CHAPTER 25

Configuring the ASA for Cisco Cloud Web Security

Cisco Cloud Web Security provides web security and web filtering services through the Software-as-a-Service (SaaS) model. Enterprises with the ASA in their network can use Cloud Web Security services without having to install additional hardware.

When Cloud Web Security is enabled on the ASA, the ASA transparently redirects selected HTTP and HTTPS traffic to the Cloud Web Security proxy servers. The Cloud Web Security proxy servers then scan the content and allow, block, or send a warning about the traffic based on the policy configured in Cisco ScanCenter to enforce acceptable use and to protect users from malware.

The ASA can optionally authenticate and identify users with Identity Firewall (IDFW) and AAA rules. The ASA encrypts and includes the user credentials (including usernames and/or user groups) in the traffic it redirects to Cloud Web Security. The Cloud Web Security service then uses the user credentials to match the traffic to the policy. It also uses these credentials for user-based reporting. Without user authentication, the ASA can supply an (optional) default username and/or group, although usernames and groups are not required for the Cloud Web Security service to apply policy.

You can customize the traffic you want to send to Cloud Web Security when you create your service policy rules. You can also configure a "whitelist" so that a subset of web traffic that matches the service policy rule instead goes directly to the originally requested web server and is not scanned by Cloud Web Security.

You can configure a primary and a backup Cloud Web Security proxy server, each of which the ASA polls regularly to check for availability.

Note This feature is also called "ScanSafe," so the ScanSafe name appears in some commands.

This chapter includes the following sections:

- Information About Cisco Cloud Web Security, page 25-2
- Licensing Requirements for Cisco Cloud Web Security, page 25-6
- Prerequisites for Cloud Web Security, page 25-7
- Guidelines and Limitations, page 25-7
- Default Settings, page 25-8
- Configuring Cisco Cloud Web Security, page 25-8
- Monitoring Cloud Web Security, page 25-17
- Configuration Examples for Cisco Cloud Web Security, page 25-18
- Related Documents, page 25-26
- Feature History for Cisco Cloud Web Security, page 25-26

Information About Cisco Cloud Web Security

This section includes the following topics:

- Redirection of Web Traffic to Cloud Web Security, page 25-2
- User Authentication and Cloud Web Security, page 25-2
- Authentication Keys, page 25-3
- ScanCenter Policy, page 25-4
- Cloud Web Security Actions, page 25-5
- Bypassing Scanning with Whitelists, page 25-6
- IPv4 and IPv6 Support, page 25-6
- Failover from Primary to Backup Proxy Server, page 25-6

Redirection of Web Traffic to Cloud Web Security

When an end user sends an HTTP or HTTPS request, the ASA receives it and optionally retrieves the user and/or group information. If the traffic matches an ASA service policy rule for Cloud Web Security, then the ASA redirects the request to the Cloud Web Security proxy servers. The ASA acts as an intermediary between the end user and the Cloud Web Security proxy server by redirecting the connection to the proxy server. The ASA changes the destination IP address and port in the client requests and adds Cloud Web Security-specific HTTP headers and then sends the modified request to the Cloud Web Security proxy server. The Cloud Web Security HTTP headers include various kinds of information, including the username and user group (if available).

User Authentication and Cloud Web Security

User identity can be used to apply policy in Cloud Web Security. User identity is also useful for Cloud Web Security reporting. User identity is not required to use Cloud Web Security. There are other methods to identify traffic for Cloud Web Security policy.

The ASA supports the following methods of determining the identity of a user, or of providing a default identity:

- AAA rules—When the ASA performs user authentication using a AAA rule, the username is retrieved from the AAA server or local database. Identity from AAA rules does not include group information. If configured, the default group is used. For information about configuring AAA rules, see Chapter 7, "Configuring AAA Rules for Network Access."
- IDFW—When the ASA uses IDFW with the Active Directory (AD), the username and group is retrieved from the AD agent when you activate a user and/or group by using an ACL in a feature such as an access rule or in your service policy, or by configuring the user identity monitor to download user identity information directly.

  For information about configuring IDFW, see Chapter 38, "Configuring the Identity Firewall," in the general operations configuration guide.
- Default username and group—Without user authentication, the ASA uses an optional default username and/or group for all users that match a service policy rule for Cloud Web Security.

Authentication Keys

Each ASA must use an authentication key that you obtain from Cloud Web Security. The authentication key lets Cloud Web Security identify the company associated with web requests and ensures that the ASA is associated with valid customer.

You can use one of two types of authentication keys for your ASA: the company key or the group key.

- Company Authentication Key, page 25-3
- Group Authentication Key, page 25-3

Company Authentication Key

A Company authentication key can be used on multiple ASAs within the same company. This key simply enables the Cloud Web Security service for your ASAs. The administrator generates this key in ScanCenter (https://scancenter.scansafe.com/portal/admin/login.jsp); you have the opportunity to e-mail the key for later use. You cannot look up this key later in ScanCenter; only the last 4 digits are shown in ScanCenter. For more information, see the Cloud Web Security documentation: http://www.cisco.com/en/US/products/ps11720/products_installation_and_configuration_guides_list.html.

Group Authentication Key

A Group authentication key is a special key unique to each ASA that performs two functions:

- Enables the Cloud Web Security service for one ASA.
- Identifies all traffic from the ASA so you can create ScanCenter policy per ASA.

For information about using the Group authentication key for policy, see the "ScanCenter Policy" section on page 25-4).

The administrator generates this key in ScanCenter (https://scancenter.scansafe.com/portal/admin/login.jsp); you have the opportunity to e-mail the key for later use. You cannot look up this key later in ScanCenter; only the last 4 digits are shown in ScanCenter.

For more information, see the Cloud Web Security documentation: http://www.cisco.com/en/US/products/ps11720/products_installation_and_configuration_guides_list.html.

ScanCenter Policy

In ScanCenter, traffic is matched against policy rules in order until a rule is matched. Cloud Web Security then applies the configured action for the rule. User traffic can match a policy rule in ScanCenter based on group association: a *directory group* or a *custom group*.

- Directory Groups, page 25-4
- Custom Groups, page 25-4
- How Groups and the Authentication Key Interoperate, page 25-5

Directory Groups

Directory groups define the group to which traffic belongs. The group, if present, is included in the HTTP header of the client request. The ASA includes the group in the HTTP header when you configure IDFW. If you do not use IDFW, you can configure a default group for traffic matching an ASA rule for Cloud Web Security inspection.

