



CHAPTER 8

Configuring DNS Sticky

This chapter describes how to configure a GSS to support Domain Name System (DNS) stickiness to answer requests received from client D-proxies. The GSS supports DNS sticky both locally and globally between GSS network peers.

This chapter contains the following major sections:

- [DNS Sticky Overview](#)
- [DNS Sticky Quick Start Guide](#)
- [Synchronizing the GSS System Clock with an NTP Server](#)
- [Configuring Sticky Using the Primary GSSM CLI](#)
- [Disabling DNS Sticky Locally on a GSS for Troubleshooting](#)

Each GSS supports a comprehensive set of **show** CLI commands to display sticky application mesh statistics for the GSS device. In addition, the primary GSSM GUI displays sticky statistics for the GSS network. See [Chapter 13, Displaying GSS Global Server Load-Balancing Statistics](#), for details about viewing sticky statistics.

DNS Sticky Overview

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 Virtual IP address (VIP) continues to be available.

For many users browsing a site, being redirected to a new site is transparent. However, customers performing e-commerce or other transactions may experience a break in the connection when redirected, which results in a loss of the e-commerce transaction. Having DNS sticky enabled on a GSS helps to ensure 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 may impose a connection limit of 30 minutes before requiring a DNS re-resolution. This time period may not be long enough for a client to complete an e-commerce transaction. A new DNS resolution can then cause the client to connect to a server that is different from the original server, disrupting the transaction. DNS sticky helps to ensure that a client completes a transaction if a DNS re-resolution occurs.

This section contains the following topics on DNS sticky in the GSS:

- [Local DNS Sticky](#)
- [Sticky Database](#)
- [Global DNS Sticky](#)

Local DNS Sticky

With local DNS sticky, each GSS device attempts to ensure that subsequent client D-proxy requests to the same domain name from 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.

Each GSS dynamically builds and maintains a local sticky database that is based on the answers that the GSS sends to the requesting client D-proxies. If a subsequent request comes from the same client D-proxy, and the answer in the database is valid, the GSS returns the cached answer to the client D-proxy.

You configure the GSS to perform sticky load-balancing operations by configuring options on DNS rules and balance clauses. You identify the sticky method used by the DNS rule by matching a hosted domain or matching a hosted domain list. When sticking on a domain, the GSS provides the same sticky answer

to all requests from a client D-proxy for that domain. When sticking on a domain list, the GSS provides the same sticky answer to all requests from a client D-proxy for all domains in that domain list.

Before returning a sticky answer to a client, the GSS verifies the keepalive status. If the resource is:

- Available (online state), the GSS uses this answer for the DNS response sent back to the D-proxy.
- Available (online state) but the VIP corresponding to the answer is overloaded, the GSS continues to use this answer for the DNS response sent back to the D-proxy. Sticky always takes precedence over an exceeded load threshold in the associated DNS rule.
- Unavailable (offline state), the GSS selects a new answer and inserts this answer into the sticky database, replacing the previous answer.

Sticky Database

The sticky database provides the core intelligence for all DNS sticky-based decisions made by a GSS, on a local or global level. The GSS collects requests from the client D-proxies and stores these requests in memory as the sticky database. Requests may be identified by the IP address of the client D-proxy or a database ID representing a list of D-proxy IP addresses (configured as a sticky group, see the [“Creating Sticky Groups”](#) section). The D-proxy IP address may also be some form of a sticky global netmask if the global subnet mask is set to a value other than the default of 255.255.255.255.

The sticky database stores the answer to each request that the DNS rule matches, which may be for a single domain (including wildcard expressions) or a configured list of domains. These components comprise each sticky database key that the GSS uses for the lookup, storage, and persistence of stickiness for DNS responses.

The primary GSSM supports the creation of sticky groups that allow you to configure multiple blocks of D-proxy IP addresses that each GSS device stores in its sticky database as a single entry. Instead of multiple sticky database entries, the GSS uses only one entry in the sticky database for multiple D-proxies. The GSS treats all D-proxies in a sticky group as a single D-proxy.

All entries in the sticky database age out based on a user-specified global sticky inactivity timeout value. The sticky inactivity timeout value identifies the time period that an answer remains valid in the sticky database. Every time the GSS returns an answer to the requesting client, the GSS resets the expiration time of the answer to this value. When the sticky inactivity timeout value elapses without the client requesting the answer, the GSS identifies the answer as invalid and purges it from the sticky database. You can specify a global sticky inactivity timeout default value for the GSS or modify the inactivity timeout value for each DNS rule.

**Note**

The sticky inactivity timeout is accurate to within 5 minutes of the specified value. Each entry persists in the sticky database for the configured sticky inactivity timeout value and may remain in the sticky database for no longer than 5 minutes past the specified value.

Upon receiving a DNS request, the GSS searches the sticky database for a matched entry based on the combination of D-proxy IP address (or sticky group ID) and requested hosted domain or domain list information in the request. If the GSS finds a matched entry (a hit), the GSS returns the original DNS answer to the requesting D-proxy and the GSS resets the user-configured sticky inactivity timeout to its starting value. If the GSS does not find a matched entry (a miss), the GSS does not return a sticky answer but, instead, performs normal load balancing for the request to locate a new answer and add the new entry into the sticky database.

The GSS supports a maximum of 400,000 entries in the sticky database. When the total number of entries in the sticky database reaches 400,000, the GSS automatically removes entries from the database based on the lowest percentage of time remaining.

Global DNS Sticky

This section provides an overview of the global DNS sticky function and the behavior of GSS devices operating in a peer mesh. It contains the following topics:

- [GSS Sticky Peer Mesh](#)
- [Sticky Mesh Conflict Resolution](#)

- [Communicating in the Sticky Peer Mesh](#)

GSS Sticky Peer Mesh

With global DNS sticky enabled, each GSS device in the network shares sticky database 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. Subsequent client D-proxy requests to the same domain name to any GSS in the network cause the client to be “stuck.”

When one GSS device in the mesh receives a query from a client for a hosted domain or domain list, global sticky enables each GSS in the network to make a best effort attempt to return the same answer to the requesting client. This action is performed regardless of which GSS in the network is selected to answer the first and subsequent requests. The individual GSS devices work together to maintain a global sticky database across the network.

Each GSS in the peer mesh receives updates from the other peers and sends local changes to its remote peers. The GSS devices share the following information with the other GSS devices in the peer mesh:

- The sticky database lookups performed
- The persistent answers provided in the response
- The related time stamp and sticky inactivity timeout details

Each GSS sends updates to its remote GSS peers when any of the following events occur:

- A D-proxy request arrives at a GSS with no previous database entry. The GSS returns a new answer to the requesting client and enters that answer in its local database.
- A GSS returns a previous answer to the requesting client. The GSS resets the expiration time for the answer to its original sticky inactivity timeout value.
- The GSS finds an existing answer in the sticky database but a keepalive determines that the answer is nonresponsive (offline). In this case, the GSS uses the DNS rule to choose a new answer, overriding the previous answer in the sticky database, and communicates this answer to all peers.

