Configuring WCCP Version 2 Services

This chapter describes how to configure the Catalyst 4500 Series Switches to redirect traffic to content engines (web caches) using Web Cache Communication Protocol (WCCP) Version 2

Note Throughout this chapter, WCCP refers to WCCP Version 2. Version 1 is *not* supported.

This chapter consists of these sections:

Note The tasks in this chapter assume that you have already configured content engines on your network. For specific information on hardware and network planning associated with Cisco Content Engines and WCCP, see the product literature and documentation links available on Cisco.com:

http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf018_ps1835_TSD_Products_Configuration_Guide_Chapter.html
and
http://www.cisco.com/en/US/tech/tk122/tk717/tsd_technology_support_protocol_home.html

Understanding WCCP

These sections describe WCCP:

Overview

The Cisco IOS WCCP feature allows use of Cisco Content Engines (or other content engines running WCCP) to localize web traffic patterns in the network, enabling content requests to be fulfilled locally. Traffic localization reduces transmission costs and download time.

WCCP enables Cisco IOS routing platforms to transparently redirect content requests. The main benefit of transparent redirection of HTTP/non-HTTP requests is that users need not configure their browsers to use a web proxy. Instead, they can use the target URL to request content, and have their requests automatically redirected to a content engine. The word “transparent” in this case means that the end user does not know that a requested file (such as a web page) came from the content engine instead of from the originally specified server.

When a content engine receives a request, it attempts to service it from its own local content. If the requested information is not present, the content engine issues its own request to the originally targeted server to get the required information. When the content engine retrieves the requested information, it forwards it to the requesting client and caches it to fulfill future requests, thus maximizing download performance and substantially reducing transmission costs.

WCCP enables a series of content engines, called a *content engine cluster*, to provide content to a router or multiple routers. Network administrators can easily scale their content engines to handle heavy traffic loads using these clustering capabilities. Cisco clustering technology enables each content member to work in parallel, resulting in linear scalability. Clustering content engines greatly improves the scalability, redundancy, and availability of your caching solution. You can cluster up to 32 content engines to scale to your desired capacity.

Beginning in Cisco IOS XE Release 3.7.1E, WCCPv2 is supported on Virtual Switching System (VSS) on Cisco Catalyst 4500-E (Supervisor Engine 7-E and Supervisor Engine 8-E) and Cisco Catalyst 4500-X switches, on the IP Base image and the Enterprise Services image.

Beginning in Cisco IOS XE Release 3.8.0E, WCCPv2 supports traffic redirection to and from Virtual Routing and Forwarding (VRF) interfaces. Ensure that you configure the content engine running WCCP such that the forward and return traffic, to and from the content engine, is from interfaces that are a part of the same VRF. The VRF used for WCCP on an interface should match the VRF configured on that interface.

Hardware Acceleration

Hardware Acceleration is enabled by default on Catalyst 4500 series switches. Layer 2 rewrite forwarding and Layer 2 return method are supported in hardware.

When the switch exhausts hardware (TCAM) or software resources, traffic is redirected in software. GRE return method is supported only in software.

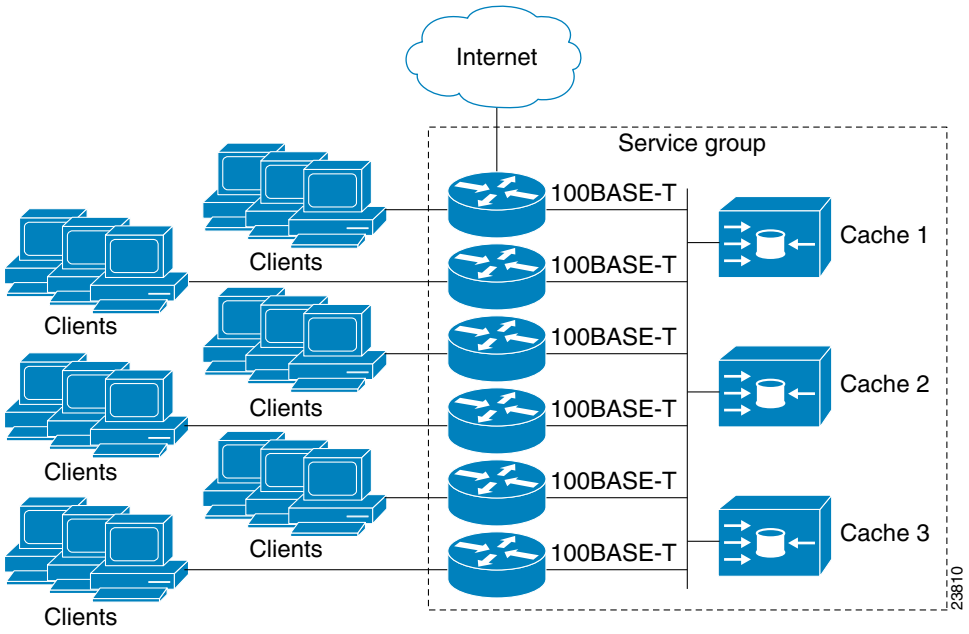
Configure a directly connected content engine to negotiate use of the WCCP Layer 2 Redirection feature (with load balancing) based on the mask assignment table. The **show ip wccp web-cache detail** command displays the redirection method in use for each cache.

