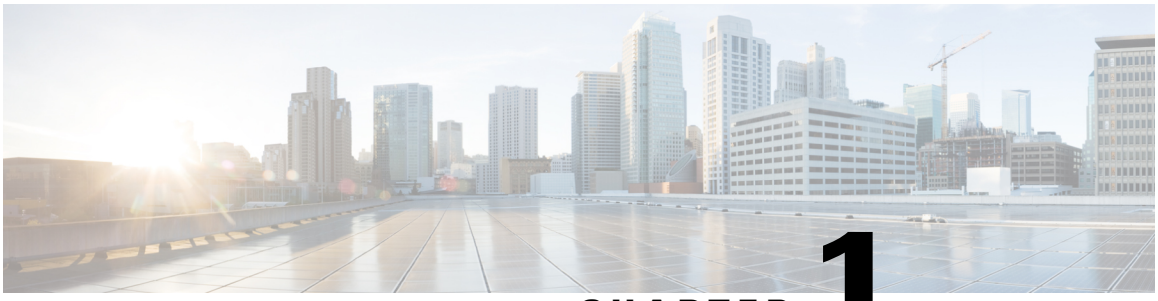
Security Configuration Guide: Cloud Web Security, Cisco IOS Release 15M&T

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Cisco Cloud Web Security

The Cisco Cloud Web Security feature provides content scanning of HTTP and secure HTTP (HTTPS) traffic and malware protection services to web traffic. The feature helps devices transparently redirect HTTP and HTTPS traffic to the Cisco Web Security cloud.

This module describes the Cisco Cloud Web Security feature and how to configure it. This module also describes the Cloud Web Security Tower Telemetry and Default User-Group Support for Authentication features.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Cisco Cloud Web Security

Ensure that both Wide Area Application Services (WAAS) and the content scanning feature are not applied on the same TCP session in the following scenarios:

- When you enable content scanning on an interface that has WAAS configured.
- When the network connection from a branch office to the Internet is over a Multiprotocol Label Switching (MPLS) cloud.

Restrictions for Cisco Cloud Web Security

- The Cloud Web Security encrypted license key changes after each reload. If you have configured the license key option 7 for encryption, you must reenter the key after each reload; or use the unencrypted license key option 0.
- Device-on-a-stick configuration is not supported.
- If Network Address Translation (NAT) is not configured on Cisco Cloud Web Security devices, only 32,000 translation sessions are supported.
- If you configure a host whitelist rule, the sender of an HTTP packet can spoof the Host field of the HTTP header with a whitelisted hostname or whitelist HTTP packets even if the destination HTTP server is not whitelisted. Content scan whitelisting does not verify whether the Host field of an HTTP request matches the destination IP address. Therefore, when providing restricted access to nonauthorized servers, use access control lists (ACLs), which are more effective than whitelists and allow entry to only configured IP addresses.
- If you configure a user agent whitelist rule, the sender of an HTTP packet can spoof the User-Agent field of the HTTP header and the spoofing can result in users accessing a host that is not whitelisted. By using the User-Agent field of the HTTP header, the sender of an HTTP packet can add any HTTP connection request to a whitelist, thus providing unauthorized users access to restricted or nonauthorized servers. Therefore, when providing restricted access to nonauthorized servers, use ACLs, which are more effective than whitelists and allow entry to only configured IP addresses.
- Loadsharing between Cisco Cloud Web Security towers is not supported.
- The web traffic that comes into a branch office is not redirected to Cisco Cloud Web Security for content scanning. Content scanning is configured on the Internet-facing WAN interface, protecting the web traffic that goes out of the branch office.
- When the network connection from a branch office to the Internet is over a Multiprotocol Label Switching (MPLS) cloud, the content scanning feature will not work without split tunneling.
- When Wide-Area Application Services (WAAS) is enabled, the content scanning feature will not work in branch deployments without split tunneling.

Information About Cisco Cloud Web Security

Overview of Cisco Cloud Web Security

The Cisco Cloud Web Security feature provides content scanning of HTTP and secure HTTP (HTTPS) traffic and malware protection service to web traffic. This feature helps devices to transparently redirect HTTP and HTTPS traffic to the cloud. Cloud refers to servers in the Cisco Cloud Web Security data center that are accessible over the public Internet and provide security as a service. Cisco Cloud Web Security servers scan

the web traffic content and either allow or block the traffic based on the configured policies and thus protect clients from malware. Servers use credentials such as private IP addresses, usernames, and user groups to identify and authenticate users and redirect the traffic for content scanning.

This feature enables branch offices to intelligently redirect web traffic to the cloud to enforce security and acceptable use of policies over the web traffic. A device authenticates and identifies users who make web traffic requests by using configured authentication and authorization methods such as user credentials (usernames and user groups) available in the traffic that the device redirects to Cisco Cloud Web Security. Cisco Cloud Web Security uses the user credentials to determine the policies that need to be applied to specific users and for user-based reporting. Cisco Cloud Web Security supports all authentication methods such as HTTP Basic, Web Authorization Proxy, and Windows NT LAN Manager (NTLM) (passive or explicit).

A device that cannot determine a client's credentials uses a default user group name to identify all clients who are connected to a specific interface on that device. Prior to CSCty48221, the user group that was configured using the **user-group** command in parameter-map type inspect configuration mode had precedence over any default user group that was configured using the **user-group default** command in interface configuration mode. With the fix for CSCty48221, a device selects a user group in the following order:

- Authentication methods.
- User group configured using the **user-group default** command on an interface.
- User group configured using the **user-group** command in parameter-map type inspect configuration mode. Configure the **parameter-map type content-scan global** command before configuring the **user-group** command.

You can configure a device in such a way that the approved web traffic does not get scanned by Cisco Cloud Web Security. Instead, the traffic goes directly to the originally requested web server. Clients are any devices that connect to a device, either directly or indirectly. When a client sends an HTTP or HTTPS request, the device receives the request, authenticates the user, and retrieves the group name from the authentication server. The device identifies the user and then consults the whitelist database to determine whether to send the HTTP or HTTPS client response to Cisco Cloud Web Security.

You can configure primary and backup Cisco Cloud Web Security proxy servers. The device regularly polls each of these proxy servers to check their availability.

Whitelists

A whitelist is an approved list of entities that are provided a particular privilege, service, mobility, access, or recognition. Whitelisting means to grant access. You can configure a device in such a way that the approved web traffic does not get redirected to Cisco Cloud Web Security for scanning. When you bypass Cisco Cloud Web Security content scanning, the device retrieves the content directly from the originally requested web server without contacting Cisco Cloud Web Security. Once the device receives a response from the web server, the device sends the data to the client. This process is called whitelisting of web traffic.

You can bypass content scanning based on the following client web traffic properties:

- IP address—You can bypass content scanning for web traffic that matches a configured numbered or named access control list (ACL). Use this method for traffic that is sent to trusted sites, such as intranet servers.
- HTTP-based header fields—You can bypass scanning for web traffic that matches a configured HTTP header field. You can match the host and user agent header fields. Use this method for user agents that do not function properly when scanned or to disable the scanning of traffic that is intended for trusted hosts, such as third-party partners.

Cisco Cloud Web Security Headers

A device that forwards web traffic to Cisco Cloud Web Security proxy servers includes additional HTTP headers in each HTTP and HTTPS request. Cisco Cloud Web Security uses these headers to obtain information about customer deployments, including information about the user who had originally made the client request and the device that sent the request. For security purposes, the information in the headers is encrypted and then hexadecimal encoded.

Cisco Cloud Web Security headers provide both asymmetric cryptography and symmetric cryptography by using industry standard algorithms. Asymmetric encryption is done by using the RSA/ECB/PKCS1Padding algorithm that uses key pairs of 512 bits. Symmetric encryption is done by using the triple "DESede" algorithm with a randomly generated triple Data Encryption Standard (DES) key of 168 bits.

The ISR adds the following CWS HTTP headers:

- X-ScanSafe—This contains a session key that is encrypted using a CWS public key (embedded in the ISR operating system).
- X-ScanSafe-Data—This contains the data CWS needs. It is encrypted with the session key from the X-CWS header.

For example, the headers in a message might look like the following text:

- X-ScanSafe: 35A9C7655CF259C175259A9B980A8DFBF5AC934720BE9374D344F7E584780ECDB9236FF90DF562A79DC4C754C3782E7C3D38C76566F0377D5689E25BD62FC5F
- X-ScanSafe-Data: 8D57AEE5D76432ACAB184AA807D94A7392986FA0D3ED9BEB

Cloud Web Security Tower Telemetry

The Cloud Web Security Tower Telemetry feature:

- Tracks the state of the content scan and the state of the device on which the Cisco Cloud Web Security feature is configured.
- Logs debug messages when delays are encountered while accessing a website.
- Identifies the source of performance issues.

Telemetry is an automated communications process in which measurements are made and data that is collected at remote sites is transmitted to receiving equipment for monitoring.

The device on which the Cisco Cloud Web Security feature is configured is monitored, and data is generated periodically. Because most of these devices do not have a large amount of memory or a secondary storage, the generated data is exported to an external device. For the Cisco Cloud Web Security feature, the generated data is stored in the Cloud Web Security tower. The device connects to a URL hosted by the Cloud Web Security tower by using the HTTP POST method to periodically send telemetry data. This method is called out-of-band telemetry.

Because the Cloud Web Security tower does not have information about all whitelisted traffic, a connector (a persistent, out-of-band secure channel between the device and the Cloud Web Security tower) periodically sends all exception rules configured on the device to the tower. Just like telemetry, the connector makes a POST request and pushes all exception rules to a URL. This URL is separate from the telemetry URL.

The Cloud Web Security tower monitors the TCP session between the client browser and the tower and the TCP session between the tower and the device. The tower also collects debug information at HTTP and TCP levels. The tower also collects information and statistics about the parent HTTP session and all subordinate sessions created by the main URL. The TCP session statistics include retransmission count, window update count, window size, duplicate acknowledgments (ACKs), and time stamps of segment arrival and departure.

Default User-Group Support for Authentication

The Default User-Group Support for Authentication feature redirects unauthorized web traffic to the Cloud Web Security server, also called the tower, for content scanning. Prior to the introduction of this feature, any unauthenticated traffic that fails all login attempts to the Cloud Web Security tower was dropped by the IP admission module and the session was moved to the service-deny state.

For the Default User-Group Support for Authentication feature, the Windows NT LAN Manager (NTLM) acts as the authentication module and updates the user-group database (IP and user-group bindings) with the user-group string that is received as authorization data from the authentication, authorization, and accounting (AAA) or Lightweight Directory Access Protocol (LDAP) servers. Port access control lists (PACLs) perform access control of the web traffic. If no PACL is configured on a port, unauthenticated user traffic is allowed. Even if a user fails the NTLM authentication, the user can be given default access based on your PACL configuration. You can configure a PACL to permit unauthorized users access to the Cloud Web Security tower by using the **permit** command.

