APPENDIX E

CLI Commands

This appendix covers the following topics:

- Configuring Port Channel, page E-1
- Configuring Last-Resort Routing, page E-6
- Configuring Standby Interfaces, page E-6
- Other CLI Commands, page E-10

Configuring Port Channel

To configure an EtherChannel, you use the **PortChannel** interface configuration command. Port Channel, also known as EtherChannel, supports the grouping of up to four same-speed network interfaces into one virtual interface. This grouping capability allows the setting or removing of a virtual interface that consists of two, three, or four Gigabit Ethernet interfaces. EtherChannel also provides interoperability with Cisco routers, switches, and other networking devices or hosts supporting EtherChannel; load balancing; and automatic failure detection and recovery based on each interface's current link status.

Note In order to achieve the best throughput, we recommend you configure a port channel for the four Gigabit Ethernet ports on the line card.

Redundant Dedicated Management Ports

On a CDE200 configured as an SE or SR there are six Gigabit Ethernet ports. All of the ports can be used for delivery traffic such as RTSP, as well as system management traffic to communicate with other CDS devices such as the CDSM. In order to prevent all the bandwidth being used by delivery traffic, a dedicated management port setup is often recommended.

In case of physical failure on a single port, channel bonding configuration of multiple Gigabit Ethernet ports is also recommended for both delivery traffic and management traffic.

Note A port channel configured with a default gateway is only for delivery traffic. Delivery traffic places highest bandwidth demand on the CDS network. A port channel configured as the primary interface carries delivery traffic.

Note If an EtherChannel (also known as port channel) is used between the upstream router or switch and the SE for streaming real-time data, the EtherChannel load balance algorithms on the upstream switch or router and the SE should be configured as "src-ip" and "dst-ip" respectively. Using this configuration ensures session stickiness and general balanced load distribution based on clients' IP addresses. Also, distribute your client IP address space across multiple subnets so that the load balancing algorithm is effective in spreading the traffic among multiple ports.

Typically, two Gigabit Ethernet ports on the motherboard (GigabitEthernet 1/0 and 2/0) can be bundled for the management port channel, and four Gigabit Ethernet ports (GigabitEthernet 3/0, 4/0, 5/0, and 6/0) on the network interface card (NIC) can be bundled for the traffic port channel.

To configure redundant dedicated management ports using the CLI, do the following:

Step 1 Configure two port channels., one with four Gigabit Ethernet ports (portchannel 1) and one with two Gigabit Ethernet ports (portchannel 2) for application traffic and management traffic respectively.

```
SE(config)# interface GigabitEthernet 1/0
SE(config-if)# channel-group 2
SE(config-if)# exit
SE(config)# interface GigabitEthernet 2/0
SE(config-if)# channel-group 2
SE(config-if)# exit
SE(config)# interface GigabitEthernet 3/0
SE(config-if)# channel-group 1
SE(config-if)# exit
SE(config)# interface GigabitEthernet 4/0
SE(config-if)# channel-group 1
SE(config-if)# exit
SE(config)# interface GigabitEthernet 5/0
SE(config-if)# channel-group 1
SE(config-if)# exit
SE(config)# interface GigabitEthernet 6/0
SE(config-if)# channel-group 1
SE(config-if)# exit
```

Note The port channel carrying delivery traffic should always be configured as channel-group 1 and set as the primary interface.

Step 2 Configure two different subnets for both port channels.

```
SE(config)# interface PortChannel 1
SE(config-if)# ip address 3.1.7.73 255.255.255.0
SE(config-if)# exit
SE(config)# interface PortChannel 2
SE(config-if)# ip address 3.1.8.200 255.255.255.0
SE(config-if)# exit
```

Step 3 Configure the delivery port channel as the primary interface.

```
SE(config)# primary-interface PortChannel 1
```

Step 4 Configure a default gateway for the delivery traffic.

```
SE(config)# ip default-gateway 3.1.7.1
```

Step 5 Set the load balancing algorithm to the destination IP address.

```
SE(config)# port-channel load-balance dst-ip
```

Step 6 Set the client and server TCP compliance to the RFC 1323 standard.

```
SE(config)# tcp client-satellite
SE(config)# tcp server-satellite
```

Step 7 Configure a static route to the CDSM (4.0.5.5) to specify that all management traffic will go through this interface.

```
SE(config)# ip route 4.0.5.5 255.255.255.255 3.1.8.1
```

