Active
```

Enabling HSRP MIB Traps

HSRP MIB supports Simple Network Management Protocol (SNMP) Get operations, to allow network devices to get reports about HSRP groups in a network from the network management station.

Enabling HSRP MIB trap support is performed through the CLI, and the MIB is used for getting the reports. A trap notifies the network management station when a router leaves or enters the active or standby state. When an entry is configured from the CLI, the RowStatus for that group in the MIB immediately goes to the active state.

The Cisco IOS software supports a read-only version of the MIB, and set operations are not supported.

This functionality supports four MIB tables, as follows:

- cHsrpGrpEntry table defined in CISCO-HSRP-MIB.my
- cHsrpExtIfTrackedEntry, cHsrpExtSecAddrEntry, and cHsrpExtIfEntry defined in CISCO-HSRP-EXT-MIB.my

The cHsrpGrpEntry table consists of all the group information defined in RFC 2281, *Cisco Hot Standby Router Protocol*; the other tables consist of the Cisco extensions to RFC 2281, which are defined in CISCO-HSRP-EXT-MIB.my.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server enable traps hsrp**
4. **snmp-server host** *host community-string* **hsrp**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`snmp-server enable traps hsrp`**
	Example:
	`Router(config)# snmp-server enable traps hsrp`
		Enables the router to send SNMP traps and informs, and HSRP notifications.

Step 4	**`snmp-server host`** *`host community-string`* **`hsrp`**
	Example:
	`Router# snmp-server host myhost.comp.com public hsrp`
		Specifies the recipient of an SNMP notification operation, and that HSRP notifications be sent to the host.

Configuring HSRP BFD Peering

In Cisco IOS Release 12.4(11)T and later releases, the HSRP BFD Peering feature introduces BFD in the HSRP group member health monitoring system. Previously, group member monitoring relied exclusively on HSRP multicast messages, which are relatively large and consume CPU memory to produce and check. In architectures where a single interface hosts a large number of groups, there is a need for a protocol with low CPU memory consumption and processing overhead. BFD addresses this issue and offers sub-second health monitoring (failure detection in milliseconds) at a relatively low CPU impact. HSRP BFD peering is enabled by default.

This section contains the following procedures:

- Configuring BFD Session Parameters on the Interface, page 44
- Configuring HSRP BFD Peering, page 44
- Verifying HSRP BFD Peering, page 46

Configuring BFD Session Parameters on the Interface

Perform this task to configure BFD on the interface by setting the baseline BFD session parameters on an interface. Repeat the steps in this procedure for each interface over which you want to run BFD sessions to BFD neighbors.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier*
5. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface FastEthernet 6/0`
		Enters interface configuration mode.

Step 4	**`bfd interval`** *`milliseconds`* **`min_rx`** *`milliseconds`* **`multiplier`** *`interval-multiplier`*
	Example:
	`Router(config-if)# bfd interval 50 min_rx 50 multiplier 5`
		Enables BFD on the interface.

Step 5	**`end`**
	Example:
	`Router(config-if)# end`
		Exits interface configuration mode.

Configuring HSRP BFD Peering

Perform this task to enable HSRP BFD peering. Repeat the steps in this procedure for each interface over which you want to run BFD sessions to HSRP peers.

HSRP supports BFD peering by default. If HSRP BFD peering has been manually disabled, you can reenable it at the router level to enable BFD support globally for all interfaces or you can reenable it on a per-interface basis at the interface level.

Prerequisites

- HSRP must be running on all participating routers.
- Cisco Express Forwarding (CEF) must be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip cef** [**distributed**]
4. **interface** *type number*
5. **ip address** *ip-address mask*
6. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
7. **standby bfd**
8. **exit**
9. **standby bfd all-interfaces**
10. **exit**
11. **show standby** [**neighbors**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip cef** [**distributed**] **Example:** `Router(config)# ip cef`	Enables CEF or distributed CEF.
Step 4	**interface** *type number* **Example:** `Router(config)# interface FastEthernet 6/0`	Enters interface configuration mode.
Step 5	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.0.0.11 255.255.255.0`	Configures an IP address for the interface.

	Command or Action	Purpose
Step 6	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Router(config-if)# standby 1 ip 10.0.0.11`	Activates HSRP.
Step 7	**standby bfd** **Example:** `Router(config-if)# standby bfd`	(Optional) Enables HSRP support for BFD on the interface.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 9	**standby bfd all-interfaces** **Example:** `Router(config)# standby bfd all-interfaces`	(Optional) Enables HSRP support for BFD on all interfaces.
Step 10	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.
Step 11	**show standby** [**neighbors**] **Example:** `Router# show standby neighbors`	(Optional) Displays information about HSRP support for BFD.

Verifying HSRP BFD Peering

To verify HSRP BFD Peering, use any of the following optional commands.

SUMMARY STEPS

1. **show standby**
2. **show standby neighbors** [*type number*]
3. **show bfd neighbor** [**details**]

DETAILED STEPS

Step 1 **show standby**

Use the **show standby** command to display HSRP information.

```
Router# show standby

FastEthernet2/0 - Group 1
  State is Active
    2 state changes, last state change 00:08:06
  Virtual IP address is 10.0.0.11
  Active virtual MAC address is 0000.0c07.ac01
    Local virtual MAC address is 0000.0c07.ac01 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 2.772 secs
  Preemption enabled
  Active router is local
  Standby router is 10.0.0.2, priority 90 (expires in 8.268 sec)
    BFD enabled !
  Priority 110 (configured 110)
    Group name is "hsrp-Fa2/0-1" (default)
```

Step 2 **show standby neighbors** [*type number*]

Use the **show standby neighbors** command to display information about HSRP peer routers on an interface.

```
Router1# show standby neighbors

HSRP neighbors on FastEthernet2/0
    10.1.0.22
    No active groups
    Standby groups: 1
    BFD enabled !

Router2# show standby neighbors

HSRP neighbors on FastEthernet2/0
    10.0.0.2
    Active groups: 1
    No standby groups
    BFD enabled !
```

Step 3 **show bfd neighbors** [**details**]

Use the **show bfd neighbors** command to display a line-by-line listing of existing Bidirectional Forwarding Detection (BFD) adjacencies. The **details** keyword displays BFD protocol parameters and timers for each neighbor.

```
Router# show bfd neighbors details

OurAddr       NeighAddr     LD/RD  RH/RS   Holdown(mult)  State     Int
10.0.0.2      10.0.0.1       5/0    Down      0    (0 )   Down      Fa2/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 0, Received Multiplier: 0
Holdown (hits): 0(0), Hello (hits): 1000(55)
Rx Count: 0, Rx Interval (ms) min/max/avg: 0/0/0 last: 3314120 ms ago
Tx Count: 55, Tx Interval (ms) min/max/avg: 760/1000/872 last: 412 ms ago
Registered protocols: HSRP !
Last packet: Version: 1            - Diagnostic: 0
             State bit: AdminDown  - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 0         - Length: 0
             My Discr.: 0          - Your Discr.: 0
             Min tx interval: 0    - Min rx interval: 0
             Min Echo interval: 0
```

What to Do Next

For more information about configuring BFD, see the “Bidirectional Forwarding Detection” document in the *Cisco IOS IP Routing Configuration Guide* at the following URL:

http://www.cisco.com/en/US/docs/ios/iproute/configuration/guide/irp_bfd.html

Configuration Examples for HSRP

This section provides the following configuration examples:

- HSRP Priority and Preemption: Example, page 48
- HSRP Object Tracking: Example, page 49
- HSRP Group Shutdown: Example, page 49
- HSRP MD5 Authentication Using Key Strings: Example, page 50
- HSRP MD5 Authentication Using Key Chains: Example, page 50
- HSRP MD5 Authentication Using Key Strings and Key Chains: Example, page 50
- HSRP Text Authentication: Example, page 51
- Multiple HSRP for Load Balancing: Example, page 51
- Improving CPU and Network Performance with HSRP Multiple Group Optimization: Example, page 53
- HSRP Support for ICMP Redirect Messages: Example, page 53
- HSRP Virtual MAC Addresses and BIA MAC Address: Example, page 54
- Linking IP Redundancy Clients to HSRP Groups: Example, page 54
- HSRP Version 2: Example, page 55
- SSO HSRP (Cisco IOS Release 12.2(25)S): Example, page 55
- HSRP MIB Traps: Example, page 56
- HSRP BFD Peering: Example, page 56

HSRP Priority and Preemption: Example

In the following example, Router A is configured to be the active router for group 1 because it has the higher priority and standby router for group 2. Router B is configured to be the active router for group 2 and standby router for group 1.

Router A Configuration

```
interface Ethernet0/0
 ip address 10.1.0.21 255.255.0.0
 standby 1 priority 110
 standby 1 preempt
 standby 1 ip 10.1.0.1
 standby 2 priority 95
 standby 2 preempt
 standby 2 ip 10.1.0.2
```

Router B Configuration

```
interface Ethernet0/0
 ip address 10.1.0.22 255.255.0.0
 standby 1 preempt
 standby 1 priority 105
 standby 1 ip 10.1.0.1
 standby 2 priority 110
 standby 2 preempt
 standby 2 ip 10.1.0.2
```

HSRP Object Tracking: Example

In the following example, the tracking process is configured to track the IP-routing capability of serial interface 1/0. HSRP on Ethernet interface 0/0 then registers with the tracking process to be informed of any changes to the IP-routing state of serial interface 1/0. If the IP state on serial interface 1/0 goes down, the priority of the HSRP group is reduced by 10.

If both serial interfaces are operational, Router A will be the HSRP active router because it has the higher priority. However, if IP routing on serial interface 1/0 in Router A fails, the HSRP group priority will be reduced and Router B will take over as the active router, thus maintaining a default virtual gateway service to hosts on the 10.1.0.0 subnet.

Router A Configuration

```
track 100 interface serial1/0 ip routing
!
interface Ethernet0/0
 ip address 10.1.0.21 255.255.0.0
 standby 1 preempt
 standby 1 priority 110
 standby 1 track 100 decrement 10
 standby 1 ip 10.1.0.1
```

Router B Configuration

```
track 100 interface serial1/0 ip routing
!
interface Ethernet0/0
 ip address 10.1.0.22 255.255.0.0
 standby 1 preempt
 standby 1 priority 105
 standby 1 track 100 decrement 10
 standby 1 ip 10.1.0.1
```

HSRP Group Shutdown: Example

In the following example, the tracking process is configured to track the IP-routing capability of serial interface 1/0. HSRP on Ethernet interface 0/0 then registers with the tracking process to be informed of any changes to the IP-routing state of serial interface 1/0. If the IP state on serial interface 1/0 goes down, the HSRP group is disabled.

If both serial interfaces are operational, Router A will be the HSRP active router because it has the higher priority. However, if IP routing on serial interface 1/0 in Router A fails, the HSRP group will be disabled and Router B will take over as the active router, thus maintaining a default virtual gateway service to hosts on the 10.1.0.0 subnet.

Router A Configuration

```
track 100 interface serial1/0 ip routing
!
interface Ethernet0/0
 ip address 10.1.0.21 255.255.0.0
 standby 1 ip 10.1.0.1
 standby 1 preempt
 standby 1 priority 110
 standby 1 track 100 shutdown
```

Router B Configuration

```
track 100 interface serial1/0 ip routing
!
interface Ethernet0/0
 ip address 10.1.0.22 255.255.0.0
 standby 1 ip 10.1.0.1
 standby 1 preempt
 standby 1 priority 105
 standby 1 track 100 shutdown
```

If an object is already being tracked by an HSRP group, you cannot change the configuration to use the HSRP Group Shutdown feature. You must first remove the tracking configuration using the **no standby track** command and then reconfigure it using the **standby track** command with the **shutdown** keyword.

The following example shows how to change the configuration of a tracked object to include the HSRP Group Shutdown feature:

```
no standby 1 track 101 decrement 10
standby 1 track 101 shutdown
```

HSRP MD5 Authentication Using Key Strings: Example

The following example shows how to configure HSRP MD5 authentication using a key string:

```
interface Ethernet0/1
 standby 1 priority 110
 standby 1 preempt
 standby 1 authentication md5 key-string 54321098452103ab timeout 30
 standby 1 ip 10.21.0.10
```

HSRP MD5 Authentication Using Key Chains: Example

In the following example, HSRP queries the key chain “hsrp1” to obtain the current live key and key ID for the specified key chain:

```
key chain hsrp1
 key 1
 key-string 54321098452103ab

interface Ethernet0/1
 standby 1 priority 110
 standby 1 preempt
 standby 1 authentication md5 key-chain hsrp1
 standby 1 ip 10.21.0.10
```

HSRP MD5 Authentication Using Key Strings and Key Chains: Example

The key ID for key-string authentication is always zero. If a key chain is configured with a key ID of zero, then the following configuration will work:

Router 1

```
key chain hsrp1
 key 0
 key-string 54321098452103ab

interface Ethernet0/1
```

```
 standby 1 authentication md5 key-chain hsrp1
 standby 1 ip 10.21.0.10
```

Router 2

```
interface Ethernet0/1
 standby 1 authentication md5 key-string 54321098452103ab
 standby 1 ip 10.21.0.10
```

HSRP Text Authentication: Example

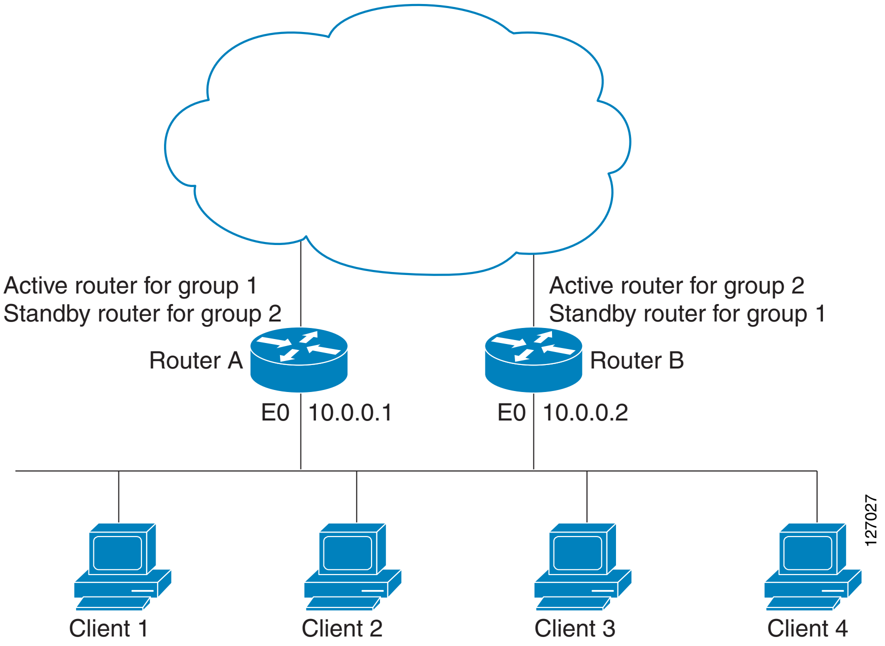
The following example shows how to configure HSRP text authentication using a text string:

```
interface Ethernet0/1
 standby 1 priority 110
 standby 1 preempt
 standby 1 authentication text company2
 standby 1 ip 10.21.0.10
```

Multiple HSRP for Load Balancing: Example

You can use HSRP or multiple HSRP groups when you configure load sharing. In Figure 4, half of the clients are configured for Router A, and half of the clients are configured for Router B. Together, the configuration for Routers A and B establish two Hot Standby groups. For group 1, Router A is the default active router because it has the assigned highest priority, and Router B is the standby router. For group 2, Router B is the default active router because it has the assigned highest priority, and Router A is the standby router. During normal operation, the two routers share the IP traffic load. When either router becomes unavailable, the other router becomes active and assumes the packet-transfer functions of the router that is unavailable. The **standby preempt** interface configuration command is necessary so that if a router goes down and then comes back up, preemption occurs and restores load sharing.

Figure 4 HSRP Load Sharing Example


Active router for group 1
Standby router for group 2
Router A
E0
10.0.0.1
Active router for group 2
Standby router for group 1
Router B
E0
10.0.0.2
Client 1
Client 2
Client 3
Client 4
127027

The following example shows Router A configured as the active router for group 1 with a priority of 110 and Router B configured as the active router for group 2 with a priority of 110. The default priority level is 100. Group 1 uses a virtual IP address of 10.0.0.3 and Group 2 uses a virtual IP address of 10.0.0.4.

Router A Configuration

```
hostname RouterA
!
interface ethernet 0
 ip address 10.0.0.1 255.255.255.0
 standby 1 priority 110
 standby 1 preempt
 standby 1 ip 10.0.0.3
 standby 2 preempt
 standby 2 ip 10.0.0.4
```

Router B Configuration

