



CHAPTER 1

Introducing the Global Site Selector

This chapter describes the Cisco Global Site Selector (GSS) and introduces you to the terms and concepts necessary to help you understand and operate the GSS.

This chapter contains the following major sections:

- [GSS Overview](#)
- [DNS Routing](#)
- [Using the GSS as a DNS Appliance](#)
- [Globally Load Balancing with the GSS](#)
- [GSS Architecture](#)
- [DDoS Detection and Mitigation](#)
- [GSS Network Deployment](#)
- [GSS Network Management](#)
- [Global Server Load-Balancing Summary](#)
- [Where to Go Next](#)

GSS Overview

Server load-balancing devices, such as the Cisco Content Services Switch (CSS), Cisco Content Switching Module (CSM), and Cisco Application Control Engine (ACE) that are connected to a corporate LAN or the Internet, can balance content requests among two or more servers containing the same content. Server load-balancing devices ensure that the content consumer is directed to the host that is best suited to handle that consumer's request.

Organizations with a global reach or businesses that provide web and application hosting services require network devices that can perform complex request routing to two or more redundant, geographically dispersed data centers. These network devices need to provide fast response times and disaster recovery and failover protection through global server load balancing, or GSLB.

The Cisco Global Site Selector (GSS) platform allows you to leverage global content deployment across multiple distributed and mirrored data locations, optimizing site selection, improving Domain Name System (DNS) responsiveness, and ensuring data center availability.

The GSS is inserted into the traditional DNS routing hierarchy and is closely integrated with the Cisco CSS, Cisco CSM, Cisco ACE, or third-party server load balancers (SLBs) to monitor the health and load of the SLBs in your data centers. The GSS uses this information and user-specified routing algorithms to select the best-suited and least-loaded data center in real time.

The GSS can detect site outages, ensuring that web-based applications are always online and that customer requests to data centers that suddenly go offline are quickly rerouted to available resources.

The GSS offloads tasks from traditional DNS servers by taking control of the domain resolution process for parts of your domain name space, responding to requests at a rate of thousands of requests per second.

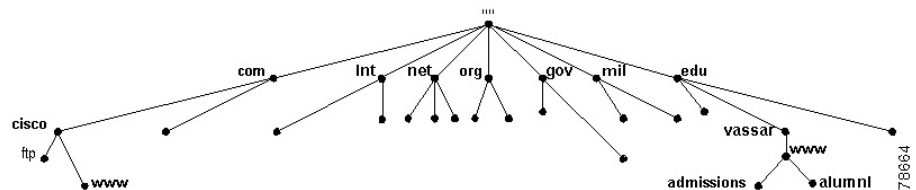
DNS Routing

This section explains some of the key DNS routing concepts behind the GSS.

Since the early 1980s, content routing on the Internet has been handled using the Domain Name System (DNS), a distributed database of host information that maps domain names to IP addresses. Almost all transactions that occur across the Internet rely on DNS, including electronic mail, remote terminal access such as Telnet, file transfers using the File Transfer Protocol (FTP), and web surfing. DNS uses alphanumeric hostnames instead of numeric IP addresses that bear no relationship to the content on the host.

With DNS, you can manage a nearly infinite number of hostnames referred to as the domain name space (see [Figure 1-1](#)). DNS allows local administration of segments (individual domains) of the overall database, but allows for data in any segment to be available across the entire network. This process is referred to as *delegation*.

Figure 1-1 Domain Name Space



This section contains the following topics:

- [DNS Name Servers](#)
- [DNS Structure](#)
- [Request Resolution](#)

DNS Name Servers

Information about the domain name space is stored on name servers that are distributed throughout the Internet. Each server stores the complete information about its small part of the total domain name space. This space is referred to as a DNS *zone*. A zone file contains DNS information for one domain (“mycompany.com”) or subdomain (“gslb.mycompany.com”).

The DNS information is organized into lines of information called resource records. Resource records describe the global properties of a zone and the hosts or services that are part of the zone. They are stored in binary format internally for use by the DNS software. However, resource records are sent across the network in a text format while they perform zone transfers.

Resource records are composed of various types of records including:

- Start of Authority (SOA)
- Name Service (NS)
- Address (A)
- Host Information (HINFO)
- Mail Exchange (MX)
- Canonical Name (CNAME)
- Pointer (PTR)

This document deals primarily with SOA and NS record types. For a detailed description of the other supported record types, as well as instructions for configuring resource records, see the *Cisco CNS Network Registrar User's Guide*. You can also consult RFC 1034 and 1035 for additional background information on resource records.

This section contains the following topics:

- [SOA Records](#)
- [Negative Caching](#)
- [SOA Records and Negative Responses](#)

SOA Records

At the top-level of a domain, the name database must contain a Start of Authority (SOA) record that identifies the best source of information for data within the domain. The SOA record also contains the current version of the DNS database and defines the behavior of a particular DNS server.

Each subdomain that is separately nameserved must have at least one corresponding NS record since name servers use these records to find each other. The zone is the region of the namespace that has a separate SOA. The format for this record is shown in the following example:

```
DOMAIN.NAME. IN SOA Hostname.Domain.Name. Mailbox.Domain.Name.
```

```
1 ; serno (serial number)
86400 ; refresh in seconds (24 hours)
7200 ; retry in seconds (2 hours)
2592000 ; expire in seconds (30 days)
345600 ; TTL in seconds (4 days)
```

Negative Caching

Busy servers have to handle hundreds or even thousands of name resolution requests each second. Therefore, it is essential that DNS server implementations employ mechanisms to improve their efficiency and cut down on unnecessary name resolution requests since each of these requests takes time and resources to resolve. Such requests also take internetwork bandwidth away from the business of transferring data.

Caching is one of the most important of these efficiency mechanisms. Caching refers to an area of memory set aside for storing information that has been recently obtained so it can be used again. In the case of DNS, caching is used by DNS name servers to store the results of recent name resolution and other requests, so that if the request occurs again it can be satisfied from the cache without requiring another complete run of the name resolution process. For more information, see the [“Request Resolution”](#) section.

Negative caching refers to the functions within a name server that maintain the nonexistence of specific DNS records. Negative caching stores negative results and reduces the response time for negative answers. It also reduces the number of messages sent between resolvers and name servers, reducing the amount of overall network traffic. Maintaining a negative cache state allows the system to quickly return a failure condition when a lookup attempt is retried.

Within the SOA record, the numeric Time to Live (TTL) fields control the frequency with which name servers poll each other to get information updates. For example, the TTL fields control the frequency with which the name servers poll each other to determine how long the data is cached. DNS allows name servers to distribute, and resolvers to cache, negative results with TTLs.

SOA record TTLs are required when forming negative responses for DNS queries since negative caching stores the knowledge that a resource record set (RRset), or domain name does not exist, or does not provide an answer.



Note

An RRset is a group of records that contain the same label, class, and type, but contains different data.

The most common negative responses indicate that a particular RRset does not exist in the DNS. Name errors (NXDOMAIN) are indicated by the presence of *name error* in the response code (RCODE) field, while NODATA is indicated by an answer with the RCODE sent to NOERROR and no relevant answers in the answer section. For such negative responses, GSS appends the SOA record of the zone in the authority section of the response.

SOA Records and Negative Responses

When the SOA record needs to be included in the negative response, the corresponding name server is queried for the SOA for the corresponding domain by the GSS. This SOA response is cached for a period mentioned in the minimum field of the SOA record. For all negative responses during this period, the cached SOA record is used, rather than querying the name server for the same domain.

**Note**

In GSS v2.0 or higher, the default behavior is to reply to queries with negative responses, whereas in GSS v1.3.3, the default is not to respond to negative queries.

If the GSS fails to obtain the SOA, the negative response is the appropriate error code. When using the cached SOA, the TTL of the negative response will be decremented by the time (in seconds) since the SOA was cached. This process is similar to the manner in which a caching-only name server decrements the TTL of the cached records.

**Note**

If you want to upgrade to GSS v3.0 but do not need any new DNS features and do not care what type of negative response will be returned for queries, you do not need to perform any additional SOA configuration. In such cases, GSS returns a type 3 negative response which does not contain the SOA information when the request cannot be answered.

To configure SOA records on the GSS to use in the negative response, you need to configure an NS answer that specifies the IP address of the authority name server for the domain and the domains hosted on the name server. See the [“Adding or Deleting an Authority Domain in an Answer Group”](#) section in [Chapter 6](#), for more details.

DNS Structure

End users who require data from a particular domain or machine generate a recursive DNS request on their client that is sent first to the local name service (NS), also referred to as the *D-proxy*. The D-proxy returns the IP address of the requested domain to the end user.

