



# IOS Server Load Balancing

---

This feature module describes the Cisco IOS Server Load Balancing (SLB) feature for Cisco IOS Release 12.1(1)E. It includes the following sections:

- Feature Overview, page 1
- Supported Platforms, page 9
- Supported Standards, MIBs, and RFCs, page 9
- Configuration Tasks, page 9
- Monitoring and Maintaining IOS SLB, page 22
- Configuration Examples, page 23
- Command Reference, page 30
- Debug Commands, page 71
- Glossary, page 75

## Feature Overview

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



**Note**

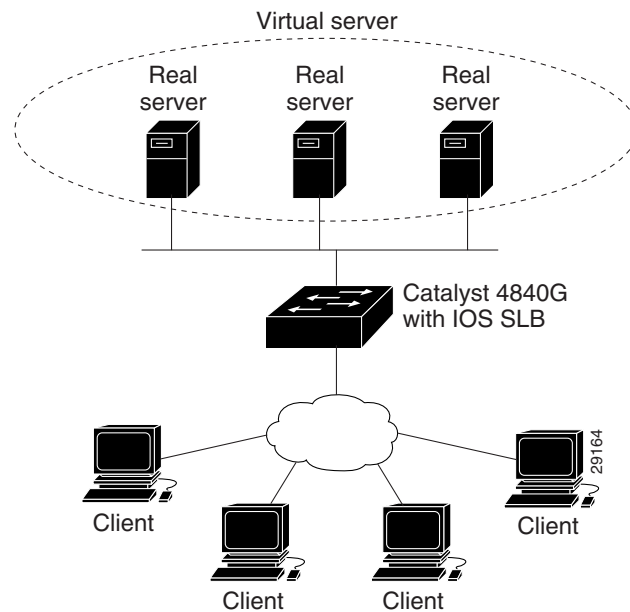
---

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.

---

Figure 1 illustrates IOS SLB.

**Figure 1** Logical View of IOS SLB



## IOS SLB Functions and Capabilities

This section describes the following functions and capabilities supported in IOS SLB:

- Algorithms for Server Load Balancing, page 3
- Port-Bound Servers, page 3
- Client-Assigned Load Balancing, page 4
- Content Flow Monitor Support, page 4
- Sticky Connections, page 4
- Maximum Connections, page 4
- Delayed Removal of TCP Connection Context, page 4
- TCP Session Reassignment, page 5
- Automatic Server Failure Detection, page 5
- Automatic Unfail, page 5
- Slow Start, page 5
- SynGuard, page 5
- Dynamic Feedback Protocol for IOS SLB, page 5
- Alternate IP Addresses, page 6
- Transparent Web Cache Balancing, page 6
- NAT, page 6
- Redundancy Enhancement—Stateless Backup, page 7

## Algorithms for Server Load Balancing

IOS SLB provides two load-balancing algorithms: weighted round robin and weighted least connections. You may specify either algorithm 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.

**Note**

---

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

---

### Weighted Least Connections

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

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

**Note**

---

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

---

### 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.

## 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 serverfarm, and assign another set of clients (such as external clients) to a different serverfarm.

## 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.

## Sticky Connections

When you use sticky connections, new connections from a client IP address or subnet are assigned to the same real server as were previous connections from that address or subnet.

IOS SLB creates sticky objects to track client assignments. The sticky objects remain in the IOS SLB database after the last sticky connection is deleted, for a period defined by a configurable *sticky timer*. If the timer is configured on a virtual server, new connections from a client are sent to the same real server that handled the previous client connection, provided one of the following is true:

- A connection for the same client already exists.
- The amount of time between the end of a previous connection from the client and the start of the new connection is within the timer duration.

Sticky connections also permit the coupling of services that are handled by more than one virtual server. This allows connection requests for related services to use the same real server. For example, Web server (HTTP) typically uses TCP port 80, and HTTP over Secure Socket Layer (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.

## Maximum Connections

The maximum connections feature allows you to configure a limit on the number of active connections that a real server can handle.

## 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*.

## 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*.

## 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.

## 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 connection for which it qualifies. 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.

## 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.

## 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 to a virtual server over a specific time interval and does not allow the number to exceed a configured *SYN threshold*. If the threshold is reached, any new SYNs are dropped.

## Dynamic Feedback Protocol for IOS SLB

The IOS SLB Dynamic Feedback Protocol (DFP) is a mechanism that allows host agents in load-balanced environments to dynamically report the change 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.

## 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.

## Transparent Web Cache Balancing

You can balance transparent Web caches if you know in advance the IP addresses they are serving. In IOS SLB, configure the IP addresses, or some common subset of them, as virtual servers.



### Note

---

A Web cache can start its own connections to real sites if pages are not available in its cache. Those connections cannot be load-balanced back to the same set of caches. IOS SLB addresses this by allowing you to configure “client exclude” statements so that IOS SLB does not load-balance connections initiated by the Web caches.

---

## NAT

Cisco IOS Network Address Translation (NAT), RFC 1631, allows unregistered “private” IP addresses to connect to the Internet by translating them into globally registered IP addresses. Cisco IOS NAT also increases network privacy by hiding internal IP addresses from external networks.

IOS SLB can operate in one of two redirection modes:

- 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 real server, translating the virtual server IP address to a real server address via NAT.
- 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.

The main advantage of dispatched mode is performance. In dispatched mode, the Layer 3 and Layer 4 addresses are not modified so IP header checksum adjustment is fast, and checksum adjustment or recalculation for TCP or UDP is not required. Dispatched mode is also simpler because packets for applications with IP addresses in the packet do not have to be examined and modified.

The main disadvantage of dispatched mode is that the virtual server IP address is not modified, so the real servers must be Layer 2 adjacent with the load balancer or intervening routers may not be able to route to the chosen real server.

NAT (directed mode) is used to solve these dispatched mode problems.

IOS SLB currently supports only server NAT. By replacing the virtual server IP address with the real server IP address (and vice versa), servers can be many hops away from the load balancer and intervening routers can route to them without requiring tunnelling. Additionally, loopback and secondary interfaces need no longer be on the real server.

**Note**

---

On the Catalyst 6000 Family switches and Cisco 7200 Series routers, if an IP address is configured as a real IP address for a NAT virtual server, you cannot balance connection requests from that address to a different virtual server (whether NAT or dispatch) on the same load balancer.

---

The network designer must ensure that outbound packets travel through IOS SLB, using one of the following methods:

- Direct wiring (all packets flow through a branch office IOS SLB device)
- Default gateways or policy-based routing
- IOS SLB NAT of client addresses, enabled as an outbound feature on server-side interfaces

A less common form of server NAT is server port translation, which involves replacement of a virtual server port. Server port translation does not require server IP address translation, but the two translations can be used together.

## Redundancy Enhancement—Stateless Backup

An IOS SLB could represent a point of failure and the servers could lose their connections to the backbone if power fails, or if a link from a switch to the distribution-layer switch is disconnected. IOS SLB supports a stateless backup option you can use to reduce that risk. Stateless backup, based on Hot Standby Router Protocol (HSRP), provides high network availability by routing IP flows from hosts on Ethernet networks without relying on the availability of a single Layer 3 switch.

HSRP is configured on Layer 3 switches that run IP over Ethernet. If a Layer 3 switch fails, HSRP automatically allows another Layer 3 switch to assume the function of the failing switch. HSRP is therefore particularly useful when you require continuous access to resources in the network.

HSRP is compatible with Novell's Internetwork Packet Exchange (IPX) and with AppleTalk.

**Note**

---

To avoid any single point of failure in an IOS SLB network, use multiple Layer 2 switches to provide connectivity between the IOS SLB devices and the servers.

---

## 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 the Cisco Catalyst 6500 switch, IOS SLB takes advantage of hardware acceleration to forward data 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-scale as well as small- and medium- sized 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.

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

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

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

## 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.**
- Operates in a standalone mode and currently does not operate as a MultiNode Load Balancing (MNLB) Services Manager. The presence of IOS SLB does not preclude the use of the existing MNLB Forwarding Agent with an external Services Manager in an MNLB environment.
- Does not support coordinating server load-balancing statistics among different IOS SLB instances for backup capability.
- Supports FTP only in dispatched mode.
- Does not support IOS SLB and Cisco Applications and Services Architecture (CASA) configured with the same virtual IP address, even if they are for different services.
- Supports Cisco IOS NAT in directed mode with no hardware data packet acceleration. (Hardware data packet acceleration is performed by the Policy Feature Card (PFC), and in directed mode the data packets are handled by the Multilayer Switched Feature Card (MSFC), not the PFC.)
- For the Catalyst 6000 Family Switches:
  - Requires the MSFC and the PFC.
  - 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 “Configuring IP Multilayer Switching” section in the *Catalyst 6000 Family MSFC (12.0) & PFC Configuration Guide, Release 5.4*.
  - When operating in dispatched mode, real servers must be Layer 2-adjacent to the IOS SLB switch (that is, not beyond an additional router), with hardware data packet acceleration performed by the PFC. All real servers that can be reached by a single IOS SLB device must be on the same VLAN. The loopback address must be configured in the real servers.
  - When operating in directed mode with server NAT, real servers need not be Layer 2-adjacent to the IOS SLB switch. This allows for more flexible network design, since servers can be placed several Layer 3 hops away from the IOS SLB switch.

- Requires that all real servers that can be reached by a single IOS SLB device must be on the same VLAN. The loopback address must be configured in the real servers.
- Supports NativeIOS only and C6sup-is-mz images.
- For the Cisco 7200 Series:
  - In dispatched mode, the servers must be Layer 2-adjacent or tag-switched. In directed mode, the servers can be one or more IP hops away.
  - Supports Cisco IOS NAT in directed mode with no hardware data packet acceleration. Provides no hardware acceleration for the IOS SLB function for either dispatched mode or directed mode.
  - Supports C7200-is-mz images.

## Supported Platforms

- Catalyst 6000 Family Switches
- Cisco 7200 Series Routers

## Supported Standards, MIBs, and RFCs

### Standards

- No new or modified standards

### MIBs

- No new or modified MIBs

### RFCs

- 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. See the following sections for configuration tasks for the IOS SLB feature. Perform these tasks in the order given. Some of the tasks are required; others are optional.

