Configuring WCCP

This section provides information about configuring WCCP.

- Introduction, on page 1
- Prerequisites for WCCP, on page 1
- Restrictions for WCCP, on page 1
- Information About WCCP, on page 3
- How to Configure WCCP, on page 8
- Configuration Examples for WCCP, on page 16
- Feature Information for WCCP, on page 21

Introduction

The Web Cache Communication Protocol (WCCP) is a Cisco-developed content-routing technology that intercepts IP packets and redirects those packets to a destination other than that specified in the IP packet. Typically the packets are redirected from their destination web server on the Internet to a content engine that is local to the client. In some WCCP deployment scenarios, redirection of traffic may also be required from the web server to the client. WCCP enables you to integrate content engines into your network infrastructure.

The tasks in this document assume that you have already configured content engines on your network.

Prerequisites for WCCP

- To use WCCP, IP must be configured on the interface connected to the Internet and another interface must be connected to the content engine.
- The interface connected to the content engine must be a Fast Ethernet or Gigabit Ethernet interface.

Restrictions for WCCP

General

The following limitations apply to Web Cache Communication Protocol Version 2 (WCCPv2):

- WCCP works only with IPv4 networks.

- WCCP bypasses Network Address Translation (NAT) when Cisco Express Forwarding is enabled.
- WCCP does not interoperate with NAT and the zone-based firewall configured together in a network.
- Service groups can comprise up to 32 content engines and 32 switches.
- For switches servicing a multicast cluster, the Time To Live (TTL) value must be set at 15 or fewer.
- All content engines in a cluster must be configured to communicate with all devices servicing the cluster.
- Multicast addresses must be from 224.0.0.0 to 239.255.255.255.
- Up to eight service groups are supported at the same time on the same client interface.
- The Layer 2 rewrite forwarding method is supported; but generic routing encapsulation (GRE) is not.
- Direct Layer 2 connectivity to content engines is required when Layer 2 mode is deployed; Layer 3 connectivity of one or more hops away is not supported.
- Ternary content addressable memory (TCAM) friendly mask-based assignment is supported, but the hash bucket-based method is not.
- When the TCAM space is exhausted, traffic is not redirected but is forwarded normally.
- The WCCP version 2 standard allows for support of up to 256 distinct masks. However, a Cisco Catalyst 9000 series switch supports only mask assignment tables with a single mask.
- A content engine configured for mask assignment that tries to join a farm where the selected assignment method is hash remains out of the farm as long as the cache engine assignment method does not match that of the existing farm.
- WCCP redirection is not supported on Multiprotocol Label Switching (MPLS) and port-channel interfaces.
- WCCP high availability is not supported in modular, stacking, and StackWise Virtual (SVL) mode.

Catalyst 9000 Series Switches Access Control Lists

When WCCP is using the mask assignment, any redirect list is merged with the mask information from the appliance and the resulting merged ACL is passed down to the Catalyst 9000 series switch hardware. Only Permit or Deny ACL entries from the redirect list in which the protocol is IP or exactly matches the service group protocol are merged with the mask information from the appliance.

The following restrictions apply to the redirect-list ACL:

- The ACL must be an IPv4 extended ACL.
- Only individual source or destination port numbers may be specified; port ranges cannot be specified.
- The only valid matching criteria in addition to individual source or destination port numbers are **dscp** or **tos**.
- The use of **fragments**, **time-range**, or **options** keywords, or any TCP flags is not permitted.
- If the redirect ACL does not meet the restrictions shown, the system will log the following error message:

```
WCCP-3-BADACE: Service <service group>, invalid access-list entry (seq:<sequence>,
reason:<reason>)
```

Information About WCCP

WCCP Overview

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

WCCP enables Cisco IOS XE platforms to transparently redirect content requests. With transparent redirection, users can fulfill content requests locally without configuring 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.

A content engine receiving a request attempts to service it from its own local cache. If the requested information is not present, the content engine issues its own request to the originally targeted server to get the required information. A content engine retrieving the requested information 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 device or multiple devices. Network administrators can easily scale their content engines to manage heavy traffic loads through these clustering capabilities. Cisco clustering technology enables each cluster 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.