When you configure a directory group, you must enter the group name exactly.

- IDFW group names are sent in the following format:

  *domain-name**group-name*

  When the ASA learns the IDFW group name, the format on the ASA is *domain-name**group-name*. However, the ASA modifies the name to use only one backslash (\) to conform to typical ScanCenter notation.

- The default group name is sent in the following format:

  [*domain*\]*group-name*

  On the ASA, you need to configure the optional domain name to be followed by 2 backslashes (\\); however, the ASA modifies the name to use only one backslash (\) to conform to typical ScanCenter notation. For example, if you specify “Cisco\\Boulder1,” the ASA modifies the group name to be “Cisco\Boulder1” with only one backslash (\) when sending the group name to Cloud Web Security.

Custom Groups

Custom groups are defined using one or more of the following criteria:

- ScanCenter Group authentication key—You can generate a Group authentication key for a custom group. Then, if you identify this group key when you configure the ASA, all traffic from the ASA is tagged with the Group key.
- Source IP address—You can identify source IP addresses in the custom group. Note that the ASA service policy is based on source IP address, so you might want to configure any IP address-based policy on the ASA instead.
- Username—You can identify usernames in the custom group.
  - IDFW usernames are sent in the following format:

    *domain-name**username*

- AAA usernames, when using RADIUS or TACACS+, are sent in the following format:

  LOCAL*username*
- AAA usernames, when using LDAP, are sent in the following format:

  *domain-name**username*
- For the default username, it is sent in the following format:

  [*domain-name*\]*username*

  For example, if you configure the default username to be "Guest," then the ASA sends "Guest." If you configure the default username to be "Cisco\Guest," then the ASA sends "Cisco\Guest."

How Groups and the Authentication Key Interoperate

Unless you need the per-ASA policy that a custom group+group key provides, you will likely use a company key. Note that not all custom groups are associated with a group key. Non-keyed custom groups can be used to identify IP addresses or usernames, and can be used in your policy along with rules that use directory groups.

Even if you do want per-ASA policy and are using a group key, you can also use the matching capability provided by directory groups and non-keyed custom groups. In this case, you might want an ASA-based policy, with some exceptions based on group membership, IP address, or username. For example, if you want to exempt users in the America\Management group across all ASAs:

1. Add a directory group for America\Management.
2. Add an exempt rule for this group.
3. Add rules for each custom group+group key after the exempt rule to apply policy per-ASA.
4. Traffic from users in America\Management will match the exempt rule, while all other traffic will match the rule for the ASA from which it originated.

Many combinations of keys, groups, and policy rules are possible.

Cloud Web Security Actions

After applying the configured policies, Cloud Web Security either blocks, allows, or sends a warning about the user request:

- Allows—When Cloud Web Security allows the client request, it contacts the originally requested server and retrieves the data. It forwards the server response to the ASA, which then forwards it to the user.
- Blocks—When Cloud Web Security blocks the client request, it notifies the user that access has been blocked. It sends an HTTP 302 "Moved Temporarily" response that redirects the client application to a web page hosted by the Cloud Web Security proxy server showing the blocked error message. The ASA forwards the 302 response to the client.
- Warns—When the Cloud Web Security proxy server determines that a site may be in breach of the acceptable use policy, it displays a warning page about the site. You can choose to heed the warning and drop the request to connect, or you can click through the warning and proceed to the requested site.

You can also choose how the ASA handles web traffic when it cannot reach either the primary or backup Cloud Web Security proxy server. It can block or allow all web traffic. By default, it blocks web traffic.

Bypassing Scanning with Whitelists

If you use AAA rules or IDFW, you can configure the ASA so that web traffic from specific users or groups that otherwise match the service policy rule is not redirected to the Cloud Web Security proxy server for scanning. When you bypass Cloud Web Security scanning, the ASA retrieves the content directly from the originally requested web server without contacting the proxy server. When it receives the response from the web server, it sends the data to the client. This process is called "whitelisting" traffic.

Although you can achieve the same results of exempting traffic based on user or group when you configure the class of traffic using ACLs to send to Cloud Web Security, you might find it more straightforward to use a whitelist instead. Note that the whitelist feature is only based on user and group, not on IP address.

IPv4 and IPv6 Support

Cloud Web Security currently supports only IPv4 addresses. If you use IPv6 internally, NAT 64 must be performed for any IPv6 flows that need to be sent to Cloud Web Security.

The following table shows the class map traffic that is supported by Cloud Web Security redirection:

Class Map Traffic	Cloud Web Security Inspection
From IPv4 to IPv4	Supported
From IPv6 to IPv4 (using NAT64)	Supported
From IPv4 to IPv6	Not Supported
From IPv6 to IPv6	Not Supported

Failover from Primary to Backup Proxy Server

When you subscribe to the Cisco Cloud Web Security service, you are assigned a primary Cloud Web Security proxy server and backup proxy server.

If any client is unable to reach the primary server, then the ASA starts polling the tower to determine availability. (If there is no client activity, the ASA polls every 15 miniutes.) If the proxy server is unavailable after a configured number of retries (the default is 5; this setting is configurable), the server is declared unreachable, and the backup proxy server becomes active.

If a client or the ASA can reach the server at least twice consecutively before the retry count is reached, the polling stops and the tower is determined to be reachable.

After a failover to the backup server, the ASA continues to poll the primary server. If the primary server becomes reachable, then the ASA returns to using the primary server.

Licensing Requirements for Cisco Cloud Web Security

Model	License Requirement
All models	Strong Encryption (3DES/AES) License to encrypt traffic between the security appliance and the Cloud Web Security server.

On the Cloud Web Security side, you must purchase a Cisco Cloud Web Security license and identify the number of users that the ASA handles. Then log into ScanCenter, and generate your authentication keys.

Prerequisites for Cloud Web Security

(Optional) User Authentication Prerequisites

To send user identity information to Cloud Web Security, configure one of the following on the ASA:

- AAA rules (username only)—See Chapter 7, "Configuring AAA Rules for Network Access."
- IDFW (username and group)—See Chapter 38, "Configuring the Identity Firewall," in the general operations configuration guide.

(Optional) Fully Qualified Domain Name Prerequisites

If you use FQDNs in ACLs for your service policy rule, or for the Cloud Web Security server, you must configure a DNS server for the ASA according to the "Configuring the DNS Server" section on page 16-8 in the general operations configuration guide.