```
hostname RouterB
!
interface ethernet 0
 ip address 10.0.0.2 255.255.255.0
 standby 1 preempt
 standby 1 ip 10.0.0.3
 standby 2 priority 110
 standby 2 preempt
 standby 2 ip 10.0.0.4
```

Improving CPU and Network Performance with HSRP Multiple Group Optimization: Example

The following example shows how to configure an HSRP client and master group:

```
interface Ethernet1/0
 no shutdown
 standby mac-refresh 30
! Client Hello message interval
!
interface Ethernet1/0.2
 no shutdown
 encapsulation dot1Q 2
 ip vrf forwarding VRF2
 ip address 10.0.0.100 255.255.0.0
 standby 1 ip 10.0.0.254
 standby 1 priority 110
 standby 1 preempt
 standby 1 name HSRP1
!Server group
!
interface Ethernet1/0.3
 no shutdown
 encapsulation dot1Q 3
 ip vrf forwarding VRF3
 ip address 10.0.0.100 255.255.0.0
 standby 2 ip 10.0.0.254
 standby 2 follow HSRP1
! Client group
!
interface Ethernet1/0.4
 no shutdown
 encapsulation dot1Q 4
 ip vrf forwarding VRF4
 ip address 10.0.0.100 255.255.0.0
 standby 2 ip 10.0.0.254
 standby 2 follow HSRP1
! Client group
```

HSRP Support for ICMP Redirect Messages: Example

The following is a configuration example for two HSRP groups that allow the filtering of ICMP redirect messages:

Router A Configuration—Active for Group 1 and Standby for Group 2

```
interface Ethernet1
 ip address 10.0.0.10 255.0.0.0
 standby redirect
 standby 1 priority 120
 standby 1 preempt delay minimum 20
 standby 1 ip 10.0.0.1
 standby 2 priority 105
 standby 2 preempt delay minimum 20
 standby 2 ip 10.0.0.2
```

Router B Configuration—Standby for Group 1 and Active for Group 2

```
interface Ethernet1
 ip address 10.0.0.11 255.0.0.0
 standby redirect
 standby 1 priority 105
 standby 1 preempt delay minimum 20
 standby 1 ip 10.0.0.1
 standby 2 priority 120
 standby 2 preempt delay minimum 20
 standby 2 ip 10.0.0.2
```

HSRP Virtual MAC Addresses and BIA MAC Address: Example

In an APPN network, an end node is typically configured with the MAC address of the adjacent network node. In the following example, if the end nodes are configured to use 4000.1000.1060, HSRP group 1 is configured to use the same MAC address:

```
interface Ethernet0/2
 ip address 10.0.0.1
 standby 1 mac-address 4000.1000.1060
 standby 1 ip 10.0.0.11
```

In the following example, the burned-in address of Token Ring interface 3/0 will be the virtual MAC address mapped to the virtual IP address:

```
interface token3/0
 standby use-bia
```

Note You cannot use the **standby use-bia** command and the **standby mac-address** command in the same configuration.

Linking IP Redundancy Clients to HSRP Groups: Example

The following example shows HSRP support for a static NAT configuration. The NAT client application is linked to HSRP via the correlation between the name specified by the **standby name** command. Two routers are acting as HSRP active and standby, and the NAT inside interfaces are HSRP enabled and configured to belong to the group named “group1.”

Active Router Configuration

```
interface BVI10
 ip address 192.168.5.54 255.255.255.255.0
 no ip redirects
 ip nat inside
 standby 10 ip 192.168.5.30
 standby 10 priority 110
 standby 10 preempt
 standby 10 name group1
 standby 10 track Ethernet2/1
!
!
 ip default-gateway 10.0.18.126
 ip nat inside source static 192.168.5.33 10.10.10.5 redundancy group1
 ip classless
 ip route 10.10.10.0 255.255.255.0 Ethernet2/1
 ip route 172.22.33.0 255.255.255.0 Ethernet2/1
 no ip http server
```

Standby Router Configuration

```
interface BVI10
 ip address 192.168.5.56 255.255.255.255.0
 no ip redirects
 ip nat inside
 standby 10 priority 95
 standby 10 preempt
 standby 10 name group1
 standby 10 ip 192.168.5.30
 standby 10 track Ethernet3/1
!
 ip default-gateway 10.0.18.126
 ip nat inside source static 192.168.5.33 3.3.3.5 redundancy group1
 ip classless
 ip route 10.0.32.231 255.255.255 Ethernet3/1
 ip route 10.10.10.0 255.255.255.0 Ethernet3/1
 no ip http server
```

HSRP Version 2: Example

The following example shows how to configure HSRP version 2 on an interface with a group number of 350:

```
interface vlan350
 standby version 2
 standby 350 priority 110
 standby 350 preempt
 standby 350 timers 5 15
 standby 350 ip 172.20.100.10
```

SSO HSRP (Cisco IOS Release 12.2(25)S): Example

The following example shows how to set the redundancy mode to SSO. HSRP is automatically SSO-aware when this mode is enabled.

```
redundancy
 mode sso
```

If SSO HSRP is disabled using the **no standby sso** command, you can reenable it as shown in the following example:

```
interface Ethernet1
 ip address 10.1.1.1 255.255.0.0
 standby priority 200
 standby preempt
 standby sso
```

HSRP MIB Traps: Example

The following examples show how to configure HSRP on two routers and enable the HSRP MIB trap support functionality. As in many environments, one router is preferred as the active one. Configuring a router's preference as the active router is realized by configuring it at a higher priority level and enabling preemption. In the following example, the active router is referred to as the primary router. The second router is referred to as the backup router:

Router A

```
interface Ethernet1
 ip address 10.1.1.1 255.255.0.0
 standby priority 200
 standby preempt
 standby ip 10.1.1.3
snmp-server enable traps hsrp
snmp-server host yourhost.cisco.com public hsrp
```

Router B

```
interface Ethernet1
 ip address 10.1.1.2 255.255.0.0
 standby priority 101
 standby ip 10.1.1.3
snmp-server enable traps hsrp
snmp-server host myhost.cisco.com public hsrp
```

HSRP BFD Peering: Example

HSRP supports BFD as a part of the HSRP group member health monitoring system. Previously, group member monitoring relied exclusively on HSRP multicast messages, which are relatively large and consume CPU memory to produce and check. BFD provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes.

In the following example, the **standby bfd** and the **standby bfd all-interfaces** commands are not displayed. HSRP support for BFD is enabled by default when BFD is configured on the router or interface using the **bfd interval** command. The **standby bfd** and **standby bfd all-interfaces** commands are needed only if BFD has been manually disabled on a router or interface.

Router A

```
ip cef
interface FastEthernet2/0
 no shutdown
 ip address 10.0.0.2 255.0.0.0
 ip router-cache cef
 bfd interval 200 min_rx 200 multiplier 3
 standby 1 ip 10.0.0.11
 standby 1 preempt
 standby 1 priority 110

 standby 2 ip 10.0.0.12
 standby 2 preempt
 standby 2 priority 110
```

Router B

```
interface FastEthernet2/0
 ip address 10.1.0.22 255.255.0.0
 no shutdown
 bfd interval 200 min_rx 200 multiplier 3
 standby 1 ip 10.0.0.11
 standby 1 preempt
 standby 1 priority 90

 standby 2 ip 10.0.0.12
 standby 2 preempt
 standby 2 priority 80
```

Additional References

The following sections provide references related to HSRP.

Related Documents

Related Topic	Document Title
BFD	*Bidirectional Forwarding Detection*
GLBP	*Configuring GLBP*
HSRP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference.*
ISSU	• *Cisco IOS In Service Software Upgrade Process* • *Configuring the Cisco IOS In Service Software Upgrade Process* section of the *Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide*, Release 12.2(31)SGA.
Object tracking	*Configuring Enhanced Object Tracking*
Troubleshooting HSRP	• *Understanding and Troubleshooting HSRP Problems in Catalyst Switch Networks* • *Hot Standby Router Protocol: Frequently Asked Questions*
VRRP	*Configuring VRRP*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
CISCO-HSRP-MIB CISCO-HSRP-EXT-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 1828	*IP Authentication Using Keyed MD5*
RFC 2281	*Cisco Hot Standby Router Protocol*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for HSRP

Table 1 lists the features in this module and provides links to specific configuration information.

Not all commands may be available in your Cisco IOS software release. For details on when support for specific commands was introduced, see the command reference documents.

For information on a feature in this technology that is not documented here, see the "Cisco IOS IP Application Services Features Roadmap" or the "FHRP Features Roadmap."

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1 Feature Information for HSRP

Feature Name	Releases	Feature Information
FHRP—HSRP BFD Peering	12.4(11)T	The FHRP—HSRP BFD Peering feature introduces BFD in the HSRP group member health monitoring system. Previously, group member monitoring relied exclusively on HSRP multicast messages, which are relatively large and consume CPU memory to produce and check. In architectures where a single interface hosts a large number of groups, there is a need for a protocol with low CPU memory consumption and processing overhead. BFD addresses this issue and offers sub second health monitoring (failure detection in milliseconds) at a relatively low CPU impact. In Cisco IOS Release 12.4(11)T, this feature was introduced on Cisco 7200 series, Cisco 7600 series, and Cisco Gigabit Switch Routers (GSRs). The following sections provide information about this feature: • HSRP BFD Peering, page 8 • Configuring HSRP BFD Peering, page 43 • Verifying HSRP BFD Peering, page 46 • HSRP BFD Peering: Example, page 56 The following commands were introduced or modified by this feature: **debug standby events neighbor**, **show standby**, **show standby neighbors**, **standby bfd**, **standby bfd all-interfaces**.
FHRP—HSRP Group Shutdown	12.4(9)T 12.2(33)SRC	The FHRP—HSRP Group Shutdown feature enables you to configure an HSRP group to become disabled (its state changed to Init) instead of having its priority decremented when a tracked object goes down. The following sections provide information about this feature: • HSRP Group Shutdown, page 7 • Configuring HSRP Object Tracking, page 15 • HSRP Group Shutdown: Example, page 49 The following commands were modified by this feature: **standby track**, **show standby**.
FHRP—HSRP-MIB	12.0(3)T 12.0(12)S Cisco IOS XE Release 2.1	The FHRP—HSRP-MIB feature introduces support for the CISCO-HRSP-MIB.

Table 1 Feature Information for HSRP (continued)

Feature Name	Releases	Feature Information
FHRP—HSRP Multiple Group Optimization	12.4(6)T 12.2(33)SRB	FHRP—HSRP Multiple Group Optimization feature improves the negotiation and maintenance of multiple HSRP groups configured on a subinterface. Only one HSRP group is required on a physical interface for the purposes of electing active and standby routers. This group is known as the *master* group. Other HSRP groups may be created on each subinterface and linked to the master group via the group name. These linked HSRP groups are known as *client* or *follow* groups. The following sections provide information about this feature: • HSRP Multiple Group Optimization, page 7 • Improving CPU and Network Performance with HSRP Multiple Group Optimization, page 28 • Improving CPU and Network Performance with HSRP Multiple Group Optimization: Example, page 53 The following commands were introduced or modified by this feature: **standby follow**, **show standby**.
FHRP—HSRP Support for IPv6	12.4(4)T 12.2(33)SRB	Support for IPv6 was added. For more information see the "Configuring First Hop Redundancy Protocols in IPv6" module of the *Cisco IOS IPv6 Configuration Guide.*
HSRP—ISSU	12.2(31)SGA 12.2(33)SRB1 Cisco IOS XE Release 2.1	The HSRP—ISSU feature enables support for ISSU in HSRP. The In Service Software Upgrade (ISSU) process allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues. In most networks, planned software upgrades are a significant cause of downtime. ISSU allows Cisco IOS software to be modified while packet forwarding continues, which increases network availability and reduces downtime caused by planned software upgrades. This document provides information about ISSU concepts and describes the steps taken to perform ISSU in a system. The following section provides information about this feature: • HSRP—ISSU, page 8 For more information about this feature, see the *Cisco IOS In Service Software Upgrade Process* document. There are no new or modified command for this feature.

Table 1 Feature Information for HSRP (continued)

Feature Name	Releases	Feature Information
HSRP MD5 Authentication	12.3(2)T 12.2(25)S 12.2(33)SRA 12.2(33)SXH	Prior to the introduction of the HSRP MD5 Authentication feature, HSRP authenticated protocol packets with a simple plain text string. The HSRP MD5 Authentication feature is an enhancement to generate an MD5 digest for the HSRP portion of the multicast HSRP protocol packet. This feature provides added security and protects against the threat from HSRP-spoofing software. The following sections provide information about this feature: • Configuring HSRP Authentication, page 17 The following commands were introduced or modified by this feature: **show standby**, **standby authentication**.
HSRP Version 2	12.3(4)T 12.2(25)S	HSRP Version 2 feature was introduced to prepare for further enhancements and to expand the capabilities beyond what is possible with HSRP version 1. HSRP version 2 has a different packet format than HSRP version 1. The following sections provide information about this feature: • Changing to HSRP Version 2, page 37 The following commands were introduced or modified by this feature: **show standby**, **standby ip**, **standby version**.
HSRP Support for MPLS VPNs	12.0(23)S, 12.0(17)ST, 12.2(28)SB, 12.2(17b)SXA, 12.2(8)T Cisco IOS XE Release 2.1	HSRP support for a Multiprotocol Label Switching (MPLS) Virtual Private Network (VPN) interface is useful when an Ethernet LAN is connected between two provider edge (PE) routers with either of the following conditions: The following section provides information about this feature: • HSRP Support for MPLS VPNs, page 7 There are no new or modified command for this feature.
SSO—HSRP	12.2(25)S 12.2(33)SRA 12.2(33)SXH Cisco IOS XE Release 2.1	The SSO—HSRP feature alters the behavior of HSRP when a router with redundant RPs is configured for SSO. When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails. The following sections provide information about this feature: • Configuring SSO HSRP, page 39 • HSRP and SSO Working Together, page 40 • Enabling SSO Aware HSRP, page 40 • Verifying SSO Aware HSRP, page 41 The following commands were introduced or modified by this feature: **debug standby events**, **standby sso**.

Glossary

active router—The primary router in an HSRP group that is currently forwarding packets for the virtual router.

active RP—The active RP that controls the system, provides network services, runs the routing protocols, and presents the system management interface.

BFD—Bidirectional Forwarding Detection. A detection protocol designed to provide fast forwarding path failure detection encapsulations, topologies, and routing protocols. In addition to fast forwarding, BFD provides a consistent failure detection method for network administrators.

client group—An HSRP group that is created on a subinterface and linked to the master group via the group name.

HSRP—Hot Standby Router Protocol. Protocol that provides high network availability and transparent network-topology changes. HSRP creates a router group with a lead router that services all packets sent to the HSRP address. The lead router is monitored by other routers in the group, and if it fails, one of these standby HSRP routers inherits the lead position and the HSRP group address.

ISSU—In Service Software Upgrade. A process that allows Cisco IOS software to be updated or otherwise modified while packet forwarding continues. In most networks, planned software upgrades are a significant cause of downtime. ISSU allows Cisco IOS software to be modified while packet forwarding continues, which increases network availability and reduces downtime caused by planned software upgrades.

master group—An HSRP group that is required on a physical interface for the purposes of electing active and standby routers.

NSF—Nonstop Forwarding. The ability of a router to continue to forward traffic to a router that may be recovering from a failure. Also, the ability of a router recovering from a failure to continue to correctly forward traffic sent to it by a peer.

RF—Redundancy Facility. A structured, functional interface used to notify its clients of active and standby state progressions and events.

RP—Route Processor. A generic term for the centralized control unit in a chassis. Platforms usually use a platform-specific term, such as RSP on the Cisco 7500, the PRE on the Cisco 10000, or the SUP+MSFC on the Cisco 7600.

RPR—Route Processor Redundancy. RPR provides an alternative to the High System Availability (HSA) feature. HSA enables a system to reset and use a standby Route Processor (RP) if the active RP fails. Using RPR, you can reduce unplanned downtime because RPR enables a quicker switchover between an active and standby RP if the active RP experiences a fatal error.

RPR+—An enhancement to RPR in which the standby RP is fully initialized.

SSO—Stateful Switchover. SSO refers to the implementation of Cisco IOS software that allows applications and features to maintain a defined state between an active and standby RP. When a switchover occurs, forwarding and sessions are maintained. Along with NSF, SSO makes an RP failure undetectable to the network.

standby group—The set of routers participating in HSRP that jointly emulate a virtual router.

standby router—The backup router in an HSRP group.

standby RP—The backup RP.

switchover—An event in which system control and routing protocol execution are transferred from the active RP to the standby RP. Switchover may be a manual operation or may be induced by a hardware or software fault. Switchover may include transfer of the packet forwarding function in systems that combine system control and packet forwarding in an indivisible unit.

virtual IP address—The default gateway IP address configured for an HSRP group.

virtual MAC address—For Ethernet and FDDI, the automatically generated MAC address when HSRP is configured. The standard virtual MAC address used is: 0000.0C07.ACxy, where *xy* is the group number in hexadecimal. The functional address is used for Token Ring. The virtual MAC address is different for HSRP version 2.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring IRDP

First Published: February 10, 2008
Last Updated: February 10, 2008

The ICMP Router Discovery Protocol (IRDP) allows IPv4 hosts to locate routers that provide IPv4 connectivity to other (non-local) IP networks. For a complete description of the IPv4 addressing commands in this module, refer to the *Cisco IOS IP Application Services Command Reference.* To locate documentation of other commands that appear in this module, use the command reference master index, or search online.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for IRDP" section on page 7.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Information About IRDP

To configure IRDP, you should understand the following concept:

- IRDP Overview, page 2

IRDP Overview

ICMP Router Discovery Protocol (IRDP) allows hosts to locate routers that can be used as a gateway to reach IP-based devices on other networks. When the device running IRDP operates as a router, router discovery packets are generated. When the device running IRDP operates as a host, router discovery packets are received. The Cisco IRDP implementation fully conforms to the router discovery protocol outlined in RFC 1256 (http://www.ietf.org/rfc/rfc1256.txt).

How to Configure IRDP

This section contains the following procedure:

- Configuring IRDP, page 2

Configuring IRDP

Perform this task to configure IRDP:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no shutdown**
5. **ip address** *ip-address mask*
6. **ip irdp**
7. **ip irdp multicast**
8. **ip irdp holdtime** *seconds*
9. **ip irdp maxadvertinterval** *seconds*
10. **ip irdp minadvertinterval** *seconds*
11. **ip irdp preference** *number*
12. **ip irdp address** *address number*
13. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`interface`** *`type number`*
	Example:
	`Router(config)# interface fastethernet 0/0`
		Specifies an interface and enters interface configuration mode.

Step 4	**`no shutdown`**
	Example:
	`Router(config-if)# no shutdown`
		Activates (enables) the interface.

Step 5	**`ip address`** *`ip-address mask`*
	Example:
	`Router(config-if)# ip address 172.16.16.1 255.255.240.0`
		Configures an IP address on the interface.

Step 6	**`ip irdp`**
	Example:
	`Router(config-if)# ip irdp`
		Enables IRDP on the interface

Step 7	**`ip irdp multicast`**
	Example:
	`Router(config-if)# ip irdp multicast`
		(Optional) Sends IRDP advertisements to the all-systems multicast address (224.0.0.1) on a specified interface.

Step 8	**`ip irdp holdtime`** *`seconds`*
	Example:
	`Router(config-if)# ip irdp holdtime 120`
		(Optional) Sets the IRDP period for which advertisements are valid.

Step 9	**`ip irdp maxadvertinterval`** *`seconds`*
	Example:
	`Router(config-if)# ip irdp maxadvertinterval 60`
		(Optional) Sets the IRDP maximum interval between advertisements.

Step 10	**`ip irdp minadvertinterval`** *`seconds`*
	Example:
	`Router(config-if)# ip irdp minadvertinterval 10`
		(Optional) Sets the IRDP minimum interval between advertisements.

	Command or Action	Purpose
Step 11	**ip irdp preference** *number* **Example:** `Router(config-if)# ip irdp preference 900`	(Optional) Sets the IRDP preference level of the device.
Step 12	**ip irdp address** *address number* **Example:** `Router(config-if)# ip irdp address 192.168.10.2 90`	(Optional) Specifies an IRDP address and preference to proxy-advertise.

Step 13	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Configuration Examples for IRDP

This section provides the following configuration example:

- Configuring IRDP: Example, page 4

Configuring IRDP: Example

The following example shows how to configure IRDP on a router:

