

Understanding CSM Load Balancing Algorithms

Document ID: 28580

Introduction

Before You Begin

- Conventions

- Prerequisites

- Components Used

Load-Balancing Algorithms

- Round Robin

- Least Connections

- Weighted Round Robin and Weighted Least Connections

- Source and/or Destination IP Hash (Subnet Mask Configurable)

- URL Hashing

- Forward

Related Information

Introduction

The Cisco Content Switching Module (CSM) supports load-balancing algorithms with the **predictor** command. Issue the **predictor** command in the Server Load Balancing (SLB) serverfarm configuration submode to specify the load-balancing algorithm for the server farm.

Before You Begin

Conventions

For more information on document conventions, see the Cisco Technical Tips Conventions.

Prerequisites

There are no specific prerequisites for this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information presented in this document was created from devices in a specific lab environment. All devices used in this document started with a cleared (default) configuration. If you are working in a live network, ensure that you understand the potential impact of any command before using it.

Load-Balancing Algorithms

Round Robin

The default algorithm, *round-robin*, keyword directs the network connection to the next server, and treats all servers as equals, regardless of the number of connections or response time. Although the CSM round-robin

predictor appears similar to a Domain Name System (DNS) round-robin, it is superior because no propagation delay or caching hinders the algorithm.

Least Connections

The *leastconns* keyword directs network connections to the server with the fewest connections. Although it may not be intuitively obvious that the *leastconns* predictor would provide effective load balancing, in fact, it is quite successful. On Web sites where there is a collection of servers with similar performance, the *leastconns* predictor is effective in smoothing distribution when a server becomes bogged down. On Web sites where there are large differences in the capacity of various servers, the *leastconns* predictor is also very effective. In maintaining the same number of connections to all servers, those servers that are capable of processing (and thus terminating) connections the fastest receive more connections over time. A server deemed to be twice as powerful as another server receives about twice as many connections per second.

Weighted Round Robin and Weighted Least Connections

The *weighted* keyword allows you to assign a performance weight to each server. Weighted load balancing is similar to the function of the *leastconns* and *round-robin* keywords, however, servers with a higher weight value receive a larger percentage of connections at any one time. CSM administrators can assign a weight to each real server, and the CSM uses this weight to determine the percentage of the current number of connections to give each server.

Issue the **weight** command in the SLB real server configuration submode to configure the capacity of the real servers in relation to the other real servers in the server farm. Issue the **no** form of this command to change the server's weight to its default capacity.

weighting-value is the value to use for the server farm predictor algorithm. The range is from 1 to 100. The default weight is 8. For example, in a configuration with five servers, the percentage of connections is calculated as follows:

Server Number	Number of Connections
Weight of server 1	7
Weight of server 2	8
Weight of server 3	2
Weight of server 4	2
Weight of server 5	5
Total weight of all servers	24

This distribution results in server 1 getting 7/24 of the current number of connections, server 2 getting 8/24, server 3 getting 2/24, and so on. If a new server, server 6, is added with a weight of 10, it receives 10/34, and so on.

Source and/or Destination IP Hash (Subnet Mask Configurable)

Source or destination IP hash method maps the IP address in the request to a real in a server farm. The configuration options are shown below.

```
predictor ip-hash netmask [source | destination] [netmask]
!--- Source IP hash or predictor hash address.
```

The default (netmask) is 255.255.255.255. The hash value is produced with the following algorithm:

```
ip_addr = (Src_or_Dest_Ip_addr & Cfg_Netmask) hash_index = (ip_addr) + (ip_addr >> 8) + ip_addr >> 16)
+ (ip_addr >> 24);
```

URL Hashing

URL hashing maps the URL (or portion) for the request to a real in a server farm. To specify only a portion of the URL, you need to configure the starting and/or ending keyword strings. This configuration is per vserver <vs-nam> object.

```
url-hash begin-pattern str [end-pattern str]
```

All characters in the keywords are included in the hash-value. In order to map it to a real, the CSM looks at the first few bits in the hash-value. If this real is disabled, the CSM finds the next available real in the list. In release 2.2(3) and later, the CSM looks at the higher bits in the hash-value if the first hash bits mapped to a disabled real. If the CSM cannot find an enabled real, it looks sequentially for the next available real.

Forward

Use the *forward* keyword to tell the CSM to forward traffic in accordance with its internal routing tables.

Note: The **nat server** command has no effect when the **predictor forward** command is configured; this is because the servers cannot be configured.

Related Information

- [Content Networking Devices Hardware Support](#)
- [Technical Support – Cisco Systems](#)

All contents are Copyright © 2006–2007 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement.

Updated: May 16, 2007

Document ID: 28580