WCCP Mask Assignment

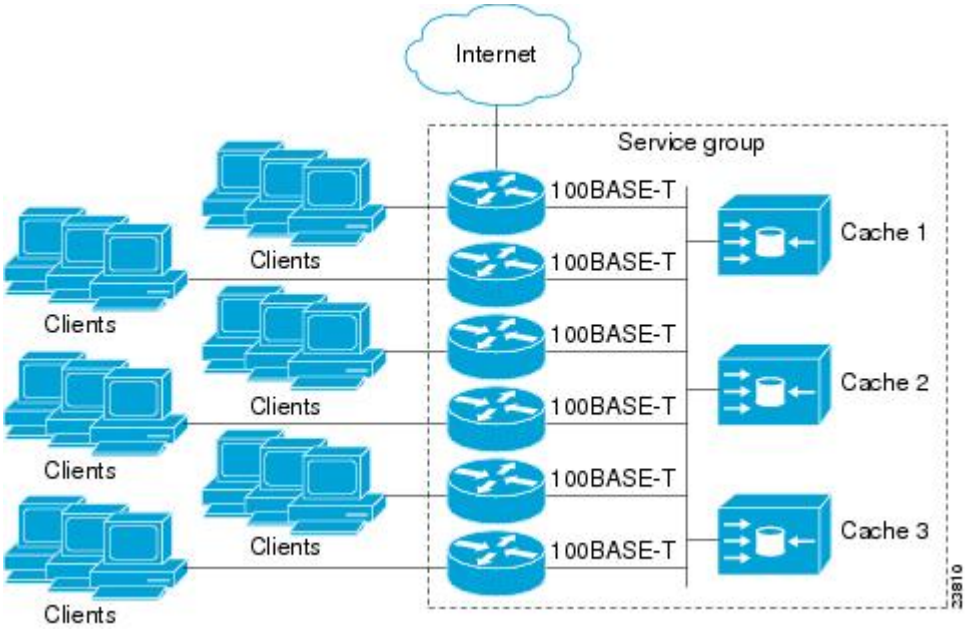
The WCCP Mask Assignment feature enables mask assignment as the load-balancing method (instead of the default hash assignment method) for a WCCP service.

For content engines running Application and Content Networking System (ACNS) software, use the **wccp custom-web-cache** command with the **mask-assign** keyword to configure mask assignment. For content engines running Cisco Wide Area Application Services (WAAS) software, use the **wccp tcp-promiscuous** command with the **mask-assign** keyword to configure mask assignment.

WCCPv2 Configuration

Multiple devices can use WCCPv2 to service a content engine cluster. The figure below illustrates a sample configuration using multiple devices.

Figure 1: Cisco Content Engine Network Configuration Using WCCPv2



The subset of content engines within a cluster and devices connected to the cluster that are running the same service is known as a service group. Available services include TCP and UDP redirection.

WCCPv2 requires that each content engine be aware of all the devices in the service group. To specify the addresses of all the devices in a service group, choose one of the following methods:

- Unicast—A list of device addresses for each of the devices in the group is configured on each content engine. In this case, the address of each device in the group must be explicitly specified for each content engine during configuration.
- Multicast—A single multicast address is configured on each content engine. In the multicast address method, the content engine sends a single-address notification that provides coverage for all switches in the service group. For example, a content engine could indicate that packets should be sent to a multicast address of 224.0.0.100, which would send a multicast packet to all devices in the service group configured for group listening using WCCP (see the **ip wccp group-listen** interface configuration command for details).

The multicast option is easier to configure because you need only specify a single address on each content engine. This option also allows you to add and remove routers from a service group dynamically, without needing to reconfigure the content engines with a different list of addresses each time.

The following sequence of events details how WCCPv2 configuration works:

1. Each content engine is configured with a list of devices.
2. Each content engine announces its presence and a list of all devices with which it has established communications. The routers reply with their view (list) of content engines in the group.
3. When 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 devices need to deploy in redirecting packets.

WCCPv2 Support for Services Other Than HTTP

WCCPv2 allows redirection of traffic other than HTTP (TCP port 80 traffic), including a variety of UDP and TCP traffic. WCCPv2 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, WCCPv2 introduced 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 keyword (such as **web-cache**). This information is used to validate that service group members are all using or providing the same service.