- You use the **sticky database delete** CLI command to delete one or more entries from the sticky database.

A GSS does not send information to its peers when it purges an answer from the sticky database when reaching the normal sticky inactivity timeout or a sticky database overflow. Each GSS in the mesh is expected to perform this task independently.

When a local GSS node receives information from one of its peers in the network, that GSS performs a lookup of each received data entry in its local sticky database. Based on the results of the lookup, the GSS performs one of the following actions:

- If the GSS does not find the entry in its sticky database, the GSS adds the answer to its local sticky database.
- If the GSS finds the same entry in its sticky database, the GSS resets the expiration time for the answer to the initial sticky inactivity timeout value.

The GSS supports encryption of all inter-GSS communications to maintain the integrity of the sticky database information transferred among the mesh peers. Each GSS uses the Message Digest 5 (MD5)-based hashing method to encrypt the application data sent throughout the mesh.

To authenticate communication between GSS devices in the mesh to prevent unauthorized device access, you can specify a secret string that is used by all GSS devices in the mesh. The secret string provides a key for authentication between GSS devices as well as for encryption (if enabled). Each local GSS uses the Challenge Handshake Authentication Protocol (CHAP) method to establish a connection with a remote peer.

Sticky Mesh Conflict Resolution

In some instances, two or more GSS devices in the mesh may answer the same sticky request at the same time. When GSS devices communicate their updates to each peer, the recipient detects a conflict. Conflicts are resolved in the peer network by each GSS that keeps the record of the entry with the greatest expiration time stamp, (that is, the newest record). If the conflicting entries have identical time stamps, the GSS uses the entry that contains the most recently configured answer based on the configuration ID.

Conflicts are far more likely to occur when multiple requests are grouped by domain list, or when you group D-proxy clients by a sticky mask or by sticky group. For example, if you configure a DNS rule for domains A and B, one client

may request GSS 1 for domain A, while a second client may make a request for domain B. If the GSS receives both requests at the same time, the two clients may receive different answers.

You can reduce global sticky mesh conflicts as follows:

- Configure sticky DNS rules for one domain only. Avoid using the **domain-list** option for the sticky method unless absolutely necessary.
- Avoid using domain wildcards. Wildcard domains pose the same issue as domain lists.
- Avoid using low DNS **ttl** values in a sticky balance clause. Setting the **ttl** value of each sticky balance clause to a high value allows the sticky database to synchronize answers before the client D-proxy attempts to re-resolve the answer.

Communicating in the Sticky Peer Mesh

You can successfully pass packets between GSS peers in the sticky mesh by ensuring that the following requirements are met:

- Synchronize the system clock of each GSS device in the mesh with a Network Time Protocol (NTP) server. If a GSS system clock is out of synchronization with the other GSS peers (by a value greater than 3 minutes), that GSS ignores update messages from other GSS devices until you synchronize its system clock. See the [“Synchronizing the GSS System Clock with an NTP Server”](#) section for details.
- Each GSS in the peer mesh has the same global subnet mask values. A GSS will drop all global sticky messages received from a GSS with a different subnet mask. A difference in global sticky masks on a peer would occur only if a configuration change were made on the primary GSSM and the peer did not receive the change due to a network failure. See the [“Configuring DNS Sticky”](#) section for details.
- Each GSS in the peer mesh has the same GSS software version.

If these conditions are not met, a GSS cannot properly receive or send packets with the other GSS peers in the sticky mesh.

A GSS leaves and rejoins the global sticky mesh when you perform one of the following actions:

- Enter the **gss restart** command to restart the GSS software on the local GSS node.
- Enter the **sticky stop** and **sticky start** command sequence on the local GSS node.
- Enter the **reload** command to perform a cold restart of the local GSS node.
- Enter the **enable global** command in sticky properties configuration mode.

Upon reentry into the mesh, the GSS attempts to load the sticky database from a peer GSS. The GSS uses the shortest round-trip time (RTT) to prioritize from which peer to request the database update. If a GSS peer is unavailable, the GSS locally restores the sticky database from the last available periodic database dump file. The GSS restores the sticky database from the database dump file any time it rejoins the mesh and cannot retrieve a database from a GSS peer in the mesh. When the load is complete, the local database on the GSS device contains a full version of the sticky database.

If you want the local GSS node to attempt synchronization with a specific GSS peer upon reentry into the sticky mesh, you can identify a favored GSS peer for that GSS device. By identifying a favored GSS peer, you can also reduce network issues with peer synchronization, which typically generate a burst of network traffic. In this case, direct network traffic to a peer other than the GSS identified as being the closest (with the shortest round-trip time).

When you identify a favored peer, the local GSS node, upon reentry into the mesh, always attempts to synchronize its sticky database entries with the favored GSS peer. If the GSS favored peer is unavailable, the local GSS node queries the remaining mesh peers to find the closest up-to-date sticky database.

Network connectivity issues, GSS devices leaving and rejoining the mesh, and GSS device restarts have a minor impact on the synchronization of the sticky database. Sticky database entries always reconverge based on their usage and the user-configurable sticky inactivity timeout values.

Logging in to the CLI and Enabling Privileged EXEC Mode

**Note**

To log in and enable privileged EXEC mode in the GSS, you must be a configured user with admin privileges. See the *Cisco Global Site Selector Administration Guide* for information on creating and managing user accounts.

To log in to the primary GSSM and enable privileged EXEC mode at the CLI, perform the following steps:

1. If you are remotely logging in to the primary GSSM through Telnet or SSH, enter the hostname or IP address of the GSSM to access the CLI.

If you are using a direct serial connection between your terminal and the GSSM, use a terminal emulation program to access the CLI. For details about making a direct connection to the GSS device using a dedicated terminal and about establishing a remote connection using SSH or Telnet, see the *Cisco Global Site Selector Getting Started Guide*.

2. Specify your GSS administrative username and password to log in to the GSSM. The CLI prompt appears.

```
gssm1.example.com>
```

3. At the CLI prompt, enable privileged EXEC mode as follows:

```
gssm1.example.com> enable
gssm1.example.com#
```

If you are accessing the GSS remotely using Telnet or SSH, the CLI prompts you for the enable password. The default password is default. For more information about the enable password and configuring a new password, see the *Cisco Global Site Selector Getting Started Guide*.

The prompt changes from the user-level EXEC right angle bracket (>) prompt to the privileged-level EXEC pound sign (#).

DNS Sticky Quick Start Guide

Table 8-1 provides a quick overview of the steps required to configure the GSS for DNS sticky operation, both local and global DNS sticky. Each step includes the primary GSSM command required to complete the task. For the procedures to configure the GSS for DNS sticky, see the sections that follow the table.

