



Managing Caching DNS Server

This chapter explains how to set the Caching DNS server parameters. Before you proceed with the tasks in this chapter, see [Introduction to the Domain Name System](#), which explains the basics of DNS.

- [Configuring Caching DNS Server Network Interfaces, on page 1](#)
- [Setting DNS Caching Server Properties, on page 1](#)
- [Running DNS Caching Server Commands, on page 23](#)

Configuring Caching DNS Server Network Interfaces

You can configure the network interfaces for the Caching DNS server from the Manage Servers page in the local web UI.

Local Advanced Web UI

- Step 1** From the **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page.
 - Step 2** Select **CDNS** from the Manage Servers pane.
 - Step 3** Click the **Network Interfaces** tab to view the available network interfaces that you can configure for the server. By default, the server uses all of them.
 - Step 4** To configure an interface, click the **Configure** icon in the Configure column for the interface. This adds the interface to the Configured Interfaces table, where you can edit or delete it.
 - Step 5** Click the name of the configured interface to edit the configured interfaces, where you can change the address, direction and port of the interface.
 - Step 6** Click **Modify Interface** when you are done editing, then click **Go to Server Interfaces** to return to the Network Interfaces page.
-

Setting DNS Caching Server Properties

You can set properties for the Caching DNS server. These include:

- **General server properties**—See [Setting General Caching DNS Server Properties, on page 2](#)
- **Log settings**—See [Specifying Log Settings, on page 3](#)

- **Packet logging**—See [Enabling Packet Logging](#), on page 3
- **Activity summary settings**—See [Specifying Activity Summary Settings](#), on page 4
- **Top names settings**—See [Specifying Top Names Settings](#), on page 13
- **Caching settings**—See [Setting Prefetch Timing](#), on page 14
- **Cache TTLs**—See [Setting Cache TTLs](#), on page 14
- **Smart caching**—See [Enabling Smart Caching](#), on page 15
- **Root name servers**—See [Defining Root Nameservers](#), on page 17
- **UDP ports**—See [Dynamic Allocation of UDP Ports](#), on page 17
- **Maximum memory cache sizes**—See [Setting Maximum Memory Cache Sizes](#), on page 18
- **Resolver settings**—See [Specifying Resolver Settings](#), on page 18
- **Network settings**—See [Specifying Network Settings](#), on page 19
- **Advanced settings**—See [Specifying Advanced Settings](#), on page 19
- **Flush cache**—See [Flushing Caching DNS Cache](#), on page 20
- **Prevent DNS cache poisoning**—See [Detecting and Preventing DNS Cache Poisoning](#), on page 21
- **Handle unresponsive nameservers**—See [Handling Unresponsive Nameservers](#), on page 22

Setting General Caching DNS Server Properties

You can view general Caching DNS server properties, such as log settings, basic cache settings, SNMP traps, and root nameservers.

The following subsections describe some of the most common property settings. They are listed in [Setting DNS Caching Server Properties](#), on page 1.

Local Basic or Advanced Web UI

-
- Step 1** To access the server properties, from the **Deploy** menu, choose **CDNS Server** under the **DNS** submenu to open the Manage DNS Caching Server page.
- Step 2** The local CDNS Server page is automatically selected when you choose the **CDNS Server** tab, either from the Deploy menu or by clicking the **CDNS Server** tab in the left pane. The page displays all the Caching DNS server attributes.
- Step 3** Click **Save** to save the Caching DNS server attribute modifications.
-

CLI Commands

Use **cdns show** to display the Caching DNS server properties (see the **cdns** command in the CLIGuide.html file in the /docs directory for syntax and attribute descriptions).

Specifying Log Settings

This setting determines which detailed events the Caching DNS server logs, as set using a bit mask. Logging these additional details can help analyze a problem. However, leaving detailed logging enabled for a long period, can fill the log files and cause the loss of important information.

The possible options are:

- **activity-summary**—Causes logging of a server statistics summary at an interval specified by *activity-summary-interval*. The summary provides detailed statistics about the servers operation.
- **config**—Controls logging pertaining to server configuration and server de-initialization (unconfiguration).
- **query**—Causes logging of all DNS queries to the server. Note that query log entries are written to a separate *cdns_query_log* file.
- **scp**—Controls logging pertaining to SCP message processing.
- **server-detailed-ops**—Controls detailed logging of server operations.
- **server-ops**—Controls high level logging of server operations.

Enabling Packet Logging

Cisco Prime Network Registrar supports packet logging for Caching DNS server to help analyze and debug the Caching DNS server activity. The packet logging settings determine the type of packet logging (summary or detail), the type of packets logged, and to which log file the messages are logged. By default, the Caching DNS server does not log any packet log messages.

Use the following server level attributes to enable packet logging for the Caching DNS server:

Table 1: Caching DNS Server Packet Logging Attributes

Attribute	Description
Packet Logging (<i>packet-logging</i>)	<p>Determines the type of packet logging that is logged to the CDNS logs. The type of packets logged can be controlled with the <i>packet-log-settings</i> attribute.</p> <ul style="list-style-type: none"> • disabled—This settings disables packet logging. • summary—This setting enables one line summary packet logging. • detail—This setting enables detailed packet tracing. <p>Note: While packet logging can be helpful for debugging and troubleshooting, it does have an impact on DNS server performance. Therefore, Cisco does not recommend leaving packet logging enabled in production environments.</p>
Packet Logging File (<i>packet-logging-file</i>)	<p>Determines the destination log of packet log messages when packet logging is enabled.</p> <ul style="list-style-type: none"> • cdns—Packet logging messages are logged to the standard CDNS log file (<i>cdns_log*</i>). • packet—Packet logging messages are logged to a separate CDNS packet log file (<i>cdns_query_log*</i>).