Step 8 Configure the port channel and VLANs on the switch that the SE is directly connected to.

```
SW3750(config)# interface Port-channel1
SW3750(config-if)# switchport access vlan 201
SW3750(config-if)# exit
SW3750(config)# interface Port-channel2
SW3750(config-if)# switchport access vlan 202
SW3750(config-if)# exit
SW3750(config)# interface GigabitEthernet1/0/1
SW3750(config-if)# description Connected to portchannel2
SW3750(config-if)# switchport access vlan 202
SW3750(config-if)# switchport mode access
SW3750(config-if)# exit
SW3750(config)# interface GigabitEthernet1/0/2
SW3750(config-if)# description Connected to portchannel2
SW3750(config-if)# switchport access vlan 202
SW3750(config-if)# switchport mode access
SW3750(config-if)# exit
SW3750(config)# interface GigabitEthernet1/0/3
SW3750(config-if)# description connected to portchannel1
SW3750(config-if)# switchport access vlan 201
SW3750(config-if)# switchport mode access
SW3750(config-if)# exit
SW3750(config)# interface GigabitEthernet1/0/4
SW3750(config)-if# description connected to portchannel1
SW3750(config-if)# switchport access vlan 201
SW3750(config-if)# switchport mode access
SW3750(config-if)# exit
SW3750(config)# interface GigabitEthernet1/0/5
SW3750(config-if)# description connected to portchannel1
SW3750(config-if)# switchport access vlan 201
SW3750(config-if)# switchport mode access
SW3750(config-if)# exit
SW3750(config)# interface GigabitEthernet1/0/6
SW3750(config-if)# description connected to portchannel1
SW3750(config-if)# switchport access vlan 201
SW3750(config-if)# switchport mode access
SW3750(config-if)# exit
SW3750(config)# interface Vlan201
SW3750(config-if)# ip address 3.1.7.1 255.255.255.0
SW3750(config-if)# exit
SW3750(config)# interface Vlan202
SW3750(config-if)# ip address 3.1.8.1 255.255.255.0
SW3750(config-if)# exit
```

Step 9 Set the load balancing algorithm to the source IP address.

```
SW3750(config)# port-channel load-balance src-ip
```

To verify the setup before application traffic is sent, use the following:

```
SE# clear statistics all
SE# show interface portChannel 1
Interface PortChannel 1 (2 physical interface(s)):
GigabitEthernet 3/0 (active)
GigabitEthernet 4/0 (active)
GigabitEthernet 5/0 (active)
GigabitEthernet 6/0 (active)
---------------------
Type:Ethernet
Ethernet address:00:04:23:D8:86:02
Internet address:3.1.7.73
Broadcast address:3.1.7.255
Netmask:255.255.255.0
Maximum Transfer Unit Size:1500
Metric:1
Packets Received: 28
Input Errors: 0
Input Packets Dropped: 0
Input Packets Overruns: 0
Input Packets Frames: 0
Packet Sent: 40
Output Errors: 0
Output Packets Dropped: 0
Output Packets Overruns: 0
Output Packets Carrier: 0
Output Queue Length:0
Collisions: 0
Flags:UP BROADCAST RUNNING MASTER MULTICAST

SE# show interface portChannel 2
Interface PortChannel 2 (4 physical interface(s)):
GigabitEthernet 1/0 (active)
GigabitEthernet 2/0 (active)
---------------------
Type:Ethernet
Ethernet address:00:30:48:33:01:26
Internet address:3.1.8.200
Broadcast address:3.1.8.255
Netmask:255.255.255.0
Maximum Transfer Unit Size:1500
Metric:1
Packets Received: 6
Input Errors: 0
Input Packets Dropped: 0
Input Packets Overruns: 0
Input Packets Frames: 0
Packet Sent: 0
Output Errors: 0
Output Packets Dropped: 0
Output Packets Overruns: 0
Output Packets Carrier: 0
Output Queue Length:0
Collisions: 0
Flags:UP BROADCAST RUNNING MASTER MULTICAST
```