```
interface fastethernet 0/1
 no shutdown
 ip address 172.16.10.1 255.255.255.0
 ip irdp
 ip irdp multicast
 ip irdp holdtime 120
 ip irdp maxadvertinterval 60
 ip irdp minadvertinterval 10
 ip irdp preference 900
 ip irdp address 192.168.10.2 90
```

Additional References

The following sections provide references related to the IRDP feature.

Related Documents

Related Topic	Document Title
IP application services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference.*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified	—

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified	—

RFCs

RFC	Title
RFC 1256	ICMP Router Discovery Messages http://www.ietf.org/rfc/rfc1256.txt

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for IRDP

Table 1 lists the features in this module. For information on a feature in this technology that is not documented here, see the other available documentation for your Cisco IOS release.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for IRDP

Feature Name	Releases	Feature Information
ICMP Router Discovery Protocol	10.0 12.2(33)SRA	The ICMP Router Discovery Protocol (IRDP) allows IPv4 hosts to locate routers that provide IPv4 connectivity to other (non-local) IP networks. The following command was introduced or modified: **ip irdp**.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring VRRP

First Published: May 2, 2005
Last Updated: May 5, 2008

The Virtual Router Redundancy Protocol (VRRP) is an election protocol that dynamically assigns responsibility for one or more virtual routers to the VRRP routers on a LAN, allowing several routers on a multiaccess link to utilize the same virtual IP address. A VRRP router is configured to run the VRRP protocol in conjunction with one or more other routers attached to a LAN. In a VRRP configuration, one router is elected as the virtual router master, with the other routers acting as backups in case the virtual router master fails.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for VRRP" section on page 27.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Restrictions for VRRP

VRRP is designed for use over multi-access, multicast, or broadcast capable Ethernet LANs. VRRP is not intended as a replacement for existing dynamic protocols.

VRRP is supported on Ethernet, Fast Ethernet, Bridge Group Virtual Interface (BVI), and Gigabit Ethernet interfaces, Multiprotocol Label Switching (MPLS) Virtual Private Networks (VPNs), VRF-aware MPLS VPNs and VLANs.

Because of the forwarding delay that is associated with the initialization of a BVI interface, it is necessary to set the VRRP advertise timer to a value equal to or greater than the forwarding delay on the BVI interface. This setting prevents a VRRP router on a recently initialized BVI interface from unconditionally taking over the master role. Use the **bridge forward-time** command to set the forwarding delay on the BVI interface. Use the **vrrp timers advertise** command to set the VRRP advertisement timer.

Information About VRRP

Before you configure VRRP, you should understand the following concepts:

- VRRP Operation, page 2
- VRRP Benefits, page 4
- Multiple Virtual Router Support, page 5
- VRRP Router Priority and Preemption, page 5
- VRRP Advertisements, page 6
- VRRP Object Tracking, page 6
- ISSU—VRRP, page 6
- SSO—VRRP, page 7

VRRP Operation

There are several ways a LAN client can determine which router should be the first hop to a particular remote destination. The client can use a dynamic process or static configuration. Examples of dynamic router discovery are as follows:

- Proxy ARP—The client uses Address Resolution Protocol (ARP) to get the destination it wants to reach, and a router will respond to the ARP request with its own MAC address.
- Routing protocol—The client listens to dynamic routing protocol updates (for example, from Routing Information Protocol [RIP]) and forms its own routing table.
- IRDP (ICMP Router Discovery Protocol) client—The client runs an Internet Control Message Protocol (ICMP) router discovery client.

The drawback to dynamic discovery protocols is that they incur some configuration and processing overhead on the LAN client. Also, in the event of a router failure, the process of switching to another router can be slow.

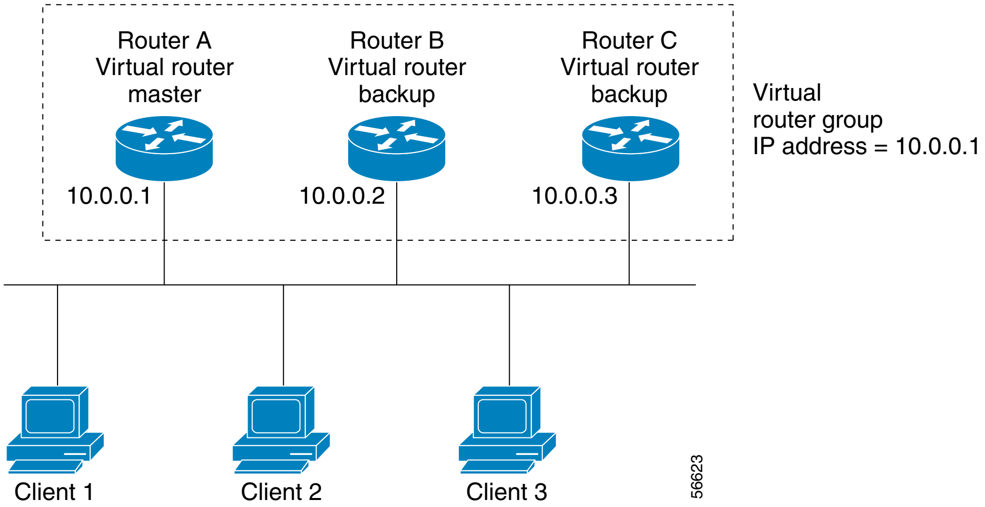
An alternative to dynamic discovery protocols is to statically configure a default router on the client. This approach simplifies client configuration and processing, but creates a single point of failure. If the default gateway fails, the LAN client is limited to communicating only on the local IP network segment and is cut off from the rest of the network.

VRRP can solve the static configuration problem. VRRP enables a group of routers to form a single *virtual router*. The LAN clients can then be configured with the virtual router as their default gateway. The virtual router, representing a group of routers, is also known as a VRRP group.

VRRP is supported on Ethernet, Fast Ethernet, BVI, and Gigabit Ethernet interfaces, on MPLS VPNs, VRF-aware MPLS VPNs and VLANs.

Figure 1 shows a LAN topology in which VRRP is configured. In this example, Routers A, B, and C are *VRRP routers* (routers running VRRP) that comprise a virtual router. The IP address of the virtual router is the same as that configured for the Ethernet interface of Router A (10.0.0.1).

Figure 1 Basic VRRP Topology

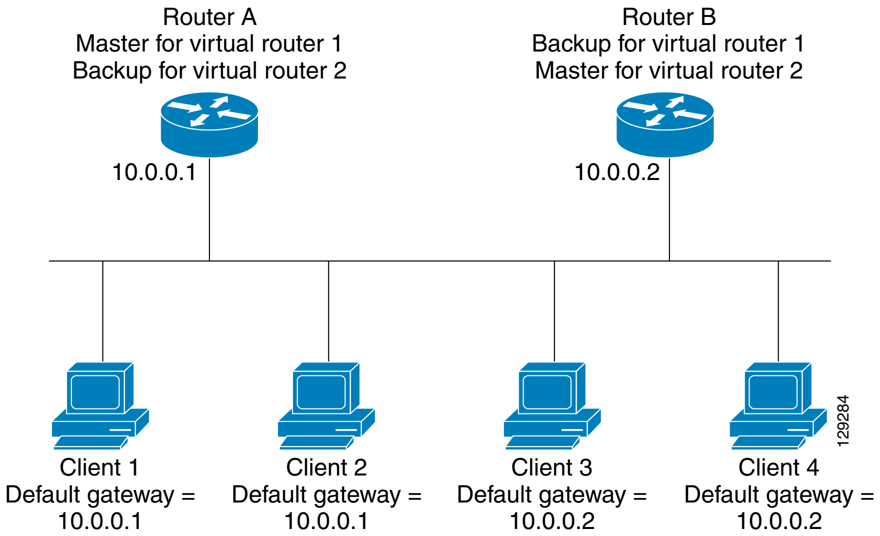


Because the virtual router uses the IP address of the physical Ethernet interface of Router A, Router A assumes the role of the *virtual router master* and is also known as the *IP address owner.* As the virtual router master, Router A controls the IP address of the virtual router and is responsible for forwarding packets sent to this IP address. Clients 1 through 3 are configured with the default gateway IP address of 10.0.0.1.

Routers B and C function as *virtual router backups*. If the virtual router master fails, the router configured with the higher priority will become the virtual router master and provide uninterrupted service for the LAN hosts. When Router A recovers, it becomes the virtual router master again. For more detail on the roles that VRRP routers play and what happens if the virtual router master fails, see the "VRRP Router Priority and Preemption" section later in this document.

Figure 2 shows a LAN topology in which VRRP is configured so that Routers A and B share the traffic to and from clients 1 through 4 and that Routers A and B act as virtual router backups to each other if either router fails.

Figure 2 Load Sharing and Redundancy VRRP Topology



In this topology, two virtual routers are configured. (For more information, see the "Multiple Virtual Router Support" section later in this document.) For virtual router 1, Router A is the owner of IP address 10.0.0.1 and virtual router master, and Router B is the virtual router backup to Router A. Clients 1 and 2 are configured with the default gateway IP address of 10.0.0.1.

For virtual router 2, Router B is the owner of IP address 10.0.0.2 and virtual router master, and Router A is the virtual router backup to Router B. Clients 3 and 4 are configured with the default gateway IP address of 10.0.0.2.

VRRP Benefits

Redundancy

VRRP enables you to configure multiple routers as the default gateway router, which reduces the possibility of a single point of failure in a network.

Load Sharing

You can configure VRRP in such a way that traffic to and from LAN clients can be shared by multiple routers, thereby sharing the traffic load more equitably among available routers.

Multiple Virtual Routers

VRRP supports up to 255 virtual routers (VRRP groups) on a router physical interface, subject to the platform supporting multiple MAC addresses. Multiple virtual router support enables you to implement redundancy and load sharing in your LAN topology.

Multiple IP Addresses

The virtual router can manage multiple IP addresses, including secondary IP addresses. Therefore, if you have multiple subnets configured on an Ethernet interface, you can configure VRRP on each subnet.

Preemption

The redundancy scheme of VRRP enables you to preempt a virtual router backup that has taken over for a failing virtual router master with a higher priority virtual router backup that has become available.

Authentication

VRRP message digest 5 (MD5) algorithm authentication protects against VRRP-spoofing software and uses the industry-standard MD5 algorithm for improved reliability and security.

Advertisement Protocol

VRRP uses a dedicated Internet Assigned Numbers Authority (IANA) standard multicast address (224.0.0.18) for VRRP advertisements. This addressing scheme minimizes the number of routers that must service the multicasts and allows test equipment to accurately identify VRRP packets on a segment. The IANA assigned VRRP the IP protocol number 112.

VRRP Object Tracking

VRRP object tracking provides a way to ensure the best VRRP router is virtual router master for the group by altering VRRP priorities to the status of tracked objects such as interface or IP route states.

Multiple Virtual Router Support

You can configure up to 255 virtual routers on a router physical interface. The actual number of virtual routers that a router interface can support depends on the following factors:

- Router processing capability
- Router memory capability
- Router interface support of multiple MAC addresses

In a topology where multiple virtual routers are configured on a router interface, the interface can act as a master for one virtual router and as a backup for one or more virtual routers.

VRRP Router Priority and Preemption

An important aspect of the VRRP redundancy scheme is VRRP router priority. Priority determines the role that each VRRP router plays and what happens if the virtual router master fails.

If a VRRP router owns the IP address of the virtual router and the IP address of the physical interface, this router will function as a virtual router master.

Priority also determines if a VRRP router functions as a virtual router backup and the order of ascendancy to becoming a virtual router master if the virtual router master fails. You can configure the priority of each virtual router backup with a value of 1 through 254 using the **vrrp priority** command.

For example, if Router A, the virtual router master in a LAN topology, fails, an election process takes place to determine if virtual router backups B or C should take over. If Routers B and C are configured with the priorities of 101 and 100, respectively, Router B is elected to become virtual router master because it has the higher priority. If Routers B and C are both configured with the priority of 100, the virtual router backup with the higher IP address is elected to become the virtual router master.

By default, a preemptive scheme is enabled whereby a higher priority virtual router backup that becomes available takes over for the virtual router backup that was elected to become virtual router master. You can disable this preemptive scheme using the **no vrrp preempt** command. If preemption is disabled, the virtual router backup that is elected to become virtual router master remains the master until the original virtual router master recovers and becomes master again.

VRRP Advertisements

The virtual router master sends VRRP advertisements to other VRRP routers in the same group. The advertisements communicate the priority and state of the virtual router master. The VRRP advertisements are encapsulated in IP packets and sent to the IP Version 4 multicast address assigned to the VRRP group. The advertisements are sent every second by default; the interval is configurable.

VRRP Object Tracking

Object tracking is an independent process that manages creating, monitoring, and removing tracked objects such as the state of the line-protocol of an interface. Clients such as the Hot Standby Router Protocol (HSRP), Gateway Load Balancing Protocol (GLBP), and now VRRP register their interest with specific tracked objects and act when the state of an object changes.

Each tracked object is identified by a unique number that is specified on the tracking command-line interface (CLI). Client processes such as VRRP use this number to track a specific object.

The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to interested client processes, either immediately or after a specified delay. The object values are reported as either up or down.

VRRP object tracking gives VRRP access to all the objects available through the tracking process. The tracking process provides the ability to track individual objects such as a the state of an interface line protocol, state of an IP route, or the reachability of a route.

VRRP provides an interface to the tracking process. Each VRRP group can track multiple objects that may affect the priority of the VRRP router. You specify the object number to be tracked and VRRP will be notified of any change to the object. VRRP increments (or decrements) the priority of the virtual router based on the state of the object being tracked.

ISSU—VRRP

VRRP supports In Service Software Upgrade (ISSU). In Service Software Upgrade (ISSU) allows a high-availability (HA) system to run in Stateful Switchover (SSO) mode even when different versions of Cisco IOS software are running on the active and standby Route Processors (RPs) or line cards.

ISSU provides the ability to upgrade or downgrade from one supported Cisco IOS release to another while continuing to forward packets and maintain sessions, thereby reducing planned outage time. The ability to upgrade or downgrade is achieved by running different software versions on the active RP and standby RP for a short period of time to maintain state information between RPs. This feature allows the system to switch over to a secondary RP running upgraded (or downgraded) software and continue forwarding packets without session loss and with minimal or no packet loss. This feature is enabled by default.

For detailed information about ISSU, see the *Cisco IOS In Service Software Upgrade Process* document at the following URL:

http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-inserv_updg.html

For detailed information about ISSU on the 7600 series routers, see the *ISSU and eFSU on Cisco 7600 Series Routers* document at the following URL:

http://www.cisco.com/en/US/partner/products/hw/routers/ps368/products_configuration_guide_chapter09186a00807f1c85.html

SSO—VRRP

With the introduction of the SSO—VRRP feature, VRRP is Stateful Switchover (SSO) aware. VRRP can detect when a router is failing over to the secondary RP and continue in its current group state.

SSO functions in networking devices (usually edge devices) that support dual Route Processors (RPs). SSO provides RP redundancy by establishing one of the RPs as the active processor and the other RP as the standby processor. SSO also synchronizes critical state information between the RPs so that network state information is dynamically maintained between RPs.

Prior to being SSO aware, if VRRP was deployed on a router with redundant RPs, a switchover of roles between the active RP and the standby RP would result in the router relinquishing its activity as a VRRP group member and then rejoining the group as if it had been reloaded. The SSO—VRRP feature enables VRRP to continue its activities as a group member during a switchover. VRRP state information between redundant RPs is maintained so that the standby RP can continue the router's activities within the VRRP during and after a switchover.

This feature is enabled by default. To disable this feature, use the **no vrrp sso** command in global configuration mode.

For more information, see the *Stateful Switchover* document at the following URL:

http://www.cisco.com/en/US/docs/ios/ha/configuration/guide/ha-stfl_swovr.html

How to Configure VRRP

This section contains the following procedures:

- Customizing VRRP, page 7 (optional)
- Enabling VRRP, page 9 (required)
- Disabling VRRP on an Interface, page 11 (optional)
- Configuring VRRP Object Tracking, page 12 (optional)
- Configuring VRRP Authentication, page 14 (optional)
- Enabling the Router to Send SNMP VRRP Notifications, page 21 (optional)

Customizing VRRP

Perform this task to customize VRRP.

Customizing the behavior of VRRP is optional. Be aware that as soon as you enable a VRRP group, that group is operating. It is possible that if you first enable a VRRP group before customizing VRRP, the router could take over control of the group and become the virtual router master before you have finished customizing the feature. Therefore, if you plan to customize VRRP, it is a good idea to do so before enabling VRRP.

How Object Tracking Affects the Priority of a VRRP Router

The priority of a device can change dynamically if it has been configured for object tracking and the object that is being tracked goes down. The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to VRRP, either immediately or after a specified delay. The object values are reported as either up or down. Examples of objects that can be tracked are the line protocol state of an interface or the reachability of an IP route. If the specified object goes down, the VRRP priority is reduced. The VRRP router with the higher priority can now become the virtual router master if it has the **vrrp preempt** command configured. See the "VRRP Object Tracking" section for more information on object tracking.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **vrrp** *group* **description** *text*
6. **vrrp** *group* **priority** *level*
7. **vrrp** *group* **preempt** [**delay minimum** *seconds*]
8. **vrrp** *group* **timers advertise** [**msec**] *interval*
9. **vrrp** *group* **timers learn**
10. **no vrrp sso**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface ethernet 0`	Enters interface configuration mode.
Step 4	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 172.16.6.5 255.255.255.0`	Configures an IP address for an interface.
Step 5	**vrrp** *group* **description** *text* **Example:** `Router(config-if)# vrrp 10 description working-group`	Assigns a text description to the VRRP group.

	Command or Action	Purpose
Step 6	`vrrp` *group* **priority** *level* **Example:** `Router(config-if)# vrrp 10 priority 110`	Sets the priority level of the router within a VRRP group. • The default priority is 100.
Step 7	`vrrp` *group* **preempt** [**delay minimum** *seconds*] **Example:** `Router(config-if)# vrrp 10 preempt delay minimum 380`	Configures the router to take over as virtual router master for a VRRP group if it has a higher priority than the current virtual router master. • The default delay period is 0 seconds. • The router that is IP address owner will preempt, regardless of the setting of this command.
Step 8	`vrrp` *group* **timers advertise** [**msec**] *interval* **Example:** `Router(config-if)# vrrp 10 timers advertise 110`	Configures the interval between successive advertisements by the virtual router master in a VRRP group. • The unit of the interval is in seconds unless the **msec** keyword is specified. The default *interval* value is 1 second. **Note** All routers in a VRRP group must use the same timer values. If the same timer values are not set, the routers in the VRRP group will not communicate with each other and any misconfigured router will change its state to master.
Step 9	`vrrp` *group* **timers learn** **Example:** `Router(config-if)# vrrp 10 timers learn`	Configures the router, when it is acting as virtual router backup for a VRRP group, to learn the advertisement interval used by the virtual router master.
Step 10	**no vrrp sso** **Example:** `Router(config)# no vrrp sso`	(Optional) Disables VRRP support of SSO. VRRP support of SSO is enabled by default.

Enabling VRRP

Perform this task to enable VRRP.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **vrrp** *group* **ip** *ip-address* [**secondary**]
6. **end**
7. **show vrrp** [**brief** | *group*]
8. **show vrrp interface** *type number* [**brief**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** Router(config)# interface ethernet 0	Enters interface configuration mode.
Step 4	**ip address** *ip-address mask* **Example:** Router(config-if)# ip address 172.16.6.5 255.255.255.0	Configures an IP address for an interface.
Step 5	**vrrp** *group* **ip** *ip-address* [**secondary**] **Example:** Router(config-if)# vrrp 10 ip 172.16.6.1	Enables VRRP on an interface. • After you identify a primary IP address, you can use the **vrrp ip** command again with the **secondary** keyword to indicate additional IP addresses supported by this group. **Note** All routers in the VRRP group must be configured with the same primary address for the virtual router. If different primary addresses are configured, the routers in the VRRP group will not communicate with each other and any misconfigured router will change its state to master.
Step 6	**end** **Example:** Router(config-if)# end	Returns to privileged EXEC mode.
Step 7	Router# **show vrrp** [**brief** | *group*] **Example:** Router# show vrrp 10	(Optional) Displays a brief or detailed status of one or all VRRP groups on the router.
Step 8	Router# **show vrrp interface** *type number* [**brief**] **Example:** Router# show vrrp interface ethernet 0	(Optional) Displays the VRRP groups and their status on a specified interface.

Disabling VRRP on an Interface

Disabling VRRP on an interface allows the protocol to be disabled, but the configuration retained. This ability was added with the introduction of the VRRP MIB, RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*.

You can use a Simple Network Management Protocol (SNMP) management tool to enable or disable VRRP on an interface. Because of the SNMP management capability, the **vrrp shutdown** command was introduced to represent a method via the CLI for VRRP to show the state that had been configured using SNMP.

When the **show running-config** command is entered, you can see immediately if the VRRP group has been configured and set to enabled or disabled. This is the same functionality that is enabled within the MIB.

The **no** form of the command enables the same operation that is performed within the MIB. If the **vrrp shutdown** command is specified using the SNMP interface, then entering the **no vrrp shutdown** command using the Cisco IOS CLI will reenable the VRRP group.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **vrrp** *group* **shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface ethernet 0`	Enters interface configuration mode.

	Command or Action	Purpose

Step 4	**`ip address`** *`ip-address mask`*
	Example:
	`Router(config-if)# ip address 172.16.6.5 255.255.255.0`
		Configures an IP address for an interface.

Step 5	**`vrrp`** *`group`* **`shutdown`**
	Example:
	`Router(config-if)# vrrp 10 shutdown`
		Disables VRRP on an interface.
		• The command is now visible on the router.
		Note You can have one VRRP group disabled, while retaining its configuration, and a different VRRP group enabled.

Configuring VRRP Object Tracking

Perform the following task to configure VRRP object tracking.

Restrictions

If a VRRP group is the IP address owner, its priority is fixed at 255 and cannot be reduced through object tracking.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **track** *object-number* **interface** *type number* {**line-protocol | ip routing**}
4. **interface** *type number*
5. **vrrp** *group* **ip** *ip-address*
6. **vrrp** *group* **priority** *level*
7. **vrrp** *group* **track** *object-number* [**decrement** *priority*]
8. **end**
9. **show track** [*object-number*]

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

	Command or Action	Purpose

Step 3	**track** *object-number* **interface** *type number* {**line-protocol** | **ip routing**}
	Example:
	`Router(config)# track 2 interface serial 6 line-protocol`
		Configures an interface to be tracked where changes in the state of the interface affect the priority of a VRRP group.
		• This command configures the interface and corresponding object number to be used with the **vrrp track** command.
		• The **line-protocol** keyword tracks whether the interface is up. The **ip routing** keyword also checks that IP routing is enabled and active on the interface.
		• You can also use the **track ip route** command to track the reachability of an IP route or a metric type object.

Step 4	**interface** *type number* **Example:** `Router(config)# interface Ethernet 2`	Enters interface configuration mode.
Step 5	**vrrp** *group* **ip** *ip-address* **Example:** `Router(config-if)# vrrp 1 ip 10.0.1.20`	Enables VRRP on an interface and identifies the IP address of the virtual router.
Step 6	**vrrp** *group* **priority** *level* **Example:** `Router(config-if)# vrrp 1 priority 120`	Sets the priority level of the router within a VRRP group.
Step 7	**vrrp** *group* **track** *object-number* [**decrement** *priority*] **Example:** `Router(config-if)# vrrp 1 track 2 decrement 15`	Configures VRRP to track an object.

Step 8	**end**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Step 9	**show track** [*object-number*] **Example:** `Router# show track 1`	Displays tracking information.

Configuring VRRP Authentication

VRRP ignores unauthenticated VRRP protocol messages. The default authentication type is text authentication.

The following sections describe configuration tasks for VRRP authentication. The task you perform depends on whether you want to use text authentication, a simple MD5 key string, or MD5 key chains for authentication.

- Configuring VRRP MD5 Authentication Using a Key String, page 15
- Configuring VRRP MD5 Authentication Using a Key Chain, page 16
- Verifying the VRRP MD5 Authentication Configuration, page 18
- Configuring VRRP Text Authentication, page 19

How VRRP MD5 Authentication Works

MD5 authentication provides greater security than the alternative plain text authentication scheme. MD5 authentication allows each VRRP group member to use a secret key to generate a keyed MD5 hash of the packet that is part of the outgoing packet. A keyed hash of an incoming packet is generated and if the generated hash does not match the hash within the incoming packet, the packet is ignored.

The key for the MD5 hash can either be given directly in the configuration using a key string or supplied indirectly through a key chain.

A router will ignore incoming VRRP packets from routers that do not have the same authentication configuration for a VRRP group. VRRP has three authentication schemes:

- No authentication
- Plain text authentication
- MD5 authentication

VRRP packets will be rejected in any of the following cases:

- The authentication schemes differ on the router and in the incoming packet.
- MD5 digests differ on the router and in the incoming packet.
- Text authentication strings differ on the router and in the incoming packet.

Restrictions

Interoperability with vendors that may have implemented the RFC 2338 method is not enabled.

Text authentication cannot be combined with MD5 authentication for a VRRP group at any one time. When MD5 authentication is configured, the text authentication field in VRRP hello messages is set to all zeroes on transmit and ignored on receipt, provided the receiving router also has MD5 authentication enabled.

Configuring VRRP MD5 Authentication Using a Key String

Perform this task to configure VRRP MD5 authentication using a key string.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask* [**secondary**]
5. **vrrp** *group* **priority** *priority*
6. **vrrp** *group* **authentication md5 key-string** [**0** | **7**] *key-string* [**timeout** *seconds*]
7. **vrrp** *group* **ip** [*ip-address* [**secondary**]]
8. Repeat Steps 1 through 7 on each router that will communicate.
9. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet0/1`	Configures an interface type and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 5	**vrrp** *group* **priority** *priority* **Example:** `Router(config-if)# vrrp 1 priority 110`	Configures VRRP priority.

	Command	Purpose

Step 6	`vrrp group authentication md5 key-string [0 | 7] key-string [timeout seconds]`
	Example:
	`Router(config-if)# vrrp 1 authentication md5 key-string d00b4r987654321a timeout 30`
		Configures an authentication string for VRRP MD5 authentication.
		• The *key* argument can be up to 64 characters in length and it is recommended that at least 16 characters be used.
		• No prefix to the *key* argument or specifying **0** means the key will be unencrypted.
		• Specifying **7** means the key will be encrypted. The key-string authentication key will automatically be encrypted if the **service password-encryption** global configuration command is enabled.
		• The timeout value is the period of time that the old key string will be accepted to allow configuration of all routers in a group with a new key.
		Note All routers within the VRRP group must be configured with the same authentication string. If the same authentication string is not configured, the routers in the VRRP group will not communicate with each other and any misconfigured router will change its state to master.

Step 7	`vrrp group ip [ip-address [secondary]]`
	Example:
	`Router(config-if)# vrrp 1 ip 10.0.0.3`
		Enables VRRP on an interface and identifies the IP address of the virtual router.

Step 8	Repeat Steps 1 through 7 on each router that will communicate.	—

Step 9	`end`
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Configuring VRRP MD5 Authentication Using a Key Chain

Perform this task to configure VRRP MD5 authentication using a key chain. Key chains allow a different key string to be used at different times according to the key chain configuration. VRRP will query the appropriate key chain to obtain the current live key and key ID for the specified key chain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **key chain** *name-of-chain*
4. **key** *key-id*

5. **key-string** *string*
6. **exit**
7. **interface** *type number*
8. **ip address** *ip-address mask* [**secondary**]
9. **vrrp** *group* **priority** *priority*
10. **vrrp** *group* **authentication md5 key-chain** *key-chain*
11. **vrrp** *group* **ip** [*ip-address* [**secondary**]]
12. Repeat Steps 1 through 11 on each router that will communicate.
13. **end**

DETAILED STEPS

	Command	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`key chain`** *`name-of-chain`*
	Example:
	`Router(config)# key chain vrrp1`
		Enables authentication for routing protocols and identifies a group of authentication keys.

Step 4	**`key`** *`key-id`*
	Example:
	`Router(config-keychain)# key 100`
		Identifies an authentication key on a key chain.
		• The *key-id* must be a number.

Step 5	**`key-string`** *`string`*
	Example:
	`Router(config-keychain-key)# key-string mno172`
		Specifies the authentication string for a key.
		• The *string* can be 1 to 80 uppercase or lowercase alphanumeric characters; the first character cannot be a number.

Step 6	**`exit`**
	Example:
	`Router(config-keychain-key)# exit`
		Returns to global configuration mode.

Step 7	**`interface`** *`type number`*
	Example:
	`Router(config)# interface Ethernet0/1`
		Configures an interface type and enters interface configuration mode.

	Command	Purpose

Step 8	`ip address ip-address mask [secondary]`
	Example:
	`Router(config-if)# ip address 10.21.8.32 255.255.255.0`
		Specifies a primary or secondary IP address for an interface.

Step 9	`vrrp group priority priority`
	Example:
	`Router(config-if)# vrrp 1 priority 110`
		Configures VRRP priority.

Step 10	`vrrp group authentication md5 key-chain key-chain`
	Example:
	`Router(config-if)# vrrp 1 authentication md5 key-chain vrrp1`
		Configures an authentication MD5 key chain for VRRP MD5 authentication.
		• The key chain name must match the name specified in Step 3.
		Note All routers within the VRRP group must be configured with the same authentication string. If the same authentication string is not configured, the routers in the VRRP group will not communicate with each other and any misconfigured router will change its state to master.

Step 11	`vrrp group ip [ip-address [secondary]]`
	Example:
	`Router(config-if)# vrrp 1 ip 10.21.8.12`
		Enables VRRP on an interface and identifies the IP address of the virtual router.

Step 12	Repeat Steps 1 through 11 on each router that will communicate.	—

Step 13	`end`
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Verifying the VRRP MD5 Authentication Configuration

To verify the MD5 authentication configuration, perform the following steps.

SUMMARY STEPS

1. **show vrrp**
2. **debug vrrp authentication**

DETAILED STEPS

Step 1 **show vrrp**

Use this command to verify that the authentication is configured correctly:

```
Router# show vrrp

Ethernet0/1 - Group 1
State is Master
Virtual IP address is 10.21.0.10
Virtual MAC address is 0000.5e00.0101
Advertisement interval is 1.000 sec
Preemption is enabled
 min delay is 0.000 sec
Priority is 100
Authentication MD5, key-string “f00d4s”, timeout 30 secs
Master Router is 10.21.0.1 (local), priority is 100
Master Advertisement interval is 1.000 sec
Master Down interval is 3.609 sec
```

This output shows that MD5 authentication is configured and the f00d4s key string is used. The timeout value is set at 30 seconds.

Step 2 **debug vrrp authentication**

Use this command to verify that both routers have authentication configured, that the MD5 key ID is the same on each router, and that the MD5 key strings are the same on each router:

```
Router# debug vrrp authentication

VRRP: Grp 1 Advertisement from 10.24.1.1 has incorrect authentication type 0 expected 254

!MD5 key IDs differ on each router.

VRRP: Grp 1 recalculate MD5 digest: “3n};oHp8_)_7-C”
VRRP: Grp 1 Advertisement from 10.24.1.1 has FAILED MD5 authentication

!The MD5 key strings differ on each router.

VRRP: Grp 1 received MD5 digest:
“_M_^uMiWo^|t?t2m”
VRRP: Grp 1 Advertisement from 10.24.1.1 has FAILED MD5 authentication

!The text authentication strings differ on each router.

VRRP: Grp 1 Advertisement from 172.24.1.1 has FAILED TEXT authentication
```

Configuring VRRP Text Authentication

Perform this task to configure VRRP text authentication.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **ip address** *ip-address mask* [**secondary**]
5. **vrrp** *group* **authentication text** *text-string*
6. **vrrp** *group* **ip** *ip-address*
7. Repeat Steps 1 through 6 on each router that will communicate.
8. **end**

DETAILED STEPS

	Command	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet0/1`	Configures an interface type and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 10.0.0.1 255.255.255.0`	Specifies a primary or secondary IP address for an interface.
Step 5	**vrrp** *group* **authentication text** *text-string* **Example:** `Router(config-if)# vrrp 1 authentication text textstring1`	Authenticates VRRP packets received from other routers in the group. • If you configure authentication, all routers within the VRRP group must use the same authentication string. • The default string is cisco. **Note** All routers within the VRRP group must be configured with the same authentication string. If the same authentication string is not configured, the routers in the VRRP group will not communicate with each other and any misconfigured router will change its state to master.
Step 6	**vrrp** *group* **ip** *ip-address* **Example:** `Router(config-if)# vrrp 1 ip 10.0.1.20`	Enables VRRP on an interface and identifies the IP address of the virtual router.

	Command	Purpose
Step 7	Repeat Steps 1 through 6 on each router that will communicate.	—

Step 8	**`end`**
	Example:
	`Router(config-if)# end`
		Returns to privileged EXEC mode.

Enabling the Router to Send SNMP VRRP Notifications

The VRRP MIB supports SNMP Get operations, which allow network devices to get reports about VRRP groups in a network from the network management station.

Enabling VRRP MIB trap support is performed through the CLI, and the MIB is used for getting the reports. A trap notifies the network management station when a router becomes a Master or backup router. When an entry is configured from the CLI, the RowStatus for that group in the MIB immediately goes to the active state.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server enable traps vrrp**
4. **snmp-server host** *host community-string* **vrrp**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		- Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`snmp-server enable traps vrrp`**
	Example:
	`Router(config)# snmp-server enable traps vrrp`
		Enables the router to send SNMP VRRP notifications (traps and informs).

Step 4	**`snmp-server host`** *`host community-string`* **`vrrp`**
	Example:
	`Router(config)# snmp-server host myhost.comp.com public vrrp`
		Specifies the recipient of an SNMP notification operation.

Configuration Examples for VRRP

This section provides the following configuration examples:

Configuring VRRP: Example

In the following example, Router A and Router B each belong to three VRRP groups.

In the configuration, each group has the following properties:

- Group 1:
  - Virtual IP address is 10.1.0.10.
  - Router A will become the master for this group with priority 120.
  - Advertising interval is 3 seconds.
  - Preemption is enabled.
- Group 5:
  - Router B will become the master for this group with priority 200.
  - Advertising interval is 30 seconds.
  - Preemption is enabled.
- Group 100:
  - Router A will become the master for this group first because it has a higher IP address (10.1.0.2).
  - Advertising interval is the default 1 second.
  - Preemption is disabled.

Router A

```
interface ethernet 1/0
 ip address 10.1.0.2 255.0.0.0
 vrrp 1 priority 120
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 vrrp 5 priority 100
 vrrp 5 timers advertise 30
 vrrp 5 timers learn
 vrrp 5 ip 10.1.0.50
 vrrp 100 timers learn
 no vrrp 100 preempt
 vrrp 100 ip 10.1.0.100
 no shutdown
```

Router B

```
interface ethernet 1/0
 ip address 10.1.0.1 255.0.0.0
 vrrp 1 priority 100
 vrrp 1 authentication cisco
 vrrp 1 timers advertise 3
 vrrp 1 timers learn
 vrrp 1 ip 10.1.0.10
 vrrp 5 priority 200
 vrrp 5 timers advertise 30
 vrrp 5 timers learn
 vrrp 5 ip 10.1.0.50
 vrrp 100 timers learn
 no vrrp 100 preempt
 vrrp 100 ip 10.1.0.100
 no shutdown
```

VRRP Object Tracking: Example

In the following example, the tracking process is configured to track the state of the line protocol on serial interface 0/1. VRRP on Ethernet interface 1/0 then registers with the tracking process to be informed of any changes to the line protocol state of serial interface 0/1. If the line protocol state on serial interface 0/1 goes down, then the priority of the VRRP group is reduced by 15.

```
track 1 interface Serial0/1 line-protocol
!
interface Ethernet1/0
 ip address 10.0.0.2 255.0.0.0
 vrrp 1 ip 10.0.0.3
 vrrp 1 priority 120
 vrrp 1 track 1 decrement 15
```

VRRP Object Tracking Verification: Example

The following examples verify the configuration shown in the "VRRP Object Tracking: Example" section:

```
Router# show vrrp

Ethernet1/0 - Group 1
  State is Master
  Virtual IP address is 10.0.0.3
  Virtual MAC address is 0000.5e00.0101
  Advertisement interval is 1.000 sec
  Preemption is enabled
   min delay is 0.000 sec
  Priority is 105
   Track object 1 state Down decrement 15
  Master Router is 10.0.0.2 (local), priority is 105
  Master Advertisement interval is 1.000 sec
  Master Down interval is 3.531 sec
```

```
Router# show track

Track 1
  Interface Serial0/1 line-protocol
  Line protocol is Down (hw down)
   1 change, last change 00:06:53
  Tracked by:
   VRRP Ethernet1/0 1
```

VRRP MD5 Authentication Configuration Using a Key String: Example

The following example shows how to configure MD5 authentication using a key string and timeout of 30 seconds:

```
interface Ethernet0/1
 description ed1-cat5a-7/10
 vrrp 1 ip 10.21.0.10
 vrrp 1 priority 110
 vrrp 1 authentication md5 key-string f00c4s timeout 30
 exit
```

VRRP MD5 Authentication Configuration Using a Key Chain: Example

The following example shows how to configure MD5 authentication using a key chain:

```
key chain vrrp1
 key 1
 key-string f00c4s
 exit
!
interface ethernet0/1
 description ed1-cat5a-7/10
 vrrp 1 priority 110
 vrrp 1 authentication md5 key-chain vrrp1
 vrrp 1 ip 10.21.0.10
```

In this example, VRRP queries the key chain to obtain the current live key and key ID for the specified key chain.

VRRP Text Authentication: Example

The following example shows how to configure VRRP text authentication using a text string:

```
interface fastethernet 0/0
 ip address 10.21.8.32 255.255.255.0
 vrrp 10 authentication text stringxyz
 vrrp 10 ip 10.21.8.10
```

Disabling a VRRP Group on an Interface: Example

The following example shows how to disable one VRRP group on Ethernet interface 0/1 while retaining VRRP for group 2 on Ethernet interface 0/2:

```
interface ethernet0/1
 ip address 10.24.1.1 255.255.255.0
 vrrp 1 ip 10.24.1.254
 vrrp 1 shutdown

interface ethernet0/2
 ip address 10.168.42.1 255.255.255.0
 vrrp 2 ip 10.168.42.254
```

VRRP MIB Trap: Example

The following example shows how to enable the VRRP MIB trap support functionality:

```
snmp-server enable traps vrrp
snmp-server host 10.1.1.0 community abc vrrp
```

Additional References

The following sections provide references related to VRRP.

Related Documents

Related Topic	Document Title
VRRP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference*
Key chains and key management commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Routing Protocols Command Reference*
Object tracking	*Configuring Enhanced Object Tracking*
HSRP	*Configuring HSRP*
GLBP	*Configuring GLBP*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
No new MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2338	*Virtual Router Redundancy Protocol*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for VRRP

Table 1 lists the features in this module and provides links to specific configuration information.

For information on a feature in this technology that is not documented here, see the "Cisco IOS IP Application Services Features Roadmap" or the "FHRP Features Roadmap."

Not all commands may be available in your Cisco IOS software release. For details on when support for a specific command was introduced, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1 Feature Information for VRRP

Feature Name	Releases	Feature Configuration Information
FHRP—VRF-Aware VRRP	12.2(15)T 12.0(18)ST 12.2(31)SG 12.2(17d)SXB	The FHRP—VRF-Aware VRRP feature adds VRRP support for VRF-Aware MPLS VPNs. • VRRP Operation, page 2 • Restrictions for VRRP, page 2 There are no new or modified command for this feature.
FHRP—VRRP Enhancements	12.3(14)T	The FHRP—VRRP Enhancements feature adds support for the following capabilities: • MD5 Authentication—Added to routers that are configured for VRRP, similar to HSRP, to provide a method of authenticating peers using a more simple method than the method in RFC 2338. • Bridged Virtual Interface (BVI)—Added the capability to configure VRRP on BVIs. This functionality is similar to the existing HSRP support for BVIs. The following sections provide information about this feature: • Restrictions for VRRP, page 2 • Configuring VRRP Authentication, page 14 The following command was introduced by this feature: **debug vrrp authentication**. The following commands were modified by this feature: **vrrp authentication** and **show vrrp**.

Table 1 Feature Information for VRRP (continued)

Feature Name	Releases	Feature Configuration Information
ISSU—VRRP	12.2(33)SRC Cisco IOS XE Release 2.1	VRRP supports In Service Software Upgrade (ISSU). ISSU allows a high-availability (HA) system to run in Stateful Switchover (SSO) mode even when different versions of Cisco IOS software are running on the active and standby Route Processors (RPs) or line cards. This feature provides customers with the same level of HA functionality for planned outages due to software upgrades as is available with SSO for unplanned outages. That is, the system can switch over to a secondary RP and continue forwarding packets without session loss and with minimal or no packet loss. This feature is enabled by default. The following sections provide information about this feature: • ISSU—VRRP, page 6 There are no new or modified commands for this feature.
SSO—VRRP	12.2(33)SRC Cisco IOS XE Release 2.1	VRRP is now SSO aware. VRRP can detect when a router is failing over to the secondary RP and continue in its current VRRP group state. This feature is enabled by default. The following sections provide information about this feature: • SSO—VRRP, page 7 • Customizing VRRP, page 7 The following commands were introduced or modified by this feature: **debug vrrp ha**, **vrrp sso**, **show vrrp**.
Virtual Router Redundancy Protocol	12.2(13)T 12.2(14)S Cisco IOS XE Release 2.1	VRRP enables a group of routers to form a single virtual router to provide redundancy. The LAN clients can then be configured with the virtual router as their default gateway. The virtual router, representing a group of routers, is also known as a VRRP group. All sections provide information about this feature. The following commands were introduced by this feature: **debug vrrp all**, **debug vrrp error**, **debug vrrp events**, **debug vrrp packets**, **debug vrrp state**, **show vrrp**, **show vrrp interface**, **vrrp authentication**, **vrrp description**, **vrrp ip**, **vrrp preempt**, **vrrp priority**, **vrrp timers advertise**, **vrrp timers learn**.

Table 1 Feature Information for VRRP (continued)

Feature Name	Releases	Feature Configuration Information
VRRP Object Tracking	12.3(2)T 12.2(25)S	The VRRP Object Tracking feature extends the capabilities of the VRRP to allow tracking of specific objects within the router that can alter the priority level of a virtual router for a VRRP group. The following sections provide information about this feature: • VRRP Object Tracking, page 6 • Configuring VRRP Object Tracking, page 12 The following command was introduced by this feature: **vrrp track**. The following command was modified by this feature: **show track**.
VRRP MIB—RFC 2787	12.3(11)T Cisco IOS XE Release 2.1	The VRRP MIB—RFC 2787 feature enables an enhancement to the MIB for use with SNMP-based network management. The feature adds support for configuring, monitoring, and controlling routers that use VRRP. The following sections provide information about this feature: • Disabling VRRP on an Interface, page 11 • Enabling the Router to Send SNMP VRRP Notifications, page 21 The following command was introduced by this feature: **vrrp shutdown**. The following commands were modified by this feature: **snmp-server enable traps** and **snmp-server host**.

Glossary

virtual router—One or more VRRP routers that form a group. The virtual router acts as the default gateway router for LAN clients. Also known as a VRRP group.

virtual router backup—One or more VRRP routers that are available to assume the role of forwarding packets if the virtual router master fails.

virtual router master—The VRRP router that is currently responsible for forwarding packets sent to the IP addresses of the virtual router. Usually the virtual router master also functions as the IP address owner.

virtual IP address owner—The VRRP router that owns the IP address of the virtual router. The owner is the router that has the virtual router address as its physical interface address.

VRRP router—A router that is running VRRP.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

UDP

Configuring IPv4 Broadcast Packet Handling

First Published: February 4, 2008
Last Updated: February 4, 2008

This module explains what IPv4 broadcast packets are, when they are used, and how to customize your router's configuration for situations when the default behavior for handling IPv4 broadcast packets isn't appropriate.

Note This module also explains some common scenarios that require customizing IPv4 broadcast packet handling by routers. For example, UDP forwarding of Dynamic Host Configuration Protocol (DHCP) traffic to ensure broadcast packets sent by DHCP clients can reach DHCP servers that are not on the same network segment as the client. Configuration tasks and examples are also provided in this module. All further references to IP addresses in this document use only IP in the text, not IP.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for IP Broadcast Packet Handling" section on page 26.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Information About IPv4 Broadcast Packet Handling

To configure IPv4 broadcast packet handling, you should understand the following concepts:

- IP Addresses, page 2
- Early IP Implementations, page 4
- DHCP, page 5
- Forwarding UDP Broadcast Packets, page 5
- Flooding UDP Broadcast Packets, page 5
- UDP Broadcast Packet Case Study, page 6

IP Addresses

This section describes the four types of IP addresses:

- IP Unicast Address, page 2
- IP Broadcast Address, page 2
- IP Directed Broadcast Address, page 3
- IP Multicast, page 4

IP Unicast Address

An IP unicast address is not a broadcast addresses. A packet with an unicast destination IP address is intended for a specific IP host. For example, 172.16.1.1/32. Only the intended host of a unicast packets receives and processes the packet. This definition of the term unicast is included in this document because this term is often used in conjunction with references to types of IP broadcast traffic. For example, when a network administrator is considering upgrading a router in a network, the amount of unicast, multicast, and broadcast traffic must be considered because each type of traffic can have a different affect on the performance of the router.

IP Broadcast Address

IP broadcast packets are sent to the destination IP broadcast address 255.255.255.255 (or the older but still occasionally used IP broadcast address of 000.000.000.000). The broadcast destination IP addresses 255.255.255.255 and 000.000.000.000 are used when a packet is intended for every IP-enabled device on a network.

Note Packets that use the broadcast IP address as the destination IP address are known as broadcast packets.

If routers forwarded IP broadcast packets by default, the packets would have to be forwarded out every interface that is enabled for IP because the 255.255.255.255 IP destination address is assumed to be reachable via every IP enabled interface in the router. Forwarding IP broadcast packets out every interface that is enabled for IP would result in what is known as a broadcast storm (network overload due to high levels of broadcast traffic). In order to avoid the IP packet broadcast storm that would be created if a router forwarded packets with a broadcast IP destination address out every IP-enabled interface, the default behavior for a router is to *not* forward broadcast packets. This is a key difference between routing IP traffic at Layer 3 versus bridging it at Layer 2. Layer 2 bridges by default forward IP broadcast traffic out every interface that is in a forwarding state, which can lead to scalability problems.

Note A thorough explanation of the differences between routing and bridging IP traffic is beyond the scope of this document. See the "Related Documents" section on page 24 for information on other resources for learning more about routing and bridging IP traffic.

Some TCP/IP protocols use the IP broadcast address to either communicate with all of the hosts on a network segment or to identify the IP address of a specific host on a network segment. For example:

- Routing Information Protocol (RIP) version 1 sends routing table information using the IP broadcast address so that any other host on the network segment running RIP version 1 can receive and process the updates.
- The Address Resolution Protocol (ARP) is used to determine the Layer 2 MAC address of the host that owns a specific Layer 3 IP address. ARP sends an IP broadcast packet (that is also a Layer 2 broadcast frame) on the local network. All of the hosts on the local network receive the ARP broadcast packet because it is sent to as a Layer 2 broadcast frame. All of the hosts on the local network process the ARP packet because it is sent to the IP broadcast address. Only the host that owns the IP address indicated in the data area of the ARP packet responds to the ARP broadcast packet.

IP Directed Broadcast Address

An IP directed broadcast is intended to reach all hosts on a remote network. A router that needs to send data to a remote IP host when only the IP network address is known uses an IP directed broadcast to reach the remote host. For example, a directed broadcast sent by a host with an IP address of 192.168.100.1 with a destination IP address of 172.16.255.255 is intended only for hosts that are in the 172.16.0.0 address space (hosts that have an IP address that begins with 172.16.0.0).

An IP directed broadcast packet is routed through the network as a unicast packet until it arrives at the target subnet, where it is converted into a Layer 2 broadcast frame (MAC address of FFFF.FFFF.FFFF). Because of the nature of the IP addressing architecture, only the last router in the chain, the one that is connected directly to the target subnet, can conclusively identify a directed broadcast. For example, only a router with an interface connected to a network using an IP address in the 172.16.0.0/16 address space such as 172.16.1.1/16 can determine that a packet sent to 172.16.255.255 is a directed broadcast and convert it to a Layer 2 broadcast that is received by all hosts on the local network. The other routers in the network that are not connected to the 172.16.0.0/16 network forward packets addressed to 172.16.255.255 as if they were for a specific IP host.

All of the hosts on the remote network receive IP directed broadcasts after they are converted to Layer 2 broadcast frames. Ideally only the intended destination host will fully process the IP directed broadcast and respond to it. It is possible, however, to use IP directed broadcasts for malicious purposes. For example, IP directed broadcasts are used in the popular "smurf" Denial of Service DoS attack and derivatives thereof. In a "smurf" attack, the attacker sends Internet Control Message Protocol (ICMP) echo requests (pings) to a directed broadcast address using the source IP address of the device that is the target of the attack. The target is usually a host inside a company's network such as a web server. The ICMP echo requests are sent to an IP directed broadcast address in the company's network that causes all the hosts on the target subnet to send ICMP echo replies to the device under attack. By sending a continuous stream of such requests, the attacker can create a much larger stream of replies, which can completely inundate the host that is under attack. For information on how IP directed broadcasts are used in DoS attacks, search the Internet for "IP directed broadcasts," "denial of service," and "smurf attacks."

Due to the security implications of allowing a router to forward directed broadcasts and the reduction in applications that require directed broadcasts, IP directed broadcasts are disabled by default in Cisco IOS Release 12.0 and later releases. If your network requires support for IP directed broadcasts, you can enable it on the interfaces that you want to translate the IP directed broadcasts to Layer 2 broadcasts using the **ip directed-broadcast** command. For example, if your router is receiving IP directed broadcasts on Fast Ethernet interface 0/0 for the network address assigned to Fast Ethernet interface 0/1, and you want the IP directed broadcasts to be translated to Layer 2 broadcasts out interface Fastethernet 0/1, configure the **ip directed-broadcast** command on Fast Ethernet interface 0/1. You can specify an access list to control which IP directed broadcasts are translated to Layer 2 broadcasts. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to Layer 2 broadcasts. For example, if you know that the only legitimate source IP address of any IP directed broadcasts in your network is 192.168.10.2, create an extended IP access list allowing traffic from 192.168.10.2 and assign the access list with the **ip directed-broadcast** *access-list* command.

IP Multicast

IP multicast addresses are intended to reach an arbitrary subset of the hosts on a local network. IP broadcast addresses create a problem because every host must receive and process the data in each packet to determine if it contains information that the host must process further. IP multicast addresses resolve this problem by using well-known IP addresses that a host must be configured to recognize before it will process packets addressed to it. When a host receives an IP multicast packet, the host compares the IP multicast address with the list of multicast addresses it is configured to recognize. If the host is not configured to recognize the IP multicast address, the host ignores the packet instead of processing it further to analyze the data in the packet. Because the host can ignore the packet it spends less time and fewer resources than it would have had to spend if the packet had been an IP broadcast that had to be processed all the way to the data layer before it was discarded.

Table 1 shows the range of IP addresses reserved for multicast addresses.

Table 1 IIP Multicast Address Range

Class	Range
D	224.0.0.0 to 239.255.255.255/32 (255.255.255.255)