Guidelines and Limitations

Context Mode Guidelines

Supported in single and multiple context modes.

In multiple context mode, the server configuration is allowed only in the system, and the service policy rule configuration is allowed only in the security contexts.

Each context can have its own authentication key, if desired.

Firewall Mode Guidelines

Supported in routed firewall mode only. Does not support transparent firewall mode.

IPv6 Guidelines

Does not support IPv6. See the "IPv4 and IPv6 Support" section on page 25-6.

Additional Guidelines

- Cloud Web Security is not supported with ASA clustering.
- Clientless SSL VPN is not supported with Cloud Web Security; be sure to exempt any clientless SSL VPN traffic from the ASA service policy for Cloud Web Security.

- When an interface to the Cloud Web Security proxy servers goes down, output from the **show scansafe server** command shows both servers up for approximately 15-25 minutes. This condition may occur because the polling mechanism is based on the active connection, and because that interface is down, it shows zero connection, and it takes the longest poll time approach.
- Cloud Web Security is not supported with the ASA CX module. If you configure both the ASA CX action and Cloud Web Security inspection for the same traffic, the ASA only performs the ASA CX action.
- Cloud Web Security inspection is compatibile with HTTP inspection for the same traffic. HTTP inspection is enabled by default as part of the default global policy.
- Cloud Web Security is not supported with extended PAT or any application that can potentially use the same source port and IP address for separate connections. For example, if two different connections (targeted to separate servers) use extended PAT, the ASA might reuse the same source IP and source port for both connection translations because they are differentiated by the separate destinations. When the ASA redirects these connections to the Cloud Web Security server, it replaces the destination with the Cloud Web Security server IP address and port (8080 by default). As a result, both connections now appear to belong to the same flow (same source IP/port and destination IP/port), and return traffic cannot be untranslated properly.
- The **match default-inspection-traffic** command does not include the default ports for the Cloud Web Security inspection (80 and 443).

Default Settings

By default, Cisco Cloud Web Security is not enabled.

Configuring Cisco Cloud Web Security

- Configuring Communication with the Cloud Web Security Proxy Server, page 25-8
- (Multiple Context Mode) Allowing Cloud Web Security Per Security Context, page 25-9
- Configuring a Service Policy to Send Traffic to Cloud Web Security, page 25-10
- (Optional) Configuring Whitelisted Traffic, page 25-15
- Configuring the Cloud Web Security Policy, page 25-16

Configuring Communication with the Cloud Web Security Proxy Server

Guidelines

The public key is embedded in the ASA software, so there is no need for you to configure it.

Detailed Steps

	Command	Purpose
Step 1	**scansafe general-options** **Example:** `ciscoasa(config)# scansafe general-options`	Enters scansafe general-options configuration mode.
Step 2	**server primary** {**ip** *ip_address* | **fqdn** *fqdn*} [**port** *port*] **Example:** `ciscoasa(cfg-scansafe)# server primary ip 192.168.43.10`	Configures the fully qualified domain name or IP address of the primary Cloud Web Security proxy server. By default, the Cloud Web Security proxy server uses port 8080 for both HTTP and HTTPS traffic; do not change this value unless directed to do so.
Step 3	**server backup** {**ip** *ip_address* | **fqdn** *fqdn*} [**port** *port*] **Example:** `ciscoasa(cfg-scansafe)# server backup fqdn server.example.com`	(Optional) Configures the fully qualified domain name or IP address of the backup Cloud Web Security proxy server. By default, the Cloud Web Security proxy server uses port 8080 for both HTTP and HTTPS traffic; do not change this value unless directed to do so.
Step 4	**retry-count** *value* **Example:** `ciscoasa(cfg-scansafe)# retry-count 2`	(Optional) Enters the value for the number of consecutive polling failures to the Cloud Web Security proxy server before determining the server is unreachable. Polls are performed every 30 seconds. Valid values are from 2 to 100, and the default is 5. See the "Failover from Primary to Backup Proxy Server" section on page 25-6.
Step 5	**license** *hex_key* **Example:** `ciscoasa(cfg-scansafe)# license F12A588FE5A0A4AE86C10D222FC658F3`	Configures the authentication key that the ASA sends to the Cloud Web Security proxy servers to indicate from which organization the request comes. The authentication key is a 16-byte hexidecimal number. See the "Authentication Keys" section on page 25-3.

Examples

The following example configures a primary and backup server:

```
scansafe general-options
 server primary ip 10.24.0.62 port 8080
 server backup ip 10.10.0.7 port 8080
 retry-count 7
 license 366C1D3F5CE67D33D3E9ACEC265261E5
```

(Multiple Context Mode) Allowing Cloud Web Security Per Security Context

In multiple context mode, you must allow Cloud Web Security per context. See the "Configuring a Security Context" section on page 8-20 in the general operations configuration guide.

Note You must configure a route pointing to the Scansafe towers in both; the admin context and the specific context. This ensures that the Scansafe tower does not become unreachable in the Active/Active failover scenario.

The following sample configuration enables Cloud Web Security in context one with the default license and in context two with the license key override:

```
! System Context
!
scansafe general-options
server primary ip 180.24.0.62 port 8080
retry-count 5
license 366C1D3F5CE67D33D3E9ACEC265261E5
!
context one
 allocate-interface GigabitEthernet0/0.1
 allocate-interface GigabitEthernet0/1.1
 allocate-interface GigabitEthernet0/3.1
 scansafe
 config-url disk0:/one_ctx.cfg
!
context two
 allocate-interface GigabitEthernet0/0.2
 allocate-interface GigabitEthernet0/1.2
 allocate-interface GigabitEthernet0/3.2
 scansafe license 366C1D3F5CE67D33D3E9ACEC26789534
 config-url disk0:/two_ctx.cfg
!
```

Configuring a Service Policy to Send Traffic to Cloud Web Security

See Chapter 1, "Configuring a Service Policy Using the Modular Policy Framework," for more information about service policy rules.

Prerequisites

(Optional) If you need to use a whitelist to exempt some traffic from being sent to Cloud Web Security, first create the whitelist according to the "(Optional) Configuring Whitelisted Traffic" section on page 25-15 so you can refer to the whitelist in your service policy rule.