The various modules interact with each other to enable the default user-group support, as follows:

- ACL module—Controls port access based on the configured policy.
- Content-Scan—Forwards web traffic from clients to the Cloud Web Security tower for content scanning.
- IP admission or NTLM module—Intercepts the traffic destined to port 80 and port 443 and authenticates users with the Microsoft Active Directory server.
- User-Group database—Maintains the IP and user-group bindings that are received from the LDAP server as part of the authorization data. This database is updated by the IP admission module after the authentication.

How to Configure Cisco Cloud Web Security

In Cisco IOS Release 15.4(2)T, some of the Cloud Web Security commands were replaced by new commands. Releases prior to the Cisco IOS Release 15.4(2)T still use the old commands.

This section consists of tasks that use the commands existing prior to Cisco IOS Release 15.4(2)T and a corresponding task that uses the commands introduced or modified in the Cisco IOS Release 15.4(2)T.

Configuring Whitelisting in Cisco IOS Release 15.4(2)T and Later Releases

Note This task applies to Cisco IOS Release 15.4(2)T and later releases.

User and user-group-based whitelisting is initially done during a TCP synchronization (SYN). No content-scan sessions are created when a session is whitelisted based on an username or user group. The order of whitelisting is: acl, user, user group, header user-agent, header host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cws whitelisting**
4. **whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**}
5. **whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**}
6. **whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**}
7. **end**

DETAILED STEPS

	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	**cws whitelisting** **Example:** `Device(config)# cws whitelisting`	Enables whitelisting of incoming traffic and enters Cloud Web Security whitelisting configuration mode.
Step 4	**whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**} **Example:** `Device(config-cws-wl)# whitelist acl name whitelistedSubnets`	Configures whitelisting of traffic based on the access control list (ACL) or the HTTP request whose header matches the configured regular expression.
Step 5	**whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**} **Example:** `Device(config-cws-wl)# whitelist header host regex whitelistedPatterns`	Configures whitelisting of traffic based on the access control list (ACL) or the HTTP request whose header matches the configured regular expression.

	Command or Action	Purpose
Step 6	**whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**} **Example:** `Device(config-cws-wl)# whitelist user regex whitelistedUsers`	Configures whitelisting of traffic based on the access control list (ACL) or the HTTP request whose header matches the configured regular expression.
Step 7	**end** **Example:** `Device(config-cws-wl)# end`	Exits Cloud Web Security whitelisting configuration mode and returns to privileged EXEC mode.

Configuring Whitelisting

Note This task applies to releases prior to Cisco IOS Release 15.4(2)T.

User and user-group-based whitelisting is initially done during a TCP synchronization (SYN). No content-scan sessions are created when a session is whitelisted based on an username or user group. The order of whitelisting is: acl, user, user group, header user-agent, header host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **content-scan whitelisting**
4. **whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**}
5. **whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**}
6. **whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**content-scan whitelisting** **Example:** `Device(config)# content-scan whitelisting`	Enables whitelisting of incoming traffic and enters content-scan whitelisting configuration mode.
Step 4	**whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**} **Example:** `Device(config-cont-scan-wl)# whitelist acl name whitelistedSubnets`	Configures whitelisting of traffic based on the access control list (ACL) or the HTTP request whose header matches the configured regular expression.
Step 5	**whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**} **Example:** `Device(config-cont-scan-wl)# whitelist header host regex whitelistedPatterns`	Configures whitelisting of traffic based on the access control list (ACL) or the HTTP request whose header matches the configured regular expression.
Step 6	**whitelist** {**acl** {*aclist* | *extended-acl-list* | *acl-name*} | **header** {**host** | **user-agent**} **regex** *regex-host* | **notify-tower**} **Example:** `Device(config-cont-scan-wl)# whitelist user regex whitelistedUsers`	Configures whitelisting of traffic based on the access control list (ACL) or the HTTP request whose header matches the configured regular expression.
Step 7	**end** **Example:** `Device(config-cont-scan-wl)# end`	Exits content-scan whitelisting configuration mode and returns to privileged EXEC mode.

Configuring Cloud Web Security in Cisco IOS Release 15.4(2)T and Later Releases

Note This task applies to Cisco IOS Release 15.4(2)T and later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type cws global**
4. **server primary ipv4** *ip-address* **port http** *port-number* **https** *port-number*
5. **server secondary ipv4** *ip-address* **port http** *port-number* **https** *port-number*
6. **license 7** *license-key*
7. **source interface** *type number*
8. **timeout server** *seconds*
9. **timeout session-inactivity** *seconds*
10. **user-group** *group-name* **username** *username*
11. **server on-failure block-all**
12. **user-group exclude** *user-group*
13. **user-group include** *user-group*
14. **exit**
15. **interface** *type number*
16. **cws out**
17. **ip virtual-reassembly in**
18. **ip virtual-reassembly out**
19. **end**
20. **show cws**

DETAILED STEPS

	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	**parameter-map type cws global** **Example:** `Device(config)# parameter-map type cws global`	Configures a global Cloud Web Security parameter map and enters parameter-map type inspect configuration mode.

	Command or Action	Purpose
Step 4	**server primary ipv4** *ip-address* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server primary ipv4 10.12.34.23 port http 8080 https 8080`	Configures a Cisco Cloud Web Security primary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 5	**server secondary ipv4** *ip-address* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server secondary ipv4 10.21.34.21 port http 8080 https 8080`	Configures a Cisco Cloud Web Security secondary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 6	**license 7** *license-key* **Example:** `Device(config-profile)# license 7 D5D4A545D7A53222E706D1A5D3B5D4E345E5B25737A737B66137242574 25A507`	Configures an encrypted license key that is sent to Cisco Cloud Web Security for authentication.
Step 7	**source interface** *type number* **Example:** `Device(config-profile)# source interface fastethernet 0/2`	Configures the source interface for content scan redirection.
Step 8	**timeout server** *seconds* **Example:** `Device(config-profile)# timeout server 5`	Specifies a server keepalive time in seconds.
Step 9	**timeout session-inactivity** *seconds* **Example:** `Device(config-profile)# timeout session-inactivity 3600`	Specifies the session inactivity time in seconds.
Step 10	**user-group** *group-name* **username** *username* **Example:** `Device(config-profile)# user-group marketing username superuser`	Specifies a default usergroup.
Step 11	**server on-failure block-all** **Example:** `Device(config-profile)# server on-failure block-all`	Blocks all traffic to a web server when communication between the web server and the Cisco Cloud Web Security server fails.

	Command or Action	Purpose
Step 12	**user-group exclude** *user-group* **Example:** `Device(config-profile)# user-group exclude marketing`	Excludes the specified user group.
Step 13	**user-group include** *user-group* **Example:** `Device(config-profile)# user-group include engineering`	Includes the specified user group.
Step 14	**exit** **Example:** `Device(config-profile)# exit`	Exits parameter-map type inspect configuration mode and enters global configuration mode.
Step 15	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Configures an interface and enters interface configuration mode.
Step 16	**cws out** **Example:** `Device(config-if)# cws out`	Configures the egress interface for Cloud Web Security content scanning.
Step 17	**ip virtual-reassembly in** **Example:** `Device(config-if)# ip virtual-reassembly in`	Enables Virtual Fragment Reassembly (VFR) on the ingress.
Step 18	**ip virtual-reassembly out** **Example:** `Device(config-if)# ip virtual-reassembly out`	Enables VFR on the egress.
Step 19	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.
Step 20	**show cws** **Example:** `Device# show cws`	Displays content scanning information.

Example

The following is sample output from the **show cws history** command:

```
Device# show cws history 6

 Protocol  Source              Destination           Bytes          URI
Time
 HTTP      192.168.100.2:1347  209.165.201.4:80    (102:45)       www.google.com
00:01:13
 HTTP      192.168.100.2:1326  209.165.201.6:80    (206:11431)    www.google.com
00:12:55
 HTTP      192.168.100.2:1324  209.165.201.5:80    (206:11449)    www.google.com
00:15:20
 HTTP      192.168.100.2:1318  209.165.201.5:80    (206:11449)    www.google.com
00:17:43
 HTTP      192.168.100.2:1316  209.165.201.4:80    (206:11449)    www.google.com
00:20:04
 HTTP      192.168.100.2:1315  10.254.145.107:80   (575:1547)     alert.scansafe.net
00:21:32
```

Configuring Cisco Cloud Web Security

Note This task applies to releases prior to Cisco IOS Release 15.4(2)T.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type content-scan global**
4. **server scansafe primary ipv4** *ip-address* **port http** *port-number* **https** *port-number*
5. **server scansafe secondary ipv4** *ip-address* **port http** *port-number* **https** *port-number*
6. **license 7** *license-key*
7. **source interface** *type number*
8. **timeout server** *seconds*
9. **timeout session-inactivity** *seconds*
10. **user-group** *group-name* **username** *username*
11. **server scansafe on-failure block-all**
12. **user-group exclude** *user-group*
13. **user-group include** *user-group*
14. **exit**
15. **interface** *type number*
16. **content-scan out**
17. **ip virtual-reassembly in**
18. **ip virtual-reassembly out**
19. **end**
20. **show content-scan**

DETAILED STEPS

	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	**parameter-map type content-scan global** **Example:** `Device(config)# parameter-map type content-scan global`	Configures a global content-scan parameter map and enters parameter-map type inspect configuration mode.
Step 4	**server scansafe primary ipv4** *ip-address* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server scansafe primary ipv4 10.12.34.23 port http 8080 https 8080`	Configures a Cisco Cloud Web Security primary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 5	**server scansafe secondary ipv4** *ip-address* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server scansafe secondary ipv4 10.21.34.21 port http 8080 https 8080`	Configures a Cisco Cloud Web Security secondary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 6	**license 7** *license-key* **Example:** `Device(config-profile)# license 7 D5D4A545D7A53222E706D1A5D3B5D4E345E5B25737A737B6613724257425A507`	Configures an encrypted license key that is sent to Cisco Cloud Web Security for authentication.
Step 7	**source interface** *type number* **Example:** `Device(config-profile)# source interface fastethernet 0/2`	Configures the source interface for content scan redirection.