Most of the TCP/IP routing protocols in use today use IP multicast addresses to send routing updates and other information to hosts on the same local network that are running the same routing protocol. Many other applications such as audio/video streaming over the Internet use IP multicast addresses. For a list of the currently assigned IP multicast addresses see *Internet Multicast Addresses* at this URL: http://www.iana.org/assignments/multicast-addresses.

Information on configuring network devices for IP multicast support is available in the following documentation:

- *Cisco IOS IP Multicast Configuration Guide*
- *Cisco IOS IP Multicast Command Reference*

Early IP Implementations

Several early IP implementations do not use the current broadcast address standard of 255.255.255.255. Instead, they use the old standard, which calls for all zeros (000.000.000.000) instead of all ones to indicate broadcast addresses. Many of these implementations do not recognize an all-1s broadcast address and fail to respond to the broadcast correctly. Others forward all-1s broadcasts by default, which causes a serious network overload known as a *broadcast storm*. Implementations that exhibit these problems include systems based on versions of Berkeley Standard Distribution (BSD) UNIX prior to Version 4.3.

DHCP

DHCP requires that the client (host requiring information from the DHCP server) send broadcast packets to find a DHCP server to request configuration information from. If the DHCP server is not on the same network segment as the client that is sending the DHCP broadcasts, the router must be configured to forward the DHCP requests to the appropriate network.

For more information on DHCP, see RFC 2131 *Dynamic Host Configuration Protocol,* at http://www.ietf.org/rfc/rfc2131.txt.

Forwarding UDP Broadcast Packets

UDP broadcast packets are used by TCP/IP protocols such as DHCP and applications that need to send the same data to multiple hosts concurrently. Because routers by default do not forward broadcast packets you need to customize your router's configuration if your network has UDP broadcast traffic on it. One option for forwarding UDP broadcast packets is to use the UDP forwarding feature. UDP forwarding rewrites the broadcast IP address of a UDP packet to either a unicast (specific host) IP address or a directed IP broadcast. After the address is rewritten the UDP packet is forwarded by all of the routers in the path to the destination network without requiring additional configuration changes on the other routers.

Flooding UDP Broadcast Packets

Another option for ensuring the UDP broadcast packets reach the hosts that need to receive them is to allow IP broadcasts to be flooded throughout your network in a controlled fashion using the forwarding database created by the Layer 2 bridging spanning-tree Protocol (STP). Enabling this feature also prevents flooding loops. In order to support this capability, the Cisco IOS software on your router must include support for transparent bridging, and transparent bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still will be able to receive broadcasts. However, the interface will never forward broadcasts it receives, and the router will never use that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

In order to be considered for flooding, packets must meet the following criteria. (these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast (FFFF.FFFF.FFFF).
- The packet must be an IP-level broadcast (255.255.255.255).
- The packet must be a Trivial File Transfer Protocol (TFTP), DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP protocol specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

If you want to send the flooded UDP packets to a specific host, you can change the Layer 3 IP broadcast address of the flooded UDP packets with the **ip broadcast-address** command in interface configuration mode. The address of the flooded UDP packets can be set to any desired IP address. The source address of the flooded UDP packet is never changed. The TTL value of the flooded UDP packet is decremented.

After a decision has been made to send the datagram out on an interface (and the destination IP address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists if they are present on the output interface.

If no actual bridging is desired, you can configure a type-code bridging filter that will deny all packet types from being bridged. Refer to the "Configuring Transparent Bridging" module of the *Cisco IOS Bridging and IBM Networking Configuration Guide* for more information about using access lists to filter bridged traffic. The spanning-tree database is still available to the IP forwarding code to use for the flooding.

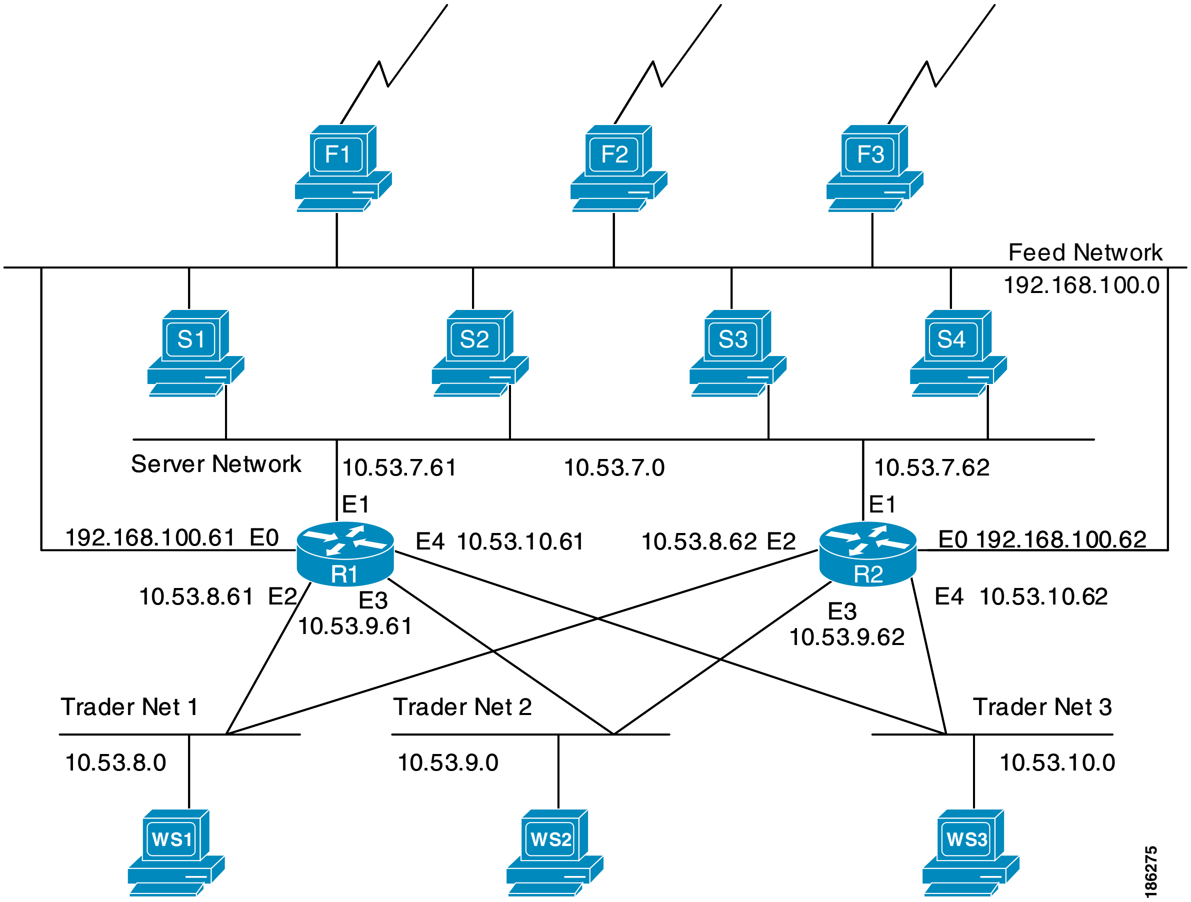
Speeding Up Flooding of IP Broadcasts

You can speed up flooding of UDP datagrams using the spanning-tree algorithm. Used in conjunction with the **ip forward-protocol spanning-tree** command in global configuration mode, this feature boosts the performance of spanning-tree-based UDP flooding by a factor of about four to five times. The feature, called *turbo flooding*, is supported over Ethernet interfaces configured for Advanced Research Projects Agency (ARPA) encapsulated, FDDI, and high-level data link control (HDLC)-encapsulated serial interfaces. However, it is not supported on Token Ring interfaces. As long as the Token Rings and the non-HDLC serial interfaces are not part of the bridge group being used for UDP flooding, turbo flooding will behave normally.

UDP Broadcast Packet Case Study

This case study is from a trading floor application in a financial company. The workstations (WS1, WS2, and WS3) in Figure 2 receive financial data from the feed network. The financial data is sent using UDP broadcasts.

Figure 1 Topology that Requires UDP Broadcast Forwarding



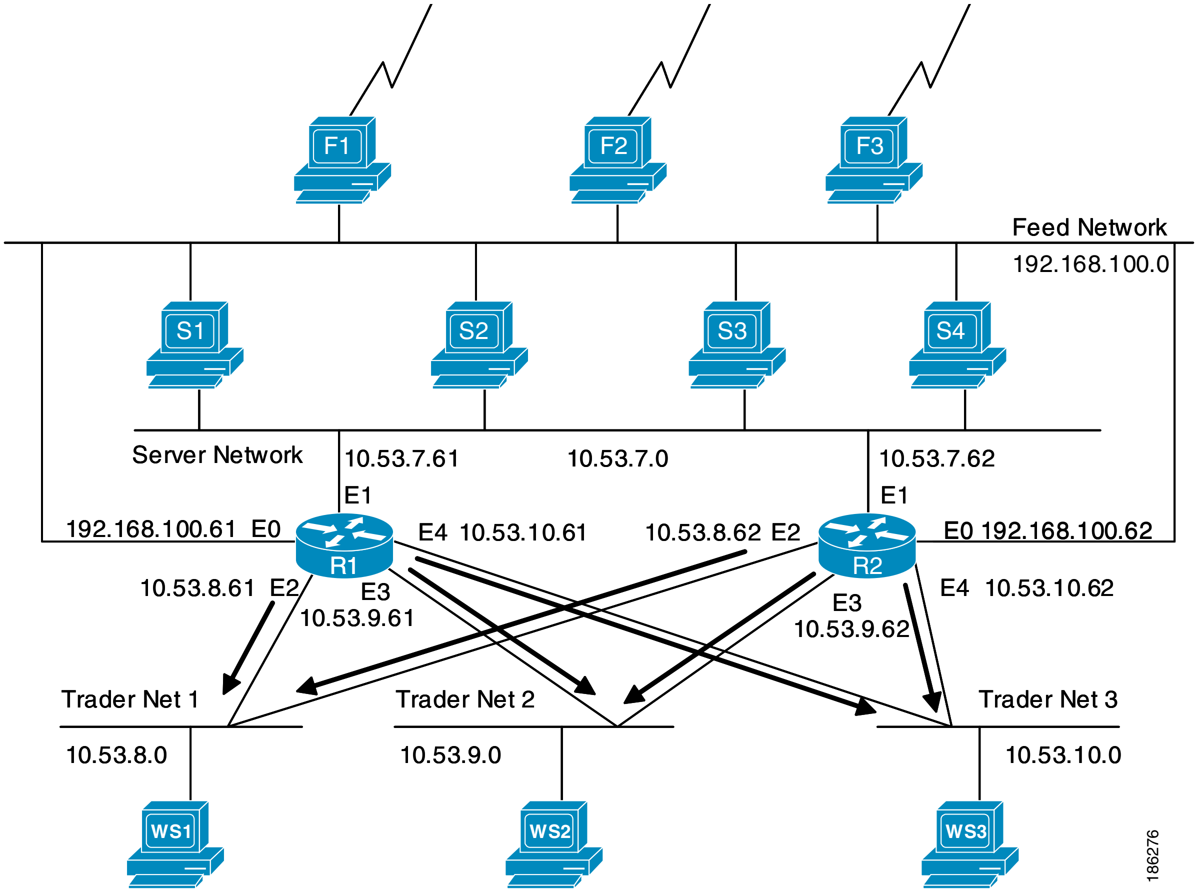
The following sections explain the possible solutions for this application:

- UDP Broadcast Packet Forwarding, page 7
- UDP Broadcast Packet Flooding, page 9

UDP Broadcast Packet Forwarding

The first option is UDP broadcast packet using helper addresses. To configure helper addressing, you must specify the **ip helper-address** command on every interface on every router that receives a UDP broadcast that needs to be forwarded. On router 1 and router 2 in Figure 1, IP helper addresses can be configured to move data from the server network to the trader networks. However IP helper addressing was determined not to be an optimal solution for this type of topology because each router receives unnecessary broadcasts from the other router, as shown in Figure 2.

Figure 2 Flow of UDP packets from routers to trader networks using IP helper addressing Packets from Routers to Trader Networks Using IP Helper Addressing



In this case, router 1 receives each broadcast sent by router 2 three times, one for each segment, and router 2 receives each broadcast sent by router 1 three times, one for each segment. When each broadcast is received, the router must analyze it and determine that the broadcast does not need to be forwarded. As more segments are added to the network, the routers become overloaded with unnecessary traffic, which must be analyzed and discarded.

When IP helper addressing is used in this type of topology, no more than one router can be configured to forward UDP broadcasts (unless the receiving applications can handle duplicate broadcasts). This is because duplicate packets arrive on the trader network. This restriction limits redundancy in the design and can be undesirable in some implementations.

To send UDP broadcasts bidirectionally in this type of topology, a second **ip helper address** command must be applied to every router interface that receives UDP broadcasts. As more segments and devices are added to the network, more ip helper address commands are required to reach them, so the administration of these routers becomes more complex over time.

Note Bidirectional traffic in this topology significantly impacts router performance.

Although IP helper addressing is well-suited to nonredundant, nonparallel topologies that do not require a mechanism for controlling broadcast loops, in view of these drawbacks, IP helper addressing does not work well in this topology. To improve performance, the network designers considered four other alternatives:

- Setting the broadcast address on the servers to all ones (255.255.255.255)—This alternative was dismissed because the servers have more than one interface, causing server broadcasts to be sent back onto the feed network. In addition, some workstation implementations do not allow all-ones broadcasts when multiple interfaces are present.
- Setting the broadcast address of the servers to the major network broadcast IP address—This alternative was dismissed because the TCP/IP implementation on the servers does not allow the use of major network IP broadcast addresses when the network is subnetted.
- Eliminating the subnets and letting the workstations use Address Resolution Protocol (ARP) to learn addresses—This alternative was dismissed because the servers cannot quickly learn an alternative route in the event of a primary router failure.
- UDP Broadcast Packet Flooding—This alternative using the spanning-tree topology created with transparent bridging to forward UDP broadcast packets in a redundant topology while avoiding loops and duplicate broadcast traffic.

After eliminating the first three alternatives which used UDP forwarding, the network designers chose the fourth option, UDP broadcast packet flooding (sometimes referred to by the shorter name of UDP flooding). UDP flooding supports redundancy without duplicating packets and ensures fast convergence and minimal loss of data when a router fails.

UDP Broadcast Packet Flooding

UDP flooding uses the spanning-tree algorithm to forward packets in a controlled manner. Bridging is enabled on each router interface for the sole purpose of building the spanning-tree. The spanning-tree prevents loops by stopping a broadcast from being forwarded out an interface on which the broadcast was received. The spanning-tree also prevents packet duplication by placing certain interfaces in the blocked state (so that no packets are forwarded) and other interfaces in the forwarding state (so that packets that need to be forwarded are forwarded).

To enable UDP flooding, the router must be running software that supports transparent bridging and bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured for an interface, the interface will receive broadcasts, but the router will not forward those broadcasts and will not use that interface as a destination for sending broadcasts received on a different interface.

Note Releases prior to Cisco IOS Software Release 10.2 do not support flooding subnet broadcasts.

When configured for UDP flooding, the router uses the destination address specified by the **ip broadcast-address** command on the output interface to assign a destination address to a flooded UDP datagram. Thus, the destination address might change as the datagram propagates through the network. The source address, however, does not change.

With UDP flooding, both routers shown in Figure 3 use a spanning-tree to control the network topology for the purpose of forwarding broadcasts. The **bridge protocol** command can specify either the **dec** keyword (for the DEC spanning-tree protocol) or the **ieee** keyword (for the IEEE Ethernet protocol). All routers in the network must enable the same spanning-tree protocol. The **ip forward-protocol spanning-tree** command uses the database created by the **bridge protocol** command. Only one broadcast packet arrives at each segment, and UDP broadcasts can traverse the network in both directions.

Because bridging is enabled only to build the spanning-tree database, use access lists to prevent the spanning-tree from forwarding non-UDP traffic. The configuration examples later in this module configure an access list that blocks all bridged packets.

To determine which interface forwards or blocks packets, the router configuration specifies a path cost for each interface. The default path cost for Ethernet is 100. Setting the path cost for each interface on router 2 to 50 causes the spanning-tree algorithm to place the interfaces in router 2 in forwarding state. Given the higher path cost (100) for the interfaces in router 1, the interfaces in router 1 are in the blocked state and do not forward the broadcasts. With these interface states, broadcast traffic flows through router 2. If router 2 fails, the spanning-tree algorithm will place the interfaces in router 1 in the forwarding state, and router 1 will forward broadcast traffic.

With one router forwarding broadcast traffic from the server network to the trader networks, it is desirable to have the other forward unicast traffic. For that reason, each router enables the ICMP Router Discovery Protocol (IRDP), and each workstation on the trader networks runs the IRDP daemon. On router 1, the **preference** keyword sets a higher IRDP preference than does the configuration for router 2, which causes each IRDP daemon to use router 1 as its preferred default gateway for unicast traffic forwarding. Users of those workstations can use **netstat -rn** to see how the routers are being used.

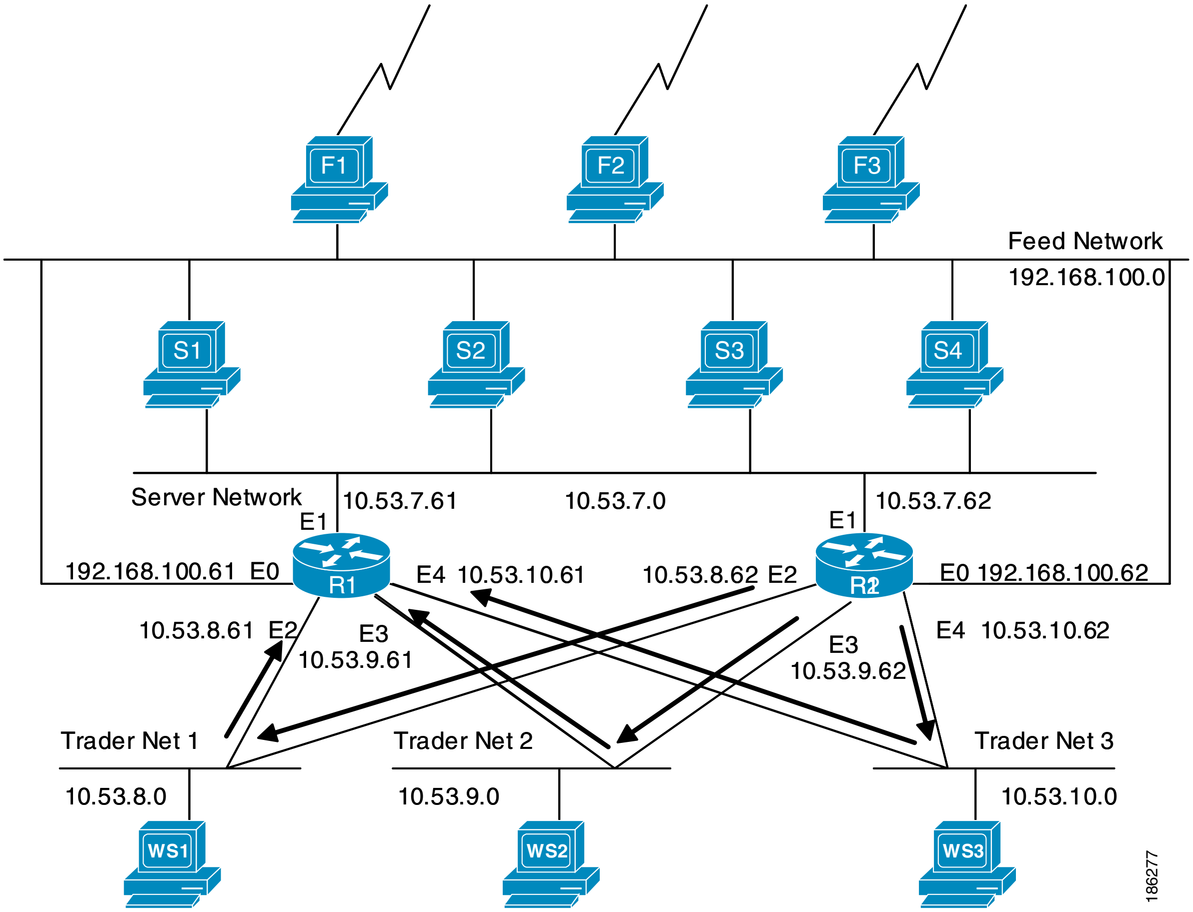
On the routers, the **holdtime**, **maxadvertinterval**, and **minadvertinterval** keywords reduce the advertising interval from the default so that the IRDP daemons running on the hosts expect to see advertisements more frequently. With the advertising interval reduced, the workstations will adopt router 2 more quickly if router 1 becomes unavailable. With this configuration, when a router becomes unavailable, IRDP offers a convergence time of less than one minute.

IRDP is preferred over the Routing Information Protocol (RIP) and default gateways for the following reasons:

- RIP takes longer to converge, typically from one to two minutes.
- Configuration of router 1 as the default gateway on each Sun workstation on the trader networks would allow those Sun workstations to send unicast traffic to router 1, but would not provide an alternative route if router 1 becomes unavailable.

Figure 3 shows how data flows when the network is configured for UDP flooding.

Figure 3 Data flow with UDP Flooding and IRDP



Note This topology is broadcast intensive—broadcasts sometimes consume 20 percent of the 10MB Ethernet bandwidth. However, this is a favorable percentage when compared to the configuration of IP helper addressing, which, in the same network, causes broadcasts to consume up to 50 percent of the 10 MB Ethernet bandwidth.

If the hosts on the trader networks do not support IRDP, the Hot Standby Routing Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), or Gateway Load Balancing Protocol (GLBP) can be used to select which router will handle unicast traffic. These protocols allow the standby router to take over quickly if the primary router becomes unavailable. For information about First Hop Redundancy Protocols see the "First Hop Redundancy Protocols" section of the *Cisco IOS IP Application Services Configuration Guide*.

Enable turbo flooding on the routers to increase the performance of UDP flooding.

Note Turbo flooding increases the amount of processing that is done at interrupt level, which increases the CPU load on the router. Turbo flooding may not be appropriate on routers that are already under high CPU load or that must also perform other CPU-intensive activities.

How to Configure IP Broadcast Packet Handling

This section contains the following tasks:

- Enabling IP Directed Broadcasts, page 12
- Enabling Forwarding of UDP Broadcast Packets, page 14
- Configuring the Default IP Broadcast Address, page 18
- Configuring UDP Broadcast Packet Flooding, page 21

Enabling IP Directed Broadcasts

IP directed broadcasts are dropped by default. Dropping IP directed broadcasts reduces the risk of DoS attacks.

You can enable forwarding of IP directed broadcasts on an interface where the broadcast becomes a physical broadcast. You enable the translation of directed IP broadcast packets to Layer 2 broadcast frames on the interface that is connected to the IP network that the IP directed broadcast is addressed to. For example, if you need to translate IP directed broadcasts with the IP destination address of 172.16.10.255 to Layer 2 broadcast frames, you enable the translation on the interface that is connected to IP network 172.16.10.0/24.

