



# Configuring a Client-Side Accelerator

---

This chapter provides an overview of the CSS Client Side Accelerator (CSA) feature and describes how to configure it for operation. Information in this chapter applies to all CSS models, except where noted.



---

**Note**

The Client Side Accelerator feature requires the CSS Enhanced feature set license.

---

This chapter provides the following major sections:

- [Overview of the Client Side Accelerator Feature](#)
- [Configuration Examples of CSA](#)
- [CSA Quick Start](#)
- [Configuring CSA Parameters](#)
- [Displaying CSA Information](#)

# Overview of the Client Side Accelerator Feature

To accelerate the retrieval of domain content, configure a CSS as a CSA, using transparent caches (TCs) to store content locally. This feature improves a user's experience by reducing the time for content to arrive in a browser.

A CSA resides on the client side of the Internet and is the first DNS server to which clients send a DNS request. When a CSA receives a DNS request for content located in a domain configured for acceleration and the number of requests exceeds the configured threshold, the CSA returns an address record (A-record) of the local virtual IP address (VIP) to the client. The client uses the IP address in the A-record to connect to the service in a local TC farm.

For non-accelerated content or unresolvable DNS requests, the CSA sends the DNS request to a DNS server forwarder. The forwarder, which is not a CSS, is a fully-functional Berkeley Internet Name Domain (BIND) DNS server. The forwarder returns a DNS response to the client transparently through the CSA.

You can configure a peer mesh of multiple CSAs that belong to one service provider but are located at various points of presence (POPs). Using Application Peering Protocol (APP), the CSAs in a peer mesh share accelerated domain records. This allows you to leverage content available at a cache farm in one POP to provide content to clients located at another POP. Once the same candidate domain has been accelerated at two POPs, cache backup can occur if a cache at one of those POPs fails.

Service providers can use CSAs to bill back domain acceleration to content providers. You can configure certain domains for acceleration, then bill back content providers based on the number of hits on the accelerated domains.

# Configuration Examples of CSA

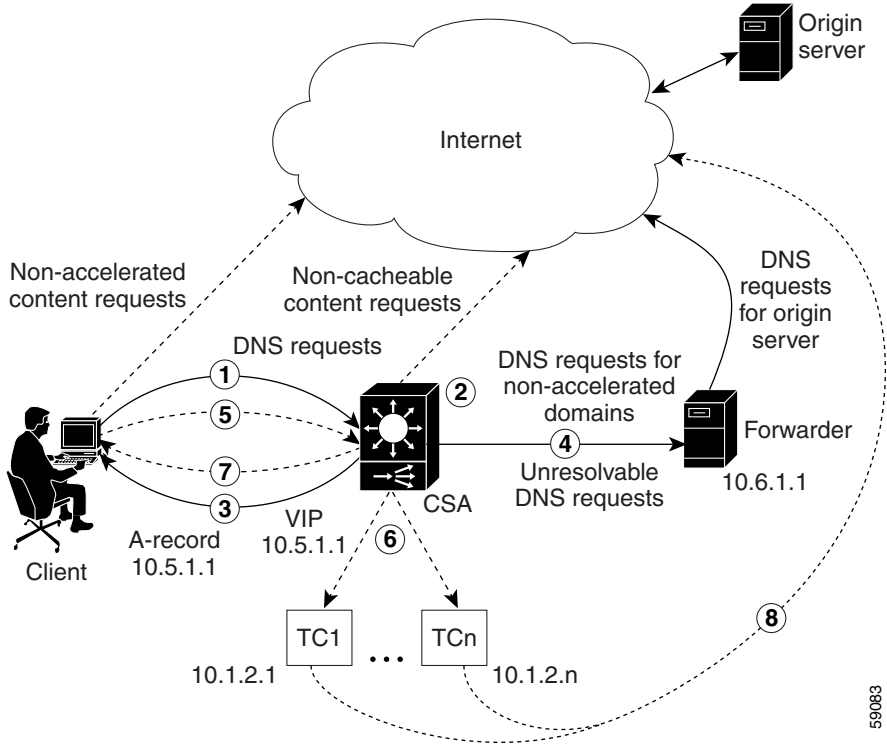
This section describes two possible CSA configurations: the first, a single POP (Figure 4-1) and the second, two POPs using an APP peer mesh (Figure 4-2).