Detailed Steps

	Command	Purpose
Step 1	**policy-map type inspect scansafe** *name1* **Example:** `ciscoasa(config)# policy-map type inspect scansafe cws_inspect_pmap1`	Creates an inspection policy map so you can configure essential parameters for the rule and also optionally identify the whitelist. An inspection policy map is required for each class of traffic that you want to send to Cloud Web Security. The *policy_map_name* argument can be up to 40 characters in length. You enter policy-map configuration mode.
Step 2	`parameters` **Example:** `ciscoasa(config-pmap)# parameters`	Parameters lets you configure the protocol and the default user or group. You enter parameters configuration mode.
Step 3	**{http | https}** **Example:** `ciscoasa(config-pmap-p)# http`	You can only specify one service type for this inspection policy map, either **http** or **https**.
Step 4	(Optional) **default** {**[user** *username*] [**group** *groupname*]} **Example:** `ciscoasa(config-pmap-p)# default group default_group`	Specifies that if the ASA cannot determine the identity of the user coming into the ASA, then the default user and/or group is included in the HTTP header.
Step 5	(Optional, for a Whitelist) **class** *whitelist_name* **Example:** `ciscoasa(config-pmap-p)# class whitelist1`	Identifies the whitelist class map name that you created in the "(Optional) Configuring Whitelisted Traffic" section on page 25-15.
Step 6	`whitelist` **Example:** `ciscoasa(config-pmap-p)# class whitelist1` `ciscoasa(config-pmap-c)# whitelist`	Performs the whitelist action on the class of traffic.

	Command	Purpose
Step 7	**policy-map type inspect scansafe** *name2* parameters **default** {**[user** *user*] [**group** *group*]} **class** *whitelist_name2* whitelist **Example:** `ciscoasa(config)# policy-map type inspect scansafe cws_inspect_pmap2` `ciscoasa(config-pmap)# parameters` `ciscoasa(config-pmap-p)# default group2 default_group2` `ciscoasa(config-pmap-p)# class whitelist2` `ciscoasa(config-pmap-c)# whitelist`	Repeat Step 1 to Step 6 to create a separate class map for HTTPS traffic (for example). You can create an inspection class map for each class of traffic you want to send to Cloud Web Security. You can reuse an inspection class map for multiple classes of traffic if desired.
Step 8	**access-list** *access_list_name* [**line** *line_number*] **extended** {**deny** | **permit**} **tcp** [*user_argument*] [*security_group_argument*] *source_address_argument* [*port_argument*] *dest_address_argument* [*port_argument*] **Example:** `ciscoasa(config)# object network cisco1` `ciscoasa(config-object-network)# fqdn www.cisco.com` `ciscoasa(config)# object network cisco2` `ciscoasa(config-object-network)# fqdn tools.cisco.com` `ciscoasa(config)# access-list SCANSAFE_HTTP extended deny tcp any4 object cisco1 eq 80` `ciscoasa(config)# access-list SCANSAFE_HTTP extended deny tcp any4 object cisco2 eq 80` `ciscoasa(config)# access-list SCANSAFE_HTTP extended permit tcp any4 any4 eq 80`	Identifies the class of traffic you want to send to Cloud Web Security. Create an ACL consisting of one or more access control entries (ACEs). For detailed information about ACLs, see Chapter 19, "Adding an Extended Access Control List," in the general operations configuration guide. Cloud Web Security only operates on HTTP and HTTPS traffic. Each type of traffic is treated separately by the ASA. Therefore, you need to create HTTP-only ACLs and HTTPS-only ACLs. Create as many ACLs as needed for your policy. A **permit** ACE sends matching traffic to Cloud Web Security. A **deny** ACE exempts traffic from the service policy rule, so it is not sent to Cloud Web Security. When creating your ACLs, consider how you can match appropriate traffic that is destined for the Internet, but not match traffic that is destined for other internal networks. For example, to prevent inside traffic from being sent to Cloud Web Security when the destination is an internal server on the DMZ, be sure to add a deny ACE to the ACL that exempts traffic to the DMZ. FQDN network objects might be useful in exempting traffic to specific servers. The *user_argument* lets you specify the IDFW username or group, either inline or by referring to an object group. The *security_group_argument* lets you specify the TrustSec security group, either inline or by referring to an object group. Note that although you can match traffic to send to Cloud Web Security by security group, the ASA does not send security group information to Cloud Web Security in the HTTP header; Cloud Web Security cannot create policy based on the security group.
Step 9	**class-map** *name1* **Example:** `ciscoasa(config)# class-map cws_class1`	Creates a class map to identify the traffic for which you want to enable Cloud Web Security filtering.

	Command	Purpose
Step 10	**match access-list** *acl1* **Example:** `ciscoasa(config-cmap)# match access-list SCANSAFE_HTTP`	Specifies an ACL created in Step 8. Although you can use other match statements for this rule, we recommend using the **match access-list** command because it is the most versatile for identifying HTTP or HTTPS-only traffic. See the "Identifying Traffic (Layer 3/4 Class Maps)" section on page 1-12 for more information.
Step 11	**class-map** *name2* **match access-list** *acl2* **Example:** `ciscoasa(config)# class-map cws_class2` `ciscoasa(config-cmap)# match access-list SCANSAFE_HTTPS`	(Optional) Creates an additional class map, for example for HTTPS traffic. You can create as many classes as needed for this service policy rule.
Step 12	**policy-map** *name* **Example:** `ciscoasa(config)# policy-map cws_policy`	Adds or edits a policy map that sets the actions to take with the class map traffic. The policy map in the default global policy is called global_policy. You can edit this policy, or create a new one. You can only apply one policy to each interface or globally.
Step 13	**class** *name1* **Example:** `ciscoasa(config-pmap)# class cws_class1`	Identifies the class map created in Step 9.
Step 14	**inspect scansafe** *scansafe_policy_name1* [**fail-open** | **fail-close**] **Example:** `ciscoasa(config-pmap-c)# inspect scansafe cws_inspect_pmap1 fail-open`	Enables Cloud Web Security inspection on the traffic in this class. Specify the inspection class map name that you created in Step 1. Specify **fail-open** to allow traffic to pass through the ASA if the Cloud Web Security servers are unavailable. Specify **fail-close** to drop all traffic if the Cloud Web Security servers are unavailable. **fail-close** is the default.
Step 15	**class** *name2* **inspect scansafe** *scansafe_policy_name2* [**fail-open** | **fail-close**] **Example:** `ciscoasa(config-pmap)# class cws_class2` `ciscoasa(config-pmap-c)# inspect scansafe cws_inspect_pmap2 fail-open`	(Optional) Identifies a second class map that you created in Step 11, and enables Cloud Web Security inspection for it. You can configure multiple class maps as needed.
Step 16	**service-policy** *policymap_name* {**global** | **interface** *interface_name*} **Example:** `ciscoasa(config)# service-policy cws_policy inside`	Activates the policy map on one or more interfaces. **global** applies the policy map to all interfaces, and **interface** applies the policy to one interface. Only one global policy is allowed. You can override the global policy on an interface by applying a service policy to that interface. You can only apply one policy map to each interface. See the "Applying Actions to an Interface (Service Policy)" section on page 1-17 for more information.