	Command or Action	Purpose
Step 8	**timeout server** *seconds* **Example:** `Device(config-profile)# timeout server 5`	Specifies a server keepalive time in seconds.
Step 9	**timeout session-inactivity** *seconds* **Example:** `Device(config-profile)# timeout session-inactivity 3600`	Specifies the session inactivity time in seconds.
Step 10	**user-group** *group-name* **username** *username* **Example:** `Device(config-profile)# user-group marketing username superuser`	Specifies a default usergroup.
Step 11	**server scansafe on-failure block-all** **Example:** `Device(config-profile)# server scansafe on-failure block-all`	Blocks all traffic to a web server when communication between the web server and the Cisco Cloud Web Security server fails.
Step 12	**user-group exclude** *user-group* **Example:** `Device(config-profile)# user-group exclude marketing`	Excludes the specified user group.
Step 13	**user-group include** *user-group* **Example:** `Device(config-profile)# user-group include engineering`	Includes the specified user group.
Step 14	**exit** **Example:** `Device(config-profile)# exit`	Exits parameter-map type inspect configuration mode and enters global configuration mode.
Step 15	**interface** *type number* **Example:** `Device(config)# interface ethernet 0/0`	Configures an interface and enters interface configuration mode.
Step 16	**content-scan out** **Example:** `Device(config-if)# content-scan out`	Configures the egress interface for content scanning.
Step 17	**ip virtual-reassembly in** **Example:** `Device(config-if)# ip virtual-reassembly in`	Enables Virtual Fragment Reassembly (VFR) on the ingress.

	Command or Action	Purpose
Step 18	**ip virtual-reassembly out** **Example:** `Device(config-if)# ip virtual-reassembly out`	Enables VFR on the egress.
Step 19	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.
Step 20	**show content-scan** **Example:** `Device# show content-scan`	Displays content scanning information.

Example

The following is sample output from the **show content-scan history** command:

```
Device# show content-scan history 6

 Protocol  Source              Destination           Bytes          URI
Time
 HTTP      192.168.100.2:1347  209.165.201.4:80    (102:45)       www.google.com
00:01:13
 HTTP      192.168.100.2:1326  209.165.201.6:80    (206:11431)    www.google.com
00:12:55
 HTTP      192.168.100.2:1324  209.165.201.5:80    (206:11449)    www.google.com
00:15:20
 HTTP      192.168.100.2:1318  209.165.201.5:80    (206:11449)    www.google.com
00:17:43
 HTTP      192.168.100.2:1316  209.165.201.4:80    (206:11449)    www.google.com
00:20:04
 HTTP      192.168.100.2:1315  10.254.145.107:80   (575:1547)     alert.scansafe.net
00:21:32
```

Enabling Out-of-Band Telemetry in Cisco IOS Release 15.4(2)T and Later Releases

Note This task applies to Cisco IOS Release 15.4(2)T and later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type cws global**
4. **out-of-band telemetry interval** *interval*
5. **end**

DETAILED STEPS

	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	**parameter-map type cws global** **Example:** `Device(config)# parameter-map type cws global`	Configures a global Cloud Web Security parameter map and enters parameter-map type inspect configuration.
Step 4	**out-of-band telemetry interval** *interval* **Example:** `Device(config-profile)# out-of-band telemetry interval 60`	Enables out-of-band telemetry and content-scan exception rules.
Step 5	**end** **Example:** `Device(config-profile)# end`	Exits parameter-map type inspect configuration mode and returns to privileged EXEC mode.

Enabling Out-of-Band Telemetry

Perform this task to enable the storing of content scan data in the Cloud Web Security server:

Note This task applies to releases prior to Cisco IOS Release 15.4(2)T.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type content-scan global**
4. **out-of-band telemetry interval** *interval*
5. **end**

DETAILED STEPS

	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	**parameter-map type content-scan global** **Example:** `Device(config)# parameter-map type content-scan global`	Configures a global content-scan parameter map and enters parameter-map type inspect configuration.
Step 4	**out-of-band telemetry interval** *interval* **Example:** `Device(config-profile)# out-of-band telemetry interval 60`	Enables out-of-band telemetry and content-scan exception rules.
Step 5	**end** **Example:** `Device(config-profile)# end`	Exits parameter-map type inspect configuration mode and returns to privileged EXEC mode.

Configuration Examples for Cisco Cloud Web Security

Example: Configuring Whitelisting in Cisco IOS Release 15.4(2)T

Note This example applies to Cisco IOS Release 15.4(2)T and later releases.

```
Device# configure terminal
Device(config)# cws whitelisting
Device(config-cws-wl)# whitelist header host regex whitelistedPatterns
Device(config-cws-wl)# whitelist acl name whitelistedSubnets
Device(config-cws-wl)# whitelist user regex whitelistedUsers
Device(config-cws-wl)# whitelist user-group regex whitelistedUserGroups
Device(config-cws-wl)# end
```

Example: Configuring Whitelisting

Note This example applies to releases prior to Cisco IOS Release 15.4(2)T.

```
Device# configure terminal
Device(config)# content-scan whitelisting
Device(config-cont-scan-wl)# whitelist header host regex whitelistedPatterns
Device(config-cont-scan-wl)# whitelist acl name whitelistedSubnets
Device(config-cont-scan-wl)# whitelist user regex whitelistedUsers
Device(config-cont-scan-wl)# whitelist user-group regex whitelistedUserGroups
Device(config-cont-scan-wl)# end
```

Example: Configuring Cisco Cloud Web Security in Cisco IOS Release 15.4(2)T and Later Releases

Note This example applies to Cisco IOS Release 15.4(2)T and later releases.

In the following example, the Cloud Web Security server IP address is used. The default action in this example is to block all traffic if Cloud Web Security servers are not reachable. This means that if both the primary and secondary Cloud Web Security servers are unreachable, users are not be able to access the Internet.

```
Device# configure terminal
Device(config)# parameter-map type cws
Device(config-profile)# server primary ipv4 10.12.34.23 port http 8080 https 8080
Device(config-profile)# server secondary ipv4 10.21.34.21 port http 8080 https 8080
Device(config-profile)# license 7
D5D4A545D7A53222E706D1A5D3B5D4E345E5B25737A737B6613724257425A507
Device(config-profile)# source interface fastethernet 0/2
Device(config-profile)# timeout server 5
Device(config-profile)# timeout session-inactivity 3600
Device(config-profile)# user-group marketing username superuser
Device(config-profile)# server on-failure block-all
Device(config-profile)# user-group exclude marketing
Device(config-profile)# user-group include engineering
```

```
Device(config-profile)# exit
Device(config)# interface ethernet 0/0
Device(config-if)# cws out
Device(config-if)# ip virtual-assembly in
Device(config-if)# ip virtual-assembly out
Device(config-if)# end
```

In the following example, the Cloud Web Security server name is used instead of the tower IP address. The secure HTTP (HTTPS) port is not specified, which means that all HTTPS traffic will be whitelisted. The default action in this example is to block all traffic if Cloud Web Security servers are not reachable.

Note Use the tower IP address over the name for faster lookups.

```
Device# configure terminal
Device(config)# parameter-map type cws
Device(config-profile)# server primary name proxy123.scansafe.net port http 8080
Device(config-profile)# server secondary name proxy456.scansafe.net port http 8080
Device(config-profile)# license 0 AA0123456789012345678901234567890123456789
Device(config-profile)# source interface GigabitEthernet 0/0
Device(config-profile)# timeout server 30
Device(config-profile)# user-group ciscogroup username ciscouser
Device(config-profile)# server on-failure block-all
Device(config-profile)# exit
```

Example: Configuring Cisco Cloud Web Security

Note This example applies to releases prior to Cisco IOS Release 15.4(2)T.

In the following example, the Cloud Web Security server IP address is used. The default action in this example is to block all traffic if Cloud Web Security servers are not reachable. This means that if both the primary and secondary Cloud Web Security servers are unreachable, users are not be able to access the Internet.

```
Device# configure terminal
Device(config)# parameter-map type content-scan
Device(config-profile)# server scansafe primary ipv4 10.12.34.23 port http 8080 https 8080
Device(config-profile)# server scansafe secondary ipv4 10.21.34.21 port http 8080 https
8080
Device(config-profile)# license 7
D5D4A545D7A53222E706D1A5D3B5D4E345E5B25737A737B6613724257425A507
Device(config-profile)# source interface fastethernet 0/2
Device(config-profile)# timeout server 5
Device(config-profile)# timeout session-inactivity 3600
Device(config-profile)# user-group marketing username superuser
Device(config-profile)# server scansafe on-failure block-all
Device(config-profile)# user-group exclude marketing
Device(config-profile)# user-group include engineering
Device(config-profile)# exit
Device(config)# interface ethernet 0/0
Device(config-if)# content-scan out
Device(config-if)# ip virtual-assembly in
Device(config-if)# ip virtual-assembly out
Device(config-if)# end
```

Example: Enabling Out-of-Band Telemetry in Cisco IOS Release 15.4(2)T and Later Releases

Note This example applies to Cisco IOS Release 15.4(2)T and later releases.

```
Device# configure terminal
Device(config)# parameter-map type cws global
Device(config-profile)# out-of-band telemetry interval 60
Device(config-profile)# end
```

Example: Enabling Out-of-Band Telemetry

Note This example applies to releases prior to Cisco IOS Release 15.4(2)T.

```
Device# configure terminal
Device(config)# parameter-map type content-scan global
Device(config-profile)# out-of-band telemetry interval 60
Device(config-profile)# end
```

Additional References for Cisco Cloud Web Security

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Firewall commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
Cisco Cloud Web Security solution guide	*Cisco ISR Web Security with Cisco ScanSafe Solution Guide*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

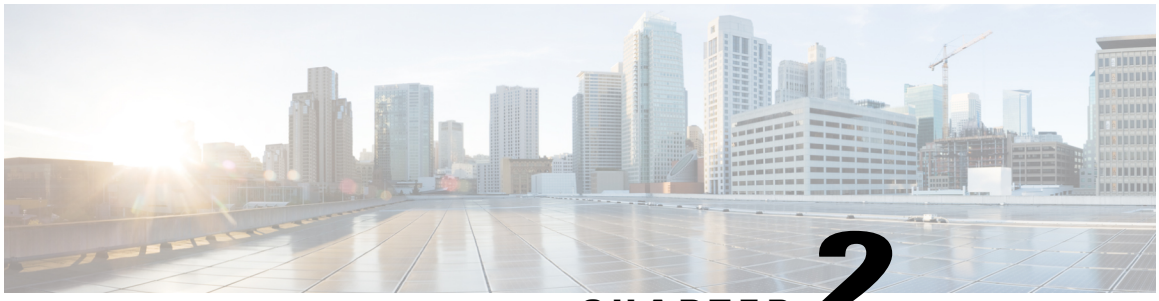
Feature Information for Cisco Cloud Web Security

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Cisco Cloud Web Security

Feature Name	Releases	Feature Information
Cisco Cloud Web Security	15.2(1)T1 15.2(4)M 15.4(2)T	The Cisco Cloud Web Security feature provides content scanning of HTTP and HTTPS traffic and malware protection services to web traffic. This feature helps a device transparently redirect HTTP and HTTPS traffic to the Cisco Web Security cloud. The following commands were introduced or modified: **clear content-scan**, **content-scan out**, **content-scan whitelisting**, **debug content-scan**, **ip admission name http-basic**, **ip admission name method-list**, **ip admission name ntlm**, **ip admission name order**, **ip admission virtual-ip**, **license (parameter-map)**, **logging (parameter-map)**, **parameter-map type content-scan global**, **publickey**, **server scan-safe**, **show content-scan**, **show ip admission**, **source (parameter-map)**, **timeout (parameter-map)**, **user-group (parameter-map)**, **whitelist**. In Cisco IOS Release 15.4(2)T, the following commands were replaced by new commands: • **clear content-scan** was replaced by the **cws content-scan** • **content-scan out** was replaced by the **cws out** • **content-scan whitelisting** was replaced by the **cws whitelisting** • **parameter-map type content-scan global** was replaced by the **parameter-map type cws global** • **server scan-safe** was replaced by the **server** • **show content-scan** was replaced by the **show cws**
Cloud Web Security Tower Telemetry	15.3(3)M 15.4(2)T	The Cloud Web Security Tower Telemetry feature: • Tracks the state of the content scan and the state of the device on which the Cisco Cloud Web Security feature is configured. • Logs debug messages when delays are encountered while accessing a website. • Identifies the source of performance issues. The following commands were introduced or modified: **out-of-band telemetry** and **test content-scan**. The **test content-scan** command was replaced by the **test cws** command in Cisco IOS Release 15.4(2)T.
Default User-Group Support for Authentication	15.3(3)M	The Default User-Group Support for Authentication feature redirects unauthorized web traffic to the Cloud Web Security server for content scanning.



