**Overview**

The Cisco Cloud Web Security (CWS) Chrome Extension allows you to configure your Chrome OS devices (such as Chromebook and Chromebox) to redirect web traffic to Cisco’s CWS service. This allows you to protect and inspect web traffic, enforce policy, and gather analytics on web usage on your Chrome OS devices. The solution is delivered as a Chrome browser extension that can be silently deployed by a system administrator to the device with no end user involvement. There is nothing for the user to configure. The extension provides seamless web security for the Chrome browser.

In the CWS Chrome Extension, the following features are available:

**■** Traffic redirection to the CWS service for HTTP and HTTPS traffic

**■** Reporting and analytics

**■** Warn, block, authenticate, and anonymize policy rules

**■** HTTPS inspection (optionally decrypt and secure HTTPS traffic)

**■** Malware scanning in the cloud

**■** SafeSearch and SearchAhead for supported search engines

**■** User identity from the user’s Google account

**■** SAML and Clientless Authentication (using cookie surrogates)

**■** Cloud Bypass/Whitelisting (define a list of IP ranges and/or domains for which requests should not go through the CWS proxy)

**■** Trusted Network Detection (disable the CWS Chrome Extension when on-premises to leverage the security of an on-premises connector appliance)

**■** Secondary failover proxy (secondary CWS proxy available for redundancy)

Supported Devices and Operating Systems

The CWS Chrome Extension is supported on:

**■** Chrome OS devices, such as Chromebook and Chromebox, running version 52 or newer.

The CWS Chrome Extension is *not* supported on:

**■** Chrome browser on OS X, Windows, and Linux.

**■** Devices running variations or third-party distributions of Chrome OS, such as Neverware CloudReady.

STEP 1: Create CWS Groups

**Have school create CWS Group and Authentication key**

****

****

****

****

Copy the authentication key to a notepad document. You will need it to configure the .json file

**STEP 2 - Chrome OS Account and Groups**

If your organization makes use of managed or supervised users for your Chrome OS devices (for example, if your users sign in to the Chrome OS device using a Google account or Active Directory account), then the CWS Chrome Extension can leverage this information for user identity. When this feature is enabled, the email address of the user is sent with every request and is visible in reports in Cisco ScanCenter. Only the email address is reported; no group information is sent.

By default, this feature is disabled. To enable this feature, ensure that the **UserIdentityEnabled** flag in the configuration file is set to true. For more information, see the Configuration section.

If you wish to also use group identity and apply group-based policies, we recommended that you either:

• Configure the Chrome OS device with a group license key as well as leveraging the Chrome OS user identity. For more information on how to configure the Chrome OS device with a group key, see the Configuration section.

• Create a custom group within Cisco ScanCenter, and add the associated user email addresses to the custom group. For more information on custom groups, see the Cisco ScanCenter Administrator Guide.

If you wish to verify that user/group identity is being correctly reported, browse to http://whoami.scansafe.net. The **authUserName** field on this page should be populated with the email address of the Chrome OS profile.

**Note:** If you set **UserIdentityEnabled** to true and also make use of Clientless Authentication or SAML, any user identity from Clientless Authentication or SAML overwrites the user identity obtained from the Chrome

**Step 3 - Recommended Chrome OS Policies**

To ensure correct operation of the CWS Chrome Extension, and to ensure that the user cannot modify or interfere with the extension, there are several Google Chrome enterprise policies that should be enabled. Use the Google Apps for Work or Education console to enable these policies for your users.

For information on how to configure Chrome OS policies, see the Google documentation at https://support.google.com/chrome/a/answer/2657289?hl=en, or contact your Google support representative.

Access the Google Chrome policy configuration by following these steps:

**1.** Log in to the Google Admin Console.

**2.** From the Admin Console, select **Device Management**.

**3.** In the menu on the left of the screen, select **Chrome Management**.

**4.** Select **User Settings**.

**5.** On the left, select the organization you wish to configure.

The recommended policies are as follows:

**■ User Experience > Developer Tools**

o Disable by selecting the **Never allow use of built-in developer tools** option.

o This ensures that your users cannot open Chrome's developer tools to manipulate the state of the CWS Chrome Extension.

**■ Network > Proxy Settings**

o Select the **Allow user to configure** option.

o This allows the CWS Chrome Extension to manage the Chrome OS proxy configuration.

o Once the CWS Chrome Extension is installed on the device, the user cannot change the proxy, despite the name of the value of this setting.

o If this setting is set to any other value, the CWS Chrome Extension may not function as expected.