The DNS structure is based on a hierarchical tree structure that is similar to common file systems. The key components in this infrastructure are as follows:

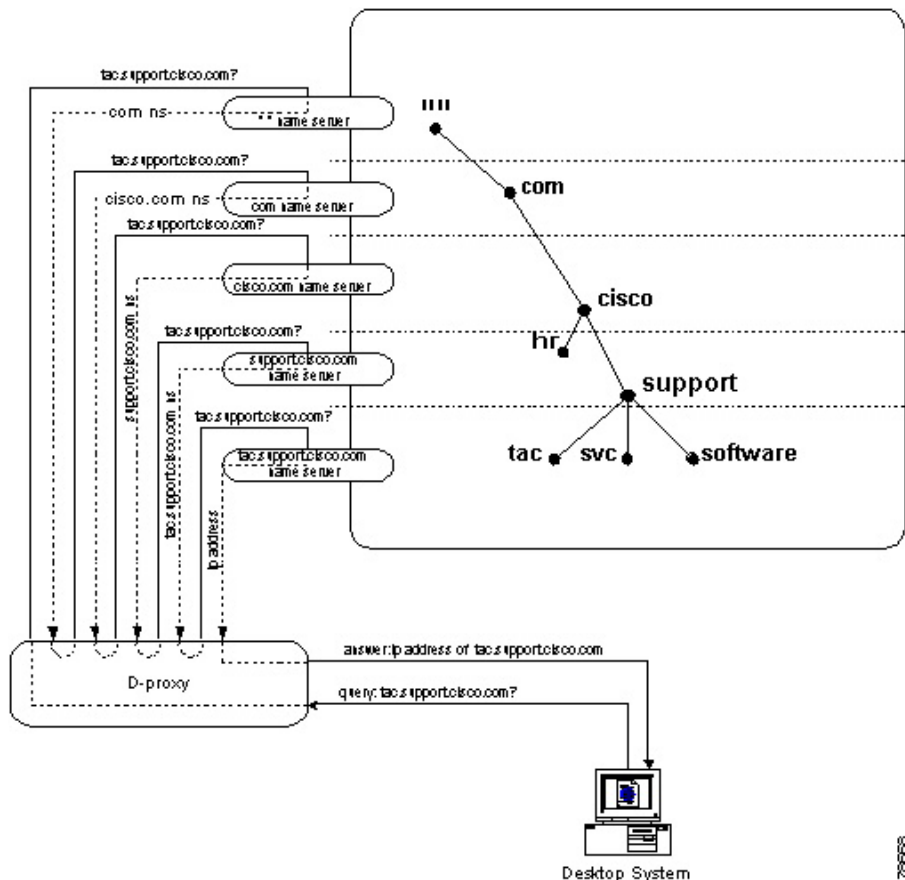
- **DNS Resolvers**—Clients that access client name servers.
- **Client Name Server**—Server that runs DNS software that has the responsibility of finding the requested web site. The client name server is also referred to as the client DNS proxy (D-proxy).
- **Root Name Servers**—Server that resides at the top of the DNS hierarchy. The root name server knows how to locate every extension after the period (.) in the hostname. There are many top-level domains. The most common top-level domains include .org, .edu, .net, .gov, and .mil. Approximately 13 root servers worldwide handle all Internet requests.
- **Intermediate Name Server**—Server that is used for scaling purposes. When the root name server does not have the IP address of the authoritative name server, it sends the requesting client name server to an intermediate name server. The intermediate name server then refers the client name server to the authoritative name server.
- **Authoritative Name Server**—Server that is run by an enterprise or outsourced to a service provider and is authoritative for the domain requested. The authoritative name server responds directly to the client name server (not to the client) with the requested IP address.

Request Resolution

If the local D-proxy does not have the information requested by the end user, it sends out iterative requests to the name servers that it knows are authoritative for the domains close to the requested domain. For example, a request for `www.cisco.com` causes the local D-proxy to check first for another name server that is authoritative for `www.cisco.com`.

[Figure 1-2](#) summarizes the sequence performed by the DNS infrastructure to return an IP address when a client tries to access the `www.cisco.com` website.

Figure 1-2 DNS Request Resolution



1. The resolver (client) sends a query for `www.cisco.com` to the local client name server (D-proxy).
2. The local D-proxy does not have the IP address for `www.cisco.com` so it sends a query to a root name server (“.”) asking for the IP address. The root name server responds to the request by doing one of the following:
 - Referring the D-proxy to the specific name server that supports the `.com` domain.

- Sending the D-proxy to an intermediate name server that knows the address of the authoritative name server for `www.cisco.com`. This method is referred to as an iterative query.
3. The local D-proxy sends a query to the intermediate name server that responds by referring the D-proxy to the authoritative name server for `cisco.com` and all the associated subdomains.
 4. The local D-proxy sends a query to the `cisco.com` authoritative name server that is the top-level domain. In this example, `www.cisco.com` is a sub-domain of `cisco.com`, so this name server is authoritative for the requested domain and sends the IP address to the name server (D-proxy).
 5. The name server (D-proxy) sends the IP address (172.16.56.76) to the client browser. The browser uses this IP address and initiates a connection to the `www.cisco.com` website.

Using the GSS as a DNS Appliance

GSS load balances geographically distributed data centers based on DNS requests. It also load balances any DNS-capable device that can be registered in the DNS system, such as origin servers, or third-party SLBs. For more information, see the [“Globally Load Balancing with the GSS”](#) section.

Typically, the GSS operates at a sublevel within the DNS hierarchy, responding only to a certain subset of DNS queries. Customers are then required to use a DNS server to process the other types of DNS queries.

With the v2.0 or higher release, GSS product capabilities have been enhanced to allow the GSS to migrate to the top level of the DNS hierarchy. This is accomplished through a product coupling with the Cisco Network Registrar (CNR) which permits the GSS to behave like a DNS appliance, thus simplifying the process of managing and configuring the DNS infrastructure.

The coupling can be viewed as two separate subsystems running on the same physical hardware with the GSS acting as the front-end DNS server and receiving all DNS requests.

Each query is processed as follows, depending upon its type:

- A Queries— The GSS processes these queries and responds if it finds a reply for the query. If it fails to find a reply, it queries the CNR subsystem for a reply. The CNR reply is then forwarded to the D-Proxy.

- All other Queries— These queries are forwarded to the CNR subsystem. The response from the CNR subsystem is forwarded back to the D-Proxy. If the response contains A records in the Additional Section, the GSS may perform its own query processing and modify the Additional Section of the Response to provide a load-balanced A records in the Additional Section.

For more information on CNR and GSS and their interaction and instructions on how to obtain and install a CNR license on the GSS, see the *Global Site Selector Administration Guide*.

Globally Load Balancing with the GSS

The GSS addresses critical disaster recovery requirements by globally load balancing distributed data centers. The GSS coordinates the efforts of geographically dispersed SLBs in a global network deployment for the following Cisco products:

- Cisco Content Services Switch (CSS) 11500, 11000, or 11150
- Cisco Content Switching Module (CSM) for the Catalyst 6500 series switches
- Cisco Application Control Engine (ACE) 4700 series appliance, ACE10-6500-K9 module, and ACE20-MOD-K9 module
- Cisco LocalDirector
- Cisco IOS SLB
- Cisco router using the DRP agent for network proximity
- Any server that is capable of responding to HTTP HEAD, ICMP, or TCP requests
- Cisco router with cache modules
- Cisco Cache Engines

The GSS supports over 4000 separate virtual IP (VIP) addresses. It coordinates the activities of SLBs by acting as the authoritative DNS server for those devices under its control.

Once the GSS becomes responsible for GSLB services, the DNS process migrates to the GSS. The DNS configuration is the same process as described in the “[Request Resolution](#)” section. The only exception is that the NS-records point to the GSSs located at each data center. The GSS determines which data center site should receive the client traffic.

As the authoritative name server for a domain or subdomain, the GSS considers the following additional factors when responding to a DNS request:

- Availability—Servers that are online and available to respond to the query
- Proximity—Server that responded to a query most quickly
- Load—Type of traffic load handled by each server in the domain
- Source of the Request—Name server (D-proxy) that requests the content
- Preference—First, second, or third choice of the load-balancing algorithm to use when responding to a query

This type of global server load balancing ensures that the end users are always directed to resources that are online, and that requests are forwarded to the most suitable device, resulting in faster response time for users.

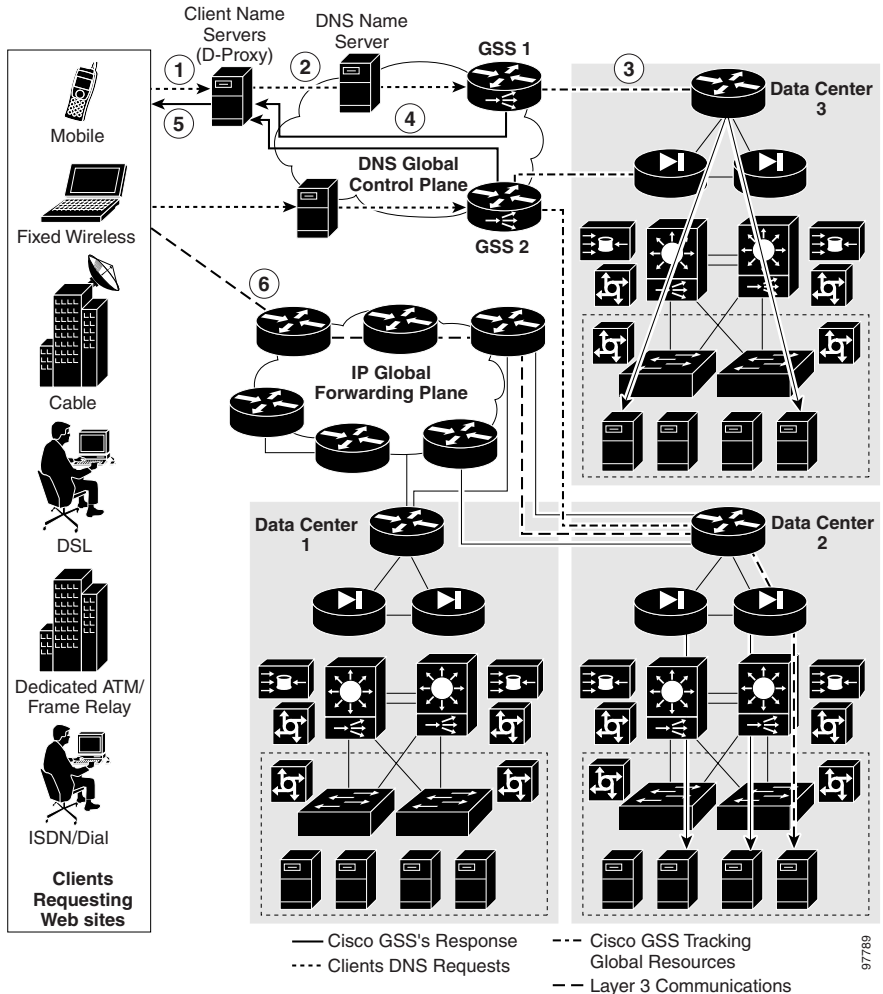
When resolving DNS requests, the GSS performs a series of distinct operations that take into account the resources under its control and return the best possible answer to the requesting client’s D-proxy.

