IOS Server Load Balancing

Feature History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This feature was introduced with support for the following platforms:</td>
</tr>
<tr>
<td></td>
<td>• Multilayer Switch Feature Card (MSFC) and Supervisor Engine 1 for Cisco</td>
</tr>
<tr>
<td></td>
<td>Catalyst 6500 family switches (including the Catalyst 6506, Catalyst 6509, and</td>
</tr>
<tr>
<td></td>
<td>Catalyst 6513)</td>
</tr>
<tr>
<td></td>
<td>• Cisco 7200 series routers</td>
</tr>
<tr>
<td></td>
<td>The following functions were provided:</td>
</tr>
<tr>
<td></td>
<td>• Algorithms for Server Load Balancing, page 6</td>
</tr>
<tr>
<td></td>
<td>• Automatic Server Failure Detection, page 8</td>
</tr>
<tr>
<td></td>
<td>• Automatic Unfail, page 8</td>
</tr>
<tr>
<td></td>
<td>• Bind ID Support, page 9</td>
</tr>
<tr>
<td></td>
<td>• Client-Assigned Load Balancing, page 9</td>
</tr>
<tr>
<td></td>
<td>• Delayed Removal of TCP Connection Context, page 9</td>
</tr>
<tr>
<td></td>
<td>• Dynamic Feedback Protocol for IOS SLB, page 10</td>
</tr>
<tr>
<td></td>
<td>• Maximum Connections, page 14</td>
</tr>
<tr>
<td></td>
<td>• Port-Bound Servers, page 18</td>
</tr>
<tr>
<td></td>
<td>• Slow Start, page 22</td>
</tr>
<tr>
<td></td>
<td>• Sticky Connections, page 22</td>
</tr>
<tr>
<td></td>
<td>• SynGuard, page 23</td>
</tr>
<tr>
<td></td>
<td>• TCP Session Reassignment, page 23</td>
</tr>
<tr>
<td>12.1(1)E</td>
<td>The following functions were added:</td>
</tr>
<tr>
<td></td>
<td>• Alternate IP Addresses, page 7</td>
</tr>
<tr>
<td></td>
<td>• Network Address Translation (NAT), page 14—Server NAT</td>
</tr>
<tr>
<td></td>
<td>• Redundancy Enhancements, page 21—Stateless Backup</td>
</tr>
<tr>
<td></td>
<td>• Transparent Web Cache Load Balancing, page 24</td>
</tr>
</tbody>
</table>
12.1(2)E The following functions were added:

- Audio and Video Load Balancing, page 8
- Content Flow Monitor Support, page 9
- Network Address Translation (NAT), page 14—Server and Client NAT
- Probes, page 18—HTTP Probes
- Redundancy Enhancements, page 21—Stateless and Stateful Backup

12.1(3a)E The following functions were added:

- Firewall Load Balancing, page 10
- Probes, page 18—HTTP and Ping Probes
- Protocol Support, page 19
- Redundancy Enhancements, page 21—Stateless and Stateful Backup, and Active Standby
- WAP Load Balancing, page 24

12.1(5a)E The following functions were added:

- Avoiding Attacks on Server Farms and Firewall Farms, page 8
- Probes, page 18—HTTP, Ping, and WSP Probes

12.1(5)T The Cisco IOS Release 12.1(1)E feature was integrated into Cisco IOS Release 12.1(5)T, supporting Cisco 7200 series routers only.

12.2 The Cisco IOS Release 12.1(5)T feature was integrated into Cisco IOS Release 12.2.

12.1(7)E Support for the following platform was added:

- Cisco 7100 series routers

The following functions were added:

- Multiple Firewall Farm Support, page 14
- Route Health Injection, page 22

12.1(8a)E Support for the following platform was added:

- MSFC2 and Supervisor Engine 2 for Cisco Catalyst 6500 family switches (including the Catalyst 6506, Catalyst 6509, and Catalyst 6513)

The following functions were added:

- Backup Server Farms, page 8
- DFP Agent Subsystem Support, page 9

12.1(9)E The following functions were added:

- GPRS Load Balancing, page 11—Support for GPRS Tunneling Protocol (GTP) v0
This document describes the Cisco IOS Server Load Balancing (IOS SLB) feature in Cisco IOS Release 12.2(14)ZA2. It includes the following sections:

- **Overview of the IOS SLB Feature**, page 4
- **Functions and Capabilities**, page 5
- **Benefits**, page 25
- **Restrictions**, page 26

### 12.1(11b)E
Support for the following platforms was added:
- MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for the Cisco 7600 Internet routers (including the Cisco 7603, Cisco 7606, and Cisco 7609)

The following functions were added:
- **Network Address Translation (NAT)**, page 14—Static NAT and Per-Packet Server Load Balancing
- **Probes**, page 18—DNS, HTTP, Ping, TCP, and WSP Probes
- **RADIUS Load Balancing**, page 20—General packet radio service (GPRS)
- **VPN Server Load Balancing**, page 24

### 12.1(12c)E
The following functions were added:
- **Probes**, page 18—Routed Probes
- **RADIUS Load Balancing**, page 20—CDMA2000 and Multiple Service Gateway Server Farms

### 12.1(13)E
This release incorporated only minor corrections and clarifications.

### 12.2(14)S
This feature was integrated into Cisco IOS Release 12.2(14)S.

Support for the following platforms was removed:
- Cisco 7100 series routers
- MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for Cisco Catalyst 6500 family switches (including the Catalyst 6506, Catalyst 6509, and Catalyst 6513)
- MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for the Cisco 7600 Internet routers (including the Cisco 7603, Cisco 7606, and Cisco 7609)

### 12.1(13)E3
Support for the following platforms was added:
- Cisco 7100 series routers
- MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for Cisco Catalyst 6500 family switches (including the Catalyst 6506, Catalyst 6509, and Catalyst 6513)
- MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for the Cisco 7600 Internet routers (including the Cisco 7603, Cisco 7606, and Cisco 7609)

The following functions were added:
- **GPRS Load Balancing**, page 11—Support for GTP v0 and GTP v1
- **GPRS Load Balancing with GTP Cause Code Inspection**, page 12
- **Probes**, page 18—DNS, HTTP, Ping, TCP, Custom UDP, and WSP Probes

### 12.2(14)ZA2
The following function was added:
- **Home Agent Director**, page 13
Overview of the IOS SLB Feature

The IOS SLB feature is an IOS-based solution that provides IP server load balancing. Using the IOS SLB feature, you can define 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 connect to the IP address of the virtual server. 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.

Note

IOS SLB 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.

IOS SLB also provides firewall load balancing, which balances flows across a group of firewalls called a firewall farm. Figure 1 illustrates a logical view of a simple IOS SLB network.
Functions and Capabilities

This section describes the following functions and capabilities provided by IOS SLB.

Note

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

- Algorithms for Server Load Balancing, page 6
- Alternate IP Addresses, page 7
- Audio and Video Load Balancing, page 8
- Automatic Server Failure Detection, page 8
- Automatic Unfail, page 8
- Avoiding Attacks on Server Farms and Firewall Farms, page 8
- Backup Server Farms, page 8
- Bind ID Support, page 9
- Client-Assigned Load Balancing, page 9
- Content Flow Monitor Support, page 9
- Delayed Removal of TCP Connection Context, page 9
- DFP Agent Subsystem Support, page 9
- Dynamic Feedback Protocol for IOS SLB, page 10
- Firewall Load Balancing, page 10
- GPRS Load Balancing, page 11
• GPRS Load Balancing with GTP Cause Code Inspection, page 12
• Home Agent Director, page 13
• Maximum Connections, page 14
• Multiple Firewall Farm Support, page 14
• Network Address Translation (NAT), page 14
• Port-Bound Servers, page 18
• Probes, page 18
• Protocol Support, page 19
• RADIUS Load Balancing, page 20
• Redundancy Enhancements, page 21
• Route Health Injection, page 22
• Slow Start, page 22
• Sticky Connections, page 22
• SynGuard, page 23
• TCP Session Reassignment, page 23
• Transparent Web Cache Load Balancing, page 24
• VPN Server Load Balancing, page 24
• WAP Load Balancing, page 24

**Algorithms for Server Load Balancing**

IOS SLB provides the following load-balancing algorithms:

• Weighted Round Robin, page 6
• Weighted Least Connections, page 7

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.

**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.
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.

### 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.

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.

The Home Agent Director does not support the weighted least connections algorithm.

### 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 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.
Audio and Video Load Balancing

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

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.

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.

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.

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.

**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.

**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.

**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(14)ZA2.
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.

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:

\[
\frac{\text{existing PDP contexts}}{\text{maximum PDP contexts}}
\]

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:

\[
\frac{\text{existing PDP contexts}}{10000} = \text{Low GGSN weight}
\]

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.

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.
- 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.

**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.

IOS SLB supports two types of GPRS load balancing:

- GPRS Load Balancing without GTP Cause Code Inspection, page 12
- GPRS Load Balancing with GTP Cause Code Inspection, page 12

**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.

- Does not support stateful backup. See the “Stateful Backup” section on page 22 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 6 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>
<thead>
<tr>
<th>Cause Code</th>
<th>IOS SLB Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Accepted</td>
<td>Establish session</td>
</tr>
<tr>
<td>No Resource Available</td>
<td>Fail current real, reassign session, drop the response</td>
</tr>
<tr>
<td>All dynamic addresses are occupied</td>
<td>Fail current real, reassign session, drop the response</td>
</tr>
<tr>
<td>No memory is available</td>
<td>Fail current real, reassign session, drop the response</td>
</tr>
<tr>
<td>System Failure</td>
<td>Fail current real, reassign session, drop the response</td>
</tr>
<tr>
<td>Missing or Unknown APN</td>
<td>Forward the response</td>
</tr>
</tbody>
</table>
Functions and Capabilities

Cisco IOS Release 12.2(14)ZA2

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 22 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 7 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.

<table>
<thead>
<tr>
<th>Cause Code</th>
<th>IOS SLB Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown PDP Address or PDP type</td>
<td>Forward the response</td>
</tr>
<tr>
<td>User Authentication Failed</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Semantic error in TFT operation</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Syntactic error in TFT operation</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Semantic error in packet filter</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Syntactic error in packet filter</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Mandatory IE incorrect</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Mandatory IE missing</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Optional IE incorrect</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Invalid message format</td>
<td>Forward the response</td>
</tr>
<tr>
<td>Version not supported</td>
<td>Forward the response</td>
</tr>
</tbody>
</table>

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

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.
- Can operate in both fast and CEF switching modes.
- Does not support stateful backup. See the “Stateful Backup” section on page 22 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 6 for more information about this algorithm.

For more information about Mobile IP, home agents, and related topics, refer to the Cisco IOS IP Configuration Guide, Release 12.2.
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” section on page 14
- “Dispatched Mode” section on page 15
- “Directed Mode” section on page 15
- “Server NAT” section on page 15
- “Client NAT” section on page 16
- “Static NAT” section on page 16
- “Server Port Translation” section on page 17

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 Logical Interfaces” chapter of the Cisco IOS Interface Configuration Guide, Release 12.2 for more information about configuring the loopback address.

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 15
- “Client NAT” section on page 16
- “Static NAT” section on page 16
- “Server Port Translation” section on page 17

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:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
</table>
| 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 Hypertext Transfer Protocol (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.

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 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.
Functions and Capabilities

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 Configuration Fundamentals Command Reference, Release 12.2 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 in the “Command Reference” section on page 116 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, Release 12.2 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**

IOS SLB supports the following protocols:

- 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)
- IP in IP Encapsulation (IPinIP)
- Internet Key Exchange (IKE, was ISAKMP)
- 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

**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.

• Can route data packets to a real server in the CSG farm, then to a real server in the SSG farm.
Functions and Capabilities

Cisco IOS Release 12.2(14)ZA2

- 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 6 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.
- The IOS SLB RADIUS username sticky database associates each subscriber’s username with a specific service gateway. In a CDMA2000 mobile wireless network, IOS SLB requires both the RADIUS framed-IP sticky database and the RADIUS username 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.

Redundancy Enhancements

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 options, based on HSRP:
- Stateless Backup, page 21
- Stateful Backup, page 22
- Active Standby, page 22

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.

GPRS load balancing without GTP cause code inspection enabled does not support stateful backup.

The Home Agent Director does 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 does not support active standby. That is, you cannot 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.

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.

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.

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.
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.

GPRS load balancing and the Home Agent Director do not support sticky connections.

**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.

**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.

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.

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 WSP 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.
Benefits

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 addresses 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.
Restrictions

IOS SLB has the following 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.

- 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.

- **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 (VIP) that is the same as one of the IOS interface 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).

- 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.

- **For 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.

- **For 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.
For firewall load balancing:
- Is no longer limited to a single firewall farm in each load-balancing device.
- Is limited to a single active firewall load-balancing device on each side of the firewall farm.
  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:
  - Active standby
  - Network Address Translation (NAT)
  - Port-bound servers
  - SynGuard
  - TCP session reassignment
  - Transparent web cache load balancing

For 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.
- 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
  - Stateful backup (unless GTP cause code inspection is enabled)
  - Sticky connections
  - Weighted least connections load-balancing algorithm

For 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
- Sticky connections

- For VPN server load balancing:
  - Does not support Internet Control Message Protocol (ICMP) and wildcard (0-protocol) virtual servers.

- For RADIUS load balancing for GPRS:
  - RADIUS load balancing requires the weighted round robin algorithm.
  - RADIUS load balancing 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).

- For RADIUS load balancing for CDMA2000:
  - RADIUS load balancing requires the weighted round robin algorithm.
  - RADIUS load balancing 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.

- For the 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

- For 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.

- For 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 Catalyst 6500 Family Switches and Cisco 7600 Internet 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 either an MSFC1 or an MSFC2, except when specifically differentiated.

  The term “PFC” refers to either a PFC1 or a PFC2, except when specifically differentiated.

  - Requires that the Multilayer Switching (MLS) flow mode be set to full. 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 the Cisco 7100 Series and Cisco 7200 Series:
  – Provides no hardware acceleration for the IOS SLB function for either dispatched mode or directed mode.
  – Supports Cisco IOS NAT in directed mode with no hardware data packet acceleration.

Related Features and Technologies

• Content Flow Monitor (CFM)
• Dynamic Feedback Protocol (DFP)
• General packet radio service (GPRS)
• Hot Standby Router Protocol (HSRP)
• Mobile IP
• Network Address Translation (NAT)
• Wireless Application Protocol (WAP)

Related Documents

• Cisco IOS IP Configuration Guide, Release 12.2
• Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2
• Cisco IOS Mobile Wireless Configuration Guide, Release 12.2
• Cisco IOS Mobile Wireless Command Reference, Release 12.2
• Dynamic Feedback Protocol Support in Distributed Director
• Using Content Flow Monitor

Supported Platforms

• Cisco 7100 series routers
• Cisco 7200 series routers
• MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for Cisco Catalyst 6500 family switches (including the Catalyst 6506, Catalyst 6509, and Catalyst 6513)
• MSFC2, Supervisor Engine 1, and Supervisor Engine 2 for the Cisco 7600 Internet routers (including the Cisco 7603, Cisco 7606, and Cisco 7609)

Determining Platform Support Through Cisco Feature Navigator
Cisco IOS software is packaged in feature sets that are supported on specific platforms. To get updated information regarding platform support for this feature, access Cisco Feature Navigator. Cisco Feature Navigator dynamically updates the list of supported platforms as new platform support is added for the feature.
Cisco Feature Navigator is a web-based tool that enables you to determine which Cisco IOS software images support a specific set of features and which features are supported in a specific Cisco IOS image. You can search by feature or release. Under the release section, you can compare releases side by side to display both the features unique to each software release and the features in common.

To access Cisco Feature Navigator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:

http://www.cisco.com/register

Cisco Feature Navigator is updated regularly when major Cisco IOS software releases and technology releases occur. For the most current information, go to the Cisco Feature Navigator home page at the following URL:

http://www.cisco.com/go/fn

**Availability of Cisco IOS Software Images**

Platform support for particular Cisco IOS software releases is dependent on the availability of the software images for those platforms. Software images for some platforms may be deferred, delayed, or changed without prior notice. For updated information about platform support and availability of software images for each Cisco IOS software release, refer to the online release notes or, if supported, Cisco Feature Navigator.

---

**Supported Standards, MIBs, and RFCs**

**Standards**

- No new or modified standards

**MIBs**

- CISCO-SLB-MIB

---

**Note** Although the objects in this MIB 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://tools.cisco.com/ITDIT/MIBS/servlet/index

If Cisco MIB Locator does not support the MIB information that you need, you can also obtain a list of supported MIBs and download MIBs from the Cisco MIBs page at the following URL:


To access Cisco MIB Locator, you must have an account on Cisco.com. If you have forgotten or lost your account information, send a blank e-mail to cco-locksmith@cisco.com. An automatic check will verify that your e-mail address is registered with Cisco.com. If the check is successful, account details with a new random password will be e-mailed to you. Qualified users can establish an account on Cisco.com by following the directions found at this URL:
Cisco IOS NAT, RFC 1631

Configuration Tasks

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” section on page 59.

For a complete description of the IOS SLB commands in this section, see the “Command Reference” section on page 116. 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 33 (Required)
- Configuring Firewall Load Balancing, page 40 (Optional)
- Configuring Probes, page 44 (Optional)
- Configuring DFP, page 48 (Optional)
- GPRS Load Balancing Configuration Task List, page 48 (Optional)
- RADIUS Load Balancing Configuration Task List, page 50 (Optional)
- VPN Server Load Balancing Configuration Task List, page 52 (Optional)
- Home Agent Director Configuration Task List, page 53 (Optional)
- Configuring NAT, page 53 (Optional)
- Configuring Static NAT, page 54 (Optional)
- Stateless Backup Configuration Task List, page 54 (Optional)
- Configuring Database Entries, page 56 (Optional)
- Configuring Buffers for the Fragment Database, page 56 (Optional)
- Clearing Databases and Counters, page 56 (Optional)
- Configuring Wildcard Searches, page 57 (Optional)
- Purging and Reassigning Connections, page 57 (Optional)
- Monitoring and Maintaining the IOS SLB Feature, page 58

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 Real Server, page 34 (Required)
- Configuring a Virtual Server, page 37 (Required)
- Verifying the Server Farm, page 39 (Optional)
- Verifying the Virtual Server, page 39 (Optional)
- Verifying the Clients, page 39 (Optional)
- Verifying IOS SLB Connectivity, page 40 (Optional)

## Configuring a Server Farm and Real Server

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

To configure an IOS SLB server farm, use the following commands beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> Router(config)# ip slb serverfarm server-farm Router(config-slb-sfarm)#</td>
<td>Adds a server farm definition to the IOS Server Load Balancing (IOS SLB) configuration and enters server farm configuration mode.</td>
</tr>
<tr>
<td><strong>Step 2</strong> Router(config-slb-sfarm)# bindid [bind-id]</td>
<td>(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.</td>
</tr>
<tr>
<td><strong>Step 3</strong> Router(config-slb-sfarm)# nat {client pool</td>
<td>server}</td>
</tr>
<tr>
<td><strong>Step 4</strong> Router(config-slb-sfarm)# predictor [roundrobin</td>
<td>leastconns]</td>
</tr>
<tr>
<td><strong>Step 5</strong> Router(config-slb-sfarm)# probe probe</td>
<td>(Optional) Associates a probe with the real server.</td>
</tr>
</tbody>
</table>
### Configuration Tasks

#### Step 6
```
Router(config-slb-sfarm)# real ip-address [port]
```
**Purpose**
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.

#### Step 7
```
Router(config-slb-real)# faildetect numconns number-of-conns [numclients number-of-clients]
```
**(Optional)** Specifies the number of consecutive connection failures and, optionally, the number of unique client connection failures, that constitute failure of the real server.

**Note**
- 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 8
```
Router(config-slb-real)# maxclients number-of-conns
```
**(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 9
```
Router(config-slb-real)# maxconns number-of-conns
```
**(Optional)** Specifies the maximum number of active connections allowed on the real server at one time.

#### Step 10
```
Router(config-slb-real)# reassign threshold
```
**(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 11
```
Router(config-slb-real)# retry retry-value
```
**(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.
## Configuration Tasks

### Step 12

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-slb-real)# weight setting</code></td>
<td>(Optional) Specifies the real server's workload capacity relative to other servers in the server farm.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>If you use Dynamic Feedback Protocol (DFP), the static weights you define using the <code>weight</code> 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.</td>
</tr>
</tbody>
</table>

### Step 13

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Router(config-slb-real)# inservice</code></td>
<td>Enables the real server for use by IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>

---

**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 Wildcard Searches” section on page 57 for more details.
## Configuring a Virtual Server

IOS SLB supports up to 500 virtual servers.  

To configure an IOS SLB virtual server, use the following commands beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1**  
Router(config)# ip slb vserver virtual-server | Identifies a virtual server and enters virtual server configuration mode. |
| **Step 2**  
Router(config-slb-vserver)# virtual ip-address [netmask [group]] {esp | gre | protocol}  
or  
Router(config-slb-vserver)# virtual ip-address [netmask [group]] {tcp | udp} [port | any] [service service] | 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. |
| **Step 3**  
Router(config-slb-vserver)# serverfarm primary-farm [backup backup-farm [sticky]] | 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**  
GPRS load balancing and the Home Agent Director do not support the `sticky` attribute.  

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.

- **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.
### Configuration Tasks

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td><code>Router(config-slb-vserver)# access interface route framed-ip</code></td>
<td>(Optional) Enables framed-IP routing to inspect the ingress interface.</td>
</tr>
<tr>
<td>5</td>
<td><code>Router(config-slb-vserver)# advertise</code></td>
<td>(Optional) Controls the installation of a static route to the Null0 interface for a virtual server address.</td>
</tr>
</tbody>
</table>
| 6    | `Router(config-slb-vserver)# client {ip-address netmask [exclude] | gtp carrier-code [code]}` | (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. |
| 7    | `Router(config-slb-vserver)# delay duration` | (Optional) Specifies the time IOS Server Load Balancing (IOS SLB) maintains TCP connection context after a connection has terminated. |
| 8    | `Router(config-slb-vserver)# hand-off radius duration` | (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. |
| 9    | `Router(config-slb-vserver)# idle {gtp request | ipmobile request | radius (request | framed-ip)) duration` | (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. |
| 10   | `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. |
| 11   | `Router(config-slb-vserver)# replicate casa listen-ip remote-ip port [interval] [password [0 | 7] password timeout]` | (Optional) Configures a stateful backup of IOS Server Load Balancing (IOS SLB) decision tables to a backup switch.  
*Note* GPRS load balancing *without* GTP cause code inspection enabled does not support this command.  
The Home Agent Director does not support this command. |
| 12   | `Router(config-slb-vserver)# sticky {duration [group group-id] | netmask netmask] | radius framed-ip [group group-id] | radius username [msid-cisco] [group group-id]}` | (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. |
Verifying the Virtual Server

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

```
Router# show ip slb vserver
```

```
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
```

Verifying the Server Farm

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 reals
```

```
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
```

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
```

Verifying the Clients

The following `show ip slb conns` command verifies the restricted client access and status:
Configuration Tasks

Router# show ip slb conns

<table>
<thead>
<tr>
<th>vserver</th>
<th>prot</th>
<th>client</th>
<th>real</th>
<th>state</th>
<th>nat</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESTRICTED_HTTP</td>
<td>TCP</td>
<td>10.4.4.0:80</td>
<td>10.1.1.20</td>
<td>CLOSING</td>
<td>none</td>
</tr>
</tbody>
</table>

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
VTEST_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

Verifying IOS SLB Connectivity

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 the IOS SLB Feature” section on page 58 for additional commands used to verify IOS SLB networks and connections.

Configuring Firewall Load Balancing

This section describes the tasks required 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 Ping Probes” section on page 46 for more details. If a firewall does not allow ping probes to be forwarded, use HTTP probes instead. See the “Configuring HTTP Probes” section on page 45 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 Wildcard Searches” section on page 57 for more details.
This section describes the following IOS SLB firewall load-balancing configuration tasks. Required and optional tasks are indicated.

- **Configuring the Firewall Farm, page 41 (Required)**
- **Verifying the Firewall Farm, page 42 (Optional)**
- **Verifying Firewall Connectivity, page 43 (Optional)**

## Configuring the Firewall Farm

To configure an IOS SLB firewall load-balancing network, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# ip slb firewallfarm firewall-farm Router(config-slb-fw)#</td>
<td>Adds a firewall farm definition to the IOS Server Load Balancing (IOS SLB) configuration and enters firewall farm configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-fw)# real ip-address</td>
<td>Identifies a firewall by IP address as a member of a firewall farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config-slb-fw-real)# probe probe</td>
<td>Associates a probe with the firewall.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Router(config-slb-fw-real)# weight setting</td>
<td>(Optional) Specifies the firewall’s workload capacity relative to other firewalls in the firewall farm.</td>
</tr>
<tr>
<td>Step 5</td>
<td>Router(config-slb-fw-real)# inservice</td>
<td>Enables the firewall for use by the firewall farm and by IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>Step 6</td>
<td>Router(config-slb-fw)# access [source source-ip netmask] [destination destination-ip netmask]</td>
<td>(Optional) Routes specific flows to a firewall farm.</td>
</tr>
<tr>
<td>Step 7</td>
<td>Router(config-slb-fw)# predictor hash address [port]</td>
<td>(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.</td>
</tr>
<tr>
<td>Step 8</td>
<td>Router(config-slb-fw)# replicate casa listen-ip remote-ip port [interval] [password [0</td>
<td>7] password [timeout]]</td>
</tr>
<tr>
<td>Step 9</td>
<td>Router(config-slb-fw)# protocol tcp</td>
<td>(Optional) Enters firewall farm TCP protocol configuration mode.</td>
</tr>
<tr>
<td>Step 10</td>
<td>Router(config-slb-fw-tcp)# delay duration</td>
<td>(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.</td>
</tr>
<tr>
<td>Step 11</td>
<td>Router(config-slb-fw-tcp)# idle duration</td>
<td>(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.</td>
</tr>
</tbody>
</table>
**Configuration Tasks**

**Verifying the Firewall Farm**

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
```

**Command Purpose**

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 12</strong> Router(config-slb-fw-tcp)# <code>maxconns number-of-conns</code></td>
<td>(Optional) For firewall farm TCP protocol configuration mode, specifies the maximum number of active TCP connections allowed on the firewall farm at one time.</td>
</tr>
</tbody>
</table>
| **Step 13** Router(config-slb-fw-tcp)# `sticky duration [netmask netmask] [source | destination]` | (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 14** Router(config-slb-fw)# `protocol datagram`                  | (Optional) Enters firewall farm datagram protocol configuration mode.                     |
| **Step 15** Router(config-slb-fw-udp)# `idle duration`                  | (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 16** Router(config-slb-fw-udp)# `maxconns number-of-conns`      | (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 17** Router(config-slb-fw-udp)# `sticky duration [netmask netmask] [source | destination]` | (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 18** Router(config-slb-fw)# `inservice`                         | Enables the firewall farm for use by IOS Server Load Balancing (IOS SLB).                  |
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**

To verify that IOS SLB firewall load balancing has been configured and is operating correctly:

---

**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
```
Configuration Tasks

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

\texttt{Router\# show ip slb conns}

\begin{tabular}{|c|c|c|c|c|}
\hline
\textbf{vserver} & \textbf{prot} & \textbf{client} & \textbf{real} & \textbf{state} & \textbf{nat} \\
\hline
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 \\
\hline
\end{tabular}

Step 6  See the “Monitoring and Maintaining the IOS SLB Feature” section on page 58 for additional commands used to verify IOS SLB networks and connections.

Configuring Probes

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 18.

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 Custom UDP Probes, page 44 (Required)
- Configuring DNS Probes, page 45 (Required)
- Configuring HTTP Probes, page 45 (Required)
- Configuring Ping Probes, page 46 (Required)
- Configuring TCP Probes, page 47 (Required)
- Configuring WSP Probes, page 47 (Required)
- Associating the Probe, page 47 (Required)
- Verifying the Probe, page 48 (Optional)

Configuring Custom UDP Probes

To configure a custom UDP probe, enter the following commands in order, beginning in global configuration mode:

\texttt{Command} | \texttt{Description} \\
\hline
\texttt{Step 1} Router(config)# ip slb probe probe custom udp | Configures the IOS Server Load Balancing (IOS SLB) probe name and enters custom User Datagram Protocol (UDP) probe configuration mode. \\
\texttt{Step 2} Router(config-slb-probe)# address [ip-address [routed]] | (Optional) Configures an IP address to which to send the custom User Datagram Protocol (UDP) probe. \\
\texttt{Step 3} Router(config-slb-probe)# interval seconds | (Optional) Configures the custom User Datagram Protocol (UDP) probe transmit timers.
# Configuration Tasks

## Configuring DNS Probes

To configure a DNS probe, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# \texttt{ip slb probe probe dns} Configures the IOS Server Load Balancing (IOS SLB) probe name and enters Domain Name System (DNS) probe configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-probe)# \texttt{address [ip-address [routed]]} (Optional) Configures an IP address to which to send the Domain Name System (DNS) probe.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config-slb-probe)# \texttt{faildetect number-of-probes} (Optional) Specifies the number of consecutive unacknowledged Domain Name System (DNS) probes that constitute failure of the real server or firewall.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Router(config-slb-probe)# \texttt{interval seconds} (Optional) Configures the Domain Name System (DNS) probe transmit timers.</td>
</tr>
<tr>
<td>Step 5</td>
<td>Router(config-slb-probe)# \texttt{lookup [ip-address]} (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.</td>
</tr>
</tbody>
</table>

## Configuring HTTP Probes

To configure an HTTP probe, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# \texttt{ip slb probe probe http} Configures the IOS Server Load Balancing (IOS SLB) probe name and enters HTTP probe configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-probe)# \texttt{address [ip-address [routed]]} (Optional) Configures an IP address to which to send the HTTP probe.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config-slb-probe)# \texttt{credentials [username [password]]} (Optional) Configures header values for the HTTP probe.</td>
</tr>
</tbody>
</table>
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 Ping Probes

To configure a ping probe, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# <code>ip slb probe probe ping</code></td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-probe)# <code>address [ip-address [routed]]</code></td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config-slb-probe)# <code>faildetect number-of-pings</code></td>
</tr>
<tr>
<td>Step 4</td>
<td>Router(config-slb-probe)# <code>interval seconds</code></td>
</tr>
</tbody>
</table>
Configuring TCP Probes

To configure a TCP probe, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# ip slb probe probe tcp</td>
</tr>
<tr>
<td></td>
<td>Configures the IOS Server Load Balancing (IOS SLB) probe name and enters TCP probe configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-probe)# address [ip-address [routed]]</td>
</tr>
<tr>
<td></td>
<td>(Optional) Configures an IP address to which to send the TCP probe.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config-slb-probe)# interval seconds</td>
</tr>
<tr>
<td></td>
<td>(Optional) Configures the TCP probe transmit timers.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Router(config-slb-probe)# port port</td>
</tr>
<tr>
<td></td>
<td>Configures the port to which the TCP probe is to connect.</td>
</tr>
</tbody>
</table>

Configuring WSP Probes

To configure a WSP probe, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# ip slb probe probe wsp</td>
</tr>
<tr>
<td></td>
<td>Configures the IOS Server Load Balancing (IOS SLB) probe name and enters Wireless Session Protocol (WSP) probe configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-probe)# address [ip-address [routed]]</td>
</tr>
<tr>
<td></td>
<td>(Optional) Configures an IP address to which to send the Wireless Session Protocol (WSP) probe.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Router(config-slb-probe)# interval seconds</td>
</tr>
<tr>
<td></td>
<td>(Optional) Configures the Wireless Session Protocol (WSP) probe transmit timers.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Router(config-slb-probe)# url [path]</td>
</tr>
<tr>
<td></td>
<td>(Optional) Configures the Wireless Session Protocol (WSP) probe URL path.</td>
</tr>
</tbody>
</table>

Associating the Probe

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 Real Server” section on page 34 and the “Configuring Firewall Load Balancing” section on page 40 for more details.