Table 8-1 *DNS Sticky Configuration Quick Start*

Task and Command Example

1. If you are using global sticky with multiple GSS devices, log in to the CLI of each GSS in the mesh, enable privileged EXEC mode, and synchronize its system clock with an NTP server.

For example, enter:

```
gssm1.example.com> enable
gssm1.example.com# config
gssm1.example.com(config)# ntp-server 172.16.1.2 172.16.1.3
gssm1.example.com(config)# ntp enable
```

2. Enter the global server load-balancing configuration mode.

For example, enter:

```
gssm1.example.com(config)# gslb
gssm1.example.com(config-gslb)#
```

3. Use the **sticky-properties** command to enter the sticky properties configuration mode.

For example, enter:

```
gssm1.example.com(config-gslb)# sticky-propertie
gssm1.example.com(config-gslb-stkyprop)#
```

4. From the sticky properties configuration mode, enable sticky and specify a mode (global or local).

For example, to enable global DNS sticky for the GSS network, enter:

```
gssm1.example.com(config-gslb-stkyprop)# enable global
gssm1.example.com(config-gslb-stkyprop)# exit
gssm1.example.com(config-gslb)#
```

Table 8-1 *DNS Sticky Configuration Quick Start (continued)***Task and Command Example**

5. Configure default DNS sticky configuration and inter-GSS global sticky mesh settings in sticky properties configuration mode. You can use the following keywords and arguments:
 - **encryption**—Enables or disables the encryption of data transmitted by GSS devices in the mesh. Valid options are as follows:
 - **enable**—Enables the encryption of data transferred between GSS peers in the mesh.
 - **key name**—Provides a secret string key for authentication between GSS devices as well as for encryption (if enabled).
 - **mask netmask**—Specifies a global subnet mask that the GSS uses to uniformly group contiguous D-proxy addresses as an attempt to increase the number of clients that the sticky database can support.
 - **timeout seconds**—Specifies the maximum time period that an unused answer remains valid in the sticky database.
 - **favored-peer**—If you want a local GSS device to attempt synchronization with a specific favored GSS peer upon reentry into the sticky mesh, specify a GSS device and a favored GSS peer (a different GSS device) combination for each local GSS device in the mesh.

See the “[Configuring DNS Sticky](#)” section for a complete description of these settings.

For example, enter:

```
gssm1.example.com(config-gslb-stkyprop)# enable global
gssm1.example.com(config-gslb-stkyprop)# encryption key SECRETKEY
gssm1.example.com(config-gslb-stkyprop)# timeout 120
gssm1.example.com(config-gslb-stkyprop)# exit
gssm1.example.com(config-gslb)#
```

6. Develop your DNS rule using the **dns rule** command.

For example, enter:

```
gssm1.example.com(config)# gslb
gssm1.example.com(config-gslb)# dns rule drule02 owner
WEB-SERVICES source-address-list WEB-GLOBAL-LISTS domain-list
E-COMMERCE query A
gssm1.example.com(config-gslb-rule[rule-name])#
```

Table 8-1 *DNS Sticky Configuration Quick Start (continued)***Task and Command Example**

7. Define how the GSS supports DNS stickiness in a DNS rule (by domain or domain-list) by entering the **sticky method** command as follows:

```
gssm1.example.com(config-gslb-rule[rule-name])# sticky method
domain
gssm1.example.com(config-gslb-rule[rule-name])#
```

8. If you want to override the global timeout value set for the DNS rule, specify a new **timeout** value.

For example, enter:

```
gssm1.example.com(config-gslb-rule[rule-name])# sticky method
domain timeout 250
gssm1.example.com(config-gslb-rule[rule-name])#
```

9. Configure Balance Clause 1 for the DNS rule by entering the **clause** command and use the **sticky enable** option to enable sticky for the DNS rule.

For example, enter:

```
gssm1.example.com(config-gslb-rule[rule-name])# clause 1
vip-group ANSGRP-VIP-01 sticky enable
gssm1.example.com(config-gslb-rule[rule-name])#
```

Table 8-1 *DNS Sticky Configuration Quick Start (continued)*

Task and Command Example

10. (Optional) Configure other **clause** command settings as appropriate. You can use the following keywords and arguments:
- **count number**—Specifies the number of address records (A-records) that you want the GSS to return for requests that match the DNS rule.
 - **ttl number**—Specifies the duration of time in seconds that the requesting DNS proxy caches the response sent from the GSS and considers it to be a valid answer.
 - **method**—(Optional) Specifies the method type for each balance clause. Method types are as follows:
 - **round-robin**—The GSS cycles through the list of answers that are available as requests are received. This is the default.
 - **ordered**—The GSS selects an answer from the list based on precedence.
 - **least-loaded**—The GSS selects an answer based on the load reported by each VIP in the answer group.
 - **weighted-round-robin**—The GSS cycles through the list of answers that are available as requests are received but sends requests to favored answers in a ratio determined by the weight value assigned to that resource.
 - **hashed**—The GSS selects the answer based on a unique value created from information stored in the request. Valid hashed method types are as follows: **source-address**, **domain-name**, or **both**.

See the [“Adding Sticky to a DNS Rule that uses VIP-Type Answer Groups”](#) section for a complete description of these settings.

For example, enter:

```
gssm1.example.com(config-gslb-rule[rule-name])# clause 1  
vip-group ANSGRP-VIP-01 sticky enable ttl 30 method ordered  
gssm1.example.com(config-gslb-rule[rule-name])#
```

Table 8-1 DNS Sticky Configuration Quick Start (continued)**Task and Command Example**

11. Using the **clause** command, repeat Steps 9 and 10 for Balance Clause 2.

For example, enter:

```
gssm1.example.com(config-gslb-rule[rule-name])# clause 2
vip-group ANSGRP-VIP-02 sticky enable ttl 60 method least-loaded
gssm1.example.com(config-gslb-rule[rule-name])#
```

Note The GSS prevents you from enabling sticky on Balance Clause 2 if you do not first enable sticky on Balance Clause 1. This restriction is also true if you attempt to enable sticky on Balance Clause 3 without first configuring sticky on Balance Clause 2.

12. Reenter the **clause** command for Balance Clause 3, and then repeat Steps 9 and 10.

13. (Optional) Group multiple D-proxy IP addresses as a single entry in the sticky database, exit from the rule configuration mode, and then use the **sticky group** command in global server load-balancing mode.

For example, enter:

```
gssm1.example.com(config-gslb-rule[rule-name])# exit
gssm1.example.com(config-gslb)# sticky group StickyGroup1 ip
192.168.3.0 netmask 255.255.255.0
```

Synchronizing the GSS System Clock with an NTP Server

If you are using global sticky in your GSS network, you must synchronize the clocks of all GSS devices in the mesh to enable a GSS to communicate with the other GSS devices in the peer mesh. If a GSS system clock is out of synchronization with the other GSS peers (by a value greater than 3 minutes), that GSS will ignore update messages from other GSS devices until you synchronize its system clock.