Note You can configure the Cisco Content Engine Release 2.2 or later to use the WCCP Layer 2 Redirection feature with the mask assignment table.

Understanding WCCP Configuration

Multiple routers can use WCCP to service a cache cluster. Figure 76-1 illustrates a sample configuration using multiple routers.

Figure 76-1 Cisco Content Engine Network Configuration Using WCCP



The subset of content engines within a cluster and routers connected to the cluster that are running the same service is known as a *service group*. Available services include TCP and User Datagram Protocol (UDP) redirection.

WCCP requires that each content engine be aware of all the routers in the service group. You must a list of IP addresses for each of the routers in the group configured on each content engine. The address of each router in the group must be explicitly specified for each content engine during configuration.

The following sequence of events describe how WCCP works:

1. Each WCCP client (content engine) is configured with a list of WCCP servers (routers).
2. Each content engine announces its presence with a `Here I Am` message and a list of routers with which it has established communication.s The routers reply with their view (list) of content engines in the service group through `I See You` messages.
3. If the view is consistent across all content engines in the cluster, one content engine is designated as the lead and sets the policy that the switches deploy in redirecting traffic.

WCCP Features

These sections describe WCCP features:

- HTTP and Non-HTTP Services Support
- Multiple Routers Support
- MD5 Security

- Web Content Packet Return

HTTP and Non-HTTP Services Support

WCCP enables redirection of HTTP traffic (TCP port 80 traffic), as well as non-HTTP traffic (TCP and UDP). WCCP supports the redirection of packets intended for other ports, including those used for proxy-web cache handling, File Transfer Protocol (FTP) caching, FTP proxy handling, web caching for ports other than 80, and real audio, video, and telephony applications.

To accommodate the various types of services available, WCCP introduces the concept of multiple *service groups*. Service information is specified in the WCCP configuration commands using dynamic services identification numbers (such as 98) or a predefined service keywords (such as web-cache). This information is used to validate that service group members are all using or providing the same service.

Note The Catalyst 4500 series switch supports up to eight service groups.

For information on supported WCCP version 2 services with ACNS version 5.2 software, refer to the *Release Notes for Cisco ACNS Software, Release 5.2.3.*

The content engines in service group specify traffic to be redirected by protocol (TCP or UDP) and port (source or destination). Each service group has a priority level assigned to it. Packets are matched against service groups in priority order and redirected by the highest priority service group that matches traffic characteristics.

Multiple Routers Support

WCCP enables you to attach multiple routers to a cluster of cache engines. The use of multiple routers in a service group enables redundancy, interface aggregation, and distribution of the redirection load.

MD5 Security

WCCP provides optional authentication that enables you to control which routers and content engines become part of the service group using passwords and the HMAC MD5 standard. Shared-secret MD5 one-time authentication (set using the **ip wccp** [**password** [**0-7**] *password*] global configuration command) enables messages to be protected against interception, inspection, and replay.

Web Content Packet Return

If a content engine is unable to provide a requested object it has cached due to error or overload, the content engine returns the request to the router for onward transmission to the originally specified destination server. WCCP verifies which requests have been returned from the content engine unserviced. Using this information, the router can then forward the request to the originally targeted server (rather than attempting to resend the request to the content cluster). This provides error handling transparency to clients.

