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.