**Note**

We strongly recommend that you synchronize the system clock of each GSS in the mesh with a Network Time Protocol (NTP) server. NTP is a protocol designed to synchronize the clocks of computers over a network with a dedicated time server.

You must specify the NTP servers for each GSS device that operates in the global mesh before you enable DNS sticky for those devices from the primary GSSM. This sequence ensures that the clocks of each GSS device are synchronized before they join the global sticky peer mesh.

**Note**

For details on logging in to a GSS device and enabling privileged EXEC mode at the CLI, see the [“Logging in to the CLI and Enabling Privileged EXEC Mode”](#) section.

You can specify one or more NTP servers for GSS clock synchronization by using the **ntp-server** global configuration mode command.

The syntax of this CLI command is as follows:

```
ntp-server ip_or_host
```

The *ip_or_host* argument specifies the IP address or hostname of the NTP time server in your network that provides the clock synchronization. You can specify a maximum of four IP addresses or hostnames. Enter the IP address in dotted-decimal notation (for example, 172.16.1.2) or a mnemonic hostname (for example, myhost.mydomain.com).

You can enable the NTP service by using the **ntp enable** global configuration mode command.

The syntax of this CLI command is as follows:

```
ntp enable
```

To specify the IP addresses of two NTP time servers for a GSS device and enable the NTP service, enter:

```
gssm1.example.com> enable  
gssm1.example.com# config  
gssm1.example.com(config)# ntp-server 172.16.1.2 172.16.1.3  
gssm1.example.com(config)# ntp enable
```

Configuring Sticky Using the Primary GSSM CLI

This section describes how to configure GSS devices for DNS sticky operation, add stickiness to a DNS rule on the primary GSSM, and manage the sticky database. It contains the following topics:

- [Configuring DNS Sticky](#)
- [Enabling Sticky in a DNS Rule](#)
- [Creating Sticky Groups](#)
- [Deleting Entries from the Sticky Database](#)
- [Dumping Sticky Database Entries](#)
- [Running a Periodic Sticky Database Backup](#)
- [Loading Sticky Database Entries](#)

Configuring DNS Sticky

The GSS includes a set of DNS sticky settings that function as the default values used by the GSS network when you enable sticky in a DNS rule. You can configure sticky only in a DNS rule that uses a VIP-type answer group. In addition, sticky is active for a DNS rule only when the following conditions exist:

- Sticky is enabled for either global or local use. In the CLI, enter the **enable global** or **enable local** command.
- A sticky method option (domain or domain list) is selected. In the CLI, enter the **sticky method domain** or **sticky method domain list** command.
- Sticky is enabled within a balance clause for the DNS rule. In the CLI, enter the **sticky enable** command.

From global server load-balancing configuration mode, use the **sticky-properties** command to enter the sticky properties configuration mode. In the sticky properties configuration mode, you can enter commands to enable sticky and modify the DNS sticky settings for the GSS network. Sticky settings are applied as soon as you exit from the sticky properties configuration mode or enter a new mode.

To enable sticky and configure the sticky settings from the sticky properties configuration mode, specify one or more of the following commands:

- **enable**—Enables DNS sticky for the global or local level. Valid options are as follows:
 - **global**—Enables global DNS sticky for each active GSS device across the entire GSS mesh. With global DNS sticky, all local sticky features are in operation and each GSS device in your network shares answers between peer GSS devices in a peer mesh. The peer mesh attempts to ensure that if any GSS device in the mesh receives the same question, then the same answer is returned to the requesting client D-proxy.
 - **local**—Enables DNS sticky for each active GSS device on a local level only. Each GSS attempts to ensure that subsequent requests for the same domain name are “stuck” to the same location as the first request. Sticky database information is not shared between GSS devices in the GSS mesh.
- **encryption**—Enables or disables the encryption of data transmitted by GSS devices in the mesh. This command is valid only if the **global** command is enabled. The GSS support encryption of all inter-GSS communications to maintain the integrity of the sticky database information transferred among the mesh peers. This command is disabled by default (data is transmitted in clear text). Valid options are as follows:
 - **enable**—Enables the encryption of data transferred between GSS peers in the mesh. Each GSS uses the Message Digest 5 (MD5)-based hashing method to encrypt the application data sent throughout the mesh.
 - **key name**—(Optional) Provides a secret string key for authentication between GSS devices as well as for encryption (if enabled). Enter a unique alphanumeric name with a maximum of 31 characters. Names that include spaces must be entered in quotes (for example, “name 1”).
- **mask netmask**—Specifies a global subnet mask that the GSS uses to uniformly group contiguous D-proxy addresses in order to increase the number of clients that the sticky database can support. This mask is applied to the client source IP address before accessing the sticky database. Enter the subnet mask in dotted-decimal notation (for example, 255.255.255.0). The default global mask is 255.255.255.255.

When you define a DNS sticky group for incoming D-proxy addresses (see the “[Creating Sticky Groups](#)” section) and the incoming D-proxy address does not match any of the entries in a defined DNS sticky group, the GSS uses this global netmask value to calculate a grouped D-proxy network address.

- **timeout seconds**—Specifies the maximum time that an unused answer remains valid in the sticky database. This value defines the sticky database entry age-out process. Every time the GSS returns an answer to the requesting client D-proxy, the GSS resets the expiration time of the answer to this value. When the sticky timeout value elapses without the client again requesting the answer, the GSS identifies the answer as invalid and purges it from the sticky database. Enter a value from 15 to 10080 minutes (168 hours), specified in 5 minute intervals (15, 20, 25, 30, and up to 10080). The default value is 60 minutes.

You can also individually set the **timeout** value for each DNS rule. When you set a **timeout** value for a DNS rule, that value overrides the global **timeout** value.


Note

The sticky timeout is accurate to within 5 minutes of the specified value. Each entry will persist in the sticky database for the configured sticky timeout value and may remain in the sticky database for no longer than 5 minutes past the specified value.

- **favored-peer**—If you want a local GSS device to attempt synchronization with a specific GSS peer upon reentry into the sticky mesh, specify a favored peer for each local GSS device in the mesh. By specifying a favored GSS peer, you can also reduce network issues with peer synchronization, which typically generate a burst of network traffic. In this case, you can direct network traffic to a peer other than the GSS identified as being the closest (with the shortest round-trip time). This command is valid only if you enable the **global** option. Specify one of the following:
 - *GSS*—Name of the local GSS device that will be associated with a favored peer GSS device. The peer GSS device is the name of another GSS device that you specify in the *GSS-peer* variable.
 - *GSS-peer*—Name of the favored GSS peer that is to be associated with the GSS device name specified as the *GSS* variable.