- Specifying a Server Farm (Required), page 10
- Specifying a Load-Balancing Algorithm (Optional), page 10
- Specifying a Bind ID (Optional), page 10
- Specifying a Real Server (Required), page 12
- Configuring Real Server Attributes (Optional), page 12
- Enabling the Real Server for Service (Required), page 12
- Specifying a Virtual Server (Required), page 13
- Associating a Virtual Server with a Server Farm (Required), page 13

- Configuring Virtual Server Attributes (Required), page 13
- Adjusting Virtual Server Values (Optional), page 13
- Preventing Advertisement of Virtual Server Address (Optional), page 14
- Enabling the Virtual Server for Service (Required), page 14
- Configuring IOS SLB Dynamic Feedback Protocol (Optional), page 14
- Configuring NAT (Optional), page 15
- Implementing IOS SLB Stateless Backup (Optional), page 15
- Verifying IOS SLB, page 18
- Troubleshooting IOS SLB, page 21

## Specifying a Server Farm (Required)

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.

To configure a server farm, use the following command in global configuration mode:

Command	Purpose
Router(config)# <b>ip slb serverfarm</b> <i>serverfarm-name</i>	Adds a server farm definition to the IOS SLB configuration and initiates server farm configuration mode. See the <b>ip slb serverfarm</b> command for more details.

## Specifying a Load-Balancing Algorithm (Optional)

To determine which real server to use for each new connection request, the IOS SLB feature uses one of two load-balancing algorithms: weighted round robin (the default) or weighted least connections. (See the “Weighted Round Robin” section on page 3 or the “Weighted Least Connections” section on page 3 for detailed descriptions of these algorithms.) To specify the load-balancing algorithm, use the following command in server farm configuration mode:

Command	Purpose
Router(config-slb-sfarm)# <b>predictor</b> [ <b>roundrobin</b>   <b>leastconns</b> ]	Specifies whether the weighted round robin algorithm or the weighted least connections algorithm is to be used to determine how a real server is selected. See the <b>predictor</b> command for more details.

## Specifying a Bind ID (Optional)

To configure a bind ID on the server farm for use by DFP, use the following command in server farm configuration mode:

Command	Purpose
Router(config-slb-sfarm)# <b>bindid</b> [ <i>bind_id</i> ]	Specifies a bind ID on the server farm for use by DFP. See the <b>bindid</b> command for more details.

## Specifying a Real Server (Required)

A server farm comprises a number of real servers. The real servers are the physical devices that provide the load-balanced services.

To identify a real server in your network, use the following command in server farm configuration mode:

Command	Purpose
Router(config-slb-sfarm)# <b>real</b> <i>ip-address</i>	Identifies a real server to the IOS SLB function and initiates real server configuration mode. See the <b>real</b> command for more details.

## Configuring Real Server Attributes (Optional)

You can configure any of the following real server attributes, by using the following commands in real server configuration mode:

Command	Purpose
Router(config-slb-real)# <b>faildetect</b> <b>numconns</b> <i>number-conns</i> [ <b>numclients</b> <i>number-clients</i> ]	Specifies the number of consecutive connection failures and, optionally, the number of unique client connection failures, that constitute failure of the real server. See the <b>faildetect</b> command for more details.
Router(config-slb-real)# <b>maxconns</b> <i>number-conns</i>	Specifies the maximum number of active connections allowed on the real server at one time. See the <b>maxconns</b> command for more details.
Router(config-slb-real)# <b>reassign</b> <i>threshold</i>	Specifies the number of consecutive unanswered SYNs that initiates assignment of the connection to a different real server. See the <b>reassign</b> command for more details.
Router(config-slb-real)# <b>retry</b> <i>retry-value</i>	Specifies the interval, in seconds, to wait between the detection of a server failure and the next attempt to connect to the failed server. See the <b>retry</b> command for more details.
Router(config-slb-real)# <b>weight</b> <i>weighting-value</i>	Specifies the real server's workload capacity relative to other servers in the server farm. See the <b>weight</b> command for more details.

## Enabling the Real Server for Service (Required)

To place the real server into service, use the following command in real server configuration mode:

Command	Purpose
Router(config-slb-real)# <b>inservice</b>	Enables the real server for use by IOS SLB. See the <b>inservice (real server)</b> command for more details.

## Specifying a Virtual Server (Required)

To specify a virtual server, use the following command in global configuration mode:

Command	Purpose
Router(config)# <b>ip slb vserver</b> <i>virtserver-name</i>	Identifies a virtual server and initiates virtual server configuration mode. See the <b>ip slb vserver</b> command for more details.

## Associating a Virtual Server with a Server Farm (Required)

To associate the virtual server with a server farm, use the following command in virtual server configuration mode:

Command	Purpose
Router(config-slb-vserver)# <b>serverfarm</b> <i>serverfarm-name</i>	Associates a real server farm with a virtual server. See the <b>serverfarm</b> command for more details.

## Configuring Virtual Server Attributes (Required)

To configure virtual server attributes, use the following command in virtual server configuration mode:

Command	Purpose
Router(config-slb-vserver)# <b>virtual</b> <i>ip-address</i> { <b>tcp</b>   <b>udp</b> } <i>port-number</i> [ <b>service</b> <i>service-name</i> ]	Specifies the virtual server IP address, type of connection, port number, and optional service coupling. See the <b>virtual</b> command for more details.

## Adjusting Virtual Server Values (Optional)

To change the default settings of the virtual server values, use the related command in virtual server configuration mode:

Command	Purpose
Router(config-slb-vserver)# <b>client</b> <i>ip-address network-mask</i>	Specifies which clients are allowed to use the virtual server. See the <b>client</b> command for more details.
Router(config-slb-vserver)# <b>delay</b> <i>duration</i>	Specifies the amount of time IOS SLB maintains TCP connection context after a connection has terminated. The default value is 10 seconds. See the <b>delay</b> command for more details.
Router(config-slb-vserver)# <b>idle</b> <i>duration</i>	Specifies the minimum amount of time IOS SLB maintains connection context in the absence of packet activity. The default value is 3600 seconds (60 hours). See the <b>idle</b> command for more details.

Command	Purpose
Router(config-slb-vserver)# <b>sticky</b> <i>duration</i> [ <b>group</b> <i>group-id</i> ]	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. See the <b>sticky</b> command for more details.
Router(config-slb-vserver)# <b>synguard</b> <i>syn-count interval</i>	Specifies the rate of TCP SYNs handled by a virtual server in order to prevent a SYN flood denial of service attack. See the <b>synguard</b> command for more details.

## Preventing Advertisement of Virtual Server Address (Optional)

By default, virtual server addresses are *advertised*. That is, static routes to the Null0 interface are installed for the virtual server addresses. To advertise these static routes using the routing protocol, you must configure redistribution of static routes for the routing protocol. To prevent the installation of a static route, use the **no** form of the **advertise** command, in virtual server configuration mode:

Command	Purpose
Router(config-slb-vserver)# <b>no advertise</b>	Omits the virtual server IP address from the routing protocol updates. See the <b>advertise</b> command for more details.

## Enabling the Virtual Server for Service (Required)

To place the virtual server into service, use the following command in virtual server configuration mode:

Command	Purpose
Router(config-slb-vserver)# <b>inservice</b>	Enables the virtual server for use by IOS SLB. See the <b>inservice (virtual server)</b> command for more details.

## Configuring IOS SLB Dynamic Feedback Protocol (Optional)

To configure IOS SLB DFP, enter the following commands in order, beginning in global configuration mode:

	Command	Purpose
<b>Step 1</b>	Router(config)# <b>ip slb dfp</b> [ <b>password</b> <i>password</i> [ <i>timeout</i> ]]	Configures DFP and, optionally, sets a password and initiates DFP configuration mode. See the <b>ip slb dfp</b> command for more details.
<b>Step 2</b>	Router(config-slb-dfp)# <b>agent</b> <i>ip_address port</i> [ <i>timeout</i> [ <i>retry_count</i> [ <i>retry_interval</i> ]]]	Configures a DFP agent. See the <b>agent</b> command for more details.

## Configuring NAT (Optional)

To configure IOS SLB NAT mode for a specific server farm, enter the following commands in order, beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# <b>ip slb serverfarm</b> <i>serverfarm-name</i>	Adds a server farm definition to the IOS SLB configuration and initiates server farm configuration mode. See the <b>ip slb serverfarm</b> command for more details.
Step 2	Router(config-slb-sfarm)# <b>nat server</b>	Configures server NAT. See the <b>nat</b> command for more details.
Step 3	Router(config-slb-sfarm)# <b>real</b> <i>ip-address</i> [ <i>port_number</i> ]	Identifies a real server to the IOS SLB function and initiates real server configuration mode. See the <b>real</b> command for more details.

## Implementing IOS SLB Stateless Backup (Optional)

Stateless backup, based on the Hot Standby Router Protocol (HSRP), provides high network availability by routing IP flows from hosts on Ethernet networks without relying on the availability of any 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.

### How IOS SLB Stateless Backup Works

A Layer 3 switch running the HSRP detects a failure by sending and receiving multicast User Datagram Protocol (UDP) “hello” packets. When the IOS SLB switch running HSRP detects that the designated active Layer 3 switch has failed, the selected backup Layer 3 switch assumes control of the HSRP group MAC and IP addresses. (You can also select a new standby Layer 3 switch at that time.) Both the primary and the backup Layer 3 switch must be on the same subnetwork.

The chosen MAC and IP addresses must be unique and must not conflict with any others on the same network segment. The MAC address is selected from a pool of Cisco MAC addresses. Configure the last byte of the MAC address by using the HSRP group number. When the HSRP is running, it selects an active Layer 3 switch and instructs its device layer to listen on an additional (dummy) MAC address.

IOS SLB switching software supports HSRP over 10/100 Ethernet, Gigabit Ethernet, FEC, GEC, and BVI (Bridge-Group Virtual Interface) connections.

HSRP uses a priority scheme to determine which HSRP-configured Layer 3 switch is to be the default active Layer 3 switch. To configure a Layer 3 switch as active, you assign it a priority higher than that of all other HSRP-configured Layer 3 switches. The default priority is 100, so if you configure just one Layer 3 switch to have a higher priority, that switch becomes the default active switch.

HSRP works by the exchange of multicast messages that advertise priority among HSRP-configured Layer 3 switches. When the active switch fails to send a hello message within a configurable period, the standby switch with the highest priority becomes the active switch. The transition of packet-forwarding functions between Layer 3 switches is completely transparent to all hosts accessing the network.

HSRP-configured Layer 3 switches exchange the following types of multicast messages:

- *Hello*—The hello message conveys the switch’s HSRP priority and state information. By default, an HSRP switch sends hello messages every three seconds.
- *Coup*—When a standby Layer 3 switch assumes the function of the active switch, it sends a coup message.
- *Resign*—The active Layer 3 switch, sends this message when it is about to shut down or when a switch that has a higher priority sends a hello message.