Note

You cannot associate a WSP probe with a firewall.
Verifying the Probe

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
Router#
```

Configuring DFP

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.

To configure IOS SLB as a DFP manager, and to identify a DFP agent with which IOS SLB can initiate connections, enter the following commands in order, beginning in global configuration mode:

```
Router(config)# ip slb dfp [password [0 | 7] password [timeout]]
Step 1
Router(config-slb-dfp)# agent ip-address port [timeout [retry-count [retry-interval]]]
Step 2
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Configures Dynamic Feedback Protocol (DFP), supplies an optional password, and enters DFP configuration mode.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Identifies a Dynamic Feedback Protocol (DFP) agent to which IOS Server Load Balancing (IOS SLB) can connect.</td>
</tr>
</tbody>
</table>

To configure IOS SLB as a DFP agent, refer to the `DFP Agent Subsystem` feature document for Cisco IOS Release 12.2(14)ZA2.

GPRS Load Balancing Configuration Task List

This section lists the tasks used to configure GPRS load balancing. Detailed configuration information is contained in the referenced sections of this or other documents. Required and optional tasks are indicated.

- **Configuring a Server Farm and Real Server, page 34** (Required)

  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.

**Configuring a Virtual Server, page 37 (Required)**

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.

**Configuring the virtual IP address as a loopback on each of the GGSNs in the server (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 a Loopback Interface” section in the *Cisco IOS Interface Configuration Guide*, Release 12.2 for more information.

**Routing each GGSN to each associated SGSN (Required)**

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*, Release 12.2 for more details.

**Routing 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.

**Configuring a GSN Idle Timer, page 50 (Optional)**

This step is applicable only if GTP cause code inspection is enabled.
Configuring a GSN Idle Timer

To configure a GPRS support node (GSN) idle timer, enter the following command in global configuration mode:

```
Router(config)# ip slb timers gtp gsn duration
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# ip slb timers gtp gsn duration</td>
<td>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).</td>
</tr>
</tbody>
</table>

RADIUS Load Balancing Configuration Task List

This section lists the tasks used to configure RADIUS load balancing. Detailed configuration information is contained in the referenced sections of this or other documents. Required and optional tasks are indicated.

- **Configuring a Server Farm and Real Server, page 34 (Required)**
  
  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` 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.

- **Configuring a Virtual Server, page 37 (Required)**

  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.
  
  - (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—for CDMA2000 networks only) 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, specify the `radius username` keywords on the `sticky` command.

If you configure 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.

- **Enabling IOS SLB to Inspect Packets for RADIUS Framed-IP Sticky Routing, page 51** (Optional)
- **Increasing 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*.

- **Configuring Probes, page 44** (Required)

To verify the health of the server, configure a ping probe.

### Enabling 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.

To enable IOS SLB to inspect packets for routing using the RADIUS framed-IP sticky database, enter the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Router(config)# ip slb route (ip-address netmask framed-ip | Enables IOS Server Load Balancing (IOS SLB) to route packets using
inter-firewall) | the RADIUS framed-IP sticky database, or to route packets from one
firewall real server back through another firewall real server. |
VPN Server Load Balancing Configuration Task List

This section lists the tasks used to configure VPN server load balancing. Detailed configuration information is contained in the referenced sections of this or other documents. Required and optional tasks are indicated.

- **Configuring a Server Farm and Real Server, page 34** (Required)
  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.

- **Configuring a Virtual Server, page 37** (Required)
  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.

- **Configuring Probes, page 44** (Required)
  To verify the health of the server, configure a ping probe.
Home Agent Director Configuration Task List

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

- **Configuring a Server Farm and Real Server, page 34 (Required)**
  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.

- **Configuring a Virtual Server, page 37 (Required)**
  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.
  - Configuring the virtual IP address as a loopback on each of the home agents in the server (Required for dispatched mode)
    Refer to the “Configuring a Loopback Interface” section in the *Cisco IOS Interface Configuration Guide*, Release 12.2 for more information.

## Configuring NAT

To configure the IOS SLB NAT client address pool for client NAT, enter the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`Router(config)# ip slb natpool pool start-ip end-ip [netmask netmask</td>
<td>prefix-length leading-1-bits]`</td>
</tr>
<tr>
<td>`</td>
<td>entries init-address [max-address]]`</td>
</tr>
</tbody>
</table>

You need not 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 Real Server” section on page 34 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

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.

To configure NAT for the server, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Configures the real server’s Network Address Translation (NAT) behavior and enters static NAT configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> If you specify the <code>virtual-ip</code> argument and you do not specify the <code>per-packet</code> option, IOS Server Load Balancing (IOS SLB) uses server port translation to distinguish between connection requests initiated by different real servers.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Configures one or more real servers to use static Network Address Translation (NAT).</td>
</tr>
</tbody>
</table>

Stateless Backup Configuration Task List

This section lists the tasks used to configure stateless backup over VLANs between IOS SLB devices. Detailed configuration information is contained in the referenced sections of this or other documents. Required and optional tasks are indicated.

- Configuring Required and Optional IOS SLB Functions, page 33 (Required for server load balancing)
- Configuring Firewall Load Balancing, page 40 (Required for firewall load balancing)
- Configuring the IP Routing Protocol (Required)
- Configuring the VLAN between the IOS SLB devices (Required)
  - Refer to the “Virtual LANs” chapter of the *Cisco IOS Switching Services Configuration Guide*, Release 12.2 for more details.
- Verifying the Stateless Backup Configuration, page 55 (Optional)

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.
Verifying the Stateless Backup Configuration

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

```
Router# show ip slb vserver

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 vserver 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 conns
-----------------------------
FIRE1 IPADDR INSERVICE 2
```

```
Router# show ip slb firewallfarm details

FIRE1, hash = IPADDRPORT, state = INSERVICE, conns = 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
```
Configuring Database Entries

To configure an initial allocation and a maximum value for IOS SLB database entries, enter the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# ip slb entries</td>
<td>Specifies an initial allocation and a maximum value for IOS Server Load Balancing (IOS SLB) database entries.</td>
</tr>
<tr>
<td>conn [init-conn [max-conn]]</td>
<td></td>
</tr>
<tr>
<td>frag [init-frag [max-frag]</td>
<td>lifetime timeout]</td>
</tr>
<tr>
<td>sticky [init-sticky [max-sticky]]</td>
<td></td>
</tr>
</tbody>
</table>

Configuring Buffers for the Fragment Database

To configure the maximum number of buffers for the IOS SLB fragment database, enter the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# ip slb maxbuffers frag buffers</td>
<td>Configures the maximum number of buffers for the IOS Server Load Balancing (IOS SLB) fragment database.</td>
</tr>
</tbody>
</table>

Clearing Databases and Counters

To clear IP IOS SLB databases and counters, enter one or more of the following commands in privileged EXEC mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router# clear ip slb connections</td>
<td>Clears the IOS Server Load Balancing (IOS SLB) connection database for one or more firewall farms, server farms, or virtual servers.</td>
</tr>
<tr>
<td>[firewallfarm firewall-farm</td>
<td>serverfarm server-farm</td>
</tr>
<tr>
<td>Router# clear ip slb counters</td>
<td>Clears the IOS Server Load Balancing (IOS SLB) counters.</td>
</tr>
<tr>
<td>Router# clear ip slb sessions</td>
<td>Clears the IOS Server Load Balancing (IOS SLB) RADIUS session database for one or more firewall farms, server farms, or virtual servers.</td>
</tr>
<tr>
<td>[firewallfarm firewall-farm</td>
<td>serverfarm server-farm</td>
</tr>
<tr>
<td>Router# clear ip slb sticky radius framed-ip</td>
<td>Clears entries from the IOS Server Load Balancing (IOS SLB) RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td>[framed-ip [netmask]]</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Wildcard Searches

To specify the behavior of IOS SLB wildcard searches, enter the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# mls ip slb search {wildcard [pfc</td>
<td>rp]</td>
</tr>
</tbody>
</table>

Purging and Reassigning 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.

To configure IOS SLB’s behavior when a real server fails, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# ip slb serverfarm server-farm</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-sfarm)# failaction {purge</td>
</tr>
</tbody>
</table>

To configure IOS SLB’s behavior when a firewall fails, enter the following commands in order, beginning in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Router(config)# ip slb firewallfarm firewall-farm</td>
</tr>
<tr>
<td>Step 2</td>
<td>Router(config-slb-fw)# failaction purge</td>
</tr>
</tbody>
</table>
## Monitoring and Maintaining the IOS SLB Feature

To obtain and display runtime information about IOS SLB, use the following commands in EXEC mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>`Router# show ip slb conns [vserver virtual-server</td>
<td>Displays all connections handled by IOS Server Load Balancing (IOS SLB), or, optionally, only those connections associated with a particular virtual server or client.</td>
</tr>
<tr>
<td>client ip-address</td>
<td>firewall firewall-farm] [detail]</td>
</tr>
<tr>
<td>`Router# show ip slb dfp [agent agent-ip port</td>
<td>Displays information about Dynamic Feedback Protocol (DFP) and DFP agents, and about the weights assigned to real servers.</td>
</tr>
<tr>
<td>manager manager-ip</td>
<td>detail</td>
</tr>
<tr>
<td>`Router# show ip slb firewallfarm [detail]</td>
<td>Displays information about firewall farms.</td>
</tr>
<tr>
<td>`Router# show ip slb fragments</td>
<td>Displays information from the IOS Server Load Balancing (IOS SLB) fragment database.</td>
</tr>
<tr>
<td>`Router# show ip slb gtp [gsn [gsn-ip-address]</td>
<td>Displays IOS Server Load Balancing (IOS SLB) GTP information.</td>
</tr>
<tr>
<td>nsapi [nsapi-key] [detail]</td>
<td></td>
</tr>
<tr>
<td>`Router# show ip slb natpool [name pool] [detail]</td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) Network Address Translation (NAT) configuration.</td>
</tr>
<tr>
<td>`Router# show ip slb probe [name probe] [detail]</td>
<td>Displays information about probes defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>`Router# show ip slb reals [sfarm server-farm] [detail]</td>
<td>Displays information about the real servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>`Router# show ip slb replicate</td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) replication configuration.</td>
</tr>
<tr>
<td>`Router# show ip slb serverfarms [name server-farm] [detail]</td>
<td>Displays information about the server farms defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>`Router# show ip slb sessions [gtp</td>
<td>gtp-inspect</td>
</tr>
<tr>
<td>`Router# show ip slb static</td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) server Network Address Translation (NAT) configuration.</td>
</tr>
<tr>
<td>`Router# show ip slb stats</td>
<td>Displays IOS Server Load Balancing (IOS SLB) statistics.</td>
</tr>
<tr>
<td>`Router# show ip slb sticky [client ip-address netmask</td>
<td>Displays information about the sticky connections defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>radius framed-ip [client ip-address netmask]</td>
<td>radius username [name string]</td>
</tr>
<tr>
<td>`Router# show ip slb vserver [name virtual-server] [redirect] [detail]</td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
Configuration Examples

This section provides real-world examples of IOS SLB configurations. For a complete description of the IOS SLB commands in this section, see the “Command Reference” section on page 116. To locate documentation of other commands that appear in this section, search online using Cisco.com.

This section includes the following examples:

- Basic IOS SLB Network Configuration Example, page 60
- Complete IOS SLB Configuration Example, page 62
- IOS SLB with Firewall Load Balancing Example, page 63
- IOS SLB with Server Load Balancing and Firewall Load Balancing Example, page 65
- IOS SLB with Multiple Firewall Farms Example, page 68
- IOS SLB with Probes Example, page 70
- IOS SLB with Routed Probe Example, page 71
- Layer 3 Switch Configured with IOS SLB Example, page 72
- IOS SLB with NAT Example, page 74
- IOS SLB with Static NAT Example, page 77
- IOS SLB with Stateless Backup Examples, page 77
- IOS SLB with Stateful Backup Example, page 86
- IOS SLB with Active Standby Example, page 89
- IOS SLB with Redistribution of Static Routes Example, page 92
- IOS SLB with WAP and UDP Load Balancing Examples, page 94
- IOS SLB with Route Health Injection Examples, page 96
- IOS SLB with GPRS Load Balancing Example, page 99
- IOS SLB with GPRS Load Balancing and NAT Example, page 103
- IOS SLB with GPRS Load Balancing, NAT, and GTP Cause Code Inspection Example, page 106
- IOS SLB with VPN Server Load Balancing Example, page 108
- IOS SLB with RADIUS Load Balancing for a GPRS Network Example, page 109
- IOS SLB with RADIUS Load Balancing for a Simple IP CDMA2000 Network Example, page 110
- IOS SLB with RADIUS Load Balancing for a Mobile IP CDMA2000 Network Example, page 111
- IOS SLB with RADIUS Load Balancing for Multiple Service Gateway Farms Example, page 112
- IOS SLB with Home Agent Director Example, page 114
- IOS SLB with Sticky Connections Example, page 114
- IOS SLB with Transparent Web Cache Load Balancing, page 115

Note

The IP and network addresses in these examples are generic; you must replace them with the actual addresses for your network.
Basic IOS SLB Network Configuration 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.

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 61
- Virtual Server Configuration, page 61
- Restricted Client Configuration, page 62
Server Farm Configuration

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

```plaintext
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:

```plaintext
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:

```plaintext
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
```

Complete IOS SLB Configuration 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
!```
**IOS SLB with 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

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:

```plaintext
!-----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 firewall farm 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:

```plaintext
!-----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 firewall farm FIRE1
!-----First firewall
   real 10.1.1.2
   probe PROBE1
!-----Enable first firewall
   inservice
```
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
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:

```plaintext
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
```
Cisco IOS Release 12.2(14)ZA2

Configuration Examples

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
**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.

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
```
IOS SLB with Probes Example

Figure 6 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 6 Sample Ping and HTTP Probe Topology

The topology shown in Figure 6 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
Cisco IOS Release 12.2(14)ZA2

! 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

IOS SLB with Routed Probe Example

Figure 7 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 7  IOS SLB with a Routed Ping Probe

Following are the IOS SLB configuration statements for the configuration shown in Figure 7:

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
Layer 3 Switch Configured with IOS SLB Example

Figure 8 shows a sample configuration with IOS SLB server connections configured as part of a server farm.

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
IOS SLB with NAT Example

Figure 9 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.
The topology in Figure 9 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

```plaintext
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

```plaintext
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

! 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
**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, 10.0.0.0, 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 10.0.0.0
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 10.0.0.0
```

**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:

- Dynamic Routing and Trunking Example, page 78
- Dynamic Routing and No Trunking Example, page 80
- Static Routing and Trunking Example, page 82
- Static Routing and No Trunking Example, page 84

**Note**  
Stateful backup is omitted from these examples in the interest of simplicity. To see an example that uses stateful backup, see the “IOS SLB with Stateful Backup Example” section on page 86.

### Dynamic Routing and Trunking Example

**Figure 10** 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 10  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
  switchport
  switchport vlan 1

plug vlan 1
  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

**Dynamic Routing and No 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 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 11  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
```

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

Static Routing and 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 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 12  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

**Static Routing and No 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 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.

![Stateless Backup with Layer 2 and No Trunking](image-url)
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
```
Configuration Examples

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

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 14 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.
Configuration Examples

Figure 14  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

```plaintext
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
```

IOS SLB with Active Standby Example

Figure 15 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.

The sample network configuration in Figure 15 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
```

**IOS SLB with Redistribution of Static Routes Example**

*Figure 16* 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 “Command Reference” section on page 116 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 16  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 16:

```
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 16:

```
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 16:

```
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 16:

```
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 16:

```
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 16:

```
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 16:

```
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 16:

```
router eigrp 666
network 10.0.0.0
network 8.0.0.0
```

**IOS SLB with WAP and UDP Load Balancing Examples**

Figure 17 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 17 IOS SLB with WAP Load Balancing](image)
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.

WAP Load Balancing Example

The following example shows the configuration for the IOS SLB device shown in Figure 17, which balances WAP flows on UDP port 9201 (WSP/WTP/UDP):

```plaintext
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
```

UDP Load Balancing Example

The following example shows the configuration for the IOS SLB device shown in Figure 17, which balances WAP flows on UDP port 9203 (WSP/WTP/WTLS/UDP):

```plaintext
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
```
IOS SLB with Route Health Injection Examples

This section contains the following examples, illustrating several different IOS SLB route health injection configurations:

- Example with Two Distributed Sites with One Web Server Each, page 96
- Example with Two Distributed Sites with Two Web Servers Each, page 97
- Example with Two Distributed Sites with One Web Server and a Backup IOS SLB Switch Each, page 98

Example with Two Distributed Sites with One Web Server Each

Figure 18 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.

When both web servers in Figure 18 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.
Example with Two Distributed Sites with Two Web Servers Each

Figure 19 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.

When all web servers in Figure 19 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.
Example with Two Distributed Sites with One Web Server and a Backup IOS SLB Switch Each

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.
- Each site has a primary IOS SLB device and a backup IOS SLB device.
- The path to SLB A has the lower weight.

When both web servers in Figure 20 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.
**IOS SLB with GPRS Load Balancing Example**

Figure 21 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 21  IOS SLB with GPRS Load Balancing**

Following are the configuration statements for the configuration shown in Figure 21:

- IOS SLB Configuration Statements, page 100
- GGSN1 Configuration Statements, page 100
- GGSN2 Configuration Statements, page 101
- GGSN3 Configuration Statements, page 102

For more detailed GGSN configuration examples, refer to the Cisco IOS Mobile Wireless Configuration Guide, Release 12.2.
**IOS SLB Configuration Statements**

```cisco
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**

```cisco
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 mroutecache
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 mroutecache
!
interface FastEthernet1/0
    description SLB
    ip address 10.1.1.202 255.255.255.0
    ip directed-broadcast
    no ip mroutecache
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
dhcpserver 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

IOS SLB with GPRS Load Balancing and NAT Example

The following example uses the same basic configuration as in the “IOS SLB with GPRS Load Balancing Example” section on page 99, including the network shown in Figure 21, but with the addition of NAT:

- IOS SLB Configuration Statements, page 103
- GGSN1 Configuration Statements, page 104
- GGSN2 Configuration Statements, page 105
- GGSN3 Configuration Statements, page 105

For more detailed GGSN configuration examples, refer to the Cisco IOS Mobile Wireless Configuration Guide, Release 12.2.

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
  !
### 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 10.1.1.1 255.255.255.0
 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
IOS SLB with GPRS Load Balancing, NAT, and GTP Cause Code Inspection Example

The following example uses the same basic configuration as in the “IOS SLB with GPRS Load Balancing and NAT Example” section on page 103, including the network shown in Figure 21, 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 (IMSI) with carrier code mcc 222 mnc 22.

Following are the configuration statements for the configuration shown in Figure 21, with the addition of NAT and GTP cause code inspection support:

- IOS SLB Configuration Statements, page 107
- GGSN1 Configuration Statements, page 104 (no change for GTP cause code inspection)
- GGSN2 Configuration Statements, page 105 (no change for GTP cause code inspection)
- GGSN3 Configuration Statements, page 105 (no change for GTP cause code inspection)

For more detailed GGSN configuration examples, refer to the *Cisco IOS Mobile Wireless Configuration Guide*, Release 12.2.
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
IOS SLB with VPN Server Load Balancing Example

Figure 22 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).

Following are the IOS SLB configuration statements for the configuration shown in Figure 22:

```
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
```
**IOS SLB with RADIUS Load Balancing for a GPRS Network Example**

Figure 23 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 23  IOS SLB with RADIUS Load Balancing for a GPRS Network*

Following are the IOS SLB configuration statements for the configuration shown in **Figure 23**:

```
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 1812 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 1813 service radius
serverfarm SSGFARM
idle radius request 20
idle radius framed-ip 7200
sticky radius framed-ip group 1
inservice

IOS SLB with RADIUS Load Balancing for a Simple IP CDMA2000 Network Example

Figure 24 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 24  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 24:

```
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
```

**IOS SLB with RADIUS Load Balancing for a Mobile IP CDMA2000 Network Example**

*Figure 25* 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.
Following are the IOS SLB configuration statements for the configuration shown in Figure 25:

```plaintext
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
```

**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):

```plaintext
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
## IOS SLB with Home Agent Director Example

The following sample configuration enables IOS SLB to balance Mobile IP RRQs among multiple home agents.

*Figure 26  IOS SLB with Home Agent Director*

Following are the IOS SLB configuration statements for the configuration shown in Figure 26:

```plaintext
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
```

## 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:

```plaintext
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:

```plaintext
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.
IOS SLB with Transparent Web Cache Load Balancing

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.

```conf
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
```
Command Reference

This section documents only new and modified commands.

- access (firewall farm)
- access (virtual server)
- address (custom UDP probe)
- address (DNS probe)
- address (HTTP probe)
- address (ping probe)
- address (TCP probe)
- address (WSP probe)
- advertise
- agent (DFP)
- bindid
- clear ip slb connections
- clear ip slb counters
- clear ip slb sessions
- clear ip slb sticky radius framed-ip
- client (virtual server)
- credentials
- debug gprs dfp
- debug ip slb
- delay (firewall farm TCP protocol)
- delay (virtual server)
- expect
- failaction (firewall farm)
- failaction (server farm)
- faildetect (DNS probe)
- faildetect (ping probe)
- faildetect (real server)
- hand-off radius
- header
- idle (firewall farm datagram protocol)
- idle (firewall farm TCP protocol)
- idle (virtual server)
- inservice (firewall farm)
- inservice (firewall farm real server)
- inservice (server farm real server)
- inservice (server farm virtual server)
- interval (custom UDP probe)
- interval (DNS probe)
- interval (HTTP probe)
- interval (ping probe)
- interval (TCP probe)
- interval (WSP probe)
- ip slb dfp
- ip slb entries
- ip slb firewallfarm
- ip slb maxbuffers frag
- ip slb natpool
- ip slb probe custom udp
- ip slb probe dns
- ip slb probe http
- ip slb probe ping
- ip slb probe tcp
- ip slb probe wsp
- ip slb route
- ip slb serverfarm
- ip slb static
- ip slb vserver
- lookup
- maxclients
- maxconns (firewall farm datagram protocol)
- maxconns (firewall farm TCP protocol)
- maxconns (server farm)
- mls aging slb normal
- mls aging slb process
- mls ip slb search wildcard
- nat
- port (custom UDP probe)
- port (HTTP probe)
- port (TCP probe)
- predictor (server farm)
- predictor hash address (firewall farm)
- probe (firewall farm real server)
- probe (server farm)
- protocol datagram
- protocol tcp
- purge radius framed-ip acct on-off (virtual server)
- real (firewall farm)
- real (server farm)
- real (static NAT)
- reassign
- replicate casa (firewall farm)
- replicate casa (virtual server)
- request (custom UDP probe)
- request (HTTP probe)
- response
- retry
- serverfarm
- show ip slb conns
- show ip slb dpf
- show ip slb firewallfarm
- show ip slb fragments
- show ip slb gtp
- show ip slb natpool
- show ip slb probe
- show ip slb reals
- show ip slb replicate
- show ip slb serverfarms
- show ip slb sessions
- show ip slb static
- show ip slb stats
- show ip slb sticky
- show ip slb vserver
- snmp-server enable traps slb
- sticky (firewall farm datagram protocol)
- sticky (firewall farm TCP protocol)
- sticky (virtual server)
- synguard (virtual server)
- url (WSP probe)
- virtual (virtual server)
- weight (firewall farm real server)
- weight (server farm)
### access (firewall farm)

To route specific flows to a firewall farm, use the `access` command in firewall farm configuration mode.
To restore the default settings, use the `no` form of this command.

```
access [source source-ip netmask] [destination destination-ip netmask]
```

```
o access [source source-ip netmask] [destination destination-ip netmask]
```

#### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>source</code></td>
<td>(Optional) Routes flows based on source IP address.</td>
</tr>
<tr>
<td><code>source-ip</code></td>
<td>(Optional) Source IP address. The default is 0.0.0.0 (all sources).</td>
</tr>
<tr>
<td><code>netmask</code></td>
<td>(Optional) Source IP network mask. The default is 0.0.0.0 (all source subnets).</td>
</tr>
<tr>
<td><code>destination</code></td>
<td>(Optional) Routes flows based on destination IP address.</td>
</tr>
<tr>
<td><code>destination-ip</code></td>
<td>(Optional) Destination IP address. The default is 0.0.0.0 (all destinations).</td>
</tr>
<tr>
<td><code>netmask</code></td>
<td>(Optional) Destination IP network mask. The default is 0.0.0.0 (all destination subnets).</td>
</tr>
</tbody>
</table>

#### Defaults

The default source IP address is 0.0.0.0 (routes flows from all sources to this firewall farm).
The default source IP network mask is 0.0.0.0 (routes flows from all source subnets to this firewall farm).
The default destination IP address is 0.0.0.0 (routes flows from all destinations to this firewall farm).
The default destination IP network mask is 0.0.0.0 (routes flows from all destination subnets to this firewall farm).

#### Command Modes

Firewall farm configuration

#### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(7)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

#### Usage Guidelines

You can specify more than one source or destination for each firewall farm. To do so, configure multiple access statements, making sure the network masks do not overlap each other.

#### Examples

The following example routes flows with a destination IP address of 10.1.6.0 to firewall farm FIRE1:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# access destination 10.1.6.0 255.255.255.0
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb firewallfarm</td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
**access (virtual server)**

To enable framed-IP routing to inspect the ingress interface, use the `access` command in virtual server configuration mode. To disable framed-IP routing, use the `no` form of this command.

```
access interface route framed-ip
no access interface route framed-ip
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface</td>
<td>Interface to be inspected.</td>
</tr>
<tr>
<td>route framed-ip</td>
<td>Routes flows using framed-IP routing.</td>
</tr>
</tbody>
</table>

**Defaults**

Framed-IP routing cannot inspect the ingress interface.

**Command Modes**

Virtual server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(12c)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can specify more than one source or destination for each firewall farm. To do so, configure multiple access statements, making sure the network masks do not overlap each other.

This command enables framed-IP routing to inspect the ingress interface when routing subscriber traffic. All framed-IP sticky database entries created as a result of RADIUS requests to this virtual server will include the interface in the entry. In addition to matching the source IP address of the traffic with the framed-IP address, the ingress interface must also match this interface when this command is configured.

You can use this command to allow subscriber data packets to be routed to multiple service gateway service farms.

**Examples**

The following example enables framed-IP routing to inspect ingress interface Vlan20:

```
Router(config)# ip slb vserver SSG_AUTH
Router(config-slb-vserver)# access Vlan20 route framed-ip
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb vserver</td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
To configure an IP address to which to send custom User Datagram Protocol (UDP) probes, use the `address` command in custom UDP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
address [ip-address] [routed]
```

```
no address [ip-address] [routed]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip-address</code></td>
<td>(Optional) Destination IP address that is to respond to the custom UDP probe.</td>
</tr>
<tr>
<td><code>routed</code></td>
<td>(Optional) Flags the probe as a routed probe, with the following considerations:</td>
</tr>
<tr>
<td></td>
<td>• Only one instance of a routed probe per server farm can run at any given time.</td>
</tr>
<tr>
<td></td>
<td>• Outbound packets for a routed probe are routed directly to <code>ip-address</code>.</td>
</tr>
</tbody>
</table>

**Defaults**

If the custom UDP probe is associated with a firewall farm, you must specify an IP address. If the custom UDP probe is associated with a server farm, and you do not specify an IP address, the address is inherited from the server farm real servers.

**Command Modes**

Custom UDP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a custom UDP probe named PROBE6, enters custom UDP probe configuration mode, and configures the probe to receive responses from IP address 13.13.13.13:

```
Router(config)# ip slb probe PROBE6 custom udp
Router(config-slb-probe)# address 13.13.13.13
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe custom udp</code></td>
<td>Configures a custom User Datagram Protocol (UDP) probe name and enters custom UDP probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
address (DNS probe)

To configure an IP address to which to send Domain Name System (DNS) probes, use the address command in DNS probe configuration mode. To restore the default settings, use the no form of this command.

```
address [ip-address [routed]]

no address [ip-address [routed]]
```

**Syntax Description**

- `ip-address` (Optional) Destination IP address that is to respond to the DNS probe.
- `routed` (Optional) Flags the 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 the specified IP address.

**Defaults**

If the DNS probe is associated with a firewall farm, you must specify an IP address.
If the DNS probe is associated with a server farm, and you do not specify an IP address, the address is inherited from the server farm real servers.

**Command Modes**

DNS probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <code>routed</code> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a DNS probe named PROBE4, enters DNS probe configuration mode, and configures the probe to receive responses from IP address 13.13.13.13:

```
Router(config)# ip slb probe PROBE4 dns
Router(config-slb-probe)# address 13.13.13.13
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe dns</code></td>
<td>Configures a Domain Name System (DNS) probe name and enters DNS probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
address (HTTP probe)

To configure an IP address to which to send HTTP probes, use the **address** command in HTTP probe configuration mode. To restore the default settings, use the **no** form of this command.

```
address [ip-address [routed]]
no address [ip-address [routed]]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip-address</code></td>
<td>(Optional) Destination IP address that is to respond to the HTTP probe.</td>
</tr>
<tr>
<td><code>routed</code></td>
<td>(Optional) Flags the probe as a routed probe, with the following considerations:</td>
</tr>
<tr>
<td></td>
<td>- Only one instance of a routed probe per server farm can run at any given time.</td>
</tr>
<tr>
<td></td>
<td>- Outbound packets for a routed probe are routed directly to the specified IP address.</td>
</tr>
</tbody>
</table>

**Defaults**

If the HTTP probe is associated with a firewall farm, you must specify an IP address.
If the HTTP probe is associated with a server farm, and you do not specify an IP address, the address is inherited from the server farm real servers.

**Command Modes**

HTTP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <strong>routed</strong> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures an HTTP probe named PROBE2, enters HTTP probe configuration mode, and configures the probe to receive responses from IP address 13.13.13.13:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# address 13.13.13.13
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb probe http</td>
<td>Configures an HTTP probe name and enters HTTP probe configuration mode.</td>
</tr>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
To configure an IP address to which to send ping probes, use the `address` command in ping probe configuration mode. To restore the default settings, use the `no` form of this command.