Attribute	Description
Packet Log Settings (<i>packet-log-settings</i>)	<p>Determines the type of packets to log when packet logging is enabled. Packet logging can be enabled by configuring the <i>packet-logging</i> attribute.</p> <ul style="list-style-type: none"> • query-in—This setting enables logging of incoming query packets. These are packets coming in from DNS clients. • query-out—This setting enables logging of outgoing query packets. These are queries going to upstream DNS servers. • response-in—This setting enables logging of incoming query response packets. These are responses coming from upstream DNS servers. • response-out—This setting enables logging of outgoing query response packets. These are responses going to DNS clients.

Local Advanced Web UI

-
- Step 1** On the Manage DNS Caching Server page, under the **Packet Logging** section, select the value for **packet-logging** from the drop-down list. The value can be **summary** or **detail**.
- Step 2** For the *packet-log-settings* attribute, check the desired check boxes.
- Step 3** Click **Save** to save the changes.
-

CLI Commands

Use **cdns set packet-logging=summary** to enable one line summary packet logging.

Use **cdns set packet-logging=detail** to enable detailed packet tracing.

Use **cdns set packet-log-settings=value** to set the type of packets to log when packet logging is enabled.



Note Reloading of Caching DNS server is not required for the *packet-logging* and *packet-log-settings* attributes to take effect immediately (similar to log settings). However, the *packet-logging-file* attribute requires a Caching DNS server reload.

Specifying Activity Summary Settings



Note To specify the activity summary settings, you have to check *activity-summary* under Log Settings.

You can specify the interval at which to log activity summary information using the Statistics Interval (*activity-summary-interval*) attribute. It has a default value of 60 seconds.

The Caching DNS server logs sample and/or total statistics based on the option you check for the Statistics Type (*activity-summary-type*) attribute. The default value is "sample".

The option checked for the Statistics Settings (*activity-summary-settings*) attribute determines the category of statistics that is logged as part of activity summary. The possible settings are:

- **cache**—Logs statistics on the RR cache.

For the list of activity summary statistics that are displayed in the logs for the **cache** setting, see [Cache Statistics, on page 6](#).

- **firewall**—Logs statistics on DNS firewall usage.

For the list of activity summary statistics that are displayed in the logs for the **firewall** setting, see [Firewall Statistics, on page 6](#).

- **memory**—Logs statistics on memory usage.

For the list of activity summary statistics that are displayed in the logs for the **memory** setting, see [Memory Statistics, on page 7](#).

- **query**—Logs statistics related to incoming queries.

For the list of activity summary statistics that are displayed in the logs for the **query** setting, see [Query Statistics, on page 8](#).

- **query-type**—Logs statistics on the RR types that are being queried.

For the list of activity summary statistics that are displayed in the logs for the **query-type** setting, see [Query by Type Statistics, on page 9](#).

- **rate-limiting**—Logs the number of rate limiting events.

For the list of activity summary statistics that are displayed in the logs for the **rate-limiting** setting, see [Rate Limiting Statistics, on page 9](#).

- **resol-queue**—Logs statistics on the resolution queue.

For the list of activity summary statistics that are displayed in the logs for the **resol-queue** setting, see [Resolution Queue Statistics, on page 10](#).

- **responses**—Logs statistics about query responses.

For the list of activity summary statistics that are displayed in the logs for the **responses** setting, see [Responses Statistics, on page 11](#).

- **system**—Logs statistics on system usage.

For the list of activity summary statistics that are displayed in the logs for the **system** setting, see [System Statistics, on page 12](#).

- **top-names**—Logs the top names queried and hit count.

For the list of activity summary statistics that are displayed in the logs for the **top-names** setting, see [Top Names Statistics, on page 13](#).

Activity Summary Statistics

Following sections describe the list of activity summary statistics that are displayed in the logs under each of the *activity-summary-settings* category.

Cache Statistics

The **cache** activity-summary-settings logs statistics on the RR cache.

Sample log message:

```
10/06/2021 10:22:44 cdns Activity Stats 0 22173 [Cache] Sample since Wed Oct 6 10:21:44
2021: hits=number, misses=number, prefetches=number, message-overflow=number,
rrset-overflow=number, remote-ns-overflow=number, key-overflow=number, smart-cache=number
```

Table 2: Cache Statistics

Activity Summary Name	Statistic ¹	Description
hits	cache-hits	Total number of queries that were answered from cache.
misses	cache-misses	Total number of queries that were not found in the cache.
prefetches	cache-prefetches	Number of prefetches performed.
rrset-overflow	mem-cache-exceeded	Number of times the RRSet cache has gone over the configured limit. This indicates that the configured limit may be undersized for its environment.
message-overflow	mem-query-cache-exceeded	Number of times the message cache has gone over the configured limit. This indicates that the configured limit may be undersized for its environment.
remote-ns-overflow	remote-ns-cache-exceeded	Number of times the remote name server cache has gone over the configured limit. This indicates that the configured limit may be undersized for its environment.
key-overflow	key-cache-exceeded	Number of times the key cache has gone over the configured limit. This indicates that the configured limit may be undersized for its environment.
smart-cache	smart-cache	Total number of times the CDNS Server employed a smart-cache response, when <i>smart-cache</i> is enabled.