CHAPTER 2

VRF-Aware Cloud Web Security

The VRF-Aware Cloud Web Security feature adds virtual routing and forwarding (VRF) support to the Cisco Cloud Web Security configuration. VRF instances in IP-based networks enable a device to have multiple instances of the routing table at the same time. Because routing instances are independent of each other, they can use the same IP addresses without any conflict.

This feature describes the VRF-Aware Cloud Web Security feature and explains how to configure it.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for VRF-Aware Cloud Web Security

- While enabling a virtual routing and forwarding (VRF) instance on a device, configure the **content-scan out** command only on one interface to ensure that the tower polling mechanism is consistent.
- The VRF-Aware Cloud Web Security feature works only in VRF-Lite scenarios.

- Overlapping IP addresses must be resolved if multiple VRF instances converge into a single VRF.

Information About VRF-Aware Cloud Web Security

VRF-Aware Cloud Web Security Overview

Cisco Cloud Web Security provides content scanning of HTTP and secure HTTP (HTTPS) traffic and malware protection services to web traffic. It also helps devices transparently redirect HTTP and HTTPS traffic to the Cisco Web Security cloud. The VRF-Aware Cloud Web Security feature adds virtual routing and forwarding (VRF) support to Cisco Cloud Web Security.

VRF instances in IP-based networks enable a device to have multiple instances of the routing table at the same time. Because routing instances are independent of each other, they use the same IP addresses without any conflict.

You can use VRFs with or without Multiprotocol Label Switching (MPLS). When VRFs are used without MPLS, it is called VRF-Lite. The VRF-Aware Cloud Web Security feature works only in VRF-Lite scenarios.

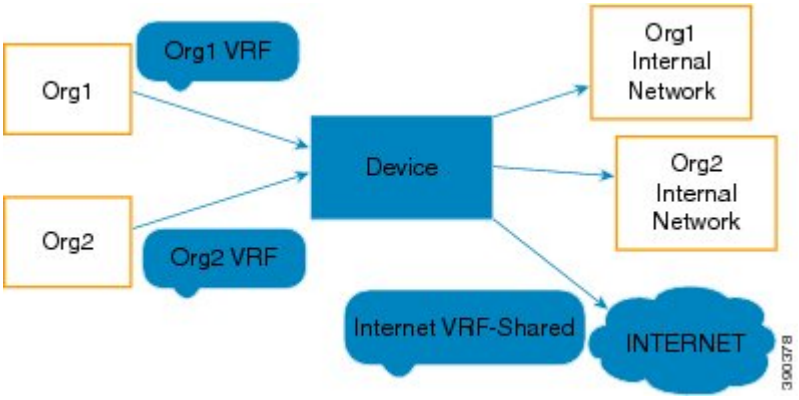
During content scan, the egress VRF ID of the interface on which the **content-scan out** command is configured is used. The VRF ID that is used during communication with the Cloud Web Security tower is same as the VRF ID of the interface on which the **content-scan out** command is configured. Based on your configuration, include the routes configured in the Cloud Web Security tower in the appropriate VRFs.

The whitelisted traffic flows through the interface on which the VRF that is connected to the Internet is configured. A whitelist is an approved list of entities that are provided a particular privilege, service, mobility, access, or recognition. Whitelisting means to grant access.

VRF-Aware Cloud Web Security Scenarios

This section describes some scenarios in which the VRF-Aware Cloud Web Security is configured:

Figure 1: VRF-Aware Cloud Web Security: Scenario 1

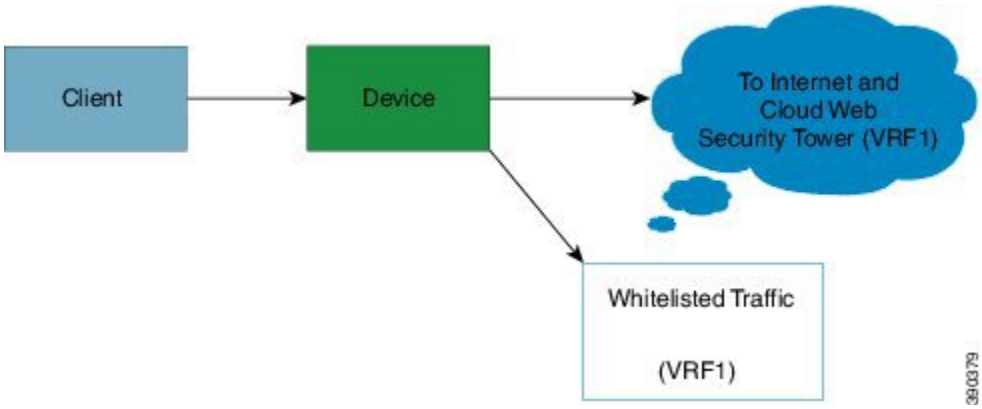


In the illustration above, there are two separate networks, Org1 and Org2. The device provides connectivity to the Internet as a shared service between these organizations. Because each organization has a separate

virtual routing and forwarding (VRF) instance, both have their individual routing table entries. The clients on Org1 and Org2 can both have the same IP addresses and still access the internal network of their organization.

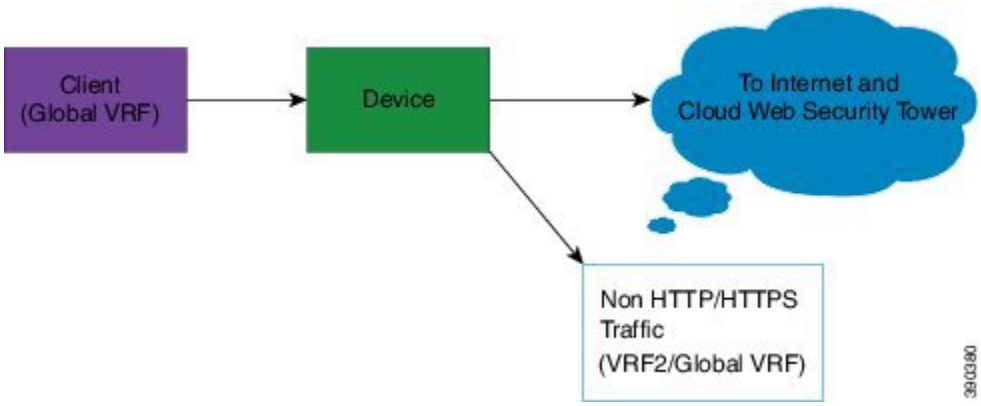
Because the Internet VRF is shared, Network Address Translation (NAT) must be configured to distinguish the traffic from both the networks. Also, the respective routes from Org1 and Org2 must be advertised into the Internet VRF and vice versa, for proper routing of traffic. In Scenario 1, you can enable Cisco Cloud Web Security on the VRF-shared Internet. Enabling Cisco Cloud Web Security ensures that the HTTP and secure HTTP (HTTPS) traffic is redirected to the configured Cloud Web Security tower. Traffic is passed to the internal networks of both organizations through whitelisting.

Figure 2: VRF-Aware Cloud Web Security: Scenario 2



In the illustration above, clients belong to a global VRF. The Internet traffic belongs to another VRF, VRF1. Whitelisted traffic also uses VRF1 because the interface that is configured for content scan must be connected to whitelisted sites. When you configure content scan on interfaces, each interface will have a unique VRF.

Figure 3: VRF-Aware Cloud Web Security: Scenario 3



In the illustration above, the client traffic comes into the global VRF. All HTTP and HTTPS traffic is sent to VRF1, and non-HTTP and non-HTTPS traffic is sent to VRF2/global VRF. Content scan redirects the HTTP/HTTPS traffic to the Cloud Web Security tower. The classification of HTTP/HTTPS traffic must be done before content-scan redirection.

How to Configure VRF-Aware Cloud Web Security

In Cisco IOS Release 15.4(2)T, some of the Cloud Web Security commands were replaced by new commands. Releases prior to Cisco IOS Release 15.4(2)T still use the old commands.

This section consists of tasks that use the commands existing prior to Cisco IOS Release 15.4(2)T and a corresponding task that uses the commands introduced or modified in the Cisco IOS Release 15.4(2)T.

Configuring a Cloud Web Security Tower in Cisco IOS Release 15.4(2)T and Later Releases

Note This task applies to Cisco IOS Release 15.4(2)T and later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type cws global**
4. **server primary ipv4** *ipv4-address* **port http** *port-number* **https** *port-number*
5. **server secondary name** *name* **port http** *port-number* **https** *port-number*
6. **license** {**0** | **7**} *authentication-key*
7. **source address ipv4** *ipv4-address*
8. **timeout server** *seconds*
9. **timeout session-inactivity** *seconds*
10. **user-group** *name* [**username** *name*]
11. **server on-failure** {**allow-all** | **block-all**}
12. **end**

DETAILED STEPS

	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.