**■ Network > Data Compression Proxy**

o Disable the user’s ability to select data compression by selecting the **Always disable data compression proxy** option.

o The data compression function provided by Google Chrome acts as a web proxy. If the data compression proxy is enabled, it may conflict with the CWS Chrome Extension and result in unexpected or undefined behavior.

**■ Apps and Extensions > Allow or Block All Apps and Extensions**

o Select **Block all apps and extensions except the ones I allow**.

o Blocking all extensions and applications except the ones you explicitly allow ensures that conflicting programs (for example, proxy or VPN extensions) cannot be installed by the user to circumvent the CWS Chrome Extension.

**16**

**■ Security > Incognito mode**

o Disable by selecting the **Disallow incognito mode** option.

o By default, the CWS Chrome Extension does not operate in incognito mode unless the user explicitly opts in to run the extension in this mode. This is due to Google Chrome policy limitations and cannot be overridden by enterprise policy. Therefore, we recommend disabling incognito mode.

**Step 4 - Bypass rules in the ASA**

It is necessary to create a bypass rule for the primary and secondary proxy sites specified in the .json file in the ASA prior to deploying the extension in order for the chromebooks to continue to be able to access the internet as currently configured. If the local TC does not have access the ASA, this change must be implemented by DIS.

Lookup current CWS towers on their ASA (Or in Krissy’s spreadsheet on J:\Departmental Information\Projects\Active\Network Security\K12 Broadband Project\All Spreadsheets\Krissy's Spreadsheets\Portals(Not Sent).xlsx; Sheet: NEW LR TOWERS, Columns M and N)

***We just need the access17xx and 6xx the rest will always be .chrome.cloudsec.cisco.com***

**Log into the ASA and enter configuration mode. (enable, \*password\*; config t)**

**Type or copy the following commands, modified with the appropriate tower numbers.**

object network obj-access1704.chrome.cloudsec.cisco.com

 fqdn access1704.chrome.cloudsec.cisco.com

exit

access-list http-traffic line 1 extended deny tcp any object obj-access1704.chrome.cloudsec.cisco.com eq www

access-list https-traffic line 1 extended deny tcp any object obj-access1704.chrome.cloudsec.cisco.com eq https

object network obj-access604.chrome.cloudsec.cisco.com

 fqdn access604.chrome.cloudsec.cisco.com

exit

access-list http-traffic line 1 extended deny tcp any object obj-access604.chrome.cloudsec.cisco.com eq www

access-list https-traffic line 1 extended deny tcp any object obj-access604.chrome.cloudsec.cisco.com eq https

exit

copy run start

confirm the creation of the objects by browsing to the ASA with ASDM-ID and looking under the Configuration/Firewall/Policy Objects for the access objects.

**Step 5 - Configuring the JSON file**

The CWS Chrome Extension is configured through a JSON file containing key and value pairs which determine the behavior and functionality of the extension.

For most configuration options, if you do not specify an option or specify an invalid value, the extension uses a default value instead. See the Configuration Properties table below for information on which options are available and their default values. Note that some options, such as the license key and proxy host, are mandatory. If you do not specify a correct value for a mandatory configurable option, all web traffic on the Chrome OS device gets blocked until either a correct configuration is provided or the extension is removed.

Below is an example of the configuration JSON file that is required to configure the CWS Chrome Extension. Use this example in conjunction with the Configuration Properties to customize a configuration file that suits your needs.

{

"LicenseKey": {

"Value": "ABCDEFGHIJKLMNOPQRSTUV1234567890" [place the key created in CWS for the group here]

},

"ProxyHost": {

"Value": "access123.chrome.cloudsec.cisco.com" [place the number of the primary CWS tower here]

},

"SecondaryProxyHost": {

"Value": "access345.chrome.cloudsec.cisco.com" [place the number of the secondary CWS tower here]

},

"TNDHost": {

"Value": ""

},

"TNDCheckIntervalMinutes": {

"Value": 1

},

"WhitelistEnabled": {

"Value": true

},

"WhitelistCheckIntervalMinutes": {

"Value": 60

},

"UserIdentityEnabled": {

"Value": true

}

}

Do not alter the remainder of the file.

Save the file where it will be easy to find, with a name incorporating Chromebook(s) or incorporating the CWS group name.

Rename the file with a .json extension.

Create a group key and a .json file for each CWS group created for Chromebooks.

**HTTPS INSPECTION (Optional)**

HTTPS Inspection is an optional capability which can be enabled.

The CWS Chrome Extension supports HTTPS Inspection functionality. HTTPS Inspection allows you to decrypt HTTPS traffic and apply policy, scan the traffic for malware, and gather reporting metrics. You can define which domains, IP addresses, and categories of HTTPS traffic get inspected. These policies are configured in Cisco ScanCenter.