The content engines in a service group specify traffic to be redirected by protocol (TCP or UDP) and up to eight source or destination ports. Each service group has a priority status assigned to it. The priority of a dynamic service is assigned by the content engine. The priority value is in the range of 0 to 255 where 0 is the lowest priority. The predefined web-cache service has an assigned priority of 240.

WCCPv2 Support for Multiple Devices

WCCPv2 allows multiple devices to be attached to a cluster of cache engines. The use of multiple devices in a service group allows for redundancy, interface aggregation, and distribution of the redirection load. WCCPv2 supports up to 32 devices per service group. Each service group is established and maintained independently.

WCCPv2 MD5 Security

WCCPv2 provides optional authentication that enables you to control which switches and content engines become part of the service group using passwords and the Hashed Message Authentication Code—Message Digest (HMAC MD5) standard. Shared-secret MD5 one-time authentication (set using the **ip wccp password** *password* global configuration command) enables messages to be protected against interception, inspection, and replay.

WCCPv2 Web Cache Packet Return

If a content engine is unable to provide a requested object it has cached due to error or overload, the content engine will return the request to the device for onward transmission to the originally specified destination server. WCCPv2 provides a check on packets that determines which requests have been returned from the content engine unserviced. Using this information, the device can then forward the request to the originally targeted server (rather than attempting to resend the request to the content engine cluster). This process 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 (for example, when IP authentication has been turned on)

WCCPv2 Load Distribution

WCCPv2 can be used to adjust the load being offered to individual content engines to provide an effective use of the available resources while helping to ensure high quality of service (QoS) to the clients. WCCPv2 allows the designated content engine to adjust the load on a particular content engine and balance the load across the content engines in a cluster. WCCPv2 uses three techniques to perform load distribution:

- Hot spot handling—Allows an individual hash bucket to be distributed across all the content engines. Prior to WCCPv2, information from one hash bucket could go to only one content engine.
- Load balancing—Allows the set of hash buckets assigned to a content engine to be adjusted so that the load can be shifted from an overwhelmed content engine to other members that have available capacity.
- Load shedding—Enables the switch to selectively redirect the load to avoid exceeding the capacity of a content engine.

The use of these hashing parameters prevents one content engine from being overloaded and reduces the potential for bottlenecking.

WCCP Bypass Packets

WCCP intercepts IP packets and redirects those packets to a destination other than the destination that is specified in the IP header. Typically the packets are redirected from a web server on the Internet to a web cache that is local to the destination.

Occasionally a web cache cannot manage the redirected packets appropriately and returns the packets unchanged to the originating device. These packets are called bypass packets and are returned to the originating device using Layer 2 forwarding without encapsulation (L2). The device decapsulates and forwards the packets normally. The VRF associated with the ingress interface (or the global table if there is no VRF associated) is used to route the packet to the destination.

WCCP Closed Services and Open Services

In applications where packets are intercepted and redirected by a Cisco switch or a router to external WCCP client devices, it may be necessary to block the packets for the application when a WCCP client device is not available. This blocking is achieved by configuring a WCCP closed service. When a WCCP service is configured as closed, the packets that fulfill the services, but do not have an active client device, are discarded.

By default, WCCP operates as an open service, wherein communication between clients and servers proceeds normally in the absence of an intermediary device.

The **ip wccp service-list** command can be used for both closed-mode and open-mode services. Use the **service-list** keyword and *service-access-list* argument to register an application protocol type or port number. Use the **mode** keyword to select an open or closed service.

WCCP Outbound ACL Check

When WCCP is enabled for redirection on an ingress interface, the packets are redirected by WCCP and instead egress on an interface other than the destination that is specified in the IP header. The packets are still subject to ACLs configured on the ingress interface. However, redirection can cause the packets to bypass the ACL configured on the original egress interface. Packets that would have been dropped because of the ACL configured on the original egress interface can be sent out on the redirect egress interface, which poses

a possible security problem. Enabling the WCCP Outbound ACL check feature ensures that redirected packets are subject to any ACL conditions configured on the original egress interface.

WCCP Service Groups

WCCP is a component of Cisco IOS XE software that redirects traffic with defined characteristics from its original destination to an alternative destination. The typical application of WCCP is to redirect traffic bound for a remote web server to a local web cache to improve response time and optimize network resource usage.