	Command or Action	Purpose
Step 3	**parameter-map type cws global** **Example:** `Device(config)# parameter-map type cws global`	Configures a global Cloud Web Security parameter map and enters parameter-map type inspect configuration mode.
Step 4	**server primary ipv4** *ipv4-address* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server primary ipv4 10.2.2.2 port http 8080 https 8080`	Configures a Cisco Cloud Web Security primary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 5	**server secondary name** *name* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server secondary name example1363.example.net port http 8080 https 8080`	Configures a Cisco Cloud Web Security secondary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 6	**license** {**0** | **7**} *authentication-key* **Example:** `Device(config-profile)# license 0 F52409C9DAF22005CF33E64A7BC524C9`	Configures an unencrypted license key that is sent to Cisco Cloud Web Security for authentication. • To configure an encrypted license key, use the **7** keyword and specify an authentication key of 66 hexadecimal characters.
Step 7	**source address ipv4** *ipv4-address* **Example:** `Device(config-profile)# source address ipv4 192.168.4.4`	Configures the source address for content scan redirection.
Step 8	**timeout server** *seconds* **Example:** `Device(config-profile)# timeout server 20`	Specifies a server keepalive time in seconds.
Step 9	**timeout session-inactivity** *seconds* **Example:** `Device(config-profile)# timeout session-inactivity 180`	Specifies the session inactivity time in seconds.
Step 10	**user-group** *name* [**username** *name*] **Example:** `Device(config-profile)# user-group group1 username user1`	Specifies a default user group.

	Command or Action	Purpose
Step 11	**server on-failure** {**allow-all** | **block-all**} **Example:** `Device(config-profile)# server on-failure block-all`	Blocks all traffic to a web server when communication between the web server and the Cisco Cloud Web Security server fails.
Step 12	**end** **Example:** `Device(config-profile)# end`	Exits parameter-map type inspect configuration mode and returns to privileged EXEC mode.

Configuring a Cloud Web Security Tower

Note This task applies to releases prior to Cisco IOS Release 15.4(2)T.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type content-scan global**
4. **server scansafe primary ipv4** *ipv4-address* **port http** *port-number* **https** *port-number*
5. **server scansafe secondary name** *name* **port http** *port-number* **https** *port-number*
6. **license** {**0** | **7**} *authentication-key*
7. **source address ipv4** *ipv4-address*
8. **timeout server** *seconds*
9. **timeout session-inactivity** *seconds*
10. **user-group** *name* [**username** *name*]
11. **server scansafe on-failure** {**allow-all** | **block-all**}
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**parameter-map type content-scan global** **Example:** `Device(config)# parameter-map type content-scan global`	Configures a global content-scan parameter map and enters parameter-map type inspect configuration mode.
Step 4	**server scansafe primary ipv4** *ipv4-address* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server scansafe primary ipv4 10.2.2.2 port http 8080 https 8080`	Configures a Cisco Cloud Web Security primary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 5	**server scansafe secondary name** *name* **port http** *port-number* **https** *port-number* **Example:** `Device(config-profile)# server scansafe secondary name example1363.example.net port http 8080 https 8080`	Configures a Cisco Cloud Web Security secondary server for content scanning. • The default Cisco Cloud Web Security port for the proxied HTTP and HTTPS traffic is 8080. • You can use either the HTTP port or the HTTPS port or both.
Step 6	**license** {**0** | **7**} *authentication-key* **Example:** `Device(config-profile)# license 0 F52409C9DAF22005CF33E64A7BC524C9`	Configures an unencrypted license key that is sent to Cisco Cloud Web Security for authentication. • To configure an encrypted license key, use the **7** keyword and specify an authentication key of 66 hexadecimal characters.
Step 7	**source address ipv4** *ipv4-address* **Example:** `Device(config-profile)# source address ipv4 192.168.4.4`	Configures the source address for content scan redirection.
Step 8	**timeout server** *seconds* **Example:** `Device(config-profile)# timeout server 20`	Specifies a server keepalive time in seconds.
Step 9	**timeout session-inactivity** *seconds* **Example:** `Device(config-profile)# timeout session-inactivity 180`	Specifies the session inactivity time in seconds.

	Command or Action	Purpose
Step 10	**user-group** *name* [**username** *name*] **Example:** `Device(config-profile)# user-group group1 username user1`	Specifies a default user group.
Step 11	**server scansafe on-failure** {**allow-all** | **block-all**} **Example:** `Device(config-profile)# server scansafe on-failure block-all`	Blocks all traffic to a web server when communication between the web server and the Cisco Cloud Web Security server fails.
Step 12	**end** **Example:** `Device(config-profile)# end`	Exits parameter-map type inspect configuration mode and returns to privileged EXEC mode.

Configuring VRF-Aware Cloud Web Security in Cisco IOS Release 15.4(2)T and Later Releases

Note This task applies to Cisco IOS Release 15.4(2)T and later releases.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip vrf** *vrf-name*
4. **exit**
5. **interface** *type number*
6. **ip vrf forwarding** *name*
7. **ip address** *ip-address mask*
8. **cws out**
9. **ip virtual-reassembly in**
10. **ip virtual-reassembly out**
11. **duplex auto**
12. **speed auto**
13. **end**

DETAILED STEPS

	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 vrf** *vrf-name* **Example:** `Device(config)# ip vrf output`	Defines a virtual routing and forwarding (VRF) instance and enters VRF configuration mode.
Step 4	**exit** **Example:** `Device(config-vrf)# exit`	Exits VRF configuration mode and enters global configuration mode.
Step 5	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0`	Configures an interface and enters interface configuration mode.
Step 6	**ip vrf forwarding** *name* **Example:** `Device(config-if)# ip vrf forwarding output`	Associates a VRF instance and configures a VRF forwarding table on an interface.
Step 7	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 192.168.4.4 255.255.255.0`	Configures an IP address for an interface.
Step 8	**cws out** **Example:** `Device(config-if)# cws out`	Configures the egress interface for Cloud Web Security content scanning.
Step 9	**ip virtual-reassembly in** **Example:** `Device(config-if)# ip virtual-reassembly in`	Enables Virtual Fragment Reassembly (VFR) on the ingress.
Step 10	**ip virtual-reassembly out** **Example:** `Device(config-if)# ip virtual-reassembly out`	Enables VRF on the egress.

	Command or Action	Purpose
Step 11	**duplex auto** **Example:** `Device(config-if)# duplex auto`	Enables autonegotiation on an interface.
Step 12	**speed auto** **Example:** `Device(config-if)# speed auto`	Configures the speed of an interface.
Step 13	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

Configuring VRF-Aware Cloud Web Security

Note This task applies to releases prior to Cisco IOS Release 15.4(2)T.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip vrf** *vrf-name*
4. **exit**
5. **interface** *type number*
6. **ip vrf forwarding** *name*
7. **ip address** *ip-address mask*
8. **content-scan out**
9. **ip virtual-reassembly in**
10. **ip virtual-reassembly out**
11. **duplex auto**
12. **speed auto**
13. **end**

DETAILED STEPS

	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 vrf** *vrf-name* **Example:** `Device(config)# ip vrf output`	Defines a virtual routing and forwarding (VRF) instance and enters VRF configuration mode.
Step 4	**exit** **Example:** `Device(config-vrf)# exit`	Exits VRF configuration mode and enters global configuration mode.
Step 5	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0`	Configures an interface and enters interface configuration mode.
Step 6	**ip vrf forwarding** *name* **Example:** `Device(config-if)# ip vrf forwarding output`	Associates a VRF instance and configures a VRF forwarding table on an interface.
Step 7	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 192.168.4.4 255.255.255.0`	Configures an IP address for an interface.
Step 8	**content-scan out** **Example:** `Device(config-if)# content-scan out`	Configures the egress interface for content scanning.
Step 9	**ip virtual-reassembly in** **Example:** `Device(config-if)# ip virtual-reassembly in`	Enables Virtual Fragment Reassembly (VFR) on the ingress.
Step 10	**ip virtual-reassembly out** **Example:** `Device(config-if)# ip virtual-reassembly out`	Enables VRF on the egress.

	Command or Action	Purpose
Step 11	**duplex auto** **Example:** `Device(config-if)# duplex auto`	Enables autonegotiation on an interface.
Step 12	**speed auto** **Example:** `Device(config-if)# speed auto`	Configures the speed of an interface.
Step 13	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

Configuration Examples for VRF-Aware Cloud Web Security

Example: Configuring a Cloud Web Security Tower in Cisco IOS Release 15.4(2)T and Later Releases

Note This example applies to Cisco IOS Release 15.4(2)T and later releases.

```
Device# configure terminal
Device(config)# parameter-map type cws global
Device(config-profile)# server primary ipv4 10.2.2.2 port http 8080 https 8080
Device(config-profile)# server secondary name example1363.example.net port http 8080 https
 8080
Device(config-profile)# license 0 F52409C9DAF22005CF33E64A7BC524C9
Device(config-profile)# source address ipv4 192.168.4.4
Device(config-profile)# timeout server 20
Device(config-profile)# timeout session-inactivity 180
Device(config-profile)# user-group group1 username user1
Device(config-profile)# server on-failure block-all
Device(config-profile)# end
```

Example: Configuring a Cloud Web Security Tower

Note This example applies to releases prior to Cisco IOS Release 15.4(2)T.

```
Device# configure terminal
Device(config)# parameter-map type content-scan global
Device(config-profile)# server scansafe primary ipv4 10.2.2.2 port http 8080 https 8080
Device(config-profile)# server scansafe secondary name example1363.example.net port http
```

```
8080 https 8080
Device(config-profile)# license 0 F52409C9DAF22005CF33E64A7BC524C9
Device(config-profile)# source address ipv4 192.168.4.4
Device(config-profile)# timeout server 20
Device(config-profile)# timeout session-inactivity 180
Device(config-profile)# user-group group1 username user1
Device(config-profile)# server scansafe on-failure block-all
Device(config-profile)# end
```

Example: VRF-Aware Cloud Web Security in Cisco IOS Release 15.4(2)T and Later Releases

Note This example applies to Cisco IOS Release 15.4(2)T and later releases.

```
Device# configure terminal
Device(config)# ip vrf output
Device(config-vrf)# exit
Device(config)# interface gigabitethernet 0/0
Device(config-if)# ip vrf forwarding output
Device(config-if)# ip address 192.168.4.4 255.255.255.0
Device(config-if)# cws out
Device(config-if)# ip virtual-reassembly in
Device(config-if)# ip virtual-reassembly out
Device(config-if)# duplex auto
Device(config-if)# speed auto
Device(config-if)# end
```

Example: Configuring VRF-Aware Cloud Web Security

Note This example applies to releases prior to Cisco IOS Release 15.4(2)T.

```
Device# configure terminal
Device(config)# ip vrf output
Device(config-vrf)# exit
Device(config)# interface gigabitethernet 0/0
Device(config-if)# ip vrf forwarding output
Device(config-if)# ip address 192.168.4.4 255.255.255.0
Device(config-if)# content-scan out
Device(config-if)# ip virtual-reassembly in
Device(config-if)# ip virtual-reassembly out
Device(config-if)# duplex auto
Device(config-if)# speed auto
Device(config-if)# end
```

Additional References for VRF-Aware Cloud Web Security

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Related Topic	Document Title
Firewall commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

