White Paper

HTTPS Inspection with Cisco CWS

What is HTTPS?

Hyper Text Transfer Protocol Secure (HTTPS) is a secure version of the Hyper Text Transfer Protocol (HTTP). It is a combination of HTTP and a cryptographic protocol Secure Sockets Layer (SSL). The HTTPS protocol leverages SSL's encryption and secure identification of web server capabilities to enable secure web transactions.

HTTPS connections are often used by web browsers and applications for transactions of sensitive financial and personal information, both for personal and corporate intentions.

When a user connects to a website via HTTPS, the Web site encrypts the session with a digital certificate that is signed by a trusted third party. Most commonly used web browsers display a padlock or similar icon to indicate that the website is secure and the site's URL begins with https:// instead of http://.

HTTPS Trends

The proportion of HTTPS traffic is rapidly growing, and this is seen in popular websites and also evident in the web traffic patterns that Cisco sees.

- Google and Yahoo enforce HTTPS for all web searches, webmail, and other provided services. Other sites are rapidly following suit.
- In many countries, all Government hosted websites must comply to HTTPS for all transactions.
- It is good practice for any financial and healthcare websites to adhere to HTTPS.
- Most cloud based applications and services such as file sharing have deployed HTTPS as the standard.
- Facebook, LinkedIn and other popular Social and Business Networking sites all encrypt web traffic via HTTPS.

- Research conducted as part of Cisco's 2016 Annual Security Report revealed that encrypted traffic, particularly HTTPS, has reached a tipping point. The research shows that it already consistently represents over 50 percent of bytes transferred (due to overhead and the larger content that is sent via HTTPS, such as transfers to file storage sites). So while not yet representing the majority of transactions, HTTPS will soon become the dominant form of traffic on the Internet.

HTTPS Data Flow

When an HTTPS request is initiated, the following steps occur:

1. The client's browser checks the certificate of the requested web server to ensure that the site is trusted.
2. The browser and the web server determine the best common encryption type they can both use to send and receive the SSL data stream. In general, they will strive to prioritize the most secure encryption method available, however browsers can be configured to use only specific encryption types.
3. The browser and server exchange public and private encryption keys to use when encrypting the information that is sent and received.
4. The browser and server start communicating using the encryption and the browser provides an indication to the user that the web pages are being processed securely.

Once an SSL connection is established, it is considered secure as it is encrypted and uninterrupted from the web server to the client's browser or application where the certificate is located.

So is the HTTPS Protocol Really Secure?

SSL encryption methods are constantly evolving and getting stronger and more sophisticated, making it extremely difficult to intercept and snoop on HTTPS traffic. However, encrypting the traffic doesn't necessarily guarantee that it is secured. In fact, in many cases HTTPS leads to a false sense of security that can introduce a rapidly growing security gap to enterprises (see the Cisco Annual Security Report).

- Encryption applies only to the transferring of data in the traffic stream between the client and the web server. The web server could still be delivering malicious content, either hosting it intentionally or as a result of a legitimate and trusted site getting compromised.
- Certificates may have expired or may have been revoked, stolen, or forged (all it takes is a valid email address to obtain a digital certificate). While most browsers can detect some of the above, the ultimate decision on whether to proceed or not is the end users'.
- As HTTPS traffic is encrypted from end to end, the encryption can also be abused to allow malicious files to pass straight through a web proxy when not decrypted for inspection. This can leave a gap in visibility into malware and other threats lurking within, preventing policy enforcement.
- Many admins are still hesitant to apply HTTPS inspection policies due to privacy and legal concerns, as well as the perceived overhead and complications that are involved.

As mentioned, it is unfortunate that many admins equate HTTPS traffic with "safe" traffic. However, what it really means is that they are blind to the content of HTTPS requests.

To summarize, while SSL encryption of web traffic is certainly an important factor that contributes towards overall security, it should probably be considered more of a privacy shield for protecting sensitive data, and therefore parts of that content should still be inspected.

HTTPS Inspection with CWS

Cisco CWS provides admins with the ability to configure flexible decryption policies for SSL encrypted web traffic and applications, enabling scanning for threats and applying of policies.

When CWS HTTPS Inspection is used, the cloud proxy initiates the HTTPS web request to the web server on behalf of the client and terminates the session in the cloud proxy where the traffic is decrypted for inspection. CWS then re-encrypts the traffic and creates an additional HTTPS stream from the cloud proxy back to the client, using Cisco's SSL certificate. This method of HTTPS decryption is also known as "Man in the Middle".

In order for clients to trust SSL connections from CWS's cloud proxy, Cisco's root certificate must be installed in the trusted certificate store on all clients. Just as a malicious or other unknown website would not be trusted by the browser, nor would CWS be trusted without installing this certificate. In this way, CWS acts as an intermediate Certificate Authority (CA) where the users assume they are connecting directly to the requested website via SSL, but they are actually connected to the cloud proxy. Alternatively, a CA root certificate that is already installed on the clients can be utilized by generating a Certificate Signing Request (CSR) in the CWS admin portal.

The Capabilities Provided with HTTPS Inspection

All web requests that are matched for HTTPS inspection will be decrypted in the CWS cloud proxy and then are subject to the standard Web Filtering policy and the various malware scanning engines.