## Single-POP CSA Configuration

The following sequence describes the steps depicted in Figure 4-1:

1. The Client population sends DNS requests to the CSA for DNS resolution of the domain name `www.acme.com`.
2. The CSA is configured to accelerate the domain `www.acme.com`.
3. When the number of requests for `www.acme.com` exceed the configured threshold (or the threshold has already been exceeded), the CSA returns the accelerated VIP in an A-record to the clients.
4. For all other requests, the CSS forwards the queries to the DNS server forwarder for resolution.
5. Clients initiate a connection with the CSA for `www.acme.com` using the VIP in the A-record.
6. The CSA matches the request on a Layer 5 content rule that has transparent caches configured as the services. The CSA performs destination NATing based on the host tag and performs MAC forwarding.
7. If the cache contains the content, the CSA returns it to the clients.
8. If the cache does not contain the content, the cache fetches the content from the origin server.

Figure 4-1 Example of a Client Side Accelerator Configuration Example



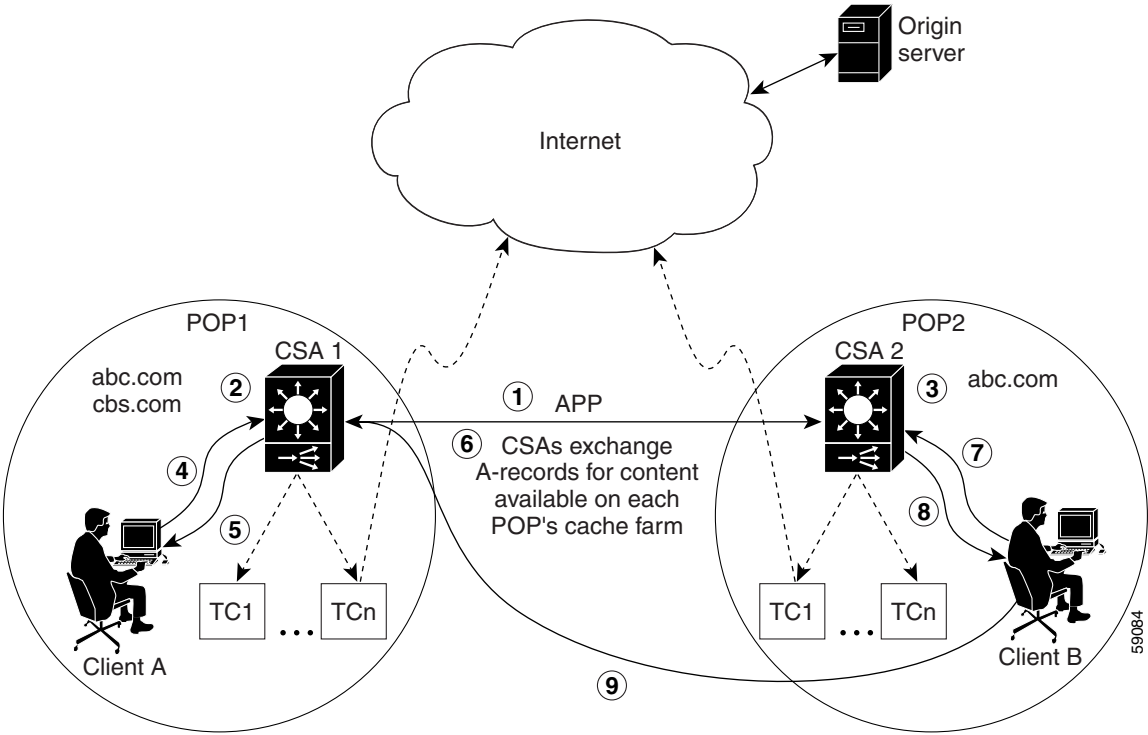
59083

## Multiple-POP CSA Configuration

The following sequence describes the steps depicted in [Figure 4-2](#).