¹ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Firewall Statistics

The **firewall** activity-summary-settings logs statistics on DNS Firewall usage.

Sample log message:

```
11/18/2021 12:39:20 cdns Activity Stats 0 22322 [Firewall] Sample since Thu Nov 18 12:38:20
2021: redirected=number, dropped=number, refused=number, redirect-nxdomain=number, rpz=number
```

Table 3: Firewall Statistics

Activity Summary Name	Statistic ²	Description
dropped	firewall-dropped	Number of times DNS Firewall dropped a query.
redirected	firewall-redirected	Number of times DNS Firewall redirected a query.
refused	firewall-refused	Number of times DNS Firewall refused a query.
redirect-nxdomain	firewall-redirect-nxdomain	Number of times DNS Firewall redirected a query with an NXDOMAIN answer.
rpz	firewall-rpz	Number of times DNS Firewall RPZ rules matched an incoming query.

² The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Memory Statistics

The **memory** activity-summary-settings logs statistics on memory usage.

Sample log message:

```
10/06/2021 10:22:44 cdns Activity Stats 0 22303 [Memory] Current: mem-cache-process=number,
mem-cache-rrset=number, mem-cache-message=number, mem-mod-iterator=number,
mem-mod-validator=number
```

Table 4: Memory Statistics

Activity Summary Name	Statistic ³	Description
mem-cache-process	mem-process	An estimate of the memory in bytes of the CDNS process.
mem-cache-rrset	mem-cache	Memory in bytes allocated to the RRset cache. Note that the allocated memory will be maintained across server reloads, unless the <i>rrset-cache-size</i> configuration has changed.
mem-cache-message	mem-query-cache	Memory in bytes allocated to the message cache. Note that the allocated memory will be maintained across server reloads, unless the <i>msg-cache-size</i> configuration has changed.
mem-mod-iterator	mem-iterator	Memory in bytes used by the CDNS iterator module.
mem-mod-validator	mem-validator	Memory in bytes used by the CDNS validator module.

³ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Query Statistics

The **query** activity-summary-settings logs statistics related to incoming queries.

Sample log message:

```
10/06/2021 10:22:44 cdns Activity Stats 0 22171 [Query] Sample since Wed Oct 6 10:21:44
2021: total=number, acl-failures=number, udp=number, tcp=number, ipv4=number, ipv6=number,
edns=number, dnssec=number, dns64-aaaa=number, dns64-ptr=number, dns64-ns=number
```

Table 5: Query Statistics

Activity Summary Name	Statistic ⁴	Description
total	queries-total	Total number of queries received by the CDNS Server.
acl-failures	queries-failing-acl	Number of queries being dropped or refused due to ACL failures.
tcp	queries-over-tcp	Total number of queries received over TCP by the CDNS Server.
udp	N/A	Total number of queries received over UDP by the CDNS Server.
ipv4	N/A	Total number of IPv4 queries received by the CDNS Server.
ipv6	queries-over-ipv6	Total number of IPv6 queries received by the CDNS Server.
edns	queries-with-edns	Number of queries with EDNS OPT RR present.
dnssec	queries-with-edns-do	Number of queries with EDNS OPT RR with DO (DNSSEC OK) bit set.
dns64-aaaa	dns64-a2aaaa-conversions	Number of times dns64 has converted a type A RR to a type AAAA RR.
dns64-ptr	dns64-ptr-conversions	Number of times dns64 has converted an IPv4 PTR RR to an IPv6 PTR RR.
unwanted-class	queries-unwanted-class	Total number of queries with an unwanted classes.

⁴ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Query by Type Statistics

The **query-type** activity-summary-settings logs statistics on the RR types that are being queried.

Sample log message:

```
10/06/2021 10:22:44 cdns Activity Stats 0 22172 [Query-by-Type] Sample since Wed Oct 6
10:21:44 2021: A=number, AAAA=number, ANY=number, CNAME=number, PTR=number, MX=number,
NS=number, SOA=number, DS=number, DNSKEY=number, RRSIG=number, NSEC=number, NSEC3=number,
Other=number
```

Table 6: Query by Type Statistics

Activity Summary Name	Statistic ⁵	Description
A	queries-type-A	Number of A queries received.
AAAA	queries-type-AAAA	Number of AAAA queries received.
CNAME	queries-type-CNAME	Number of CNAME queries received.
PTR	queries-type-PTR	Number of PTR queries received.
NS	queries-type-NS	Number of NS queries received.
SOA	queries-type-SOA	Number of SOA queries received.
MX	queries-type-MX	Number of MX queries received.
DS	queries-type-DS	Number of DS queries received.
DNSKEY	queries-type-DNSKEY	Number of DNSKEY queries received.
RRSIG	queries-type-RRSIG	Number of RRSIG queries received.
NSEC	queries-type-NSEC	Number of NSEC queries received.
NSEC3	queries-type-NSEC3	Number of NSEC3 queries received.
Other	queries-type-other	All other queries received.
ANY	queries-type-ANY	Number of ANY queries received.

⁵ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Rate Limiting Statistics

The **rate-limiting** activity-summary-settings logs the number of rate limiting events.