Reenter the **favored-peer** command as many times as required to assign favored GSS peers to the GSS devices in the sticky mesh.

A GSS joins the mesh when any of the following actions occur:

- A reload.
- A power up.
- When you issue the **gss stop** and **gss start** command sequence.
- When you enter the **reload** command.
- When you enter the **sticky stop** and **sticky start** command sequence.
- When you enable the **global** option.

Upon reentry into the mesh, the local GSS device first attempts to synchronize its sticky database entries with its favored GSS peer. If the favored peer is unavailable, the GSS queries the remaining mesh peers to find the closest up-to-date sticky database (with the shortest round-trip time).

For example, assume there are four GSS devices in a mesh (*gss_1*, *gss_2*, *gss_3*, and *gss_4*), and both *gss_1* and *gss_2* are in the bootup process. You can direct local device *gss_1* to *gss_3* as its GSS peer, and direct local device *gss_2* to *gss_4* as its GSS peer. The identification of favored GSS peers in the mesh can prevent those GSS devices that are booting from waiting for another database request to complete before their database synchronization request can be serviced.

If you do not specify any favored GSS peers, the GSS uses the shortest round-trip time to prioritize which peers to request a database update.

- **mask** *netmask*—Specifies a global subnet mask that the GSS uses to uniformly group contiguous D-proxy addresses as an attempt to increase the number of clients that the sticky database can support. This mask is applied to the client source IP address before accessing the sticky database. Enter the subnet mask in dotted-decimal notation (for example, 255.255.255.0). The default global mask is 255.255.255.255.

When you define a DNS sticky group for incoming D-proxy addresses (see the “[Creating Sticky Groups](#)” section) and any incoming D-proxy address does not match any of the entries in a defined DNS sticky group, the GSS uses this global netmask value to calculate a grouped D-proxy network address.

- **timeout** *minutes*—Specifies the maximum time that an unused answer remains valid in the sticky database. This value defines the sticky database entry age-out process. Every time the GSS returns an answer to the requesting client D-proxy, the GSS resets the expiration time of the answer to this value. When the sticky timeout value elapses without the client again requesting the

answer, the GSS identifies the answer as invalid and purges it from the sticky database. Enter a value from 15 to 10080 minutes (168 hours), specified in 5-minute intervals (15, 20, 25, 30, and up to 10080). The default value is 60 minutes.

You can also individually set the **timeout** value for each DNS rule. When you set a **timeout** value for a DNS rule, that value overrides the global **timeout** value.



Note The sticky timeout is accurate to within 5 minutes of the specified value. Each entry will persist in the sticky database for the configured sticky timeout value and may remain in the sticky database for no longer than 5 minutes past the specified value.

For example, to enable global stickiness and specify encryption key and timeout values, enter:

```
gssm1.example.com(config-gslb)# sticky-properties
gssm1.example.com(config-gslb-stkyprop)# enable global
gssm1.example.com(config-gslb-stkyprop)# encryption key SECRETKEY
gssm1.example.com(config-gslb-stkyprop)# timeout 120
gssm1.example.com(config-gslb-stkyprop)# exit
gssm1.example.com(config-gslb)#
```

To reset a stickiness setting to its default, use the **no** form of the command as follows:

```
gssm1.example.com(config-gslb-stkyprop)# no timeout 120
gssm1.example.com(config-gslb-stkyprop)# no mask 255.255.255.0
gssm1.example.com(config-gslb-stkyprop)# exit
gssm1.example.com(config-gslb)#
```

Enabling Sticky in a DNS Rule

This section contains the following topics:

- [Sticky DNS Rule Overview](#)
- [Adding Sticky to a DNS Rule that uses VIP-Type Answer Groups](#)

Sticky DNS Rule Overview

After you enable DNS sticky and configure sticky settings, you add stickiness to a DNS rule using the **sticky method** and **clause** commands in rule configuration mode. The GSS supports DNS stickiness in a DNS rule on either a matching domain (**domain** option) or on a matching domain list (**domain-list** option). The domain and domain-list sticky methods instruct the GSS that all requests from a client D-proxy for a matching hosted domain or domain list are to be given the same answer for the duration of a user-configurable sticky time period.

Enabling sticky in a DNS rule clause causes the GSS to look up in the sticky database for a matched entry based on a combination of D-proxy IP address and requested domain information, and if the answer is found, to return the answer as the DNS response to the requesting D-proxy. If the answer is in the offline state, or the GSS does not find the answer, it evaluates the balance method clauses in the DNS rule to choose a new answer.

You can configure sticky individually for each balance clause in a DNS rule. However, the GSS prevents you from enabling sticky on Balance Clause 2 if you do not first enable sticky on Balance Clause 1. This restriction is also true if you attempt to enable sticky on Balance Clause 3 without first configuring sticky on Balance Clause 2.



Note

If you use DNS sticky and network proximity in your DNS rule, stickiness always takes precedence over proximity. When a valid sticky answer exists for a given DNS rule match, the GSS does not consider proximity when returning an answer to a client D-proxy.

Adding Sticky to a DNS Rule that uses VIP-Type Answer Groups

To add sticky to a DNS rule that uses VIP-type answer groups, perform the following steps:

1. If you have not already done so, enable local or global DNS sticky. See the “[Configuring DNS Sticky](#)” section for details.
2. Develop your DNS rule by using the **dns rule** command, as described in the “[Building DNS Rules](#)” section of [Chapter 7, Building and Modifying DNS Rules](#).

3. Define how the GSS supports DNS stickiness in a DNS rule by using the **sticky method** command with one of the following options:
 - **domain**—Enables DNS stickiness on a domain. For all requests from a single D-proxy, the GSS sends the same answer for a domain. For rules matching on a domain wildcard (for example, *.cisco.com), entries are stuck together using the global configuration ID assigned to the wildcard. The GSS does not attempt to distinguish the individual domains that match the wildcard.
 - **domain-list**—Enables DNS stickiness on a matching domain list. The GSS groups all domains in the domain list and treats them as a single hosted domain. The GSS treats wildcards in domain lists the same as non-wildcard domains.

**Note**

Sticky is disabled by default. When disabled, the GSS answers DNS requests for all domains and clients that pertain to the DNS rule, subject to DNS rule matching, without accessing the sticky database or sharing sticky database information between peers in the network.

4. Override the global timeout value set for a DNS rule by specifying a new **timeout** value. Enter the maximum time interval that can pass without the sticky database receiving a lookup request for an entry. Every time the GSS returns an answer to the requesting client D-proxy, the GSS resets the expiration time of the answer to this value. When the sticky timeout value elapses without the client again requesting the answer, the GSS identifies the answer as invalid and purges it from the sticky database. Enter a value from 15 to 10080 minutes, defined in 5-minute intervals (15, 20, 25, 30, and up to 10080).