1. An APP mesh is configured for the CSAs in POP1 and POP2.
2. CSA1 in POP1 is configured to accelerate the domains abc.com and cbs.com.
3. CSA2 in POP2 is configured to accelerate the domain abc.com.
4. Client A population sends DNS requests to CSA1 in POP1 for abc.com.
5. When the number of requests for abc.com exceeds the configured threshold, CSA1 creates an A-record for abc.com and returns it to the clients. Clients in POP1 initiate a connection with CSA1 using the VIP in the A-record.
6. CSA1 also sends the A-record for abc.com out on the APP mesh.
7. Client B population sends DNS requests for abc.com to CSA2 in POP2. If CSA2 is configured to accelerate abc.com in multiple locations and if the domain becomes hot (requests exceed configured threshold), repeat Steps 5 and 6 for CSA2 in POP2.
8. If abc.com is not yet hot in POP2 or if CSA2 is configured to accelerate the domain in a single location, CSA2 sends the A-record in its domain database (learned in Step 6) to the Client B population.
9. The clients in POP2 request content in abc.com from CSA1 in POP1.

Figure 4-2 Example of a Client Side Accelerator APP Mesh Configuration



59064

# CSA Quick Start

Table 4-1 provides a quick overview of the steps required to configure the Client Side Accelerator feature on a CSS. Each step includes the CLI command required to complete the task. For a complete description of each feature and all the options associated with the CLI command, see the sections following the table.

**Table 4-1 Client Side Accelerator Configuration Quick Start**

---

**Task and Command Example**

---

1. Enter config mode.

```
# config  
(config)#
```

---

2. Enable Application Peering Protocol (APP).

```
(config)# app
```

---

3. Configure an APP session with each remote CSA peer to create a peer mesh. See the “Configuring an APP Session” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

```
(config)# app session 172.27.16.2
```

---

4. Configure back-end remapping as the preferred connection reset method. Refer to the *Cisco Content Services Switch Content Load-Balancing Configuration Guide*. This step is recommended, but not required.

```
(config)# persistence reset remap  
(config)# bypass persistence disable
```

---

5. Configure candidate domains for acceleration. See “Configuring Accelerated Domains” later in this chapter.

```
(config)# dns-record accel abc.com 192.168.1.3  
(config)# dns-record accel cbs.com 192.168.1.3
```

---

6. Configure CSA and enable sharing of content between peer CSAs. See “Enabling the CSA Feature” later in this chapter.

```
(config)# dns-server accelerate domains 50 30 256 single-location
```

---

7. Configure the DNS server and the number of CSA peers. See the “Configuring DNS Server Zones” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

```
(config)# dns-server zone 1 tier2
```

---

**Table 4-1 Client Side Accelerator Configuration Quick Start (continued)****Task and Command Example**

8. Configure the DNS server forwarder. See the “Configuring a DNS Forwarder” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

```
(config)# dns-server forwarder primary 192.168.1.10
```

9. Enable the DNS server.

```
(config)# dns-server
```

10. Configure the transparent caches as services. Refer to the *Cisco Content Services Switch Content Load-Balancing Configuration Guide*.

```
(config)# service transHosttag1
(config-service[transHosttag])# ip address 10.1.2.1
(config-service[transHosttag])# protocol tcp
(config-service[transHosttag])# port 80
(config-service[transHosttag])# type transparent
(config-service[transHosttag])# transparent-hosttag
(config-service[transHosttag])# active
```

Repeat for each additional cache.

11. Configure an EQL.

```
(config)# eql cacheable
(config-eql[cacheable])# extension .jpg
```



**Table 4-1 Client Side Accelerator Configuration Quick Start (continued)****Task and Command Example**

12. Configure a content rule with the same VIP address as that used for the accelerated records. Add the transparent caches as services. Refer to the *Cisco Content Services Switch Content Load-Balancing Configuration Guide*.

```
(config)# owner accelerate
(config-owner[accelerate])# content 15-accel
(config-owner-content[accelerate-15-accel])# vip address
192.168.1.3
(config-owner-content[accelerate-15-accel])# protocol TCP
(config-owner-content[accelerate-15-accel])# port 80
(config-owner-content[accelerate-15-accel])# url "/"* eql
cacheable
(config-owner-content[accelerate-15-accel])# add service
transHosttag1
(config-owner-content[accelerate-15-accel])# add service
transHosttagn
(config-owner-content[accelerate-15-accel])# no persistent
(config-owner-content[accelerate-15-accel])# balance url
(config-owner-content[accelerate-15-accel])# active
```