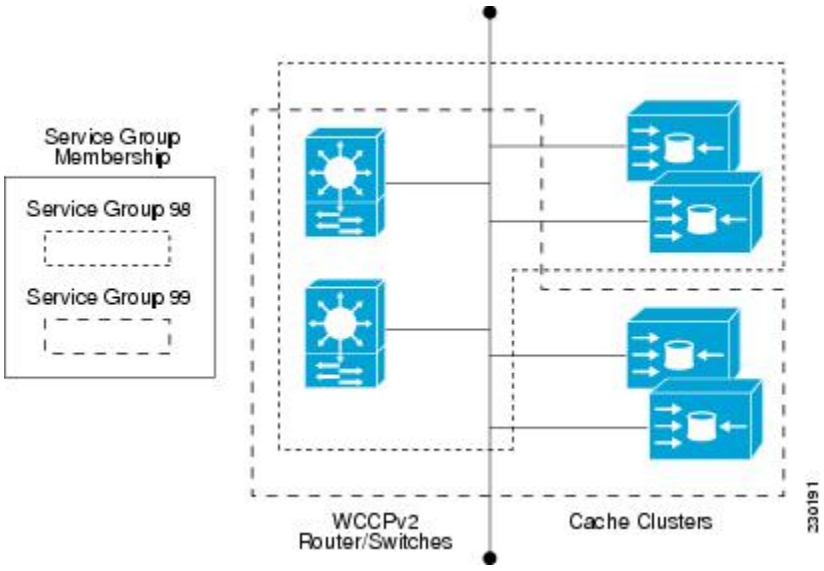
The nature of the selected traffic for redirection is defined by service groups (see figure below) specified on content engines and communicated to switches or routers using WCCP.

WCCPv2 supports up to 32 switches per service group. Each service group is established and maintained independently.

WCCPv2 uses service groups based on logical redirection services, deployed for intercepting and redirecting traffic. The standard service is web cache, 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 switch and content engines. A description of a well-known service is not required beyond a service identification. To specify the standard web cache service, use the **ip wccp** command with the **web-cache** keyword.

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

Figure 2: WCCP Service Groups



The dynamic services are defined by the content engines; the content engine instructs the switch which protocol or ports to intercept, and how to distribute the traffic. The switch 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.

Cisco Content Engines, for example, use dynamic service 99 to specify a reverse-proxy service. However, other content engine devices may use this service number for some other service.

WCCP—Check All Services

An interface may be configured with more than one WCCP service. When more than one WCCP service is configured on an interface, the precedence of a service depends on the relative priority of the service compared to the priority of the other configured services. Each WCCP service has a priority value as part of its definition. When an interface is configured with more than one WCCP service, the precedence of the packets is matched against service groups in priority order.

Note The priority of a WCCP service group cannot be configured via Cisco IOS XE software.

With the **ip wccp check services all** command, WCCP can be configured to check all configured services for a match and perform redirection for those services if appropriate. The caches to which packets are redirected can be controlled by a redirect ACL and by the service priority. The **ip wccp check services all** commands must be configured at global level to support multiple WCCP services.

If no WCCP services are configured with a redirect ACL, the services are considered in priority order until a service is found that matches the IP packet. If no services match the packet, the packet is not redirected. If a service matches the packet and the service has a redirect ACL configured, then the IP packet will be checked against the ACL. If the packet is rejected by the ACL, the packet will not be passed down to lower priority services unless the **ip wccp check services all** command is configured. When the **ip wccp check services all** command is configured, WCCP will continue to attempt to match the packet against any remaining lower priority services configured on the interface.

WCCP Troubleshooting Tips

CPU usage may be very high when WCCP is enabled. The WCCP counters enable a determination of the bypass traffic directly on the switch and can indicate whether the cause is high CPU usage due to enablement of WCCP. In some situations, 10 percent bypass traffic may be normal; in other situations, 10 percent may be high. However, any figure above 25 percent should prompt a closer investigation of what is occurring in the web cache.

If the counters suggest that the level of bypass traffic is high, the next step is to examine the bypass counters in the content engine and determine why the content engine is choosing to bypass the traffic. You can log in to the content engine console and use the CLI to investigate further. The counters allow you to determine the percent of traffic being bypassed.