At any time, HSRP-configured Layer 3 switches are in one of the following states:


- *Active*—The switch is performing packet-transfer functions.
- *Standby*—The switch is prepared to assume packet-transfer functions if the active router fails.
- *Speaking and listening*—The switch is sending and receiving hello messages.
- *Listening*—The switch is receiving hello messages.

## Configuring IOS SLB Stateless Backup

Configuration of stateless backup requires the following:

- You must configure IOS SLB switches to run HSRP between interfaces on the server side.
- You can configure multiple IOS SLB switches that share a virtual IP address as long as the client ranges are exclusive and you use policy routing to forward the flows to the correct IOS SLB switch.

To configure stateless backup over VLANs between IOS SLB switches, perform the following tasks in order:

- 
- Step 1** Configure the server farms—See the “Specifying a Server Farm (Required)” section on page 10.
  - Step 2** Configure the real servers—See the “Specifying a Real Server (Required)” section on page 12.
  - Step 3** Configure the virtual servers—See the “Specifying a Virtual Server (Required)” section on page 13.
-  **Note** When you use the **inservice (virtual server)** command to configure the virtual server as “in-service” you must use the optional **standby** command and configure an HSRP group name.
- 
- Step 4** Configure the IP routing protocol—See the “IP Routing Protocols” chapter of the *Cisco IOS IP and IP Routing Configuration Guide*.
  - Step 5** Configure the VLAN between the switches—See the “Virtual LANs” chapter of the *Cisco IOS Switching Services Configuration Guide*.
  - Step 6** Enable HSRP—See the “Enabling HSRP” section on page 17.
  - Step 7** Customize group attributes—See the “Customizing Group Attributes” section on page 17.
  - Step 8** Verify the IOS SLB HSRP configuration—See the “Verifying the IOS SLB Stateless Backup Configuration” section on page 18.
- 

A sample stateless backup configuration is shown in the “Sample IOS SLB Stateless Backup Configuration” section on page 18.

## Enabling HSRP

To enable HSRP on an IOS SLB interface, enable the protocol, then customize it for the interface. Enter the following command in interface configuration mode:

Command	Purpose
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>ip</b> [ <i>ip-address</i> ] [ <b>secondary</b> ]	Enables HSRP.

## Customizing Group Attributes

To customize “hot standby” group attributes, use one or more of the following commands in interface configuration mode:

Command	Purpose
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>authentication</b> <i>string</i>	Selects an authentication string to be carried in all HSRP messages.
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>name</b> <i>group-name</i>	Specifies an HSRP group name with which to associate an IOS SLB interface.
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>preempt</b>	Specifies that if the local router has priority over the current active router, the local router should attempt to take its place as the active router.
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>priority</b> <i>priority</i>	Sets the hot standby priority used to choose the active router.
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>timers</b> <i>hellotime holdtime</i>	Configures the time between hello packets and the hold time before other routers declare the active router to be down.
Router(config-if)# <b>standby</b> [ <i>group-number</i> ] <b>track</b> <i>type-number</i> [ <i>interface-priority</i> ]	Configures the interface to track other interfaces, so that if one of the other interfaces goes down, the hot standby priority for the device is lowered.

## Verifying the IOS SLB Stateless Backup Configuration

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

```
Router# show ip slb vservers
```

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

```
Router# show ip slb vservers detail
```

```
VS1, state = INSERVICE, 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 = INOFSERVICE, 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
```

## Sample IOS SLB Stateless Backup Configuration

The following commands enable the HSRP standby group 100 IP address, priority, preempt, timers, configure a name and authentication for Device A in Figure 4:

```
Router(config-if)# standby 100 ip 172.20.100.10
Router(config-if)# standby 100 priority 110
Router(config-if)# standby 100 preempt
Router(config-if)# standby 100 timers 5 15
Router(config-if)# standby 100 name Web_group1
Router(config-if)# standby 100 authentication Secret
Router(config-if)# exit
Router#
```

## Verifying IOS SLB

This section describes how to verify the following different aspects of the IOS SLB feature:

- Verifying IOS SLB Installation, page 19
- Verifying Server Failure Detection, page 19

## Verifying IOS SLB Installation

To verify that the IOS SLB is installed and working properly, follow these steps:

- 
- Step 1** Telnet to the IOS SLB device.
  - Step 2** Ping from that device to each of the clients and real servers. If it is not precluded by firewalls or network configuration, ping from the client side to each of the real servers.
  - Step 3** From the client side, ping the virtual server. Pings are answered by IOS SLB even if no real servers are in service, so this ensures that the IOS SLB device is reachable.
  - Step 4** For the selected protocol, start a client connection to the virtual server.
  - Step 5** If you want sticky connections:
    - a. Configure the sticky connections.
    - b. Start a client connection.
    - c. Enter the **show ip slb reals detail** and **show ip slb conns** commands.
    - d. Examine the real server connection counts. The real server whose count increased is the one to which the client connection is assigned.
    - e. Enter the **show ip slb sticky** command to display the sticky relationships that IOS SLB stored.
    - f. End the connection.
    - g. Ensure that the real server's connection count decreased.
    - h. Restart the connection, after waiting no longer than the sticky timeout value.
    - i. Enter the **show ip slb conns** command again.
    - j. Examine the real server connection counts again, and verify that the sticky connection is assigned to the same real server as before.
  - Step 6** Start additional client connections.
  - Step 7** Enter the **show ip slb reals detail** command.
  - Step 8** Verify that the the connection counts are increasing.
- 

## Verifying Server Failure Detection

To verify that server failures are detected correctly, follow these steps:

- 
- Step 1** Use a large client population. If the number of clients is very small, tune the **numclients** keyword on the **faildetect** command so that the servers are not displayed as **failed**.
  - Step 2** Enter the **show ip slb reals detail** command to show the status of the real servers.

**Step 3** Examine the status and connection counts of the real servers.

- Servers that failed show a status of **failed**, **testing**, or **ready\_to\_test**, based on whether IOS SLB is checking that the server came back up when the command was sent.
  - When a real server fails, connections that are assigned but not established (no SYN or ACK is received) are reassigned to another real server on the first inbound SYN after the **reassign** threshold is met. However, any connections that were already established are forwarded to the same real server because, while it may not be accepting new connections, it may be servicing existing ones.
  - For weighted least connections, a real server that has just been placed in service starts slowly so that it is not overloaded with new connections. (See the “Slow Start” section on page 5 for more information on this feature.) 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.
-

## Troubleshooting IOS SLB

The following questions and answers can help you troubleshoot IOS SLB, if you have problems.

Question	Answer
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 <b>numclients</b> , <b>numconns</b> , and <b>delay</b> keywords. If you have a very small client population (for example, in a test environment), the <b>numclients</b> 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 is IOS SLB not marking my connections as <b>established</b> 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.
Why does IOS SLB show my real server as <b>inservice</b> even though I have taken it down or physically disconnected it?	The <b>inservice</b> and <b>outofservice</b> states indicate whether the network administrator <i>intends</i> for that real server to be used when it is operational. A real server that was <b>inservice</b> but was removed from the selection list dynamically by IOS SLB as a result of automatic failure detection, is marked as <b>failed</b> . Use the <b>show ip slb reals detail</b> command to display these real server states.  Beginning with release 12.1(1)E, <b>inservice</b> is changed to <b>operational</b> , 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 <b>show mls flow</b> command:  Router# <b>sh mls flow</b> current ip flowmask for unicast: full flow current ipx flowmask for unicast: destination only  The current IP flowmask must be <b>full flow</b> . If it is not, correct the problem using the <b>mls flow ip full</b> command:  Router# <b>config t</b> Enter configuration commands, one per line. End with CNTL/Z. Router(config)# <b>mls flow ip full</b> Router(config)#

# Monitoring and Maintaining IOS SLB

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

Command	Purpose
Router# <b>show ip slb conns</b> [ <b>vserver</b> <i>virtserver-name</i> ] [ <b>client</b> <i>ip-address</i> ] [ <b>detail</b> ]	Displays all connections handled by IOS SLB, or, optionally, only those connections associated with a particular virtual server or client. See the <b>show ip slb conns</b> command for more details.
Router# <b>show ip slb dfp</b> [ <b>agent</b> <i>ip_addr port</i> ] [ <b>detail</b> ] [ <b>weights</b> ]	Displays information about DFP and DFP agents, and about the weights assigned to real servers. See the <b>show ip slb dfp</b> command for more details.
Router# <b>show ip slb reals</b> [ <b>vserver</b> <i>virtserver-name</i> ] [ <b>detail</b> ]	Displays information about the real servers defined to IOS SLB. See the <b>show ip slb reals</b> command for more details.
Router# <b>show ip slb serverfarms</b> [ <b>name</b> <i>serverfarm-name</i> ] [ <b>detail</b> ]	Displays information about the server farms defined to IOS SLB. See the <b>show ip slb serverfarms</b> command for more details.
Router# <b>show ip slb stats</b>	Displays IOS SLB statistics. See the <b>show ip slb stats</b> command for more details.
Router# <b>show ip slb sticky</b> [ <b>client</b> <i>ip-address</i> ]	Displays information about the sticky connections defined to IOS SLB. See the <b>show ip slb sticky</b> command for more details.
Router# <b>show ip slb vservers</b> [ <b>name</b> <i>virtserver-name</i> ] [ <b>detail</b> ]	Displays information about the virtual servers defined to IOS SLB. See the <b>show ip slb vservers</b> command for more details.

# Configuration Examples

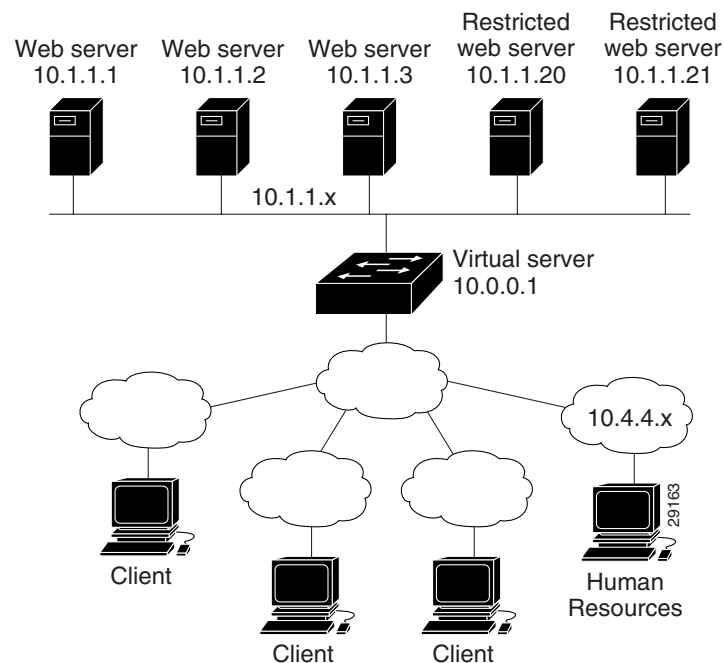
This section provides the following IOS SLB configuration examples:

- Sample IOS SLB Network Configuration, page 23
- Sample NAT Configuration, page 25
- Sample HSRP Configuration, page 27

## Sample IOS SLB Network Configuration

This section provides a configuration example based on the network layout shown in Figure 2.