13. Configure a service to bypass the cache farm for noncacheable content. Refer to the *Cisco Content Services Switch Content Load-Balancing Configuration Guide*.

```
(config)# service bypassCache
(config-service[bypassCache])# ip address 0.0.0.0
(config-service[bypassCache])# protocol tcp
(config-service[bypassCache])# port 80
(config-service[bypassCache])# keepalive type none
(config-service[bypassCache])# bypass-hosttag
(config-service[bypassCache])# active
```

**Table 4-1 Client Side Accelerator Configuration Quick Start (continued)****Task and Command Example**

14. Configure a content rule for non-cacheable content on domains that you want to accelerate. Add the bypass cache as the only service. Refer to the *Cisco Content Services Switch Content Load-Balancing Configuration Guide*.

```
(config)# owner accelerate
(config-owner[accelerate])# content nonCacheable
(config-owner-content[accelerate-nonCacheable])# vip address
192.168.1.3
(config-owner-content[accelerate-nonCacheable])# protocol TCP
(config-owner-content[accelerate-nonCacheable])# port 80
(config-owner-content[accelerate-nonCacheable])# url "/"
(config-owner-content[accelerate-nonCacheable])# add service
bypassCache
(config-owner-content[accelerate-nonCacheable])# no persistent
(config-owner-content[accelerate-nonCacheable])# active
```

The following running-config example shows the results of entering the commands described in [Table 4-1](#).

```
!***** GLOBAL *****
persistence reset remap
bypass persistence disable

dns-record accel abc.com 192.168.1.2
dns-record accel cbs.com 192.168.1.2
dns-server accelerate domains 50 30 256 single-location
dns-server zone 1 tier2
dns-server forwarder primary 192.168.1.10
dns-server

app
app session 172.27.16.2
!***** SERVICE *****
service bypassCache
ip address 0.0.0.0
protocol tcp
port 80
keepalive type none
bypass-hosttag
active
```

```
service transHosttag1
  ip address 10.1.2.1
  protocol tcp
  port 80
  type transparent-cache
  transparent-hosttag
  active

!***** EQL *****
eql cacheable
  extension .jpg

!***** OWNER *****
owner accelerate

content l5-accel
  vip address 192.168.1.3
  protocol tcp
  port 80
  url "/" eql cacheable
  add service transHosttag1
  no persistent
  balance url
  active

content nonCacheable
  vip address 192.168.1.3
  protocol tcp
  port 80
  url "/"
  add service bypassCache
  no persistent
  active
```

# Configuring CSA Parameters

The following sections describe the CLI commands and their options and variables that you use to configure the CSA feature on a CSS.

## Enabling the CSA Feature

To enable the CSA functionality on a CSS, use the **dns-server accelerate domains** command. This command enables the acceleration of domains that have been or will be configured for acceleration using the **dns-record accel** command. The CSA uses the *threshold* variable to determine if it should accelerate a candidate domain. You can also configure whether or not the CSA shares content with peer CSAs.

The syntax for this global configuration mode command is:

```
dns-server accelerate domains threshold interval max_number  
[single_location|multi_location]
```

The variables and options for this command are:

- *threshold* - The number of hits per *interval* below which the CSA does not accelerate a domain. When the number of hits equals or exceeds the threshold during the configured *interval*, the CSA accelerates the domain. Enter an integer from 0 to 65535. The default is 0, indicating immediate acceleration.
- *interval* - The time period in minutes over which the CSA samples the hits on a domain and compares the number of hits with the configured *threshold* value to determine hot content and domain acceleration. Enter an integer from 1 to 60 minutes. The default is 5 minutes.
- *max\_number* - The maximum number of domains that the CSA can accelerate. Enter an integer from 0 to 4096. The default is 1024.
- **single\_location** - Allows the acceleration of a domain at one cache farm (POP) at a time. This is the default behavior.

- **multi\_location** - Allows multiple CSAs to accelerate the same domain, possibly resulting in multiple cache farms maintaining the same content. This can occur when two or more CSAs (located in different POPs) are configured for `multi_location` and accelerate the same domain. Each cache farm will maintain the same content after:
  - The CSAs accelerate the same domain
  - A cache in each POP retrieves the same content from the origin server

The following command example:

- Accelerates domains that are accessed at least at the rate of 50 hits per minute.
- Accelerates a maximum of 100 candidate domains at any given time.
- Forces this CSA to allow acceleration of a given domain to occur in only one cache farm at a time.

```
(config)# dns-server accelerate domains 50 1 100 single_location
```

To disable CSA functionality on a CSS, enter:

```
(config)# no dns-server accelerate domains
```

## Configuring the Domain Cache

To enable the tracking of DNS request counts and to configure the domain cache on the CSA, use the **dns-server domain-cache** command. As requests arrive at the CSA, entries are created or updated in the domain cache with the hit counts. You can use this command with the **show dns-server domain-cache** command to determine which domains you want to accelerate based on their hit counts.