For example, enter:

```
gssm1.example.com(config-gslb)# dns rule drule02
gssm1.example.com(config-gslb-rule[rule-name])# sticky method
domain timeout 250
gssm1.example.com(config-gslb-rule[rule-name])#
```

**Note**

The sticky **timeout** is accurate to within 5 minutes of the specified value. Each entry will persist in the sticky database for the configured sticky timeout value and may remain in the sticky database for no longer than 5 minutes past the specified value.

5. Configure Balance Clause 1 using the **clause number vip-group name** command in the rule configuration mode. The syntax of this command is as follows:

```
clause number vip-group name [count number | sticky {enable | disable}] |  
ttl number | method {round-robin | least-loaded | ordered |  
weighted-round-robin | hashed {domain-name | source-address | both}} ]]
```

The keywords and arguments for this command are as follows:

- **number**—Balance Clause 1, 2, or 3. You can specify a maximum of three balance clauses that use VIP-type answers.
- **vip-group name**—Specifies the name of a previously created VIP-type answer group.
- **count number**—(Optional) Specifies the number of address records (A-records) that you want the GSS to return for requests that match the DNS rule. The default is 1 record.
- **sticky**—(Optional) Specify **enable** to activate sticky for the clause. Specify **disable** (the default) to deactivate sticky for the clause. To specify **enable**, make sure that the **sticky method** option (see Step 3) is set to **domain** or **domain-list**.
- **ttl number**—(Optional) Specifies the time-to-live duration in seconds that the requesting DNS proxy caches the response sent from the GSS and considers it to be a valid answer. Valid entries are 0 to 604,800 seconds. The default is 20 seconds.
- **method**—(Optional) Specifies the method type for each of your balance clauses. Method types are as follows:
 - **round-robin**—The GSS cycles through the list of answers that are available as requests are received. This is the default.
 - **least-loaded**—The GSS selects an answer based on the load reported by each VIP in the answer group. The answer reporting the lightest load is chosen to respond to the request. The least-loaded option is available only for VIP-type answer groups that use a KAL-AP or Scripted keepalive.
 - **ordered**—The GSS selects an answer from the list based on precedence; answers with a lower order number are tried first, while answers further down the list are tried only if preceding answers are unavailable to respond to the request. The GSS supports numbering gaps in an ordered list.



Note For answers that have the same order number in an answer group, the GSS will use only the first answer that contains the number. We recommend that you specify a unique order number for each answer in an answer group.

- **weighted-round-robin**—The GSS cycles through the list of answers that are available as requests are received but sends requests to favored answers in a ratio determined by the weight value assigned to that resource.
 - **hashed**—The GSS selects the answer based on a unique value created from information stored in the request. The GSS supports two hashed balance methods. The GSS allows you to apply one or both hashed balance methods to the specified answer group. Enter one of the following:
 - **domain-name**—The GSS selects the answer based on a hash value created from the requested domain name.
 - **source-address**—The GSS selects the answer based on a hash value created from the source address of the request.
 - **both**—The GSS selects the answer based on both source-address and domain name.
6. Repeat the configuration process for Balance Clauses 2 and 3 by entering the **clause** command.



Note The GSS prevents you from enabling sticky on Balance Clause 2 if you do not first enable sticky on Balance Clause 1. This restriction also applies if you attempt to enable sticky on Balance Clause 3 without first configuring sticky on Balance Clause 2.

To set up Balance Clauses 1 and 2 with stickiness for the previously created DNS rule named `drule02`, enter:

```
gssm1.example.com(config-gslb)# dns rule drule02
gssm1.example.com(config-gslb-rule[rule-name])# clause 1 vip-group
ANSGRP-VIP-01 sticky enable method ordered
gssm1.example.com(config-gslb-rule[rule-name])# clause 2 vip-group
ANSGRP-VIP-02 sticky enable method least-loaded
gssm1.example.com(config-gslb-rule[rule-name])#
```

To delete a balance clause, use the **no** form of the **clause** command as follows:

```
gssm1.example.com(config-gslb-rule[rule-name])# no clause 2 vip-group
ANSGRP-VIP-02 sticky enable method least-loaded
gssm1.example.com(config-gslb-rule[rule-name])#
```

Creating Sticky Groups

The primary GSSM supports the creation of sticky groups. A sticky group allows you to configure multiple blocks of D-proxy IP addresses that each GSS device stores in its sticky database as a single entry. Instead of multiple sticky database entries, the GSS uses only one entry in the sticky database for multiple D-proxies. The GSS treats all D-proxies in a sticky group as a single D-proxy.

This section contains the following topics:

- [DNS Sticky Group Overview](#)
- [Creating a DNS Sticky Group](#)
- [Deleting a Sticky Group IP Address Block](#)
- [Deleting a Sticky Group](#)

DNS Sticky Group Overview

Create sticky groups from the primary GSSM CLI to obtain better scalability of your configuration and to allow easy sticky group creation through automated scripts. The primary GSSM supports a maximum of 800 sticky groups. Each sticky group contains one to 30 blocks of IP addresses and subnet masks (in dotted-decimal notation).

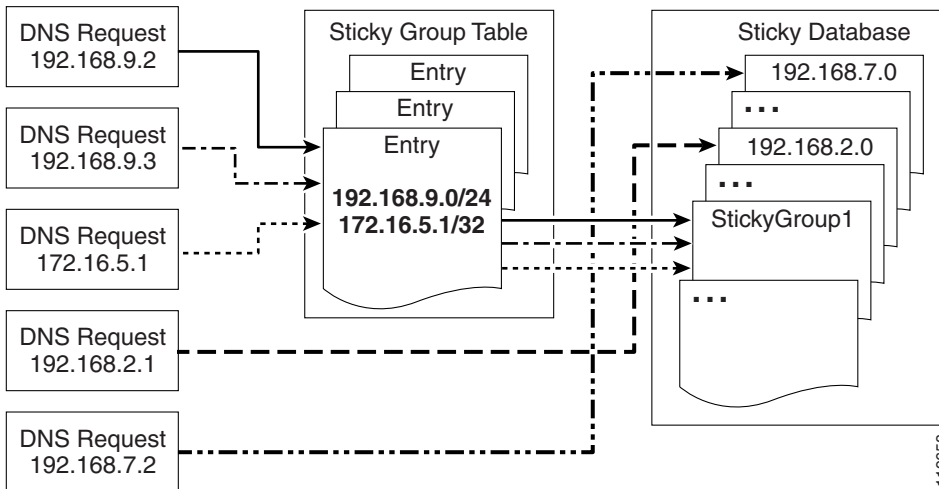
Grouping D-proxy IP addresses in the sticky database allows you to address proxy hopping. Certain ISPs rotate their D-proxies. A user's browser may use DNS server A to resolve a hostname and later use DNS server B to resolve the same name. This technique is referred to as proxy hopping because the DNS sticky function remembers the client's D-proxy IP address and not the IP address of the actual client. Thus, rotating D-proxies appear to the GSS as unique clients. Sticky grouping allows you to globally group sets of D-proxies to solve this proxy hopping problem.