To verify the setup after application traffic is sent, use the following:

```
SE# show interface portChannel 1
Interface PortChannel 1 (4 physical interface(s)):
GigabitEthernet 3/0 (active)
GigabitEthernet 4/0 (active)
GigabitEthernet 5/0 (active)
GigabitEthernet 6/0 (active)
--------------------
Type:Ethernet
Ethernet address:00:04:23:D8:86:02
Internet address:3.1.7.73
Broadcast address:3.1.7.255
Netmask:255.255.255.0
Maximum Transfer Unit Size:1500
Metric:1
Packets Received: 1875
Input Errors: 0
Input Packets Dropped: 0
Input Packets Overruns: 0
Input Packets Frames: 0
Packet Sent: 5221
Output Errors: 0
Output Packets Dropped: 0
Output Packets Overruns: 0
Output Packets Carrier: 0
Output Queue Length:0
Collisions: 0
Flags:UP BROADCAST RUNNING MASTER MULTICAST

SE# show interface portChannel 2
Interface PortChannel 2 (2 physical interface(s)):
GigabitEthernet 1/0 (active)
GigabitEthernet 2/0 (active)
---------------------
Type:Ethernet
Ethernet address:00:30:48:33:01:26
Internet address:3.1.8.200
Broadcast address:3.1.8.255
Netmask:255.255.255.0
Maximum Transfer Unit Size:1500
Metric:1
Packets Received: 21
Input Errors: 0
Input Packets Dropped: 0
Input Packets Overruns: 0
Input Packets Frames: 0
Packet Sent: 0
Output Errors: 0
Output Packets Dropped: 0
Output Packets Overruns: 0
Output Packets Carrier: 0
Output Queue Length:0
Collisions: 0
Flags:UP BROADCAST RUNNING MASTER MULTICAST

SE# show statistics wmt streamstat
Detailed Stream Statistics
==========================
Incoming Streams:
=================
Stream-Id Type Source State Bytes_Recd Duration Bandwidth Url_Requested
Outgoing Streams:
=================
Client-IP Type Transport Source State Pkts_sent Bytes_sent Duration BW Server-IP Filename
2.224.22.50 VOD RTSPT LOCAL Play 5221 7312466 42 502 local 100kbps.wmv

SE# show cms info
CDN information :
Model = SE200
Node Id = 424
```

```
Device Mode = se
Current CDSM address = 4.0.5.5
Registered with CDSM = 4.0.5.5
Status = Online
Time of last config-sync = Thu May 3 16:56:54 2007
CMS services information :
Service cms_se is running
```

In the Devices Table page on the CDSM (**Devices > Devices**), the SE or SR status should be "Online." The IP address for the device always shows the IP address of the primary interface.

Configuring Last-Resort Routing

Last-resort routing is applicable when load-based routing is enabled and all Service Engines have exceeded their thresholds or all Service Engines in the domain are offline. The Service Router can redirect requests to a configurable alternate domain when all Service Engines serving a client network region are overloaded.

Note If the last-resort domain is not configured and the Service Engine thresholds are exceeded, requests are redirected to the origin server.

To configure last-resort routing use the **service-router** global configuration command, where *domain* is the service routing domain name, and *alternate* is where to route requests.

```
service-router lastresort domain domain alternate alternate
```

In the example below, srfqdn.cisco.com is the service routing domain name, and www.cisco.com is the alternate domain name.

```
SR(config)# service-router ?
    lastresort Configure lastresort domain
    leastloaded Enable Load Based Routing
    location-based-routing Configure location based routing
SR(config)# service-router lastresort ?
    domain Configure domain
SR(config)# service-router lastresort domain srfqdn.cisco.com ?
    alternate Configure alternate domain
SR(config)# service-router lastresort domain srfqdn.cisco.com alternate ?
    WORD Configure alternate domain name
SR(config)# service-router lastresort domain srfqdn.cisco.com alternate www.cisco.com ?
    <cr>
SE(config)# service-router lastresort domain srfqdn.cisco.com alternate www.cisco.com
```

Configuring Standby Interfaces

You can configure one ore more interfaces to act as a backup interface (a standby interface) for another interface on a Service Engine. This feature is called standby interface support. Standby groups, which are logical groups of interfaces, are used to implement this feature. When an active network interface fails (because of cable trouble, Layer 2 switch failure, high error count, or other failures) and that interface is part of a standby group, a standby interface can become active and take the load off the failed interface.