```
address [ip-address [routed]]

no address [ip-address [routed]]
```

### Syntax Description
- **ip-address** (Optional) Destination IP address that is to respond to the ping probe.
- **routed** (Optional) Flags the 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 the specified IP address.

### Defaults
If the ping probe is associated with a firewall farm, you must specify an IP address.
If the ping probe is associated with a server farm, and you do not specify an IP address, the address is inherited from the server farm real servers.

### Command Modes
Ping probe configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <code>routed</code> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Examples
The following example configures a ping probe named PROBE1, enters ping probe configuration mode, and configures the probe to receive responses from IP address 13.13.13.13:

```
Router(config)# ip slb probe PROBE1 ping
Router(config-slb-probe)# address 13.13.13.13
```

### Related Commands
- **Command**
  - `ip slb probe ping` Configures a ping probe name and enters ping probe configuration mode.
  - `show ip slb probe` Displays information about an IOS Server Load Balancing (IOS SLB) probe.
address (TCP probe)

To configure an IP address to which to send TCP probes, use the `address` command in TCP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
address [ip-address [routed]]
no address [ip-address [routed]]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip-address</code></td>
<td>(Optional) Destination IP address that is to respond to the TCP probe.</td>
</tr>
</tbody>
</table>
| `routed`           | (Optional) Flags the 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 the specified IP address. |

**Defaults**

If the TCP probe is associated with a firewall farm, you must specify an IP address.
If the TCP probe is associated with a server farm, and you do not specify an IP address, the address is inherited from the server farm real servers.

**Command Modes**

TCP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <code>routed</code> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a TCP probe named PROBE5, enters TCP probe configuration mode, and configures the probe to receive responses from IP address 13.13.13.13:

```
Router(config)# ip slb probe PROBE5 tcp
Router(config-slb-probe)# address 13.13.13.13
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe tcp</code></td>
<td>Configures a TCP probe name and enters TCP probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
address (WSP probe)

To configure an IP address to which to send Wireless Session Protocol (WSP) probes, use the `address` command in WSP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
address [ip-address [routed]]
no address [ip-address [routed]]
```

**Syntax Description**

- **ip-address** (Optional) Destination IP address that is to respond to the WSP probe.
- **routed** (Optional) Flags the 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 the specified IP address.

**Defaults**

If the WSP probe is associated with a firewall farm, you must specify an IP address.
If the WSP probe is associated with a server farm, and you do not specify an IP address, the address is inherited from the server farm real servers.
In dispatched mode, the `ip-address` argument value is the same as the virtual server IP address. In directed Network Address Translation (NAT) mode, an IP address is unnecessary.

**Command Modes**

WSP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(5a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <code>routed</code> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a WSP probe named PROBE3, enters WSP probe configuration mode, and configures the probe to receive responses from IP address 13.13.13.13:

```
Router(config)# ip slb probe PROBE3 wsp
Router(config-slb-probe)# address 13.13.13.13
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe wsp</code></td>
<td>Configures a Wireless Session Protocol (WSP) probe name and enters WSP probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
advertise

To control the installation of a static route to the Null0 interface for a virtual server address, use the `advertise` command in virtual server configuration mode. To prevent the installation of a static route for the virtual server IP address, use the `no` form of this command.

```
advertise [active]
no advertise [active]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td>(Optional) Indicates that the host route is to be advertised only when the virtual IP address is available (that is, when there is at least one real server in OPERATIONAL, DFP_THROTTLED, or MAXCONNS state).</td>
</tr>
</tbody>
</table>

**Defaults**

The virtual server IP address is advertised. That is, a static route to the Null0 interface is installed for the virtual server IP addresses and it is added to the routing table.

If you do not specify the `active` keyword, the host route is advertised regardless of whether the virtual IP address is available.

**Command Modes**

Virtual server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(7)E</td>
<td>The <code>active</code> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Advertisement of a static route using the routing protocol requires that you configure redistribution of static routes for the routing protocol.

The `advertise` command does not affect virtual servers used for transparent web cache load balancing.

HTTP probes and route health injection 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 and route health injection to function correctly.

- For HTTP probes, 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). If you specify either the `no advertise` or the `advertise active` command, you must specify a default route.
- For route health injection, the route must be a default route.

HTTP probes and route health injection can both use the same default route; you need not specify two unique default routes.
Examples

The following example prevents advertisement of the virtual server’s IP address in routing protocol updates:

Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# no advertise

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb vserver</td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
agent (DFP)

To identify a Dynamic Feedback Protocol (DFP) agent with which the IOS Server Load Balancing (IOS SLB) feature can initiate connections, use the `agent` command in DFP configuration mode. To remove a DFP agent definition from the DFP configuration, use the `no` form of this command.

```
agent ip-address port [timeout [retry-count [retry-interval]]]

no agent ip-address port
```

### Syntax Description

- `ip-address`: Agent IP address.
- `port`: Agent TCP or User Datagram Protocol (UDP) port number.
- `timeout`: (Optional) Time period, in seconds, during which the DFP manager must receive an update from the DFP agent. The valid range is 0 to 65535 seconds. The default is 0 seconds, which means there is no timeout.
- `retry-count`: (Optional) Number of times the DFP manager attempts to establish the TCP connection to the DFP agent. The valid range is 0 to 65535 times. The default is 0 retries, which means there are infinite retries.
- `retry-interval`: (Optional) Interval, in seconds, between retries. The valid range is 1 to 65535 seconds. The default is 180 seconds.

### Defaults

- The default timeout is 0 seconds (no timeout).
- The default retry count is 0 (infinite retries).
- The default retry interval is 180 seconds.

### Command Modes

DFP configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The password specified in the `ip slb dfp` command for the DFP manager must match the password specified in the `password` command for the DFP agent.

You can identify up to 1024 DFP agents.
Examples  The following example sets the DFP password to Password1 (to match the DFP agent’s password), sets the timeout to 360 seconds, enters DFP configuration mode, and enables IOS SLB to connect to the DFP agent with IP address 10.1.1.1 and port number 2221:

```
Router(config)# ip slb dfp password Password1 360
Router(config-slb-dfp)# agent 10.1.1.1 2221 30 0 10
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip dfp agent</td>
<td>Identifies a Dynamic Feedback Protocol (DFP) agent subsystem and enters DFP agent configuration mode.</td>
</tr>
<tr>
<td>ip slb dfp</td>
<td>Configures Dynamic Feedback Protocol (DFP), supplies an optional password, and enters DFP configuration mode.</td>
</tr>
</tbody>
</table>
bindid

To configure a bind ID, use the `bindid` command in server farm configuration mode. To remove a bind ID from the server farm configuration, use the `no` form of this command.

```
bindid [bind-id]
no bindid [bind-id]
```

**Syntax Description**
- `bind-id` (Optional) Bind ID number. The default bind ID is 0.

**Defaults**
The default bind ID is 0.

**Command Modes**
Server farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
You can configure one bind ID on each `bindid` command.

The bind ID allows a single physical server to be bound to multiple virtual servers, and to 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.

In general packet radio service (GPRS) load balancing, bind IDs are not supported. Therefore do not use the `bindid` command in a GPRS load-balancing environment.

**Examples**
The following example configures bind ID 309:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# bindid 309
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb dfp</code></td>
<td>Configures Dynamic Feedback Protocol (DFP), supplies an optional password, and enters DFP configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb serverfarms</code></td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) server farms.</td>
</tr>
</tbody>
</table>
**clear ip slb connections**

To clear IP IOS Server Load Balancing (IOS SLB) connections, use the `clear ip slb connections` command in privileged EXEC mode.

`clear ip slb connections [firewallfarm firewall-farm | serverfarm server-farm | vserver virtual-server]`

**Syntax Description**

- **firewallfarm firewall-farm** (Optional) Clears the IOS SLB connection database for the specified firewall farm.
- **serverfarm server-farm** (Optional) Clears the IOS SLB connection database for the specified server farm.
- **vserver virtual-server** (Optional) Clears the IOS SLB connection database for the specified virtual server.

**Defaults**

The IOS SLB connection database is cleared for all firewall farms, server farms, and virtual servers.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(1)E</td>
<td>This command was introduced as part of the <code>clear ip slb</code> command.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>This command was separated from the <code>clear ip slb</code> command.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

In general packet radio service (GPRS) load balancing, the `clear ip slb connections` command clears connections, but does not clear sessions.

**Examples**

The following example clears the connection database of server farm FARM1:

```
Router# clear ip slb connections serverfarm FARM1
```

The following example clears the connection database of virtual server VSERVER1:

```
Router# clear ip slb connections vserver VSERVER1
```
## clear ip slb connections

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb conns</td>
<td>Displays information about active IOS Server Load Balancing (IOS SLB) connections.</td>
</tr>
<tr>
<td>show ip slb firewallfarm</td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td>show ip slb serverfarms</td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) server farms.</td>
</tr>
<tr>
<td>show ip slb vserver</td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
clear ip slb counters

To clear IP IOS Server Load Balancing (IOS SLB) counters, use the clear ip slb counters command in privileged EXEC mode.

    clear ip slb counters

Syntax Description
This command has no arguments or keywords.

Defaults
IP IOS SLB counters are not cleared.

Command Modes
Privileged EXEC

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(1)E</td>
<td>This command was introduced as part of the clear ip slb command.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>This command was separated from the clear ip slb command.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

Examples

The following example clears the IP IOS SLB counters:

    Router# clear ip slb counters

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb stats</td>
<td>Displays IOS Server Load Balancing (IOS SLB) statistics.</td>
</tr>
</tbody>
</table>
clear ip slb sessions

To clear the IP IOS Server Load Balancing (IOS SLB) sessions database, use the `clear ip slb sessions` command in privileged EXEC mode.

```
clear ip slb sessions [firewallfarm firewall-farm | serverfarm server-farm | vserver virtual-server]
```

**Syntax Description**
- `firewallfarm firewall-farm` (Optional) Clears the IOS SLB session database for the specified firewall farm.
- `serverfarm server-farm` (Optional) Clears the IOS SLB session database for the specified server farm.
- `vserver virtual-server` (Optional) Clears the IOS SLB session database for the specified virtual server.

**Command Modes**
- Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no optional keywords or arguments are specified, the IOS SLB sessions database is cleared of all firewall farms, server farms, and virtual servers.

**Examples**

The following example clears the session database of server farm FARM1:

```
Router# clear ip slb sessions serverfarm FARM1
```

The following example clears the session database of virtual server VSERVER1:

```
Router# clear ip slb sessions vserver VSERVER1
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) firewall farms.</td>
</tr>
<tr>
<td><code>show ip slb serverfarms</code></td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) server farms.</td>
</tr>
<tr>
<td><code>show ip slb sessions</code></td>
<td>Displays information about sessions handled by IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
clear ip slb sticky radius framed-ip

To clear entries from the IOS Server Load Balancing (IOS SLB) RADIUS framed-IP sticky database, use the `clear ip slb sticky radius framed-ip` command in privileged EXEC mode.

`clear ip slb sticky radius framed-ip [framed-ip [netmask]]`

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>framed-ip</td>
<td>(Optional) Framed-IP address of entries to be cleared.</td>
</tr>
<tr>
<td>netmask</td>
<td>(Optional) Subnet mask specifying a range of entries to be cleared.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no optional arguments are specified, all entries are cleared from the IOS SLB RADIUS framed-IP sticky database.

**Examples**

The following example clears all entries from the IOS SLB RADIUS framed-IP sticky database:

```
Router# clear ip slb sticky radius framed-ip
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb sticky</code></td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) sticky database.</td>
</tr>
</tbody>
</table>
client (virtual server)

To define which clients are allowed to use the virtual server, use the client command in virtual server configuration mode. To remove a client definition from the IOS Server Load Balancing (IOS SLB) configuration, use the no form of this command.

```
client {ip-address netmask [exclude] | gtp carrier-code [code]}
no client {ip-address netmask [exclude] | gtp carrier-code [code]}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>(Optional) Client IP address. The default is 0.0.0.0 (all clients).</td>
</tr>
<tr>
<td>netmask</td>
<td>(Optional) Client IP network mask. The default is 0.0.0.0 (all subnets).</td>
</tr>
<tr>
<td>exclude</td>
<td>(Optional) Ignores connections initiated by the client IP address from the load-balancing scheme.</td>
</tr>
<tr>
<td>gtp carrier-code</td>
<td>(Optional) For general packet radio service (GPRS) Tunneling Protocol (GTP) cause code inspection, configures the virtual server to accept Packet Data Protocol (PDP) context creates only from the specified IMSI carrier code.</td>
</tr>
<tr>
<td>code</td>
<td>(Optional) For GTP cause code inspection, identifies the IMSI carrier code from which this virtual server is to accept PDP context creates.</td>
</tr>
</tbody>
</table>

The code has the format:

```
mcc mcc-code mnc mnc-code
```

where:

- `mcc-code` is the Mobile Country Code (MCC)
- `mnc-code` is the Mobile Network Code (MNC)

If you do not specify a code, the virtual server accepts PDP context creates from any IMSI carrier code.

**Defaults**

- The default client IP address is 0.0.0.0 (all clients).
- The default client IP network mask is 0.0.0.0 (all subnets).
- Taken together, the default is client 0.0.0.0 0.0.0.0 (allows all clients on all subnets to use the virtual server).
- If you specify gtp carrier-code and you do not specify a code, the virtual server accepts PDP context creates from any IMSI carrier code.

**Command Modes**

Virtual server configuration
client (virtual server)

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(1)E</td>
<td>The <code>exclude</code> keyword was added.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
<tr>
<td>12.1(13)E3</td>
<td>The <code>gtp request</code> keyword and <code>code</code> argument were added.</td>
</tr>
</tbody>
</table>

Usage Guidelines

You can use more than one client command to define more than one client.

The `netmask` value is applied to the source IP address of incoming connections. The result must match the `ip-address` value for the client to be allowed to use the virtual server.

If you configure probes in your network, you must also do one of the following:

- Configure the `exclude` keyword on the `client` command on the virtual server to exclude connections initiated by the client IP address from the load-balancing scheme.
- Configure IP addresses on the IOS SLB device that are Layer 3-adjacent to the real servers used by the virtual server.

Configure separate `client` commands to specify the clients that can use the virtual server, and to specify the IMSI carrier code from which the virtual server is to accept PDP context creates.

Examples

The following example allows clients from only 10.4.4.0 access to the virtual server:

```plaintext
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# client 10.4.4.0 255.255.255.0
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td><code>virtual (virtual server)</code></td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
credentials

To configure basic authentication values for the HTTP IOS Server Load Balancing (IOS SLB) probe, use the credentials command in HTTP probe configuration mode. To remove a credentials configuration, use the no form of this command.

```
credentials username [password]
no credentials username [password]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>Authentication username of the HTTP probe header. The character string is limited to 15 characters.</td>
</tr>
<tr>
<td>password</td>
<td>(Optional) Authentication password of the HTTP probe header. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

**Defaults**

Basic authentication values for the HTTP IOS SLB probe are not configured.

**Command Modes**

HTTP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures an HTTP probe named PROBE2, enters HTTP probe configuration mode, sets the HTTP authentication to username Username1, and sets the password to develop:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# credentials Username1 develop
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
debug gprs dfp

To display debugging messages for general packet radio service (GPRS) Dynamic Feedback Protocol (DFP) weight calculation, use the `debug gprs dfp` command in user EXEC or privileged EXEC mode. To disable debugging output, use the `no` form of this command.

```
debug gprs dfp
no debug gprs dfp
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
No default behavior or values

**Command Modes**
User EXEC or privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(9)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
This command displays debugging messages for GPRS DFP weight calculation. To display debugging messages for the DFP agent subsystem, use the `debug ip dfp agent` command.

⚠️ **Caution**
Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use `debug` commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use `debug` commands during periods of lower network flows and fewer users. Debugging during these periods reduces the effect these commands have on other users on the system.

**Examples**
The following example configures a debugging session to check all GPRS DFP weight calculation:

```
Router# debug gprs dfp
GPRS DFP debugging is on
Router#
```

The following example stops all debugging:

```
Router# no debug all
All possible debugging has been turned off
Router#
```
**debug ip slb**

To display debugging messages for IOS Server Load Balancing (IOS SLB), use the `debug ip slb` command in user EXEC or privileged EXEC mode. To disable debugging output, use the `no` form of this command.

```
default ip slb {conns [acl-number] | dfp | firewallfarm | fragments | gtp | icmp | natpool | probe | reals | replication | route | sessions [gtp | ipmobile | radius] | vservers | all}
no default ip slb {conns [acl-number] | dfp | firewallfarm | fragments | gtp | icmp | natpool | probe | reals | replication | route | sessions [gtp | ipmobile | radius] | vservers | all}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>all</strong></td>
<td>Displays all debugging messages for IOS SLB.</td>
</tr>
<tr>
<td><strong>conns [acl-number]</strong></td>
<td>Displays debugging messages for all connections being handled by IOS SLB, including Wireless Session Protocol (WSP) events and states.</td>
</tr>
<tr>
<td></td>
<td>The optional <code>acl-number</code> argument references an IP access control list (ACL). This argument limits the information displayed based on the client IP address, real server IP address, or virtual server IP address:</td>
</tr>
<tr>
<td></td>
<td>• For simple ACLs, IOS SLB checks the client IP address.</td>
</tr>
<tr>
<td></td>
<td>• For extended ACLs, IOS SLB checks the client real and virtual IP addresses.</td>
</tr>
<tr>
<td></td>
<td>For more information about ACLs, refer to the “Configuring IP Services” chapter of the <em>Cisco IOS IP Configuration Guide</em>, Release 12.2.</td>
</tr>
<tr>
<td><strong>dfp</strong></td>
<td>Displays debugging messages for Dynamic Feedback Protocol (DFP).</td>
</tr>
<tr>
<td></td>
<td>• To display debugging messages for the DFP agent subsystem, use the <code>debug ip dfp agent</code> command.</td>
</tr>
<tr>
<td></td>
<td>• To display debugging messages for the general packet radio service (GPRS) DFP weight calculation, use the <code>debug gprs dfp</code> command.</td>
</tr>
<tr>
<td><strong>firewallfarm</strong></td>
<td>Displays debugging messages related to firewall load balancing.</td>
</tr>
<tr>
<td><strong>fragments</strong></td>
<td>Displays debugging messages related to the IOS SLB fragment database.</td>
</tr>
<tr>
<td><strong>gtp</strong></td>
<td>Displays all GPRS Tunneling Protocol (GTP)-related packet handler, gateway GPRS support node (GGSN), serving GPRS support node (SGSN), and NSAPI debugging messages for IOS SLB.</td>
</tr>
<tr>
<td><strong>icmp</strong></td>
<td>Displays all Internet Control Message Protocol debugging messages for IOS SLB.</td>
</tr>
<tr>
<td><strong>natpool</strong></td>
<td>Displays debugging messages related to the IOS SLB client Network Address Translation (NAT) pool.</td>
</tr>
<tr>
<td><strong>probe</strong></td>
<td>Displays debugging messages related to probes.</td>
</tr>
<tr>
<td><strong>reals</strong></td>
<td>Displays debugging messages for all real servers defined to IOS SLB.</td>
</tr>
</tbody>
</table>
**debug ip slb**

**Defaults**
No default behavior or values

**Command Modes**
User EXEC or privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(2)E</td>
<td>The <strong>natpool</strong> and <strong>replication</strong> keywords were added.</td>
</tr>
<tr>
<td>12.1(3a)E</td>
<td>The <strong>firewallfarm</strong> keyword was added.</td>
</tr>
<tr>
<td>12.1(7)E</td>
<td>The <strong>vservers</strong> keyword was added.</td>
</tr>
<tr>
<td>12.1(9)E</td>
<td>The <strong>sessions</strong> keyword was added.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>The <strong>route</strong> keyword, the <strong>acl-number</strong> argument, and the <strong>radius</strong> option on the <strong>sessions</strong> keyword were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
<tr>
<td>12.1(13)E3</td>
<td>The <strong>gtp</strong> keyword and the <strong>gtp</strong> option on the <strong>sessions</strong> keyword were added.</td>
</tr>
<tr>
<td>12.2(14)ZA2</td>
<td>The <strong>ipmobile</strong> keyword was added.</td>
</tr>
</tbody>
</table>

**replication**
Displays debugging messages related to IOS SLB stateful backup virtual server.

**route**
Displays debugging messages for all routing handled by the IOS SLB RADIUS framed-IP sticky database.

**sessions [gtp | ipmobile | radius]**
Displays debugging messages for all sessions being handled by IOS SLB.

- The optional **gtp** keyword enables users to limit the information displayed to only GTP sessions.
- The optional **ipmobile** keyword enables users to limit the information displayed to only Mobile IP sessions.
- The optional **radius** keyword enables users to limit the information displayed to only RADIUS sessions.

**vservers**
Displays debugging messages for all virtual servers defined to IOS SLB.
This command displays debugging messages for IOS SLB.

**Caution**

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use `debug` commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use `debug` commands during periods of lower network flows and fewer users. Debugging during these periods reduces the effect these commands have on other users on the system.

**Examples**

The following example configures a debugging session to check all IP IOS SLB parameters:

```
Router# debug ip slb all
SLB All debugging is on
Router#
```

The following example stops all debugging:

```
Router# no debug all
All possible debugging has been turned off
Router#
```

The following example configures debugging to check IP IOS SLB replication used with stateful backup and displays the output from the send or transmit virtual server:

```
Router# debug ip slb replication
*Mar 2 08:02:38.019: SLB Replicate: (send) update vs: VS1 update_count 42
```
To change the amount of time IOS Server Load Balancing (IOS SLB) maintains TCP connection context after a connection has terminated, use the `delay` command in firewall farm TCP protocol configuration mode. To restore the default delay timer, use the `no` form of this command.

```
 delay duration

no delay
```

### Syntax Description

- **duration**
  
  Delay timer duration in seconds. The valid range is 1 to 600 seconds. The default value is 10 seconds.

### Defaults

The default duration is 10 seconds.

### Command Modes

Firewall farm TCP protocol configuration

### Command History

- **Release**
  
  **Modification**
  
  12.1(3a)E  
  This command was introduced.
  
  12.2(14)S  
  This command was integrated into Cisco IOS Release 12.2(14)S.

### Usage Guidelines

The delay timer allows out-of-sequence packets and final acknowledgments (ACKs) to be delivered after a TCP connection ends. Do not set this value to zero (0).

If you are configuring a delay timer for HTTP flows, choose a low number such as 5 seconds as a starting point.

### Examples

The following example specifies that IOS SLB maintains TCP connection context for 30 seconds after a connection has terminated:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol tcp
Router(config-slb-fw-tcp)# delay 30
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>protocol tcp</strong></td>
<td>Enters firewall farm TCP protocol configuration mode.</td>
</tr>
<tr>
<td><strong>show ip slb firewallfarm</strong></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
delay (virtual server)

To change the amount of time IOS Server Load Balancing (IOS SLB) maintains TCP connection context after a connection has terminated, use the `delay` command in virtual server configuration mode. To restore the default delay timer, use the `no` form of this command.

```
delay duration

no delay
```

**Syntax Description**

- **duration**
  - Delay timer duration in seconds. The valid range is 1 to 600 seconds.
  - The default value is 10 seconds.

**Defaults**

The default duration is 10 seconds.

**Command Modes**

Virtual server configuration

**Command History**

- **Release**
  - **Modification**
  - 12.0(7)XE: This command was introduced.
  - 12.1(5)T: This command was integrated into Cisco IOS Release 12.1(5)T.
  - 12.2: This command was integrated into Cisco IOS Release 12.2.
  - 12.2(14)S: This command was integrated into Cisco IOS Release 12.2(14)S.

**Usage Guidelines**

The delay timer allows out-of-sequence packets and final acknowledgments (ACKs) to be delivered after a TCP connection ends. Do not set this value to zero (0).

If you are configuring a delay timer for HTTP flows, choose a low number such as 5 seconds as a starting point.

For the Home Agent Director, the `delay` command has no meaning and is not supported.

**Examples**

The following example specifies that IOS SLB maintains TCP connection context for 30 seconds after a connection has terminated:

```
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# delay 30
```

**Related Commands**

- **Command**
  - `show ip slb vserver`
  - Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).
- **virtual (virtual server)**
  - Configures the virtual server attributes.
**expect**

To configure a status code or regular expression to expect information from the HTTP probe, use the `expect` command in HTTP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
 expect [status status-code] [regex expression]

 no expect [status status-code] [regex expression]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>status status-code</th>
<th>(Optional) Configures the expected HTTP status code. The valid range is 100 to 599. The default expected status code is 200.</th>
</tr>
</thead>
<tbody>
<tr>
<td>regex expression</td>
<td>(Optional) Configures the regular expression expected in the HTTP response.</td>
</tr>
</tbody>
</table>

**Defaults**

The default expected status code is 200.
There is no default expected regular expression.

**Command Modes**

HTTP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(3a)E</td>
<td>The <code>regex</code> keyword and <code>expression</code> argument were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `expect` command configures the expected status code or regular expression to be received from the servers. A real server is considered to have failed and is taken out of service if any of the following events occurs:

- A status number other than the expected one is received.
- The expected regular expression is not received in the first 2920 bytes of probe output. (IOS SLB searches only the first 2920 bytes for the expected status code or regular expression.)
- The server fails to respond.

For IOS SLB firewall load balancing, configure the HTTP probe to expect status code 401.

**Examples**

The following example configures an HTTP probe named PROBE2, enters HTTP configuration mode, and configures the HTTP probe to expect the status code 401 and the regular expression Copyright:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# expect status 401 regex Copyright
```
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb probe http</td>
<td>Configures an HTTP probe name and enters HTTP probe configuration mode.</td>
</tr>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
failaction (firewall farm)

To configure IOS Server Load Balancing (IOS SLB)’s behavior when a firewall fails, use the **failaction** command in firewall farm configuration mode.

failaction purge

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>purge</td>
<td>Enables IOS SLB to automatically remove connections to failed firewalls from the connection database even if the idle timers have not expired.</td>
</tr>
</tbody>
</table>

**Defaults**

If you do not specify the **failaction** command, IOS SLB does not automatically remove connections to failed firewalls.

**Command Modes**

Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(9)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is useful for applications that do not rotate the source port (such as Internet Key Exchange [IKE]), and for protocols that do not have ports to differentiate flows (such as Encapsulation Security Payload [ESP]).

**Examples**

In the following example, IOS SLB removes all connections to failed firewalls in firewall farm FIRE1:

```bash
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# failaction purge
```
**failaction (server farm)**

To configure IOS Server Load Balancing (IOS SLB)'s behavior when a real server fails, use the `failaction` command in server farm configuration mode. To restore the default settings, use the `no` form of this command.

```
failaction {purge | radius reassign}

no failaction {purge | radius reassign}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>purge</td>
<td>Enables IOS SLB to automatically remove connections to failed real servers from the connection database even if the idle timers have not expired.</td>
</tr>
<tr>
<td>radius reassign</td>
<td>Enables IOS SLB to automatically reassign to a new real server RADIUS sticky objects that are destined for a failed real server.</td>
</tr>
</tbody>
</table>

**Defaults**

If you do not specify the `failaction` command, IOS SLB does not automatically remove connections to failed real servers, nor does it reassign RADIUS sticky objects.

**Command Modes**

Server farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(9)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>The <code>radius reassign</code> keywords were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is useful for applications that do not rotate the source port (such as Internet Key Exchange [IKE]), and for protocols that do not have ports to differentiate flows (such as Encapsulation Security Payload [ESP]).

You can specify `no failaction purge`, but it has no effect on the connection database.

If you specify `failaction radius reassign`, IOS SLB reassigns RADIUS sticky objects without seeing any new RADIUS messages. The assumption is that, in the event of a failure, the RADIUS proxy gateways can handle user flows without seeing the RADIUS messages. If the RADIUS proxy gateways cannot do so, do not specify the `failaction radius reassign` command.

**Examples**

In the following example, IOS SLB removes all connections to failed real servers in server farm PUBLIC:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# failaction purge
```
faildetect (DNS probe)

To specify the conditions that indicate a server failure, use the `faildetect` command in DNS probe configuration mode. To restore the default values that indicate a server failure, use the `no` form of this command.

```
faildetect number-of-probes

no faildetect
```

**Syntax Description**

- `number-of-probes`: Number of consecutive unacknowledged Domain Name System (DNS) probes allowed before a real server is considered to have failed. Valid range is 1 to 65535. The default value is three (3) unacknowledged DNS probes.

**Defaults**

The default value is three (3) unacknowledged DNS probes.

**Command Modes**

DNS probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

In the following example the unacknowledged DNS probe threshold is set to 16:

```
Router(config)# ip slb probe PROBE4 dns
Router(config-slb-probe)# faildetect 16
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe dns</code></td>
<td>Configures a Domain Name System (DNS) probe name and enters DNS probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
faildetect (ping probe)

To specify the conditions that indicate a server failure, use the `faildetect` command in ping probe configuration mode. To restore the default values that indicate a server failure, use the `no` form of this command.

```
faildetect number-of-pings

no faildetect
```

**Syntax Description**

- **number-of-pings**
  
  Number of consecutive unacknowledged pings allowed before a real server is considered to have failed. Valid range is 1 to 65535. The default is ten (10) unacknowledged pings.

**Defaults**

The default value is ten (10) unacknowledged pings.

**Command Modes**

Ping probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

In the following example the unacknowledged ping threshold is set to 16:

```sh
Router(config)# ip slb probe PROBE1 ping
Router(config-slb-probe)# faildetect 16
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb probe ping</td>
<td>Configures a ping probe name and enters ping probe configuration mode.</td>
</tr>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
faildetect (real server)

To specify the conditions that indicate a real server failure, use the `faildetect` command in real server configuration mode. To restore the default values that indicate a server failure, use the `no` form of this command.

```
faildetect numconns number-of-conns [numclients number-of-clients]
no faildetect
```

### Syntax Description

<table>
<thead>
<tr>
<th>Numconns number-of-conns</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of consecutive connection failures allowed before IOS Server Load Balancing (IOS SLB) fails the real server. The valid range is 1 to 255. The default value is 8.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Numclients number-of-clients</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Optional) Number of unique client IP addresses that can experience connection failures before IOS SLB fails the real server. The valid range is 1 to 8. The default value is 2.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>If there is only one client in your network (for example, one serving GPRS support node [SGSN] in a general packet radio service [GPRS] load-balancing environment), then you must specify <code>numclients 1</code>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>In RADIUS load balancing, for automatic session-based failure detection, specify <code>numclients 1</code>.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Defaults

If you do not specify the `faildetect` command, the default value of the connection failure threshold is 8. If you specify the `faildetect` command but do not specify the `numclients` keyword, the default value of the client connection failure threshold is 2.

### Command Modes

Real server configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(9)E</td>
<td>This command was modified to support GPRS load balancing.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

IOS SLB does not fail the real server until both of the following conditions are met:

- There have been `number-of-conns` consecutive connection failures.
- There have been `number-of-clients` unique client connection failures.

That is, there can be many consecutive connection failures, but until there have also been `number-of-clients` unique client connection failures, IOS SLB does not fail the real server.
Similarly, there can be many unique client connection failures, but until there have also been \textit{number-of-conns} consecutive connection failures, IOS SLB does not fail the real server.

GPRS load balancing has the following features:

- The \texttt{numconns} keyword specifies the number of consecutive Create Packet Data Protocol (PDP) requests allowed before IOS SLB fails the gateway GPRS support node (GGSN).
- The \texttt{numclients} keyword specifies the number of unique client Create PDP request failures allowed before IOS SLB fails the GGSN.

\textbf{Examples}

In the following example, the \texttt{numconns} keyword is set to 10 and the \texttt{numclients} keyword is set to 3:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1
Router(config-slb-real)# faildetect numconns 10 numclients 3
```

With those settings, IOS SLB will not fail the real server until there have been ten (10) consecutive connection failures and there have been three (3) unique client connection failures.

\textbf{Related Commands}

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{real (server farm)}</td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>\texttt{show ip slb reals }</td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td>\texttt{show ip slb serverfarms}</td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
hand-off radius

To change 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, use the `hand-off radius` command in virtual server configuration mode. To restore the default hand-off timer, use the `no` form of this command.

`hand-off radius duration`

`no hand-off radius`

**Syntax Description**

<table>
<thead>
<tr>
<th>duration</th>
<th>Hand-off timer duration in seconds. The valid range is 1 to 43200 seconds.</th>
</tr>
</thead>
</table>

**Defaults**

No default behavior or values

**Command Modes**

Virtual server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.2(14)ZA2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The hand-off radius timer is valid only for RADIUS virtual servers that have the `service radius` keywords specified on the `virtual` command.

**Examples**

The following example specifies that IOS SLB waits for 30 seconds after a foreign agent hand-off:

```
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# hand-off radius 30
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td><code>virtual (virtual server)</code></td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
header

To configure the basic authentication values for the HTTP probe, use the header command in HTTP probe configuration mode. To remove a header HTTP probe configuration, use the no form of this command.

```
header field-name [field-value]

no header field-name [field-value]
```

**Syntax Description**

- **field-name**: Configures the name of the HTTP probe header. The character string is limited to 15 characters.
- **field-value**: (Optional) Configures the value of the HTTP probe header.

**Defaults**

The following headers are inserted in the request by default:

- Accept: */*
- Connection: close
- User-Agent: cisco-slb-probe/1.0
- Host: virtual IP address

**Command Modes**

HTTP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The header command in HTTP probe configuration mode configures the name and value parameters of the header.

**Note**

The colon (:) separating the field name and field value is automatically inserted if not provided. Multiple headers with the same name are not supported.