You can specify an access list to control which directed broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts.

IP directed broadcasts are disabled by default in Cisco IOS Release 12.0 and newer releases. If your network requires support for IP directed broadcasts, perform one of the following tasks:

- Enabling IP Directed Broadcasts Without an Access List
- Enabling IP Directed Broadcasts With an Access List

Enabling IP Directed Broadcasts Without an Access List

Perform this task to permit the forwarding of IP directed broadcasts from any source.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *address mask*
5. **ip directed-broadcast**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/1`	Specifies an interface and enters interface configuration mode.
Step 4	**ip address** *address mask* **Example:** `Router(config-if)# ip address 172.16.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 5	**ip directed-broadcast** **Example:** `Router(config-if)# ip directed-broadcast`	Enables IP directed broadcasts on the interface. Configure this command on the interface that is connected to the IP network address of the directed broadcast packets. In this example the directed broadcast packets are addressed to 172.16.10.255.

Step 6	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Enabling IP Directed Broadcasts With an Access List

Perform this task to limit the forwarding of IP directed broadcasts by applying an access list to the **ip directed-broadcast** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list list 100-199 permit ip** *source-address mask destination-address mask*
4. **interface** *type number*
5. **ip address** *address mask*
6. **ip directed-broadcast** *access-list*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**access-list list 100-199 permit ip** *source-address mask destination-address mask* **Example:** `Router(config)# access-list 100 permit ip 10.4.9.167 0.0.0.0 172.16.10.0 0.0.0.255`	Creates an access-list to limit the IP directed broadcasts that are forwarded. In this example the IP directed broadcasts are sent by the host with the IP address of 10.4.9.167 to the IP directed broadcast address 172.16.10.255.
Step 4	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 5	**ip address** *address mask* **Example:** `Router(config-if)# ip address 172.16.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 6	**ip directed-broadcast** *access-list* **Example:** `Router(config-if)# ip directed-broadcast 100`	Enables IP directed broadcasts on the interface for broadcast packets that are allowed by the access list you assigned. Configure this command on the interface that is connected to the IP network address of the directed broadcast packets. In this example the directed broadcast packets are addressed to 172.16.10.255.

Step 7	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Enabling Forwarding of UDP Broadcast Packets

You can enable forwarding of UDP broadcast packets, such as DHCP requests, to a host, or to multiple hosts on the same target network. When a UDP broadcast packet is forwarded, the destination IP address is rewritten to match the address that you configure. For example, the **ip helper-address 172.16.10.2** command rewrites the IP destination address from 255.255.255.255 to 172.16.10.2.

To enable UDP broadcast packet forwarding to specific host, use a specific host IP address as the helper address when you configure the **ip helper-address** *address* command. To enable UDP broadcast packet forwarding to a range of hosts to allow for load sharing and redundancy, use an IP directed broadcast address as the helper address when you configure the **ip helper-address** *address* command.

Default UDP Port Numbers

If a helper address is specified and UDP forwarding is enabled, broadcast packets destined to the following port numbers are forwarded by default:

- Trivial File Transfer Protocol (TFTP) (port 69)
- Domain Naming System (port 53)
- Time service (port 37)
- NetBIOS Name Server (port 137)
- NetBIOS Datagram Server (port 138)
- Boot Protocol (BOOTP) client and server packets (ports 67 and 68)
- TACACS service (port 49)
- IEN-116 Name Service (port 42)

If your network requires support for forwarding UDP broadcasts, perform one of the following tasks:

- Enabling Forwarding of UDP Broadcast Packets to a Specific Host
- Enabling Forwarding of UDP Broadcast Packets to a Range of Hosts

Enabling Forwarding of UDP Broadcast Packets to a Specific Host

Perform this task to enable UDP broadcast packet forwarding to a single host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip forward-protocol udp**
4. **interface** *type number*
5. **ip address** *address mask*
6. **ip helper-address** *address*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip forward-protocol udp** **Example:** `Router(config)# ip forward-protocol udp`	Enables forwarding of UDP broadcasts.
Step 4	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/1`	Specifies an interface and enters interface configuration mode.
Step 5	**ip address** *address mask* **Example:** `Router(config-if)# ip address 172.16.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 6	**ip helper-address** *address* **Example:** `Router(config-if)# ip helper-address 172.16.10.2`	Enables an IP helper address for the interface that is receiving the UDP broadcast packets. In this example the IP destination address of the IP UDP broadcast packets is rewritten to 172.16.10.2.

Step 7	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Enabling Forwarding of UDP Broadcast Packets to a Range of Hosts

Perform this task to enable UDP broadcast packet forwarding to a range of hosts to allow for load sharing between the destination hosts and to provide redundancy in the event one or more of the destination hosts fail.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip forward-protocol udp**
4. **interface** *type number*

5. **ip address** *address mask*
6. **ip helper-address** *address*
7. **interface** *type number*
8. **ip address** *address mask*
9. **ip directed-broadcast**
10. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`ip forward-protocol udp`**
	Example:
	`Router(config)# ip forward-protocol udp`
		Enables forwarding of UDP broadcasts.

Step 4	**`interface`** *`type number`*
	Example:
	`Router(config)# interface fastethernet 0/0`
		Specifies an interface and enters interface configuration mode.

Step 5	**`ip address`** *`address mask`*
	Example:
	`Router(config-if)# ip address 192.168.10.1 255.255.255.0`
		Assigns an IP address to the interface.

	Command or Action	Purpose
Step 6	**ip helper-address** *address* **Example:** `Router(config-if)# ip helper-address 172.16.10.255`	Enables an IP helper address for the interface that is receiving the UDP broadcast packets. In this example an IP directed broadcast address is used. The IP destination address of the IP UDP broadcast packets is rewritten to 172.16.10.255. All of the hosts on the 172.16.10.0/24 network that support the application or service that the UDP broadcast packets are intended for will respond to the UDP broadcast packets. **Note** This often results in the source of the UDP broadcast packets receiving responses from two or more hosts. In most circumstances the source of the UDP broadcast packets accepts the first response and ignores any subsequent responses. In some situations the source of the UDP broadcast packets cannot handle duplicate responses and reacts by crashing, or other unexpected behavior.
Step 7	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/1`	Specifies an interface and enters interface configuration mode.
Step 8	**ip address** *address mask* **Example:** `Router(config-if)# ip address 172.16.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 9	**ip directed-broadcast** **Example:** `Router(config-if)# ip directed-broadcast`	Enables IP directed broadcasts on the interface that is transmitting the UDP broadcasts.

Step 10	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Configuring the Default IP Broadcast Address

The Cisco IOS software supports sending IP broadcasts on both LANs and WANs. There are several ways to indicate an IP broadcast address. Currently, the most popular way, and the default, is an address consisting of all ones (255.255.255.255), although the software can be configured to generate any form of IP broadcast address such as all zeros (0.0.0.0), and directed broadcasts such as 172.16.255.255. Cisco IOS software can receive and process most IP broadcast addresses.

If your network requires that you modify the default IP broadcast address, perform one of these tasks:

- Changing the Default IP Broadcast Address for all Interfaces to 0.0.0.0 on Routers Without Nonvolatile Memory

- Changing the Default IP Broadcast Address for All Interfaces to 0.0.0.0 on Routers with Nonvolatile Memory
- Changing the IP Broadcast Address to any IP Address on One or More Interfaces in a Router

Changing the Default IP Broadcast Address for all Interfaces to 0.0.0.0 on Routers Without Nonvolatile Memory

If you router does not have nonvolatile memory (NVRAM), and you need to change the IP broadcast address to 0.0.0.0, you must change the IP broadcast address manually by setting jumpers in the processor configuration register. Setting bit 10 causes the device to use all 0s. Bit 10 interacts with bit 14, which controls the network and subnet portions of the broadcast address. Setting bit 14 causes the device to include the network and subnet portions of its address in the broadcast address. Table 2 shows the combined effect of setting bits 10 and 14.

Table 2 Configuration Register Settings for Broadcast Address Destination

Bit 14	Bit 10	Address (<net><host>)
Out	Out	<ones><ones>
Out	In	<zeros><zeros>
In	In	<net><zeros>
In	Out	<net><ones>

For additional information on setting the hardware jumpers on your router, see the hardware documentation that was supplied with you router.

Changing the Default IP Broadcast Address for All Interfaces to 0.0.0.0 on Routers with Nonvolatile Memory

Cisco IOS-based routers with NVRAM have software configuration registers that allow you to modify several behaviors of the router such as where it looks for images to load, what IP broadcast address it uses, and the console line speed. The factory default value for the configuration register is 0x2102 where *0X* indicates this a hexadecimal number. The **config-register** command is used to modify the settings of the software configuration registers.

Information on configuring other behaviors with the software configuration registers using the **config-register** command is available in the following documentation:

- "Loading and Managing System Images" chapter of the *Cisco IOS Configuration Fundamentals Configuration Guide*
- *Cisco IOS Configuration Fundamentals Command Reference*

Caution You need to be very careful when you change the software configuration registers on your router because if you inadvertently alter the console port line speed, you will not be able to configure the router with a terminal server on the console port unless you know the speed that you set for the console port, and you know how to change the line speed for your terminal application. If your router is configured for alternate access to the CLI such as using Telnet or a web browser, you can use this method to log in to the router and change the software configuration register back to 0x2102.

Perform this task to set the IP broadcast address on every interface to 0.0.0.0 while maintaining the remainder of the default values for the software configuration register settings.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **config-register** *value*
4. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`config-register`** *`value`*
	Example:
	`Router(config)# config-register 0x2502`
		Sets the IP broadcast address to 0.0.0.0 on every interface while maintaining the remainder of the default values for the other software configuration register settings.

Step 4	**`end`**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Changing the IP Broadcast Address to any IP Address on One or More Interfaces in a Router

Perform this task if you network requires an IP broadcast address other than 255.255.255.255 or 0.0.0.0, or you want to change the IP broadcast address to 0.0.0.0 on a subset of the interfaces on the router instead of on all of the interfaces on the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *address mask*
5. **ip broadcast-address** *address*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/1`	Specifies an interface and enters interface configuration mode.
Step 4	**ip address** *address mask* **Example:** `Router(config-if)# ip address 172.16.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 5	**ip broadcast-address** *address* **Example:** `Router(config-if)# ip broadcast-address 172.16.10.255`	Specifies the IP broadcast address In this example IP broadcasts are sent to 172.16.10.255.

Step 6	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Configuring UDP Broadcast Packet Flooding

You can allow IP broadcasts to be flooded throughout your network in a controlled fashion using the database created by the bridging Spanning Tree Protocol (STP). Enabling this feature also prevents loops. In order to support this capability, the routing software must include transparent bridging, and bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still will be able to receive broadcasts. However, the interface will never forward broadcasts it receives, and the router will never use that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

Prerequisites

The version of Cisco IOS software on your router must support transparent bridging.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **bridge** *number* **protocol ieee**
4. **ip forward-protocol spanning-tree**
5. **ip forward-protocol turbo-flood**
6. **ip forward-protocol udp**
7. **interface** *type number*
8. **ip address** *address mask*
9. **bridge-group** *number*
10. **interface** *type number*
11. **ip address** *address mask*
12. **bridge-group** *number*
13. **end**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**`enable`**
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	**`configure terminal`**
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	**`bridge`** *`number`* **`protocol ieee`**
	Example:
	`Router(config)# bridge 1 protocol ieee`
		Enables spanning-tree bridging and specifies the bridging protocol.

Step 4	**`ip forward-protocol spanning-tree`**
	Example:
	`Router(config)# ip forward-protocol spanning-tree`
		Enables using the spanning-tree forwarding table to flood broadcast packets.

Step 5	**`ip forward-protocol turbo-flood`**
	Example:
	`Router(config)# ip forward-protocol turbo-flood`
		(Optional) Enables fast forwarding of broadcast packets using the spanning-tree forwarding table.

Step 6	**`ip forward-protocol udp`**
	Example:
	`Router(config)# ip forward-protocol udp`
		Enables forwarding of UDP broadcasts.

	Command or Action	Purpose
Step 7	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 8	**ip address** *address mask* **Example:** `Router(config-if)# ip address 192.168.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 9	**bridge-group** *number* **Example:** `Router(config-if)# bridge-group 1`	Places the interface in the spanning-tree bridge group specified.
Step 10	**interface** *type number* **Example:** `Router(config-if)# interface fastethernet 0/1`	Specifies an interface and enters interface configuration mode.
Step 11	**ip address** *address mask* **Example:** `Router(config-if)# ip address 172.16.10.1 255.255.255.0`	Assigns an IP address to the interface.
Step 12	**bridge-group** *number* **Example:** `Router(config-if)# bridge-group 1`	Places the interface in the spanning-tree bridge group specified.

Step 13	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Configuration Examples for IP Broadcast Packet Handling

This section provides the following configuration examples:

- Enabling IP Directed Broadcasts With an Access List: Example, page 23
- Configuring UDP Broadcast Packet Flooding: Example, page 24

Enabling IP Directed Broadcasts With an Access List: Example

The following example shows how to configure IP directed broadcasts with an access list to control the directed broadcasts that are forwarded.