You can use the **clear wccp** command to remove all WCCP statistics (counts) maintained on the device for a particular service.

You can use the **show wccp** command to display all WCCP global statistics (counts).

How to Configure 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 functionality on your routers or switches. Refer to the Cisco Cache Engine User Guide for content engine configuration and setup tasks.

Configuring WCCP

Perform this task to configure WCCP.

Until you configure a WCCP service using the **ip wccp**{**web-cache** | *service-number*} global configuration command, WCCP is disabled on the device. The first use of a form of the **ip wccp** command enables WCCP.

Use the **ip wccp web-cache password** command to set a password for a device and the content engines in a service group. MD5 password security requires that each device and content engine that wants to join a service group be configured with the service group password. The password must be up to eight characters in length. Each content engine or device in the service group will authenticate the security component in a received WCCP packet immediately after validating the WCCP message header. Packets failing authentication will be discarded.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password* [**0** | **7**]] **Example:** `Device(config)# ip wccp web-cache password pwd`	Specifies a web-cache or dynamic service to enable on a device, specifies the IP multicast address used by the service group, specifies any access lists to use, specifies whether to use MD5 authentication, and enables the WCCP service. • **Note** The password length must not exceed 8 characters.
Step 4	**interface** *type number* **Example:** `Device(config)# interface Gigabitethernet 0/0`	Targets an interface number for which the web cache service will run, and enters interface configuration mode.
Step 5	**ip wccp** {**web-cache** | *service-number*} **redirect** {**in** | **out**} **Example:**	Enables packet redirection on an outbound or inbound interface using WCCP.

	Command or Action	Purpose
	`Device(config-if)# ip wccp web-cache redirect in`	• As indicated by the **out** and **in** keyword options, redirection can be specified for outbound interfaces or inbound interfaces.
Step 6	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/2/0`	Targets an interface number on which to exclude traffic for redirection, and enters interface configuration mode.
Step 8	**ip wccp redirect exclude in** **Example:** `Device(config-if)# ip wccp redirect exclude in`	(Optional) Excludes traffic on the specified interface from redirection.

Configuring Closed Services

Perform this task to specify the number of service groups for WCCP, to configure a service group as a closed or open service, and to optionally specify a check of all services.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	Enter one of the following commands: • **ip wccp** *service-number* [**service-list** *service-access-list* **mode** {**open** | **closed**}] • or • **ip wccp web-cache mode** {**open** | **closed**} **Example:**	Configures a dynamic WCCP service as closed or open. or Configures a web-cache service as closed or open. **Note**

	Command or Action	Purpose

	`Device(config)# ip wccp 90 service-list 120 mode closed`
	or
	`Device(config)# ip wccp web-cache mode closed`
		When configuring the web-cache service as a closed service, you cannot specify a service access list.
		Note When configuring a dynamic WCCP service as a closed service, you must specify a service access list.

Step 4	**ip wccp check services all** **Example:** `Device(config)# ip wccp check services all`	(Optional) Enables a check of all WCCP services. • Use this command to configure WCCP to check the other configured services for a match and perform redirection for those services if appropriate. The caches to which packets are redirected can be controlled by the redirect ACL and not just the service description. **Note** The **ip wccp check services all** command is a global WCCP command that applies to all services and is not associated with a single service.
Step 5	**ip wccp** {**web-cache** | *service-number*} **Example:** `Device(config)# ip wccp 201`	Specifies the WCCP service identifier. • You can specify the standard web-cache service or a dynamic service number from 0 to 255. • The maximum number of services that can be specified is 256.
Step 6	**exit** **Example:** `Device(config)# exit`	Exits to privileged EXEC mode.

Registering a Device to a Multicast Address

If you decide to use the multicast address option for your service group, you must configure the device to listen for the multicast broadcasts on an interface.

For network configurations where redirected traffic needs to traverse an intervening device, the device being traversed must be configured to perform IP multicast routing. You must configure the following two components to enable traversal over an intervening device:

- Enable IP multicast routing using the **ip multicast-routing** global configuration command.