**Figure 2** IOS SLB Network Configuration



As shown in the following sample code, the example topology has three public Web servers and two restricted Web servers for privileged clients in subnet 10.4.4.x. 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.

This configuration is coded as follows:

```

ip slb serverfarm PUBLIC
  predictor leastconns
  real 10.1.1.1
    weight 16
    inservice
  real 10.1.1.2
    weight 4
    maxconns 1000
    inservice
  real 10.1.1.3
    weight 24
    inservice

ip slb serverfarm RESTRICTED
  predictor leastconns
  real 10.1.1.20
    in-service
  real 10.1.1.21
    in-service

ip slb vserver PUBLIC_HTTP
  virtual 10.0.0.1 tcp www
  serverfarm PUBLIC
  inservice

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 idle 120 group 1
  inservice

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 idle 120 group 1
  inservice

```

*Unrestricted Web server farm*  
*Use weighted least connections algorithm*  
*First real server*

*Second real server*

*Restrict maximum number of connections*

*Third real server*

*Restricted Web server farm*  
*Use weighted least connections algorithm*  
*First real server*

*Second real server*

*Unrestricted Web virtual server*  
*Handle HTTP requests*  
*Use public Web server farm*

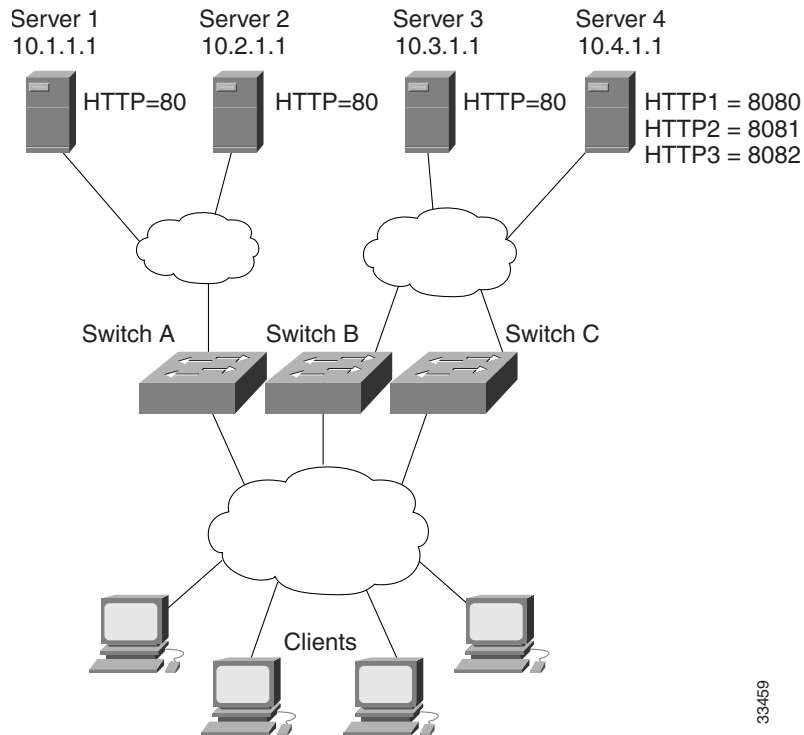
*Restricted HTTP virtual server*  
*Handle HTTP requests*  
*Use restricted Web server farm*  
*Only allow clients from 10.4.4.x*  
*Couple connections with RESTRICTED\_SSL*

*Restricted SSL virtual server*  
*Handle SSL requests*  
*Use restricted Web server farm*  
*Only allow clients from 10.4.4.x*  
*Couple connections with RESTRICTED\_HTTP*

## Sample NAT Configuration

This section provides a configuration example based on the network layout shown in Figure 3.

**Figure 3 IOS SLB NAT Topology**



The topology in Figure 3 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.

Servers 1 and 2 are load balanced using Switch A, which is performing server address translation.

Servers 3 and 4 are load balanced using Switches B and C. These two switches are performing server address translation. These switches also perform server port translation for HTTP packets to and from Server 4.

The configuration statements for Switch A are:

```
ip slb serverfarm FARM1
! Translate server addresses
  nat server
! Server 1 port 80
  real 10.1.1.1
  inservice
! Server 2 port 80
  real 10.2.1.1
  inservice
!
ip slb vserver HTTP1
! Handle HTTP (port 80) requests
  virtual 128.1.0.1 tcp www
  serverfarm FARM1
  inservice
```

The configuration statements for Switch B are:

```
ip slb serverfarm FARM2
! Translate server addresses
  nat server
! Server 3 port 80
  real 10.3.1.1
    inservice
! Server 4 port 8080
  real 10.4.1.1 port 8080
    inservice
! Server 4 port 8081
  real 10.4.1.1 port 8081
    inservice
! Server 4 port 8082
  real 10.4.1.1 port 8082
    inservice
!
ip slb vserver HTTP2
! Handle HTTP (port 80) requests
  virtual 128.2.0.1 tcp www
  serverfarm FARM2
  inservice
```

The configuration statements for Switch C are:

```
ip slb serverfarm FARM2
! Translate server addresses
  nat server
! Server 3 port 80
  real 10.3.1.1
    inservice
! Server 4 port 8080
  real 10.4.1.1 port 8080
    inservice
! Server 4 port 8081
  real 10.4.1.1 port 8081
    inservice
! Server 4 port 8082
  real 10.4.1.1 port 8082
    inservice
!
ip slb vserver HTTP2
! Handle HTTP (port 80) requests
  virtual 128.4.0.1 tcp www
  serverfarm FARM2
  inservice
```

## Sample HSRP Configuration

Figure 4 shows the topology of an IP network with two Layer 3 switches configured for HSRP. In this network:

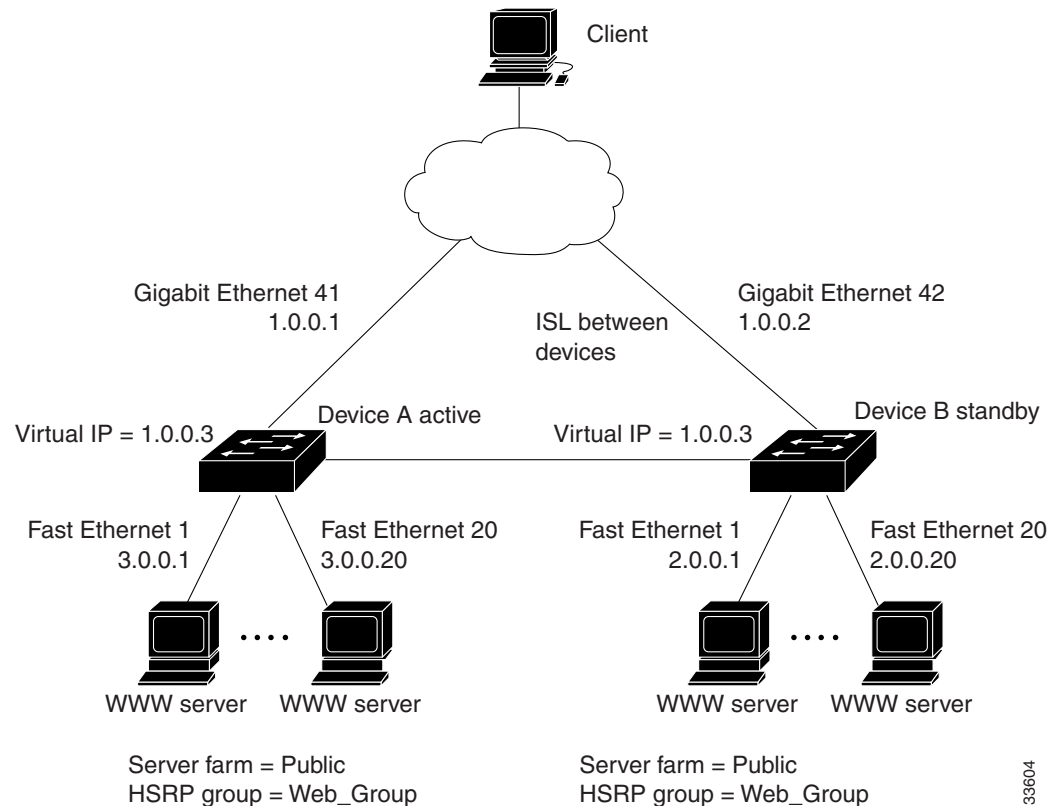
- Device A is the active HSRP Layer 3 switch and handles packets to the real servers with IP addresses 3.0.0.1 through 3.0.0.20.
- Device B handles packets to real servers with IP addresses 2.0.0.1 through 2.0.0.20.
- All hosts accessing the network use the IP address of the virtual router (in this case, 1.0.0.3).
- The configuration shown uses the Enhanced Interior Gateway Routing Protocol (Enhanced IGRP), but HSRP can be used with any other routing protocol supported by the Cisco IOS software, such as Open Shortest Path First (OSPF).



**Note** Some configurations that use HSRP still require a routing protocol for convergence when a topology change occurs. The standby Layer 3 switch becomes active, but connectivity does not occur until convergence occurs.

If the connection between Device A and the client accessing virtual IP 1.0.0.3 fails, fast-converging routing protocols (such as Enhanced IGRP and OSPF) can respond within seconds, ensuring that Device B is prepared to transfer packets that would have gone through Device A.