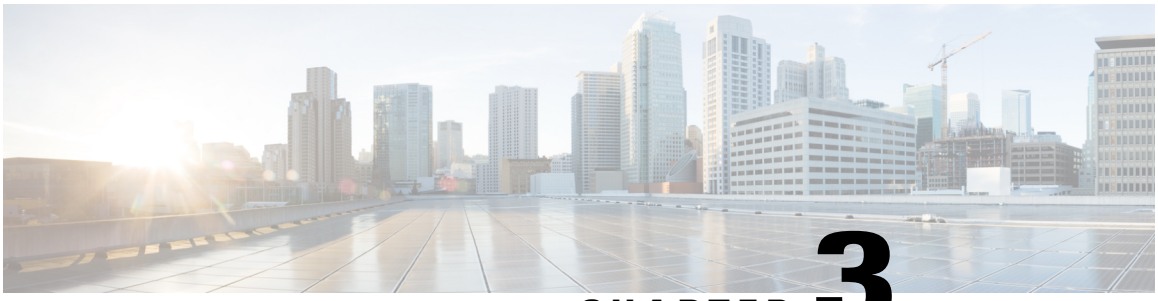
Feature Information for VRF-Aware Cloud Web Security

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 2: Feature Information for VRF-Aware Cloud Web Security

Feature Name	Releases	Feature Information
VRF-Aware Cloud Web Security	15.4(1)T 15.4(2)T	The VRF-Aware Cloud Web Security feature adds virtual routing and forwarding (VRF) support to the Cisco Cloud Web Security configuration. VRF instances in IP-based networks enable a device to have multiple instances of the routing table at the same time. Because routing instances are independent of each other, they can use the same IP addresses without any conflict. The following command was introduced or modified: **show content-scan**. In Cisco IOS Release 15.4(2)T, the **show content-scan** command was replaced by the **show cws** command.



CHAPTER 3

Browser-Based Authentication Bypass

The Browser-Based Authentication Bypass feature enables web browsers to bypass authentication methods such as HTTP Basic, Web Authorization Proxy, and Windows NT LAN Manager (NTLM) (passive or explicit). Specific web browsers can be configured for authentication, and other browsers can be configured to bypass authentication.

This module provides information about the feature and how to configure it.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Browser-Based Authentication Bypass

- You must configure at least one of these authentication methods—HTTP Basic, Web Authorization Proxy, or Windows NTLM—with browser-based authentication bypass.
- Use browser-based authentication bypass with the Default User-Group Policy feature.

Information About Browser-Based Authentication Bypass

Browser-Based Authentication Bypass Overview

While using web browsers, as part of the user authentication, a pop-up or dialog box appears in some web browsers. The Browser-Based Authentication Bypass feature helps to bypass this user authentication and thus avoid the authentication pop-ups.

With the Browser-Based Authentication Bypass feature, you can configure web browsers that must be authenticated and browsers that can bypass user authentication. Bypassing is supported for authentication methods such as HTTP Basic, Web Authorization Proxy, and Windows NT LAN Manager (NTLM) (passive or explicit).

The Browser-Based Authentication Bypass feature supports the following web browsers:

- Chrome
- Firefox
- Internet Explorer 8 (IE8)
- IE9
- Safari

A network administrator configures a list of regular expression (regex) patterns in the IP admission module. When the IP admission module receives the HTTP Get request, the module compares the user-agent string in the HTTP header to the regex pattern that the administrator has configured for the bypass method.

The following rules apply to the Browser-Based Authentication Bypass feature:

- If a configured regex pattern does not match the user-agent field, a web browser is authenticated on the basis of the configured web authentication method.
- If a configured regex pattern matches the user-agent field, authentication is bypassed for the web browser and the HTTP traffic goes through to the Cisco Web Security cloud.

How to Configure Browser-Based Authentication Bypass

Configuring Browser-Based Authentication Bypass

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type regex** *regex-map*
4. **pattern** *expression*
5. **exit**
6. **ip admission name** *admission-name* **bypass regex** *regex-map* [**absolute-timer** *minutes*]
7. Perform one of the following tasks:
   - **ip admission name** *admission-name* **ntlm**
   - **ip admission name** *admission-name* **http-basic**
   - **ip admission name** *admission-name* **proxy http**
8. **interface** *type number*
9. **ip admission** *admission-name*
10. **end**

DETAILED STEPS

	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	**parameter-map type regex** *regex-map* **Example:** `Device(config)# parameter-map type regex regex-map1`	Configures a parameter-map type with a regular expression (regex) to match a specific traffic pattern and enters parameter-map type inspect configuration mode.

	Command or Action	Purpose
Step 4	**pattern** *expression* **Example:** `Device(config-profile)# pattern Chrome`	Configures a matching pattern that compares the user-agent field in the HTTP Get request and the regex pattern.
Step 5	**exit** **Example:** `Device(config-profile)# exit`	Exits parameter-map type inspect configuration mode and enters global configuration mode.
Step 6	**ip admission name** *admission-name* **bypass regex** *regex-map* [**absolute-timer** *minutes*] **Example:** `Device(config)# ip admission name rule1 bypass regex regex-map1 absolute-timer 10`	Creates an IP Network Admission Control (NAC) rule to enable browser-based authentication bypass.

Step 7	Perform one of the following tasks:
	• **ip admission name** *admission-name* **ntlm**
	• **ip admission name** *admission-name* **http-basic**
	• **ip admission name** *admission-name* **proxy http**
	
	Example:
	`Device(config)# ip admission name rule1 ntlm`
	
	`Device(config)# ip admission name rule1 http-basic`
	
	`Device(config)# ip admission name rule1 proxy http`
		Configures one of the following authentication methods:
		• Windows NT LAN Manager (NTLM)
		• HTTP Basic
		• Web Authorization Proxy

Step 8	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet0/1/0`	Configures an interface and enters interface configuration mode.
Step 9	**ip admission** *admission-name* **Example:** `Device(config-if)# ip admission rule1`	Creates a Layer 3 Network Admission Control (NAC) rule to be applied to the interface.

	Command or Action	Purpose
Step 10	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

What to Do Next

For any parameter-map change to be reflected, remove and configure the **ip admission name** *admission-name* **bypass regex** *regex-map* [**absolute-timer** *minutes*] command in global configuration mode.

Verifying Browser-Based Authentication Bypass

SUMMARY STEPS

1. **enable**
2. **show ip admission cache**
3. **show ip admission configuration**

DETAILED STEPS

Step 1 **enable**
Enables privileged EXEC mode.

- Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 **show ip admission cache**
Displays the current list of network admission entries and verifies the browser authentication bypass.

Example:

```
Device# show ip admission cache

Client Name N/A, Client IP 172.31.108.123, Port 63142, timeout 60, Time Remaining 60, state ESTAB
(Browser Auth Bypass)
```

Step 3 **show ip admission configuration**
Displays the Network Admission Control (NAC) configuration.

Example:

```
Device# show ip admission configuration

Auth-proxy name webauth-profile
!
browser bypass, regex parameter-map name: reg-map inactivity-time 12 minutes absolute-timer 10 minutes
```

Configuration Examples for Browser-Based Authentication Bypass

Example: Configuring Browser-Based Authentication Bypass

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type regex regex-map1
Device(config-profile)# pattern Chrome
Device(config-profile)# exit
Device(config)# ip admission name rule1 bypass regex regex-map1 absolute-timer 10
Device(config)# ip admission name rule1 ntlm
Device(config)# interface gigabitethernet0/1/0
Device(config-if)# ip admission rule1
Device(config-if)# end
```

Additional References for Browser-Based Authentication Bypass

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Related Topic	Document Title
Security commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
Cisco Web Security	"Cisco Web Security" module in the *Security Configuration Guide: Zone-Based Policy Firewall*
Authenticating and authorizing connections	"Configuring Authentication Proxy" module in the *Authentication Proxy Configuration Guide*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

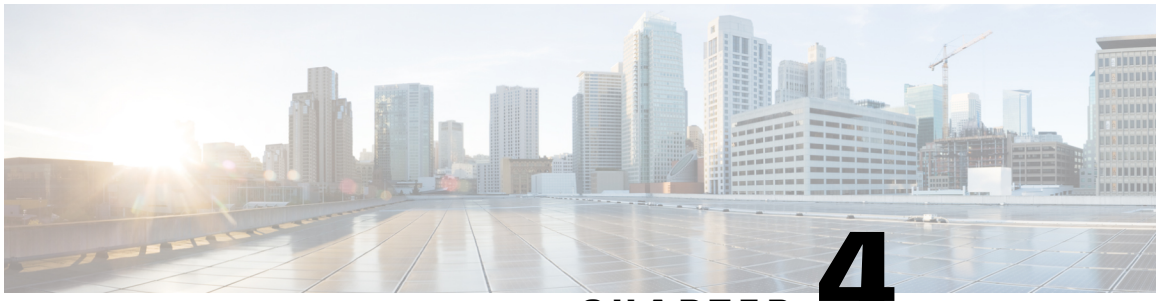
Feature Information for Browser-Based Authentication Bypass

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 3: Feature Information for Browser-Based Authentication Bypass

Feature Name	Releases	Feature Information
Browser-Based Authentication Bypass	15.3(3)M	The Browser-Based Authentication Bypass feature enables web browsers to bypass authentication methods such as HTTP Basic, Web Authorization Proxy, and Windows NTLM (passive or explicit). The following command was introduced: **ip admission name bypass regex**.



CHAPTER 4

LDAP Server State

The LDAP Server State feature enables users to capture information about Lightweight Directory Access Protocol (LDAP) server reachability before a request is sent to the server.

LDAP provides applications with a standard method for accessing and modifying the information stored in a directory. LDAP is integrated into the Cisco software as an authentication, authorization, and accounting (AAA) protocol alongside the existing AAA protocols such as RADIUS, TACACS+, Kerberos, and Diameter.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for LDAP Server State

The Lightweight Directory Access Protocol (LDAP) server should be marked as DEAD by default to get the exact state of the server and the server group.

Restrictions for LDAP Server State

When configuring a Lightweight Directory Access Protocol (LDAP) server, we assume that the server is in DEAD state and is not reachable. The correct state of the server is obtained after the deadtime (the period during which new authentication requests are not sent to an LDAP server that has failed to respond to a previous request) expiry is reached. Within this time frame, even if the server is reachable, no requests should be sent to the server.

Information About LDAP Server State

Overview of LDAP Server State

The LDAP Server State feature reduces the load on the network if the servers are not reachable and avoids unnecessary processing of retransmits.

The authentication, authorization, and accounting (AAA) servers are used to validate users or subscribers before they access a network. If one of the servers is not reachable, the next configured server specified in the configuration is contacted.

AAA client components make use of the DEAD and ALIVE states to keep track of each server state to handle protocol transactions effectively. If the state is DEAD, the client component applies a default set of policies to users or subscribers and allows them to access the default web content. If the state is ALIVE, the client component gets the actual policies from the Lightweight Directory Access Protocol (LDAP) server.

If the **automate-tester** command is configured along with the **deadtime** command, after every deadtime expiry, the AAA test APIs send a dummy bind request packet to the LDAP server.

- If a bind response is received, the server state is updated as ALIVE and further dummy bind requests are not sent.
- If a bind response is not received, the server state remains as DEAD and after every deadtime expiry, AAA test APIs send dummy bind request packets to the LDAP server.

If the **deadtime** command is configured when the server is not reachable, the server state remains DEAD until the deadtime expiry is reached, after which the state changes to ALIVE.

Note If one of the servers in a server group is ALIVE, the server group is marked as ALIVE.

How to Configure LDAP Server State

Perform this task to enable the server state notification functionality in a Lightweight Directory Access Protocol (LDAP) server. By default, all servers are marked as DEAD during configuration.

Configuring LDAP Server State

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username** *user* **password** {**0** | **7**} *password*
4. **aaa new-model**
5. **ldap server** *name*
6. **deadtime** *minutes*
7. **automate-tester username** *name* **probe-on**
8. **end**
9. **show ldap server**

DETAILED STEPS

	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	**username** *user* **password** {**0** | **7**} *password* **Example:** `Device(config)# username user1 password 0 pwd1`	Configures an unencrypted password that is automatically picked up by the **automate-tester** command.
Step 4	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables the authentication, authorization, and accounting (AAA) access control system.
Step 5	**ldap server** *name* **Example:** `Device(config)# ldap server server1`	Configures a device to use the LDAP protocol and enters LDAP server configuration mode.
Step 6	**deadtime** *minutes* **Example:** `Device(config-ldap-server)# deadtime 1`	Configures the deadtime expiry value (in minutes) for the LDAP server.

	Command or Action	Purpose
Step 7	**automate-tester username** *name* **probe-on** **Example:** `Device(config-ldap-server)# automate-tester username user1 probe-on`	Assigns the state of the LDAP server as DEAD by default when configured along with the **deadtime** *minutes* command.
Step 8	**end** **Example:** `Device(config-ldap-server)# end`	Exits LDAP server configuration mode and returns to privileged EXEC mode.
Step 9	**show ldap server** **Example:** `Device# show ldap server`	Displays the LDAP server state information and various other counters for the server.

Configuration Examples for LDAP Server State

Example: Configuring LDAP Server State