```
access-list 100 permit ip 10.4.9.167 0.0.0.0 172.16.10.0 0.0.0.255
interface fastethernet 0/0
```

```
 ip address 172.16.10.1 255.255.255.0
 ip directed-broadcast 100
```

Configuring UDP Broadcast Packet Flooding: Example

The following examples shows how to configure UDP broadcast packet flooding.

```
bridge 1 protocol ieee
ip forward-protocol spanning-tree
ip forward-protocol turbo-flood
ip forward-protocol udp
interface fastethernet 0/0
 ip address 192.168.10.1 255.255.255.0
 bridge-group 1
interface fastethernet 0/1
 ip address 172.16.10.1 255.255.255.0
 bridge-group 1
```

Additional References

The following sections provide references related to the Configuring IP broadcast packet handling.

Related Documents

Related Topic	Document Title
Currently assigned IP multicast addresses	*Internet Multicast Addresses* http://www.iana.org/assignments/multicast-addresses
Configuration fundamentals configuration tasks	*Cisco IOS Configuration Fundamentals Configuration Guide*
Configuration fundamentals configuration tasks	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS bridging and IBM networking configuration tasks	*Cisco IOS Bridging and IBM Networking Configuration Guide*
Cisco IOS bridging and IBM networking configuration tasks	*Cisco IOS Bridging and IBM Networking Command Reference*

Standards

Standard	Title
IEEE spanning-tree Bridging	802.1D MAC Bridges http://www.ieee802.org/1/pages/802.1D-2003.html

MIBs

MIB	MIBs Link
—	No new or modified MIBs are supported, and support for existing MIBs has not been modified.

RFCs

RFC	Title
RFC 1812	*Requirements for IP Version 4 Routers* http://www.ietf.org/rfc/rfc1812.txt
RFC 2131	*Dynamic Host Configuration Protocol* http://www.ietf.org/rfc/rfc2131.txt.

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for IP Broadcast Packet Handling

Table 3 lists the features in this module and provides links to specific configuration information.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 3 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 3 Feature Information for IP Broadcast Packet Handling

Feature Name	Releases	Feature Information
IP Directed Broadcasts	10.0	Enables the translation of a directed broadcast to physical broadcasts. The following section provides information about this feature: • Enabling IP Directed Broadcasts, page 12 The following command was introduced or modified by this feature: **ip directed-broadcast**.
UDP Broadcast Packet Forwarding	10.0	Enables the forwarding of UDP broadcast packets. The following section provides information about this feature: • Enabling Forwarding of UDP Broadcast Packets, page 14 The following commands were introduced or modified by this feature: **ip forward-protocol**, **ip helper-address**.

Table 3 Feature Information for IP Broadcast Packet Handling (continued)

Feature Name	Releases	Feature Information
Flooding Packets Using spanning-tree	10.0	Enables the forwarding of UDP broadcast packets using the spanning-tree forwarding table. The following section provides information about this feature: • Configuring UDP Broadcast Packet Flooding, page 21 The following commands were introduced or modified by this feature: **ip forward-protocol spanning-tree**, **ip forward-protocol turbo-flood**.
Specifying an IP Broadcast Address	10.0	Specifies the IP broadcast address for an interface. The following section provides information about this feature: • Enabling IP Directed Broadcasts, page 12 The following command was introduced or modified by this feature: **ip broadcast-address**.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Configuring UDP Forwarding Support for IP Redundancy Virtual Router Groups

First Published: February 10, 2008
Last Updated: February 10, 2008

User Datagram Protocol (UDP) forwarding is a feature used in Cisco IOS software to forward broadcast and multicast packets received for a specific IP address. Virtual Router Group (VRG) support is currently implemented with the Hot Standby Routing Protocol (HSRP) and it allows a set of routers to be grouped as a logical router that answers to a well-known IP address. The UDP Forwarding Support for IP Redundancy Virtual Router Groups feature enables UDP forwarding to be VRG aware, resulting in forwarding only to the active router in the VRG.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for UDP Forwarding Support for IP Redundancy Virtual Router Groups" section on page 6.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Information About UDP Forwarding Support for IP Redundancy Virtual Router Groups

Before you configure the UDP Forwarding Support of Virtual Router Group feature, you should understand the following concepts:

- Benefits of the UDP Forwarding Support for Virtual Router Groups Feature, page 2

Benefits of the UDP Forwarding Support for Virtual Router Groups Feature

Forwarding is limited to the active router in the VRG instead of all routers within the VRG. Prior to the implementation of this feature the only VRG support was HSRP. Within a VRG that is formed by HSRP, the forwarding of UDP-based broadcast and multicast packets is done by all the routers within the VRG. This process can cause some DHCP servers to operate incorrectly. By making the UDP forwarding code VRG aware, forwarding will be limited to the active router in the VRG.

VRG awareness is achieved with IP Redundancy Service (IRS). The IRS API provides for notification updates of a specific VRG, addition and deletion of a VRG, and querying of the current state of a VRG. State change notification is provided to avoid the performance impact of querying the state of the VRG each time it is needed. The UDP forwarding code caches the VRG state for each required helper address defined. Each time the UDP forwarding code needs to execute, it checks the current state of the VRG associated with the helper address and forwards only for VRGs that are active.

Note The UDP Forwarding Support for virtual Router Groups feature is available only on platforms that support VRGs.

How to Configure UDP Forwarding Support for IP Redundancy Virtual Router Groups

This section contains the following procedure:

- Configuring UDP Forwarding Support for IP Redundancy Virtual Router Groups, page 2

Configuring UDP Forwarding Support for IP Redundancy Virtual Router Groups

Perform this task to configure UDP Forwarding Support for IP Redundancy Virtual Router Groups.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip helper-address** *address* **redundancy** *vrg-name*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface fastethernet 0/0`	Specifies an interface and enters interface configuration mode.
Step 4	**ip helper-address** *address* **redundancy** *vrg-name* **Example:** `Router(config-if)# ip helper-address 10.1.1.1 redundancy shop`	Enables UDP forwarding support for the VRG.

Step 5	**end**
	Example:
	`Router(config-if)# end`
		Exits the current configuration mode and returns to privileged EXEC mode.

Configuration Examples for UDP Forwarding Support for IP Redundancy Virtual Router Groups

This section provides the following configuration example:

- Configuring UDP Forwarding Support for IP Redundancy Virtual Router Groups: Example, page 3

Configuring UDP Forwarding Support for IP Redundancy Virtual Router Groups: Example

The following example shows how to configure UDP Forwarding Support for IP Redundancy Virtual Router Groups:

```
interface fastethernet 0/0
 no shutdown
 ip address 172.16.10.1 255.255.255.0
 ip helper-address 10.1.1.1 redundancy shop
```

Additional References

The following sections provide references related to the UDP Forwarding Support for IP Redundancy Virtual Router Groups feature.

Related Documents

Related Topic	Document Title
IP application services commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Application Services Command Reference*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified	—

MIBs

MIB	MIBs Link
No new or modified MIBs are supported, and support for existing MIBs has not been modified	—

RFCs

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for UDP Forwarding Support for IP Redundancy Virtual Router Groups

Table 1 lists the features in this module. For information on a feature in this technology that is not documented here, see the other available documentation for your Cisco IOS release.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for UDP Forwarding Support for IP Redundancy Virtual Router Groups

Feature Name	Releases	Feature Information
UDP Forwarding Support for IP Redundancy Virtual Router Group	12.2(15)	User Datagram Protocol (UDP) forwarding is a feature used in Cisco IOS software to forward broadcast and multicast packets received for a specific IP address. Virtual Router Group (VRG) support is currently implemented with the Hot Standby Routing Protocol (HSRP) and it allows a set of routers to be grouped as a logical router that answers to a well known well-known IP address. The UDP Forwarding Support for IP Redundancy Virtual Router Groups feature enables UDP forwarding to be VRG aware, resulting in forwarding only to the active router in the VRG. The following command was introduced or modified: **ip helper-address**.

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.