**Examples**

The following example configures an HTTP probe named PROBE2, enters HTTP configuration mode, and configures the HTTP probe header name as HeaderName and value as HeaderValue:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# header HeaderName HeaderValue
```
<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>ip slb probe http</code></td>
<td>Configures an HTTP probe name and enters HTTP probe configuration mode.</td>
</tr>
<tr>
<td></td>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
idle (firewall farm datagram protocol)

To specify the minimum time IOS Server Load Balancing (IOS SLB) maintains connection information in the absence of packet activity, use the idle command in firewall farm datagram protocol configuration mode. To restore the default idle duration value, use the no form of this command.

idle duration

no idle

Syntax Description

duration

Idle connection timer duration in seconds. Valid values range from 10 to 65535 seconds. The default is 3600 seconds (1 hour).

Defaults

The default idle duration is 3600 seconds.

Command Modes

Firewall farm datagram protocol configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

Examples

The following example instructs IOS SLB to maintain connection information for an idle connection for 120 seconds:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol datagram
Router(config-slb-fw-udp)# idle 120
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>protocol datagram</td>
<td>Enters firewall farm datagram protocol configuration mode.</td>
</tr>
<tr>
<td>show ip slb firewallfarm</td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
idle (firewall farm TCP protocol)

To specify the minimum time IOS Server Load Balancing (IOS SLB) maintains connection information in the absence of packet activity, use the `idle` command in firewall farm TCP protocol configuration mode. To restore the default idle duration value, use the `no idle` form of this command.

```
idle duration

no idle
```

**Syntax Description**

| duration | Idle connection timer duration in seconds. Valid values range from 10 to 65535 seconds. The default is 3600 seconds (1 hour). |

**Defaults**

The default idle duration is 3600 seconds.

**Command Modes**

Firewall farm TCP protocol configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If a client sends a TCP packet that is not a sequence number (SYN) or reset (RST) packet, and IOS SLB does not have a TCP connection object in its table (possibly due to expiration of the idle timer), IOS SLB sends a TCP RST to the client.

If you are configuring an idle timer for HTTP flows, choose a low number such as 120 seconds as a starting point. A low number ensures that the IOS SLB connection database maintains a manageable size if problems at the server, client, or network result in a large number of connections. However, do not choose a value under 60 seconds; such a low value can reduce the efficiency of IOS SLB.

**Examples**

The following example instructs IOS SLB to maintain connection information for an idle connection for 120 seconds:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol tcp
Router(config-slb-fw-tcp)# idle 120
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protocol tcp</code></td>
<td>Enters firewall farm TCP protocol configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
idle (virtual server)

To specify the minimum time IOS Server Load Balancing (IOS SLB) maintains connection information in the absence of packet activity, use the `idle` command in virtual server configuration mode. To restore the default idle duration value, use the `no` form of this command.

```
idle [gtp request | ipmobile request | radius {request | framed-ip}] duration
```

```
no idle [gtp request | ipmobile request | radius {request | framed-ip}] duration
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>gtp request</code></td>
<td>(Optional) For general packet radio service (GPRS) Tunneling Protocol (GTP) cause code inspection, configures the duration for Packet Data Protocol (PDP) context create, update, or delete request messages to a real gateway GPRS support node (GGSN) to go unanswered, before IOS SLB cleans up the session object.</td>
</tr>
<tr>
<td><code>ipmobile request</code></td>
<td>(Optional) For Home Agent Director, configures the duration for IOS SLB to wait for a Mobile IP Registration Request (RRQ), before IOS SLB cleans up the session object.</td>
</tr>
<tr>
<td><code>radius request</code></td>
<td>(Optional) Configures the duration for RADIUS entries in the IOS SLB session database.</td>
</tr>
<tr>
<td><code>radius framed-ip</code></td>
<td>(Optional) Configures the duration for entries in the IOS SLB RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td><code>duration</code></td>
<td>Idle connection timer duration in seconds. Valid values range from 4 to 65535 seconds. The default values are:</td>
</tr>
<tr>
<td></td>
<td>• 10 seconds in the Home Agent Director.</td>
</tr>
<tr>
<td></td>
<td>• 30 seconds in general packet radio service (GPRS) load balancing.</td>
</tr>
<tr>
<td></td>
<td>• 30 seconds for RADIUS entries in the IOS SLB session database.</td>
</tr>
<tr>
<td></td>
<td>• 7200 seconds for entries in the IOS SLB RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td></td>
<td>• 3600 seconds (1 hour) in all other environments.</td>
</tr>
</tbody>
</table>

### Defaults

The default idle duration is:

- 10 seconds in the Home Agent Director
- 30 seconds in GPRS load balancing
- 30 seconds for RADIUS entries in the IOS SLB session database
- 7200 seconds for entries in the IOS SLB RADIUS framed-IP sticky database
- 3600 seconds (1 hour) in all other environments

### Command Modes

Virtual server configuration
If a client sends a TCP packet that is not a sequence number (SYN) or reset (RST) packet, and IOS SLB does not have a TCP connection object in its table (possibly due to expiration of the idle timer), IOS SLB sends a TCP RST to the client.

If you are configuring an idle timer for HTTP flows, choose a low number such as 120 seconds as a starting point. A low number ensures that the IOS SLB connection database maintains a manageable size if problems at the server, client, or network result in a large number of connections. However, do not choose a value under 60 seconds (except in GPRS load balancing); such a low value can reduce the efficiency of IOS SLB.

In most environments, the idle timer times out data paths. However, in GPRS load balancing, it times out the session context for signaling paths (not data paths).

In GPRS load balancing without GTP cause code inspection enabled, you must specify an idle timer greater than the longest possible interval between PDP context requests on the serving GPRS support node (SGSN). The longest interval can be expressed using the following algorithm:

Longest interval = T3 x 2(N3-2)

where T3 is the SGSN’s T3-RESPONSE counter value and N3 is the SGSN’s N3-REQUESTS counter value.

For example, if the T3-RESPONSE counter value is 3 and the N3-REQUESTS counter value is 6, then:

Longest interval = 3 x 2(6-2) = 3 x 2(4) = 3 x 16 = 48 seconds

Given those values, you must specify an idle timer of at least 49 seconds.

The following example instructs IOS SLB to maintain connection information for an idle connection for 120 seconds:

Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# idle 120
inservice (firewall farm)

To enable the firewall farm for use by IOS Server Load Balancing (IOS SLB), use the `inservice` command in firewall farm configuration mode. To remove the firewall farm from service, use the `no` form of this command.

```
inservice [standby group-name]
```

```
no inservice [standby group-name]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>standby</code></td>
<td>(Optional) Configures the Hot Standby Router Protocol (HSRP) standby firewall farm for use with stateless and stateful backup.</td>
</tr>
<tr>
<td><code>group-name</code></td>
<td>(Optional) HSRP group name with which the IOS SLB firewall farm is associated.</td>
</tr>
</tbody>
</table>

**Defaults**

The firewall farm is defined to IOS SLB but is not used.

**Command Modes**

Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When you use the `no` form of this command to remove a firewall farm from service, the firewall farm acquiesces gracefully. No new connections are assigned, and existing connections are allowed to complete.

**Examples**

In the following example, the firewall farm is enabled for use by the IOS SLB feature:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# inservice
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb firewallfarm</code></td>
<td>Identifies a firewall by IP address farm and enters firewall farm configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
inservice (firewall farm real server)

To enable the firewall for use by IOS Server Load Balancing (IOS SLB), use the `inservice` command in firewall farm real server configuration mode. To remove the firewall from service, use the `no` form of this command.

```
inservice

no inservice
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

The firewall is defined to IOS SLB but is not used.

**Command Modes**

Firewall farm real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>T2.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>T2.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

IOS SLB firewall load balancing uses probes to detect failures. Therefore, if you have not configured a probe, the firewall is not placed in service.

When you use the `no` form of this command to remove a firewall from service, the firewall acquiesces gracefully. No new connections are assigned, and existing connections are allowed to complete.

**Examples**

In the following example, the firewall is enabled for use by the IOS SLB feature:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# real 10.10.1.1
Router(config-slb-fw-real)# inservice
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>real (firewall farm)</code></td>
<td>Identifies a firewall by IP address as a member of a firewall farm and enters real server configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td><code>show ip slb reals</code></td>
<td>Displays information about the real servers.</td>
</tr>
</tbody>
</table>
inservice (server farm real server)

To enable the real server for use by IOS Server Load Balancing (IOS SLB), use the `inservice` command in server farm real server configuration mode. To remove the real server from service, use the `no` form of this command.

```
inservice
no inservice
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

The real server is defined to IOS SLB but is not used.

**Command Modes**

Server farm real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When you use the `no` form of this command to remove a real server from service, the real server acquiesces gracefully. No new connections are assigned, and existing connections are allowed to complete.

**Examples**

In the following example, the real server is enabled for use by the IOS SLB feature:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1
Router(config-slb-sfarm-real)# inservice
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>real (server farm)</code></td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb reals</code></td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td><code>show ip slb serverfarms</code></td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
inservice (server farm virtual server)

To enable the virtual server for use by IOS Server Load Balancing (IOS SLB), use the `inservice` command in server farm virtual server configuration mode. To remove the virtual server from service, use the `no` form of this command.

```
inservice [standby group-name]
no inservice [standby group-name]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>standby</td>
<td>(Optional) Configures the Hot Standby Router Protocol (HSRP) standby virtual server for use with stateless and stateful backup.</td>
</tr>
<tr>
<td>group-name</td>
<td>(Optional) HSRP group name with which the IOS SLB virtual server is associated.</td>
</tr>
</tbody>
</table>

**Defaults**
The virtual server is defined to IOS SLB but is not used.

**Command Modes**
Server farm virtual server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(1)E</td>
<td>The <code>standby</code> keyword and <code>group-name</code> argument were added.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
When you use the `no` form of this command to remove a virtual server from service, the virtual server acquiesces gracefully. No new connections are assigned, and existing connections are allowed to complete.

**Examples**
In the following example, the virtual server is enabled for use by the IOS SLB feature:

```
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# inservice
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td><code>virtual (virtual server)</code></td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
interval (custom UDP probe)

To configure a custom User Datagram Protocol (UDP) probe interval, use the `interval` command in custom UDP probe configuration mode. To remove a custom UDP probe `interval` configuration, use the `no` form of this command.

```
interval seconds

no interval seconds
```

### Syntax Description

**seconds**

Number of seconds to wait before reattempting the probe. Valid values range from 1 to 65535 seconds. The default interval is 10 seconds.

### Defaults

The default custom UDP probe interval value is 10 seconds.

### Command Modes

Custom UDP probe configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Examples

The following example configures a custom UDP probe named PROBE6, enters custom UDP configuration mode, and configures the custom UDP probe timer interval to send every 11 seconds:

```
Router(config)# ip slb probe PROBE6 tcp
Router(config-slb-probe)# interval 11
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe custom udp</code></td>
<td>Configures a custom User Datagram Protocol (UDP) probe name and enters custom UDP probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
interval (DNS probe)

To configure a DNS probe interval, use the `interval` command in Domain Name System (DNS) probe configuration mode. To remove a DNS probe `interval` configuration, use the `no` form of this command.

```
interval seconds

no interval seconds
```

**Syntax Description**
- `seconds` Number of seconds to wait before reattempting the probe. Valid values range from 1 to 65535 seconds. The default interval is 10 seconds.

**Defaults**
The default DNS probe interval value is 10 seconds.

**Command Modes**
DNS probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**
The following example configures a DNS probe named PROBE4, enters DNS configuration mode, and configures the DNS probe timer interval to send every 11 seconds:

```
Router(config)# ip slb probe PROBE4 dns
Router(config-slb-probe)# interval 11
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe dns</code></td>
<td>Configures a Domain Name System (DNS) probe name and enters DNS probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
interval (HTTP probe)

To configure an HTTP probe interval, use the `interval` command in HTTP probe configuration mode. To remove an HTTP probe `interval` configuration, use the `no` form of this command.

```
interval seconds
```

```
no interval seconds
```

**Syntax Description**

| seconds | Number of seconds to wait before reattempting the probe. Valid values range from 1 to 65535 seconds. The default interval is 8 seconds. |

**Defaults**

The default HTTP probe interval value is 8 seconds.

**Command Modes**

HTTP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures an HTTP probe named PROBE2, enters HTTP configuration mode, and configures the HTTP probe timer interval to send every 11 seconds:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# interval 11
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe http</code></td>
<td>Configures an HTTP probe name and enters HTTP probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
interval (ping probe)

To configure a ping probe interval, use the interval command in ping probe configuration mode. To remove a ping probe interval configuration, use the no form of this command.

```
interval seconds

no interval seconds
```

**Syntax Description**
- `seconds`  
  Number of seconds to wait before reattempting the probe. Valid values range from 1 to 65535 seconds. The default interval is 1 second.

**Defaults**
The default ping probe interval value is 1 second.

**Command Modes**
Ping probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a ping probe named PROBE1, enters ping configuration mode, and configures the ping probe timer interval to send every 11 seconds:

```
Router(config)# ip slb probe PROBE1 ping
Router(config-slb-probe)# interval 11
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ip slb probe ping</strong></td>
<td>Configures a ping probe name and enters ping probe configuration mode.</td>
</tr>
<tr>
<td><strong>show ip slb probe</strong></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
To configure a TCP probe interval, use the `interval` command in TCP probe configuration mode. To remove a TCP probe interval configuration, use the `no` form of this command.

```
interval seconds

no interval seconds
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>seconds</th>
<th>Number of seconds to wait before reattempting the probe. Valid values range from 1 to 65535 seconds. The default interval is 10 seconds.</th>
</tr>
</thead>
</table>

**Defaults**

The default TCP probe interval value is 10 seconds.

**Command Modes**

TCP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a TCP probe named PROBE5, enters TCP configuration mode, and configures the TCP probe timer interval to send every 11 seconds:

```
Router(config)# ip slb probe PROBE5 tcp
Router(config-slb-probe)# interval 11
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb probe tcp</td>
<td>Configures a TCP probe name and enters TCP probe configuration mode.</td>
</tr>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
interval (WSP probe)

To configure a Wireless Session Protocol (WSP) probe interval, use the **interval** command in WSP probe configuration mode. To remove a WSP probe **interval** configuration, use the **no** form of this command.

```
interval seconds

no interval seconds
```

**Syntax Description**

| seconds | Number of seconds to wait before reattempting the probe. Valid values range from 1 to 65535 seconds. The default interval is 8 seconds. |

**Defaults**

The default WSP probe interval value is 8 seconds.

**Command Modes**

WSP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(5a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a ping probe named PROBE3, enters WSP probe configuration mode, and configures the WSP probe timer interval to send every 11 seconds:

```
Router(config)# ip slb probe PROBE3 wsp
Router(config-slb-probe)# interval 11
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb probe wsp</td>
<td>Configures a Wireless Session Protocol (WSP) probe name and enters WSP probe configuration mode.</td>
</tr>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
To configure Dynamic Feedback Protocol (DFP), supply an optional password, and enter DFP configuration mode, use the `ip slb dfp` command in global configuration mode. To remove the DFP configuration, use the `no` form of this command.

```
ip slb dfp [password [0 | 7] password [timeout]]
no ip slb dfp
```

### Syntax Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>password</code></td>
<td>(Optional) Password for Message Digest Algorithm Version 5 (MD5) authentication.</td>
</tr>
<tr>
<td><code>0</code></td>
<td>(Optional) Password is unencrypted. This value is the default setting.</td>
</tr>
<tr>
<td><code>7</code></td>
<td>(Optional) Password is encrypted.</td>
</tr>
<tr>
<td><code>password</code></td>
<td>(Optional) Password value for MD5 authentication. This password must match the password configured on the host agent.</td>
</tr>
<tr>
<td><code>timeout</code></td>
<td>(Optional) Delay period, in seconds, during which both the old password and the new password are accepted. The valid range is 0 to 65535 seconds. The default value is 180 seconds, if a password is specified.</td>
</tr>
</tbody>
</table>

### Defaults

The default password encryption is 0 (unencrypted).
The default password timeout is 180 seconds, if a password is specified.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(3a)E</td>
<td>The <code>0</code> and <code>7</code> keywords were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The password specified in the `ip slb dfp` command for the DFP manager must match the password specified in the `password` command for the DFP agent.

The timeout option allows you to change the password without stopping messages between the DFP agent and its manager. The default value is 180 seconds.

During the timeout, the agent sends packets with the old password (or null, if there is no old password), and receives packets with either the old or new password. After the timeout expires, the agent sends and receives packets only with the new password; received packets that use the old password are discarded.
If you are changing the password for an entire load-balanced environment, set a longer timeout to allow enough time for you to update the password on all agents and servers before the timeout expires. Setting a longer timeout also prevents mismatches between agents and servers that have begun running the new password and agents, and servers on which you have not yet changed the old password.

**Examples**

The following example configures DFP, sets the DFP password to Password1 and the timeout to 360 seconds, and enters DFP configuration mode:

```
Router(config)# ip slb dfp password Password1 360
Router(config-slb-dfp)#
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agent (DFP)</td>
<td>Identifies a Dynamic Feedback Protocol (DFP) agent to which IOS Server Load Balancing (IOS SLB) can connect.</td>
</tr>
<tr>
<td>ip dfp agent</td>
<td>Identifies a Dynamic Feedback Protocol (DFP) agent subsystem and enters DFP agent configuration mode.</td>
</tr>
</tbody>
</table>
To configure an initial allocation and a maximum value for IOS Server Load Balancing (IOS SLB) database entries, use the **ip slb entries** command in global configuration mode. To restore the default values, use the **no** form of this command.

```
ip slb entries [conn [init-conn [max-conn]] | frag [init-frag [max-frag]] | lifetime timeout] | sticky [init-sticky [max-sticky]]]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>conn</td>
<td>(Optional) Configures an initial allocation and a maximum value for IOS SLB connection database entries.</td>
<td></td>
</tr>
<tr>
<td>init-conn</td>
<td>(Optional) Initial allocation of connection database entries. When the number of available entries is reduced to less than half of the <em>init-conn</em> argument, IOS SLB begins allocating additional entries. The number of entries can grow dynamically up to the number specified by the <em>max-conn</em> argument.</td>
<td>Be careful when setting the <em>init-conn</em> argument to a very high value, such as 1000000, because IOS SLB immediately allocates those entries, which can cause the router or switch to pause indefinitely. Start with a lower value, such as 125000.</td>
</tr>
<tr>
<td>max-conn</td>
<td>(Optional) Maximum number of connection database entries that can be allocated.</td>
<td>Valid range is 1 to 1000000 connection database entries. The default is 8000 connection database entries.</td>
</tr>
<tr>
<td>frag</td>
<td>(Optional) Configures an initial allocation and a maximum value for IOS SLB fragment database entries.</td>
<td></td>
</tr>
<tr>
<td>init-frag</td>
<td>(Optional) Initial allocation of routing entries in the fragment database. When the number of available entries is reduced to less than half of the <em>init-frag</em> argument, IOS SLB begins allocating additional entries. The number of entries can grow dynamically up to the number specified by the <em>max-frag</em> argument.</td>
<td>Be careful when setting the <em>init-frag</em> argument to a very high value, such as 1000000, because IOS SLB immediately allocates those entries, which can cause the router or switch to pause indefinitely. Start with a lower value, such as 125000.</td>
</tr>
<tr>
<td>max-frag</td>
<td>(Optional) Maximum number of fragment database entries that can be allocated.</td>
<td>Valid range is 1 to 8000000 fragment database entries. The default is 32000 fragment database entries.</td>
</tr>
</tbody>
</table>
### ip slb entries

| lifetime | (Optional) Lifetime of an entry in the IOS SLB fragment database, in seconds.
| sticky | (Optional) Configures an initial allocation and a maximum value for IOS SLB sticky connection database entries.
| init-sticky | (Optional) Initial allocation of sticky database entries. When the number of available entries is reduced to less than half of the init-sticky argument, IOS SLB begins allocating additional entries. The number of entries can grow dynamically up to the number specified by the max-sticky argument.
| max-sticky | (Optional) Maximum number of sticky database entries that can be allocated. Valid range is 1 to 8000000 sticky database entries. The default is 8000000 sticky database entries.

### Defaults

For connections, the default initial allocation is 8000 connections, and the default maximum is 8000000 connections.
For fragments, the default initial allocation is 2000 fragments, and the default maximum is 8000000 fragments. The default lifetime is 10 seconds.
For sticky connections, the default initial allocation is 4000 sticky connections, and the default maximum is 3200 sticky connections.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>The lifetime keyword and timeout argument were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

If you configure an initial allocation value that exceeds the amount of available memory, memory might not be available for other features. In extreme cases, the router or switch might not boot properly. Therefore, be careful when you configure initial allocation values.

### Examples

The following example configures an initial allocation of 128,000 connections, which can grow dynamically to a limit of 512,000 connections:

```
Router(config)# ip slb entries conn 128000 512000
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb conns</td>
<td>Displays all connections handled by IOS Server Load Balancing (IOS SLB), or, optionally, only those connections associated with a particular virtual server or client.</td>
</tr>
</tbody>
</table>
**ip slb firewallfarm**

To identify a firewall farm and enter firewall farm configuration mode, use the `ip slb firewallfarm` command in global configuration mode. To remove the firewall farm from the IOS Server Load Balancing (IOS SLB) configuration, use the `no` form of this command.

```
ip slb firewallfarm firewall-farm
no ip slb firewallfarm firewall-farm
```

**Syntax Description**

| firewall-farm | Character string used to identify the firewall farm. The character string is limited to 15 characters. |

**Defaults**

No default behavior or values

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Grouping real servers into firewall farms is an essential part of IOS SLB firewall load balancing. Using firewall farms enables IOS SLB to assign new connections to the real servers based on their weighted capacities, and on the load-balancing algorithms used.

**Examples**

The following example identifies a firewall farm named FIRE1:

```
Router(config)# ip slb firewallfarm FIRE1
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real (firewall farm)</td>
<td>Identifies a firewall by IP address as a member of a firewall farm and enters real server configuration mode.</td>
</tr>
</tbody>
</table>
To configure the maximum number of buffers for the IOS Server Load Balancing (IOS SLB) fragment database, use the `ip slb maxbuffers frag` command in global configuration mode. To restore the default setting, use the `no` form of this command.

```
ip slb maxbuffers frag buffers
no ip slb maxbuffers frag
```

### Syntax Description

- **buffers**: Maximum number of out-of-order trailing fragments to be buffered simultaneously in the IOS SLB fragment database, waiting for the leader fragment. This value can help prevent IOS SLB memory from being overrun in the event of a fragment attack.

  Valid range is 0 to 65535 buffers. The default value is 100 buffers.

### Defaults

The default maximum is 100 buffers.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Examples

The following example sets the maximum number of buffers for the IOS SLB fragment buffer to 300:

```
Router(config)# ip slb maxbuffers frag 300
```
ip slb natpool

To configure an IOS Server Load Balancing (IOS SLB) Network Address Translation (NAT) to create at least one client address pool, use the `ip slb natpool` command in global configuration mode. To remove an `ip slb natpool` configuration, use the `no` form of this command.

```
ip slb natpool pool start-ip end-ip [netmask netmask | prefix-length leading-1-bits] [entries init-address [max-address]]
no ip slb natpool pool
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>pool</code></td>
<td>Character string used to identify this client address pool. The character string is limited to 15 characters.</td>
</tr>
<tr>
<td><code>start-ip</code></td>
<td>Starting IP address that defines the range of addresses in the address pool.</td>
</tr>
<tr>
<td><code>end-ip</code></td>
<td>Ending IP address that defines the range of addresses in the address pool.</td>
</tr>
<tr>
<td><code>netmask</code></td>
<td>(Optional) Configures the mask for the associated IP subnet. Specifications how many bits of the netmask are ones (that is, how many bits of the address indicate the network).</td>
</tr>
<tr>
<td><code>prefix-length leading-1-bits</code></td>
<td>Specifies how many bits of the netmask are ones (that is, how many bits of the address indicate the network).</td>
</tr>
<tr>
<td><code>entries</code></td>
<td>(Optional) Configures an initial allocation and optional maximum value for IOS SLB client NAT address entries for the <code>pool</code> argument.</td>
</tr>
<tr>
<td><code>init-address</code></td>
<td>(Optional) Initial allocation of client NAT address entries. The number of client NAT address entries can grow dynamically: When the number of available client NAT address entries is less than half of the init-address argument, IOS SLB allocates additional client NAT address entries.</td>
</tr>
<tr>
<td><code>max-address</code></td>
<td>(Optional) Maximum number of client NAT address entries that can be allocated. Valid range is 1 to 8000 client NAT address entries. The default is 8000 client NAT address entries.</td>
</tr>
</tbody>
</table>

**Defaults**

The default initial allocation is 8000 client NAT address entries. The default maximum number of client NAT address entries that can be allocated is the maximum number of ports that can be allocated within the IP address range.

**Command Modes**

Global configuration
If you want to use client NAT, you must create at least one client address pool.

The range of IP addresses in the address pool, configured with the `start-ip` and `end-ip` arguments, must not overlap the IP address for a VLAN as specified on the `ip address` interface configuration command.

The following example configures an IOS SLB NAT server farm pool of addresses with the name `web-clients`, the IP address range from 128.3.0.1 to 128.3.0.254, and a subnet mask of 255.255.0.0:

```
Router(config)# ip slb natpool web-clients 128.3.0.1 128.3.0.254 netmask 255.255.0.0
```
ip slb probe custom udp

To configure a custom User Datagram Protocol (UDP) probe name and enter custom UDP probe configuration mode, use the `ip slb probe custom udp` command in global configuration mode. To remove an `ip slb probe custom udp` configuration, use the `no` form of this command.

```
ip slb probe probe custom udp
no ip slb probe probe
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>probe</code></td>
<td>Name of the custom UDP probe. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

**Defaults**

No custom UDP probe is configured.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command configures the custom UDP probe name and application protocol and enters custom UDP configuration mode.

The custom UDP probe cannot be unconfigured while it is being used by the server farm or firewall farm.

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.

**Examples**

The following example configures an IOS Server Load Balancing (IOS SLB) probe named PROBE6, then enters custom UDP probe configuration mode:

```
Router(config)# ip slb probe PROBE6 udp
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>address (custom UDP probe)</strong></td>
<td>Configures an IP address to which to send custom User Datagram Protocol (UDP) probes.</td>
</tr>
<tr>
<td><strong>interval (custom UDP probe)</strong></td>
<td>Configures a custom User Datagram Protocol (UDP) probe interval.</td>
</tr>
<tr>
<td><strong>port (custom UDP probe)</strong></td>
<td>Specifies the port to which a custom User Datagram Protocol (UDP) probe is to connect.</td>
</tr>
<tr>
<td><strong>request (custom UDP probe)</strong></td>
<td>Defines the payload of the User Datagram Protocol (UDP) request packet to be sent by a custom UDP probe.</td>
</tr>
<tr>
<td><strong>response</strong></td>
<td>Defines the data string to match against custom User Datagram Protocol (UDP) probe response packets.</td>
</tr>
<tr>
<td><strong>show ip slb probe</strong></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
ip slb probe dns

To configure a Domain Name System (DNS) probe name and enter DNS probe configuration mode, use the `ip slb probe dns` command in global configuration mode. To remove an `ip slb probe dns` configuration, use the `no` form of this command.

```
   ip slb probe probe dns
   no ip slb probe probe
```

Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>probe</code></td>
<td>Name of the DNS probe. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

Defaults

No DNS probe is configured.

Command Modes

Global configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

Usage Guidelines

DNS probes send domain name resolve requests to real servers and verify the returned IP addresses.

This command configures the DNS probe name and application protocol and enters DNS configuration mode.

The DNS probe cannot be unconfigured while it is being used by the server farm or firewall farm.

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.

Examples

The following example configures an IOS Server Load Balancing (IOS SLB) probe named PROBE4, then enters DNS probe configuration mode:

```
Router(config)# ip slb probe PROBE4 dns
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
ip slb probe http

To configure an HTTP probe name and enter HTTP probe configuration mode, use the `ip slb probe http` command in global configuration mode. To remove an `ip slb probe http` configuration, use the `no` form of this command.

```
ip slb probe probe http
no ip slb probe probe
```

**Syntax Description**

<table>
<thead>
<tr>
<th><code>probe</code></th>
<th>Name of the HTTP probe. The character string is limited to 15 characters.</th>
</tr>
</thead>
</table>

**Defaults**

No HTTP probe is configured.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command configures the HTTP probe name and application protocol and enters HTTP configuration mode.

The HTTP probe cannot be unconfigured while it is being used by the server farm or firewall farm.

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.

**Note**

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).

**Examples**

The following example configures an IOS Server Load Balancing (IOS SLB) probe named PROBE2, then enters HTTP probe configuration mode:

```
Router(config)# ip slb probe PROBE2 http
```
show ip slb probe

Displays information about an IOS Server Load Balancing (IOS SLB) probe.
ip slb probe ping

To configure a ping probe name and enter ping probe configuration mode, use the `ip slb probe ping` command in global configuration mode. To remove an `ip slb probe ping` configuration, use the `no` form of this command.

```
  ip slb probe probe ping
  no ip slb probe probe
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>probe</code></td>
<td>Name of the ping probe. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

**Defaults**

No ping probe is configured.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command configures the ping probe name and application protocol and enters ping configuration mode.

The ping probe cannot be unconfigured while it is being used by the server farm or firewall farm.

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.

**Examples**

The following example configures an IOS Server Load Balancing (IOS SLB) probe named PROBE1, then enters ping probe configuration mode:

```
Router(config)# ip slb probe PROBE1 ping
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
ip slb probe tcp

To configure a TCP probe name and enter TCP probe configuration mode, use the `ip slb probe tcp` command in global configuration mode. To remove an `ip slb probe tcp` configuration, use the `no` form of this command.

```
ip slb probe probe tcp
no ip slb probe probe
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>probe</code></td>
<td>Name of the TCP probe. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

### Defaults

No TCP probe is configured.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command configures the TCP probe name and application protocol and enters TCP configuration mode.

The TCP probe cannot be unconfigured while it is being used by the server farm or firewall farm.

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.

### Examples

The following example configures an IOS Server Load Balancing (IOS SLB) probe named PROBE5, then enters TCP probe configuration mode:

```
Router(config)# ip slb probe PROBE5 tcp
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
**ip slb probe wsp**

To configure a Wireless Session Protocol (WSP) probe name and enter WSP probe configuration mode, use the `ip slb probe wsp` command in global configuration mode. To remove an `ip slb probe wsp` configuration, use the `no` form of this command.

```
ip slb probe probe wsp
no ip slb probe probe
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>probe</code></td>
<td>Name of the WSP probe. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

**Defaults**

No WSP probe is configured.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(5a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command configures the WSP probe name and application protocol and enters WSP probe configuration mode.

The WSP probe cannot be unconfigured while it is being used by the server farm or firewall farm.

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.

**Examples**

The following example configures an IOS Server Load Balancing (IOS SLB) probe named PROBE3, then enters WSP probe configuration mode:

```
Router(config)# ip slb probe PROBE3 wsp
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
ip slb route

To enable 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, use the `ip slb route` command in global configuration mode. To route packets normally, use the `no` form of this command.

```
ip slb route {ip-address netmask framed-ip | inter-firewall}
no ip slb route {ip-address netmask framed-ip | inter-firewall}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>(Optional) IP address of packets to be inspected.</td>
</tr>
<tr>
<td>netmask</td>
<td>(Optional) Subnet mask specifying a range of packets to be inspected.</td>
</tr>
<tr>
<td>framed-ip</td>
<td>(Optional) Packets are to be routed using the IOS SLB RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td>inter-firewall</td>
<td>(Optional) Enables IOS SLB to route packets from one firewall real server back through another firewall real server, if the flows to the destination IP would otherwise have been firewall load-balanced. This can be done within the same firewall farm or across different firewall farms.</td>
</tr>
</tbody>
</table>

### Defaults

IOS SLB cannot route packets using the RADIUS framed-IP sticky database, nor can it route packets from one firewall real server back through another firewall real server.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
<tr>
<td>12.1(13)E3</td>
<td>The <code>inter-firewall</code> keyword was added.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command enables IOS SLB to inspect packets whose source IP addresses match the specified IP address and subnet mask. IOS SLB then searches for the packet’s source IP address in the RADIUS framed-IP sticky database. If the database contains a matching entry, IOS SLB routes the packet to the associated real server. If the database does not contain a matching entry, IOS SLB routes the packet normally.