[Figure 1-3](#) outlines how the GSS interacts with various clients as part of the website selection process to return the IP address of the requested content site.

1. A client starts to download an updated version of software from www.cisco.com and types **www.cisco.com** in the location or address field of the browser. This application is supported at three different data centers.
2. The DNS global control plane infrastructure processes the request and the request arrives at a GSS device.
3. The GSS sends the IP address of the “best” server load balancer to the client, in this case the SLB at Data Center 2.
4. The web browser processes the transmitted IP address.
5. The client is directed to the SLB at Data Center 2 by the IP control and forwarding plane.

- The GSS offloads the site selection process from the DNS global control plane. The request and site selection are based on the load and health information with user-controlled load-balancing algorithms. The GSS selects in real time a data center that is available and not overloaded.

Figure 1-3 *GLSB Using the Cisco Global Site Selector*



GSS Architecture

This section describes the key components of a GSS deployment, including hardware and software, as well as GSS networking concepts. It contains the following topics:

- [Global Site Selectors and Global Site Selector Managers](#)
- [DNS Rules](#)
- [Locations and Regions](#)
- [Owners](#)
- [Source Addresses and Source Address Lists](#)
- [Hosted Domains and Domain Lists](#)
- [Answers and Answer Groups](#)
- [Keepalives](#)
- [Balance Methods](#)
- [Traffic Management Load Balancing](#)

Global Site Selectors and Global Site Selector Managers

All GSS devices in the network, including the primary GSSM and standby GSSM, are delegated authority for domains, respond to DNS queries and perform keepalives, and use their local CLI for basic network management. All GSS devices depend on the primary GSSM to provide centralized, shared global server load-balancing functionality.

This section contains the following topics:

- Primary GSSM
- GSS
- Standby GSSM

Primary GSSM

The primary GSSM is a GSS that runs the GSS software. It performs content routing in addition to centralized management and shared global server load-balancing functions for the GSS network.

The primary GSSM hosts the embedded GSS database that contains configuration information for all your GSS resources, such as individual GSSs and DNS rules. All connected GSS devices report their status to the primary GSSM.

On the primary GSSM, you monitor and administer GSS devices using either of the following methods:

- CLI commands
- GUI (graphical user interface) functions, as described in the *Cisco Global Site Selector GUI-Based Global Server Load-Balancing Configuration Guide*

All configuration changes are communicated automatically to each device managed by the primary GSSM.

Any GSS device can serve as the single, primary GSSM on a configured system.

GSS

The GSS runs the GSS software and routes DNS queries based on DNS rules and conditions configured using the primary GSSM.

Each GSS is known to and synchronized with the primary GSSM.

You manage each GSS individually through its command-line interface (CLI). Support for the graphical-user interface (GUI) is not available on a GSS or on a standby GSSM.

Standby GSSM

The standby GSSM is a GSS that runs the GSS software and routes DNS queries based on DNS rules and conditions configured using the primary GSSM. Additionally, the standby GSSM is configured to function as the primary GSSM if the designated primary GSSM goes offline or becomes unavailable to communicate with other GSS devices.

When the standby GSSM operates as the interim primary GSSM, it contains a duplicate copy of the embedded GSS database currently installed on the primary GSSM. Both CLI and GUI support are also available on the standby GSSM once you configure it as the interim primary GSSM. While operating as the primary GSSM, you can monitor GSS behavior and make configuration changes, as necessary.

Any configuration or network changes that affect the GSS network are synchronized between the primary and the standby GSSM so the two devices are never out of sequence.

To enable the standby GSSM as the primary GSSM, use the **gssm standby-to-primary** CLI command. Ensure that your original primary GSSM is offline before you attempt to enable the standby GSSM as the new primary GSSM.

**Caution**

Having two primary GSSMs active at the same time may result in the inadvertent loss of configuration changes for your GSS network. If this dual primary GSSM configuration occurs, the two primary GSSMs revert to standby mode and you must reconfigure one of the GSSMs as the primary GSSM.

The standby GSSM can temporarily assume the role of the primary GSSM if the primary GSSM is unavailable (for example, you need to move the primary GSSM or you want to take it offline for repair or maintenance). Switching roles between the designated primary GSSM and the standby GSSM is intended to be a temporary GSS network configuration until the original primary GSSM can be brought back online. Once the original primary GSSM is available, reassign the two GSSMs to their original roles in the GSS network as described in the *Cisco Global Site Selector Administration Guide*.

DNS Rules

At the primary GSSM, you can configure DNS rules to do the following:

- Provide you with centralized command and control of how the GSS globally load balances a given hosted domain
- Define the IP addresses to send to the client's name server (D-proxy)
- Define the recovery method to use (using a maximum of three load-balance clauses)

Each DNS rule determines how the GSS responds to each query it receives by matching requests received from a known source, or D-proxy, to the most suitable member of a collection of name servers or virtual IP addresses (VIPs).

Each DNS rule takes into account the following variables:

- The source IP address of the requesting D-proxy.
- The requested hosted domain.
- An answer group, which is a group of resources considered for the response.
- A balance method, which is an algorithm for selecting the best server; a balance method and an answer group makes up a clause.
- Advanced traffic management load-balancing functions such as DNS sticky and network proximity.

A DNS rule defines how a request is handled by the GSS by answering the following question:

When traffic arrives from a DNS proxy, querying a specific domain name, which resources should be considered for the response, and how should they be balanced?

Each GSS network supports a maximum of 4000 DNS rules.

A maximum of three possible response answer group and balance method clauses are available for each DNS rule. Each clause specifies that a particular answer group serve the request and a specific balance method be used to select the best resource from that answer group. These clauses are evaluated in order, with parameters established to determine when a clause should be skipped if the first answer group and balance method specified does not yield an answer, and the next clause is to be used.

See [Chapter 7, Building and Modifying DNS Rules](#), for procedures on constructing the DNS rules that govern all global server load balancing on your GSS network.

Locations and Regions

As your GSS network expands, the job of organizing and administering your GSS resources—locations, regions, answers and answer groups, domain lists, and DNS rules—becomes more complex. The GSS provides the following features to help you organize your resources:

- Locations—Logical groupings for GSS resources that correspond to geographical areas such as a city, data center, or content site
- Regions—Higher-level geographical groupings that contain one or more locations

In addition to allowing you to easily sort and navigate long lists of answers and DNS rules, the use of logical groupings such as locations and regions makes it easier to perform bulk administration of GSS resources. For example, from the primary GSSM, you can suspend or activate all answers linked to a particular GSS data center, shutting down a site for scheduled maintenance and then bringing it back online with only a few mouse clicks.

See [Chapter 2, Configuring Resources](#), for information about configuring locations and regions.

Owners

An owner is an entity that owns web content and uses the GSS to manage access to the content. As locations and regions allow you to geographically configure your GSS network, owners allow you to organizationally configure your GSS network.

For example, a service provider using the GSS to manage multiple hosting sites might create an owner for each web- or application-hosting customer. With this organizational scheme, you can associate and manage the following elements through each owner: domain lists containing that owner's hosted content, DNS rules, answer groups, and source address lists that specify how traffic to those domains should be processed.

Deployed on a corporate intranet, you can configure owners to segregate GSS resources on a department-by-department basis, or to allocate specific resources to IT personnel. For example, you can create an owner for the finance, human resources, and sales departments so that resources corresponding to each can be viewed and managed together.

See [Chapter 2, Configuring Resources](#), for information about configuring owners.

Source Addresses and Source Address Lists

A source address refers to the source of DNS queries received by the GSS. Source addresses typically point to an IP address or block of addresses that represent client D-proxies from which the queries originate.

Using a DNS rule, the GSS matches source addresses to domains hosted by the GSS using one of a number of different balance methods.

Source addresses are taken from the D-proxy (the local name server) to which a requesting client issued a recursive request. The D-proxy sends the client queries to multiple name servers, eventually querying the GSS, which matches the D-proxy source address against its list of configured source addresses.

DNS queries received by the GSS do not have to match a specific D-proxy to be routed; default routing can be performed on requests that do not emanate from a known source address. By default, the GSS provides a fail-safe “Anywhere” source address list. Incoming queries that do not match your configured source address lists are matched to this list.

Source addresses are grouped into lists, referred to as source address lists, for the purposes of routing requests. Source address lists can contain 1 to 30 source addresses or unique address blocks. Each GSS supports a maximum of 60 source address lists.

See [Chapter 3, Configuring Source Address Lists](#), for information about configuring source address lists.

Hosted Domains and Domain Lists

A hosted domain (HD) is any domain or subdomain that has been delegated to the GSS and configured using the primary GSSM for DNS query responses. A hosted domain is a DNS domain name for which the GSS is authoritative.

All DNS queries must match a domain that belongs to a configured domain list, or the GSS denies the query. Queries that do not match domains on any GSS domain lists can also be forwarded by the GSS to an external DNS name server for resolution.

Hosted domain names are limited to 128 characters. The GSS supports domain names that use wildcards. The GSS also supports POSIX 1003.2-extended regular expressions when matching wildcards.

The following examples show domain or subdomain names configured on the GSS:

```
cisco.com
www.cisco.com
www.support.cisco.com
.*\.cisco\.com
```

Domain lists are groups of hosted domains that have been delegated to the GSS. Each GSS can support a maximum of 2000 hosted domains and 2000 hosted domain lists, with a maximum of 500 hosted domains supported for each domain list.