Sample log message:

```
11/30/2021 16:20:37 cdns tid: 0 Activity Stats 0 22388 [Ratelimit] Sample since Tue Nov 30
16:19:37 2021: client-ratelimited=number, domain-ratelimited=number
```

```
11/30/2021 16:20:37 cdns tid: 0 Activity Stats 0 22390 [Ratelimit-Domain] from 16:19:37 to
```

```
16:20:33; interval=number, num-ratelimited=number, total-counted=number, not-counted=number
11/30/2021 16:20:37 cdns tid: 0 Activity Stats 0 22390 [Ratelimit-Client] from 08:29:43 to
08:30:43; interval=number, num-ratelimited=number, total-counted=number, not-counted=number
```

Table 7: Rate Limiting Statistics

Activity Summary Name	Logging Sub Category	Statistic ⁶	Description
client-ratelimited	Ratelimit	client-rate-limit	Number of times a client was rate limited.
domain-ratelimited	Ratelimit	domain-rate-limit	Number of times a domain was rate limited.
interval	Ratelimit-Domain	N/A	Length of data collection period.
num-ratelimited	Ratelimit-Domain	N/A	Total number of domains that were rate limited.
total-counted	Ratelimit-Domain	N/A	Total number of times a domain was rate limited.
not-counted	Ratelimit-Domain	N/A	Number of times the domain rate limiting table overflowed.
interval	Ratelimit-Client	N/A	Length of data collection period.
num-ratelimited	Ratelimit-Client	N/A	Total number of clients that were rate limited.
total-counted	Ratelimit-Client	N/A	Total number of times a client was rate limited.
not-counted	Ratelimit-Client	N/A	Number of times the client rate limiting table overflowed.

⁶ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Resolution Queue Statistics

The **resol-queue** activity-summary-settings logs statistics on the resolution queue.

Sample log message:

```
10/06/2021 10:22:44 cdns Activity Stats 0 22174 [Resolution-Queue] Sample since Wed Oct 6
10:21:44 2021: num-entries=number, user-queries=number, system-queries=number,
average-num-entries=number, max-num-entries=number, entries-overwritten=number,
exceeded-limit=number, replies-sent=number exceeded-max-target-count=number
```

Table 8: Resolution Queue Statistics

Activity Summary Name	Statistic ⁷	Description
num-entries	requestlist-total	Total number of queued requests waiting for recursive replies.
user-queries	requestlist-total-user	Total number of queued user requests waiting for recursive replies.
system-queries	requestlist-total-system	Total number of queued system requests waiting for recursive replies.
average-num-entries	requestlist-total-average	Average number of requests on the request list.
max-num-entries	requestlist-total-max	Maximum number of requests on the request list.
entries-overwritten	requestlist-total-overwritten	Number of requests on the request list that were overwritten by newer entries.
exceeded-limit	requestlist-total-exceeded	Number of requests dropped because the request list was full.
replies-sent	recursive-replies-total	Total number of query replies that were not found in the cache and required external resolution.
exceeded-max-target-count	exceeded-max-target-count	Number of queries that exceeded the maximum number of name servers glue lookups allowed.

⁷ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Responses Statistics

The **responses** activity-summary-settings logs statistics about query responses.

Sample log message:

```
10/06/2021 10:22:44 cdns Activity Stats 0 22175 [Responses] Sample since Wed Oct 6 10:21:44
2021: no-error=number, no-data=number, formerr=number, servfail=number, nxdomain=number,
notimp=number, refused=number, notauth=number, other-errors=number, secure=number,
unsecure=number, rrsset-unsecure=number, unwanted=number
```

Table 9: Responses Statistics

Activity Summary Name	Statistic ⁸	Description
no-error	answers-with-NOERROR	Number of answers from cache or recursion that result in rcode of NOERROR being returned to client.
nxdomain	answers-with-NXDOMAIN	Number of answers from cache or recursion that result in rcode of NXDOMAIN being returned to client.

Activity Summary Name	Statistic ⁸	Description
no-data	answers-with-NODATA	Number of answers that result in pseudo rcode of NODATA being returned to client.
other-errors	answers-with-other-errors	Number of answers that result in pseudo rcode of NODATA being returned to client.
secure	answers-secure	Number of answers that correctly validated.
unsecure	answers-unsecure	Number of answers that did not correctly validate.
rrset-unsecure	answers-rrset-unsecure	Number of RRSets marked as bogus by the validator.
unwanted	answers-unwanted	Number of replies that were unwanted or unsolicited. High values could indicate spoofing threat.
refused	answers-with-REFUSED	Number of answers from cache or recursion that result in rcode of REFUSED being returned to client.
servfail	answers-with-SERVFAIL	Number of answers from cache or recursion that result in rcode of SERVFAIL being returned to client.
formerr	answers-with-FORMERR	Number of answers from cache or recursion that result in rcode of FORMERR being returned to client.
notauth	answers-with-NOTAUTH	Number of answers from cache or recursion that result in rcode of NOTAUTH being returned to client.
notimp	answers-with-NOTIMP	Number of answers from cache or recursion that result in rcode of NOTIMP being returned to client.

⁸ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

System Statistics

The **system** activity-summary-settings logs statistics on system usage.

Sample log message:

```
10/26/2021 6:04:44 cdns tid: 0 Activity Stats 0 22375 [System] Current: conntrack-max=number,
conntrack-count=number, conntrack-usage=number
```

Table 10: System Statistics

Activity Summary Name	Description
contrack-max	Maximum number of connection tracking entries allowed.
contrack-count	Number of connection tracking entries currently in use.
contrack-usage	Percentage of connection tracking entries in use.