### Examples

The following example enables IOS SLB to inspect packets with the source IP address 10.10.10.1:

```
Router(config)# ip slb route 10.10.10.1 255.255.255.255 framed-ip
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb sticky</code></td>
<td>Displays the IOS Server Load Balancing (IOS SLB) sticky database.</td>
</tr>
</tbody>
</table>
ip slb serverfarm

To identify a server farm and enter server farm configuration mode, use the `ip slb serverfarm` command in global configuration mode. To remove the server farm from the IOS Server Load Balancing (IOS SLB) configuration, use the `no` form of this command.

```
ip slb serverfarm server-farm
no ip slb serverfarm server-farm
```

### Syntax Description

<table>
<thead>
<tr>
<th><code>server-farm</code></th>
<th>Character string used to identify the server farm. The character string is limited to 15 characters.</th>
</tr>
</thead>
</table>

### Defaults

No server farm is identified.

### Command Modes

- **Global configuration**

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Grouping real servers into server farms is an essential part of IOS SLB. Using server farms enables IOS SLB to assign new connections to the real servers based on their weighted capacities, and on the load-balancing algorithms used.

### Examples

The following example identifies a server farm named PUBLIC:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)#?
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>real (server farm)</code></td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
</tbody>
</table>
ip slb static

To configure a real server’s Network Address Translation (NAT) behavior and enter static NAT configuration mode, use the `ip slb static` command in global configuration mode. To restore the real server’s default NAT behavior, use the `no` form of this command.

```
ip slb static {drop | nat [virtual | virtual-ip [per-packet | sticky]]}
no ip slb static {drop | nat [virtual | virtual-ip [per-packet | sticky]]}
```

**Syntax Description**

- `drop` Indicates that IOS Server Load Balancing (IOS SLB) is to drop packets from this real server 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.

- `nat virtual` Configures the real server to use server NAT, and to use the virtual IP address that is configured on the `real` command in static NAT configuration mode when translating addresses.

- `nat virtual-ip` Configures the real server to use server NAT, and to use the specified virtual IP address when translating addresses.

- `per-packet` (Optional) 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.

- `sticky` (Optional) Indicates that IOS SLB is not to maintain connection state for packets originating from the real server, unless those packets match a sticky object. That is, if IOS SLB can find a matching sticky object, it builds the connection. Otherwise, IOS SLB does not build the connection.

**Defaults**

If you do not specify either the `per-packet` or `sticky` keyword, IOS SLB maintains connection state for packets originating from the real server.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If you specify the `virtual-ip` argument and you do not specify the `per-packet` option, IOS SLB uses server port translation to distinguish between connection requests initiated by different real servers.
Examples

The following example specifies that the real server is to use server NAT and to use virtual IP address 13.13.13.13 when translating addresses, and that IOS SLB is not to maintain connection state for any packets originating from the real server:

```
Router(config)# ip slb static nat 13.13.13.13 per-packet
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb static</td>
<td>Displays information about the static Network Address Translation (NAT) configuration.</td>
</tr>
</tbody>
</table>
ip slb timers gtp gsn

To change the amount of time IOS Server Load Balancing (IOS SLB) maintains sessions to and from an idle gateway general packet radio service (GPRS) support node (GGSN) or serving GPRS support node (SGSN), use the `ip slb timers gtp gsn` command in global configuration mode. To restore the default GPRS support node (GSN) idle timer, use the `no` form of this command.

```
ip slb timers gtp gsn duration
no ip slb timers gtp gsn duration
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>duration</code></td>
<td>GSN idle timer duration in seconds, which defines how long IOS SLB is to allow a GGSN or SGSN to be idle (that is, to go without echoing or signaling through IOS SLB). When the timer expires, IOS SLB cleans up all sessions that are using the idle GGSN or SGSN. The valid range is 1 to 65535 seconds. The default value is 90 seconds.</td>
</tr>
</tbody>
</table>

**Defaults**

The default duration is 90 seconds.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command sets the GSN idle timer for all IOS SLB virtual servers that are configured for GPRS Tunneling Protocol (GTP) cause code inspection. When the GSN idle timer expires, IOS SLB destroys all sessions to and from the idle GGSN or SGSN.

**Examples**

The following example specifies that IOS SLB maintains sessions for 45 seconds after a GGSN or SGSN becomes idle:

```
Router(config)# ip slb timers gtp gsn 45
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>virtual (virtual server)</td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
ip slb vserver

To identify a virtual server and enter virtual server configuration mode, use the `ip slb vserver` command in global configuration mode. To remove a virtual server from the IOS Server Load Balancing (IOS SLB) configuration, use the `no` form of this command.

```
ip slb vserver virtual-server
no ip slb vserver virtual-server
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>virtual-server</td>
<td>Character string used to identify the virtual server. The character string is limited to 15 characters.</td>
</tr>
</tbody>
</table>

| Defaults | No virtual server is identified. |

| Command Modes | Global configuration |

<table>
<thead>
<tr>
<th>Command History</th>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td></td>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td></td>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td></td>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
<th>The following example identifies a virtual server named PUBLIC_HTTP:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Router(config)# ip slb vserver PUBLIC_HTTP</td>
</tr>
<tr>
<td></td>
<td>Router(config-slb-vserver)#</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>serverfarm</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>show ip slb vserver</td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
**lookup**

To configure an IP address of a real server that a Domain Name System (DNS) server should supply in response to a domain name resolve request, use the `lookup` command in DNS probe configuration mode. To remove an IP address from the expected list, use the `no` form of this command.

```plaintext
lookup ip-address

no lookup ip-address
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th><code>ip-address</code></th>
<th>IP address of a real server that a DNS server should supply in response to a domain name resolve request.</th>
</tr>
</thead>
</table>

**Defaults**

No lookup IP address is configured.

**Command Modes**

DNS probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a DNS probe named PROBE4, enters DNS probe configuration mode, and specifies 13.13.13.13 as the IP address to resolve:

```plaintext
Router(config)# ip slb probe PROBE4 dns
Router(config-slb-probe)# lookup 13.13.13.13
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe dns</code></td>
<td>Configures a Domain Name System (DNS) probe name and enters DNS probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
maxclients

To specify the maximum number of IOS Server Load Balancing (IOS SLB) RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server, use the maxclients command in real server configuration mode. To remove the limit, use the no form of this command.

```
maxclients maximum-number

no maxclients
```

**Syntax Description**

- **maxclients maximum-number**
  - Maximum number of IOS SLB RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server:

  - If the **radius framed-ip** keyword is specified in the sticky command for the virtual server (that is, if the virtual server is configured to create the IOS SLB RADIUS framed-IP sticky database), a sticky subscriber is an entry in the IOS SLB RADIUS framed-IP sticky database.
  
  - If the **radius username** keyword is specified in the sticky command for the virtual server (that is, if the virtual server is configured to create the IOS SLB RADIUS username sticky database), a sticky subscriber is an entry in the IOS SLB RADIUS username sticky database.
  
  - If both the **radius framed-ip** and **radius username** keywords are specified in the sticky command for the virtual server, a sticky subscriber is an entry in the IOS SLB RADIUS username sticky database.

  By default, there is no limit on the number of IOS SLB RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server.

**Defaults**

There is no limit on the number of IOS SLB RADIUS and GTP sticky subscribers that can be assigned to an individual virtual server.

**Command Modes**

Real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>This command was modified to support RADIUS load balancing for CDMA2000, a third-generation (3-G) version of Code Division Multiple Access (CDMA).</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>
The following example specifies that up to 10 IOS SLB RADIUS sticky subscribers can be assigned to an individual real server:

Router(config-slb-real)# maxclients 10

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb route</td>
<td>Enables IOS Server Load Balancing (IOS SLB) to inspect packets for RADIUS framed-IP sticky routing.</td>
</tr>
<tr>
<td>show ip slb sticky</td>
<td>Displays the IOS Server Load Balancing (IOS SLB) sticky database.</td>
</tr>
</tbody>
</table>
maxconns (firewall farm datagram protocol)

To limit the number of active datagram connections to the firewall farm, use the `maxconns` command in firewall farm datagram protocol configuration mode. To restore the default of 4294967295, use the `no` form of this command.

```
maxconns maximum-number

no maxconns
```

**Syntax Description**

- `maximum-number` Maximum number of simultaneous active datagram connections using the firewall farm. Valid values range from 1 to 4294967295. The default is 4294967295.

**Defaults**

The default maximum number of simultaneous active datagram connections using the firewall farm is 4294967295.

**Command Modes**

Firewall farm datagram protocol configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example limits the real server to a maximum of 1000 simultaneous active connections:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol datagram
Router(config-slb-fw-udp)# maxconns 1000
```

**Related Commands**

- `protocol datagram` Enters firewall farm datagram protocol configuration mode.
- `show ip slb firewallfarm` Displays information about the firewall farm configuration.
- `show ip slb reals` Displays information about the real servers.
maxconns (firewall farm TCP protocol)

To limit the number of active TCP connections to the firewall farm, use the `maxconns` command in firewall farm TCP protocol configuration mode. To restore the default of 4294967295, use the `no` form of this command.

```
maxconns maximum-number

no maxconns
```

**Syntax Description**

```
maxconns maximum-number
```

- **maximum-number**: Maximum number of simultaneous active TCP connections using the firewall farm. Valid values range from 1 to 4294967295. The default is 4294967295.

**Defaults**

The default maximum number of simultaneous active TCP connections using the firewall farm is 4294967295.

**Command Modes**

Firewall farm TCP protocol configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example limits the real server to a maximum of 1000 simultaneous active connections:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol tcp
Router(config-slb-fw-tcp)# maxconns 1000
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protocol tcp</code></td>
<td>Enters firewall farm TCP protocol configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td><code>show ip slb reals</code></td>
<td>Displays information about the real servers.</td>
</tr>
</tbody>
</table>
maxconns (server farm)

To limit the number of active connections to the real server, use the maxconns command in real server configuration mode. To restore the default of 4294967295, use the no form of this command.

```
maxconns maximum-number

no maxconns
```

**Syntax Description**

| maximum-number | Maximum number of simultaneous active connections on the real server. Valid values range from 1 to 4294967295. The default is 4294967295. |

**Defaults**

The default maximum number of simultaneous active connections on the real server is 4294967295.

**Command Modes**

Real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example limits the real server to a maximum of 1000 simultaneous active connections:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1
Router(config-slb-real)# maxconns 1000
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real (server farm)</td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>show ip slb reals</td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td>show ip slb serverfarms</td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
To configure the aging time for flows, use the `mls aging slb normal` global configuration command. To restore the default setting, use the `no` form of this command.

```
mls aging slb normal time
no mls aging slb normal time
```

**Syntax Description**

- `time`: Idle time, in milliseconds, before a flow is aged. The valid range is 1 milliseconds to 10000 milliseconds. The default setting is 2000 milliseconds.

**Note**

Heavier-than-normal loads can age flows more aggressively than this time.

**Defaults**

The default aging idle time is 2000 milliseconds.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(8)E</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is supported for Catalyst 6000 Family Switches only.

**Examples**

The following example sets the idle time to 4000 milliseconds:

```
Router(config)# mls aging slb normal 4000
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb firewallfarm</code></td>
<td>Identifies a firewall farm and initiates firewall farm configuration mode.</td>
</tr>
<tr>
<td><code>ip slb serverfarm</code></td>
<td>Associates a real server farm with a virtual server.</td>
</tr>
<tr>
<td><code>ip slb vserver</code></td>
<td>Identifies a virtual server.</td>
</tr>
<tr>
<td><code>mls aging slb process</code></td>
<td>Controls how often the aging process runs.</td>
</tr>
</tbody>
</table>
**mls aging slb process**

To control how often the aging process runs, use the `mls aging slb process` global configuration command. To restore the default setting, use the `no` form of this command.

```
mls aging slb process time
no mls aging slb process time
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Time</th>
<th>Aging process interval, in milliseconds. The valid range is 1 millisecond to 10000 milliseconds. The default setting is 2000 seconds.</th>
</tr>
</thead>
</table>

**Defaults**

The default aging process interval is 2000 milliseconds.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(8)E</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is supported for Catalyst 6000 Family Switches only.

**Examples**

The following example sets the aging process interval to 4000 milliseconds:

```
Router(config)# mls aging slb process 4000
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb firewallfarm</td>
<td>Identifies a firewall farm and initiates firewall farm configuration mode.</td>
</tr>
<tr>
<td>ip slb serverfarm</td>
<td>Associates a real server farm with a virtual server.</td>
</tr>
<tr>
<td>ip slb vserver</td>
<td>Identifies a virtual server.</td>
</tr>
<tr>
<td>mls aging slb normal</td>
<td>Configures the aging time for flows.</td>
</tr>
</tbody>
</table>
To specify the behavior of IOS Server Load Balancing (IOS SLB) wildcard searches, use the `mls ip slb search wildcard` command in global configuration mode. To restore the default setting, use the `no` form of this command.

```
mls ip slb search {wildcard [pfc | rp] | icmp}

no mls ip slb search {wildcard [pfc | rp] | icmp}
```

**Syntax Description**

- `wildcard`  
  IOS SLB wildcard searches are to be performed by the Policy Feature Card (PFC). This value is the default setting.

- `pfc`  
  (Optional) IOS SLB wildcard searches are to be performed by the Policy Feature Card (PFC). This value is the default setting.

- `rp`  
  (Optional) IOS SLB wildcard searches are to be performed by the route processor.

- `icmp`  
  Disables ICMP handling by IOS SLB. (Pings to IOS SLB virtual IP addresses are still answered.) Use this command to reduce CPU usage when IOS SLB is configured in locations with a high volume of ICMP flows, such as in the network core.

  **Note** Use of the `icmp` keyword can result in minor ICMP errors, such as flows returned to the client with no Network Address Translation (NAT).

**Defaults**

The default setting is for the PFC to perform IOS SLB wildcard searches.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(7)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is supported for Catalyst 6500 Family Switches only.

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 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. However, be aware that this command can result in a slight increase in route processor utilization.
Examples  The following example limits wildcard searches to the route processor:

```
Router(config)# mls ip slb search wildcard rp
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb firewallfarm</code></td>
<td>Identifies a firewall by IP address farm and enters firewall farm configuration mode.</td>
</tr>
<tr>
<td><code>ip slb serverfarm</code></td>
<td>Associates a real server farm with a virtual server.</td>
</tr>
<tr>
<td><code>ip slb vserver</code></td>
<td>Identifies a virtual server.</td>
</tr>
</tbody>
</table>
nat

To configure IOS Server Load Balancing (IOS SLB) Network Address Translation (NAT) and specify a
NAT mode, use the `nat` command in server farm configuration mode. To remove a NAT configuration,
use the `no` form of this command.

```
nat {client pool | server}

no nat {client | server}
```

### Syntax Description

- **client pool**
  
  Configures the client address in load-balanced packets using addresses from the client address pool. The pool name must match the `pool` argument from a previous `ip slb natpool` command.

  This mode is commonly referred to as *directed client NAT*, or simply *client NAT*.

- **server**
  
  Configures the destination address in load-balanced packets sent to the real server as the address of the real server chosen by the server farm load-balancing algorithm.

  This mode is commonly referred to as *directed server NAT*, or simply *server NAT*.

### Defaults

No IOS SLB NAT is configured.

### Command Modes

Server farm configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(1)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(2)E</td>
<td>The <code>client</code> keyword and <code>pool</code> argument were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The `no nat` command is allowed only if the virtual server was removed from service with the `no inservice` command.
Examples

The following example enters server farm configuration mode and configures NAT mode as server address translation on server farm FARM2:

```
Router# ip slb serverfarm FARM2
Router(config-slb-sfarm)# nat server
```

The following example configures the NAT mode on server farm FARM2 to client translation mode and, using the `real` command in server farm configuration mode, configures the real server IP address as 10.3.1.1:

```
Router(config-slb-sfarm)# nat client web-clients
Router(config-slb-sfarm)# real 10.3.1.1
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb serverfarm</td>
<td>Associates a real server farm with a virtual server.</td>
</tr>
<tr>
<td>real (server farm)</td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>show ip slb serverfarms</td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
**port (custom UDP probe)**

To specify the port to which a custom User Datagram Protocol (UDP) probe is to connect, use the `port` command in custom UDP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
port port
no port port
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>port</code></td>
<td>UDP port number to which the custom UDP probe is to connect.</td>
</tr>
</tbody>
</table>

**Defaults**

In dispatched mode, the port number is inherited from the virtual server. If port translation is configured for the real server, that port number is used. See the `real (server farm)` command for more details.

**Command Modes**

Custom UDP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a custom UDP probe named PROBE6, enters custom UDP probe configuration mode, and configures the probe to connect to port number 8:

```
Router(config)# ip slb probe PROBE6 custom UDP
Router(config-slb-probe)# port 8
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe custom udp</code></td>
<td>Configures a custom User Datagram Protocol (UDP) probe name and enters custom UDP probe configuration mode.</td>
</tr>
<tr>
<td><code>real (server farm)</code></td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
port (HTTP probe)

To specify the port to which an HTTP probe is to connect, use the `port` command in HTTP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
port port

no port port
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>port</code></td>
<td>TCP or User Datagram Protocol (UDP) port number to which the HTTP probe is to connect.</td>
</tr>
</tbody>
</table>

**Defaults**

In dispatched mode, the port number is inherited from the virtual server. If port translation is configured for the real server, that port number is used. See the `real (server farm)` command for more details.

**Command Modes**

HTTP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures an HTTP probe named PROBE2, enters HTTP probe configuration mode, and configures the probe to connect to port number 8:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# port 8
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe http</code></td>
<td>Configures an HTTP probe name and enters HTTP probe configuration mode.</td>
</tr>
<tr>
<td><code>real (server farm)</code></td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
To specify the port to which a TCP probe is to connect, use the `port` command in TCP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
port port
no port port
```

**Syntax Description**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>port</code></td>
<td>TCP port number to which the TCP probe is to connect.</td>
</tr>
</tbody>
</table>

**Defaults**

In dispatched mode, the port number is inherited from the virtual server. If port translation is configured for the real server, that port number is used. See the `real (server farm)` command for more details.

**Command Modes**

TCP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a TCP probe named PROBE5, enters TCP probe configuration mode, and configures the probe to connect to port number 8:

```
Router(config)# ip slb probe PROBE5 tcp
Router(config-slb-probe)# port 8
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe tcp</code></td>
<td>Configures a TCP probe name and enters TCP probe configuration mode.</td>
</tr>
<tr>
<td><code>real (server farm)</code></td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
predictor (server farm)

To specify the load-balancing algorithm for selecting a real server in the server farm, use the `predictor` command in server farm configuration mode. To restore the default load-balancing algorithm of weighted round robin, use the `no` form of this command.

```
predictor [roundrobin | leastconns]

no predictor
```

**Syntax Description**

- `roundrobin` (Optional—IOS SLB feature only) Uses the weighted round robin algorithm for selecting the real server to handle the next new connection for the server farm. See the “Weighted Round Robin” section on page 6 for a detailed description of this algorithm. This algorithm is the default value.

  - RADIUS load balancing requires the weighted 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.
  - The Home Agent Director requires the weighted round robin algorithm.

- `leastconns` (Optional—IOS SLB feature only) Uses the weighted least connections algorithm for selecting the real server to handle the next new connection for this server farm. See the “Weighted Least Connections” section on page 7 for a detailed description of this algorithm.

**Defaults**

If you do not enter a `predictor` command, or if you enter the `predictor` command without specifying a load-balancing algorithm, the weighted round robin algorithm is used.

**Command Modes**

Server farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

- RADIUS load balancing requires the weighted round robin algorithm.
- GPRS load balancing without 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 Server Load Balancing (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 Server Load Balancing (IOS SLB) issues an error message.

**Examples**

The following example specifies the weighted least connections algorithm:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# predictor leastconns
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb serverfarms</td>
<td>Displays information about the server farm configuration.</td>
</tr>
<tr>
<td>weight (server farm)</td>
<td>Specifies the real server’s capacity, relative to other real servers in the server farm.</td>
</tr>
</tbody>
</table>
predictor hash address (firewall farm)

To specify the load-balancing algorithm for selecting a firewall in the firewall farm, use the `predictor hash address` command in firewall farm configuration mode. To restore the default load-balancing algorithm, use the `no` form of this command.

```
predictor hash address [port]
no predictor
```

**Syntax Description**

| Port | (Optional) Uses the source and destination TCP or User Datagram Protocol (UDP) port numbers, in addition to the source and destination IP addresses, when selecting a firewall. |

**Defaults**

IOS Server Load Balancing (IOS SLB) uses the source and destination IP addresses when selecting a firewall.

**Command Modes**

Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example specifies that source and destination IP addresses are to be used when selecting a firewall:

```
Router(config)# ip slb firewall FIRE1
Router(config-slb-fw)# predictor hash address
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td><code>weight (firewall farm real server)</code></td>
<td>Specifies the firewall’s capacity, relative to other firewalls in the firewall farm.</td>
</tr>
</tbody>
</table>
probe (firewall farm real server)

To associate a probe with a firewall farm, use the `probe` command in firewall farm real server configuration mode. To remove the association, use the `no` form of this command.

```
probe probe
no probe probe
```

**Syntax Description**

| `probe` | Name of the probe to associate with this firewall farm. |

**Defaults**

No probe is associated with a firewall farm.

**Command Modes**

Firewall farm real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can configure more than one probe for each firewall in a firewall farm.

If you configure probes in your network, you must also do one of the following:

- Configure the `exclude` keyword on the `client` command on the virtual server, to exclude connections initiated by the client IP address from the load-balancing scheme.
- Configure IP addresses on the IOS Server Load Balancing (IOS SLB) device that are Layer 3-adjacent to the real servers used by the virtual server.

**Examples**

The following example associates probe FireProbe with server farm FIRE1:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw-real)# probe FireProbe
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
probe (server farm)

To associate a probe with a server farm, use the **probe** command in server farm configuration mode. To remove the association, use the **no** form of this command.

```
probe probe
no probe probe
```

### Syntax Description

**probe**

Name of the probe to associate with this server farm.

### Defaults

No probe is associated with a server farm.

### Command Modes

Server farm configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

You can configure more than one probe for each server farm.

If you configure probes in your network, you must also do one of the following:

- Configure the **exclude** keyword on the **client** command on the virtual server, to exclude connections initiated by the client IP address from the load-balancing scheme.
- Configure IP addresses on the IOS Server Load Balancing (IOS SLB) device that are Layer 3-adjacent to the real servers used by the virtual server.

### Examples

The following example associates probe PROBE1 with server farm PUBLIC:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# probe PROBE1
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show ip slb serverfarms</strong></td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
**protocol datagram**

To enter firewall farm datagram protocol configuration mode, use the **protocol datagram** command in firewall farm configuration mode.

```plaintext
protocol datagram
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

No default behavior or values

**Command Modes**

Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced, replacing the <strong>udp</strong> command.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>This command was integrated into Cisco IOS Release 12.1(12c)E, replacing the <strong>protocol udp</strong> command.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Firewall farm datagram protocol configuration applies to the Encapsulation Security Payload (ESP), Generic Routing Encapsulation (GRE), IP in IP encapsulation, and User Datagram Protocol (UDP) protocols.

**Examples**

The following example enters firewall farm datagram protocol configuration mode:

```plaintext
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol datagram
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show ip slb firewallfarm</strong></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
To enter firewall farm TCP protocol configuration mode, use the `protocol tcp` command in firewall farm configuration mode.

`protocol tcp`

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
No default behavior or values

**Command Modes**
Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced, replacing the tcp command.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**
The following example enters firewall farm TCP protocol configuration mode:

```
Router(config)＃ ip slb firewallfarm FIRE1
Router(config-slb-fw)＃ protocol tcp
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
</tbody>
</table>
purge radius framed-ip acct on-off (virtual server)

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, use the `purge radius framed-ip acct on-off` virtual server configuration command. To disable this behavior, use the `no` form of this command.

```
purge radius framed-ip acct on-off
no purge radius framed-ip acct on-off
```

**Syntax Description**  
This command has no arguments or keywords.

**Defaults**  
IOS SLB purges entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting ON or OFF message.

**Command Modes**  
Virtual server configuration

**Command History**  
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**  
The following example prevents IOS SLB from purging entries in the IOS SLB RADIUS framed-ip sticky database upon receipt of an Accounting ON or OFF message:

```
Router(config)# ip slb vserver VS1
Router(config-slb-vserver)# no purge radius framed-ip acct on-off
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sticky (virtual server)</td>
<td>Assigns all connections from a client to the same real server.</td>
</tr>
</tbody>
</table>
real (firewall farm)

To identify a firewall as a member of a firewall farm and enter real server configuration mode, use the real command in firewall farm configuration mode. To remove the firewall from the IOS Server Load Balancing (IOS SLB) configuration, use the no form of this command.

```
real ip-address

no real ip-address
```

**Syntax Description**

| ip-address       | Real server IP address. |

**Defaults**

No firewall is identified as a member of a firewall farm.

**Command Modes**

Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A firewall farm comprises a number of firewalls. The firewalls are the physical devices that provide the firewall load-balanced services.

**Examples**

The following example identifies a firewall as a member of firewall farm FIRE1:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# real 10.1.1.1
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inservice (firewall farm real server)</td>
<td>Enables the firewall for use by IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>show ip slb firewallfarm</td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td>show ip slb reals</td>
<td>Displays information about the real servers.</td>
</tr>
</tbody>
</table>
real (server farm)

To identify a real server as a member of a server farm and enter real server configuration mode, use the `real` command in server farm configuration mode. To remove the real server from the IOS Server Load Balancing (IOS SLB) configuration, use the `no` form of this command.

```
real ip-address [port]
no real ip-address [port]
```

**Syntax Description**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>Real server IP address.</td>
</tr>
<tr>
<td>port</td>
<td>(Optional) Port translation for the server. Valid values range from 1 to 65535.</td>
</tr>
</tbody>
</table>

**Defaults**

No real server is identified as a member of a server farm.

**Command Modes**

Server farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(2)E</td>
<td>The <code>port</code> argument was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A server farm comprises a number of real servers. The real servers are the physical devices that provide the load-balanced services.

In general packet radio service (GPRS) load balancing, this command identifies a gateway GPRS support node (GGSN) that is a member of the server farm. Also, remember that the Cisco GGSN IP addresses are virtual template IP addresses, not real interface IP addresses.

In Virtual Private Network (VPN) server load balancing, this command identifies a real server acting as a VPN terminator.

**Examples**

The following example identifies a real server as a member of the server farm:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.1.1.1
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>inservice (server farm real server)</code></td>
<td>Enables the real server for use by IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td><code>show ip slb reals</code></td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td><code>show ip slb serverfarms</code></td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
real (static NAT)

To configure one or more real servers to use static Network Address Translation (NAT), use the `real` command in static NAT configuration mode. To restore the default behavior, use the `no` form of this command.

```
real ip-address [port]

no real ip-address [port]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip-address</code></td>
<td>IP address of the real server that is to use static NAT.</td>
</tr>
<tr>
<td><code>port</code></td>
<td>(Optional) Layer 4 source port number, used by IOS Server Load Balancing (IOS SLB) to differentiate between User Datagram Protocol (UDP) responses from the real server and connections initiated by the real server.</td>
</tr>
</tbody>
</table>

**Defaults**

No real server is configured to use static NAT.

**Command Modes**

Static NAT configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no port number is specified, IOS SLB uses static NAT for all packets outbound from the real server.

**Examples**

The following example configures real server 10.0.0.0 to use static NAT:

```
Router(config)# ip slb static nat
Router(config-slb-static)# real 10.0.0.0
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb static</code></td>
<td>Configures a real server’s Network Address Translation (NAT) behavior and enters static NAT configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb reals</code></td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td><code>show ip slb static</code></td>
<td>Displays information about the static Network Address Translation (NAT) configuration.</td>
</tr>
</tbody>
</table>
reassign

To specify 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, use the `reassign` command in real server configuration mode. To restore the default reassignment threshold, use the `no` form of this command.

```
reassign threshold

no reassign
```

**Syntax Description**

| `threshold` | Number of unacknowledged TCP SYNs (or Create PDP requests, in general packet radio service [GPRS] load balancing) that are directed to a real server before the connection is reassigned to a different real server. An unacknowledged SYN is one for which no SYN or ACKnowledgment (ACK) is detected before the next SYN arrives from the client. IOS Server Load Balancing (IOS SLB) allows 30 seconds for the connection to be established or for a new SYN to be received. If neither of these occurs within that time, the connection is removed from the IOS SLB database.

The 30-second timer is restarted for each SYN as long as the number of connection reassignments specified in the `faildetect (real server)` command’s `numconns` keyword is not exceeded. See the `faildetect (real server)` command for more information.

Valid threshold values range from one (1) to four (4) SYNs. The default value is three (3).

---

**Defaults**

The default threshold is three (3) SYNs.

**Command Modes**

Real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(9)E</td>
<td>This command was modified to support general packet radio service (GPRS) load balancing.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

IOS SLB does not reassign sticky connections if either of the following conditions is true:

- The real server is not OPERATIONAL or MAXCONNS_THROTTLED.
- The connection is the first for this sticky connection.
In GPRS load balancing, this command specifies the number of consecutive unacknowledged Create PDP requests (not TCP SYNs) that are directed to a gateway GPRS support node (GGSN) before the connection is reassigned to a different GGSN. You must specify a reassign threshold less than the serving GPRS support node (SGSN)’s N3-REQUESTS counter value:

\[ \text{threshold} < \text{n3-requests} \]

**Examples**

The following example sets the threshold of unacknowledged SYNs to 2:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1
Router(config-slb-real)# reassign 2
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>real (server farm)</strong></td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td><strong>show ip slb reals</strong></td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td><strong>show ip slb serverfarms</strong></td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
replicate casa (firewall farm)

To configure a stateful backup of IOS Server Load Balancing (IOS SLB) decision tables to a backup switch, use the `replicate casa` command in firewall farm configuration mode. To remove a `replicate casa` configuration, use the `no` form of this command.

```
replicate casa listen-ip remote-ip port [interval] [password [0 | 7] password [timeout]]

no replicate casa listen-ip remote-ip port
```

**Syntax Description**

- `listen-ip`  
  Listening IP address for state exchange messages that are advertised.

- `remote-ip`  
  Destination IP address for all state exchange signals.

- `port`  
  TCP or User Datagram Protocol (UDP) port number or port name for all state exchange signals.

- `interval`  
  (Optional) Maximum replication delivery interval from 1 to 300 seconds. The default value is 10 seconds.

- `password`  
  (Optional) Specifies the password for Message Digest Algorithm Version 5 (MD5) authentication.

- `0`  
  (Optional) Indicates that the password is unencrypted. This value is the default setting.

- `7`  
  (Optional) Indicates that the password is encrypted.

- `password`  
  (Optional) Password value for MD5 authentication. This password must match the password configured on the host agent.

- `timeout`  
  (Optional) Delay period, in seconds, during which both the old password and the new password are accepted. The default value is 180 seconds.

**Defaults**

The default interval is 10 seconds.
The default password encryption is 0 (unencrypted).
The default password timeout is 180 seconds.

**Command Modes**

Firewall farm configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `timeout` option allows you to change the password without stopping messages between the backup and primary Layer 3 switches. The default value is 180 seconds.

During the timeout, the backup sends packets with the old password (or null, if there is no old password), and receives packets with either the old or new password. After the timeout expires, the backup sends and receives packets only with the new password.
When setting a new password timeout, remember the following considerations:

- If you are configuring a new backup, set the timeout to 0 (send packets with the new password immediately). This configuration prevents password mismatches between the new backup and its primary.

- If you are changing the password for an existing backup, set a longer timeout to allow enough time for you to update the password on the primary before the timeout expires. Setting a longer timeout also prevents mismatches between the backup and primary.

**Examples**

The following example configures a stateful backup Layer-3 switch with a listening IP address of 10.10.10.11 and a remote IP address of 10.10.11.12 over HTTP port 4231:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# replicate casa 10.10.10.11 10.10.11.12 4231
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb firewallfarm</td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td>show ip slb replicate</td>
<td>Displays the configuration of IOS Server Load Balancing (IOS SLB) IP replication.</td>
</tr>
</tbody>
</table>
replicate casa (virtual server)

To configure a stateful backup of IOS Server Load Balancing (IOS SLB) decision tables to a backup switch, use the `replicate casa` command in virtual server configuration mode. To remove a `replicate casa` configuration, use the `no` form of this command.

