Troubleshooting IOS SLB

Last Updated: April 26, 2011

Question	Answer
Can I use IOS SLB to load-balance clients and real servers that are on the same LAN or VLAN?	NO! IOS SLB does not support load balancing of flows between clients and real servers that are on the same LAN or VLAN. The packets being load-balanced cannot enter and leave the load-balancing device on the same interface.
Why is IOS SLB not marking my connections as ESTABLISHED even though I'm transferring data?	If you are using dispatched mode, make sure there are no alternate paths that allow outbound flows to bypass IOS SLB. Also, make sure the clients and real servers are not on the same IP subnet (that is, they are not on the same LAN or VLAN).
Why am I able to connect to real servers directly, but unable to connect to the virtual server?	Make sure that the virtual IP address is configured as a loopback in each of the real servers (if you are running in dispatched mode).
Why is IOS SLB not marking my real server as failed when I disconnect it from the network?	Tune the values for the **numclients**, **numconns**, and **delay** keywords. If you have a very small client population (for example, in a test environment), the **numclients** keyword could be causing the problem. This parameter prevents IOS SLB from mistaking the failure of a small number of clients for the failure of a real server.

Question	Answer
Why does IOS SLB show my real server as INSERVICE even though I have taken it down or physically disconnected it?	The INSERVICE and OUTOFSERVICE states indicate whether the network administrator *intends* for that real server to be used when it is operational. A real server that was INSERVICE but was removed from the selection list dynamically by IOS SLB as a result of automatic failure detection, is marked as FAILED. Use the **show ip slb reals detail** command to display these real server states. Beginning with release 12.1(1)E, INSERVICE is changed to OPERATIONAL, to better reflect what is actually occurring.
How can I verify that IOS SLB sticky connections are working properly?	Use the following procedure: 1 Configure the sticky connections. 2 Start a client connection. 3 Enter the **show ip slb reals detail** and **show ip slb conns** commands. 4 Examine the real server connection counts. The real server whose count increased is the one to which the client connection is assigned. 5 Enter the **show ip slb sticky** command to display the sticky relationships stored by IOS SLB. 6 End the connection. 7 Ensure that the real server's connection count decreased. 8 Restart the connection, after waiting no longer than the sticky timeout value. 9 Enter the **show ip slb conns** command again. 10 Examine the real server connection counts again, and verify that the sticky connection is assigned to the same real server as before.
How can I verify that server failures are being detected correctly?	Use the following procedure: 1 Use a large client population. If the number of clients is very small, tune the **numclients** keyword on the **faildetect numconns (real server)** command so that the servers are not displayed as FAILED. 2 Enter the **show ip slb reals detail** command to show the status of the real servers. 3 Examine the status and connection counts of the real servers. • Servers that failed show a status of FAILED, TESTING, or

Question	Answer
	READY_TO_TEST, based on whether IOS SLB is checking that the server came back up when the command was sent.
	• When a real server fails, connections that are assigned but not established (no SYN or ACK is received) are reassigned to another real server on the first inbound SYN after the **reassign** threshold is met. However, any connections that were already established are forwarded to the same real server because, while it might not be accepting new connections, it might be servicing existing ones.
	• For weighted least connections, a real server that has just been placed in service starts slowly so that it is not overloaded with new connections. (See the "Slow Start" section for more information.) Therefore, the connection counts displayed for a new real server show connections going to other real servers (despite the new real server's lower count). The connection counts also show "dummy connections" to the new real server, which IOS SLB uses to artificially inflate the connection counts for the real server during the slow start period.
Does the **no inservice** command take a resource out of service immediately?	When you use the **no** form of the **inservice** command to remove a firewall, firewall farm, real server, or virtual server from service, the resource acquiesces gracefully. No new connections are assigned, and existing connections are allowed to complete.
	To stop all existing connections for an entire firewall farm or virtual server immediately, use the **clear ip slb connections** command.
I configured both IOS SLB and input ACLs on the same Catalyst 6500 Family Switch, and now I see TCAM Capacity Exceeded messages. Why?	If you configure IOS SLB and either input ACLs or firewall load balancing on the same Catalyst 6500 Family Switch, you can exceed the capacity of the Telecommunications Access Method (TCAM) on the Policy Feature Card (PFC). To correct the problem, use the mls ip slb search wildcard rp command to reduce the amount of TCAM space used by IOS SLB, but be aware that this command can result in a slight increase in route processor use.

Question	Answer
Which IOS releases and platforms support IOS SLB VRF?	Virtual Private Network (VPN) routing and forwarding (VRF) for IOS SLB is supported in IOS release 12.2(18)SXE or later on the Supervisor Engine 720 with an MSFC3 (SUP720-MSFC3) for the Cisco 7600 series routers.
What can cause IOS SLB out-of-sync messages on the Supervisor?	If you are using one Supervisor Engine with **replicate slave** configured, you might receive out-of-sync messages on the Supervisor.
Can IOS SLB provide both firewall load balancing and RADIUS load balancing on the same Supervisor?	IOS SLB can provide both firewall load balancing and RADIUS load balancing on the same Supervisor Engine 720 (SUP720-MSFC3).