Domain lists are used by the GSS to match incoming DNS requests to DNS rules. After the query domain is found in a domain list and matched to a DNS rule, the balance method clauses of the DNS rule define how the GSS will choose the best answer (a VIP, for example) that can service the request.

See [Chapter 4, Configuring Domain Lists](#), for information about configuring domain lists.

Answers and Answer Groups

In a GSS network, answers refer to resources to which the GSS resolves DNS requests that it receives. The three types of possible answers on a GSS network are as follows:

- **VIP**—Virtual IP (VIP) addresses associated with an SLB such as the Cisco CSS, Cisco CSM, Cisco IOS-compliant SLB, Cisco LocalDirector, a web server, a cache, or any other geographically dispersed device in a global network deployment.
- **Name Server**—Configured DNS name server that can answer queries that the GSS cannot resolve.



Note When configured to operate in standalone mode, the GSS requires access to a properly configured name server to enable it to successfully operate and perform DNS resolutions. However, the GSS does not require a name server if you enable Cisco Network Registrar (CNR) on the GSS.

- **CRA**—Content routing agents that use a resolution process called DNS race to send identical and simultaneous responses back to a user's D-proxy.

As with domains and source addresses, answers are configured using the primary GSSM by identifying the IP address to which queries can be directed.

Once created, you group answers together as resource pools called answer groups. From the available answer groups, the GSS can use a maximum of three possible response answer group and balance method clauses in a DNS rule to select the most appropriate resource to serve a user request. Each balance method provides a different algorithm for selecting one answer from a configured answer group. Each clause specifies that a particular answer group serve the request and a specific balance method be used to select the best resource from that answer group.

Depending on the type of answer, further criteria can be applied to DNS queries to choose the best host. For example, a request that is routed to a VIP associated with a Cisco CSS is routed to the best resource based on load and availability, as determined by the CSS. A request that is routed to a content routing agent (CRA) is routed to the best resource based on proximity, as determined in a DNS race conducted by the GSS.

See [Chapter 6, Configuring Answers and Answer Groups](#), for information on configuring GSS answers and answer groups.

This section contains the following topics:

- [VIP Answers](#)
- [Name Server Answers](#)
- [CRA Answers](#)

VIP Answers

SLBs use VIP answers to represent content hosted on one or more servers under their control. The use of VIP answers enables the GSS to balance traffic among multiple origin servers, application servers, or transaction servers in a way that results in faster response times for users and less network congestion for the host.

When queried by a client's D-proxy for a domain associated with a VIP answer type, the GSS responds with the VIP address of the SLB best suited to handle that request. The requesting client then contacts the SLB, which load balances the request to the server best suited to respond to the request.

Name Server Answers

A name server answer specifies the IP address of a DNS name server to which DNS queries are forwarded from the GSS.

Using the name server forwarding feature, queries are forwarded to an external (non-GSS) name server for resolution, with the answer passed back to the GSS name server, then on to the requesting D-proxy. A name server answer can act as a guaranteed fallback resource, a way to resolve requests that the GSS cannot resolve itself. The GSS may not be able to resolve such requests for the following reasons:

- The requested content is unknown to the GSS.
- The resources that typically handle such requests are unavailable.

The external DNS name server answer forwarded by the GSS may be able to perform the following functions:

- Use DNS server features that are not supported by the GSS, such as mail exchanger (type MX) records
- Use a third-party content provider for failover and error recovery
- Provide access to a tiered DNS system

When a client D-proxy sends a query to a GSS that has CNR loaded and is also configured to use external name servers, the following sequence of actions occur:

1. The GSS performs a global server load balancing (GSLB) lookup on the query to see if the answer is contained within its database. The GSS performs one of the following actions depending on the answer type and the answer operating state:
 - If the answer type is VIP and the answer is online, the GSS sends the answer to the client D-proxy.
 - If the answer type is VIP and the answer is offline, the GSS retrieves and sends a fallback answer to the client D-proxy.
 - If the last clause answer type is VIP and the answer is offline, the GSS sends a SERVFAIL response to the client D-proxy.
 - If the answer type is NS, the GSS forwards the query to the name server called for in the NS Forwarding definition. The name server responds to the D-proxy through the GSS (the GSS acts as a proxy).

2. If the GSS performs a GSLB lookup and cannot find an answer to the query, the GSS behaves as follows depending on the CNR operating state:
 - CNR enabled—The GSS forwards the query to CNR and returns the CNR-supplied answer to the client D-proxy.
 - CNR disabled—The GSS sends a negative response (NXDOMAIN) to the client D-proxy.

CRA Answers

The CRA answer relies on content routing agents and the GSS to choose a suitable answer for a given query based on the proximity of two or more possible hosts to the requesting D-proxy.

With the CRA answer, requests received from a particular D-proxy are served by the content server that responds first to the request. Response time is measured using a DNS race, coordinated by the GSS and content routing agents running on each content server. In the DNS race, multiple hosts respond simultaneously to an A-record request. The server with the fastest response time (the shortest network delay between itself and the client's D-proxy) is chosen to serve the content.

The GSS requires the following information before it can initiate a DNS race:

- The delay between the GSS and each of the CRAs in each data center. With this data, the GSS computes how much time to delay the race from each data center so that each CRA starts the race simultaneously.
- The online status of the CRA through the use of keepalives.

The boomerang balance method uses the DNS race to determine the best site. See the [“DNS Race \(Boomerang\) Method”](#) section for more information on this balance method.

Keepalives

In addition to specifying a resource, each answer also provides you with the option of specifying a keepalive for that resource. A keepalive is the method by which the GSS periodically checks to determine if a resource is still active. A keepalive is a specific interaction (handshake) between the GSS and another device using a commonly supported protocol. A keepalive is designed to test if a specific protocol on the device is functioning properly. If the handshake is

successful, then the device is available, active, and able to receive traffic. If the handshake fails, then the device is considered to be unavailable and inactive. All answers are validated by configured keepalives and are not returned by the GSS to the D-proxy if the keepalive indicates that the answer is not viable.

The GSS uses keepalives to collect and track information from the online status of VIPs to services and applications running on a server. You can configure a keepalive to continually monitor the online status of a resource and report that information to the primary GSSM. Routing decisions involving that resource consider the reported online status information.

The GSS also supports the use of shared keepalives to minimize traffic between the GSS and the SLBs that it is monitoring. A shared keepalive identifies a common address or resource that can provide status for multiple answers. Shared keepalives are not used with name server or CRA answers.

When configuring a VIP-type answer, you have the option to configure one of several different keepalive types or multiple keepalive types to test for that answer. The primary GSSM supports the assignment of multiple keepalives and destination ports for a specific VIP answer. You can configure a maximum of five different keepalives for a VIP answer in a mix and match configuration of ICMP, TCP, HTTP HEAD, and KAL-AP VIP keepalive types. For TCP or HTTP HEAD keepalives, you may also specify different destination ports to a VIP server.

The following sections provide additional detail about keepalives on the GSS:

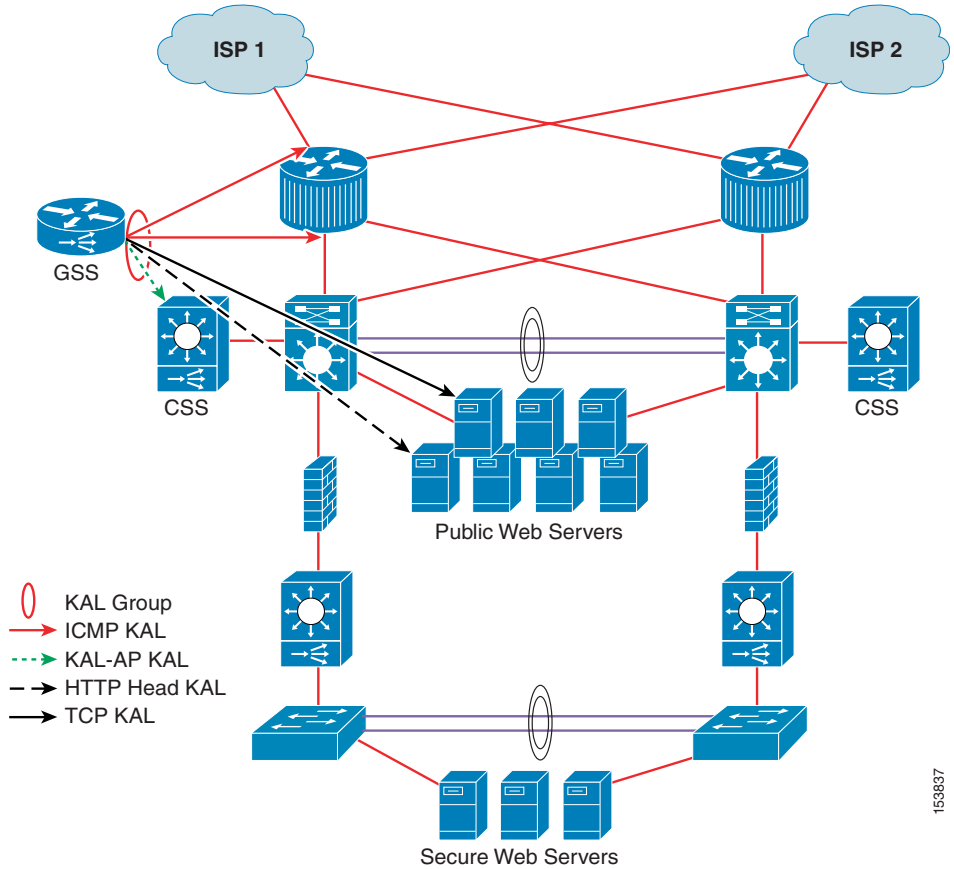
- [ICMP](#)
- [TCP](#)
- [HTTP HEAD](#)
- [KAL-AP](#)
- [Scripted Keepalive](#)
- [CRA](#)
- [Name Server](#)
- [None](#)
- [Adjusting Failure Detection Time for Keepalives](#)

Multiport Keepalives

GSS supports the ability to monitor multiple devices through the use of multiport keepalives for VIP-type answers. You can configure keepalives of different types to monitor multiple ports on the VIP server. You can also configure keepalives that specify IP addresses other than that of the VIP server (for example, a router, a back-end database server, a Catalyst 6500 series switch, or a CSS in a data center configuration).