```
replicate casa listen-ip remote-ip port [interval] [password [0 | 7] password [timeout]]

no replicate casa listen-ip remote-ip port
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>listen-ip</code></td>
<td>Listening IP address for state exchange messages that are advertised.</td>
</tr>
<tr>
<td><code>remote-ip</code></td>
<td>Destination IP address for all state exchange signals.</td>
</tr>
<tr>
<td><code>port</code></td>
<td>TCP or User Datagram Protocol (UDP) port number or port name for all state exchange signals.</td>
</tr>
<tr>
<td><code>interval</code></td>
<td>(Optional) Maximum replication delivery interval from 1 to 300 seconds. The default value is 10 seconds.</td>
</tr>
<tr>
<td><code>password</code></td>
<td>(Optional) Specifies the password for Message Digest Algorithm Version 5 (MD5) authentication.</td>
</tr>
<tr>
<td><code>0</code></td>
<td>(Optional) Indicates that the password is unencrypted. This value is the default setting.</td>
</tr>
<tr>
<td><code>7</code></td>
<td>(Optional) Indicates that the password is encrypted.</td>
</tr>
<tr>
<td><code>password</code></td>
<td>(Optional) Password value for MD5 authentication. This password must match the password configured on the host agent.</td>
</tr>
<tr>
<td><code>timeout</code></td>
<td>(Optional) Delay period, in seconds, during which both the old password and the new password are accepted. The default value is 180 seconds.</td>
</tr>
</tbody>
</table>

### Defaults

The default interval is 10 seconds.
The default password encryption is 0 (unencrypted).
The default password timeout is 180 seconds.

### Command Modes

Virtual server configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(3a)E</td>
<td>The 0 and 7 keywords were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>
Usage Guidelines

The **timeout** option allows you to change the password without stopping messages between the backup and primary Layer 3 switches. The default value is 180 seconds.

During the timeout, the backup sends packets with the old password (or null, if there is no old password), and receives packets with either the old or new password. After the timeout expires, the backup sends and receives packets only with the new password.

When setting a new password timeout, remember the following considerations:

- If you are configuring a new backup, set the timeout to 0 (send packets with the new password immediately). This configuration prevents password mismatches between the new backup and its primary.
- If you are changing the password for an existing backup, set a longer timeout to allow enough time for you to update the password on the primary before the timeout expires. Setting a longer timeout also prevents mismatches between the backup and primary.

General packet radio service (GPRS) load balancing without GPRS Tunneling Protocol (GTP) cause code inspection enabled does not support the **replicate casa** command in virtual server configuration mode.

The Home Agent Director does not support the **replicate casa** command in virtual server configuration mode.

Examples

The following example configures a stateful backup Layer-3 switch with a listening IP address of 10.10.10.11 and a remote IP address of 10.10.11.12 over HTTP port 4231:

```
Router(config)# ip slb vserver VS1
Router(config-slb-vserver)# replicate casa 10.10.10.11 10.10.11.12 4231
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb replicate</code></td>
<td>Displays the configuration of IOS Server Load Balancing (IOS SLB) IP replication.</td>
</tr>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
request (custom UDP probe)

To define the payload of the User Datagram Protocol (UDP) request packet to be sent by a custom UDP probe, use the `request` command in custom UDP probe configuration mode.

```
request data {start-byte | continue} hex-data-string
```

**Syntax Description**

- **data start-byte** Identifies the payload offset at which the `hex-data-string` is to be placed into the packet.
- **data continue** String of characters represented by the `hex-data-string` argument is to be placed after the last defined byte in the request packet.
- **hex-data-string** Payload of the UDP request packet, up to 100 bytes of data in hexadecimal format.

**Defaults**

The payload of the UDP request packet is not defined.

**Command Modes**

Custom UDP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can enter more than one `request` command, to specify the entire UDP payload.

**Examples**

The following example generates custom UDP probe PROBE6, with the specified 119-byte UDP payload.

```
Router(config)# ip slb probe PROBE6 custom UDP
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
Router(config-slb-probe)# request data 20 01 07 63 68 72 69 73 28 06 00 00 00 01 2C 0A 30 30 30 30
Router(config-slb-probe)# request data 40 30 30 42 07 06 00 00 00 07 1E 10 63 75 66 66 2E 63 69 73 63
Router(config-slb-probe)# request data 60 6F 2E 63 6F 6D 1F 0C 39 31 39 33 39 32 39 31 36 39 08 06 0A
Router(config-slb-probe)# request data 80 0A 01 01 2D 06 00 00 00 01 3D 06 00 00 00 05 05 06 00 00 00
Router(config-slb-probe)# request data 100 00 06 06 00 00 00 02 04 06 0A 0A 18 0A 29 06 00 00 00
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe custom udp</code></td>
<td>Configures the IOS Server Load Balancing (IOS SLB) IP probe name.</td>
</tr>
<tr>
<td><code>response</code></td>
<td>Defines the data string to match against custom User Datagram Protocol (UDP) probe response packets.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
request (HTTP probe)

To configure an HTTP probe to check the status of the real servers, use the request command in HTTP probe configuration mode. To remove a request configuration, use the no form of this command.

```
request [method {get | post | head | name name}] [url path]
no request [method {get | post | head | name name}] [url path]
```

### Syntax Description

- **method** (Optional) Configures the way the data is requested from the server.
- **get** (Optional) Configures the Get method to request data from the server.
- **post** (Optional) Configures the Post method to request data from the server.
- **head** (Optional) Configures the header data type to request data from the server.
- **name name** (Optional) Configures the name string of the data to send to the servers to request data. The character string is limited to 15 characters.
- **url path** (Optional) Configures the path from the server.

### Defaults

No HTTP probe is configured to check the status of the real servers.

### Command Modes

HTTP probe configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The request command configures the IOS Server Load Balancing (IOS SLB) HTTP probe method used to receive data from the server. Only one IOS SLB HTTP probe can be configured for each server farm. If no values are configured following the method keyword, the default is Get.

If no URL path is set to the server, the default is /.

### Examples

The following example configures an IOS SLB HTTP probe named PROBE2, enters HTTP probe configuration mode, and configures HTTP requests to use the post method and the URL /probe.cgi?all:

```
Router(config)# ip slb probe PROBE2 http
Router(config-slb-probe)# request method post url /probe.cgi?all
```
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb probe http</td>
<td>Configures the IOS Server Load Balancing (IOS SLB) IP probe name.</td>
</tr>
<tr>
<td>show ip slb probe</td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
response

To define the data string to match against custom User Datagram Protocol (UDP) probe response packets, use the `response` command in custom UDP probe configuration mode.

```
response clause-number data start-byte hex-data-string
```

**Syntax Description**

- **clause-number**: Identifies the response clause that is being modified. Up to 8 response clauses can be specified, on individual `response` commands.
- **data start-byte**: Byte in the UDP response packet at which the `hex-data-string` is to be matched.
- **hex-data-string**: Up to 100 bytes of data, in hexadecimal format, that is to be matched against the UDP response packet payload. If the data does not match, the probe fails.

**Defaults**

The data string to match against custom UDP probe response packets is not defined.

**Command Modes**

Custom UDP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can enter up to 8 individual response commands, to parse up to 8 non-contiguous bytes of data.

**Examples**

In the following example, if the 26th and 27th bytes of the response from `PROBE6` are not `FF FF`, and the 44th and 45th bytes are not `DD DD`, the probe fails.

```
Router(config)# ip slb probe PROBE6 custom UDP
Router(config-slb-probe)# response 1 data 26 FF FF
Router(config-slb-probe)# response 2 data 44 DD DD
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe custom udp</code></td>
<td>Configures the IOS Server Load Balancing (IOS SLB) IP probe name.</td>
</tr>
<tr>
<td><code>request (custom UDP probe)</code></td>
<td>Defines the payload of the User Datagram Protocol (UDP) request packet to be sent by a custom UDP probe.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
retry

To specify how long to wait before a new connection is attempted to a failed server, use the `retry` command in real server configuration mode. To restore the default retry value, use the `no` form of this command.

```
retry retry-value

no retry
```

**Syntax Description**

- `retry-value` Time, in seconds, to wait after the detection of a server failure before a new connection to the server is attempted.

If the new connection attempt succeeds, the real server is placed in OPERATIONAL state. If the connection attempt fails, the timer is reset, the connection is reassigned, and the process repeats until it is successful or until the server is placed in the OUTOFSERVICE state by the network administrator.

Valid values range from 1 to 3600. The default value is 60 seconds.

A value of 0 means do not attempt a new connection to the server when it fails.

**Defaults**

The default retry-value is 60 seconds.

**Command Modes**

Real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example specifies that 120 seconds must elapse after the detection of a server failure before a new connection is attempted:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1
Router(config-slb-real)# retry 120
```
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real (server farm)</td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>show ip slb reals</td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td>show ip slb serverfarms</td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
serverfarm

To associate a real server farm with a virtual server, and optionally configure a backup server farm and specify that sticky connections are to be used in the backup server farm, use the serverfarm command in virtual server configuration mode. To remove the server farm association from the virtual server configuration, use the no form of this command.

```
serverfarm primary-farm [backup backup-farm [sticky]]
```

```
o serverfarm primary-farm [backup backup-farm [sticky]]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>primary-farm</td>
<td>Name of a server farm that has already been defined using the <code>ip slb serverfarm</code> command.</td>
</tr>
<tr>
<td>backup backup-farm</td>
<td>(Optional) Specifies the name of a backup server farm that has already been defined using the <code>ip slb serverfarm</code> command.</td>
</tr>
<tr>
<td>sticky</td>
<td>(Optional) Specifies that sticky connections are to be used in the backup server farm.</td>
</tr>
</tbody>
</table>

### Defaults

No real server farm is associated with a virtual server.

### Command Modes

Virtual server configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(8a)E</td>
<td>The <code>backup</code> and <code>sticky</code> keywords and the <code>backup-farm</code> argument were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

In general packet radio service (GPRS) load balancing and the Home Agent Director, the `sticky` keyword is not supported.

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.

If `backup backup-farm` is not specified, no backup server farm is configured.

If a backup server farm is configured but the `sticky` keyword is not specified, sticky connections are not used in the backup server farm.
The following example shows how the `ip slb vserver`, `virtual`, and `serverfarm` commands are used to associate the real server farm named PUBLIC with the virtual server named PUBLIC_HTTP.

Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# virtual 10.0.0.1 tcp www
Router(config-slb-vserver)# serverfarm PUBLIC

<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>ip slb serverfarm</strong></td>
<td>Identifies a server farm and enters server farm configuration mode.</td>
</tr>
<tr>
<td></td>
<td><strong>show ip slb vserver</strong></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td></td>
<td><strong>virtual (virtual server)</strong></td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
show ip slb conns

To display the active IOS Server Load Balancing (IOS SLB) connections, use the `show ip slb conns` command in privileged EXEC mode.

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

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vserver</td>
<td>(Optional) Displays only those connections associated with the specified virtual server.</td>
</tr>
<tr>
<td>client</td>
<td>(Optional) Displays only those connections associated with the specified client IP address.</td>
</tr>
<tr>
<td>firewall</td>
<td>(Optional) Displays only those connections associated with the specified firewall farm.</td>
</tr>
<tr>
<td>detail</td>
<td>(Optional) Displays detailed connection information.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(7)E</td>
<td>The <code>firewall</code> keyword and <code>firewall-farm</code> argument were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no options are specified, the command displays output for all active IOS SLB connections.

**Examples**

The following is sample output from the `show ip slb conns` command:

```
Router# show ip slb conns

vserver    prot    client              real              state
-------------------------------------------------------------------------
TEST       TCP      7.150.72.183:328  80.80.90.25:80        INIT
TEST       TCP      7.250.167.226:423 80.80.90.26:80        INIT
TEST       TCP      7.234.60.239:317  80.80.90.26:80        ESTAB
TEST       TCP      7.110.233.96:747   80.80.90.26:80        ESTAB
TEST       TCP      7.162.0.201:770    80.80.90.30:80        CLOSING
TEST       TCP      7.22.225.219:995  80.80.90.26:80        CLOSING
TEST       TCP      7.2.170.148:169   80.80.90.30:80        ZOMBIE
```
Table 2 describes the fields shown in the display.

**Table 2  show ip slb conns Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vserver</td>
<td>Name of the virtual server whose connections are being monitored and displayed. Information about each connection is displayed on a separate line.</td>
</tr>
<tr>
<td>prot</td>
<td>Protocol being used by the connection.</td>
</tr>
<tr>
<td>client</td>
<td>Client IP address being used by the connection.</td>
</tr>
<tr>
<td>real</td>
<td>Real IP address of the connection.</td>
</tr>
<tr>
<td>state</td>
<td>Current state of the connection:</td>
</tr>
<tr>
<td></td>
<td>• CLOSING—IOS SLB TCP connection deactivated (awaiting a delay timeout before cleaning up the connection).</td>
</tr>
<tr>
<td></td>
<td>• ESTAB—IOS SLB TCP connection processed a SYNchronize sequence number (SYN)-SYN-to-ACKnowledgment (ACK) exchange between the client and server.</td>
</tr>
<tr>
<td></td>
<td>• FINCLIENT—IOS SLB TCP connection processed a finish (FIN) from the client.</td>
</tr>
<tr>
<td></td>
<td>• FINSERVER—IOS SLB TCP connection processed a FIN from the server.</td>
</tr>
<tr>
<td></td>
<td>• INIT—Initial state of the IOS SLB TCP connection.</td>
</tr>
<tr>
<td></td>
<td>• SYNBOTH—IOS SLB TCP connection processed one or more TCP SYNs from both the client and the server.</td>
</tr>
<tr>
<td></td>
<td>• SYNCLIENT—IOS SLB TCP connection processed one or more client TCP SYNs.</td>
</tr>
<tr>
<td></td>
<td>• SYNSERVER—IOS SLB TCP connection processed one or more server 1 TCP SYNs.</td>
</tr>
<tr>
<td></td>
<td>• ZOMBIE—Deletion of the IOS SLB TCP connection failed, possibly because of bound flows. Deletion will proceed when the flows are unbound.</td>
</tr>
</tbody>
</table>
show ip slb dfp

To display Dynamic Feedback Protocol (DFP) manager and agent information, such as passwords, timeouts, retry counts, and weights, use the `show ip slb dfp` command in privileged EXEC mode.

```
show ip slb dfp [agent agent-ip port | manager manager-ip | detail | weights]
```

**Syntax Description**

- **agent** (Optional) Displays information about an agent.
- **agent-ip** (Optional) Agent IP address.
- **port** (Optional) Agent TCP or User Datagram Protocol (UDP) port number.
- **manager** (Optional) Displays information about the specified manager.
- **manager-ip** (Optional) Manager IP address.
- **detail** (Optional) Displays all data available.
- **weights** (Optional) Displays information about weights assigned to real servers for load balancing.

**Defaults**

If no options are specified, the command displays summary information.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(5a)E</td>
<td>The <strong>manager</strong> keyword and manager-ip argument were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no options are specified, the command displays summary information.

**Examples**

The following sample output from the `show ip slb dfp` command displays high-level information about all DFP agents and managers:

```
Router# show ip slb dfp
DFP Manager:
  Current passwd:NONE Pending passwd:NONE
  Passwd timeout:0 sec

Agent IP          Port    Timeout   Retry Count   Interval
---------------------------------------------------------------
161.44.2.34       61936   0         0             180 (Default)
```
Table 3 describes the fields shown in the display.

### Table 3  show ip slb dfp Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent IP</td>
<td>IP address of the agent about which information is being displayed.</td>
</tr>
<tr>
<td>Port</td>
<td>TCP or UDP port number of the agent. The valid range is 1 to 65535.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Time period, in seconds, during which the DFP manager must receive an update from the DFP agent. A value of 0 means there is no timeout.</td>
</tr>
<tr>
<td>Retry Count</td>
<td>Number of times the DFP manager attempts to establish the TCP connection to the DFP agent. A value of 0 means there are infinite retries.</td>
</tr>
<tr>
<td>Interval</td>
<td>Interval, in seconds, between retries.</td>
</tr>
</tbody>
</table>

The following example displays detailed information about DFP agents and managers:

```
Router# show ip slb dfp detail
DFP Manager
    Current passwd <none> Pending passwd <none>
    Passwd timeout 0 sec
    Unexpected errors 0
% No DFP Agents configured
```

The following example displays detailed information about DFP manager 55.55.55.2:

```
Router# show ip slb dfp manager 55.55.55.2
DFP Manager 55.55.55.2 Connection state Connected
    Timeout = 20
    Last message sent 033537 UTC 01/02/00
```

The following example displays detailed information about weights assigned to real servers for load balancing:

```
Router# show ip slb dfp weights
Real IP Address 17.17.17.17 Protocol TCP Port 22 Bind_ID 111 Weight 111
    Set by Agent 161.44.2.3458490 at 132241 UTC 12/03/99
Real IP Address 17.17.17.17 Protocol TCP Port www Bind_ID 1 Weight 1
    Set by Agent 161.44.2.3458490 at 132241 UTC 12/03/99
Real IP Address 68.68.68.68 Protocol TCP Port www Bind_ID 4 Weight 4
    Set by Agent 161.44.2.3458490 at 132241 UTC 12/03/99
Real IP Address 85.85.85.85 Protocol TCP Port www Bind_ID 5 Weight 5
    Set by Agent 161.44.2.3458490 at 132241 UTC 12/03/99
```
show ip slb firewallfarm

To display firewall farm information, use the **show ip slb firewallfarm** command in privileged EXEC mode.

```
show ip slb firewallfarm [detail]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>detail</td>
<td>(Optional) Displays detailed information.</td>
</tr>
</tbody>
</table>

### Command Modes

Privileged EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Examples

The following is sample output from the **show ip slb firewallfarm** command:

```
Router# show ip slb firewallfarm

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

Table 4 describes the fields shown in the display.

### Table 4  show ip slb firewallfarm Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>firewall farm</td>
<td>Name of the firewall farm.</td>
</tr>
<tr>
<td>hash</td>
<td>Load-balancing algorithm used to select a firewall for the firewall farm:</td>
</tr>
<tr>
<td></td>
<td>- IPADDR—Uses the source and destination IP addresses in the algorithm.</td>
</tr>
<tr>
<td></td>
<td>- IPADDRPORT—Uses the source and destination TCP or User Datagram Protocol (UDP) port numbers, in addition to the source and destination IP addresses, in the algorithm.</td>
</tr>
<tr>
<td></td>
<td>See the <strong>predictor hash address (firewall farm)</strong> command for more details.</td>
</tr>
<tr>
<td>state</td>
<td>Current state of the firewall farm:</td>
</tr>
<tr>
<td></td>
<td>- OPERATIONAL—Functioning properly.</td>
</tr>
<tr>
<td></td>
<td>- OUTOFSERVICE—Removed from the load-balancing predictor lists.</td>
</tr>
<tr>
<td></td>
<td>- STANDBY—Backup firewall farm, ready to become operational if the active firewall farm fails.</td>
</tr>
<tr>
<td>reals</td>
<td>Number of firewalls that are members of the firewall farm.</td>
</tr>
</tbody>
</table>
show ip slb fragments

To display information from the IOS Server Load Balancing (IOS SLB) fragment database, use the `show ip slb fragments` command in privileged EXEC mode.

show ip slb fragments

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following sample output from the `show ip slb fragments` command shows fragment information for virtual server 10.11.11.11:

```
Router# show ip slb fragments

<table>
<thead>
<tr>
<th>ip src</th>
<th>id</th>
<th>forward</th>
<th>src nat</th>
<th>dst nat</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.11.2.128</td>
<td>12</td>
<td>10.11.2.128</td>
<td>10.11.11.11</td>
<td>10.11.2.128</td>
</tr>
<tr>
<td>10.11.2.128</td>
<td>13</td>
<td>10.11.2.128</td>
<td>10.11.11.11</td>
<td>10.11.2.128</td>
</tr>
<tr>
<td>10.11.2.128</td>
<td>14</td>
<td>10.11.2.128</td>
<td>10.11.11.11</td>
<td>10.11.2.128</td>
</tr>
<tr>
<td>10.11.2.128</td>
<td>15</td>
<td>10.11.2.128</td>
<td>10.11.11.11</td>
<td>10.11.2.128</td>
</tr>
<tr>
<td>10.11.2.128</td>
<td>16</td>
<td>10.11.2.128</td>
<td>10.11.11.11</td>
<td>10.11.2.128</td>
</tr>
</tbody>
</table>
```

**Table 5** describes the fields shown in the display.

**Table 5**  show ip slb fragments Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip src</td>
<td>Source IP address of the fragment.</td>
</tr>
<tr>
<td>id</td>
<td>IP ID of the fragment, set by the packet originator.</td>
</tr>
<tr>
<td>forward</td>
<td>IP address to which the fragment is being forwarded.</td>
</tr>
<tr>
<td>src nat</td>
<td>If using Network Address Translation (NAT), new source IP address after NAT.</td>
</tr>
<tr>
<td>dst nat</td>
<td>If using NAT, new destination IP address after NAT.</td>
</tr>
</tbody>
</table>
show ip slb gtp

To display IOS Server Load Balancing (IOS SLB) general packet radio service (GPRS) Tunneling Protocol (GTP) information, use the `show ip slb gtp` command in privileged EXEC mode.

```bash
show ip slb gtp { gsn [gsn-ip-address] | nsapi [nsapi-key] [detail] }
```

**Syntax Description**

- **gsn**: (Optional) Displays IOS SLB database information for the specified gateway GPRS support node (GGSN) or serving GPRS support node (SGSN).
- **gsn-ip-address**: (Optional) IP address of the GGSN or SGSN for which information is to be displayed. If you do not specify a `gsn-ip-address`, IOS SLB displays information for all GGSNs and SGSNs.
- **nsapi**: (Optional) Displays IOS SLB database information for the specified NSAPI.
- **nsapi-key**: (Optional) Key of the NSAPI for which information is to be displayed. If you do not specify an `nsapi-key`, IOS SLB displays information for all NSAPIs.
- **detail**: (Optional) Displays additional, more detailed information.

**Defaults**

If you specify `gsn` and you do not specify a `gsn-ip-address`, IOS SLB displays information for all GGSNs and SGSNs.

If you specify `nsapi` and you do not specify an `nsapi-key`, IOS SLB displays information for all NSAPIs.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(13)E3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ip slb gtp gsn` command for a specific GGSN or SGSN:

```bash
Router# show ip slb gtp gsn 12.12.12.1

type ip  recovery-ie purging
-------------------------------
SGSN 12.12.12.1 UNKNOWN N
```

*Table 6* describes the fields shown in the display.

**Table 6  show ip slb gtp gsn Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Type of GSN (either GGSN or SGSN).</td>
</tr>
<tr>
<td>ip</td>
<td>IP address of the GGSN or SGSN.</td>
</tr>
</tbody>
</table>
The following is sample output from the `show ip slb gtp nsapi` command:

```plaintext
Router# show ip slb gtp nsapi
nsapi key        real                   nsapi count session count
-----------------------------------------------------------------
11111111111111F1 112.112.112.3          1           1
```

The following is sample output from the `show ip slb gtp nsapi` command for a specific NSAPI key:

```plaintext
Router# show ip slb gtp nsapi 11111111111111F1
nsapi key        real                   nsapi count session count
-----------------------------------------------------------------
11111111111111F1 112.112.112.3          1           1
```

Table 7 describes the fields shown in the display.

**Table 7 show ip slb gtp nsapi Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nsapi key</td>
<td>Key for the session. This is the IMSI.</td>
</tr>
<tr>
<td>real</td>
<td>Real server to which the session is assigned.</td>
</tr>
<tr>
<td>nsapi count</td>
<td>Number of NSAPIs bound to the session. This is the number of PDP contexts (mobile sessions) on the GGSN associated with the IMSI.</td>
</tr>
<tr>
<td>session count</td>
<td>Number of sessions to which the NSAPI is currently bound. Normally, the NSAPI is bound to one session, but it is bound to two sessions in transition during an update.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ip slb gtp nsapi detail` command:

```plaintext
Router# show ip slb gtp nsapi detail
IMSI key = 11111111111111F1, real = 112.112.112.3, nsapi count = 1, session count = 1
no vserver          key              client                 state      seq
---------------------------------------------------------------------------
5 SERVER1            0009E8810009E881 12.12.12.1:2123        GTP_INIT   0
```
Table 8 describes the fields shown in the display.

**Table 8**  
*show ip slb gtp nsapi detail Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMSI key</td>
<td>IMSI key for the session.</td>
</tr>
<tr>
<td>real</td>
<td>Real server to which the session is assigned.</td>
</tr>
<tr>
<td>nsapi count</td>
<td>Number of NSAPIs bound to the session. This is the number of PDP contexts (mobile sessions) on the GGSN associated with this IMSI.</td>
</tr>
<tr>
<td>session count</td>
<td>Number of sessions to which the NSAPI is currently bound. Normally, the NSAPI is bound to one session, but it is bound to two sessions in transition during an update.</td>
</tr>
<tr>
<td>no</td>
<td>NSAPI number.</td>
</tr>
<tr>
<td>vserver</td>
<td>Name of the virtual server.</td>
</tr>
<tr>
<td>key</td>
<td>Session key.</td>
</tr>
<tr>
<td>client</td>
<td>SGSN IP address and port number.</td>
</tr>
<tr>
<td>state</td>
<td>State of the session. Possible states are:</td>
</tr>
<tr>
<td></td>
<td>• GTP_ESTAB—The session has been established successfully.</td>
</tr>
<tr>
<td></td>
<td>• GTP_INIT—The PDP contexts have been deleted as a result of a delete request or a deletion in GGSN, and IOS SLB is waiting to destroy the session after the GTP_TIMEOUT.</td>
</tr>
<tr>
<td></td>
<td>• GTP_IOCTL_REQ_CLIENT—Waiting for a response from the real server.</td>
</tr>
<tr>
<td>seq</td>
<td>Sequence number in the last delete request.</td>
</tr>
</tbody>
</table>
show ip slb natpool

To display the IP IOS Server Load Balancing (IOS SLB) Network Address Translation (NAT) configuration, use the `show ip slb natpool` command in privileged EXEC mode.

```
show ip slb natpool [name pool] [detail]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name pool</td>
<td>(Optional) Displays the specified NAT pool.</td>
</tr>
<tr>
<td>detail</td>
<td>(Optional) Lists all the interval ranges currently allocated in the client NAT pool.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the default `show ip slb natpool` command:

```
Router# show ip slb natpool
nat client B 1.1.1.6 1.1.1.8 Netmask 255.255.255.0
nat client A 1.1.1.1 1.1.1.5 Netmask 255.255.255.0
```

The following is sample output from the `show ip slb natpool` command with the `detail` parameter:

```
Router# show ip slb natpool detail
nat client A 1.1.1.1 1.1.1.5 Netmask 255.255.255.0
<table>
<thead>
<tr>
<th>Start NAT</th>
<th>Last NAT</th>
<th>Count</th>
<th>ALLOC/FREE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.1.1:11001</td>
<td>1.1.1.1:16333</td>
<td>0005333</td>
<td>ALLOC</td>
</tr>
<tr>
<td>1.1.1.1:16334</td>
<td>1.1.1.1:19000</td>
<td>0002667</td>
<td>ALLOC</td>
</tr>
<tr>
<td>1.1.1.1:19001</td>
<td>1.1.1.5:65535</td>
<td>0264675</td>
<td>FREE</td>
</tr>
</tbody>
</table>

nat client B 1.1.1.6 1.1.1.8 Netmask 255.255.255.0
<table>
<thead>
<tr>
<th>Start NAT</th>
<th>Last NAT</th>
<th>Count</th>
<th>ALLOC/FREE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.1.6:11001</td>
<td>1.1.1.6:16333</td>
<td>0005333</td>
<td>ALLOC</td>
</tr>
<tr>
<td>1.1.1.6:16334</td>
<td>1.1.1.6:19000</td>
<td>0002667</td>
<td>ALLOC</td>
</tr>
<tr>
<td>1.1.1.6:19001</td>
<td>1.1.1.8:65535</td>
<td>0155605</td>
<td>FREE</td>
</tr>
</tbody>
</table>
```

Table 9 describes the fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start NAT</td>
<td>Starting NAT address in a range of addresses in the client NAT pool.</td>
</tr>
<tr>
<td>Last NAT</td>
<td>Last NAT address in a range of addresses in the client NAT pool.</td>
</tr>
</tbody>
</table>
show ip slb natpool

Table 9 show ip slb natpool detail Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Number of NAT addresses in the range.</td>
</tr>
<tr>
<td>ALLOC/FREE</td>
<td>Indicates whether the range of NAT addresses has been allocated or is free.</td>
</tr>
</tbody>
</table>

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip slb natpool</td>
<td>Configures the IOS Server Load Balancing (IOS SLB) Network Address Translation (NAT).</td>
</tr>
</tbody>
</table>
show ip slb probe

To display information about an IOS Server Load Balancing (IOS SLB) probe, use the `show ip slb probe` command in privileged EXEC mode.

    show ip slb probe [name probe] [detail]

Syntax Description

- **name probe** (Optional) Displays information about the specified probe.
- **detail** (Optional) Displays detailed information, including the SA Agent operation ID, which you can correlate with the output of the `show rtr operational-state` command in EXEC mode.

Command Modes

Privileged EXEC

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the `show ip slb probe` 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
```

Table 10 describes the fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server:Port</td>
<td>IP address and port of the real server.</td>
</tr>
<tr>
<td>State</td>
<td>Operational state of the probe:</td>
</tr>
<tr>
<td></td>
<td>• FAILED—The probe has succeeded in the past but has currently failed.</td>
</tr>
<tr>
<td></td>
<td>• OPERATIONAL—The probe is functioning normally.</td>
</tr>
<tr>
<td></td>
<td>• TESTING—The probe has never succeeded, due to no response. IOS SLB keeps</td>
</tr>
<tr>
<td></td>
<td>no counters or timers for this state.</td>
</tr>
<tr>
<td></td>
<td>For a detailed listing of real server states, see the <code>show ip slb reals</code></td>
</tr>
<tr>
<td></td>
<td>command.</td>
</tr>
<tr>
<td>Outages</td>
<td>Number of intervals between successful probes.</td>
</tr>
<tr>
<td>Current</td>
<td>Time since the last probe success. That is, the duration (so far) of the</td>
</tr>
<tr>
<td></td>
<td>current outage.</td>
</tr>
<tr>
<td>Cumulative</td>
<td>Total time the real server has been under test by the probe and has failed</td>
</tr>
<tr>
<td></td>
<td>the probe test. This value is the sum of the Current time plus the total</td>
</tr>
<tr>
<td></td>
<td>time of all previous outages.</td>
</tr>
</tbody>
</table>
**show ip slb reals**

To display information about the real servers, use the `show ip slb reals` command in privileged EXEC mode.