```
Device# configure terminal
Device(config)# username user1 password 0 pwd1
Device(config)# aaa new-model
Device(config)# ldap server server1
Device(config-ldap-server)# deadtime 1
Device(config-ldap-server)# automate-tester username user1 probe-on
Device(config-ldap-server)# end
```

The following output is displayed on entering the **automate-tester username** *name* **probe-on** command:

```
*Feb 24 09:14:55.139: LDAP_SERVER 192.0.2.10 Server state is UP
```

The following sample output from the **show ldap server** command shows the Lightweight Directory Access Protocol (LDAP) server state information of *server1* server and various other counters for the server.

```
Device# show ldap server server1 summary

Server Information for server1
================================
Server name :server1
Server Address :192.0.2.10
Server listening Port :389
Bind Root-dn :user1
Server mode :Non-Secure
Cipher Suite :0x00
Authentication Seq :Search first. Then Bind/Compare password next
Authentication Procedure:Bind with user password
Request timeout :30
Deadtime in Mins :1
State :ALIVE
No. of active connections :0
---------------------------------
```

Additional References for LDAP Server State

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Security Command Reference: Commands A to C • Security Command Reference: Commands D to L • Security Command Reference: Commands M to R • Security Command Reference: Commands S to Z
LDAP configuration tasks	"Configuring LDAP" chapter in *AAA LDAP Configuration Guide*

Standards and RFCs

Standard/RFC	Title
RFC 4511	*Lightweight Directory Access Protocol (LDAP)*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

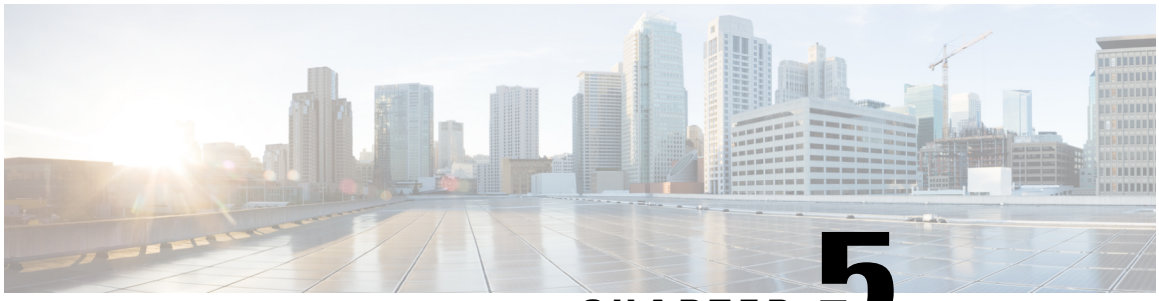
Feature Information for LDAP Server State

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 4: Feature Information for LDAP Server State

Feature Name	Releases	Feature Information
LDAP Server State	15.4(2)T	The LDAP Server State feature enables users to capture information about LDAP server reachability before a request is sent to the server. The following commands were introduced or modified: **automate-tester**, **deadtime**.



CHAPTER 5

Source Interface and VRF Support in LDAP

The Source Interface and VRF Support in LDAP feature allows you to configure a dedicated LDAP source interface IP address and virtual routing and forwarding (VRF) details on Cisco Integrated Services Routers (ISR) Generation 2. The source interface address (the address can be an IPv4 or IPv6 address) and VRF details are populated while creating a TCP connection between the Cisco ISR Generation 2 and the LDAP server. This module describes how to configure this feature.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Source Interface and VRF Support in LDAP

Source Interface and VRF Support in LDAP Overview

When Cisco Cloud Web Security and Cisco Integrated Services Routers (ISR) Generation 2 (G2) are deployed back-to-back, they require a Lightweight Directory Access Protocol (LDAP) request to traverse the VPN

tunnel between Cloud Web Security and the Cisco ISR G2. In such cases, the source interface IP address (example, the IP address of the LAN interface) must be specified in the LDAP query. Prior to the introduction of the Source Interface and VRF Support in LDAP feature, the source interface address cannot be specified in the source IP field of the LDAP query; instead the tunnel interface IP address was used in the source IP field.

The Source Interface and VRF Support in LDAP feature helps you configure a dedicated LDAP source interface address on Cisco ISR G2. The source interface address is configured on the Cisco ISR G2, and the device uses this interface address to originate all LDAP packets it sends to the LDAP server. The source interface address is also used for polling the end-server to ensure the reachability of the end-server.

The source interface IP (either an IPv4 or IPv6 address) address and virtual routing and forwarding (VRF) details are populated in the LDAP query while creating a TCP connection between the Cisco ISR G2 (client) and the LDAP server.

The VRF instance is configured on the Cisco ISR G2 and VRF table ID details are set in the socket option before creating a TCP connection to allow multiple instances of a routing table to coexist on the same device at the same time. Because routing instances are independent of each other, the same or overlapping IP address can be used without conflict.

Cloud Web Security with LDAP Source Interfaces

The following illustration shows a Cloud Web Security deployment that uses an Authentication, Authorization, and Accounting (AAA) configuration that supports source interface address and virtual routing and forwarding (VRF) details, while establishing a TCP connection between Cisco Integrated Services Routers (ISR) Generation 2 (G2) and Cloud Web Security.

The following section describes the packet flow that happens in the deployment scenario shown in the illustration:

1. A AAA process posts a bind or search request to the Lightweight Directory Access Protocol (LDAP) process.
2. The LDAP process processes the AAA request.
3. A TCP connection is established <<between what >>before sending the request to the LDAP server.

   While creating the TCP connection, the source IP address and the VRF table details are set in the LDAP socket context.
4. - If the {**ip** | **ipv6**} **ldap source-interface** command is configured under the **aaa group server ldap** command, the source IP address and VRF details are populated before the TCP connection is established.
   - If the {**ip** | **ipv6**} **ldap source-interface** command is configured in global configuration mode; globally for the box, the source IP address and VRF details are populated after the TCP connection is established.
   - If the {**ip** | **ipv6**} **ldap source-interface** command is not configured, the best local IP address and the default table ID details are populated in the TCP packet while establishing the connection.
   - If you have configured the source interface address both under the **aaa group server ldap** command and in global configuration mode, the configuration under the **aaa group server ldap** command has the highest priority.
5. The LDAP process uses the TCP connection to send or receive packets.

6 If the source interface address or VRF configurations are changed or removed, the LDAP process tears down all existing TCP connections and establishes a new TCP connection with a new source interface address or the best local IP address when sending an LDAP packet.

How to Configure Source Interface and VRF Support in LDAP

Configuring LDAP Source Interface and VRF

If you have configured the source interface address and virtual routing and forwarding (VRF) instance under the **aaa group server ldap** command and in global configuration mode, the configuration under the **aaa group server ldap** command has the highest priority.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa group server ldap** *group-name*
5. {**ip** | **ipv6**} **ldap source-interface** *interface-type interface-number*
6. {**ip** | **ipv6**} **vrf forwarding** *vrf-name*
7. **server** *name*
8. **exit**
9. {**ip** | **ipv6**} **ldap source-interface** *interface-type interface-number* [**vrf** *vrf-name*]
10. **end**

DETAILED STEPS

	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(config)# configure terminal`	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables the authentication, authorization, and accounting (AAA) access control model.

	Command or Action	Purpose
Step 4	**aaa group server ldap** *group-name* **Example:** `Device(config)# aaa group server ldap ldap-server-group`	Groups different Lightweight Directory Access Protocol (LDAP) servers into distinct lists and methods and enters LDAP server-group configuration mode.
Step 5	{**ip** | **ipv6**} **ldap source-interface** *interface-type interface-number* **Example:** `Device(config-ldap-sg)# ip ldap source-interface gigabitethernet 0/0/0`	Specifies the source interface IP address in the LDAP packets.
Step 6	{**ip** | **ipv6**} **vrf forwarding** *vrf-name* **Example:** `Device(config-ldap-sg)# ip vrf forwarding cws-vrf`	Configures a virtual routing and forwarding (VRF) reference of an authentication, authorization, and accounting (AAA) LDAP server group.
Step 7	**server** *name* **Example:** `Device(config-ldap-sg)# server ldap-server`	Specifies the LDAP server.
Step 8	**exit** **Example:** `Device(config-ldap-sg)# exit`	Exits LDAP server-group configuration mode and returns to global configuration mode.
Step 9	{**ip** | **ipv6**} **ldap source-interface** *interface-type interface-number* [**vrf** *vrf-name*] **Example:** `Device(config)# ip ldap source-interface gigabitethernet 0/1/0 vrf cws-vrf-1`	Specifies the source interface IP address in the LDAP packets.
Step 10	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Configuration Examples for Source Interface and VRF Support in LDAP

Example: Configuring LDAP Source Interface and VRF