Top Names Statistics

The **top-names** activity-summary-settings logs the top names queried and hit count.

Sample log message:

```
10/26/2021 12:07:08 cdns Activity Stats 0 22371 [Top-Names] from 12:06:48 to 12:06:58;
interval=number, total-counted=number
```

Table 11: Top Names Statistics

Activity Summary Name	Statistic ⁹	Description
interval	N/A	Length of data collection period. It corresponds to the CDNS <i>top-names-max-age</i> setting, which controls how long it has to collect the top names for each log entry. It then lists a configurable number of top names (default 10) and the number of queries for those names.
total-counted	total-counted	Total number of queries counted in this collection period.

⁹ The statistics listed in this column are the server statistics displayed in the web UI and CLI. The REST API calls will have the statistic name camel-cased without dashes (that is, queries-total is queriesTotal in the REST API). Note that the activity summary and statistics are keyed off the same server data, but the activity-summary names are shortened to conserve space in the log message.

Specifying Top Names Settings

The *top-names* attribute specifies if top names data should be collected. When enabled, a snapshot of the cache hits for the top names that are queried is collected for each interval set by the *top-names-max-age* value. The list of top names that is reported with activity summary statistics is the most current snapshot.

You can specify the maximum age (based on last access time) of a queried name allowed in the list of top names by using the *top-names-max-age* attribute.



Note The *top-names-max-age* attribute has a default value of 60 seconds.

You can specify the maximum number of entries in the list of top names queried by using the *top-names-max-count* attribute. This limit is applied to the lists of top names that are logged or returned as part of activity summary.

Local Basic or Advanced Web UI

To enable Top Names, on the Edit Local CDNS Server tab, under the **Top Names Settings** section, enable the *top-names* attribute by selecting the **enabled** option, and then click **Save** to save the changes.

Top Names Statistics

The Top Names tab displays the relevant information with respect to top N domains and other important statistics attributes.

Local Basic or Advanced Web UI

- Step 1** From the **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page.
- Step 2** Select **CDNS** in the Manage Servers pane.
- Step 3** Click the **Top Names** tab available in the Local CDNS Server page.
-

CLI Commands

Use `cdns getStats top-names` to view the Top Names statistics.

Setting Prefetch Timing

Use the *Prefetch* attribute to set whether message cache elements should be prefetched before they expire to keep the cache up to date. Turning it **on** gives about 10 percent more traffic and load on the machine, but can increase the query performance for popular DNS names.

When *Prefetch* is enabled, records are assigned a prefetch time that is within 10 percent of the expiration time. As the server processes client queries and looks up the records, it checks the prefetch time. Once the record is within 10 percent of its expiration, the server will issue a query for the record to keep it from expiring.

Setting Cache TTLs

Time to Live (TTL) is the amount of time that any nameserver is allowed to cache data learned from other nameservers. Each record added to the cache arrives with some TTL value. When the TTL period expires, the server must discard the cached data and get new data from the authoritative nameservers the next time it sends a query. TTL attributes, *cache-min-ttl* and *cache-max-ttl* defines the minimum and maximum time Cisco Prime Network Registrar retains the cached information. These parameters limit the lifetime of records in the cache whose TTL values are very large.

Local Basic or Advanced Web UI

- Step 1** On the Edit Local CDNS Server tab, you can find:
- The Maximum Cache TTL (*cache-max-ttl*) attribute, set it to the desired value (the default value is 24 hours)
 - The Min Cache TTL (*cache-min-ttl*) attribute, set it to the desired value (the preset value is 0)

Step 2 Click **Save** to save the changes.

CLI Commands

Use `cdns set cache-max-ttl=value` to set the maximum Cache TTL value.

Use `cdns set cache-min-ttl =value` to set the minimum Cache TTL value.

Enabling Smart Caching

Whenever Authoritative DNS servers face an outage or are offline for other reasons, this could cause issues with being able to reach Internet services that are likely not impacted. Smart caching allows the Caching DNS server to continue to serve the expired data (last known answer) when it cannot reach the authoritative name servers. The Caching DNS server will still continue to contact the authoritative name servers and when the name servers are once again functional, the Caching DNS server will update its expired data.



Note Enabling Smart Cache (*smart-cache*) automatically enables prefetch.

Smart Cache Configuration Settings

In Cisco Prime Network Registrar, Caching DNS Smart Cache is not enabled by default. To use Smart Cache, the *smart-cache* attribute must be enabled at the Caching DNS server level.

When the Caching DNS server receives a query for data that has expired and if the *smart-cache* attribute is enabled, it will continue to respond with its expired cached data and increment the *smart-cache* counter under the **Query Details** section in the Statistics tab.



Note Smart Cache is available in Advanced mode and requires a Caching DNS server reload to take effect.

Table 12: Smart Cache Attributes

Attribute	Description
Smart Cache (<i>smart-cache</i>)	Specifies if the Caching DNS server should use Smart Caching. When <i>smart-cache</i> is enabled, the Caching DNS server continues to use its last best known answer when cached responses have expired and it cannot reach the authoritative name servers. The RRs in smart cache responses will have a 0 TTL. Smart Caching is useful to mitigate network outages and possible DDoS attacks that make the authoritative name servers unavailable. Enabling <i>smart-cache</i> automatically enables prefetch.
Smart Cache Expiration (<i>smart-cache-expiration</i>)	When <i>smart-cache</i> is enabled, specifies a time limit for responding with expired RRs. The default is 0, which allows the server to respond with expired answers as long as they remain in the cache.