Typical reasons why a content engine would reject packets and initiate the packet return feature include the following:

- Instances when the content engine is overloaded and has no room to service the packets.
- Instances when the content engine is filtering for certain conditions that make caching packets counterproductive (such as, when IP authentication has been turned on).

Restrictions for WCCP

The following limitations apply to WCCP:

- Time To Live (TTL) value of Layer 3 switches servicing a cluster must be 15 seconds or less.
- A service group can comprise up to 32 content engines and 32 devices.
- All the content engines in a cluster must be configured to communicate with all the devices servicing the cluster.
- A total of eight active IPv4 and IPv6 service groups are supported on a switch. If used in pairs, up to four service-group pairs can be configured simultaneously.
- The Layer 2 rewrite forwarding method is supported in the hardware.
- The Layer 2 return method is supported in the hardware and is recommended.
- The content engine must be directly connected to the device.
- Input/output redirection configuration is not supported on content engines facing interfaces.
- WCCPv2 supports up to 256 distinct masks. However, a Catalyst 4500 series switch only supports a single mask.

Configuring WCCP

The following configuration tasks assume that you have already installed and configured the content engines you want to include in your network. You must configure the content engines in the cluster before configuring WCCP on your device. Refer to the *Cisco Content Engine User Guide* for content engine configuration and setup tasks.

IP must be configured on the device interface connected to the cache engines. Examples of device configuration tasks follow this section. For complete descriptions of the command syntax, refer to the *Cisco IOS Configuration Fundamentals Command Reference*, *Cisco IOS Release 12.3*.

These sections describe how to configure WCCP:

- Configuring a Service Group Using WCCP, page 76-5 (Required)
- Using Access Lists for a WCCP Service Group, page 76-8 (Optional)
- Setting a Password for a Switch and Cache Engine, page 76-8 (Optional)

Configuring a Service Group Using WCCP

WCCP uses service groups based on logical redirection services. The standard service is the content engine, which intercepts TCP port 80 (HTTP) traffic and redirects that traffic to the content engines. This service is referred to as a well-known service, because the characteristics of the web cache service are known by both the router and content engines. A description of a well-known service is not required beyond a service identification (the command line interface (CLI) provides a **web-cache** keyword in the command syntax).

For information on supported WCCP services with ACNS version 5.2 software, refer to the *Release Notes for Cisco ACNS Software, Release 5.2.3.*

In addition to the web cache service, there can be up to seven dynamic services running concurrently on the switch.

Note More than one service can run on a switch at the same time, and routers and content engines can be part of multiple service groups at the same time.

The dynamic services are defined by the content engines; the content engine instructs the router which protocol or ports to intercept, and how to distribute the traffic. The router itself does not have information on the characteristics of the dynamic service group's traffic, because this information is provided by the first content engine to join the group. In a dynamic service, up to eight ports can be specified within a single protocol TCP or UDP).

Cisco Content Engines, for example, use dynamic service 99 to specify a reverse-proxy service. However, other content engines may use this service number for some other service. The following configuration information deals with enabling general services on Cisco routers. Refer to the content engine documentation for information on configuring services on content engines.

To enable a service on a Catalyst 4500 series switch, perform this task:

	Command	Purpose

Step 1	`Switch(config)# ip wccp [vrf vrf-name][group-address groupaddress] [redirect-list access-list] [group-list access-list] [password password]`
	
	For IPv6
	
	`Switch(config)# ipv6 wccp [vrf vrf-name] [group-address groupaddress] [redirect-list access-list] [group-list access-list]`
		Specifies the following:
		• A dynamic service to enable on the switch,
		• The IP multicast address used by the service group (optional)
		• The redirect access list to control the traffic to be redirected (optional)
		• The group list to use for content engine membership (optional)
		• Whether to use MD5 authentication (optional)
		Enables the WCCP service.

Step 2	`Switch(config-if)# [no] ip wccp check services all`
	
	For IPv6
	
	`Switch(config-if)# [no] ipv6 wccp check services all`
		If a service matches the packet and the service has a redirect access list configured, then the IP packet will be checked against the access list. If the packet is rejected by the access-list, the packet will not be passed down to lower priority services unless the **ip wccp check services all** command is configured. After the **ip wccp check services all** command is configured, WCCP will continue to attempt to match the packet against any remaining low priority services configured on the interface.