**Figure 4 HSRP Example Network Topology**



33604

The following is the configuration for Device A:

```
hostname Device A

interface GigabitEthernet 41
ip address 1.0.0.1 255.0.0.0
standby 1 ip 1.0.0.3
standby 1 preempt
standby 1 priority 110
standby 1 authentication denmark
standby 1 timers 5 15
standby 1 name Web-Group

interface FastEthernet 1
ip address 3.0.0.1 255.0.0.0

router eigrp 1
network 1.0.0.0
network 3.0.0.0
```

The following is the configuration for Device B:

```
hostname Device B

interface GigabitEthernet 41
ip address 1.0.0.2 255.0.0.0
standby 1 ip 1.0.0.3
standby 1 preempt
standby 1 authentication denmark
standby 1 timers 5 15
standby 1 name Web-Group

interface FastEthernet 41
ip address 2.0.0.1 255.0.0.0

router eigrp 1
network 1.0.0.0
network 2.0.0.0
```

The **standby ip** interface configuration command enables HSRP and establishes 1.0.0.3 as the IP address of the virtual router. The configurations of both Layer 3 switches include this command so that both switches share the same virtual IP address. The number 1 establishes Hot Standby group 1. (If you do not specify a group number, the default is group 0.) The configuration for at least one of the Layer 3 switches in the Hot Standby group must specify the IP address of the virtual router; specifying the IP address of the virtual router is optional for other routers in the same Hot Standby group.

The **standby preempt** interface configuration command allows the Layer 3 switch to become the active switch when its priority is higher than all other HSRP-configured switches in this Hot Standby group. The configurations of both switches include this command so that each can be the standby Layer 3 switch for the other switch. The number 1 indicates that this command applies to Hot Standby group 1. If you do not use the **standby preempt** command in the configuration for a Layer 3 switch, that switch cannot become the active Layer 3 switch.

The **standby priority** interface configuration command sets the Layer 3 switch's HSRP priority to 110, which is higher than the default priority of 100. Only the configuration of Device A includes this command, which makes Device A the default active Layer 3 switch. The number 1 indicates that this command applies to Hot Standby group 1.

The **standby authentication** interface configuration command establishes an authentication string whose value is an unencrypted eight-character string that is incorporated in each HSRP multicast message. This command is optional. If you choose to use it, each HSRP-configured Layer 3 switch in the group should use the same string so that each switch can authenticate the source of the HSRP messages that it receives. The number 1 indicates that this command applies to Hot Standby group 1.

The **standby timers** interface configuration command sets the interval (in seconds) between hello messages (called the *hello time*) to five seconds, and sets the interval (in seconds) that a Layer 3 switch waits before it declares the active Layer 3 switch to be down (called the *hold time*) to eight seconds. (The defaults are three and 10 seconds, respectively.) To modify the default values, you must configure each Layer 3 switch to use the same hello time and hold time. The number 1 indicates that this command applies to Hot Standby group 1.

The **standby name** interface configuration command associates the IOS SLB interface with an HSRP group name (in this case, Web-Group), previously specified on an **inservice (virtual server)** command. The number 1 indicates that this command applies to Hot Standby group 1.

# Command Reference

This section documents new or modified commands. All other commands used with this feature are documented in the Cisco IOS Release 12.0 and 12.0(7)T command reference publications.

- **advertise**, page 31
- **agent**, page 32
- **bindid**, page 34
- **clear ip slb**, page 35
- **client**, page 36
- **delay**, page 37
- **faildetect**, page 38
- **idle**, page 39
- **inservice (real server)**, page 40
- **inservice (virtual server)**, page 41
- **ip slb dfp**, page 42
- **ip slb serverfarm**, page 44
- **ip slb vserver**, page 45
- **maxconns**, page 46
- **nat**, page 47
- **predictor**, page 48
- **real**, page 49
- **reassign**, page 50
- **retry**, page 51
- **serverfarm**, page 52
- **show ip slb conns**, page 53
- **show ip slb dfp**, page 55
- **show ip slb reals**, page 57
- **show ip slb serverfarms**, page 59
- **show ip slb stats**, page 60
- **show ip slb sticky**, page 62
- **show ip slb vservers**, page 63
- **standby name**, page 64
- **sticky**, page 65
- **synguard**, page 67
- **virtual**, page 68
- **weight**, page 70

# advertise

By default, virtual server addresses are *advertised*. That is, static routes to the Null0 interface are installed for the virtual server addresses.

To control the installation of a static route to the Null0 interface for a virtual server address, use the **advertise** virtual server configuration command. Advertisement of this static route using the routing protocol requires that you configure redistribution of static routes for the routing protocol. To prevent the installation of a static route for the virtual server IP address, use the **no** form of this command.

**advertise**

**no advertise**

## Syntax Description

This command has no arguments or keywords.

## Defaults

The virtual server IP address is added to the routing table.

## Command Modes

Virtual server configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## 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

Command	Description
<b>show ip slb vservers</b>	Displays information about the virtual servers.

# agent

To configure a DFP agent, use the **agent** DFP configuration command. To remove an 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	
<i>ip-address</i>	Agent IP address
<i>port</i>	Agent port number
<i>timeout</i>	(Optional) Time period, in seconds, during which the DFP manager must receive an update from the DFP agent. The default is 0 seconds, which means there is no timeout.
<i>retry_count</i>	(Optional) Number of times DFP manager attempts to establish the TCP connection to the DFP agent. The default is 0 retries, which means there are infinite retries.
<i>retry_interval</i>	(Optional) Interval, in seconds, between retries. 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

Release	Modification
12.0(7)XE	This command was introduced.

## Usage Guidelines

You can configure up to 1024 agents.

A DFP agent collects status information about a server's load capability and reports that information to a load manager. The DFP agent may reside on the server, or it may be a separate device that collects and consolidates the information from several servers before reporting to the load manager.

## Examples

The following example configures a DFP agent on the DFP manager, sets the DFP password to *Cookies* and the timeout to *360* seconds, changes the CLI to DFP configuration mode, sets the IP address of the DFP agent to *10.1.1.1*, and sets the port number of the DFP agent to *2221* (FTP):

```
Router(config)# ip slb dfp password Cookies 360  
Router(config-slbf-dfp)# agent 10.1.1.1 2221
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>ip slb dfp</b>	Configures the IOS SLB DFP.

# bindid

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

**bindid** *[bind\_id]*

**no bindid** *[bind\_id]*

<b>Syntax Description</b>	<i>bind_id</i> (Optional) Bind ID number. The default bind ID is 0.
---------------------------	---

<b>Defaults</b>	The default bind ID is 0.
-----------------	---------------------------

<b>Command Modes</b>	Server farm configuration
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

<b>Usage Guidelines</b>	<p>You can configure one bind ID on each <b>bindid</b> command.</p> <p>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. DFP uses the bind ID to identify for which instance of the real server a given weight is specified.</p>
-------------------------	--

<b>Examples</b>	The following example configures bind ID 309:
-----------------	---

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

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>ip slb dfp</b>	Configures the IOS SLB DFP.

# clear ip slb

To clear IP IOS SLB connections or counters, use the **clear ip slb** command.

```
clear ip slb {connections [serverfarm farm_name | vserver server_name] | counters}
```

Syntax Description	connections	Clears the IP IOS SLB connection database.
	<b>serverfarm</b>	Clears the connection database for the serverfarm named.
	<i>farm_name</i>	Character string used to identify the serverfarm.
	<b>vserver</b>	Clears the connection database for the virtual server named.
	<i>server_name</i>	Character string used to identify the virtual server.
	<b>counters</b>	Clears the IP IOS SLB counters.

**Defaults** No default behavior or values.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.1(1)E	This command was introduced.

**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
```

The following example clears the IOS SLB counters:

```
Router# clear ip slb counters
```

Related Commands	Command	Description
	<b>show ip slb serverfarms</b>	Displays information about the IOS SLB server farms.
	<b>show ip slb vservers</b>	Displays information about the IOS SLB virtual servers.
	<b>show ip slb conns</b>	Displays information about the IOS SLB connections.

# client

To define which clients are allowed to use the virtual server, use the **client** virtual server configuration command. You can use more than one client command to define more than one client. To remove a client definition from the IOS SLB configuration, use the **no** form of this command.

**client** *ip-address network-mask*

**no client** *ip-address network-mask*

## Syntax Description

<i>ip-address</i>	Client IP address. The default is 0.0.0.0 (all clients).
<i>network-mask</i>	Client IP network mask. The default is 0.0.0.0 (all subnetworks).

## Defaults

The default IP address is 0.0.0.0 (all clients).

The default network mask is 0.0.0.0 (all subnetworks).

Taken together, the default is **client 0.0.0.0 0.0.0.0** (allows all clients on all subnetworks to use the virtual server).

## Command Modes

Virtual server configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Usage Guidelines

The *network-mask* 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.

## Examples

The following example allows only clients from 10.4.4.x access to the virtual server:

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

## Related Commands

Command	Description
<b>show ip slb vservers</b>	Displays information about the virtual servers.
<b>virtual</b>	Configures the virtual server attributes.

# delay

To change the amount of time IOS SLB maintains TCP connection context after a connection has terminated, use the **delay** virtual server configuration command. To restore the default delay timer, use the **no** form of this command.

**delay** *duration*

**no delay**

<b>Syntax Description</b>	<i>duration</i>	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

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

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

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>show ip slb vservers</b>	Displays information about the virtual servers.
	<b>virtual</b>	Configures the virtual server attributes.

# faildetect

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

**faildetect numconns** *number-conns* [**numclients** *number-clients*]

**no faildetect**

## Syntax Description

<b>numconns</b>	Number of consecutive TCP connection reassignments allowed before a real server is considered to have failed.
<i>number-conns</i>	Connection reassignment threshold value in the range from 1 to 255. The default is 8 connection failures.
<b>numclients</b>	(Optional) Number of unique client connection failures allowed before a real server is considered to have failed.
<i>number-clients</i>	(Optional) Client connection reassignment threshold value in the range from 1 to 8. The default is 2 client connection failures.

## Defaults

If you do not specify the **faildetect** command, the default value of the connection reassignment threshold is 8.

If you do not specify the **numclients** keyword, the default value of the unique client failure threshold is 2.

## Command Modes

Real server configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Examples

In the following example the connection reassignment threshold is set to 16 and, because the **number-clients** keyword is not configured, the threshold for unique client connection failure is set to the default value 8. The real server is considered to have failed when 8 unique clients have had connection failures and there have been 16 connection reassignments.