Multiple keepalives, each configured to probe a specified device, but acting as a group, monitor the online status of your configuration. As long as all keepalives are successful, the GSS considers the configuration active and continues to direct traffic to the data center. See [Figure 1-4](#) for a keepalive configuration example that probes multiple devices on a data center.

Figure 1-4 Using Multiple Keepalives to Monitor a Data Center



153637

**Note**

The primary GSSM allows you to configure multiple shared keepalives, as well as a single KAL-AP keepalive when specifying multiple keepalive types.

See [Chapter 5, Configuring Keepalives](#), for information about modifying global keepalive parameters and creating shared keepalives.

ICMP

Use an ICMP keepalive when testing a GSS answer that is a VIP address, IP address, or a virtual server IP address. The Internet Control Message Protocol (ICMP) keepalive type monitors the health of resources by issuing queries containing ICMP packets to the configured VIP address (or a shared keepalive address) for the answer. Online status is determined by a response from the targeted address, indicating simple connectivity to the network. The GSS supports a maximum of 750 ICMP keepalives when using the standard detection method and a maximum of 150 ICMP keepalives when using the fast detection method. See the [“Adjusting Failure Detection Time for Keepalives”](#) section for details.

TCP

Use a TCP keepalive when testing a GSS answer that is a GSLB device that may be something other than a CSS or CSM. GSLB remote devices may include webservers, LocalDirectors, Wireless Application Protocol (WAP) gateways, and other devices that can be checked using a TCP keepalive. The TCP keepalive initiates a TCP connection to the remote device by performing the three-way handshake sequence.

Once the TCP connection is established, the GSS terminates the connection. You can choose to terminate the connection from two termination methods: Reset (immediate termination using a hard reset) or Graceful (standard three-way handshake termination).

The GSS supports a maximum of 1500 TCP keepalives when using the standard detection method and a maximum of 150 TCP keepalives when using the fast detection method. See the [“Adjusting Failure Detection Time for Keepalives”](#) section for details.

HTTP HEAD

Use an HTTP HEAD keepalive when testing a GSS answer that is an HTTP web server acting as a standalone device or managed by an SLB device such as a Cisco CSS, Cisco CSM, Cisco IOS-compliant SLB, or Cisco LocalDirector. The HTTP HEAD keepalive type sends a TCP-formatted HTTP HEAD request to a web server at an address that you specify. The online status of the device is returned in the form of an HTTP Response Status Code of 200 (for example, HTTP/1.0 200 OK).

Once the HTTP HEAD connection is established, the GSS terminates the connection. There are two methods to terminate the connection: Reset (immediate termination using a hard reset) or Graceful (standard three-way handshake termination).

The GSS supports a maximum of 500 HTTP HEAD keepalives when using the standard detection method and a maximum of 100 HTTP HEAD keepalives when using the fast detection method. See the [“Adjusting Failure Detection Time for Keepalives”](#) section for details.

KAL-AP

Use a KeepAlive-Appliance Protocol (KAL-AP) keepalive when testing a GSS answer that is a VIP associated with a Cisco CSS or a Cisco CSM. The KAL-AP keepalive type sends a detailed query to both a primary (master) and an optional secondary (backup) circuit address that you specify. The online status and load of each VIP that is specified in the KAL-AP keepalive are returned.

Depending on your GSS network configuration, you can use the KAL-AP keepalive to either query a VIP address directly (KAL-AP By VIP) or query an address with an alphanumeric tag (KAL-AP By Tag). Using a KAL-AP By Tag keepalive query can be useful in the following cases:

- You are attempting to determine the online status of a device that is located behind a firewall that is performing Network Address Translation (NAT).
- There are multiple content rule choices on the SLB.

The GSS supports a maximum of 128 primary and 128 secondary KAL-AP keepalives when using the standard detection method and a maximum of 40 primary and 40 secondary KAL-AP keepalives when using the fast detection method. See the [“Adjusting Failure Detection Time for Keepalives”](#) section for details.

Scripted Keepalive

Use a Scripted keepalive when you wish to probe third-party devices and obtain the load information. The Scripted keepalive uses the SNMP get request to fetch the load information from the target device.

**Note**

A Scripted keepalive must always be a shared keepalive.

The GSS supports a maximum of 384 Scripted keepalives when using the standard detection method and 120 Scripted keepalives when using the fast detection method. See the [“Adjusting Failure Detection Time for Keepalives”](#) section for details. Secondary Scripted keepalives are not supported in the GSS.

CRA

Use the CRA keepalive when testing a CRA answer that responds to DNS race requests. The CRA keepalive type tracks the time required (in milliseconds) for a packet of information to reach the CRA and return to the GSS. The GSS supports a maximum of 200 CRA keepalives.

Name Server

Use the name server keepalive to send a query to the IP address of the name server for a specified query domain (for example, www.cisco.com). The online status for the name server answer is determined by the ability of the name server for the query domain to respond to the query and assign the domain to an address. The GSS supports a maximum of 100 name server keepalives.

None

With the keepalive set to None, the GSS assumes that the answer is always online. Setting the keepalive type to None prevents your GSS from taking online status or load into account when routing. However, a keepalive of None can be useful under certain conditions, such as when adding devices to your GSS network that are not suited to other keepalive types. ICMP is a simple and flexible keepalive type that works with most devices. Using ICMP is often preferable to using the None option.

Adjusting Failure Detection Time for Keepalives

Failure detection time, for the GSS, is the amount of time between when a device fails (the answer resource goes offline) and when the GSS realizes the failure occurred. If a response packet fails to arrive back to the GSS within this window, the answer is marked offline.

The GSS supports two failure detection modes: standard and fast.

With standard mode, the failure detection time is typically 60 seconds before the GSS detects that a failure has occurred. Standard mode allows adjustment of the following parameters:

- **Response Timeout**—Length of time allowed before the GSS retransmits data to a device that is not responding to a request. The valid entries are 20 to 60 seconds. The default is 20 seconds.
- **Minimum Interval**—Minimum interval with which the GSS attempts to schedule a keepalive. The valid entries are 40 to 255 seconds. The default is 40 seconds.

With fast mode, the GSS controls the failure detection time by using the following keepalive transmission interval formula:

$$(\# \text{ Ack'd Packets} * (\text{Response TO} + (\text{Retry TO} * \# \text{ of Retries}))) + \text{Timed Wait}$$

where:

Ack'd Packets = Number of packets that require some form of acknowledgement

Response TO = Response Timeout, which is the length of time to wait for a reply for a packet that requires an acknowledgement

Retry TO = Retry Timeout, which is the length of time to wait for a reply for a retransmitted packet

of Retries = Number of Retries, which is the number of times the GSS retransmits packets to a potentially failed device before declaring the device offline

Timed Wait = Time for the remote side of the connection to close (TCP-based keepalive only)

[Table 1-1](#) summarizes how the GSS calculates the fast keepalive transmission rates for a single keepalive per answer.

Table 1-1 *Keepalive Transmission Rates for a Single Keepalive Per Answer*

	# Ack'd Packets (Fixed Value)	Response TO (Fixed Value)	Retry TO (Fixed Value)	# of Retries (User Selectable)	Timed Wait (Fixed Value)	Transmission Interval
KAL-AP	1	2 seconds	2 seconds	1	0	4 seconds
ICMP	1	2 seconds	2 seconds	1	0	4 seconds

Table 1-1 *Keepalive Transmission Rates for a Single Keepalive Per Answer (continued)*

	# Ack'd Packets (Fixed Value)	Response TO (Fixed Value)	Retry TO (Fixed Value)	# of Retries (User Selectable)	Timed Wait (Fixed Value)	Transmission Interval
TCP (RST)	1	2 seconds	2 seconds	1	0	4 seconds
TCP (FIN)	2	2 seconds	1 second	1	2 seconds	10 seconds
HTTP HEAD (RST)	2	2 seconds	2 seconds	1	0	8 seconds
HTTP HEAD (FIN)	3	2 seconds	2 seconds	1	2 seconds	14 seconds

For a TCP (RST) connection, the default transmission interval for a TCP keepalive is as follows:

$$(1 * (2 + (2 * 1))) + 0 = 4 \text{ seconds}$$

You can adjust the number of retries for the ICMP, TCP, HTTP HEAD, and KAL-AP keepalive types. The number of retries defines the number of times that the GSS retransmits packets to a potentially failed device before declaring the device offline. The GSS supports a maximum of 10 retries, with a default of 1. As you adjust the number of retries, you change the detection time determined by the GSS. By increasing the number of retries, you increase the detection time. Reducing the number of retries decreases the detection time.