A standby group must have at least two interfaces. Interfaces that are part of a standby group are called member interfaces. After you create a standby group, you define which interfaces should be assigned to this logical group. As part of defining the member interfaces, you specify the priority of each member interface in a standby group. The member interface with the highest assigned priority is the active interface for that particular standby group. If the active interface fails, the operational member interface with the next highest priority in the standby group comes up, and so forth. If all member interfaces of a particular standby group are down and then one of the member interfaces comes up, the CDS software detects this situation and brings up the standby group on the member interface that just came up.

The failure or failover of member interfaces within a standby group triggers alarms and traps (if alarms and traps are enabled on the Service Engine). Alarms are sent out when failover occurs between member interfaces in a standby group. Specifically, minor alarms are sent out when member interfaces fail, and these alarms are cleared automatically when the interface failover has been successfully completed. Major alarms are sent out if the standby group goes down (no member interface in a standby group can be brought up).

Note A physical interface can belong to more than one standby group, and a single interface can act as a standby interface for more than one standby group.

To configure standby interfaces, interfaces are logically assigned to standby groups. The following rules define the standby group relationships:

- Each standby group is assigned a unique standby IP address, shared by all member interfaces of the standby group. The IP address of the standby group is shared among the member interfaces; however, only the active interface of the standby group uses this shared IP address at any one time. This shared IP address is configured as an alias on the active interface.
- The duplex and speed settings of the member interfaces can be configured for better reliability.
- If a physical interface is a member of a port-channel group, it cannot join a standby group. If a physical interface is a member of a standby group, it cannot join a port-channel group.
- The maximum number of standby groups on a Service Engine is four.

Note Interface IP addresses and standby group IP addresses must be on different subnets to ensure reliable operation. You can use dummy IP addresses in the private address space to serve as interface primary IP addresses, and use the real Service Engine IP address to serve as the standby group IP address in a different subnet to satisfy this requirement. When dummy IP addresses are used, these interface IP addresses serve only as substitutes to bring up the interface. For example, the Service Engine interface requires an IP address on an interface for initialization. Make sure to configure the interface default gateway using the **ip default-gateway** global configuration command instead of the **ip route** command.

- Each interface in a standby group is assigned a priority. The operational interface with the highest priority in a standby group is the active interface. Only the active interface uses the group IP address.
- The priority of an interface in a standby group can be changed at run time. The member interface that has the highest priority after this change becomes the new active interface (the default action is to preempt the currently active interface if an interface with higher priority exists).
- The maximum number of errors allowed on the active interface before the interface is shut down and the standby is brought up is configured with the **errors** option, which is disabled by default.

Tip If an interface belongs to more than one standby group, you can configure the interface with a different priority in each standby group for better load balancing. For example, interfaces FE 0/0 and FE 0/1 are both in standby group 1 and in standby group 2. If you configure FE 0/0 with the highest priority in standby group 1 and configure FE 0/1 with the highest priority in standby group 2, standby group 1 will use FE 0/0 as the active interface, while standby group 2 will use FE 0/1 as the active interface. This configuration allows each interface to back up the other one, if one of them fails.

Use the **interface standby** global configuration command to create standby groups on Service Engines.

Note Unlike port channels, standby groups do not support IP ACLs at a group level. However, you can configure a member interface of a standby group to support an IP ACL at the interface level. For example, you can individually configure the two member interfaces of Standby Group 1 (the Fast Ethernet slot 0/port 0 interface and the Fast Ethernet slot 0/port 1 interface) to support an IP ACL named ACL1 but you cannot configure the Standby Group 1 to support ACL1.

To configure an interface to be a backup for another interface, use the **standby** interface configuration command. To restore the default configuration of the interface, use the **no** form of this command.

standby *group_number* {**description** *text* | **errors** *max-errors* | **ip** *ip-address netmask* | **priority** *piority_level* | **shutdown**}

no standby *group_number* {**description** *text* | **errors** *max-errors* | **ip** *ip-address netmask* | **priority** *priority_level* | **shutdown**}

Syntax Description

group_number	Standby group number (1–4).
description	(Optional) Sets the description for the specified interface.
text	Description for the specified interface. The maximum length of the description text is 240 characters.
errors	Sets the maximum number of errors allowed on the active interface before the interface is shut down and the standby interface is brought up. This option is disabled by default.
max-errors	Maximum number of errors (o-4294967295).
ip	Sets the IP address for the specified standby group (Standby Group 1, 2, 3, or 4).
ip-address	IP address of the specified standby group (Standby Group 1, 2, 3, or 4). The group IP address and netmask of a standby group must be configured on all of the member interfaces.
netmask	Netmask of the specified standby group (Standby Group 1, 2, 3, or 4).
priority	Sets the priority of the member interface within a standby group. The priority of a member interface can be changed at run time. The member interface that has the highest priority after this change becomes the new active interface (the default action is to preempt the currently active interface if an interface with higher priority exists).