Examples

The following example configures two classes: one for HTTP and one for HTTPS. Each ACL exempts traffic to www.cisco.com and to tools.cisco.com, and to the DMZ network, for both HTTP and HTTPS. All other traffic is sent to Cloud Web Security, except for traffic from several whitelisted users and groups. The policy is then applied to the inside interface.

```
hostname(config)# class-map type inspect scansafe match-any whitelist1
hostname(config-cmap)# match user user1 group cisco
hostname(config-cmap)# match user user2
hostname(config-cmap)# match group group1
hostname(config-cmap)# match user user3 group group3

hostname(config)# policy-map type inspect scansafe cws_inspect_pmap1
hostname(config-pmap)# parameters
hostname(config-pmap-p)# http
hostname(config-pmap-p)# default group default_group
hostname(config-pmap-p)# class whitelist1
hostname(config-pmap-c)# whitelist

hostname(config)# policy-map type inspect scansafe cws_inspect_pmap2
hostname(config-pmap)# parameters
hostname(config-pmap-p)# https
hostname(config-pmap-p)# default group2 default_group2
hostname(config-pmap-p)# class whitelist1
hostname(config-pmap-c)# whitelist

hostname(config)# object network cisco1
hostname(config-object-network)# fqdn www.cisco.com
hostname(config)# object network cisco2
hostname(config-object-network)# fqdn tools.cisco.com
hostname(config)# object network dmz_network
hostname(config-object-network)# subnet 10.1.1.0 255.255.255.0

hostname(config)# access-list SCANSAFE_HTTP extended deny tcp any4 object cisco1 eq 80
hostname(config)# access-list SCANSAFE_HTTP extended deny tcp any4 object cisco2 eq 80
hostname(config)# access-list SCANSAFE_HTTP extended deny tcp any4 object dmz_network eq
80
hostname(config)# access-list SCANSAFE_HTTP extended permit tcp any4 any4 eq 80

hostname(config)# access-list SCANSAFE_HTTPS extended deny tcp any4 object cisco1 eq 443
hostname(config)# access-list SCANSAFE_HTTPS extended deny tcp any4 object cisco2 eq 443
hostname(config)# access-list SCANSAFE_HTTP extended deny tcp any4 object dmz_network eq
443
hostname(config)# access-list SCANSAFE_HTTPS extended permit tcp any4 any4 eq 443

hostname(config)# class-map cws_class1
hostname(config-cmap)# match access-list SCANSAFE_HTTP
hostname(config)# class-map cws_class2
hostname(config-cmap)# match access-list SCANSAFE_HTTPS

hostname(config)# policy-map cws_policy
hostname(config-pmap)# class cws_class1
hostname(config-pmap-c)# inspect scansafe cws_inspect_pmap1 fail-open
hostname(config-pmap)# class cws_class2
hostname(config-pmap-c)# inspect scansafe cws_inspect_pmap2 fail-open
hostname(config)# service-policy cws_policy inside
```

(Optional) Configuring Whitelisted Traffic

If you use user authentication, you can exempt some traffic from being filtered by Cloud Web Security based on the username and/or groupname. When you configure your Cloud Web Security service policy rule, you can reference the whitelisting inspection class map. Both IDFW and AAA user credentials can be used with this feature.

Although you can achieve the same results of exempting traffic based on user or group when you configure the service policy rule, you might find it more straightforward to use a whitelist instead. Note that the whitelist feature is only based on user and group, not on IP address.

Detailed Steps

	Command	Purpose
Step 1	**class-map type inspect scansafe** [**match-all** | **match-any**] *name* **Example:** `ciscoasa(config)# class-map type inspect scansafe match-any whitelist1`	Creates an inspection class map for whitelisted users and groups. The *class_map_name* argument is the name of the class map up to 40 characters in length. The **match-all** keyword is the default, and specifies that traffic must match all criteria to match the class map. The **match-any** keyword specifies that the traffic matches the class map if it matches at least one of the criteria. The CLI enters class-map configuration mode, where you can enter one or more **match** commands.
Step 2	**match** [**not**] {[**user** *username*] [**group** *groupname*]} **Example:** `ciscoasa(config-cmap)# match`	The **match** keyword, followed by a specific username or groupname, specifies a user or group to whitelist. The **match not** keyword specifies that the user and/or group should be filtered using Web Cloud Security. For example, if you whitelist the group "cisco," but you want to scan traffic from users "johncrichton" and "aerynsun," you can specify **match not** for those users. Repeat this command to add as many users and groups as needed.

Example

The following example whitelists the same users and groups for the HTTP and HTTPS inspection policy maps:

```
hostname(config)# class-map type inspect scansafe match-any whitelist1
hostname(config-cmap)# match user user1 group cisco
hostname(config-cmap)# match user user2
hostname(config-cmap)# match group group1
hostname(config-cmap)# match user user3 group group3

hostname(config)# policy-map type inspect scansafe cws_inspect_pmap1
hostname(config-pmap)# parameters
hostname(config-pmap-p)# http
hostname(config-pmap-p)# default group default_group
hostname(config-pmap-p)# class whitelist1
hostname(config-pmap-c)# whitelist

hostname(config)# policy-map type inspect scansafe cws_inspect_pmap2
hostname(config-pmap)# parameters
```

```
hostname(config-pmap-p)# https
hostname(config-pmap-p)# default group2 default_group2
hostname(config-pmap-p)# class whitelist1
hostname(config-pmap-c)# whitelist
```

(Optional) Configuring the User Identity Monitor

When you use IDFW, the ASA only downloads user identity information from the AD server for users and groups included in active ACLs; the ACL must be used in a feature such as an access rule, AAA rule, service policy rule, or other feature to be considered active. Because Cloud Web Security can base its policy on user identity, you may need to download groups that are not part of an active ACL to get full IDFW coverage for all your users. For example, although you can configure your Cloud Web Security service policy rule to use an ACL with users and groups, thus activating any relevant groups, it is not required; you could use an ACL based entirely on IP addresses.The user identity monitor feature lets you download group information directly from the AD agent.