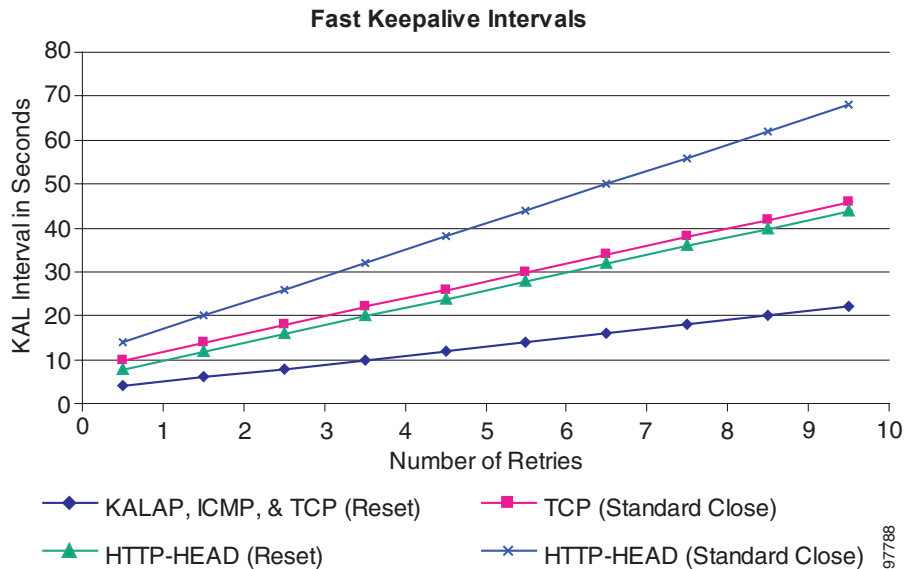
The GSS associates the number of retries value with every packet that requires some form of acknowledgement before continuing with a keepalive cycle (ICMP requests, TCP SYN, or TCP FIN). For example, to fully complete a TCP-based keepalive cycle, the TCP-based keepalive retries the SYN packet for the specified number of retries and then retries the FIN packet for the specified number of retries.

In the above example of a TCP (RST) connection, if you change the number of retries from the default value of 1 to a setting of 5, the transmission interval would be as follows:

$$(1 * (2 + (2 * 5))) + 0 = 12 \text{ seconds}$$

Figure 1-5 shows the effect on the keepalive transmission interval as you increase the number of retries value.

Figure 1-5 Effect of the Number of Retries Value on the Keepalive Transmission Interval



You can also define the number of consecutive successful keepalive attempts (probes) that must occur before the GSS identifies that an offline answer is online. The GSS monitors each keepalive attempt to determine if the attempt was successful. The **successful-probes** keyword identifies how many consecutive successful keepalive attempts the GSS must recognize before bringing an answer back online and reintroducing it back into the GSS network.

The primary GSSM allows you to assign multiple keepalives for a single VIP answer. You can configure a maximum of five different keepalives for a VIP answer in a mix and match configuration of ICMP, TCP, HTTP HEAD, and KAL-AP VIP keepalive types. In this configuration, the failure detection times are based on the calculated transmission levels identified for each of the different keepalives associated with an answer.

Balance Methods

The GSS supports six unique balance methods that allow you to specify how a GSS answer should be selected to respond to a given DNS query. Each balance method provides a different algorithm for selecting one answer from a configured answer group. This section explains the balance methods supported by the GSS and includes the following topics:

- [Ordered List Method](#)
- [Round-Robin Method](#)
- [Weighted Round-Robin Method](#)
- [Least-Loaded Method](#)
- [Hashed Method](#)
- [DNS Race \(Boomerang\) Method](#)

Ordered List Method

When the GSS uses the ordered list balance method, each resource within an answer group (for example, an SLB VIP or a name server) is assigned a number that corresponds to the rank of that answer within the group. The number you assign represents the order of the answer on the list. Subsequent VIPs or name servers on the list are only used if preceding VIPs or name servers on the list are unavailable. The GSS supports gaps in numbering in an ordered list.

**Note**

For answers that have the same order number in an answer group, the GSS uses only the first answer that contains the number. You should specify a unique order number for each answer in an answer group.

Using the ranking of each answer, the GSS tries each resource in the order that has been assigned, selecting the first available live answer to serve a user request. List members are given precedence and tried in order, and a member is not used unless all previous members fail to provide a suitable result.

The ordered list method allows you to manage resources across multiple content sites in which a deterministic method for selecting answers is required.

See the [“Balance Method Options for Answer Groups”](#) section for information about how the GSS determines which answer to select when using the ordered list balance method.

Round-Robin Method

When the GSS uses the round-robin balance method, each resource within an answer group is tried in turn. The GSS cycles through the list of answers, selecting the next answer in line for each request. In this way, the GSS can resolve requests by evenly distributing the load among possible answers.

The round-robin balance method is useful when balancing requests among multiple, active data centers that are hosting identical content; for example, between SLBs at a primary and at an active standby site that serves requests.

See the [“Balance Method Options for Answer Groups”](#) section for information about how the GSS determines which answer to select when using the round-robin balance method.

Weighted Round-Robin Method

As performed by the round-robin balance method, the weighted round-robin method also cycles through a list of defined answers to choose each available answer in turn. However, with weighted round-robin, an additional weight factor is assigned to each answer, biasing the GSS toward certain servers so that they are used more often.

See the [“Balance Method Options for Answer Groups”](#) section for information about how the GSS determines which answer to select when using the weighted round-robin balance method.

Least-Loaded Method

When the GSS uses the least-loaded balance method, the GSS resolves requests to the least loaded of all resources, as reported by the KAL-AP or Scripted keepalive process, which provides the GSS with detailed information on the SLB load and availability.

The least-loaded balance method resolves the request by determining the least number of connections on a CSM or the least-loaded CSS.

See the “[Balance Method Options for Answer Groups](#)” section for information about how the GSS determines which answer to select when using the least-loaded balance method.

Hashed Method

When the GSS uses the hashed balance method, elements of the client’s DNS proxy IP address and the requesting client’s domain are extracted to create a unique value, referred to as a hash value. The unique hash value is attached to and used to identify a VIP that is chosen to serve the DNS query.

The use of hash values makes it possible to stick traffic from a particular requesting client to a specific VIP, ensuring that future requests from that client are routed to the same VIP. This type of continuity can be used to facilitate features, such as online shopping baskets, in which client-specific data is expected to persist even when client connectivity to a site is terminated or interrupted.

The GSS supports the following two hashed balance methods. You can apply one or both hashed balance methods to the specified answer group:

- **By Source Address**—The GSS selects the answer based on a hash value created from the source address of the request.
- **By Domain Name**—The GSS selects the answer based on a hash value created from the requested domain name.

DNS Race (Boomerang) Method

The GSS supports the DNS race (boomerang) method of proximity routing, which is a type of DNS resolution initiated by the GSS to load balance 2 to 20 sites.

The boomerang method is based on the concept that instantaneous proximity can be determined if a CRA within each data center sends an A-record (IP address) at the exact same time to the client’s D-proxy. The DNS race method of DNS resolution gives all CRAs (Cisco content engines or content services switches) a chance at resolving a client request and allows for proximity to be determined without probing the client’s D-proxy. The first A-record received by the D-proxy is, by default, considered to be the most proximate.

For the GSS to initiate a DNS race, it needs to establish the following information for each CRA:

- The delay between the GSS and each of the CRAs in each data center. With this data, the GSS computes the length of time to delay the race from each data center, so that each CRA starts the race simultaneously.
- The online status of the CRAs. With this data, the GSS knows not to forward requests to any CRA that is not responding.

The boomerang server on the GSS gathers this information by sending keepalive messages at predetermined intervals. The boomerang server uses this data, along with the IP addresses of the CRAs, to request the exact start time of the DNS race.

If the CRA response is to be accepted by the D-proxy, each CRA must spoof the IP address of the GSS to which the original DNS request was sent.

Balance Method Options for Answer Groups

For most balance methods supported by the GSS, there are additional configuration options when you group specific answers in an answer group. These configuration options ensure the GSS properly applies the balance method for answers, and that you receive the best possible results from your GSS device.

[Table 1-2](#) describes the available answer group options for each answer type (VIP, CRA, or NS) and balance method combination.

Table 1-2 *Answer Group Options*

Answer Type	Balance Methods Used	Answer Group Options
VIP	Hashed	Order
	Least-loaded	Load threshold
	Ordered list	Weight
	Round-robin	
	Weighted round-robin	
Name server	Hashed	Order
	Ordered list	Weight
	Round-robin	
	Weighted round-robin	
CRA	Boomerang (DNS race)	None

This section explains each of the options available for the answers in an answer group. It contains the following topics:

- [Order](#)
- [Weight](#)
- [Load Threshold](#)

Order

Use the Order option when the balance method for the answer group is Ordered List. Answers on the list are given precedence based upon their position in the list in responding to requests.

Weight

Use the answer group Weight option when the balance method for the answer group is weighted round-robin or least-loaded. You specify a weight by entering a value from 1 and 10. This value indicates the capacity of the answer to respond to requests. The weight creates a ratio that the GSS uses when directing requests to each answer. For example, if Answer A has a weight of 10 and Answer B has a weight of 1, Answer A receives 10 requests for every 1 request directed to Answer B.

When you specify a weight for the weighted round-robin balance method, the GSS creates a ratio of the number of times that the answer is used to respond to a request before trying the next answer on the list.

When you specify a weight for the least-loaded balance method, the GSS uses that value as the divisor for calculating the load number associated with the answer. The load number creates a bias in favor of answers with a greater capacity.

Load Threshold

Use the Load Threshold option when the answer type is VIP and the keepalive method is KAL-AP to determine whether an answer is available, regardless of the balance method used. The load threshold is a number from 2 and 254 that is compared to the load being reported by the answer device. If the reported load is greater than the specified threshold, the answer is considered offline and unavailable to serve further requests.

Traffic Management Load Balancing

The GSS includes DNS sticky and network proximity traffic management functions to provide advanced global server load-balancing capabilities in a GSS network.

DNS sticky ensures that e-commerce sites provide undisrupted services and remain open for business by supporting persistent sticky network connections between customers and e-commerce servers. Persistent network connections ensure that active connections are not interrupted and shopping carts are not lost before purchase transactions are completed.

Network proximity selects the closest or most proximate server based on measurements of round-trip time to the requesting client's D-proxy location, improving the efficiency within a GSS network. The proximity calculation is typically identical for all requests from a given location (D-proxy) if the network topology remains constant. This approach selects the best server based on a combination of site health (availability and load) and the network distance between a client and a server zone.