priority_level	Each member interface is assigned a priority number. The member interface with the highest priority number is the active interface for that standby group. Only the active interface uses the group IP address. If the **priority** option is specified without a priority number, the default value of 100 is used.
shutdown	(Optional) Shuts down the specified standby group (Standby Group 1, 2, 3, or 4). You can shut down a standby group even if you have not configured a group IP address of the standby group. **Note** When a standby group is shut down, all of the alarms previously raised by this standby group are cleared.

Examples

The following example configures three Fast Ethernet interfaces to be part of the same standby group, with interface 3/0 as the active interface:

```
Console(config-if)# interface fastEthernet 3/0 standby 1 ip 172.16.10.10 255.255.254.0
Console(config-if)# interface fastEthernet 3/1 standby 1 ip 172.16.10.10 255.255.254.0
Console(config-if)# interface fastEthernet 3/2 standby 1 ip 172.16.10.10 255.255.254.0
Console(config-if)# interface fastEthernet 3/0 standby 1 priority 300
Console(config-if)# interface fastEthernet 3/1 standby 1 priority 200
Console(config-if)# interface fastEthernet 3/2 standby 1 priority 100
Console(config-if)# interface fastEthernet 3/0 standby 1 errors 10000
Console(config-if)# interface fastEthernet 3/1 standby 1 errors 10000
Console(config-if)# interface fastEthernet 3/2 standby 1 errors 10000
```

The following example displays information about the standby group configuration by entering the **show standby** EXEC command. In the following sample command output, one standby group (Standby Group 1) is configured on this Service Engine. The command output also shows which member interface is the active interface. In this case, the active interface is the Fast Ethernet slot 3/port 0 interface.

```
ServiceEngine# show standby
Standby Group:1
IP address: 172.16.10.10, netmask: 255.255.254.0
Maximum errors allowed on the active interface: 10000
        Member interfaces:
                FastEthernet 3/0 priority: 300
                FastEthernet 3/1 priority: 200
                FastEthernet 3/2 priority: 100

        Active interface: FastEthernet 3/0
```

Note To display information about a specific standby group configuration, enter the **show interface standby** *group_number* EXEC command.

The following example creates a standby group, Standby Group 1:

```
ServiceEngine# configure
ServiceEngine(config)# interface standby 1
ServiceEngine(config-if)#
```

The following example assigns a group IP address of 10.10.10.10 and a netmask of 255.0.0.0 to Standby Group 1.

```
ServiceEngine(config-if)# ip address 10.10.10.10 255.0.0.0
ServiceEngine(config-if)# errors 500
```

The following example shows how to add two Fast Ethernet interfaces to Standby Group 1 and then assign each of these member interfaces a priority within the group:

1. Add a Fast Ethernet interface (slot 0/port 0) to Standby Group 1 and assign a priority of 150.

```
ServiceEngine(config)# interface FastEthernet 0/0
ServiceEngine(config-if)# standby 1 priority 150
```

2. Add a second Fast Ethernet interface (slot 0/port 1) to Standby Group 1 and assign a priority of 100.

(the default value):

```
ServiceEngine(config)# interface FastEthernet 0/1
ServiceEngine(config-if)# standby 1
ServiceEngine(config-if)# exit
ServiceEngine(config)#
```

Because Fast Ethernet 0/0 is assigned the highest priority (a priority number of 150) of all the member interfaces in the group, it will be chosen as the active interface for the group if it can be brought up.