- Enable the interfaces to which the cache engines will connect to receive multicast transmissions using the **ip wccp group-listen** interface configuration command.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip multicast-routing** [**vrf** *vrf-name*] [**distributed**] **Example:** `Device(config)# ip multicast-routing`	Enables IP multicast routing.
Step 4	**ip wccp** {**web-cache** | *service-number*} **group-address** *multicast-address* **Example:** `Device(config)# ip wccp 99 group-address 239.1.1.1`	Specifies the multicast address for the service group.
Step 5	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Enables the interfaces to which the content engines will connect to receive multicast transmissions for which the web cache service will run, and enters interface configuration mode.
Step 6	**ip pim** {**sparse-mode** | **sparse-dense-mode** | **dense-mode** [**proxy-register** {**list** *access-list* | **route-map** *map-name*}]} **Example:** `Device(config-if)# ip pim dense-mode`	(Optional) Enables Protocol Independent Multicast (PIM) on an interface. **Note** To ensure correct operation of the **ip wccp group-listen** command on Catalyst 9000 series switches, you must enter the **ip pim** command in addition to the **ip wccp group-listen** command.
Step 7	**ip wccp** {**web-cache** | *service-number*} **group-listen** **Example:**	Configures an interface to enable or disable the reception of IP multicast packets for WCCP.

	Command or Action	Purpose
	`Device(config-if)# ip wccp 99 group-listen`	

Using Access Lists for a WCCP Service Group

Perform this task to configure the device to use an access list to determine which traffic should be directed to which content engines.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**access-list** *access-list-number* **remark** *remark* **Example:** `Device(config)# access-list 1 remark Give access to user1`	(Optional) Adds a user-friendly comment about an access list entry. • A remark of up to 100 characters can precede or follow an access list entry.
Step 4	**access-list** *access-list-number* **permit** {*source* [*source-wildcard*] | **any**} [**log**] **Example:** `Device(config)# access-list 1 permit 172.16.5.22 0.0.0.0`	Creates an access list that enables or disables traffic redirection to the cache engine and permits the specified source based on a source address and wildcard mask. • Every access list needs at least one permit statement; it does not need to be the first entry. • Standard IP access lists are numbered 1 to 99 or 1300 to 1999. • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the keyword **any** as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255.

	Command or Action	Purpose
		• In this example, host 172.16.5.22 is allowed to pass the access list.
Step 5	**access-list** *access-list-number* **remark** *remark* **Example:** `Device(config)# access-list 1 remark Give access to user1`	(Optional) Adds a user-friendly comment about an access list entry. • A remark of up to 100 characters can precede or follow an access list entry.
Step 6	**access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} | [**log**] **Example:** `Device(config)# access-list 1 deny 172.16.7.34 0.0.0.0`	Denies the specified source based on a source address and wildcard mask. • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the abbreviation any as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255. • In this example, host 172.16.7.34 is denied passing the access list.
Step 7	Repeat some combination of Steps 3 through 6 until you have specified the sources on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit **deny** statement at the end of the access list.
Step 8	**ip wccp web-cache group-list** *access-list* **Example:** `Device(config) ip wccp web-cache group-list 1`	Indicates to the device from which IP addresses of content engines to accept packets.
Step 9	**ip wccp web-cache redirect-list** *access-list* **Example:** `Device(config)# ip wccp web-cache redirect-list 1`	(Optional) Disables caching for certain clients.

Enabling the WCCP Outbound ACL Check

Note When all redirection is performed in the hardware, the mode of redirection will change when outbound ACL checking is enabled. The first packet is switched in software to allow the extra ACL check to be performed before a shortcut is installed.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip wccp** {**web-cache** | *service-number*} [**group-address** *multicast-address*] [**redirect-list** *access-list*] [**group-list** *access-list*] [**password** *password*] **Example:** `Device(config)# ip wccp web-cache`	Enables the support for a Cisco content engine service group or any content engine service group and configures a redirect ACL list or group ACL. **Note** The **web-cache** keyword is for WCCP version 1 and version 2 and the *service-number* argument is for WCCP version 2 only.
Step 4	**ip wccp check acl outbound** **Example:** `Device(config)# ip wccp check acl outbound`	Checks the access control list (ACL) for egress interfaces for packets redirected by WCCP.
Step 5	**exit** **Example:** `Device(config)# exit`	Exits global configuration.