### Note

---

Do not use this command during normal CSA operation. It causes unnecessary overhead on the CSA.

---

The syntax for this global configuration command is:

```
dns-server domain-cache {cache_size ageout|purge {dns_name}|zero  
  {dns_name}}
```

The variables and options for this command are:

- *cache\_size* - Specifies the number of domains that the CSA can cache. Enter an integer from 1 to 4096. The default is 1024.
- *ageout* - The maximum number of seconds that the domain entry remains in the cache. Enter an integer from 0 to 60. The default is 10 seconds. A value of zero causes the domain entry to never age out.
- **purge** - Removes all entries or the specified entries from the domain cache.
- *dns\_name* - The DNS entry that you want to remove from the domain cache. To see a list of DNS entries, enter:  

```
(config)# dns-server domain-cache purge ?
```
- **zero** - Resets all counters for all entries or the specified entry in the domain cache to zero.
- *dns\_name* - The DNS entry for which you want to reset counters.

For example:

```
(config)# dns-server domain-cache 512
```

The above command creates a domain cache that can hold up to 512 most recently requested domain entries. The entries will age out and be removed from the domain cache after 10 seconds (the default).



#### Note

The operation of the domain cache can impact the DNS request/response rate performance. Use the domain cache only when you need to identify potential acceleration candidates.

## Configuring a DNS Server Forwarder

To configure a DNS server forwarder on a CSS configured as a CSA, use the **dns-server forwarder** command. If the CSA cannot resolve a DNS request, it sends the request to the forwarder to obtain a suitable response. For details on configuring a DNS server forwarder, see the “[Configuring a DNS Forwarder](#)” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

## Configuring Accelerated Domains

To specify the domains you want to accelerate, use the **dns-record accel** command. Map the domain name to a content rule using an IP address.

**Note**

If the content rule bound to the acceleration candidate domain is suspended or cannot provide service for content requests, the CSA does not accelerate the domain.

The syntax for this global configuration mode command is:

```
dns-record accel dns_name ip_address {ageout}
```

**Note**

You cannot configure a domain name as two different DNS record types on the same CSS. For example, if you have configured abc.com as **dns-record accel**, you cannot configure it as **dns-record a** or **dns-record ns** on the same CSS. For information on configuring other types of DNS records, see the “[Configuring Domain Records](#)” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

The variables and options for this command are:

- *dns\_name* - The domain name that you want to map to the acceleration record. Enter the name as a case-sensitive unquoted text string with no spaces and a maximum of 63 characters.
- *ip\_address* - The IP address of the local content rule that will handle content requests for the DNS name during content acceleration.
- *ageout* - The optional number of minutes that the domain remains accelerated. Enter an integer from 0 to 525600. The default is 180 minutes. If you enter 0, the accelerated domain record never ages out.

The following command example creates an acceleration record for the domain abc.com. When the number of requests for the domain exceeds the threshold specified in the **dns-server accelerate domains** command, the CSA accelerates the domain for six minutes. Clients can access the domain’s content based on the content rule with the IP address 192.168.11.1.

```
(config)# dns-record accel abc.com 192.168.11.1 6
```

To delete the domain acceleration record, enter:

```
(config)# no dns-record accel abc.com
```

## Resetting the DNS Record Statistics

To reset the DNS record statistics displayed by the **show dns-record** command, use the **dns-record zero** command. For details on resetting DNS record statistics, see the “[Resetting the DNS Record Statistics](#)” and “[Displaying DNS-Record Statistics](#)” sections in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

## Configuring the CSA DNS Server Zones

To configure the number of DNS server zones on a CSA, use the **dns-server zone** command. For details on DNS server zones, see the “[Configuring DNS Server Zones](#)” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

## Displaying CSA Information

To display information and statistics for your CSA configuration, use the **show** commands in this section.