In addition to creating DNS sticky groups of multiple D-proxy IP addresses from the CLI, you can configure a global netmask to uniformly group contiguous D-proxies (see the [“Configuring DNS Sticky”](#) section). The global netmask is

used by the GSS device when no DNS sticky group matches the incoming D-proxy address. The GSS uses the full incoming D-proxy IP address (255.255.255.255) and the global netmask as the key to look up in the DNS sticky database. The default global mask is 255.255.255.255.

Figure 8-1 shows how through DNS sticky group entries 192.168.9.0 255.255.255.0 and 172.16.5.1 255.255.255.255, the DNS requests from D-proxies 192.168.9.2, 192.168.9.3, and 172.16.5.1 all map to the identified group name, *StickyGroup1*. If no match is found in the sticky group table for an incoming D-proxy IP address, the GSS applies a user-specified global netmask to calculate a network address as the database key. In this example, DNS requests from 192.168.2.1 and 192.168.7.2 use the database entries keyed as 192.168.2.0 and 192.168.7.0 with a specified global netmask of 255.255.255.0.

Figure 8-1 Locating a Grouped Sticky Database Entry



Creating a DNS Sticky Group

You can create a DNS sticky group by using the **sticky group** global server load-balancing command from the primary GSSM CLI to identify the name of the DNS sticky group and add an IP address block to the group. Use the **no** form of the command to delete a previously configured IP address block from a sticky group or to delete a sticky group.

At the CLI of the primary GSSM, you create sticky groups to obtain better scalability of your configuration and to allow easy sticky group creation through automated scripts. The sticky groups are saved in the primary GSSM database and all GSS devices in the network receive the same sticky group configuration. You cannot create sticky groups using the CLI of a standby GSSM or individual GSS devices.

The syntax of this command is as follows:

```
sticky group groupname ip ip-address netmask netmask
```

The keywords and arguments are as follows:

- *groupname*—Unique alphanumeric name for the DNS sticky group; Use a maximum of 80 characters and avoid names that include spaces since they are not allowed.
- **ip** *ip-address*—Specifies the IP address block in dotted-decimal notation (for example, 192.168.9.0).
- **netmask** *netmask*—Specifies the subnet mask of the IP address block in dotted-decimal notation (for example, 255.255.255.0).

This example shows how to create a sticky group called *StickyGroup1* with an IP address block of 192.168.9.0 255.255.255.0:

```
gssm1.example.com# config  
gssm1.example.com(config)# gslb  
gssm1.example.com(config-gslb)# sticky group StickyGroup1 ip  
192.168.9.0 netmask 255.255.255.0
```

Reenter the **sticky group** command if you want to perform one of the following tasks:

- Add multiple IP address blocks to a DNS sticky group
- Create additional DNS sticky groups

Each sticky group can have a maximum of 30 blocks of defined IP addresses and subnet masks. The GSS prohibits duplication of IP addresses and subnet masks among DNS sticky groups.

Deleting a Sticky Group IP Address Block

You can delete a previously configured IP address block from a sticky group by using the **no** form of the **sticky group ip** command as follows:

```
gssm1.example.com# config
```

```
gssm1.example.com(config)# gslb
gssm1.example.com(config-gslb)# no sticky group StickyGroup1 ip
192.168.3.0 netmask 255.255.255.0
```

Deleting a Sticky Group

You can delete a sticky group by using the **no** form of the **sticky group** command as follows:

```
gssm1.example.com# config
gssm1.example.com(config)# gslb
gssm1.example.com(config-gslb)# no sticky group StickyGroup1
```

Deleting Entries from the Sticky Database

You can remove entries from the sticky database of each GSS device by using the **sticky database delete** command. When operating in a global sticky configuration, the result of the **sticky database delete** command propagates throughout the GSS mesh to maintain synchronization between the peers in the GSS network.



Caution

Use the **sticky database delete all** command when you want to remove all entries and empty the sticky database. Ensure that you want to permanently delete entries from the sticky database before you enter this command since you cannot retrieve sticky database entries once you delete them.

To view the entries in the sticky database to identify the sticky entries that you want to delete, use the **show sticky database** command (see the “[Displaying the Sticky Database Status](#)” section in [Chapter 13, Displaying GSS Global Server Load-Balancing Statistics](#)).

Use the **sticky database delete** command to remove entries from the sticky database.

The syntax of this command is as follows:

```
sticky database delete { all | answer { name/ip_address } | domain { name } |
domain-list { name } | group { name } | inactive minimum { minutes }
maximum { minutes } | ip { ip_address } netmask { netmask } | rule
{ rule_name } }
```

The keywords and arguments are as follows:

- **all**—Removes all entries from the sticky database memory. The prompt “Are you sure?” appears to confirm the deletion of all database entries. Specify **y** to delete all entries or **n** to cancel the deletion operation.
- **answer** *name/ip_address*—Removes all sticky entries related to a particular answer. Specify the name of the answer. If there is no name for the answer, specify the IP address of the sticky answer in dotted-decimal notation (for example, 192.168.9.0).
- **domain** *name*—Removes all sticky entries related to a domain. Specify the exact name of a previously created domain.
- **domain-list** *name*—Removes all sticky entries related to a domain list. Specify the exact name of a previously created domain list.
- **group** *name*—Removes all sticky entries related to a sticky group. Specify the exact name of a previously created sticky group.
- **inactive** **minimum** *minutes* **maximum** *minutes*—Removes all sticky entries that have not received a lookup request by a client D-proxy in the specified minimum and maximum time interval. Valid entries are 0 to 10100 minutes. If you do not specify a maximum value, the GSS deletes all entries that have been inactive for the specified minimum value or longer. The GSS returns an error if one of the following situations occur:
 - The maximum value is set to a value that is less than the minimum value.
 - The minimum and maximum values are not within the allowable range of values for the sticky inactivity timeout.
- **ip** *ip_address* **netmask** *netmask*—Removes all sticky entries related to a D-proxy IP address and subnet mask. Specify the IP address of the requesting client’s D-proxy in dotted-decimal notation (for example, 192.168.9.0) and specify the subnet mask in dotted-decimal notation (for example, 255.255.255.0).
- **rule** *rulename*—Removes all sticky entries related to a DNS rule. Specify the exact name of a previously created DNS rule.