This section contains the following topics:

- [DNS Sticky GSLB](#)
- [Network Proximity GSLB](#)

DNS Sticky GSLB

Stickiness, also known as persistent answers or answer caching, enables a GSS to remember the DNS response returned for a client D-proxy and to later return that same answer when the client D-proxy makes the same request. When you enable stickiness in a DNS rule, the GSS makes a best effort to always provide identical A-record responses to the requesting client D-proxy, assuming that the original VIP continues to be available.

DNS sticky on a GSS ensures that e-commerce clients remain connected to a particular server for the duration of a transaction even when the client's browser refreshes the DNS mapping. While some browsers allow client connections to remain for the lifetime of the browser instance or for several hours, other browsers impose a connection limit of 30 minutes before requiring a DNS re-resolution. This time may not be long enough for a client to complete an e-commerce transaction.

With local DNS sticky, each GSS device attempts to ensure that subsequent client D-proxy requests to the same domain name to the same GSS device will be stuck to the same location as the first request. DNS sticky guarantees that all requests from a client D-proxy to a particular hosted domain or domain list are given the same answer by the GSS for the duration of a user-configurable sticky inactivity time interval, assuming the answer is still valid.

With global DNS sticky enabled, each GSS device in the network shares answers with the other GSS devices in the network, operating as a fully connected peer-to-peer mesh. Each GSS device in the mesh stores the requests and responses from client D-proxies in its own local database and shares this information with the other GSS devices in the network. As a result, subsequent client D-proxy requests to the same domain name to any GSS in the network causes the client to be stuck.

The DNS sticky selection process is initiated as part of the DNS rule balance method clause.

See [Chapter 8, Configuring DNS Sticky](#) for information about configuring local and global DNS sticky for GSS devices in your network.

Network Proximity GSLB

The GSS responds to DNS requests with the most proximate answers (resources) relative to the requesting D-proxy. In this context, proximity refers to the distance or delay in terms of network topology (not geographical distance) between the requesting client's D-proxy and its answer.

To determine the most proximate answer, the GSS communicates with a proximity probing agent, a Cisco IOS-based router or another GSS configured as a DRP agent, located in each proximity zone to gather round-trip time (RTT) metric information measured between the requesting client's D-proxy and the zone. Each GSS directs client requests to an available server with the lowest RTT value.

The proximity selection process is initiated as part of the DNS rule balance method clause. When a request matches the DNS rule and balance clause with proximity enabled, the GSS responds with the most proximate answer.

See [Chapter 9, Configuring Network Proximity](#) for information about configuring proximity for GSS devices in your network.

DDoS Detection and Mitigation

Distributed Denial of Service (DDoS) attacks are designed to deny legitimate users access to a specific computer or network resource by flooding the target with traffic from a single host or from multiple hosts. These attacks may send several thousand spoofed DNS requests to a target device. The target then treats these requests as valid and processes the malicious requests.

Because the target is busy replying to the attack, it drops valid DNS requests from legitimate D-proxies. When the number of requests is in the thousands, the attack can potentially generate a multi-gigabit flood of DNS replies, thus causing network congestion.

In such cases, the following network points are affected:

- The performance of the target device is degraded because it is busy processing spoofed requests.
- The traffic generated by the replies traverses the internet backbone affecting the ISP and any upstream providers.
- A host with an IP address similar to the one used in the spoofing operation receives large amounts of inbound DNS traffic.

To combat such problems, the GSS contains a licensed DDoS detection and mitigation module. For more information about obtaining and installing a DDoS license, see the *Global Site Selector Administration Guide*.

Typically, the DDoS module prevents the following types of attacks:

- Reflector attacks where the attacker spoofs the IP address of the victim (that is, the GSS). See the “[Mitigation Rules](#)” section for more information.
- Attacks where malformed DNS packets are transmitted
- Attacks where DNS queries are sent:
 - For any domain (that is, a DoS replay attack) from a specific source IP.
 - For domains not configured on the GSS
 - From different source IPs globally exceeding the GSS packet processing rate.
 - From spoofed IP addresses

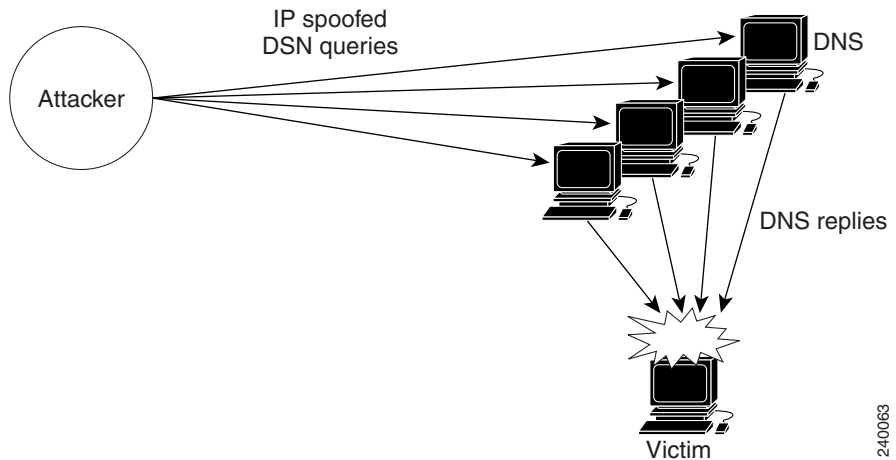
The DDoS module prevents these attacks by performing three primary functions, each of which is explained in the sections that follow:

- [Mitigation Rules](#)
- [Rate Limits](#)
- [Anti-Spoofing Mechanism](#)

Mitigation Rules

A reflector attack occurs when the attacker spoofs the IP address of the victim (in this case, the GSS) and sends multiple DNS requests to a DNS server or multiple DNS servers posing as the victim (see [Figure 1-6](#)). The amplification effect is based on the fact that small queries can generate larger UDP packets in response and bombard the victim with a high-volume of DNS response traffic.

Figure 1-6 *Reflector Attack Diagram*



240063

The following GSS basic mitigation rules help reduce the reflector problem:

- DNS packets are dropped if they come from a source port other than 53.
- DNS packets are dropped if they have a destination port of 53.
- DNS packets are dropped with a source port neither equal to 53 nor greater than 1024.

By default, mitigation rules are enabled. For more information on enabling mitigation, see [Chapter 10, Configuring DDoS Prevention](#).

Rate Limits

The GSS enforces a limit on the number of DNS packets per minute for each individual D-proxy and an overall global rate limit. The final rate limits for each D-proxy and the global rate limit are determined by multiplying the rate limits learned during peacetime (or configured via the CLI) with a tolerance factor. You can configure this value by using the **rate-limit global** and **scaling-factor global** CLI commands in `ddos` configuration mode.

The GSS also enforces a rate limit (unknown rate-limit) that limits the number of anti-spoofing tests to be performed by the GSS in a minute. Once this limit is reached, the GSS drops DNS packets from new sources during that minute. By default, the GSS performs spoof tests for 1000 new D-proxies per minute. You can change this limit by configuring the unknown rate limit.

The GSS enforces rate limits for DNS traffic only; it does not enforce limits for all traffic. You can configure the rate limit for DNS packets from a particular D-proxy only by providing the IP address.

For more details, see [Chapter 10, Configuring DDoS Prevention](#).

Anti-Spoofing Mechanism

Spoofed packets contain an IP address in the header that is not the actual IP address of the originating device. Spoofed attacks aim to saturate the target site links and the target site server resources or zone. The source IP addresses of the spoofed packets can be random, or have specific, focused addresses.

Spoofed attacks can be generated easily in a high volume, even from a single device because they cannot be stopped using access lists (ACLs) or filters. The reason is that the attacker can continuously change the source IP address of the packets.

To overcome spoofed attacks, the GSS uses an anti-spoofing mechanism called Redirect to TCP. This mechanism is used for DNS queries. It is based on forcing the client to resend its query using TCP. The D-proxy sends a UDP request, and the GSS responds with TC or truncated bit. If the D-proxy replies using TCP and

the TCP handshake is successful, then the GSS sends the TCP reply and considers the D-proxy to be valid for one hour. DDoS allows only traffic from such D-proxies to pass to the selector or the Cisco Network Registrar (CNR).

GSS provides anti-spoofing for all request packets (identified by `qrbit=0`), with the exception of TSIG, DDNS (`opcode=5`) and DNS notify (`opcode=4`) requests.

Nonspoofed D-proxies stay trusted for one hour. This nonspoofed timeout is not configurable.

If an anti-spoofing check fails (there is no response to the TCP connection), the D-proxy is blacklisted for one minute. This spoofed timeout is not configurable.

You may disable anti-spoofing for a particular D-proxy if that D-proxy does not support the option to respond using TCP. You can manually configure a D-proxy as either trusted or spoofed. If you configure a D-proxy as trusted, the GSS does not perform the anti-spoofing test for that IP address. If you configure a D-proxy as spoofed, the GSS drops all requests from that IP address.

**Note**

You may also disable the anti-spoofing mechanism on the GSS by using the **disable-as** command. This command should be used only when the D-proxies are unable to respond using TCP. We do not recommend that you disable the anti-spoofing mechanism.

See [Chapter 10, Configuring DDoS Prevention](#) for specific instructions about enabling DDoS and configuring filters, rate limits, and anti-spoofing mechanisms.

GSS Network Deployment

A typical GSS deployment may contain a maximum of 16 GSS devices deployed on a corporate intranet or the Internet. At least one GSS must be configured as a primary GSSM. Optionally, a second GSS can be configured as a standby GSSM. The primary GSSM monitors the other GSS devices on the network and offers features for managing and monitoring request routing services using CLI commands or a GUI accessible through secure HTTP. Only one GSSM can be active at any time, with the second GSSM serving as a standby, or backup device.