```
show ip slb reals [sfarm server-farm] [detail]
```

**Syntax Description**

- `sfarm server-farm` (Optional) Displays information about those real servers associated with the specified server farm or firewall farm.
- `detail` (Optional) Displays detailed information.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(13)E</td>
<td>The <code>vserver</code> keyword and <code>virtual-server</code> argument were replaced with the <code>sfarm</code> keyword and <code>server-farm</code> argument.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no options are specified, the command displays information about all real servers.

**Examples**

The following is sample output from the `show ip slb reals` command:

```
Router# show ip slb reals
real             farm name        weight   state           conns
--------------------------------------------------------------------
80.80.2.112      FRAG             8        OUTOFSERVICE    0
80.80.5.232      FRAG             8        OPERATIONAL     0
80.80.15.124     FRAG             8        OUTOFSERVICE    0
80.80.15.124     LINUX            8        OPERATIONAL     0
80.80.15.125     LINUX            8        OPERATIONAL     0
80.80.15.126     LINUX            8        OPERATIONAL     0
80.80.90.25      SRE              8        OPERATIONAL     220
80.80.90.26      SRE              8        OPERATIONAL     220
80.80.90.27      SRE              8        OPERATIONAL     216
80.80.90.28      SRE              8        TESTING         1
80.80.90.29      SRE              8        OPERATIONAL     221
80.80.90.30      SRE              8        OPERATIONAL     224
80.80.30.3       TEST             100      READY_TO_TEST   0
80.80.30.4       TEST             100      READY_TO_TEST   0
80.80.30.5       TEST             100      READY_TO_TEST   0
80.80.30.6       TEST             100      READY_TO_TEST   0
```
Table 11 describes the fields shown in the display.

**Table 11  show ip slb reals Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real</td>
<td>IP address of the real server about which information is being displayed. Used to identify each real server. Information about each real server is displayed on a separate line.</td>
</tr>
<tr>
<td>farm name</td>
<td>Name of the server farm or firewall farm with which the real server is associated.</td>
</tr>
<tr>
<td>weight</td>
<td>Weight assigned to the real server. The weight identifies the real server’s capacity, relative to other real servers in the server farm.</td>
</tr>
<tr>
<td>state</td>
<td>Current state of the real server.</td>
</tr>
<tr>
<td></td>
<td>• DFP_THROTTLED—The Dynamic Feedback Protocol (DFP) agent sent a weight of 0 for this real server (send no further connections to this real server).</td>
</tr>
<tr>
<td></td>
<td>• FAILED—The real server has failed as a result of either no response or reset (RST) responses to client traffic. (See the faildetect (real server) command for more information about controlling tolerance for no responses and RSTs.) The real server has been removed from use by the predictor algorithms. The retry timer has started.</td>
</tr>
<tr>
<td></td>
<td>• MAXCONNS_THROTTLE—The number of connections on the real server exceeds the configured maximum number of simultaneous active connections (maxconns).</td>
</tr>
<tr>
<td></td>
<td>• OPERATIONAL—The real server is functioning properly and is being used for load-balancing.</td>
</tr>
<tr>
<td></td>
<td>• OPER_WAIT—The real server is waiting to become operational (waiting for a timeout or some other condition to be met).</td>
</tr>
<tr>
<td></td>
<td>• OUTOFSERVICE—The real server was configured with no inservice and has been removed from the load-balancing predictor lists.</td>
</tr>
<tr>
<td></td>
<td>• PROBE_FAILED—The probe has succeeded in the past but has currently failed. This failure might occur at the same time user connections fail, or it might not.</td>
</tr>
<tr>
<td></td>
<td>• PROBE_TESTING—The probe has never succeeded, due to no response. The initial probe timed out waiting for a success.</td>
</tr>
<tr>
<td></td>
<td>• READY_TO_TEST—The real server is queued for testing after being in FAILED state until the retry timer expired.</td>
</tr>
<tr>
<td></td>
<td>• TESTING—The real server is queued for assignment. When a single user connection is assigned to a real server that is in READY_TO_TEST state, the real server is placed in TESTING state. If the test succeeds, the real server is placed back in OPERATIONAL state.</td>
</tr>
<tr>
<td></td>
<td>• TEST_WAIT—The real server is waiting to begin testing (waiting for a timeout or some other condition to be met).</td>
</tr>
<tr>
<td>conns</td>
<td>Number of connections associated with the real server.</td>
</tr>
<tr>
<td></td>
<td>In general packet radio service (GPRS) load balancing, number of sessions associated with the real server.</td>
</tr>
<tr>
<td></td>
<td>In per-packet server load balancing, number of request packets that have been load balanced to each real server, using the connection count.</td>
</tr>
</tbody>
</table>
The following is sample output from the `show ip slb reals detail` command for a real server in a server farm:

Router# `show ip slb reals detail`

10.10.1.7, S, state = OPERATIONAL, type = server
  conns = 0, dummy_conns = 0, maxconns = 4294967295
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  reassign = 3, retry = 60
  failconn threshold = 8, failconn count = 0
  failclient threshold = 2, failclient count = 0
  total conns established = 0, total conn failures = 0
  server failures = 0

The following is sample output from the `show ip slb reals detail` command for a real server in a firewall farm:

Router# `show ip slb reals detail`

10.10.3.2, F, state = OPERATIONAL, type = firewall
  conns = 0, dummy_conns = 0, maxconns = 4294967295
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 8377, hash count = 0
  server failures = 0
  interface FastEthernet1/0, MAC 0000.0c41.1063

Table 12 describes the fields shown in the above detail displays.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address</td>
<td>IP address of the real server about which information is being displayed.</td>
</tr>
<tr>
<td></td>
<td>Used to identify each real server. Information about each real server is</td>
</tr>
<tr>
<td></td>
<td>displayed on a separate line.</td>
</tr>
<tr>
<td>farm name</td>
<td>Name of the server farm or firewall farm with which the real server is</td>
</tr>
<tr>
<td></td>
<td>associated.</td>
</tr>
</tbody>
</table>

Cisco IOS Release 12.2(14)ZA2
Table 12  show ip slb reals detail Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>Current state of the real server.</td>
</tr>
<tr>
<td></td>
<td>• DFP_THROTTLED—The Dynamic Feedback Protocol (DFP) agent sent a weight of 0 for this real server.</td>
</tr>
<tr>
<td></td>
<td>• FAILED—The real server has failed as a result of either no response or reset (RST) responses to client traffic. (See the faildetect (real server) command for more information about controlling tolerance for no responses and RSTs.) The real server has been removed from use by the predictor algorithms. The retry timer has started.</td>
</tr>
<tr>
<td></td>
<td>• MAXCONNS_THROTTLE—The number of connections on the real server exceeds the configured maximum number of simultaneous active connections (maxconns).</td>
</tr>
<tr>
<td></td>
<td>• OPERATIONAL—The real server is functioning properly and is being used for load-balancing.</td>
</tr>
<tr>
<td></td>
<td>• OPER_WAIT—The real server is waiting to become operational (waiting for a timeout or some other condition to be met).</td>
</tr>
<tr>
<td></td>
<td>• OUTOFSERVICE—The real server was configured with no inservice and has been removed from the load-balancing predictor lists.</td>
</tr>
<tr>
<td></td>
<td>• PROBE_FAILED—The probe has succeeded in the past but has currently failed. This failure might occur at the same time user connections fail, or it might not.</td>
</tr>
<tr>
<td></td>
<td>• PROBE_TESTING—The probe has never succeeded, due to no response. The initial probe timed out waiting for a success.</td>
</tr>
<tr>
<td></td>
<td>• READY_TO_TEST—The real server is queued for testing after being in FAILED state until the retry timer expired.</td>
</tr>
<tr>
<td></td>
<td>• TESTING—The real server is queued for assignment. When a single user connection is assigned to a real server that is in READY_TO_TEST state, the real server is placed in TESTING state. If the test succeeds, the real server is placed back in OPERATIONAL state.</td>
</tr>
<tr>
<td></td>
<td>• TEST_WAIT—The real server is waiting to begin testing (waiting for a timeout or some other condition to be met).</td>
</tr>
<tr>
<td>type</td>
<td>Indicates whether the real server is associated with a server farm (server) or firewall farm (firewall).</td>
</tr>
<tr>
<td>conns</td>
<td>Number of connections associated with the real server.</td>
</tr>
<tr>
<td></td>
<td>In general packet radio service (GPRS) load balancing, number of sessions associated with the real server.</td>
</tr>
<tr>
<td></td>
<td>In per-packet server load balancing, number of request packets that have been load balanced to each real server, using the connection count.</td>
</tr>
<tr>
<td>dummy_conns</td>
<td>Internal counter used in debugging.</td>
</tr>
<tr>
<td>maxconns</td>
<td>Maximum number of active connections allowed on the real server at one time.</td>
</tr>
<tr>
<td>weight</td>
<td>Weight assigned to the real server. The weight identifies the real server’s capacity, relative to other real servers in the server farm. This value could be changed by DFP.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>weight(admin)</td>
<td>Configured (or default) weight assigned to the real server.</td>
</tr>
<tr>
<td>metric</td>
<td>Internal counter used in debugging.</td>
</tr>
<tr>
<td>remainder</td>
<td>Internal counter used in debugging.</td>
</tr>
<tr>
<td>reassign</td>
<td>Total number of consecutive unacknowledged SYNchronize sequence numbers (SYNs) or Create Packet Data Protocol (PDP) requests since the last time the <code>clear ip slb counters</code> command was issued.</td>
</tr>
<tr>
<td>retry</td>
<td>Interval, in seconds, to wait between the detection of a failure on the real server and the next attempt to connect to the server.</td>
</tr>
<tr>
<td>failconn threshold</td>
<td>Maximum number of consecutive connection failures allowed before the real server is considered to have failed.</td>
</tr>
<tr>
<td>failconn count</td>
<td>Total number of consecutive connection failures since the last time the <code>clear ip slb counters</code> command was issued.</td>
</tr>
<tr>
<td>failclient threshold</td>
<td>Maximum number of unique client connection failures allowed before the real server is considered to have failed.</td>
</tr>
<tr>
<td>failclient count</td>
<td>Total number of unique client connection failures since the last time the <code>clear ip slb counters</code> command was issued.</td>
</tr>
<tr>
<td>total conns established</td>
<td>Total number of successful connection assignments since the last time the <code>clear ip slb counters</code> command was issued.</td>
</tr>
<tr>
<td>total conn failures</td>
<td>Total number of unsuccessful connection assignments since the last time the <code>clear ip slb counters</code> command was issued.</td>
</tr>
<tr>
<td>server failures</td>
<td>Total number of times this real server has been marked failed.</td>
</tr>
<tr>
<td>hash count</td>
<td>Total number of times the hash algorithm has been called.</td>
</tr>
<tr>
<td>interface</td>
<td>Type of interface.</td>
</tr>
<tr>
<td>MAC</td>
<td>MAC address of the firewall.</td>
</tr>
</tbody>
</table>
show ip slb replicate

To display the IOS Server Load Balancing (IOS SLB) replication configuration, use the `show ip slb replicate` command in privileged EXEC mode.

```
show ip slb replicate
```

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(2)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the `show ip slb replicate` command:

```
Router# show ip slb replicate
VS1, local = 10.10.99.132 remote = 10.10.99.99 port = 1024
   current password = none pending password = none
   password timeout = 180 sec (Default)
   unsent conn updates: 0
   conn updates received: 32
   conn updates transmitted: 471
   update packets received: 12
   update packets transmitted: 34
   failovers: 0
```

Table 13 describes the fields shown in the display.

```
Table 13  show ip slb replicate Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>local</td>
<td>Listening IP address for state exchange messages that are advertised.</td>
</tr>
<tr>
<td>remote</td>
<td>Destination IP address for all state exchange signals.</td>
</tr>
<tr>
<td>port</td>
<td>TCP or User Datagram Protocol (UDP) port number or port name for all state exchange signals.</td>
</tr>
<tr>
<td>current password</td>
<td>Current password for Message Digest Algorithm Version 5 (MD5) authentication, if any.</td>
</tr>
<tr>
<td>pending password</td>
<td>Pending password for MD5 authentication, if any.</td>
</tr>
<tr>
<td>unsent conn updates</td>
<td>Number of connection updates waiting to be sent.</td>
</tr>
<tr>
<td>conn updates received</td>
<td>Number of connection updates received.</td>
</tr>
<tr>
<td>conn updates transmitted</td>
<td>Number of connection updates sent.</td>
</tr>
<tr>
<td>update packets received</td>
<td>Number of connection update packets received.</td>
</tr>
</tbody>
</table>
```
Table 13  show ip slb replicate Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>update packets transmitted</td>
<td>Number of connection update packets sent.</td>
</tr>
<tr>
<td>failovers</td>
<td>Number of failovers detected.</td>
</tr>
</tbody>
</table>

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>request (HTTP probe)</td>
<td>Configures an HTTP probe to check the status of the real servers.</td>
</tr>
</tbody>
</table>
show ip slb serverfarms

To display information about the server farms, use the `show ip slb serverfarms` command in privileged EXEC mode.

```
show ip slb serverfarms [name server-farm] [detail]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name server-farm</td>
<td>(Optional) Displays information about the specified server farm.</td>
</tr>
<tr>
<td>detail</td>
<td>(Optional) Displays detailed server farm information.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ip slb serverfarms` command:

```
Router# show ip slb serverfarms

+--------+---------+-------+-------+
<table>
<thead>
<tr>
<th>server farm</th>
<th>predictor</th>
<th>reals</th>
<th>bind id</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRAG</td>
<td>ROUNDROBIN</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>LINUX</td>
<td>ROUNDROBIN</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>SRE</td>
<td>ROUNDROBIN</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>TEST</td>
<td>ROUNDROBIN</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

```

Table 14 describes the fields shown in the display.

**Table 14  show ip slb serverfarms Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>server farm</td>
<td>Name of the server farm about which information is being displayed. Information about each server farm is displayed on a separate line.</td>
</tr>
<tr>
<td>predictor</td>
<td>Type of load-balancing algorithm (ROUNDROBIN or LEASTCONNS) used by the server farm.</td>
</tr>
<tr>
<td>reals</td>
<td>Number of real servers configured in the server farm.</td>
</tr>
<tr>
<td>bind id</td>
<td>Bind ID configured on the server farm.</td>
</tr>
</tbody>
</table>
show ip slb sessions

To display information about sessions handled by IOS Server Load Balancing (IOS SLB), use the show ip slb sessions command in privileged EXEC mode.

```
show ip slb sessions [gtp | gtp-inspect | ipmobile | radius] [vserver virtual-server] [client ip-address netmask] [detail]
```

### Syntax Description

- **gtp** (Optional) Displays information about general packet radio service (GPRS) Tunneling Protocol (GTP) sessions being handled by IOS SLB.
- **gtp-inspect** (Optional) Displays information about GTP sessions being handled by IOS SLB that have GTP cause code inspection enabled.
- **ipmobile** (Optional) Displays information about Mobile IP sessions being handled by IOS SLB.
- **radius** (Optional) Displays information about RADIUS sessions being handled by IOS SLB.
- **vserver virtual-server** (Optional) Displays information about sessions being handled by the specified virtual server.
- **client ip-address netmask** (Optional) Displays information about sessions associated with the specified client IP address or subnet.
- **detail** (Optional) Displays detailed information.

### Command Modes

Privileged EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
<tr>
<td>12.1(13)E3</td>
<td>The <em>gtp</em> and <em>gtp-inspect</em> keywords were added.</td>
</tr>
<tr>
<td>12.2(14)ZA2</td>
<td>The <em>ipmobile</em> keyword was added.</td>
</tr>
</tbody>
</table>

### Examples

The following is sample output from the show ip slb sessions command for RADIUS sessions:

```
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
```

*Table 15* describes the fields shown in the display.
The following example shows IOS SLB GTP session data:

```
Router# show ip slb sessions gtp
```

```
vserver    key              client           real                  state
----------------------------------------------------------------------------------
10.10.10.10 1234567890123456 5.5.5.5          10.10.1.1             GTP_ESTAB
```

Table 16 describes the fields shown in the display.

The following example shows IOS SLB Mobile IP session data:

```
Router# show ip slb sessions ipmobile
```

```
vserver        NAI hash          client          real               retries
---------------------------------------------------------------------------
VIRTUAL_HA     0xFFFF            1.1.1.1/434     10.10.1.1          1
```

Table 17 describes the fields shown in the display.
### Table 17  `show ip slb sessions ipmobile` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vserver</td>
<td>Name of the virtual server whose Mobile IP sessions are being monitored and displayed. Information about each session is displayed on a separate line.</td>
</tr>
<tr>
<td>NAI hash</td>
<td>Network access identifier (NAI) in the Registration Request (RRQ), used by IOS SLB as a unique identifier.</td>
</tr>
<tr>
<td>client</td>
<td>Client IP address being used by the Mobile IP session.</td>
</tr>
<tr>
<td>real</td>
<td>Real IP address of the Mobile IP session.</td>
</tr>
<tr>
<td>retries</td>
<td>Number of foreign agent retries for the Mobile IP session.</td>
</tr>
</tbody>
</table>
show ip slb static

To display the IOS Server Load Balancing (IOS SLB) server Network Address Translation (NAT) configuration, use the **show ip slb static** command in privileged EXEC mode.

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

The default behavior is to display the entire IOS SLB server NAT configuration.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the **show ip slb static** 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
```
Table 18 describes the fields shown in the display.

### Table 18  `show ip slb static Field Descriptions`

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real</td>
<td>IP address of the real server.</td>
</tr>
<tr>
<td>action</td>
<td>Action to be taken by the real server:</td>
</tr>
<tr>
<td></td>
<td>• drop—The real server is configured to have its packets dropped by IOS SLB, if the packets do not correspond to existing connections.</td>
</tr>
<tr>
<td></td>
<td>• NAT—The real server is configured to use server NAT, and to use its own virtual IP address when translating addresses.</td>
</tr>
<tr>
<td></td>
<td>• NAT per-packet—The real server is configured to use server NAT and per-packet server load balancing.</td>
</tr>
<tr>
<td></td>
<td>• NAT sticky—The real server is configured to use server NAT for sticky connections.</td>
</tr>
<tr>
<td></td>
<td>• pass-thru—The real server is not configured to use server NAT.</td>
</tr>
<tr>
<td>address</td>
<td>Virtual IP address used by the real server when translating addresses using server NAT. Address 0.0.0.0 means the real server is not configured for server NAT.</td>
</tr>
<tr>
<td>counter</td>
<td>For actions drop and NAT per-packet, indicates the number of packets processed by the real server.</td>
</tr>
<tr>
<td></td>
<td>For actions NAT and NAT sticky, indicates the number of packets received by, but not necessarily processed by, the real server.</td>
</tr>
</tbody>
</table>
show ip slb stats

To display IOS Server Load Balancing (IOS SLB) statistics, use the **show ip slb stats** command in privileged EXEC mode.

```
show ip slb stats
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(9)E</td>
<td>This command was modified to support general packet radio service (GPRS) load balancing.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the **show ip slb stats** command:

```
Router# show ip slb stats
Pkts via normal switching: 0
Pkts via special switching: 12534087
Pkts via slb routing: 0
Pkts dropped: 0
Connections Created: 12466960
Connections Established: 12240100
Connections Destroyed: 12466960
Connections Reassigned: 0
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
```
Table 19 describes the fields shown in the display.

**Table 19  show ip slb stats Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pkts via normal switching</td>
<td>Number of packets handled by IOS SLB via normal switching since the last time counters were cleared. Normal switching is when IOS SLB packets are handled on normal IOS switching paths (CEF, fast switching, and process level switching).</td>
</tr>
<tr>
<td>Pkts via special switching</td>
<td>Number of packets handled by IOS SLB via special switching since the last time counters were cleared. Special switching is when IOS SLB packets are handled on hardware-assisted switching paths.</td>
</tr>
<tr>
<td>Pkts via slb routing</td>
<td>Number of packets handled by IOS SLB via SLB routing since the last time counters were cleared.</td>
</tr>
<tr>
<td>Pkts dropped</td>
<td>Number of packets dropped or consumed by IOS SLB since the last time counters were cleared.</td>
</tr>
<tr>
<td>Connections Created</td>
<td>Number of connections (or sessions, in general packet radio service [GPRS] load balancing and the Home Agent Director) created since the last time counters were cleared.</td>
</tr>
<tr>
<td>Connections Established</td>
<td>Number of connections (or sessions, in GPRS load balancing and the Home Agent Director) created and that have become established since the last time counters were cleared.</td>
</tr>
<tr>
<td>Connections Destroyed</td>
<td>Number of connections (or sessions, in GPRS load balancing and the Home Agent Director) destroyed since the last time counters were cleared.</td>
</tr>
<tr>
<td>Connections Reassigned</td>
<td>Number of connections (or sessions, in GPRS load balancing and the Home Agent Director) reassigned to a different real server since the last time counters were cleared.</td>
</tr>
<tr>
<td>Zombie Count</td>
<td>Number of connections (or sessions, in GPRS load balancing and the Home Agent Director) that are currently pending destruction (awaiting a timeout or some other condition to be met).</td>
</tr>
<tr>
<td>Connections Reused</td>
<td>Number of zombie connections (or sessions, in GPRS load balancing and the Home Agent Director) reused since the last time counters were cleared. A zombie connection is reused if it receives a TCP SYNchronize sequence number (SYN) or User Datagram Protocol (UDP) packet and succeeds in connecting to a real server. The zombie connection becomes a real connection and the zombie count is decremented.</td>
</tr>
<tr>
<td>Connection Flowcache Purges</td>
<td>Number of times the connection flow cache was purged since the last time counters were cleared.</td>
</tr>
<tr>
<td>Failed Connection Allocs</td>
<td>Number of times the allocation of a connection (or session, in GPRS load balancing) failed since the last time counters were cleared.</td>
</tr>
<tr>
<td>Failed Real Assignments</td>
<td>Number of times the assignment of a real server failed since the last time counters were cleared.</td>
</tr>
</tbody>
</table>
### Table 19  `show ip slb stats` Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RADIUS framed-ip Sticky Count</td>
<td>Number of entries in the RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td>RADIUS username Sticky Count</td>
<td>Number of entries in the RADIUS username sticky database.</td>
</tr>
</tbody>
</table>
show ip slb sticky

To display the IOS Server Load Balancing (IOS SLB) sticky database, use the **show ip slb sticky** command in privileged EXEC mode.

```
show ip slb sticky [client ip-address netmask] | radius framed-ip [client ip-address netmask] | radius username [name string]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>client ip-address netmask</td>
<td>(Optional) Displays only those sticky database entries associated with the specified client IP address or subnet.</td>
</tr>
<tr>
<td>radius framed-ip</td>
<td>(Optional) Displays only entries associated with the IOS SLB RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td>radius username</td>
<td>(Optional) Displays only entries associated with the IOS SLB RADIUS username sticky database.</td>
</tr>
<tr>
<td>name string</td>
<td>(Optional) Displays only those sticky database entries associated with the username.</td>
</tr>
</tbody>
</table>

**Command Modes**

Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>The <strong>radius</strong> keyword was added.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <strong>framed-ip</strong>, <strong>username</strong>, <strong>name</strong>, <strong>netmask</strong>, and <strong>string</strong> keywords and arguments were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no options are specified, the command displays information about all virtual servers or firewall farms.

**Examples**

The following is sample output from the **show ip slb sticky** 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
```
Table 20 describes the fields shown in the display.

Table 20  *show ip slb sticky Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>client</td>
<td>Client IP address or subnet which is bound to this sticky assignment.</td>
</tr>
<tr>
<td>netmask</td>
<td>Subnet mask for this sticky assignment.</td>
</tr>
<tr>
<td>group</td>
<td>Group ID for this sticky assignment.</td>
</tr>
<tr>
<td>real</td>
<td>Real server used by all clients connecting with the client IP address or subnet detailed on this line.</td>
</tr>
<tr>
<td>conns</td>
<td>Number of connections currently sharing this sticky assignment.</td>
</tr>
</tbody>
</table>

The following is sample output from the *show ip slb sticky radius framed-ip* command:

```
Router# show ip slb sticky radius framed-ip
framed-ip       group id     server real route i/f
-----------------------------------------------------
1.1.1.1         15           10.10.10.1   <any>
```

Table 21 describes the fields shown in the display.

Table 21  *show ip slb sticky radius framed-ip Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>framed-ip</td>
<td>IP address bound to a Cisco Service Selection Gateway (SSG) RADIUS proxy in the IOS SLB RADIUS framed-IP sticky database.</td>
</tr>
<tr>
<td>group id</td>
<td>Group ID for this sticky assignment.</td>
</tr>
<tr>
<td>server real</td>
<td>IP address of the SSG RADIUS proxy server.</td>
</tr>
<tr>
<td>route i/f</td>
<td>Route interface.</td>
</tr>
</tbody>
</table>

The following is sample output from the *show ip slb sticky radius username* command:

```
Router# show ip slb sticky radius username
username        group id     server real framed-ips
-----------------------------------------------------
9198783355      15           10.10.10.1   1
```

Table 22 describes the fields shown in the display.

Table 22  *show ip slb sticky radius username Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>Username bound to an SSG RADIUS proxy in the IOS SLB RADIUS username sticky database.</td>
</tr>
<tr>
<td>group id</td>
<td>Group ID for this sticky assignment.</td>
</tr>
<tr>
<td>server real</td>
<td>IP address of the SSG RADIUS proxy server.</td>
</tr>
<tr>
<td>framed-ips</td>
<td>Number of IP addresses bound to the SSG RADIUS proxy in the IOS SLB RADIUS framed-IP sticky database.</td>
</tr>
</tbody>
</table>
**show ip slb vserver**

To display information about the virtual servers, use the `show ip slb vserver` command in privileged EXEC mode.

```
show ip slb vserver [name virtual-server] [redirect] [detail]
```

### Syntax Description

- **name virtual-server** *(Optional)* Displays information about the specified virtual server.
- **redirect** *(Optional)* Displays information about redirect virtual servers.
- **detail** *(Optional)* Displays detailed information.

### Command Modes

Privileged EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

If no options are specified, the command displays information about all virtual servers.

### Examples

The following is sample output from the `show ip slb vserver` command:

```
Router# show ip slb vserver

slb vserver prot virtual state       conns
-----------------------------------------
TEST     TCP 80.80.254.3:80 OPERATIONAL 1013
TEST21   TCP 80.80.254.3:21 OUTOFSERVICE 0
TEST23   TCP 80.80.254.3:23 OUTOFSERVICE 0
```

*Table 23* describes the fields shown in the display.

### Table 23  _show ip slb vserver Field Descriptions_

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>slb vserver</td>
<td>Name of the virtual server about which information is being displayed. Information about each virtual server is displayed on a separate line.</td>
</tr>
<tr>
<td>prot</td>
<td>Protocol being used by the virtual server.</td>
</tr>
<tr>
<td>virtual</td>
<td>Virtual IP address of the virtual server, including the network mask, if configured.</td>
</tr>
</tbody>
</table>
The following sample output from the `show ip slb vserver detail` command shows detailed data for a virtual server with route health injection (advertise=TRUE):

```
Router# show ip slb vserver detail
RH1, state = OPERATIONAL, v_index = 6
  virtual = 5.5.5.5/32:80, TCP, service = NONE, advertise = TRUE
  server farm = RHSF, delay = 10, idle = 3600
  backup server farm = BACKUP, use count = 0, backup sticky = FALSE
  sticky timer = 0, sticky subnet = 255.255.255.255
  sticky group id = 0
  synguard counter = 0, synguard period = 0
  conns = 1, total conns = 31484, syns = 0, syn drops = 0
  standby group = None
```

Table 24 describes the fields shown in the display.

**Table 24  show ip slb vserver detail Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RH1</td>
<td>Name of the virtual server about which information is being displayed (in this case, RH1). Information about each virtual server is displayed on a separate line.</td>
</tr>
<tr>
<td>state</td>
<td>Current state of the virtual server:</td>
</tr>
<tr>
<td></td>
<td>FAILED—Real server represented by this virtual server has been removed from use by the predictor algorithms; retry timer started.</td>
</tr>
<tr>
<td></td>
<td>OPERATIONAL—Functioning properly.</td>
</tr>
<tr>
<td></td>
<td>OUTOFSERVICE—Removed from the load-balancing predictor lists.</td>
</tr>
<tr>
<td></td>
<td>STANDBY—Backup virtual server, ready to become operational if active virtual server fails.</td>
</tr>
<tr>
<td>v_index</td>
<td>Virtual index, out of a maximum of 500.</td>
</tr>
<tr>
<td>virtual</td>
<td>Virtual IP address of the virtual server, including the network mask, if configured.</td>
</tr>
<tr>
<td>TCP</td>
<td>Protocol being used by the virtual server (in this case, TCP).</td>
</tr>
<tr>
<td>service</td>
<td>Service, such as HTTP or Telnet, associated with the virtual server.</td>
</tr>
<tr>
<td>advertise</td>
<td>Current state of host route advertisement for this virtual server:</td>
</tr>
<tr>
<td></td>
<td>TRUE—Host route is being advertised.</td>
</tr>
<tr>
<td></td>
<td>FALSE—Host route is not being advertised.</td>
</tr>
</tbody>
</table>
### show ip slb vserver detail Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>server farm</td>
<td>Name of the server farm associated with the virtual server.</td>
</tr>
<tr>
<td>delay</td>
<td>Delay timer duration, in seconds, for this virtual server.</td>
</tr>
<tr>
<td>idle</td>
<td>Idle connection timer duration, in seconds, for this virtual server.</td>
</tr>
<tr>
<td>backup server farm</td>
<td>Name of the backup server farm associated with the virtual server.</td>
</tr>
<tr>
<td>use count</td>
<td>Number of times the backup server farm has taken over for the primary server farm in this period.</td>
</tr>
<tr>
<td>backup sticky</td>
<td>Indicates whether sticky connections are used in the backup server farm:</td>
</tr>
<tr>
<td></td>
<td>- TRUE—Sticky connections are used in the backup server farm.</td>
</tr>
<tr>
<td></td>
<td>- FALSE—Sticky connections are not used in the backup server farm.</td>
</tr>
<tr>
<td>sticky timer</td>
<td>Sticky timer duration, in seconds, for this virtual server.</td>
</tr>
<tr>
<td>sticky subnet</td>
<td>Sticky subnet in which this virtual server is placed, for coupling of services.</td>
</tr>
<tr>
<td>sticky group id</td>
<td>Sticky group in which this virtual server is placed, for coupling of services.</td>
</tr>
<tr>
<td>synguard counter</td>
<td>Number of unacknowledged SYNchronize sequence numbers (SYNs) that are allowed to be outstanding to this virtual server.</td>
</tr>
<tr>
<td>synguard period</td>
<td>Interval, in milliseconds, for SYN threshold monitoring for this virtual server.</td>
</tr>
<tr>
<td>conns</td>
<td>Number of active connections currently associated with the virtual server.</td>
</tr>
<tr>
<td>total conns</td>
<td>Total number of connections that have been associated with the virtual server since coming INSERVICE.</td>
</tr>
<tr>
<td>syns</td>
<td>Number of SYNs handled by the virtual server in this period.</td>
</tr>
<tr>
<td>syn drops</td>
<td>Number of SYNs dropped by the virtual server in this period.</td>
</tr>
<tr>
<td>standby group</td>
<td>Hot Standby Router Protocol (HSRP) group name with which the virtual server is associated.</td>
</tr>
</tbody>
</table>
snmp-server enable traps slb

To enable IOS SLB traps for real- and virtual-server state changes, use the `snmp-server enable traps slb` global configuration command. To disable the traps use the `no` form of this command.

```
  snmp-server enable traps slb {real | virtual}
  no snmp-server enable traps slb {real | virtual}
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>real</th>
<th>Enables traps for real server state changes.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>virtual</td>
<td>Enables traps for virtual server state changes.</td>
</tr>
</tbody>
</table>

Defaults

IOS SLB traps for real- and virtual-server state changes are not enabled.

Command Modes

Global configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(11b)E</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Examples

The following example enables IOS SLB traps for real server state changes:

```
Router(config)# snmp-server enable traps slb real
```
To assign all connections from a client to the same firewall, use the sticky command in firewall farm datagram protocol configuration mode. To remove the client/server coupling, use the no form of this command.

```
sticky duration [netmask netmask] [source | destination]
```

```
no sticky
```

### Syntax Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>duration</code></td>
<td>Sticky timer duration in seconds. Valid values range from 0 to 65535.</td>
</tr>
<tr>
<td><code>netmask</code></td>
<td>(Optional) Places the virtual server as part of a sticky subnet, for coupling of services.</td>
</tr>
<tr>
<td><code>source</code></td>
<td>(Optional) Bases sticky on source IP address.</td>
</tr>
<tr>
<td><code>destination</code></td>
<td>(Optional) Bases sticky on destination IP address.</td>
</tr>
</tbody>
</table>

### Defaults

Virtual servers are not associated with any groups.

### Command Modes

Firewall farm datagram protocol configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(12c)E</td>
<td>The <code>source</code> and <code>destination</code> keywords were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

### Examples

The following example specifies that if a client’s subsequent request for a firewall farm is made within 60 seconds of the previous request, then the same firewall is used for the connection:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol datagram
Router(config-slb-fw-udp)# sticky 60
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protocol datagram</code></td>
<td>Enters firewall farm datagram protocol configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td><code>show ip slb sticky</code></td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) database.</td>
</tr>
</tbody>
</table>
sticky (firewall farm TCP protocol)

To assign all connections from a client to the same firewall, use the `sticky` command in firewall farm TCP protocol configuration mode. To remove the client/server coupling, use the `no` form of this command.