## Displaying a CSA Configuration

To display the CSA configuration on a CSS, use the **show dns-server accelerate domains** command. The syntax for this global configuration mode command is:

```
show dns-server accelerate domains
```



Table 4-2 describes the fields in the `show dns-server accelerate domains` output.

**Table 4-2** *Field Descriptions for the show dns-server accelerate domains Command*

| Field                  | Description  |
|------------------------|--|
| Current CSA Config     | The state of the CSA configuration: disabled or enabled.   |
| Threshold              | The configured hits threshold used to determine whether or not a domain is accelerated. When the hits on the domain are greater than or equal to the threshold, the CSA accelerates the domain. The range is from 0 to 65535. The default is 0, indicating that the candidate domains are always accelerated.  |
| Interval               | The configured time interval, in minutes, over which the CSS samples the hits on the domain and compares it with the threshold. The range is from 1 to 60 minutes. The default is 5 minutes.   |
| Expirations            | The number of times that the configured interval expired. You can use this value to determine whether domains are being accelerated or decelerated as expected. The CSA decelerates an accelerated domain only after the interval expires.   |
| Max. to Accel          | The maximum number of domains that can be accelerated. The range is 0 to 4096. The default is 1024.  |
| Location               | Indicates whether a single or multiple CSAs maintain the same content. <ul style="list-style-type: none"> <li>• Single-location, the default setting, allows the acceleration of a domain at one cache farm (POP) at a time.</li> <li>• Multi-location allows multiple CSAs to accelerate the same domain resulting in multiple cache farms maintaining the same content.</li> </ul> |
| Candidates Total       | The total number of candidate domains that are configured on the CSA.  |
| Candidates Accelerated | The total number of domains that are currently accelerated.  |

## Displaying DNS Server Domain Cache Statistics

To display statistics for the DNS server domain cache, use the **show dns-server domain-cache** command. Use this command with the **dns-server domain-cache** command to help you determine which domains you want to accelerate based on their hit counts. The syntax for this global configuration mode command is:

```
show dns-server domain-cache {summary}
```

[Table 4-3](#) describes the fields in the **show dns-server domain-cache** output.

**Table 4-3** *Field Descriptions for the show dns-server domain-cache Command*

| Field   | Description                  |
|---------|------------------------------|
| Domain  | The domain name of the entry |
| Counter | The number of DNS requests   |



### Note

If you include the **summary** option, the command output displays the domain cache configuration and state only.

## Displaying DNS Server Zones

To display information about communication with, and the state of, the specified DNS zone or all zones in a peer mesh, use the **show zone command**. For details on displaying DNS server zones, see the “[Displaying DNS Server Zones](#)” section [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

## Displaying DNS Record Keepalive Information

To display DNS record keepalive information, use the **show dns-record keepalive** command. For details on displaying DNS record keepalives, see the “[Displaying DNS Record Keepalive Information](#)” section in [Chapter 1, Configuring the CSS as a Domain Name System Server](#).

## Displaying Domain Acceleration Records Statistics

To view statistics associated with domain acceleration records, use the **show dns-record accel** command. The syntax for this global configuration mode command is:

```
show dns-record accel dns-name
```

The variable for this command is *dns\_name*, the domain name mapped to the acceleration record that you want to display. Enter the name as a case-sensitive unquoted text string with no spaces and a maximum of 63 characters.

[Table 4-4](#) describes the fields in the **show dns-record accel** output.

**Table 4-4** *Field Descriptions for the show dns-record accel Command*

| Field           | Description  |
|-----------------|--|
| <Name>          | The domain name for the acceleration record.   |
| State           | The state of the acceleration record, either ACCEL or NOT ACCEL. <ul style="list-style-type: none"> <li><b>ACCEL</b> - Indicates that the record is currently accelerated</li> <li><b>NOT ACCEL</b> - Indicates the record is currently not accelerated</li> </ul> |
| Vip Address     | The virtual IP address of the local content rule that handles the content requests for the domain name during content acceleration.  |
| Secs til Ageout | The number of seconds remaining until the CSA decelerates the domain record. The range is from 0 to 525600. The default is 180 seconds.  |
| Interval Hits   | The number of content hits that occurred during the interval set with the <b>dns-server domain-cache</b> command.  |
| Total Hits      | The total number of DNS hits for this record.  |
| AccelCount      | The number of times that content was accelerated.  |

■ **Displaying CSA Information**