Verifying and Monitoring WCCP Configuration Settings

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip wccp** [**web-cache** |*service-number*] [**detail view**]	Displays global information related to WCCP, including the protocol version running, the

	Command or Action	Purpose
	Example: `Device# show ip wccp 24 detail`	number of content engines in the router service group, which content engine group is allowed to connect to the router, and which access list is being used. • *service-number*—(Optional) Dynamic number of the web-cache service group being controlled by the content engine. The range is from 0 to 99. For web caches that use Cisco Content Engines, the reverse proxy service is indicated by a value of 99. • **web-cache**—(Optional) statistics for the web-cache service. • **detail**—(Optional) other members of a particular service group or web cache that have or have not been detected. • **view**—(Optional) information about a router or all web caches.
Step 3	**show ip interface** **Example:** `Device# show ip interface`	Displays status about whether any **ip wccp redirection** commands are configured on an interface; for example, "Web Cache Redirect is enabled / disabled."
Step 4	**more system:running-config** **Example:** `Device# more system:running-config`	(Optional) Displays contents of the running configuration file (equivalent to the **show running-config** command).

Configuration Examples for WCCP

Example: Configuring a General WCCPv2 Session

```
Device# configure terminal
Device(config)# ip wccp web-cache group-address 224.1.1.100 password password
Device(config)# ip wccp source-interface GigabitEthernet 0/1/0
Device(config)# ip wccp check services all
! Configures a check of all WCCP services.
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device(config)# interface GigabitEthernet 0/2/0
Device(config-if)# ip wccp redirect exclude in
Device(config-if)# exit
```

Example: Setting a Password for a Device and Content Engines

```
Device# configure terminal
Device(config)# ip wccp web-cache password password1
```

Example: Configuring a Web Cache Service

```
Device# configure terminal
Device(config)# ip wccp web-cache
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device# copy running-config startup-config
```

The following example shows how to configure a session in which redirection of HTTP traffic arriving on Gigabit Ethernet interface 0/1/0 is enabled:

```
Device# configure terminal
Device(config)# interface GigabitEthernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
Device(config-if)# exit
Device# show ip interface GigabitEthernet 0/1/0
.
.
.
WCCP Redirect inbound is enabled
WCCP Redirect exclude is disabled
.
.
.
```

Example: Running a Reverse Proxy Service

The following example assumes that you are configuring a service group using Cisco cache engines, which use dynamic service 99 to run a reverse proxy service:

```
Device# configure terminal
Device(config)# ip wccp 99
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp 99 redirect out
```

Example: Registering a Device to a Multicast Address

```
Device# configure terminal
Device(config)# ip wccp web-cache group-address 224.1.1.100
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp web-cache group-listen
```

The following example shows a device configured to run a reverse proxy service, using the multicast address of 224.1.1.1. Redirection applies to packets going out through the Gigabit Ethernet interface 0/1/0:

```
Device# configure terminal
Device(config)# ip wccp 99 group-address 224.1.1.1
```

```
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp 99 redirect out
```

Example: Using Access Lists

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

```
Device(config)# access-list 10 permit host 10.1.1.1
Device(config)# access-list 10 permit host 10.1.1.2
Device(config)# access-list 10 permit host 10.1.1.3
Device(config)# ip wccp web-cache group-list 10
```

To disable caching for certain clients, servers, or client/server pairs, you can use WCCP access lists. The following example shows that any requests coming from 10.1.1.1 to 10.3.1.1 will bypass the cache, and that all other requests will be serviced normally:

```
Device(config)# ip wccp web-cache redirect-list 120
Device(config)# access-list 120 deny tcp host 10.1.1.1 any
Device(config)# access-list 120 deny tcp any host 10.3.1.1
Device(config)# access-list 120 permit ip any any
```

The following example configures a device to redirect web-related packets received via Gigabit Ethernet interface 0/1/0, destined to any host except 209.165.200.224:

```
Device(config)# access-list 100 deny ip any host 209.165.200.224
Device(config)# access-list 100 permit ip any any
Device(config)# ip wccp web-cache redirect-list 100
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip wccp web-cache redirect in
```