Step 3	`Switch(config)# interface type number`	Specifies the client interface to be configured and enters interface configuration mode.

Step 4	`Switch(config-if)# ip wccp [vrf vrf-name]{web-cache | service-number} redirect {in | out}`
	
	For IPv6
	
	`Switch(config-if)# ipv6 wccp [vrf vrf-name] redirect in`
		For IPv4, enables WCCP redirection for ingress or egress traffic on the specified client interface.
		For IPv6, enables WCCP redirection for ingress traffic on the specified client interface.

	Command	Purpose
Step 5	`Switch(config)# interface type number`	Specifies the interface to be configured for egress redirection exclusion
Step 6	`Switch(config-if)# ip wccp redirect exclude in`	Specifies that packets received on this interface be excluded from egress redirection. This command MUST be configured on the content engine interface if Layer 2 return method is used by the content engine and egress redirection is configured on the server interface.

Specifying a Web Cache Service

To configure a web cache service and ingress redirection for IPv4 perform this task:

	Command	Purpose
Step 1	`Switch(config)# ip wccp [vrf vrf-name] web-cache`	Enables the web cache service on the switch.
Step 2	`Switch(config)# interface type number`	Targets a client interface number for which the web cache service runs, and enters interface configuration mode.
Step 3	`Switch(config-if)# ip wccp [vrf vrf-name] web-cache redirect in`	Enables the verification on packets to determine if they qualify to be redirected to a content engine, using the client interface specified in Step 2.

To configure a web cache service and egress redirection for IPv4 traffic, perform this task:

	Command	Purpose
Step 1	`Switch(config)# ip wccp [vrf vrf-name] web-cache`	Enables the web cache service on the switch.
Step 2	`Switch(config)# interface type number`	Targets a server interface number for the web cache service, and enters interface configuration mode.
Step 3	`Switch(config-if)# ip wccp web-cache redirect out`	Enables the verification on packets to determine if they qualify to be redirected to a content engine, using the client interface specified in Step 2.
Step 4	`Switch(config)# interface type number`	Specifies the content engine interface number, and enters interface configuration mode.
Step 5	`Switch(config-if)# ip wccp redirect exclude in`	Specifies that packets received on this interface be excluded from egress redirection. This prevents the packets returned by the content engine through the L2-return method or the packets generated by the content engine from being redirected back to the content engine.

Using Access Lists for a WCCP Service Group

A Catalyst 4500 series switch can use an access list to restrict the content engines that can join a service group.

To restrict a content engine, perform this task:

	Command	Purpose
Step 1	`Switch(config)# access-list access-list permit ip host host-address [destination-address | destination-host | any]`	Creates an access list based on the unicast address of the content engines.

Step 2	`Switch(config)# ip wccp web-cache redirect-list access-list`
	
	For IPv6
	
	`Switch(config)# ipv6 wccp [vrf vrf-name]{service-number} redirect-list access-list`
		Indicates to the switch which content engines are allowed or disallowed to form a service group.

Setting a Password for a Switch and Cache Engine

MD5 password security requires that each content engine and Catalyst 4500 series switch that wants to join a service group be configured with the service group password. The password can consist of up to seven characters. Each content engine or Catalyst 4500 series switch in the service group authenticates the security component in a received WCCP packet immediately after validating the WCCP message header. Packets failing authentication are discarded.

To configure an MD5 password for use by the Catalyst 4500 series switch in WCCP communications, perform this task:

Command	Purpose
`Switch(config)# ip wccp web-cache password password`	Sets an MD5 password on the Catalyst 4500 series switch.

Verifying and Monitoring WCCP Configuration Settings

To verify and monitor the configuration settings for WCCP, use the following commands in EXEC mode:

Command	Purpose

`Switch# show ip wccp [vrf vrf-name] [web-cache | service-number]`
 	
For IPv6
 	
`Switch# show ipv6 wccp [vrf vrf-name]`
	Displays global information related to WCCP, including the protocol version that is currently running, the number of content engines in the routers service group, the content engine group is allowed to connect to the device, and the access list being used.