Restrictions

The ASA can only monitor a maximum of 512 groups, including those configured for the user identity monitor and those monitored through active ACLs.

Detailed Steps

Command	Purpose
user-identity monitor {**user-group** [*domain-name*\\]*group-name* | **object-group-user** *object-group-name*} **Example:** `ciscoasa(config)# user-identity monitor user-group CISCO\\Engineering`	Downloads the specified user or group information from the AD agent. • **user-group**—Specifies a group name inline. Although you specify 2 backslashes (\\) between the domain and the group, the ASA modifies the name to include only one backslash when it sends it to Cloud Web Security, to comply with Cloud Web Security notation conventions. • **object-group-user**—Specifies an **object-group user** name. This group can include multiple groups.

Configuring the Cloud Web Security Policy

After you configure the ASA service policy rules, launch the ScanCenter Portal to configure Web content scanning, filtering, malware protection services, and reports.

Detailed Steps

Go to: https://scancenter.scansafe.com/portal/admin/login.jsp.

For more information, see the Cisco ScanSafe Cloud Web Security Configuration Guides:

http://www.cisco.com/en/US/products/ps11720/products_installation_and_configuration_guides_list.html

Monitoring Cloud Web Security

Command	Purpose
show scansafe server	Shows the status of the server, whether it is the current active server, the backup server, or unreachable.
show scansafe statistics	Shows total and current HTTP(S) connections.
show conn scansafe	Shows all Cloud Web Security connections, as noted by the capitol Z flag.
show service policy inspect scansafe	Shows the number of connections that are redirected or white listed by a particular policy.
See the following URL: http://Whoami.scansafe.net	From a client, access this web site to determine if your traffic is going to the Cloud Web Security server.

The **show scansafe server** command shows whether or not the Cloud Web Security proxy servers are reachable:

```
hostname# show scansafe server
ciscoasa# Primary: proxy197.scansafe.net (72.37.244.115) (REACHABLE)*
ciscoasa# Backup: proxy137.scansafe.net (80.254.152.99)
```

The **show scansafe statistics** command shows information about Cloud Web Security activity, such as the number of connections redirected to the proxy server, the number of current connections being redirected, and the number of whitelisted connections:

```
hostname# show scansafe statistics
Current HTTP sessions : 0
Current HTTPS sessions : 0
Total HTTP Sessions : 0
Total HTTPS Sessions : 0
Total Fail HTTP sessions : 0
Total Fail HTTPS sessions : 0
Total Bytes In : 0 Bytes
Total Bytes Out : 0 Bytes
HTTP session Connect Latency in ms(min/max/avg) : 0/0/0
HTTPS session Connect Latency in ms(min/max/avg) : 0/0/0
```

The **show service policy inspect scansafe** command shows the number of connections that are redirected or whitelisted by a particular policy:

```
hostname(config)# show service-policy inspect scansafe
Global policy:
  Service-policy: global_policy
    Class-map: inspection_default
Interface inside:
  Service-policy: scansafe-pmap
    Class-map: scansafe-cmap
      Inspect: scansafe p-scansafe fail-open, packet 0, drop 0, reset-drop 0,
v6-fail-close 0
Number of whitelisted connections: 0
Number of connections allowed without scansafe inspection because of "fail-open" config: 0
Number of connections dropped because of "fail-close" config: 0
Number of HTTP connections inspected: 0
Number of HTTPS connections inspected: 0
Number of HTTP connections dropped because of errors: 0
Number of HTTPS connections dropped because of errors: 0
```

Configuration Examples for Cisco Cloud Web Security

- Single Mode Example, page 25-18
- Multiple Mode Example, page 25-19
- Whitelist Example, page 25-19
- Directory Integration Examples, page 25-20
- Cloud Web Security with Identity Firewall Example, page 25-22

Single Mode Example

The following example shows a complete configuration for Cisco Cloud Web Security:

Configure ACLs

We recommend that you split the traffic by creating separate HTTP and HTTPS class maps so that you know how many HTTP and HTTPS packets have gone through.

Then, if you need to troubleshoot you can run debug commands to distinguish how many packets have traversed each class map and find out if you are pushing through more HTTP or HTTPS traffic:

```
hostname(config)# access-list web extended permit tcp any any eq www
hostname(config)# access-list https extended permit tcp any any eq https
```

Configure Class Maps

```
hostname(config)# class-map cmap-http
hostname(config-cmap)# match access-list web

hostname(config)# class-map cmap-https
hostname(config-cmap)# match access-list https
```

Configure Inspection Policy Maps

```
hostname(config)# policy-map type inspect scansafe http-pmap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# default group httptraffic
hostname(config-pmap-p)# http

hostname(config)# policy-map type inspect scansafe https-pmap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# default group httpstraffic
hostname(config-pmap-p)# https
```

Configure Policy Maps

```
hostname(config)# policy-map pmap-webtraffic
hostname(config-pmap)# class cmap-http
hostname(config-pmap-c)# inspect scansafe http-pmap fail-close

hostname(config-pmap)# class cmap-https
hostname(config-pmap-c)# inspect scansafe https-pmap fail-close
```

Configure Service Policy

```
hostname(config)# service-policy pmap-webtraffic interface inside
```

Configure Cloud Web Security on the ASA

```
hostname(config)# scansafe general-options
```

```
hostname(cfg-scansafe)# server primary ip 192.168.115.225 web 8080
hostname(cfg-scansafe)# retry-count 5
hostname(cfg-scansafe)# license 366C1D3F5CE67D33D3E9ACEC265261E5
```

Multiple Mode Example

The following example enables Cloud Web Security in context one with the default license and in context two with the authentication key override:

```
! System Context
!
ciscoasa(config)#scansafe general-options
ciscoasa(cfg-scansafe)#server primary ip 180.24.0.62 port 8080
ciscoasa(cfg-scansafe)#retry-count 5
ciscoasa(cfg-scansafe)#license FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
ciscoasa(cfg-scansafe)#publickey <path to public key>
!
context one
 allocate-interface GigabitEthernet0/0.1
 allocate-interface GigabitEthernet0/1.1
 allocate-interface GigabitEthernet0/3.1
 scansafe
 config-url disk0:/one_ctx.cfg
!
context two
 allocate-interface GigabitEthernet0/0.2
 allocate-interface GigabitEthernet0/1.2
 allocate-interface GigabitEthernet0/3.2
 scansafe license 366C1D3F5CE67D33D3E9ACEC265261E5
!
config-url disk0:/two_ctx.cfg
!
```