Attribute	Description
Smart Cache Expiration Reset (<i>smart-cache-expiration-reset</i>)	When <i>smart-cache</i> is enabled and <i>smart-cache-expiration</i> is greater than 0, will reset the expiration time on active queries. This allows active queries to return expired answers, while allowing others to return SERVFAIL responses for a short period. Once the queries become active, will return expired answers. Default is disabled.
Prefetch (<i>prefetch</i>)	<p>Sets whether message cache elements should be prefetched before they expire to keep the cache up to date. Turning it on gives about 10 percent more traffic and load on the machine, but popular items do not expire from the cache.</p> <p>When <i>Prefetch</i> is enabled, records are assigned a prefetch time that is within 10 percent of the expiration time. As the server processes client queries and looks up the records, it checks the prefetch time. Once the record is within 10 percent of its expiration, the server will issue a query for the record in order to keep it from expiring.</p>



Note From Cisco Prime Network Registrar 10.1, the *Prefetch* attribute is available under the **Smart Cache** section and it is an Advanced mode feature.

Local Advanced Web UI

To enable Smart Cache, do the following:

- Step 1** From the **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page. Click **CDNS** on the Manage Servers pane.
- Step 2** On the Edit Local CDNS Server tab, under the **Smart Cache** section, enable the *smart-cache* attribute by selecting the **enabled** option.
- Step 3** Click **Save** to save the changes.

CLI Commands

Use **cdns enable smart-cache** to enable Smart Caching.

Use **cdns set smart-cache-expiration=*value*** to specify a time limit for responding with expired RRs, when *smart-cache* is enabled. For example:

```
nrcmd> cdns set smart-cache-expiration=5m
```

Use **cdns enable smart-cache-expiration-reset** to reset the expiration time on active queries, when *smart-cache* is enabled and *smart-cache-expiration* is greater than 0.

Defining Root Nameservers

Root nameservers know the addresses of the authoritative nameservers for all the top-level domains. When you first start a newly installed Cisco Prime Network Registrar Caching DNS server, it uses a set of preconfigured root servers, called root hints, as authorities to ask for the current root nameservers.

When Cisco Prime Network Registrar gets a response to a root server query, it caches it and refers to the root hint list. When the cache expires, the server repeats the process. The TTL on the official root server records is preconfigured and you can specify a different cache TTL value (see [Setting Cache TTLs, on page 14](#)).

As the configured servers are only hints, they do not need to be a complete set. You should periodically (every month to six months) look up the root servers to see if the information needs to be altered or augmented.

Local Basic or Advanced Web UI

On the Edit Local CDNS Server tab, under the **Root Name Servers** section, enter the domain name and IP address of each additional root nameserver, clicking **Add Root Namerserver** after each one, then click **Save**.

CLI Commands

Use `cdns addRootHint name addr [addr ...]` to add the name of a root server and the root name server address(es).

Dynamic Allocation of UDP Ports

The Caching DNS server uses a large number of UDP port numbers, by default approximately 60000 port numbers. These numbers are divided among the processing threads. The large number of port numbers reduce the risk of cache poisoning via Birthday Attacks. The Caching DNS server uses the default pool of UDP ports (2048) and the maximum allowable size of the default pool of UDP ports is 4096.

Currently, Cisco Prime Network Registrar uses the port range from 1024 to 65535. Based on the number of outstanding resolution queries, the Caching DNS server adjusts the pool size by adding or removing ports. The Caching DNS server allocates and releases the UDP ports dynamically when the server is running. If you reload the server, all the UDP ports are released and randomly picked again.

Cisco Prime Network Registrar uses the *outgoing-range-avoid* attribute that allows you to define ports or ranges of ports that will be excluded from use by the DNS server when sending queries.



Note You need to ensure that UDP ports needed by other applications are in the port exclusion list. Otherwise, these applications may not be able to bind to their port(s) if the DNS server is using the port.

Local Basic or Advanced Web UI

On the Edit Local CDNS Server tab, expand the **Additional Attributes** section to view various attributes and their values. For the *query-source-port-exclusion-list* attribute value, enter a range of ports that need to be excluded. Then, click **Modify Server**.

Setting Maximum Memory Cache Sizes

The maximum memory cache size property specifies how much memory space you want to reserve for the DNS in-memory cache. The larger the memory cache, the less frequently the Caching DNS server will need to re-resolve unexpired records.

Local Advanced Web UI

On the Edit Local CDNS Server tab, under the **Caching** section, set the desired value for the RRSets Cache Size (*rrset-cache-size*) attribute, then click **Save**. The default size is 1 GB.

To set the size of the message cache, use the Message Cache Size (*msg-cache-size*) attribute. The message cache stores query responses. The default size is 1 GB.

CLI Commands

- Use `cdns set rrset-cache-size` to set RRSets Cache Size.
- Use `cdns set msg-cache-size` to set Message Cache Size.

Specifying Resolver Settings

Glue record(s) is/are A record(s) for name server(s) that cannot be found through normal DNS processing because they are inside the zone they define. When the *harden-glue* attribute is enabled, the Caching DNS server will ignore glue records that are not within the zone that is queried. The *harden-glue* attribute is on by default.

Domain randomization allows a DNS server to send upstream queries for resolution with a randomly generated query name. A valid name server responds with the query name unchanged and therefore this technique can be used to ensure that the response was valid.