```
Device(config)# configure terminal
Device(config)# aaa new-model
```

```
Device(config)# aaa group server ldap ldap-server-group
Device(config-ldap-sg)# ip ldap source-interface gigabitethernet 0/0/0
Device(config-ldap-sg)# ip vrf forwarding cws-vrf
Device(config-ldap-sg)# server ldap-server
Device(config-ldap-sg)# exit
Device(config)# ip ldap source-interface gigabitethernet 0/1/0 vrf cws-vrf-1
Device(config)# end
```

Additional References for Source Interface and VRF Support in LDAP

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Security Command Reference: Commands A to C • Security Command Reference: Commands D to L • Security Command Reference: Commands M to R • Security Command Reference: Commands S to Z
LDAP configuration tasks	"Configuring LDAP" chapter in *AAA LDAP Configuration Guide*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

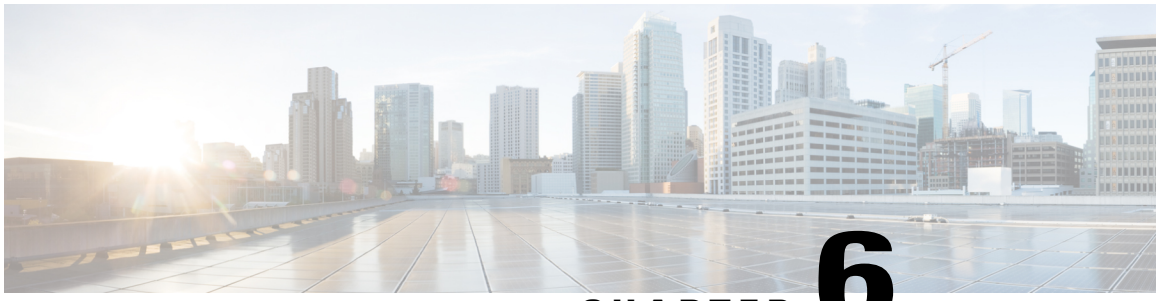
Feature Information for Source Interface and VRF Support in LDAP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 5: Feature Information for Source Interface and VRF Support in LDAP

Feature Name	Releases	Feature Information
Source Interface and VRF Support in LDAP	15.2(3)E 15.4(3)M	The Source Interface and VRF Support feature allows you to configure a dedicated LDAP source interface on Cisco Integrated Services Routers (ISR) Generation 2. The source interface, which can be an IPv4 or IPv6 interface, and virtual routing and forwarding (VRF) details are populated while creating a TCP connection between the Cisco ISR Generation 2 and the LDAP server. This feature was integrated into the Cisco IOS Release 15.2(3)E. The following command was introduced or modified: **aaa group server ldap**, **ip ldap source-interface**, **ldap source-interface**, and **server (LDAP)**.



CHAPTER 6

Whitelist Download from Tower for Proxy Cloud Web Security

The Whitelist Download from Tower for Proxy Cloud Web Security feature supports the download of whitelists from the Cloud Web Security tower.

This module provides more information about the feature and explains how to configure it.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Whitelist Download from Tower for Proxy Cloud Web Security

Install the Trusted Core Trust-Store certificate that can be obtained from http://www.cisco.com/security/pki/trs/ios_core.p7b. One of the root certificates under the trust store is chained with the certificate used by Cloud Web Security Tower. This root certificate will validate the certificate from the Cloud Web Security tower and establish a Secure HTTP connection to fetch the exception lists.xml file.

To install the trust store directly on your Cisco router, configure the following command:

```
Device(config)# crypto pki trustpool import url
http://www.cisco.com/security/pki/trs/ios_core.p7b
```

Restrictions for Whitelist Download from Tower for Proxy Cloud Web Security

- The Cloud Web Security tower does not support IPv6 addresses. Only IPv4 addresses with wild card masks are supported.

Information About Whitelist Download from Tower for Proxy Mode in Cloud Web Security

Whitelist Download from Tower Overview

In Cisco IOS Release 15.5(1)T and later releases, Cloud Web Security supports the download of whitelists from the Cloud Web Security tower. You can download host, user-agent, and IP-based whitelist from the tower. Prior to the introduction of this feature, network administrators had to configure a whitelist through the CLI on all devices in the network. When whitelists are downloaded from the tower, it helps maintain the same configuration across all devices in the network.

Use the **whitelist dowload enable** command to download whitelists from the tower at regular intervals.

The following section explains how the whitelist information is downloaded from the Cloud Web Security tower:

- A router (for example, an Integrated Services Router [ISR] Generation 2) initiates a request for whitelist patterns from the Cloud Web Security tower.
- Whitelist patterns are sent in the form of an XML file.
- The communication between the router and the Cloud Web Security tower happens over secure HTTP (HTTPS). It is mandatory to have Certificate Authority (CA) certificate on the router without which the whitelist download will not work.

The Cloud Web Security tower validates the request from a device by using the x-Scansafe header fields. All header fields details are in encrypted form.

How Whitelist Download Works

Whitelisting bypasses the HTTP request-header matching traffic to a web server instead of the Cloud Web Security tower or server.

Header-based whitelisting includes domain-based whitelisting and user agent-based whitelisting. Domain-based whitelisting includes domain names and regex patterns. Whitelisting can either be configured through the CLI or as patterns that are downloaded from the Cloud Web Security tower in XML format.

When a device requests for the whitelist configuration, the Cloud Web Security tower sends the whitelist configuration file in XML format. This XML file is parsed to retrieve the encoding type and the list of whitelisted domain names, user-agent patterns, and IPv4 addresses. These parsed patterns are added to respective regex tree for whitelisting.

Whitelist patterns from the Cloud Web Security tower are not stored in the configuration. Whitelist patterns configured through the CLI are stored in the configuration. Whitelist patterns configured via the CLI and patterns downloaded from the tower can be used for whitelisting. To view the list of downloaded whitelist patterns, use the **show cws tower-whitelist** command.

When an XML file is received and parsed successfully, all previous domain names are removed and newly received domain names are saved. Locally configured domain names are not affected; only domain names from the tower are removed. If patterns added to the regex file fails, all successfully added patterns are retained for whitelisting.

The XML file consists of a list of domain names or patterns and the full IPv4 address of each domain. The maximum length of a domain should be 256 characters or less. Wild card characters supported for domain patterns are ., *, ^, +, ?, $, [], and [^]. The first character of a pattern cannot be + or *.

In IP-based whitelisting, the Cloud Web Security tower does not verify whether duplicate entries exist in access control lists (ACLs) configured through the CLI. Traffic matching any ACL entry configured through the CLI or downloaded from the tower is bypassed from Cloud Web Security tower redirection.

If header-based or IP-based whitelisting is enabled via the CLI and also downloaded from the tower, both whitelist configurations are applied to incoming packets. If the header-based or IP-based whitelisting is disabled via the CLI, only the whitelist configuration downloaded from the Cloud Web Security tower is applied to incoming packets.

Request and Response Handling for Whitelist Download

Request and response handling is supported for header-based and IP-based whitelisting.

The device on which Cloud Web Security is configured, uses secure HTTP (HTTPS), to request the exception list or the list of whitelisted traffic from the Cloud Web Security tower. The timestamp field in the HTTP header is used to check for updates or changes to the whitelist configuration.

If the whitelist configuration is not modified after the last whitelist download, the Cloud Web Security tower responds by indicating that there are no changes to the configuration.

If the whitelist configuration is modified after the last whitelist download, the Cloud Web Security tower sends the updated whitelist configuration file in XML format with the updated timestamp.

When you configure the **whitelist download enable** command without any time interval, devices send an HTTPS request to the Cloud Web Security tower every 60 minutes. To reflect changes to the whitelist configuration, you must reconfigure the timer to with a different value download the latest whitelist configuration file to the device.

How to Configure Whitelist Download from Tower for Proxy Cloud Web Security

Enabling Whitelist File Download

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type cws global**
4. **whitelist download enable** [**interval** *minutes*]
5. **end**
6. **show cws tower-whitelist** [**stats**]

DETAILED STEPS

	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	**parameter-map type cws global** **Example:** `Device(config)# parameter-map type cws global`	Configures a global Cloud Web Security parameter map and enters parameter-map type inspect configuration mode.
Step 4	**whitelist download enable** [**interval** *minutes*] **Example:** `Device(config-profile)# whitelist download enable interval 20`	Enables the download of Cloud Web Security whitelist configuration file. • The default download interval is 60 minutes.
Step 5	**end** **Example:** `Device(config-profile)# end`	Exits parameter-map type inspect configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show cws tower-whitelist** [**stats**] **Example:** `Device(config-profile)# show cws tower-whitelist`	Displays a list of whitelist patterns downloaded from the Cloud Web Security tower.

The following is sample output from the **show cws tower-whitelist** command:

```
Device# show cws tower-whitelist

Last modified time at tower : Wed, 06 Nov 2014 05:47:52 UTC
Domain names:
  .*redhat.*
  .*xerox.*
  .*yahoo.*.
Extended IP access list cws-internal-dnld-wl-acl
    10 permit ip 10.10.1.16 0.0.0.15 any
    20 permit ip any host 202.3.77.184
User-agent patterns:
  mozilla
  Safari
```

The following sample output from the **show cws tower-whitelist stats** command displays information about whitelist download:

```
Device# show cws tower-whitelist stats

Total Connect Request:                                    13
Total Connect Response:                                   13
Total WL download request:                                13
SSL failures:                                             0
WL download response:                                     13
     Total success response:                              1
     Total no config change:                              7
     Total no config:                                     0
     Total other responses(Other than 200/304/404):       5
     Total other failures(no encoding/HTTP version):      0
XML parse errors:                                         0
Memory failures:                                          0

XML parser stats:
  Src ACLs    Dst ACLs   Domain-name     User-agent
    1           1          1               2
```

Configuration Example for Whitelist Download from Tower for Proxy Cloud Web Security

Example: Enabling Whitelist File Download

```
Device# configure terminal
Device(config)# parameter-map type cws global
Device(config-profile)# whitelist download enable interval 20
Device(config-profile)# end
```

Additional References Whitelist Download from Tower for Proxy Cloud Web Security

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Security Command Reference: Commands A to C • Security Command Reference: Commands D to L • Security Command Reference: Commands M to R • Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Whitelist Download from Tower for Proxy Cloud Web Security

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for Whitelist Download from Tower for Proxy Cloud Web Security

Feature Name	Releases	Feature Information
Whitelist Download from Tower for Proxy Cloud Web Security	15.5(1)T	The Whitelist Download from Tower for Proxy Cloud Web Security feature supports the download of whitelists to devices that have Cloud Web Security configured. The following command was introduced or modified: **whitelist download enable** and **show cws tower-whitelist**.