In order for the HTTPS Inspection functionality to work, you must first set up a Certificate Authority (CA) in Cisco ScanCenter. Since Chromebooks only support certificates in PEM format, the certificate generated using or uploaded to the ScanCenter portal can be converted from CRT to PEM format using the following command:

*openssl x509 –inform DER –outform PEM –text –in certificateName.crt -out certificateName.pem*

Once the conversion is done, install the PEM CA certificate on all Chrome OS devices where the HTTPS Inspection policy is being applied. The Google Apps for Work or Education consoles provide a mechanism for deployment of trusted CA certificates to devices within an organization. For more information, see https://support.google.com/chrome/a/answer/3505249?hl=en&ref\_topic=3504941, or contact your Google support representative. When installing the certificate via the Google Apps for Work or Education consoles, make sure to select the **Use this certificate as an HTTPS certificate authority** check box.

Google advises that certain Google-owned or Google-affiliated domains are exempted from any SSL decryption processes, such as the HTTPS inspection functionality provided by CWS. The list of domains can be found on Google’s support pages at https://support.google.com/chrome/a/answer/6334001?hl=en&ref\_topic=3504941.

**Note:** This list changes from time to time. We recommend that you ensure the domains included in this list are not included in any HTTPS inspection policy. If you have a HTTPS inspection policy that decrypts all HTTPS traffic or decrypts categories of traffic that may clash with the above domains, you can add the domains as exceptions. For more information, see the Cisco ScanCenter Administrator Guide

**Step 6 - Deployment through Google Apps for Work or Education**

In order to deploy the CWS Chrome Extension, a Google Apps for Work or Google Apps for Education subscription is required. For each Chrome OS device that you wish to deploy the extension on, you must also have a Chrome Management license. Cisco does not provide, support, or sell subscriptions to these services. For more information, contact your Google sales representative, or see https://www.google.com/work/chrome/management-console/.

Once you have your JSON configuration file ready, you can deploy the extension to one or more organizations through the Google Apps for Work or Education console.

To deploy the extension, complete the following steps:

**1.** Log in to your Google Apps for Work or Education console.

**2.** Click the **Device Management** icon.

Cisco Cloud Web Security Chrome Extension Administrator Guide Compatibility with Other Connectors

**8**

**3.** Under **Device Settings** in the menu to the left of the screen, select **Chrome Management**.

**4.** You are taken to the Chrome Management page. Now select **App Management**.

**5.** In the left side panel, enter the **unique identifier (see supporting text file)** for the CWS Chrome Extension in the **Find or Update Apps** search box. Press the **Search** button.

**6.** In the results list, click on **CWS Chrome Extension**.

**7.** You are taken to a page where you can configure the extension. Select **User settings**.

**8.** Now, for each organization you wish to deploy the CWS Chrome Extension, select it under the Orgs menu. Toggle **Force installation** to on, and upload the JSON configuration file for that relevant organization. Then select **Save**.

**9.** Once you have completed these steps, the extension and configuration should be silently pushed to the target Chrome OS devices within minutes. If you uploaded the extension to an internal server, ensure the Chrome OS device is switched on and connected to your corporate network. If the extension still does not appear, try rebooting the device.

If you wish to update the configuration after deployment, you can upload a new configuration file using the Google Admin Console, as you did when performing the initial deployment. In most cases, the CWS Chrome Extension detects and updates the configuration accordingly within a few minutes. In some cases, the user may have to log out and back in to the Chrome OS device for the changes to take effect.

**Troubleshooting and Logging**

The CWS Chrome Extension generates log messages which can be used to analyze and debug problems with the extension. These log messages are sent to the Google Chrome console within the Developer Tools interface. Ensure that you enable the Developer Tools on only the device you are using for troubleshooting, not on all of your user devices.

To gather these troubleshooting logs, follow these steps:

**1.** Temporarily enable Chrome's Developer Tools if your Chrome OS policy has disabled it.

**Note:** We recommend you keep Chrome's Developer Tools disabled for security purposes. However, in order to gather debug logs, you need to temporarily re-enable this functionality.

**2.** Open Google Chrome on the Chrome OS device and reproduce the issue.

**3.** From the Google Chrome menu, select **More Tools** and then select **Extensions**.

**4.** The extensions page opens. In the top right of this page, select **Developer mode**.

**5.** Under the Cisco Cloud Web Security extension, click the **html/background.html** link under **Inspect Views**.

**6.** Developer tools opens. Select the **Console** tab.

**7.** You can now see the CWS Chrome Extension debug logs. To save them, right click any of the log messages and select **Save as**.