In certain occasions, attacker issues a request and then flood the server with fake responses in an attempt to poison the DNS server's cache with rogue data. Randomizing the case gives the server another level of protection against types of attacks.

Cisco Prime Network Registrar supports randomizing upstream queries, but there are some name servers that do not maintain the randomized case. Therefore, if you enable case randomization, you may block out valid name servers. The *randomize-query-case-exclusion* attribute allows you to create an exclusion list, so that you can continue to use case randomization, but exclude name servers that do not maintain the case but still respond with a valid answer.

Table 13: Resolver Settings Attributes

Attribute	Description
<i>harden-glue</i>	Specified if glue should only be trusted if it is within the servers authority.
<i>randomize-query-case</i>	Enables the use of 0x20-encoded random bits in the query to foil spoof attempts. This perturbs the lowercase and uppercase of query names sent to authority servers and checks if the reply still has the correct casing.

Attribute	Description
<i>randomize-query-case-exclusion</i>	Allows to create an exclusion list for randomization of upstream queries. This attribute will be used when <i>randomize-query-case</i> is enabled.

Configuring Case Randomization Exclusions

The *randomize-query-case-exclusion* attribute is available under the **Resolver Settings** section on the Manage DNS Caching Server page. The *randomize-query-case* is not enabled by default. To use randomize query case exclusion, the *randomize-query-case* attribute must be enabled at the Caching DNS server level.

Both *randomize-query-case* and *randomize-query-case-exclusion* attributes are available in the web UI in Advanced mode.

Local Advanced Web UI

-
- Step 1** From the **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page. Click **CDNS** on the Manage Servers pane.
- Step 2** On the Edit Local CDNS Server tab, under the **Resolver Settings** section:
- Enable the *randomize-query-case* attribute by selecting the **enabled** option.
 - In the *randomize-query-case-exclusion* field, enter the list of domains (comma separated) that you want to exclude from case randomization.
- Step 3** Click **Save** to save the changes.
-



Note You must reload the Caching DNS server for the changes to take effect.

CLI Commands

Use **cdns enable randomize-query-case** to enable the case randomization.

Use the **cdns set** and **cdns unset** commands to set or unset *randomize-query-case-exclusion*. For example:

```
nrcmd> cdns set randomize-query-case-exclusion="cisco.com"
nrcmd> cdns set randomize-query-case-exclusion="cisco.com, example.com"
nrcmd> cdns unset randomize-query-case-exclusion
```

Specifying Network Settings

The *listen-ip-version* attribute lets you to choose the IP packets to accept and issue. You can check IPv4, IPv6, both, or none. The *listen-protocol* attribute lets you to choose the packet protocol to answer and issue. You can check UDP, TCP, both, or none.

Specifying Advanced Settings

The *minimal-responses* attribute controls whether the DNS Caching server omits or includes records from the authority and data sections of query responses when these records are not required. Enabling this attribute may improve query performance such as when the DNS server is configured as a caching server.

The *remote-ns-host-ttl* attribute sets TTL for entries in the remote name server cache. The remote name server cache contains roundtrip timing (RTT), lameness and EDNS support information. Once an entry expires, it is removed from the remote name server cache and the next time the server is contacted a new entry will be added.

Note that RTT is used to decide which name server to query. If a timeout occurs, the RTT value of that server is doubled. If a server starts to become unresponsive, a probing scheme is applied in which a few queries are selected to probe the IP address. If that fails, the name server is blocked for 15 minutes (*remote-ns-host-ttl*) and re-probed with one query after that. Therefore, it may be necessary to decrease the *remote-ns-host-ttl* to allow probing more frequently. The remote name server cache is not flushed after a CDNS server reload, but can be flushed using the **cdns execute flush-ns-cache** command.

The *remote-ns-cache-numhosts* attribute lets you to set the number of hosts for which information is cached.

Enabling Round-Robin

A query might return multiple A records for a nameserver. To compensate for most DNS clients starting with, and limiting their use to, the first record in the list, you can enable round-robin to share the load. This method ensures that successive clients resolving the same name will connect to different addresses on a revolving basis. The DNS server then rearranges the order of the records each time it is queried. It is a method of load sharing, rather than load balancing, which is based on the actual load on the server.

Local Advanced Web UI

On the Edit Local CDNS Server tab, under the **Advanced Settings** section, find the *round-robin* attribute.

CLI Commands

Use **cdns get round-robin** to see if round-robin is enabled (it is by default). If not, use **cdns enable round-robin**.

Flushing Caching DNS Cache

Cisco Prime Network Registrar cache flushing function lets you remove all or a portion of cached data in the memory cache of the server.

Local Basic or Advanced Web UI

Step 1 From the **Deploy** menu, choose **CDNS Server** under the **DNS** submenu to open the Manage DNS Caching Server page.

Step 2 On the Manage DNS Caching Server page, click the **Commands** button to open the CDNS Command dialog box. There will be two types of cache flushing commands.

- **Flush the CDNS cache**—Allows you to either flush all cache entries for a particular zone or the entire cache if no zone is provided. To remove all data for a specific zone, enter the zone name in the Zone field. To clear the whole cache, leave the Zone field empty.
- **Flush Resource Record**—Allows you to flush an RR name or an RRSet when the type field is specified.
 - Remove common RR types (A, AAAA, NS, SOA, CNAME, DNAME, MX, PTR, SRV, NAPTR, and TXT) from a specific domain—Enter the required RR name as the FQDN for the Flush Resource Record command and leave the RR type field empty.
 - Remove a specified RR type for a domain—Specify the domain in the FQDN field, and the RR type in the RR type field.