`Switch# show ip wccp [vrf vrf-name] {web-cache | service-number} detail`
 	
For IPv6
 	
`Switch# show ipv6 wccp [vrf vrf-name] detail`
	Queries the device for information on which content engines of a specific service group that the device has detected. The information can be displayed for either the web cache service or for the specified dynamic service.

`Switch# show ip interface`	Displays the status about whether redirection commands are configured on a client interface. For example, `Web Cache Redirect is enabled / disabled`.

`Switch# show ip wccp [vrf vrf-name] {web-cache | service-number} view`
 	
For IPV6
 	
`Switch# show ipv6 wccp [vrf vrf-name] view`
	Displays the devices in a particular service group that have been detected and the content engines that are not visible to all other devices to which the current device is connected.
	
	The **view** keyword indicates a list of addresses of the service group. The information can be displayed for either the web cache service or the specified dynamic service.
	
	Note For further troubleshooting information, use the **show ip wccp** {**web-cache** | *service number*} **service** command.

WCCP Configuration Examples

This section provides the following configuration examples:

Example: Performing a General WCCP Configuration

The following example shows a general WCCP configuration session. VLAN 20 is for the client interface. VLAN 50 is for the content engine interface.

```
Switch# configure terminal
Switch(config)# ip wccp web-cache group-address 224.1.1.100 password alaska1
Switch(config)# interface vlan 20
Switch(config-if)# ip wccp web-cache redirect in
Switch(config)# interface vlan 50
Switch(config-if)# ip wccp web cache group-listen
```

The following example shows a general IPv6 WCCP configuration where GigabitEthernet 0/1/0 is the client interface and GigabitEthernet 0/2/0 is the content engine interface:

```
Switch# configure terminal
Switch(config)# ipv6 wccp interface GigabitEthernet 0/1/0
Switch(config)# ipv6 wccp check services all
Switch(config)# interface GigabitEthernet 0/1/0
witch(config-if)# ipv6 wccp redirect in
Switch(config)# interface GigabitEthernet 0/2/0
```

Example: Running a Web Cache Service

The following example shows a web cache service configuration session with ingress redirection, for IPv4:

```
Switch# configure terminal
Switch(config)# ip wccp web-cache
Switch(config)# interface vlan 20
Switch(config-if)# ip wccp web-cache redirect in
Switch# copy running-config startup-config
Switch# show ip interface vlan 20 | include WCCP Redirect

WCCP Redirect inbound is enabled
WCCP Redirect exclude is disabled
```

The following example shows a web-cache service configuration session with ingress redirection, for an IPv4 VRF interface:

```
Switch#configure terminal
Switch(config)#ip wccp vrf test web-cache
Switch(config)#interface vlan 10
Switch(config-if)#vrf forwarding test
Switch(config-if)#ip wccp vrf test web-cache redirect in
Switch# copy running-config startup-config
```

Example: Running a Reverse Proxy Service

Note The WCCP reverse proxy service is not supported for IPv6 traffic.

The following example assumes you a configuring a service group using Cisco Content Engines, which use dynamic service 99 to run a reverse proxy service. The following example illustrates how to configure egress redirection, where VLAN 40 reflects the server interface and VLAN 50 reflects the content engine interface:

```
Switch# configure terminal
Switch(config)# ip wccp 99
Switch(config)# interface vlan 40
Switch(config-if)# ip wccp 99 redirect in
Switch(config# interface vlan 50
Switch(config-if)# ip wccp redirect exclude in
```

For IPv6

```
Switch# configure terminal
Switch(config)# ipv6 wccp 99
Switch(config)# interface vlan 40
Switch(config-if)# ipv6 wccp 99 redirect in
Switch(config# interface vlan 50
```

Example: Running TCP-Promiscuous Service

The following example shows how to configure TCP promiscuous service, where VLAN 40 represents the server interface and VLAN 50 represents the content engine interface:

```
Switch# configure terminal
Switch(config)# ip wccp 61
Switch(config)# ip wccp 62
Switch(config)# interface vlan 30
Switch(config-if)# ip wccp 61 redirect in
Switch(config)# interface vlan 40
Switch(config-if)# ip wccp 62 redirect in
Switch(config)# interface vlan 50
Switch(config-if)# ip wccp redirect exclude in
```