```
sticky duration [netmask netmask] [source | destination]
```

```
no sticky
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>duration</code></td>
<td>Sticky timer duration in seconds. Valid values range from 0 to 65535.</td>
</tr>
<tr>
<td><code>netmask</code></td>
<td>(Optional) Places the virtual server as part of a sticky subnet, for coupling of services.</td>
</tr>
<tr>
<td><code>source</code></td>
<td>(Optional) Bases sticky on source IP address.</td>
</tr>
<tr>
<td><code>destination</code></td>
<td>(Optional) Bases sticky on destination IP address.</td>
</tr>
</tbody>
</table>

**Defaults**

Virtual servers are not associated with any groups.

**Command Modes**

Firewall farm TCP protocol configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(12c)E</td>
<td>The <code>source</code> and <code>destination</code> keywords were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example specifies that if a client’s subsequent request for a firewall farm is made within 60 seconds of the previous request, then the same firewall is used for the connection:

```
Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# protocol tcp
Router(config-slb-fw-tcp)# sticky 60
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protocol tcp</code></td>
<td>Enters firewall farm TCP protocol configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb firewallfarm</code></td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td><code>show ip slb sticky</code></td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) database.</td>
</tr>
</tbody>
</table>
sticky (virtual server)

To assign all connections from a client to the same real server, use the **sticky** command in virtual server configuration mode. To remove the client/server coupling, use the **no** form of this command.

```
sticky { duration [group group-id] [netmask netmask] | radius framed-ip [group group-id] | radius username [msid-cisco] [group group-id] }
```

```
no sticky { duration [group group-id] [netmask netmask] | radius framed-ip [group group-id] | radius username [msid-cisco] [group group-id] }
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>duration</strong></td>
<td>Sticky timer duration in seconds. Valid values range from 0 to 65535.</td>
</tr>
<tr>
<td><strong>group group-id</strong></td>
<td>(Optional) Places the virtual server in the specified sticky group, for coupling of services. In essence, the <strong>group</strong> keyword and <strong>group-id</strong> argument tie multiple virtual servers together. Valid values range from 0 to 255.</td>
</tr>
<tr>
<td><strong>netmask netmask</strong></td>
<td>(Optional) Places the virtual server as part of the specified sticky subnet, for coupling of services. Client sessions whose source IP addresses fall within the <strong>netmask</strong> are directed to the same real server.</td>
</tr>
<tr>
<td><strong>radius framed-ip</strong></td>
<td>Enables IOS Server Load Balancing (IOS SLB) to create the IOS SLB RADIUS framed-IP sticky database and direct RADIUS requests and non-RADIUS flows from a given end user to the same service gateway.</td>
</tr>
<tr>
<td><strong>radius username</strong></td>
<td>Enables IOS SLB to create the IOS SLB RADIUS username sticky database and direct RADIUS requests from a given end user to the same service gateway.</td>
</tr>
<tr>
<td><strong>msid-cisco</strong></td>
<td>Enables IOS SLB to support Cisco PDSNs that provide MSID-based access (also known as MSID-based access, Cisco variant).</td>
</tr>
</tbody>
</table>

### Defaults

Sticky connections are not tracked.
Virtual servers are not associated with any groups.

### Command Modes

Virtual server configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(2)E</td>
<td>The <strong>netmask</strong> keyword and <strong>netmask</strong> argument were added.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>The <strong>radius framed-ip</strong> keywords were added.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <strong>radius username</strong> and <strong>msid-cisco</strong> keywords were added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>
Usage Guidelines

The last real server that was used for a connection from a client is stored for the set duration seconds. If a new connection from the client to the virtual server is initiated during that time, the same real server that was used for the previous connection is chosen for the new connection. If two virtual servers are placed in the same group, coincident connection requests for those services from the same IP address are handled by the same real server.

In Virtual Private Network (VPN) server load balancing, remember the following requirements:

- For IPSec flows, you must specify a sticky connection between the User Datagram Protocol (UDP) virtual server and the Encapsulation Security Payload (ESP) virtual server.
- For PPTP flows, you must specify a sticky connection between the TCP virtual server and the Generic Routing Encapsulation (GRE) virtual server.
- You must specify a duration of at least 15 seconds.

In general packet radio service (GPRS) load balancing and the Home Agent Director, the sticky command is not supported.

In RADIUS load balancing, remember the following requirements:

- If you configure the sticky radius framed-ip command, you must also configure the virtual command with the service radius keywords specified.
- If you configure 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.

Examples

The following example specifies that if a client’s subsequent request for a virtual server is made within 60 seconds of the previous request, then the same real server is used for the connection. This example also places the virtual server in group 10.

Router(config)# ip slb vserver VS1
Router(config-slb-vserver)# sticky 60 group 10

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip slb sticky</td>
<td>Displays information about the IOS Server Load Balancing (IOS SLB) database.</td>
</tr>
<tr>
<td>show ip slb vsserver</td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td>virtual (virtual server)</td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
synguard (virtual server)

To limit the rate of TCP SYNchronize sequence numbers (SYNs) handled by a virtual server to prevent a SYN flood denial-of-service attack, use the `synguard` command in virtual server configuration mode. To remove the threshold, use the `no` form of this command.

```
synguard syn-count [interval]
no synguard
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>syn-count</code></td>
<td>Number of unacknowledged SYNs that are allowed to be outstanding to a virtual server. Valid values range from 0 (off) to 4294967295. The default is 0.</td>
</tr>
<tr>
<td><code>interval</code></td>
<td>(Optional) Interval, in milliseconds, for SYN threshold monitoring. Valid values range from 50 to 5000. The default is 100 milliseconds (ms).</td>
</tr>
</tbody>
</table>

**Defaults**

The default number of unacknowledged SYNs that are allowed to be outstanding to a virtual server is 0 (off). The default interval is 100 ms.

**Command Modes**

Virtual server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

In general packet radio service (GPRS) load balancing and the Home Agent Director, the `synguard` command has no meaning and is not supported.

**Examples**

The following example sets the threshold of unacknowledged SYNs to 50:

```
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# synguard 50
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
<tr>
<td><code>virtual (virtual server)</code></td>
<td>Configures the virtual server attributes.</td>
</tr>
</tbody>
</table>
url (WSP probe)

To specify the URL path that a Wireless Session Protocol (WSP) probe is to request from the server, use the `url` command in WSP probe configuration mode. To restore the default settings, use the `no` form of this command.

```
url [path]
```

```
no url [path]
```

**Syntax Description**

- `path` (Optional) Path from the server. This argument is case-sensitive.

**Defaults**

If no URL path is specified, the default is `/`.

**Command Modes**

WSP probe configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(5a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Examples**

The following example configures a ping probe named PROBE3, enters WSP probe configuration mode, and configures the probe to request URL path `http://localhost/test.txt`:

```
Router(config)# ip slb probe PROBE3 wsp
Router(config-slb-probe)# url http://localhost/test.txt
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb probe wsp</code></td>
<td>Configures a Wireless Session Protocol (WSP) probe name and enters WSP probe configuration mode.</td>
</tr>
<tr>
<td><code>show ip slb probe</code></td>
<td>Displays information about an IOS Server Load Balancing (IOS SLB) probe.</td>
</tr>
</tbody>
</table>
virtual (virtual server)

To configure virtual server attributes, use the `virtual` command in virtual server configuration mode. To remove the attributes, use the `no` form of this command.

Encapsulation Security Payload (ESP) and Generic Routing Encapsulation (GRE) Protocols

```
virtual ip-address [netmask [group]] {esp | gre | protocol}
```

```
no virtual ip-address [netmask [group]] {esp | gre | protocol}
```

TCP and User Datagram Protocol (UDP)

```
virtual ip-address [netmask [group]] {tcp | udp} [port | any] [service service]
```

```
no virtual ip-address [netmask [group]] {tcp | udp} [port | any] [service service]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>IP address for this virtual server instance, used by clients to connect to the server farm.</td>
</tr>
<tr>
<td>netmask</td>
<td>(Optional) IP network mask for transparent web cache load balancing. The default is 0.0.0.0 (all subnets).</td>
</tr>
<tr>
<td>group</td>
<td>(Optional) Allows the virtual subnet to be advertised. If you do not specify the group keyword, the virtual subnet cannot be advertised.</td>
</tr>
<tr>
<td>esp</td>
<td>Performs load balancing for only Encapsulation Security Payload (ESP) connections.</td>
</tr>
<tr>
<td>gre</td>
<td>Performs load balancing for only Generic Routing Encapsulation (GRE) connections.</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol for which load balancing is performed. The valid range is 2 to 127.</td>
</tr>
<tr>
<td>tcp</td>
<td>Performs load balancing for only TCP connections.</td>
</tr>
<tr>
<td>udp</td>
<td>Performs load balancing for only User Datagram Protocol (UDP) connections.</td>
</tr>
</tbody>
</table>
**port**

(Optional) IOS Server Load Balancing (IOS SLB) virtual port (the TCP or UDP port number or port name). If specified, only the connections for the specified port on the server are load-balanced. The ports and the valid name or number for the `port` argument are as follows:

- All ports: any 0
- Connectionless secure Wireless Session Protocol (WSP): `wsp-wtls 9202`
- Connectionless WSP: `wsp 9200`
- Connection-oriented secure WSP: `wsp-wtp-wtls 9203`
- Connection-oriented WSP: `wsp-wtp 9201`
- Domain Name System: `dns 53`
- File Transfer Protocol: `ftp 21`
- General packet radio service (GPRS) tunneling protocol (GTP): `gtp 3386`
- HTTP over Secure Socket Layer: `https 443`
- Internet Key Exchange (IKE): `isakmp 500`
- Mapping of airline traffic over IP, Type A: `matip-a 350`
- Network News Transport Protocol: `nntp 119`
- Post Office Protocol v2: `pop2 109`
- Post Office Protocol v3: `pop3 110`
- Simple Mail Transport Protocol: `smtp 25`
- Telnet: `telnet 23`
- X.25 over TCP (XOT): `xot 1998`
- World Wide Web (HTTP): `www 80`

Specify a port number of 0 to configure an all-port virtual server (that is, a virtual server that accepts flows destined for all ports except GTP ports).

---

**any**

(Optional) Performs load balancing on all ports.

---

**service service**

(Optional) Couples connections associated with a given service, such as HTTP or Telnet, so all related connections from the same client use the same real server. The following are the valid types of connection coupling:

- `ftp`—Couples FTP data connections with the control session that created them.
- `gtp`—Enables GPRS load balancing without general packet radio service (GPRS) tunneling protocol (GTP) cause code inspection enabled, which allows load-balancing decisions to be made using Layer 5 information. You can balance UDP flows without awareness of GTP by omitting the `service gtp` keywords.
- `gtp-inspect`—Enables GPRS load balancing with GTP cause code inspection enabled.
- `ipmobile`—Enables the Home Agent Director.
- `per-packet`—Does not maintain connection objects for packets destined for this virtual server.
- `radius`—Enables IOS SLB to build RADIUS session objects for RADIUS load balancing.
**virtual (virtual server)**

**Defaults**
No default behavior or values

**Command Modes**
Virtual server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.1(5a)E</td>
<td>The <strong>wsp</strong>, <strong>wsp-wtp</strong>, <strong>wsp-wtls</strong>, and <strong>wsp-wtp-wtls</strong> keywords were added.</td>
</tr>
<tr>
<td>12.1(9)E</td>
<td>The <strong>gtp</strong> option was added as a new value on the <strong>service</strong> argument.</td>
</tr>
<tr>
<td>12.1(11b)E</td>
<td>The following keywords, arguments, and options were added:</td>
</tr>
<tr>
<td></td>
<td>- The <strong>esp</strong>, <strong>gre</strong>, and <strong>all</strong> keywords</td>
</tr>
<tr>
<td></td>
<td>- The <strong>protocol</strong> argument</td>
</tr>
<tr>
<td></td>
<td>- The <strong>isakmp</strong> option on the <strong>port</strong> argument</td>
</tr>
<tr>
<td></td>
<td>- The <strong>per-packet</strong> and <strong>radius</strong> options on the <strong>service</strong> argument</td>
</tr>
<tr>
<td></td>
<td>The <strong>wsp</strong>, <strong>wsp-wtp</strong>, <strong>wsp-wtls</strong>, and <strong>wsp-wtp-wtls</strong> keywords were changed to options for the <strong>port</strong> argument.</td>
</tr>
<tr>
<td>12.1(12c)E</td>
<td>The <strong>group</strong> keyword was added.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
<tr>
<td>12.1(13)E3</td>
<td>The <strong>gtp-inspect</strong> option was added as a new value on the <strong>service</strong> argument.</td>
</tr>
<tr>
<td>12.2(14)ZA2</td>
<td>The <strong>ipmobile</strong> option was added as a new value on the <strong>service</strong> argument.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The **no virtual** command is allowed only if the virtual server was removed from service by the **no inservice** command.

For some applications, it is not feasible to configure all the virtual server TCP or UDP port numbers for IOS SLB. To support such applications, you can configure IOS SLB virtual servers to accept flows destined for all ports. To configure an all-port virtual server, specify a port number of 0 or any.

In general, you should use port-bound virtual servers instead of all-port virtual servers. When you use all-port virtual servers, 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. To prevent this problem, use one of the following procedures:

- Configure one or more port-bound virtual servers.
- Configure access control lists on the IOS SLB ingress interface to permit only ports for supported applications on the real server.
- Configure **faildetect numconns 255 numclients 8** on the real server.

Specifying port 9201 for connection-oriented WSP mode also activates the Wireless Application Protocol (WAP) finite state machine (FSM), which monitors WAP and drives the session FSM accordingly.
In RADIUS load balancing, IOS SLB maintains session objects in a database to ensure that re-sent RADIUS requests are load-balanced to the same real server.

**Examples**

The following example specifies that the virtual server with the IP address 10.0.0.1 performs load balancing for TCP connections for the port named www. The virtual server processes HTTP requests.

```bash
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# virtual 10.0.0.1 tcp www
```

The following example specifies that the virtual server with the IP address 10.0.0.13 performs load balancing for UDP connections for all ports. The virtual server processes HTTP requests.

```bash
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# virtual 10.0.0.13 udp 0
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip slb vserver</code></td>
<td>Identifies a virtual server.</td>
</tr>
<tr>
<td><code>show ip slb vserver</code></td>
<td>Displays information about the virtual servers defined to IOS Server Load Balancing (IOS SLB).</td>
</tr>
</tbody>
</table>
weight (firewall farm real server)

To specify a real server’s capacity, relative to other real servers in the firewall farm, use the weight command in firewall farm real server configuration mode. To restore the default weight value, use the no form of this command.

weight setting

no weight

Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>setting</td>
<td>Weight setting to use for the real server predictor algorithm. Valid settings range from 1 to 255. The default weight setting is 8.</td>
</tr>
</tbody>
</table>

Defaults

The default setting to use for the real server predictor algorithm is 8.

Command Modes

Firewall farm real server configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1(3a)E</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

Examples

The following example specifies the relative weights of three real servers as 16, 8 (by default), and 24, respectively:

Router(config)# ip slb firewallfarm FIRE1
Router(config-slb-fw)# real 10.10.1.1
Router(config-slb-fw-real)# weight 16
Router(config-slb-fw-real)# inservice
Router(config-slb-fw-real)# exit
Router(config-slb-fw)# real 10.10.1.2
Router(config-slb-fw-real)# inservice
Router(config-slb-fw-real)# exit
Router(config-slb-fw)# real 10.10.1.3
Router(config-slb-fw-real)# weight 24

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real (server farm)</td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>show ip slb firewallfarm</td>
<td>Displays information about the firewall farm configuration.</td>
</tr>
<tr>
<td>show ip slb reals</td>
<td>Displays information about the real servers.</td>
</tr>
</tbody>
</table>
weight (server farm)

To specify a real server's capacity, relative to other real servers in the server farm, use the `weight` command in real server configuration mode. To restore the default weight value, use the `no` form of this command.

```
weight setting

no weight
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>setting</td>
<td>Weight setting to use for the real server predictor algorithm. Valid settings range from 1 to 255. The default weight setting is 8.</td>
</tr>
</tbody>
</table>

**Defaults**

The default setting to use for the real server predictor algorithm is 8.

**Command Modes**

Real server configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.0(7)XE</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1(5)T</td>
<td>This command was integrated into Cisco IOS Release 12.1(5)T.</td>
</tr>
<tr>
<td>12.2</td>
<td>This command was integrated into Cisco IOS Release 12.2.</td>
</tr>
<tr>
<td>12.2(14)S</td>
<td>This command was integrated into Cisco IOS Release 12.2(14)S.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The static weights you define using this command are overridden by the weights calculated by Dynamic Feedback Protocol (DFP). If DFP is removed from the network, IOS Server Load Balancing (IOS SLB) reverts to these static weights.
The following example specifies the relative weights of three real servers as 16, 8 (by default), and 24, respectively:

```bash
Router(config)# ip slb serverfarm PUBLIC
!-----First real server
Router(config-slb-sfarm)# real 10.10.1.1
!-----Assigned weight of 16
Router(config-slb-real)# weight 16
!-----Enabled
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
!-----Second real server
Router(config-slb-sfarm)# real 10.10.1.2
!-----Enabled with default weight
Router(config-slb-real)# inservice
Router(config-slb-real)# exit
!-----Third real server
Router(config-slb-sfarm)# real 10.10.1.3
!-----Assigned weight of 24, not enabled
Router(config-slb-real)# weight 24
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>real (server farm)</td>
<td>Identifies a real server by IP address and optional port number as a member of a server farm and enters real server configuration mode.</td>
</tr>
<tr>
<td>show ip slb reals</td>
<td>Displays information about the real servers.</td>
</tr>
<tr>
<td>show ip slb serverfarms</td>
<td>Displays information about the server farm configuration.</td>
</tr>
</tbody>
</table>
# FAQ (Frequently Asked Questions)

The following questions and answers can help you troubleshoot IOS SLB, if you have problems.

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
</table>
| 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. |
| Why is IOS SLB not balancing correctly? I am using dispatched mode, the servers are leaving sockets open, and I am seeing RSTs in response to a number of SYNs. Curiously, sometimes things work fine. | Enter the `show mls flow` command:  
```  
Router# show mls flow  
current ip flowmask for unicast: full flow  
current ipx flowmask for unicast: destination only  
```
  
The current IP flowmask must be full flow. If it is not, correct the problem using the `mls flow ip full` command:  
```  
Router# configure terminal  
Enter configuration commands, one per line.  
End with CNTL/Z.  
Router(config)# mls flow ip full  
Router(config)#  
```
<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>How can I verify that IOS SLB sticky connections are working properly?</td>
<td>Use the following procedure:</td>
</tr>
<tr>
<td></td>
<td>1. Configure the sticky connections.</td>
</tr>
<tr>
<td></td>
<td>2. Start a client connection.</td>
</tr>
<tr>
<td></td>
<td>3. Enter the <code>show ip slb reals detail</code> and <code>show ip slb conns</code> commands.</td>
</tr>
<tr>
<td></td>
<td>4. Examine the real server connection counts. The real server whose count increased is the one to which the client connection is assigned.</td>
</tr>
<tr>
<td></td>
<td>5. Enter the <code>show ip slb sticky</code> command to display the sticky relationships stored by IOS SLB.</td>
</tr>
<tr>
<td></td>
<td>6. End the connection.</td>
</tr>
<tr>
<td></td>
<td>7. Ensure that the real server’s connection count decreased.</td>
</tr>
<tr>
<td></td>
<td>8. Restart the connection, after waiting no longer than the sticky timeout value.</td>
</tr>
<tr>
<td></td>
<td>9. Enter the <code>show ip slb conns</code> command again.</td>
</tr>
<tr>
<td></td>
<td>10. Examine the real server connection counts again, and verify that the sticky connection is assigned to the same real server as before.</td>
</tr>
<tr>
<td>How can I verify that server failures are being detected correctly?</td>
<td>Use the following procedure:</td>
</tr>
<tr>
<td></td>
<td>1. Use a large client population. If the number of clients is very small, tune the <code>numclients</code> keyword on the <code>faildetect (real server)</code> command so that the servers are not displayed as FAILED.</td>
</tr>
<tr>
<td></td>
<td>2. Enter the <code>show ip slb reals detail</code> command to show the status of the real servers.</td>
</tr>
<tr>
<td></td>
<td>3. Examine the status and connection counts of the real servers.</td>
</tr>
<tr>
<td></td>
<td>– 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.</td>
</tr>
<tr>
<td></td>
<td>– 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 <code>reassign</code> 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.</td>
</tr>
<tr>
<td></td>
<td>– 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 22 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.</td>
</tr>
</tbody>
</table>
**Question**

Does the **no inservice** command take a resource out of service immediately?

**Answer**

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.

---

**Glossary**

**access control list**—See **ACL**.

**ACL**—access control list. Mechanism used to limit the kind of information clients can access, and limit what they can do with the information.

**active standby**—Redundancy scheme in which two IOS SLB devices can load-balance the same virtual IP address while at the same time acting as backups for each other. See also **stateful backup** and **stateless backup**.

**bearer network**—Network that carries messages of a transport-layer protocol, and ultimately also of the session-layer protocols, between physical devices. A single session can use more than one bearer network.

**CASA**—Cisco Application Services Architecture. CASA is a protocol designed to allow network appliances to selectively control the flow of IP packets through a router, switch, or other network device.

**CDMA**—Code Division Multiple Access. Digital spread-spectrum modulation technique used mainly with personal communication devices such as mobile phones.

**CDMA2000**—Third-generation (3-G) version of CDMA.

**client NAT**—Translation scheme in which the client IP address is replaced with an IP address associated with one of a group of load-balancing devices, resulting in outbound flows being routed to the correct device. See also **directed mode** and **server NAT**.

**client subsystem**—Users, such as IOS SLB or GPRS, of the DFP agent function.

**cluster**—Set of computer systems that are connected through multisystem hardware or software to supply services traditionally provided by a single system. This arrangement provides higher availability and better scalability of the system.

**Code Division Multiple Access**—See **CDMA**.

**content-aware networking**—Networking strategy that enables content to be dynamically distributed. Because content can be dynamically cached, it can be located at any given place at any given time and distributed between the servers and the location of the web cache. Cisco has developed the ContentFlow architecture and the DFP to enable networks to provide content-aware networking services.

**ContentFlow architecture**—Cisco’s content-aware networking architecture that describes message flows and actions in a distributed environment.
DFP—Dynamic Feedback Protocol. Allows host agents to dynamically report the change in status of the host systems providing a virtual service. The status reported is a relative weight that specifies a host server’s capacity to perform work.

DFP agent—Object in a load-balanced environment that dynamically reports changes in status of the host systems that provide a virtual service. The status reported is a relative weight that specifies a host server’s capacity to perform work. See also DFP manager.

DFP manager—Object in a load-balanced environment that collects status reports from DFP agents. See also DFP agent.

directed mode—Session redirection mode in which 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 real server, translating the virtual server IP address to a real server IP address through NAT. See also dispatched mode and NAT.

dispatched mode—Session redirection mode in which the virtual server address is known to the real servers. The virtual server IP address must be configured 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. See also directed mode.

Dynamic Feedback Protocol—See DFP.

Encapsulation Security Payload—See ESP.

ESP—Encapsulation Security Payload. Most common IPSec protocol, used for both encryption and data authentication.

FA—Foreign agent. Router on a mobile node's visited network which provides routing services to the mobile node while registered. The foreign agent detunnels and delivers datagrams to the mobile node that were tunneled by the mobile node's home agent. For datagrams sent by a mobile node, the foreign agent may serve as a default router for registered mobile nodes.

firewall—Router or access server, or several routers or access servers, designated as a buffer between any connected public networks and a private network. A firewall router uses access lists and other methods to ensure the security of the private network.

firewall farm—Group of firewalls.

firewall load balancing—Load-balancing scheme in which the network administrator configures a group of firewalls into a firewall farm. When a client initiates a connection, IOS SLB chooses a firewall for the connection based on a hash algorithm.

foreign agent—See FA.

gateway GPRS support node—See GGSN.

General packet radio service—See GPRS.

Generic Routing Encapsulation—See GRE.

GGSN—Gateway GPRS Support Node. A GPRS network entity that serves as the mobile wireless gateway between an SGSN and PDNs. The GGSN allows mobile wireless users to access PDNs. See also GPRS, GSM, GTP, and SGSN.

Global System for Mobile Communications—See GSM.

GPRS—General packet radio service. An ETSI standard that defines the implementation of packet data services on a GSM network. See also GGSN, GSM, GTP, and SGSN.

GPRS Tunneling Protocol—See GTP.
GSM—Global System for Mobile Communications. A second-generation (2G) mobile wireless networking standard defined by ETSI and deployed widely throughout the world. See also GGSN, GPRS, GTP, and SGSN.

GSN—GGSN or SGSN. See GGSN and SGSN.

GRE—generic routing encapsulation. Encapsulates flows with new packet headers to ensure delivery to specific destinations.

GTP—GPRS Tunneling Protocol. Protocol that handles the flow of user packet data and signaling information between the SGSN and GGSN in a GPRS network. GTP is defined on both the Gn and Gp interfaces of a GPRS network. See also GGSN, GPRS, GSM, and SGSN.

HA—Home agent. Router on a mobile node's home network which tunnels packets to the mobile node while it is away from home. It keeps current location information for registered mobile nodes called a mobility binding.

home address—IP address that is assigned for an extended time to a mobile node. It remains unchanged regardless of where the node is attached to the Internet.

home agent—See HA.

home network—The network or virtual network which matches the subnet address of the mobile node.

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. See also redundancy, stateful backup, and stateless backup.

HTTP redirect load balancing—Load-balancing scheme in which all HTTP requests that belong to the same transaction are directed to the same real server.

ICMP—Internet Control Message Protocol. Message control and error-reporting protocol between a host server and a gateway to the Internet. ICMP uses Internet Protocol (IP) datagrams, but the messages are processed by the IP software and are not directly apparent to the application user.

IKE—Internet Key Exchange Protocol. Defines the procedures for authenticating a communicating peer, creation and management of Security Associations, key generation techniques, and mitigation of threats (such as denial of service and replay attacks), all of which are necessary to establish and maintain secure communications in an Internet environment.

Internet Control Message Protocol—See ICMP.

Internet Key Exchange Protocol—See IKE.

IOS SLB—IOS Server Load Balancing. Load-balancing function in which 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. When a client initiates a connection to the virtual server, IOS SLB chooses a real server for the connection based on a configured load-balancing algorithm.

IP Security—See IPSec.

IPSec—IP Security. Family of protocols designed to provide security to packets sent over IP.

load balancing—Spreading user requests among available servers within a cluster of servers, based on a variety of algorithms.

MD5—Message Digest Algorithm Version 5. Neighbor router authentication scheme used to ensure reliability and security when routing updates are exchanged between neighbor routers.

Message Digest Algorithm Version 5—See MD5.
**mobile node**—A host or router that changes its point of attachment from one network or subnet to another. A mobile node may change its location without changing its IP address; it may continue to communicate with other Internet nodes at any location using its home IP address, assuming link-layer connectivity to a point of attachment is available.

**Mobile Station Identification**—See *MSID*.

**MSID**—Mobile Station Identification. A number that uniquely identifies a mobile station.

**NAS**—Network Access Server. Device providing local network access to users across a remote access network such as the PSTN.

**NAT**—Network Address Translation. Modification of one or more of the following fields in an IP packet: source IP address, destination IP address, source TCP/UDP port, destination TCP/UDP port. See also *client NAT*, and *server NAT*.

**NetFlow switching**—High-performance network-layer switching path that captures as part of its switching function a rich set of traffic statistics including user, protocol, port, and type of service information.

**Network Access Server**—See *NAS*.

**Network Address Translation**—See *NAT*.

**packet data protocol**—See *PDP*.

**Packet Data Serving Node**—See *PDSN*.

**PDP**—Packet data protocol. Network protocol used by external packet data networks that communicate with a GPRS network. IP is an example of a PDP supported by GPRS.

**PDSN**—Packet Data Serving Node. Cisco’s standards-compliant solution that enables packet data services in a CDMA environment. The PDSN acts as an access gateway for simple IP and mobile stations. It provides foreign agent support and packet transport for virtual private networking. It also acts as an authentication, authorization, and accounting (AAA) client.

**Point-to-Point Tunneling Protocol**—See *PPTP*.

**port-bound**—Server configuration scheme in which a virtual server IP address represents one set of real servers for one service, such as Hypertext Transfer Protocol (HTTP), and a different set of real servers for another service, such as Telnet.

**PPTP**—Point-to-Point Tunneling Protocol. Protocol used to ensure that messages sent from one VPN node to another are secure.

**RADIUS**—Remote Authentication Dial In User Service. Protocol used for authentication authorization and configuration information between a NAS and a shared authentication server.

**real server**—The specification of a physical server associated with a virtual server. The specification includes the real server’s IP address and an optional weight to be used by the virtual server predictor.

**redundancy**—The duplication of devices, services, or connections so that, in the event of a failure, the redundant devices, services, or connections can perform the work of those that failed. See also *stateful backup*, and *stateless backup*.

**Remote Authentication Dial In Use Service**—See *RADIUS*.

**round robin**—See *weighted round robin*.

**Secure Socket Layer**—See *SSL*.

**server cluster**—See *server farm*.

**server farm**—Also called a server cluster. Group of real servers that provide various applications and services.
Server Load Balancing—See IOS SLB.

server NAT—Translation scheme in which the virtual server IP address is replaced with the real server IP address (and vice versa), allowing servers to be many hops away from the load-balancing device, and enabling intervening routers to route to them without requiring tunnelling. See also client NAT, directed mode, and server port translation.

server port translation—Translation scheme in which the virtual server port number is replaced with the real server port number (and vice versa), allowing servers to be many hops away from the load-balancing device, and enabling intervening routers to route to them without requiring tunnelling. See also client NAT, directed mode, and server port translation.

generated GPRS support node—See SGSN.

SGSN—serving GPRS support node. A GPRS network entity that sends data to and receives data from mobile stations, and maintains information about the location of a mobile station. The SGSN communicates between the mobile station and the GGSN; the GGSN provides access to the data network. See also GGSN, GPRS, GSM, and GTP.

SLB—See IOS SLB.

SSL—Secure Socket Layer. Encryption technology for the web used to provide secure transactions such as the transmission of credit card numbers for e-commerce.

stateful backup—Redundancy scheme that enables IOS SLB to incrementally backup its load-balancing decisions, or “keep state,” between primary and backup switches. See also active standby and stateless backup.

stateless backup—Redundancy scheme that provides high network availability by routing IP flows from hosts on Ethernet networks without relying on the availability of a single Layer 3 switch. See also active standby and stateful backup.

sticky connections—Load-balancing scheme in which new connections from a client IP address or subnet are assigned to the same real server (for server load balancing) or firewall (for firewall load balancing) as were previous connections from that address or subnet.

Virtual Private Network—See VPN.

virtual server—Presents a single address that represents an application server farm to clients.

VPN—Virtual Private Network. Private network that uses the Internet to connect some nodes.

WAP—Wireless Application Protocol. Suite of protocols used to deliver services to wireless devices.

weighted least connection—Load-balancing algorithm in which the next real server chosen for a new connection to the virtual server is the server with the fewest active connections. 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. 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 \( n_i/(n_1+n_2+n_3,...) \).

weighted round robin—Load-balancing algorithm in which the real server used for a new connection to the virtual server is chosen 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. New connections are assigned to a given real server \( n \) times before the next real server in the list is chosen.
Wireless Application Protocol—See WAP.

Wireless Session Protocol—See WSP.

Wireless Transaction Protocol—See WTP.

Wireless Transport Security Layer—See WTLS.

Workload agents—Value-added software components developed for specific platforms by third-party developers. Workload agents run on server platforms or on platforms that manage server farms. Workload agents deliver server and application information to the services manager. This information enables the services manager to make optimum server selection.


WTLS—Wireless Transport Security Layer. Layer that provides security between WAP clients and WAP gateways.