Whitelist Example

Configure what access-list traffic should be sent to Cloud Web Security:

```
access-list 101 extended permit tcp any4 any4 eq www
access-list 102 extended permit tcp any4 any4 eq https

class-map web
 match access-list 101
class-map https
 match access-list 102
```

To configure the whitelist to ensure user1 is in this access-list range to bypass Cloud Web Security:

```
class-map type inspect scansafe match-any whiteListCmap
 match user LOCAL\user1
```

To attach class-maps to the Cloud Web Security Policy map:

```
policy-map type inspect scansafe ss
 parameters
  default user user1 group group1
  http
 class whiteListCmap
  whitelist

policy-map type inspect scansafe ss2
```

```
 parameters
  default user user1 group group1
  https
 class whiteListCmap
  whitelist
```

After creating this inspect policy, attach it to the policy map to be assigned to the service group:

```
policy-map pmap
 class web
  inspect scansafe ss fail-close
class https
  inspect scansafe ss2 fail-close
```

Then attach the policy map to a service-policy to make it in effect globally or by ASA interface:

```
service-policy pmap interface inside
```

Directory Integration Examples

This section contains various example configurations for directory integration. See also Chapter 38, "Configuring the Identity Firewall," in the general operations configuration guide.

- Configuring the Active Directory Server Using LDAP, page 25-20
- Configuring the Active Directory Agent Using RADIUS, page 25-21
- Creating the ASA as a Client on the AD Agent Server, page 25-21
- Creating a Link Between the AD Agent and DCs, page 25-21
- Testing the AD Agent, page 25-21
- Configuring the Identity Options on the ASA, page 25-21
- Configuring the User Identity Options and Enabling Granular Reporting, page 25-21
- Monitoring the Active Directory Groups, page 25-22
- Downloading the Entire Active-User Database from the Active Directory Server, page 25-22
- Downloading the Database from the AD Agent, page 25-22
- Showing a List of Active Users, page 25-22

Configuring the Active Directory Server Using LDAP

The following example shows how to configure the Active Directory server on your ASA using LDAP:

```
hostname(config)# aaa-server AD protocol ldap
hostname(config-aaa-server-group)# aaa-server AD (inside) host 192.168.116.220
hostname(config-aaa-server-host)# ldap-base-dn DC=ASASCANLAB,DC=local
hostname(config-aaa-server-host)# ldap-scope subtree
hostname(config-aaa-server-host)# server-type microsoft
hostname(config-aaa-server-host)# server-port 389
hostname(config-aaa-server-host)# ldap-login-dn
cn=administrator,cn=Users,dc=asascanlab,dc=local
hostname(config-aaa-server-host)# ldap-login-password Password1
```

Configuring the Active Directory Agent Using RADIUS

The following example shows how to configure the Active Directory Agent on your ASA using RADIUS:

```
hostname(config)# aaa-server adagent protocol radius
hostname(config-aaa-server-group)# ad-agent-mode
hostname(config-aaa-server-group)# aaa-server adagent (inside) host 192.168.116.220
hostname(config-aaa-server-host)# key cisco123
hostname(config-aaa-server-host)# user-identity ad-agent aaa-server adagent
```

Creating the ASA as a Client on the AD Agent Server

The following example shows how to create the ASA as a client on the Active Directory agent server:

```
c:\IBF\CLI\adacfg client create -name ASA5520DEVICE -ip 192.168.116.90 -secret cisco123
```

Creating a Link Between the AD Agent and DCs

The following example shows how to create a link between the Active Directory Agent and all DCs for which you want to monitor logon/logoff events:

```
c:\IBF\CLI\adacfg.exe dc create -name DCSERVER1 -host W2K3DC -domain
W2K3DC.asascanlab.local -user administrator -password Password1
c:\IBF\CLI\adacfg.exe dc list
```

Running the last command should show the status as "UP."

For the AD_Agent to monitor logon/logoff events, you need to ensure that these are logged on ALL DCs that are actively being monitored. To do this, choose:

Start > Administrative Tools > Domain Controller Security Policy

Local policies > Audit Policy > Audit account logon events (success and failure)

Testing the AD Agent

The following example shows how to configure the test Active Directory Agent so that it can communicate with the ASA:

```
hostname# test aaa-server ad-agent adagent
Server IP Address or name: 192.168.116.220
INFO: Attempting Ad-agent test to IP address <192.168.116.220> (timeout: 12 seconds)
INFO: Ad-agent Successful
```

See also the following command: **show user-identity ad-agent**.

Configuring the Identity Options on the ASA

The following example shows how to configure the identity options on the ASA:

```
hostname(config)# user-identity domain ASASCANLAB aaa-server AD
hostname(config)# user-identity default-domain ASASCANLAB
```

Configuring the User Identity Options and Enabling Granular Reporting

The following example shows how to configure the user identity options that send user credentials to the ASA and enable granular user reporting from the proxy server:

```
hostname(config)# user-identity inactive-user-timer minutes 60
hostname(config)# user-identity action netbios-response-fail remove-user-ip
hostname(config)# user-identity user-not-found enable
hostname(config)# user-identity action mac-address-mismatch remove-user-ip
hostname(config)# user-identity ad-agent active-user-database full-download
```

If you are using more than one domain, then enter the following command:

```
hostname(config)# user-identity domain OTHERDOMAINNAME
```

Monitoring the Active Directory Groups

The following example shows how to configure Active Directory groups to be monitored:

```
hostname(config)# user-identity monitor user-group ASASCANLAB\\GROUPNAME1
hostname(config)# user-identity monitor user-group ASASCANLAB\\GROUPNAME2
hostname(config)# user-identity monitor user-group ASASCANLAB\\GROUPNAME3
```

Caution Remember to save your configuration once the above is completed.

Downloading the Entire Active-User Database from the Active Directory Server

The following command updates the specified import user group database by querying the Active Directory server immediately without waiting for the expiration of poll-import-user-group-timer:

```
hostname(config)# user-identity update import-user
```

Downloading the Database from the AD Agent

The following example shows how to manually start the download of the database from the Active Directory Agent if you think the user database is out of sync with Active Directory:

```
hostname(config)# user-identity update active-user-database
```

Showing a List of Active Users

The following example shows how to show the Active users:

```
hostname# show user-identity user active list detail
```

There are two download modes with Identify Firewall: Full download and On-demand.