The GSSM functionality is embedded on each GSS, and any GSS device can be configured to act as a primary GSSM or a standby GSSM.

You can configure additional GSS devices on the GSS network to respond to DNS requests and transmit periodic keepalives to provide resource state information about devices. The GSS devices do not perform primary GSSM network management tasks.

This section describes a typical network deployment of the GSS and contains the following topics:

- [Locating GSS Devices](#)
- [Locating GSS Devices Behind Firewalls](#)
- [Communication Between GSS Nodes](#)
- [Deployment Within Data Centers](#)

Locating GSS Devices

Although your organization determines where your GSS devices are deployed in your network, you should follow these guidelines when deploying these devices.

Because the GSS serves as the authoritative name server for one or more domains, each GSS must be publicly or privately addressable on your enterprise network to allow the D-proxy clients requesting content to find the GSSs assigned to handle DNS requests.

Options are available for delegating responsibility for your domain to your GSS devices, depending on traffic patterns to and from your domain. For example, given a network containing five GSS devices, you might choose to modify your parent domain DNS servers so that all traffic sent to your domain is directed to your GSS network. You may also choose to have a subset of your traffic delegated to one or more of your GSSs, with other devices handling other segments of your traffic.

See [Chapter 7, Building and Modifying DNS Rules](#) for information about modifying your network DNS configuration to accommodate the addition of GSS devices to your network.

Locating GSS Devices Behind Firewalls

Deploying a firewall can prevent unauthorized access to your GSS network and eliminate common denial of service (DoS) attacks on your GSS devices. In addition to being deployed behind your corporate firewall, the GSS packet-filtering features can enable GSS administrators to permit and deny traffic to any GSS device.

When positioning your GSS behind a firewall or enabling packet filtering on the GSS itself, you must properly configure each device (the firewall and the GSS) to allow valid network traffic to reach the GSS device on specific ports. In addition to requiring HTTPS traffic to access the primary GSS graphical user interface, you may want to configure your GSSs to allow FTP, Telnet, and SSH access through certain ports. In addition, GSSs must be able to communicate their status to and receive configuration information from the GSSM. Also, primary and standby GSSMs must be able to communicate and synchronize with one another. Finally, if global DNS sticky is enabled on the GSS network, all GSSs in the sticky mesh must be able to communicate with each other to share the sticky database.

See the *Cisco Global Site Selector Administration Guide* for information about access lists to limit incoming traffic. See the “[Deploying GSS Devices Behind Firewalls](#)” section for information on which ports must be enabled and left open for the GSS to function properly.

Communication Between GSS Nodes

All GSS devices, including the primary GSSM and standby GSSM, respond to DNS queries and perform keepalives to provide global server load-balancing. Additionally, the primary GSSM acts as the central management device and hosts the embedded GSS database that contains shared configuration information, such as DNS rules, for each GSS that it controls. Use the primary GSSM to make configuration changes, which are automatically communicated to each registered GSS device that the primary GSSM manages.

The standby GSSM performs GSLB functions for the GSS network. The standby GSSM can act as the interim primary GSSM for the GSS network if the designated primary GSSM suddenly goes offline or becomes unavailable to communicate with other GSS devices. If the primary GSS goes offline, the GSS network continues to function and does not impact global server load balancing.

The GSS performs routing of DNS queries based on the DNS rules and conditions created from the primary GSSM. Each GSS device on the network delegates authority to the parent domain GSS DNS server that serves the DNS requests.

Each GSS is known to and synchronized with the primary GSSM. Unless global DNS sticky is enabled, individual GSSs do not report their presence or status to one another. If a GSS unexpectedly goes offline, the other GSSs on the network that are responsible for the same resources remain unaffected.

With both a primary and a standby GSSM deployed on your GSS network, device configuration information and DNS rules are automatically synchronized between the primary GSSM and a data store maintained on the standby GSSM.

Synchronization occurs automatically between the two devices whenever the GSS network configuration changes. Updates are packaged and sent to the standby GSSM using a secure connection between the two devices.

See the *Cisco Global Site Selector Administration Guide* for instructions on enabling each GSS device in the GSS network and for details about changing the GSSM role in the GSS network.

Deployment Within Data Centers

A typical GSS network consists of multiple content sites, such as data centers and server farms. Access to a data center or server farm is managed by one or more SLBs, such as the Cisco CSS or Cisco CSM. One or more virtual IP addresses (VIPs) represent each SLB. Each VIP acts as the publicly addressable front end of the data center. Behind each SLB are transaction servers, database servers, and mirrored origin servers offering a wide variety of content, from websites to applications.

The GSS communicates directly with the SLBs representing each data center by collecting statistics on availability and load for each SLB and VIP. The GSS uses the data to direct requests to the most optimum data centers and the most available resources within each data center.

In addition to SLBs, a typical data center deployment may also contain DNS name servers that are not managed by the GSS. These DNS name servers can resolve requests through name server forwarding that the GSS is unable to resolve.

GSS Network Management

Management of your GSS network is divided into two types:

- [CLI-Based GSS Management](#)
- [GUI-Based Primary GSSM Management](#)

Certain GSS network management tasks require that you use the CLI (initial device setup, sticky and proximity group configuration, for example). Other tasks require that you use the GUI (User Views and Roles, for example). In most cases, you have the option of using either the CLI or the GUI at the primary GSSM to perform GSLB configuration and monitoring.

Choosing when to use the CLI and when to use the GUI are also a matter of personal or organizational choice. Additionally, you can create your GSLB configuration using one method and then modify it using the alternate method.

This configuration guide describes how to use the CLI to perform global server load balancing. In cases where you must use the GUI to perform a particular task (configuring DNS rule filters, for example), the task is listed and a reference to the appropriate chapter in the *Global Site Selector GUI-Based Global Load-Balancing Configuration Guide* is provided.

CLI-Based GSS Management

You can use the CLI to configure the following installation, management, and global server load-balancing tasks for your GSS:

- Initial setup and configuration of GSS and GSSM (primary and standby) devices
- Software upgrades and downgrades on GSSs and GSSMs
- Database backups, configuration backups, and database restore operations
- Global server load balancing configuration and DNS request handling by creating DNS rules and monitoring keepalives at the primary GSSM

In addition, you can use the CLI for the following network configuration tasks:

- Network address and hostname configuration
- Network interface configuration

- Access control for your GSS devices, including IP filtering and traffic segmentation

You can also use the CLI for local status monitoring and logging for each GSS device.

See the *Cisco Global Site Selector Command Reference* for an alphabetical list of all GSS CLI commands including syntax, options, and related commands.

GUI-Based Primary GSSM Management

The primary GSSM offers a single, centralized graphical user interface (GUI) for monitoring and administering your entire GSS network. You can use the primary GSSM GUI to perform the following tasks:

- Configure DNS request handling and global server load balancing by creating DNS rules and monitoring keepalives
- Activate GSSs that are configured on the GSS network
- Monitor GSS network resources
- Monitor request routing and GSS statistics

For more information about the GUI, see the *Global Site Selector GUI-Based Global Load-Balancing Configuration Guide*.

Global Server Load-Balancing Summary

After you create your GSSM (primary and standby) and GSS devices and configure them to connect to your network, you are ready to begin configuring request routing and global server load balancing for your GSS network. See the *Cisco Global Site Selector Getting Started Guide* for procedures on getting your GSSM (primary and standby) and GSS devices set up, configured, and ready to perform global server load balancing.

Use CLI commands or the GUI on the primary GSSM to configure global server load balancing for your GSS network. You configure keepalives to monitor the health of SLBs and servers on your network, and you create and manage DNS rules and the associated global server load-balancing configuration to process incoming DNS requests.

To configure your GSS devices and resources from the primary GSSM for global server load balancing, perform the following steps:

1. Create regions, locations, and owners—Optional. Use these groupings to organize your GSS network resources by customer account, physical location, owner, or other organizing principle. See [Chapter 2, Configuring Resources](#), for details.
2. Create one or more source address lists—Optional. Use these lists of IP addresses to identify the name servers (D-proxy) that forward requests for the specified domains. The default source address list is Anywhere to match any incoming DNS request to the domains. See [Chapter 3, Configuring Source Address Lists](#), for details.
3. Create one or more domain lists—Establish lists of Internet domains, possibly using wildcards, that are managed by the GSS and queried by users. See [Chapter 4, Configuring Domain Lists](#), for details.
4. Modify the default global keepalive settings or create any shared keepalives—Optional. The GSS regularly polls to monitor the online status of one or more GSS resources linked to the keepalive. Shared keepalives are required for any answer that uses the KAL-AP keepalive type. See [Chapter 5, Configuring Keepalives](#), for details.
5. Create one or more answers and answer groups—Answers are resources that match requests to domains. Answer groups are collections of resources that balance requests for content. See [Chapter 6, Configuring Answers and Answer Groups](#), for details.
6. Build the DNS rules that will control global server load balancing on your GSS network. See [Chapter 7, Building and Modifying DNS Rules](#), for details.
7. If you plan to use DNS sticky for your global server load balancing, configure local or global DNS sticky for GSS devices in your network —Stickiness enables the GSS to remember the DNS response returned for a client D-proxy and to later return that answer when the client makes the same request. See [Chapter 8, Configuring DNS Sticky](#), for details.
8. If you plan to use network proximity for your global server load balancing, configure proximity for GSS devices in your network—Proximity determines the best (most proximate) resource for handling global load-balancing requests. See [Chapter 9, Configuring Network Proximity](#), for details.

9. If you plan to use the GSLB configuration file functionality, create, modify, and execute GSLB configuration files to automate the global server load-balancing process for your network. See [Chapter 11, Creating and Playing GSLB Configuration Files](#), for details.

Where to Go Next

[Chapter 2, Configuring Resources](#) describes how to organize resources on your GSS network as locations, regions, and owners.