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1
Router(config-slb-real)# faildetect numconns 16
```

## Related Commands

Command	Description
<b>real</b>	Identifies a real server.
<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
<b>show ip slb reals</b>	Displays information about the real servers.

# idle

To specify the minimum amount of time IOS SLB maintains connection information in the absence of packet activity, use the **idle** virtual server configuration command. To restore the default idle duration value, use the **no** form of this command.

**idle** *duration*

**no idle**

<b>Syntax Description</b>	<i>duration</i>	Idle connection timer duration in seconds. Valid values range from 10 to 65535. The default is 3600 seconds (1 hour).
---------------------------	-----------------	---

<b>Defaults</b>	The default duration is 3600 seconds.
-----------------	---------------------------------------

<b>Command Modes</b>	Virtual server configuration
----------------------	------------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

<b>Usage Guidelines</b>	<p>TCP connections that do not send flows or keepalives before the idle timer expires are assumed to be inactive and are reset (RST).</p> <p>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.</p>
-------------------------	---

<b>Examples</b>	<p>The following example instructs IOS SLB to maintain connection information for an idle connection for 120 seconds.</p>
-----------------	---

```
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# idle 120
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>show ip slb vservers</b>	Displays information about the virtual servers.
	<b>virtual</b>	Configures the virtual server attributes.

## inservice (real server)

To enable the real server for use by IOS SLB, use the **inservice** real server configuration command. 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** If you do not specify the **inservice** command, the real server is defined to IOS SLB but is not used.

**Command Modes** Real server configuration

### Command History

Release	Modification
12.0(7)XE	This command was introduced.

### Examples

The following example enables the real server 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-real)# inservice
```

### Related Commands

Command	Description
<b>real</b>	Identifies a real server.
<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
<b>show ip slb reals</b>	Displays information about the real servers.

# inservice (virtual server)

To enable the virtual server for use by IOS SLB, use the **inservice** virtual server configuration command. 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	standby	(Optional) Configures the HSRP standby virtual server.
	<i>groupname</i>	(Optional) Specifies the HSRP group name with which the IOS SLB virtual server is associated.

**Defaults** If you do not specify the **inservice** command, the virtual server is defined to IOS SLB but is not used.

**Command Modes** Virtual server configuration

Command History	Release	Modification
	12.0(7)XE	This command was introduced.
	12.1(1)E	The <b>standby</b> keyword and <i>group-name</i> variable were added.

**Examples** The following example enables the real server for use by the IOS SLB feature:

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

Related Commands	Command	Description
	<b>show ip slb vservers</b>	Displays information about the virtual servers.
	<b>virtual</b>	Configures the virtual server attributes.

# ip slb dfp

To configure DFP and supply an optional password, use the **ip slb dfp** global configuration command. To remove the DFP configuration, use the **no** form of this command.

```
ip slb dfp [password password [timeout]]
```

```
no ip slb dfp
```

## Syntax Description

<b>password</b>	(Optional) Specifies a password for MD5 authentication.
<i>password</i>	(Optional) Password value for MD5 authentication. This password must match the password configured on the host agent.
<i>timeout</i>	(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 password timeout default is 180 seconds.

## Command Modes

Global configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Usage Guidelines

The optional password, if configured, must match the password configured on the host 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. This allows enough time for you to update the password on all agents and servers before the timeout expires. It 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 password to *flounder*, and configures a timeout period of 60 seconds and changes to DFP agent configuration mode:

```
Router(config)# ip slb dfp flounder 60
Router(config-slb-dfp)#
```

Related Commands	Command	Description
	agent	Configures a DFP agent.

# ip slb serverfarm

To identify a server farm and enter server farm configuration mode, use the **ip slb serverfarm** global configuration command. To remove the server farm from the IOS SLB configuration, use the **no** form of this command.

**ip slb serverfarm** *serverfarm-name*

**no ip slb serverfarm** *serverfarm-name*

## Syntax Description

<i>serverfarm-name</i>	Character string used to identify the server farm. The character string is limited to 15 characters.
------------------------	--

## Defaults

No default behavior or values.

## Command Modes

Global configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Examples

The following example identifies a server farm named PUBLIC:

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

## Related Commands

Command	Description
<b>real</b>	Identifies a real server.

# ip slb vserver

To identify a virtual server and enter virtual server configuration mode, use the **ip slb vserver** global configuration command. To remove a virtual server from the IOS SLB configuration, use the **no** form of this command.

**ip slb vserver** *virtserver-name*

**no ip slb vserver** *virtserver-name*

<b>Syntax Description</b>	<i>virtserver-name</i>	Character string used to identify the virtual server. The character string is limited to 15 characters.
---------------------------	------------------------	---

<b>Defaults</b>	No default behavior or values.
-----------------	--------------------------------

<b>Command Modes</b>	Global configuration
----------------------	----------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

**Examples** The following example identifies a virtual server named PUBLIC\_HTTP:

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

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>serverfarm</b>	Associates a real server farm with a virtual server.
	<b>show ip slb vservers</b>	Displays information about the virtual servers.

# maxconns

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

**maxconns** *maximum-number*

**no maxconns**

<b>Syntax Description</b>	<i>maximum-number</i>	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 is 4294967295.

**Command Modes** Real server configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

**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
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>real</b>	Identifies a real server.
	<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
	<b>show ip slb reals</b>	Displays information about the real servers.

# nat

To configure IOS SLB NAT and specify a NAT mode, use the **nat** server farm configuration command. To remove a NAT configuration, use the **no** form of this command.

**nat server**

**no nat server**

<b>Syntax Description</b>	<b>server</b>	Specifies that the destination address in load-balanced packets sent to the real server is the address of the real server chosen by the server farm load-balancing algorithm.
---------------------------	---------------	---

**Defaults** No default behavior or values.

**Command Modes** Server farm configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.1(1)E	This command was introduced.

**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 changes to IOS SLB 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
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>ip slb serverfarm</b>	Associates a real server farm with a virtual server.
	<b>real</b>	Identifies a real server as a member of a server farm.
	<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.

# predictor

To specify the load-balancing algorithm for selecting a real server in the server farm, use the **predictor** server farm configuration command. 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

<b>roundrobin</b>	(Optional) Use 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 3 for a detailed description of this algorithm.
<b>leastconns</b>	(Optional) Use 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 3 for a detailed description of this algorithm.

## Defaults

The default predictor is weighted round robin.

## Command Modes

Server farm configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Examples

The following example specifies the weighted least connections algorithm:

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

## Related Commands

Command	Description
<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
<b>weight</b>	Specifies the real server’s capacity, relative to other real servers in the server farm.

# real

To identify a real server as a member of a server farm, use the **real** server farm configuration command. To remove the real server from the IOS SLB configuration, use the **no** form of this command.

**real** *ip-address*

**no real** *ip-address*

<b>Syntax Description</b>	<i>ip-address</i>	Real server IP address.
---------------------------	-------------------	-------------------------

<b>Defaults</b>	No default behavior or values.
-----------------	--------------------------------

<b>Command Modes</b>	Server farm configuration
----------------------	---------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

**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
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>in service</b>	Enables the real server for use by IOS SLB.
	<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
	<b>show ip slb reals</b>	Displays information about the real servers.

# reassign

Use the **reassign** real server configuration command to specify the threshold of consecutive unanswered synchronizations that, if exceeded, result in an attempted connection to a different real server. To restore the default reassignment threshold, use the **no** form of this command.

**reassign** *threshold*

**no reassign**

## Syntax Description

*threshold*

Number of unanswered TCP SYNs that are directed to a real server before the connection is reassigned to a different real server. An unanswered SYN is one for which no SYN or ACK is detected before the next SYN arrives from the client. 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 on the **faildetect** command's **numconns** keyword is not exceeded. See the **faildetect** command for more information.

Valid threshold values range from 1 to 4 SYNs. The default value is 3.

## Defaults

The default threshold is 3 SYNs.

## Command Modes

Real server configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Examples

The following example sets the threshold of unanswered 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

Command	Description
<b>real</b>	Identifies a real server.
<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
<b>show ip slb reals</b>	Displays information about the real servers.

# retry

To specify how long to wait before a new connection is attempted to a failed server, use the **retry** real server configuration command. To restore the default retry value, use the **no** form of this command.

**retry** *retry-value*

**no** **retry**

<b>Syntax Description</b>	<i>retry-value</i>	<p>Time, in seconds, to wait after the detection of a server failure before a new connection to the server is attempted.</p> <p>If the new connection attempt succeeds, the real server is placed in <b>OPERATIONAL</b> 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 <b>OUTOFSERVICE</b> by the network administrator.</p> <p>Valid values range from 1 to 3600. The default value is 60 seconds.</p> <p>A value of 0 means do not attempt a new connection to the server when it fails.</p>
---------------------------	--------------------	--

**Defaults** The retry-value default is 60 seconds.

**Command Modes** Real server configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

**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
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>real</b>	Identifies a real server.
	<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
	<b>show ip slb reals</b>	Displays information about the real servers.

# serverfarm

To associate a real server farm with a virtual server, use the **serverfarm** virtual server configuration command. To remove the server farm association from the virtual server configuration, use the **no** form of this command.

**serverfarm** *serverfarm-name*

**no serverfarm**

<b>Syntax Description</b>	<i>serverfarm-name</i>	Name of a server farm that has already been defined using the <b>ip slb serverfarm</b> command.
---------------------------	------------------------	---

**Defaults** No default behavior or values.

**Command Modes** Virtual server configuration

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(7)XE	This command was introduced.

**Examples** 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
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>show ip slb vservers</b>	Displays information about the virtual servers.
	<b>virtual</b>	Configures the virtual server attributes.

# show ip slb conns

To display the active IOS SLB connections, use the **show ip slb conns** privileged EXEC command.

**show ip slb conns** [*vserver* *virtserver-name*] [*client* *ip-address*] [*detail*]

Syntax Description		
<b>vserver</b>	(Optional) Displays only those connections associated with a particular virtual server.	
<i>virtserver-name</i>	(Optional) Name of the virtual server to be monitored.	
<b>client</b>	(Optional) Displays only those connections associated with a particular client IP address.	
<i>ip-address</i>	(Optional) IP address of the client to be monitored.	
<b>detail</b>	(Optional) Displays detailed connection information.	

**Defaults** If no options are specified, the command displays output for all active IOS SLB connections.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows IOS SLB active connection data:

```
router# show ip slb conns
```

vserver	prot	client	real	state
TEST	TCP	7.150.72.183:328	80.80.90.25:80	CLOSING
TEST	TCP	7.250.167.226:423	80.80.90.26:80	CLOSING
TEST	TCP	7.234.60.239:317	80.80.90.26:80	CLOSING
TEST	TCP	7.110.233.96:747	80.80.90.26:80	CLOSING
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	CLOSING

**Table 1** *show ip slb conns* Command Field Information

Field	Description
vserver	Name of the virtual server whose connections are being monitored and displayed. Information about each connection is displayed on a separate line.
prot	Protocol being used by the connection.
client	Client IP address being used by the connection.

<b>Field (continued)</b>	<b>Description (continued)</b>
real	Real IP address of the connection.
state	<p>Current state of the connection:</p> <ul style="list-style-type: none"> <li>• CLOSING—IOS SLB TCP connection deactivated (awaiting a delay timeout before cleaning up the connection)</li> <li>• ESTAB—IOS SLB TCP connection processed a SYN-SYN/ACK exchange between the client and server</li> <li>• FINCLIENT—IOS SLB TCP connection processed a FIN from the client</li> <li>• FINSERVER—IOS SLB TCP connection processed a FIN from the server</li> <li>• INIT—Initial state of the IOS SLB TCP connection</li> <li>• SYNBOTH—IOS SLB TCP connection processed one or more TCP SYNs from both the client and the server</li> <li>• SYNCLIENT—IOS SLB TCP connection processed one or more client TCP SYNs</li> <li>• SYNSEVER—IOS SLB TCP connection processed one or more server 1 TCP SYNs</li> <li>• ZOMBIE—Destruction of the IOS SLB TCP connection failed, possibly because of bound flows. Destruction will proceed when the flows are unbound.</li> </ul>

# show ip slb dfp

To display DFP manager and agent information, such as passwords, timeouts, retry counts, and weights, use the **show ip slb dfp** privileged EXEC command.

**show ip slb dfp [agent *ip\_address* port | detail | weights]**

Syntax Description	
<b>agent</b>	(Optional) Displays information about an agent.
<i>ip_address</i>	(Optional) Agent IP address.
<i>port</i>	(Optional) Agent port number.
<b>detail</b>	(Optional) Displays all data available.
<b>weights</b>	(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	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows IOS SLB DFP data:

```
router# show ip slb dfp detail
DFP Manager:
  Current passwd:NONE Pending passwd:NONE
  Passwd timeout:0 sec
  Uned errors:0
DFP Agent 161.44.2.34:61936 Connection state:Connected
  Timeout = 0      Retry Count = 0      Interval = 180 (Default)
  Security errors = 0
  Last message received:10:20:26 UTC 11/02/99
  Last reported Real weights for Protocol TCP, Port www
    Host 17.17.17.17 1      Weight 1
    Host 68.68.68.68  Bind ID 4      Weight 4
    Host 85.85.85.85  Bind ID 5      Weight 5
  Last reported Real weights for Protocol TCP, Port 22
    Host 17.17.17.17  Bind ID 111     Weight 111
```

## ■ show ip slb dfp

```

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

```

```

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 2** show ip slb dfp Command Field Information

Field	Description
Agent IP	IP address of the agent about which information is being displayed.
Port	Port number of the agent.
Timeout	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.
Retry Count	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.
Interval	Interval, in seconds, between retries.

# show ip slb reals

To display information about the real servers, use the **show ip slb reals** privileged EXEC command.

**show ip slb reals** [*vserver vrtserver-name*] [**detail**]

Syntax Description		
<b>vserver</b>	(Optional) Displays information about only those real servers associated with a particular virtual server.	
<i>vrtserver-name</i>	(Optional) Name of the virtual server.	
<b>detail</b>	(Optional) Displays detailed information.	

**Defaults** If no options are specified, the command displays information about all real servers.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows IOS SLB real server data:

```
router# show ip slb reals
```

```

real          server farm    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
80.254.2.2    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   216
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 3** *show ip slb reals Command Field Information*

<b>Field</b>	<b>Description</b>
real	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.
server farm	Name of the server farm to which the real server.
weight	Weight assigned to the real server. The weight identifies the real server's capacity, relative to other real servers in the server farm.
state	Current state of the real server. <ul style="list-style-type: none"> <li>• DFP_THROTTLED—DFP agent sent a weight of <b>0</b> for this real server (send no further connections to this real server)</li> <li>• FAILED—Removed from use by the predictor algorithms; retry timer started</li> <li>• MAXCONNS—Maximum number of simultaneous active connections reached</li> <li>• OPERATIONAL—Functioning properly</li> <li>• OUTOFSERVICE—Removed from the load-balancing predictor lists</li> <li>• READY_TO_TEST—Queued for testing</li> <li>• TESTING—Queued for assignment</li> </ul>

# show ip slb serverfarms

To display information about the server farms, use the **show ip slb serverfarms** privileged EXEC command.

**show ip slb serverfarms** [**name** *serverfarm-name*] [**detail**]

Syntax Description	name	(Optional) Displays information about only a particular server farm.
	<i>serverfarm-name</i>	(Optional) Name of the server farm.
	<b>detail</b>	(Optional) Displays detailed server farm information.

**Defaults** No default behavior or values.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows IOS SLB server farm data:

```
router# show ip slb serverfarms

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

**Table 4** *show ip slb serverfarms* Command Field Information

Field	Description
server farm	Name of the server farm about which information is being displayed. Information about each server farm is displayed on a separate line.
predictor	Type of load-balancing algorithm (ROUNDROBIN or LEASTCONNS) used by the server farm.
reals	Number of real servers configured in the server farm.
bind id	Bind ID configured on the server farm.

# show ip slb stats

To display IOS SLB statistics, use the **show ip slb stats** privileged EXEC command.

**show ip slb stats**

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

**Defaults** No default behavior or values.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows IOS SLB statistics:

```
router# show ip slb stats
Pkts via normal switching: 530616
Pkts via special switching:1812710
Connections Created:      783774
Connections Established:  633418
Connections Destroyed:   782752
Connections Reassigned:   0
Zombie Count:            0
```

**Table 5** *show ip slb stats Command Field Information*

Field	Description
Pkts via normal switching	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, FastSwitching, and Process Level).
Pkts via special switching	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.
Connections Created	Number of connections created since the last time counters were cleared.
Connections Established	Number of connections created and have become established since the last time counters were cleared.
Connections Destroyed	Number of connections destroyed since the last time counters were cleared.

---

Connections Reassigned	Number of connections reassigned to a different real server since the last time counters were cleared.
Zombie Count	Number of connections that are currently pending destruction, awaiting a timeout or some other condition to be met.

---

# show ip slb sticky

To display the IOS SLB sticky database, use the **show ip slb sticky** privileged EXEC command.

```
show ip slb sticky [client ip_address]
```

Syntax Description	client	(Optional) Displays only those sticky database entries associated with a particular client IP address.
	ip-address	(Optional) IP address of the client.

**Defaults** If no options are specified, the command displays information about all virtual servers.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows the IOS SLB sticky database:

```
router# show ip slb sticky

client          group  real          conns  ftp-cntrl
-----
10.10.2.12      4097   10.10.3.2    1      0
```

**Table 6** show ip slb sticky Command Field Information

Field	Description
client	Client IP address which is bound to this sticky assignment.
group	Group ID for this sticky assignment.
real	Real server used by all clients connecting with the client IP address detailed on this line.
conns	Number of connections currently sharing this sticky assignment.
ftp-cntrl	Number of File Transfer Protocol (FTP) control connections currently using this sticky assignment.

# show ip slb vservers

To display information about the virtual servers, use the **show ip slb vservers** privileged EXEC command.

**show ip slb vservers** [*name virtserver-name*] [**detail**]

Syntax Description	name	(Optional) Displays information about only this virtual server.
	<i>virtserver-name</i>	(Optional) Name of the virtual server.
	<b>detail</b>	(Optional) Displays detailed information.

**Defaults** If no options are specified, the command displays information about all virtual servers.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples** The following example shows virtual server data:

```
router# show ip slb vservers

slb vservers      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 7** show ip slb vservers Command Field Information

Field	Description
slb vservers	Name of the virtual server about which information is being displayed. Information about each virtual server is displayed on a separate line.
prot	Protocol being used by the virtual server detailed on a given line.
virtual	Virtual IP address of the virtual server detailed on a given line.
state	Current state of the virtual server detailed on a given line.
conns	Number of connections associated with the virtual server detailed on a given line.

## standby name

To specify an HSRP group name with which to associate an IOS SLB interface, use the **standby name** interface configuration command. To remove the group name association on the interface, use the **no** form of this command.

**standby** [*group-number*] **name** *group-name*

**no standby** [*group-number*] **name** *group-name*

Syntax Description		
	<i>group-number</i>	(Optional) Group number of the interface to which the timers apply. The default is 0.
	<i>group-name</i>	Specifies the HSRP group name with which the IOS SLB virtual server is associated.

**Defaults** The default group number is 0.

**Command Modes** Interface configuration

Command History	Release	Modification
	12.1(1)E	This command was introduced.

**Usage Guidelines** The HSRP group name must first be specified on the **inservice (virtual server)** command.

**Examples** In the following example, HSRP is enabled for group number 1, group name Web-Group, on Ethernet port 0 on the EIP that is installed in slot 5:

```
Router(config)# interface Ethernet5/0
Router(config-if)# ip address 172.18.48.154 255.255.255.128
Router(config-if)# standby 1 ip 172.18.48.124
Router(config-if)# standby 1 priority 2 preempt
Router(config-if)# standby 1 name Web-Group
```

Related Commands	Command	Description
	<b>inservice</b>	Enables the virtual server for use by IOS SLB.

# sticky

To assign all connections from a client to the same real server, use the **sticky** virtual server configuration command. To remove the client/server coupling use the **no** form of this command.

**sticky** *duration* [**group** *group-id*]

**no sticky**

Syntax Description		
<i>duration</i>		Sticky timer duration in seconds. Valid values range from 0 to 65535.
<b>group</b>		(Optional) Places the virtual server in a sticky group, for coupling of services.
<i>group-id</i>		(Optional) Number identifying the sticky group to which the virtual server belongs. Valid values range from 0 to 255.

## Defaults

Sticky connections are not tracked.  
Virtual servers are not associated with any groups.

## Command Modes

Virtual server configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## 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.

## 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 s1b vserver VS1
Router(config-s1b-vserver)# sticky 60 group 10
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>show ip slb sticky</b>	Displays information about the virtual server or firewall farm sticky configuration.
	<b>show ip slb vservers</b>	Displays information about the virtual servers.
	<b>virtual</b>	Configures the virtual server attributes.

# synguard

To limit the rate of TCP SYNs handled by a virtual server to prevent a SYN flood denial of service attack, use the **synguard** virtual server configuration command. To remove the threshold, use the **no** form of this command.