For IPV6

```
Switch# configure terminal
Switch(config)# ipv6 wccp 51
Switch(config)# ipv6 wccp 52
Switch(config)# interface vlan 30
Switch(config-if)# ipv6 wccp 51 redirect in
Switch(config)# interface vlan 40
Switch(config-if)# ipv6 wccp 52 redirect in
```

The following example shows how to configure the TCP promiscuous service for IPv4 VRF interfaces, where VLAN 40 represents the server interface and VLAN 50 represents the content engine interface:

```
Switch# configure terminal
Switch(config)# ip wccp vrf abc 91
Switch(config)# ip wccp vrf abc 92
Switch(config)# interface vlan 30
Switch(config-if)# vrf forwarding abc s
Switch(config-if)# ip wccp vrf abc 91 redirect in
Switch(config)# interface vlan 40
Switch(config-if)# vrf forwarding abc
Switch(config-if)# ip wccp vrf abc 92 redirect in
Switch(config)# interface vlan 50
Switch(config-if)# vrf forwarding abc
```

Example: Running Redirect Access List

The following example shows how to redirect traffic only from subnet 10.1.1.0:

```
Switch(config)# ip access-list extended 100
Switch(config-ext-nacl)# permit ip 10.1.1.0 255.255.255.0 any
Switch(config-ext-nacl)# exit
Switch(config)# ip wccp web-cache redirect-list 100
Switch(config)# interface vlan 40
Switch(config-if)# ip wccp web-cache redirect in
Switch(config)# interface vlan 50
Switch(config-if)# ip wccp redirect exclude in
```

The following example shows how to redirect IPv6 traffic only from 2001::1/64 2004::1/64 eq www:

```
switch(config)# ipv6 access-list ACL_1
switch(config-ipv6-acl)# permit tcp 2001::1/64 2004::1/64 eq www
switch(config-ipv6-acl)# exit
switch(config)# ipv6 wccp 61 redirect-list ACL_1
switch(config)# interface vlan 40
witch(config-if)# ipv6 wccp 61 redirect in
```

Example: Using Access Lists

To achieve better security, you can use a standard access list to notify the Catalyst 4500 series switch to which IP addresses are valid for a content engine attempting to register with the current switch. The following example shows a standard access list configuration session where the access list number is 10 for some sample hosts:

```
switch(config)# access-list 10 permit host 11.1.1.1
switch(config)# access-list 10 permit host 11.1.1.2
switch(config)# access-list 10 permit host 11.1.1.3
switch(config)# ip wccp web-cache group-list 10
```

The following examples shows a standard access list configuration for IPv6:

```
switch(config)#ipv6 access-list ACL_1
switch(config-ipv6-acl)#permit tcp 2001::1/64 2004::1/64 eq www
switch(config)#ipv6 wccp 61 redirect-list ACL_1
```

Example: Setting a Password for a Switch and Content Engines

The following example shows a WCCP password configuration session where the password is alaska1:

```
Switch# configure terminal
Switch(config)# ip wccp web-cache password alaska1
```

Example: Verifying WCCP Settings

To verify your configuration changes, use the **more system:running-config** EXEC command. The following example shows that both the web cache service and dynamic service 99 are enabled on the Catalyst 4500 series switch:

WCCP Unicast Mode

```
Switch# more system:running-config

    Building configuration...
    Current configuration:
    !
    version 12.2
    service timestamps debug uptime
    service timestamps log uptime
    no service password-encryption
    service udp-small-servers
    service tcp-small-servers
    ! !
    enable secret 5 $1$nSVy$faliJsVQXVPW.KuCxZNTh1
    enable password alabama1
    !
    ip subnet-zero
    ip wccp web-cache
    ip wccp 99
    !
    interface Vlan200
    ip address 10.3.1.2 255.255.255.0
    ip wccp web-cache redirect in

    interface Vlan300
    ip address 10.4.1.1 255.255.255.0
    ip wccp redirect exclude in

    interface Vlan400
    ip address 10.5.1 255.255.255.0
    ip wccp 99 redirect out



    ip default-gateway 10.3.1.1
    ip classless
    ip route 0.0.0.0 0.0.0.0 10.3.1.1
    no ip http server
    !
    !
```