Note When no type is specified, the server flushes types A, AAAA, NS, SOA, CNAME, DNAME, MX, PTR, SRV, TXT, and NAPTR.

CLI Commands

- To remove all cached entries at or below a given domain, use **cdns flushCache** *domain*. If no domain is given, it flushes all RRs in the cache.
- To flush RRs from the cache associated with the given RR name, use **cdns flushName** *name type*. When type is provided, it flushes all entries with the given name and type. If no type is provided, it flushes types A, AAAA, NS, SOA, CNAME, DNAME, MX, PTR, SRV, TXT, and NAPTR.

Detecting and Preventing DNS Cache Poisoning

Cisco Prime Network Registrar enhances the Caching DNS server performance to address the CDNS related issues such as DNS cache poisoning attacks (CSCsq01298), as addressed in a Cisco Product Security Incident Response Team (PSIRT) document number PSIRT-107064 with Advisory ID cisco-sa-20080708-dns, available at:

<http://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20080708-dns>

DNS Cache Poisoning Attacks

A cache poisoning attack can change an existing entry in the DNS cache as well as insert a new invalid record into the DNS cache. This attack causes a hostname to point to the wrong IP address. For example, let us say that `www.example.com` is mapped to the IP address `192.168.0.1`, and this mapping is present in the cache of a DNS server. An attacker can poison the DNS cache and map `www.example.com` to `10.0.0.1`. If this happens, if you try to visit `www.example.com`, you will end up contacting the wrong web server.

A DNS server that uses a single static port for receiving responses to forwarded queries are susceptible to malicious clients sending forged responses.

The DNS transaction ID and source port number used to validate DNS responses are not sufficiently randomized and can easily be predicted, which allows an attacker to create forged responses to DNS queries. The DNS server will consider such responses as valid.

Handling DNS Cache Poisoning Attacks

To reduce the susceptibility to the DNS cache poisoning attack, the DNS server randomizes the UDP source ports used for forwarded queries. Also, a resolver implementation must match responses to the following attributes of the query:

- Remote address
- Local address
- Query port
- Query ID
- Question name (not case-sensitive)
- Question class and type, before applying DNS trustworthiness rules (see [RFC2181], section 5.4.1)



Note The response source IP address must match the query's destination IP address and the response destination IP address must match the query's source IP address. A mismatch must be considered as format error, and the response is invalid.

Resolver implementations must:

- Use an unpredictable source port for outgoing queries from a range (either 53, or > 1024) of available ports that is as large as possible and practicable.
- Use multiple different source ports simultaneously in case of multiple outstanding queries.
- Use an unpredictable query ID for outgoing queries, utilizing the full range available (0 to 65535). By default, CDNS uses about 60000 port numbers.

The Expert mode Caching DNS server attribute *randomize-query-case*, when enabled, specifies that when sending a recursive query, the query name is pseudo-randomly camel-cased and the response is checked to see if this camel-casing is unchanged. If *randomize-query-case* is enabled and the casing has changed, then the response is discarded. The *randomize-query-case* is disabled by default, disabling this feature.

Local Basic or Advanced Web UI

The Caching DNS server statistics appears on the Statistics tab of the Manage DNS Caching Server page. The Statistics displays the *answers-unwanted* values. You can refresh the DNS Caching Server Statistics by clicking the **Refresh Server Statistics** icon at the top of the statistics table.

Handling Unresponsive Nameservers

When trying to resolve query requests, Caching DNS servers may encounter unresponsive nameservers. A nameserver may be unresponsive to queries or respond late. This affects the performance of the local DNS server and remote nameservers.

Using Cisco Prime Network Registrar, you can resolve these problems by barring unresponsive nameservers. You can configure a global ACL of unresponsive nameservers that are to be barred, using the *acl-do-not-query* attribute.

When Cisco Prime Network Registrar receives a list of remote nameservers to transmit a DNS query request to, it checks for the nameservers listed in the *acl-do-not-query* list and removes them from this list. Conversely, all incoming DNS requests from clients or other nameservers are also filtered against *acl-blacklist*.

Use the *acl-query* attribute to specify which clients are allowed to query the server. By default, any client is allowed to query the server. A client that is not in this list will receive a reply with status REFUSED. Clients on the *acl-blacklist* do not get any response whatsoever.

Local Advanced Web UI

On the Edit Local CDNS Server tab, expand the **Query Access Control** section to view the various attributes and their values. For the Do Not Query (*acl-do-not-query*) attribute, enter the value (for example, 10.77.240.73). Then, click **Save**.

Running DNS Caching Server Commands

Access the DNS Caching server commands using the Commands button. Clicking the **Commands** button opens the CDNS Commands dialog box in the local web UI. Each command has its own Run icon (click it, then close the dialog box):

- **Flush the CDNS cache**— This command allows you to flush either all RRs or RRs for a particular zone from the in-memory cache. See [Flushing Caching DNS Cache, on page 20](#).
- **Flush Resource Record**— This command that lets you specify an RR name and optionally a type to remove from the in-memory cache.



Note To remove all the entries from the in-memory cache, you need to reload the Caching DNS server.



Note If you find a server error, investigate the server log file for a configuration error, correct the error, return to this page, and refresh the page.