The following example removes the Fast Ethernet slot 0/port 1 interface) from Standby Group 1 using the **no** form of the **standby** command:

```
ServiceEngine(config)# interface FastEthernet 0/1
ServiceEngine(config-if)# no standby 1
ServiceEngine(config-if)# exit
ServiceEngine(config)#
```

The following example shows how to shut down Standby Group 1. When a standby group is shut down, all of the alarms previously raised by this standby group are cleared.

```
ServiceEngine(config)# interface standby 1
ServiceEngine(config-if)# shutdown
ServiceEngine(config)# exit
```

The following example shows how to tear down Standby Group 1:

```
ServiceEngine(config)# interface standby 1
ServiceEngine(config-if)# no ip address 10.10.10.10 255.0.0.0
Please remove member interface(s) from this standby group first.
ServiceEngine(config)# interface GigabitEthernet 2/0
ServiceEngine(config-if)# no standby 1
ServiceEngine(config-if)# exit
ServiceEngine(config)# interface standby 1
ServiceEngine(config-if)# no ip address 10.10.10.10 255.0.0.0
ServiceEngine(config-if)# exit
ServiceEngine(config)# no interface standby 1
ServiceEngine(config)# exit
```

Other CLI Commands

This section lists other CLI commands that may be useful.

CDNFS cleanup Command

To manage the CDS network file system (cdnfs), use the **cdnfs** EXEC command.

cdnfs {**browse** | **cleanup** {**info** | **start** | **stop**} | **delete-unused-ecdnfs-files**}

There are no default behavior or values.

Syntax Description

browse	Browses the cdnfs directories and files.
cleanup	Cleans up the unwanted entries in the cdnfs.
info	Summarizes the information about unwanted entries without starting the cleanup process.
start	Starts the cleanup of unwanted entries in the cdnfs.
stop	Stops the cleanup of unwanted entries in the cdnfs.

The CDNFS cleanup command is an EXEC command.

The CDS network file systems (cdnfs) stores the prefetched CDS network content by all supported protocols. The **cdnfs cleanup** command cleans up the content of deleted channels from the acquisition distribution database. In certain cases, the Content Acquirer is not notified by the Centralized System (CMS) about deleted delivery services, and it fails to clear all unified name space (UNS) cases, the **cdnfs cleanup** EXEC command can be used to clean up all UNS content associated with deleted delivery services.

Note You can use **cdnfs cleanup start** to clean up the orphan content. Orphan content is content that is not associated with any delivery service to which a Service Engine is subscribed.

The **cdnfs browse** command is an interactive command and has the following interactive commands used to view the CDS network files and directories:

```
SE# cdnfs browse
------ CDNFS interactive browsing ------
dir, ls: list directory contents
cd,chdir: change current working directory
info: display attributes of a file
more: page through a file
cat: display a file
exit,quit: quit CDNFS browse shell
```

The **cdnfs cleanup** command synchronizes the state of the acquisition and distribution database with the content stored on the cdnfs. You should use this command after replacing a failed disk drive.

Disk Commands

The **disk** commands in the EXEC mode allow you to perform disk configuration and maintenance commands.

- **disk mark** *diskname* {**bad** | **good**}
- **disk recover-system-volumes**
- **disk reformat** *diskname*
- **disk unuse** *diskname*

Syntax Description

mark	Marks a disk drive as good or bad.
diskname	Name of the disk to be added (disk01, disk02, and so on.)
recover-system-volumes	Recovers the system disk.

reformat	Performs a low-level format of the SCSI, IDE, or SATA disks and remaps disk errors.
diskname	Name of the disk to be added (disk01, disk02, and so on.)
unuse	Stops applications from using a disk drive.
diskname	Name of the disk to be added (disk01, disk02, and so on.)

The **disk** commands are EXEC commands.

The **disk reformat** *diskname* EXEC command performs a low-level format of the SCSI, IDE, or SATA disks. This command erases all of the content on the disk.

If a disk drive continues to report a failure after you have used the **disk reformat** command, you must replace the disk drive.

Caution Be careful when using the **disk reformat** *diskname* command because this command causes all content on the specified disk to be deleted.

The **disk unuse** command stops and restarts all applications that are currently using the specified disk drive (for example, disk02 [/local/local2] or disk03), unmounts, and deletes all the partitions on the specified disk. Using this command unmounts all file systems.

Note For information about replacing a disk, see the "Disk Maintenance" section on page 8-26.

cache content Command

To set the number of cache entries in the CDS network file system (cdnfs), use the **cache content** global configuration command.

cache content max-cached-entries *1-10000000*

Syntax Description

content	Browses the cdnfs directories and files.
max-cached-entries	Cleans up the unwanted entries in the cdnfs.
1-10000000	Summarizes the information about unwanted entries without starting the cleanup process.

Maximum cache entries is set to 3000000 by default.

The CDS, by default, allows a maximum of three million cached entries, regardless of the amount of space available in CDNFS. Use this command to restrict the maximum number of contents in the CDS.