Example: WCCP Outbound ACL Check Configuration

The following configuration example shows that the access list prevents traffic from network 10.0.0.0 leaving Gigabit Ethernet interface 0/1/0. Because the outbound ACL check is enabled, WCCP does not redirect that traffic. WCCP checks packets against the ACL before they are redirected.

```
Device(config)# ip wccp web-cache
Device(config)# ip wccp check acl outbound
Device(config)# interface gigabitethernet 0/1/0
Device(config-if)# ip access-group 10 out
Device(config-if)# exit
Device(config)# ip wccp web-cache redirect-list redirect-out
Device(config)# access-list 10 deny 10.0.0.0 0.255.255.255
Device(config)# access-list 10 permit any
```

If the outbound ACL check is disabled, the HTTP packets from network 10.0.0.0 would be redirected to a web cache. Users with that network address could retrieve web pages even though the network administrator wanted to prevent it.

Example: Verifying WCCP Settings

The following example shows how to verify your configuration changes by using the **more system:running-config** command in privileged EXEC mode. The following example shows that both the web cache service and dynamic service 99 are enabled on the device:

```
Device# more system:running-config

 Building configuration...
 Current configuration:
 !
 version 12.0
 service timestamps debug uptime
 service timestamps log uptime
 no service password-encryption
 service udp-small-servers
 service tcp-small-servers
 !
 hostname router4
 !
 enable secret 5 $1$nSVy$faliJsVQXVPW.KuCxZNTh1
 enable password password1
 !
 ip subnet-zero
 ip wccp web-cache
 ip wccp 99
 ip domain-name cisco.com
 ip name-server 10.1.1.1
 ip name-server 10.1.1.2
 ip name-server 10.1.1.3
 !
 !
 !
 interface GigabitEthernet0/1/1
 ip address 10.3.1.2 255.255.255.0
 no ip directed-broadcast
 ip wccp web-cache redirect in
 ip wccp 99 redirect in
 no ip route-cache
 no ip mroute-cache
 !
 interface GigabitEthernet0/1/0
 ip address 10.4.1.1 255.255.255.0
 no ip directed-broadcast
 ip wccp 99 redirect in
 no ip route-cache
 no ip mroute-cache
 !
 interface Serial0
 no ip address
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
 shutdown
 !
 interface Serial1
 no ip address
 no ip directed-broadcast
 no ip route-cache
 no ip mroute-cache
 shutdown
 !
```

```
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
!
!
!
line con 0
transport input none
line aux 0
transport input all
line vty 0 4
password password1
login
!
end
```

The following example shows how to display global statistics related to WCCP:

```
Device# show ip wccp web-cache detail

WCCP Client information:
WCCP Client ID:      10.1.1.2
Protocol Version:    2.0
State:               Usable
Redirection:         L2
Packet Return:       L2
Packets Redirected:  0
Connect Time:        00:20:34
Assignment:          MASK
Mask  SrcAddr    DstAddr    SrcPort DstPort
----  -------    -------    ------- -------
0000: 0x00000000 0x00001741 0x0000  0x0000
Value SrcAddr    DstAddr   SrcPort DstPort CE-IP
----- -------    -------   ------- ------- -----
0000: 0x00000000 0x00000000 0x0000 0x0000 0x3C010102 (10.1.1.2)
0001: 0x00000000 0x00000001 0x0000 0x0000 0x3C010102 (10.1.1.2)
0002: 0x00000000 0x00000040 0x0000 0x0000 0x3C010102 (10.1.1.2)
0003: 0x00000000 0x00000041 0x0000 0x0000 0x3C010102 (10.1.1.2)
0004: 0x00000000 0x00000100 0x0000 0x0000 0x3C010102 (10.1.1.2)
0005: 0x00000000 0x00000101 0x0000 0x0000 0x3C010102 (10.1.1.2)
0006: 0x00000000 0x00000140 0x0000 0x0000 0x3C010102 (10.1.1.2)
```

For more information about the **show ip wccp web-cache** command, see the *Cisco IOS IP Application Services Command Reference*.

Feature Information for WCCP

Table 1: Feature Information for WCCP

Feature Name	Releases	Feature Information
WCCP Support on Cisco Catalyst 9400 Series Swtiches	Cisco IOS XE Everest 16.6.1	The Web Cache Communication Protocol (WCCP) is a Cisco-developed content-routing technology that intercepts IP packets and redirects those packets to a destination other than that specified in the IP packet. WCCP enables you to integrate content engines into your network infrastructure. .