- Full download—Whenever a user logs into the network, the IDFW tells the ASA the User identity immediately (recommended on the ASA 5510 and above).
- On-demand—Whenever a user logs into the network, the ASA requests the user identity from AD (ADHOC) (recommended on the ASA 5505 due to memory constraints).

Cloud Web Security with Identity Firewall Example

The following example shows how to configure Cloud Web Security with Identity Firewall on the ASA:

```
hostname# sh run
ASA Version 100.8(24)32
!
hostname QFW-201-QASS
```

```
domain-name uk.scansafe.net
enable password liqhNWIOSfzvir2g encrypted
passwd liqhNWIOSfzvir2g encrypted
names
!
interface Ethernet0/0
 nameif inside
 security-level 100
 ip address 192.168.116.90 255.255.255.0
!
interface Ethernet0/1
 nameif outside
 security-level 0
 ip address 192.168.114.90 255.255.254.0
!
interface Ethernet0/2
 shutdown
 no nameif
 no security-level
 no ip address
!
interface Ethernet0/3
 shutdown
 no nameif
 no security-level
 no ip address
!
interface Management0/0
 shutdown
 no nameif
 no security-level
 no ip address
!
boot system disk0:/asa100824-32-k8.bin
ftp mode passive
dns server-group DefaultDNS
 domain-name uk.scansafe.net
object network obj0192.168.116.x
 subnet 192.168.116.0 255.255.255.0
access-list 101 extended permit tcp any any eq www
access-list 101 extended permit tcp any any eq https
access-list web extended permit tcp any any eq www
access-list icmp extended permit icmp any any
access-list https extended permit tcp any any eq https
!
scansafe general-options
 server primary ip 192.168.115.225 web 8080
 retry-count 5
 license 366C1D3F5CE67D33D3E9ACEC26789534f
!
pager lines 24
logging buffered debugging
mtu inside 1500
mtu outside 1500
no failover
icmp unreachable rate-limit 1 burst-size 1
no asdm history enable
arp timeout 14400
!
object network obj0192.168.116.x
 nat (inside,outside) dynamic interface
access-group 101 in interface outside
route outside 0.0.0.0 0.0.0.0 192.168.114.19 1
timeout xlate 3:00:00
```

```
timeout pat-xlate 0:00:30
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00 mgcp-pat 0:05:00
timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00 sip-disconnect 0:02:00
timeout sip-provisional-media 0:02:00 uauth 0:05:00 absolute
timeout tcp-proxy-reassembly 0:01:00
timeout floating-conn 0:00:00
dynamic-access-policy-record DfltAccessPolicy
aaa-server AD protocol ldap
aaa-server AD (inside) host 192.168.116.220
 server-port 389
 ldap-base-dn DC=ASASCANLAB,DC=local
 ldap-scope subtree
 ldap-login-password *****
 ldap-login-dn cn=administrator,cn=Users,dc=asascanlab,dc=local
 server-type microsoft
aaa-server adagent protocol radius
 ad-agent-mode
aaa-server adagent (inside) host 192.168.116.220
 key *****
user-identity domain ASASCANLAB aaa-server AD
user-identity default-domain ASASCANLAB
user-identity action netbios-response-fail remove-user-ip
user-identity poll-import-user-group-timer hours 1
user-identity ad-agent aaa-server adagent
user-identity user-not-found enable
user-identity monitor user-group ASASCANLAB\\GROUP1
user-identity monitor user-group ASASCANLAB\\GROUPNAME
no snmp-server location
no snmp-server contact
crypto ca trustpool policy
telnet timeout 5
ssh 192.168.0.0 255.255.255.0 inside
ssh 192.168.21.0 255.255.255.0 inside
ssh timeout 30
console timeout 0
threat-detection basic-threat
threat-detection statistics access-list
no threat-detection statistics tcp-intercept
!
class-map cmap-https
 match access-list https
class-map inspection_default
 match default-inspection-traffic
class-map cmap-http
 match access-list web
!
!
policy-map type inspect dns preset_dns_map
 parameters
  message-length maximum client auto
  message-length maximum 512
policy-map type inspect scansafe ss
 parameters
  default user john group qa
  http
policy-map type inspect scansafe https-pmap
 parameters
  https
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225
```

```
  inspect h323 ras
  inspect ip-options
  inspect netbios
  inspect rsh
  inspect rtsp
  inspect skinny
  inspect esmtp
  inspect sqlnet
  inspect sunrpc
  inspect tftp
  inspect sip
  inspect xdmcp
policy-map type inspect scansafe http-pmap
 parameters
  default group http-scansafe
  http
policy-map pmap-http
 class cmap-http
  inspect scansafe http-pmap fail-open
 class cmap-https
  inspect scansafe https-pmap fail-open
!
service-policy pmap-http global
prompt hostname context
no call-home reporting anonymous
call-home
 profile CiscoTAC-1
  no active
  destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService
  destination address email callhome@cisco.com
  destination transport-method http
  subscribe-to-alert-group diagnostic
  subscribe-to-alert-group environment
  subscribe-to-alert-group inventory periodic monthly
  subscribe-to-alert-group configuration periodic monthly
  subscribe-to-alert-group telemetry periodic daily
Cryptochecksum:667ba936945b370c394806a63548e7a0
: end
QFW-201-QASS#
```

Related Documents

Related Documents	URL
Cisco ScanSafe Cloud Web Security Configuration Guides	http://www.cisco.com/en/US/products/ps11720/products_installation_and_configuration_guides_list.html

Feature History for Cisco Cloud Web Security

Table 25-1 lists each feature change and the platform release in which it was implemented.

Table 25-1 Feature History for Cloud Web Security

Feature Name	Platform Releases	Feature Information
Cloud Web Security	9.0(1)	This feature was introduced. Cisco Cloud Web Security provides content scanning and other malware protection service for web traffic. It can also redirect and report about web traffic based on user identity. We introduced or modified the following commands: **class-map type inspect scansafe**, **default user group**, **http**[**s**] (parameters), **inspect scansafe**, **license**, **match user group**, **policy-map type inspect scansafe**, **retry-count**, **scansafe**, **scansafe general-options**, **server** {**primary** | **backup**}, **show conn scansafe**, **show scansafe server**, **show scansafe statistics**, **user-identity monitor**, **whitelist**.