To remove the D-proxy IP address 192.168.8.0 and subnet mask 255.255.255.0, enter:

```
gssm1.example.com# sticky database delete ip 192.168.8.0 netmask  
255.255.255.0
```

Dumping Sticky Database Entries

The GSS automatically dumps sticky database entries to a backup file on disk approximately every 20 minutes. The GSS uses this backup file to initialize the sticky database upon system restart or reboot to enable the GSS to recover the contents of the database. When global sticky is enabled, the GSS restores from the database dump file any time it reenters the mesh and cannot retrieve the sticky database contents from a GSS peer in the mesh.

You can dump all entries or selected entries from the sticky database to a named file as a user-initiated backup file. You can then use the **ftp** command in EXEC or global configuration mode to launch the FTP client and transfer the file to and from remote machines.

To view the entire contents of a sticky database XML output file from the GSS, use the **type** command. See the *Cisco Global Site Selector Administration Guide* for details about displaying the contents of a file.

The GSS includes options that provide a level of granularity for dumping entries from the sticky database. The GSS supports binary and XML output formats. Optionally, you can specify the entry type filter to clarify the information dumped from the sticky database.

If you specify a format but do not specify an entry type, the GSS automatically dumps all entries from the sticky database.

If you attempt to overwrite an existing sticky database dump file with the same filename, the GSS displays the following message:

Sticky Database dump failed, a file with that name already exists.

You can output entries from the sticky database by using the **sticky database dump** command.

The syntax of this command is as follows:

```
sticky database dump {filename} format {binary | xml} entry-type {all | group | ip}
```

The keywords and arguments are as follows:

- *filename*—Name of the output file that contains the sticky database entries on the GSS disk. This file resides in the /home directory.

- **format**—Dumps the sticky database entries in a binary or XML format. Choose the binary encoding as the format type if you intend to load the contents of the file into the sticky database of another GSS. The valid entries are as follows:
 - **binary**—Dumps the assigned sticky entries in true binary format. This file can be used only with the **sticky database load** command.
 - **xml**—Dumps the assigned sticky entries in an XML format. The contents of an XML file includes the data fields and the data descriptions. The contents of this file can be viewed using the **type** command. See [Appendix B, “Sticky and Proximity XML Schema Files”](#) for information on defining how content appears in output XML files.



Note Dumping sticky database entries in an XML format can be a resource intensive operation and may take from 2 to 4 minutes to complete depending on the size of the sticky database and the GSS platform in use. We recommend that you do not perform a sticky database dump in an XML format during the routine operation of the GSS to avoid a degradation in performance.

- **entry-type**—Specifies the type of sticky database entries to dump. The valid entries are as follows:
 - **all**—Dumps all entries from the sticky database
 - **group**—Dumps all entries that have sticky group IDs from the database
 - **ip**—Dumps all entries that have source IP addresses from the database

This example shows how to dump the D-proxy source IP addresses from the sticky database to the `sdb2004_06_30` file in XML format. For large numbers of entries, progress messages may appear.

```
gssm1.example.com# sticky database dump sdb2004_06_30 format xml
entry-type ip
Starting Sticky Database dump.
```

```
gssm1.example.com# sticky database dump sdb2004_06_30 format xml
entry-type ip
Sticky Database dump is in progress...
Sticky Database has dumped 15678 of 34512 entries
```

```
gssm1.example.com# sticky database dump sdb2004_06_30 format xml
entry-type ip
```

```
Sticky Database dump completed. The number of dumped entries: 34512
gssm1.example.com#
```

When the dump finishes, a “completed” message displays and the CLI prompt reappears.

Running a Periodic Sticky Database Backup

You can instruct the GSS to dump sticky database entries to an output file on the GSS disk before the scheduled time. You may want to initiate a sticky database dump as a database recovery method to ensure that you store the latest sticky database entries before you shut down the GSS.

You can force an immediate backup of the sticky database residing in GSS memory by using the **sticky database periodic-backup now** command. The GSS sends the sticky database entries to the system dump file as the sticky database file. Upon a reboot or restart, the GSS reads this file and loads the contents to initialize the sticky database at boot time.

The syntax of this command is as follows:

```
sticky database periodic-backup now
```

For example, enter:

```
gssm1.example.com# sticky database periodic-backup now
```

Loading Sticky Database Entries

The GSS supports the loading and merging of sticky database entries from a file into the existing sticky database in GSS memory. The merge capability supports the addition of entries from one GSS to another GSS. The file must be in binary format for loading into GSS memory.

The GSS validates the loaded database entries, checks the software version for compatibility, and then adds the sticky database entries in memory. The GSS does not overwrite duplicate entries in the sticky database.

You can load and merge a sticky database from disk into the existing sticky database in GSS memory by using the **sticky database load** command.

The syntax of this command is as follows:

```
sticky database load filename
```

**Note**

If you want to load and replace all sticky database entries from a GSS instead of merging the entries with the existing sticky database, enter the **sticky database delete all** command to remove all entries from sticky database memory before you enter the **sticky database load** command.

Specify the name of the sticky database file to load and merge with the existing sticky database on the GSS device. The file must be in binary format for loading into GSS memory (see the “[Dumping Sticky Database Entries](#)” section). Use the **ftp** command in EXEC or global configuration mode to launch the FTP client and transfer the sticky database file to the GSS from a remote GSS.

To load and merge the entries from the GSS3SDB file with the existing entries in the GSS sticky database, enter:

```
gssml.example.com# sticky database load GSS3SDB
```

Disabling DNS Sticky Locally on a GSS for Troubleshooting

You can disable DNS sticky for a single GSS when you need to locally override the globally-enabled sticky option to troubleshoot or debug the device. The GSS does not store the local-disable setting in its running-configuration file. When you restart the device and sticky has been enabled from the primary GSSM, the GSS reenables DNS sticky.

You can locally override the sticky enable option of the primary GSSM by using the **sticky stop** and **sticky start** commands.

When you enter the **sticky stop** command, the GSS immediately stops the following operations:

- Sticky lookups in the sticky database
- Accessing the sticky database for new requests
- Periodic sticky database dumps
- Sticky database entry age-out process

The GSS continues to answer DNS requests according to the DNS rules and keepalive status.

When you locally disable sticky on a GSS, sticky remains disabled until you perform one of the following actions:

- Enter the **sticky start** CLI command.
- Enter the **gss restart** CLI command to restart the GSS software.
- Enter the **reload** CLI command to perform a cold restart of the GSS device.

If you are using global DNS sticky in your network, upon reentry of the GSS device into the peer mesh, the GSS attempts to synchronize the database entries with the other peers in the mesh. The GSS queries each peer to find the closest up-to-date sticky database. If no update is available from a peer, the GSS initializes the sticky database entries from the previously saved database on the disk if a file is present and valid. Otherwise, the GSS starts with an empty sticky database.

To use the **sticky stop** command to locally disable DNS sticky on a GSS device, enter:

```
gssm1.example.com# sticky stop
```

To use the **sticky start** command to locally reenable DNS on the GSS device, enter:

```
gssm1.example.com# sticky start
```