```
synguard syn-count [interval]
```

```
no synguard
```

Syntax Description		
	<i>syn-count</i>	Number of unanswered SYNs that are allowed to be outstanding to a virtual server. Valid values range from 0 (off) to 4294967295. The default is 0.
	<i>interval</i>	(Optional) Interval, in milliseconds, for SYN threshold monitoring. Valid values range from 50 to 5000. The default is 100 ms.

**Defaults**

The default SYN count is 0 (off).  
The default interval is 100 ms.

**Command Modes**

Virtual server configuration

Command History	Release	Modification
	12.0(7)XE	This command was introduced.

**Examples**

The following example sets the threshold of unanswered SYNs to 50:

```
Router(config)# ip slb vserver PUBLIC_HTTP  
Router(config-slb-vserver)# synguard 50
```

Related Commands	Command	Description
	<b>show ip slb vservers</b>	Displays information about the virtual servers.
	<b>virtual</b>	Configures the virtual server attributes.

# virtual

To configure virtual server attributes, use the **virtual** virtual server configuration command. To remove the attributes, use the **no** form of this command.

**virtual** *ip-address* {**tcp** | **udp**} *port-number* [**service** *service-name*]

**no virtual**

Syntax Description	
<i>ip-address</i>	IP address for this virtual server instance, used by clients to connect to the server farm.
<b>tcp</b>	Perform load balancing for only TCP connections.
<b>udp</b>	Perform load balancing for only UDP connections.
<i>port-number</i>	<p>(Optional) 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 <i>port-number</i> argument are as follows:</p> <ul style="list-style-type: none"> <li>• Domain Name System: <b>dns 53</b></li> <li>• File Transfer Protocol: <b>ftp 21</b></li> <li>• HTTP over Secure Socket Layer: <b>https 443</b></li> <li>• Mapping of Airline Traffic over IP, Type A: <b>matip-a 350</b></li> <li>• Network News Transport Protocol: <b>nntp 119</b></li> <li>• Post Office Protocol v2: <b>pop2 109</b></li> <li>• Post Office Protocol v3: <b>pop3 110</b></li> <li>• Simple Mail Transport Protocol: <b>smtp 25</b></li> <li>• Telnet: <b>telnet 23</b></li> <li>• World Wide Web (HTTP): <b>www 80</b></li> </ul> <p>Specify a port number of <b>0</b> to configure an all-port virtual server (that is, a virtual server that accepts flows destined for all ports).</p>
<b>service</b>	(Optional) Couple connections associated with a given service, such as HTTP or Telnet, so all related connections from the same client use the same real server.
<i>service-name</i>	<p>(Optional) Type of connection coupling. Currently, the only choice is:</p> <ul style="list-style-type: none"> <li>• <b>ftp</b>—Couple FTP data connections with the control session that created them.</li> </ul>

**Defaults** No default behavior or values.

**Command Modes** Virtual server configuration

**Command History**

Release	Modification
12.0(7)XE	This command was introduced.

**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**.

**Note**

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 servers reject these flows, IOS SLB might fail the server and remove it from load balancing.

**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.

```
Router(config)# ip slb vserver PUBLIC_HTTP
Router(config-slb-vserver)# virtual 10.0.0.1 tcp www
```

**Related Commands**

Command	Description
<b>ip slb vserver</b>	Identifies a virtual server.
<b>show ip slb vservers</b>	Displays information about the virtual servers.

# weight

To specify a real server's capacity, relative to other real servers in the server farm, use the **weight** real server configuration command. To restore the default weight value, use the **no** form of this command.

**weight** *weighting-value*

**no weight**

## Syntax Description

<i>weighting-value</i>	Weighting value to use for real server predictor algorithm. Valid values range from 1 to 155. The default weighting value is 8.
------------------------	---

## Defaults

The default weighting-value is 8.

## Command Modes

Real server configuration

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Examples

The following example specifies the relative weighting values of three real servers as 16, 8 (by default), and 24, respectively:

```
Router(config)# ip slb serverfarm PUBLIC
Router(config-slb-sfarm)# real 10.10.1.1           First real server
Router(config-slb-real)# weight 16                Assigned weight of 16
Router(config-slb-real)# inservice                Enabled
Router(config-slb-real)# exit
Router(config-slb-sfarm)# real 10.10.1.2           Second real server
Router(config-slb-real)# inservice                Enabled; default weight
Router(config-slb-real)# exit
Router(config-slb-sfarm)# real 10.10.1.3           Third real server
Router(config-slb-real)# weight 24                Assigned weight of 24;
not enabled
```

## Related Commands

Command	Description
<b>real</b>	Identifies a real server.
<b>show ip slb serverfarms</b>	Displays information about the server farm configuration.
<b>show ip slb reals</b>	Displays information about the real servers.

# Debug Commands

This section documents the following new **debug** command related to the IOS SLB feature:

- **debug ip slb**, page 72

# debug ip slb

To display debug messages for IOS SLB, use the **debug ip slb** EXEC command. To stop debug output, use the **no** form of this command.

```
debug ip slb {conns | dfp | icmp | reals | all}
```

```
no debug ip slb {conns | dfp | icmp | reals | all}
```

## Syntax Description

<b>all</b>	Displays all debug messages for IOS SLB.
<b>conns</b>	Displays debug messages for all connections being handled by IOS SLB.
<b>dfp</b>	Displays debug messages for the IOS SLB DFP and DFP agents.
<b>icmp</b>	Displays all Internet Control Message Protocol debug messages for IOS SLB.
<b>reals</b>	Displays debug messages for all real servers defined to IOS SLB.

## Defaults

No default behavior or values.

## Command History

Release	Modification
12.0(7)XE	This command was introduced.

## Usage Guidelines

See the following caution before using **debug** commands:



### Caution

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, only use **debug** commands 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 debug 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 shows IOS SLB DFP debug output:

```
router# debug ip slb dfp
SLB DFP debugging is on
router#
022048 SLB DFP Queue to main queue - type 2 for Agent 161.44.2.3458229
022048 SLB DFP          select_rc = -1  readset = 0
022048 SLB DFP          Sleeping ...
022049 SLB DFP          readset = 0
022049 SLB DFP          select_rc = -1  readset = 0
022049 SLB DFP Processing Q event for Agent 161.44.2.3458229 - OPEN
022049 SLB DFP Queue to conn_proc_q - type 2 for Agent 161.44.2.3458229
022049 SLB DFP          readset = 0
022049 SLB DFP Set SLB_DFP_SIDE_QUEUE
022049 SLB DFP Processing Conn Q event for Agent 161.44.2.3458229 - OPEN
022049 SLB DFP Open to Agent 161.44.2.3458229 succeeded, socket = 0
022049 SLB DFP Agent 161.44.2.3458229 start connect
022049 SLB DFP Connect to Agent 161.44.2.3458229 successful - socket 0

022049 SLB DFP Queue to main queue - type 6 for Agent 161.44.2.3458229
022049 SLB DFP Processing Conn Q unknown MAJOR 80
022049 SLB DFP Reset SLB_DFP_SIDE_QUEUE
022049 SLB DFP          select_rc = -1  readset = 0
022049 SLB DFP          Sleeping ...
022050 SLB DFP          readset = 1
022050 SLB DFP          select_rc = 1  readset = 1
022050 SLB DFP Agent 161.44.2.3458229 fd = 0 readset = 1
022050 SLB DFP Message length 44 from Agent 161.44.2.3458229
022050 SLB DFP Agent 161.44.2.3458229 setting Host 17.17.17.17, Bind ID 1 Weight 1
022050 SLB DFP Agent 161.44.2.3458229 setting Host 34.34.34.34, Bind ID 2 Weight 2
022050 SLB DFP Agent 161.44.2.3458229 setting Host 51.51.51.51, Bind ID 3 Weight 3
022050 SLB DFP Processing Q event for Agent 161.44.2.3458229 - WAKEUP

022050 SLB DFP          readset = 1
022050 SLB DFP          select_rc = 1  readset = 1
022050 SLB DFP Agent 161.44.2.3458229 fd = 0 readset = 1
022050 SLB DFP Message length 64 from Agent 161.44.2.3458229
022050 SLB DFP Agent 161.44.2.3458229 setting Host 17.17.17.17, Bind ID 1 Weight 1
022050 SLB DFP Agent 161.44.2.3458229 setting Host 68.68.68.68, Bind ID 4 Weight 4
022050 SLB DFP Agent 161.44.2.3458229 setting Host 85.85.85.85, Bind ID 5 Weight 5
022050 SLB DFP Agent 161.44.2.3458229 setting Host 17.17.17.17, Bind ID 111 Weight 111
022050 SLB DFP          readset = 1

022115 SLB DFP Queue to main queue - type 5 for Agent 161.44.2.3458229
022115 SLB DFP          select_rc = -1  readset = 0
022115 SLB DFP          Sleeping ...
022116 SLB DFP          readset = 1
022116 SLB DFP          select_rc = -1  readset = 0
022116 SLB DFP Processing Q event for Agent 161.44.2.3458229 - DELETE
```

```
022116 SLB DFP Queue to conn_proc_q - type 5 for Agent 161.44.2.3458229
022116 SLB DFP          readset = 1
022116 SLB DFP Set SLB_DFP_SIDE_QUEUE
022116 SLB DFP Processing Conn Q event for Agent 161.44.2.3458229 - DELETE
022116 SLB DFP Connection to Agent 161.44.2.3458229 closed
022116 SLB DFP Agent 161.44.2.3458229 deleted
022116 SLB DFP Processing Conn Q unknown MAJOR 80
022116 SLB DFP Reset SLB_DFP_SIDE_QUEUE
022116 SLB DFP Set SLB_DFP_SIDE_QUEUE
022116 SLB DFP Reset SLB_DFP_SIDE_QUEUE
```

# Glossary

**CASA**—Content Aware 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.

**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.

**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**—Host agent 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**—Host manager 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 address through NAT. See also *dispatched mode*, *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*.

**Hot Standby Routing Protocol**—See *HSRP*.

**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*, *stateless backup*.

**IOS SLB**—IOS Server Load Balancing. Load-balancing scheme 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.

**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*.

**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.

**NetFlow switching**—High-performance network-layer switching path that captures as part of its switching function a rich set of flow statistics including user, protocol, port, and type of service information.

**Network Address Translation**—See *NAT*.

**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.

**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 *stateless backup*.

**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 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 *directed mode*.

**services manager**—Functionality built into IOS SLB that makes load-balancing decisions based on application availability, server capacity, and load distribution algorithms such as weighted round robin or weighted least connections, or the DFP. The services manager determines a real server for the packet flow using load balancing and server/application feedback.

**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.

**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.

**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 server**—Presents a single address that represents an application server farm to clients.

**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  $n_1/(n_1+n_2+n_3\dots)$ .

**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.

**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.