With HTTPS Inspection enabled, CWS checks SSL certificates of the requested hosts for suspicious activity such as:

- Stolen certificates
- Certificates that are revoked as a result of an Online Certificate Status Protocol (OCSP) query
- Expired certificates
- Bogus certificates

If a host's certificate falls into any of the above, CWS will detect this and replicate the error into the certificate sent to the user, which will then trigger the relevant security error in the browser. The user will thus be alerted and given the opportunity to abort their connection to that site as usual.

If the requested HTTPS site is using Intermediate certificates, these too will be checked in the same way.

Even without HTTPS Inspection enabled, CWS checks the names of requested hosts. This is done by reading the Host/IP from CONNECT headers, or processing the Server Name Indication (SNI) requested by the browser upon the initial SSL handshake, and from those fields it is able to derive the host name of the requested site. Once the host name is known, a basic level of URL Filtering (based on the category of the requested host) and Web Reputation checking can be applied against the request at time of connection. The full requested URL is not known when HTTPS Inspection is not used, so the functionality described is based only on the host name seen in the request, and the content is still encrypted from end to end, so nothing is scanned by the cloud proxy.

Creating SSL Certificates

Trusted certificates are generated by admins in the CWS Admin Portal. There are two options available for doing this:

1. Generate a new SSL certificate in the admin portal. Admins can choose the expiry period to span up to seven years. When this certificate is issued, Cisco CWS is the Certificate Authority and in order for the certificate to be trusted, it should be exported from ScanCenter and installed into the bowsers of all clients that are browsing through the CWS service.
2. Download a Certificate Signing Request (CSR) and use it with a tool such as Microsoft Certificate Services or OpenSSL to generate and upload your own certificate to the admin portal. In this scenario the customer's organization is the Certificate Authority. The advantage of using this method is that admins can sign a certificate that is already installed and trusted by the users' browsers, without any further need to distribute and install the certificates.

Cisco TAC can assist customers who have a need to import their own 3rd party certificate with their public and private keys.

What to Inspect and What Not to Inspect

As mentioned, there is no silver bullet when it comes to selecting which HTTPS content to inspect and various organizations will have different needs and considerations. For this reason, CWS provides different options for admins to create flexible and granular decryption rules for HTTPS web traffic.

- Inspection rules can be based on URL categories; the same categories that are used in the web filtering rules.
- Specific domains and URL's can be listed for decryption and also used in an exclusion list that has higher priority than the URL categories.
- Through a single checkbox, all web applications listed in the Application Visibility & Control (AVC) page of the web filtering filters that are SSL encrypted will also be decrypted for inspection. The AVC controls provide access control to applications and the ability to apply granular control over activities and actions performed on these sites, but as most of the popular web applications are today encrypted, the users' activities cannot be detected without decrypting the traffic.

All of the above options can be applied alone, or together in separate decryption rules, and the rules can be associated with specific groups of users and the SSL certificates that were generated.

Data Privacy Concerns and Solutions Provided

When it comes to protecting users' privacy, CWS provides full flexibility to apply best practices and decrypt only the content that should be inspected. Through granular controls, admins and security professionals can avoid interfering with their users' sensitive and personal interactions with websites under specific categories such as banking and financial, healthcare, and government.

In most jurisdictions organizations are required by law to inform their users that their encrypted traffic is being inspected. A customizable HTML warning page can be presented to end-users before the SSL connection is established, stating that their session will be decrypted, giving the user the choice to continue or to opt out. Delivering this disclaimer message before the SSL connection is established enables organizations to comply with privacy laws.

To further protect users' sensitive information, when HTTPS traffic is scanned CWS does not log the Path and Query attributes of the URL, only the Host will be logged. For example, if a user browses to Google and searches for "cisco cloud web security" and hits Enter, the full URL will be: https://www.google.com/?gws_rd=ssl#q=cisco+cloud+web+security

That full URL can be broken down to these three attributes:

- Host: https://www.google.com
- Path: /?gws_rd=ssl# (where on the site the user browsed to)
- Query: q=cisco+cloud+web+security (what the user searched for)

For privacy reasons CWS will log only the Host and not the Path nor the Query for HTTPS traffic that is inspected. When HTTPS traffic is not inspected, the Path and Query fields will not be known in any case.

Legal Responsibilities and Compliance

It is the admin's responsibility to determine if it is legal to inspect HTTPS traffic in their jurisdiction. By configuring the HTTPS Inspection function, admins are in effect allowing the service to inspect their users' HTTPS traffic. While all such inspection is carried out automatically rather than by individuals, such decryption may nonetheless be in breach of privacy laws in certain countries. By enabling this functionality, the admin agrees that they have the legal right to decrypt this traffic in all relevant jurisdictions where applied and that they have obtained all necessary consents from their users to do so. In case of global deployments spanning different regions and countries, flexible policies can cater for admins to apply HTTPS Inspection only in the locations where it is allowed and exclude countries where tampering with HTTPS traffic is against the law.

When the admin first configures HTTPS Inspection, they will be reminded of their responsibility through a disclaimer page.

Other Good Practices

- Due to the nature of the HTTPS decryption technology, there may occasionally be some sites or content within pages that do not display correctly, and certain applications may not function as designed. Together with privacy considerations, it is less likely for an organization to want to inspect all HTTPS content.
- With the granularity that CWS provides for HTTPS Inspection, it is possible to select only a few specific categories and domains that should be decrypted for inspection. Some customers do not want to inspect HTTPS content in general, but only want visibility into web applications so that they can be controlled through the AVC filters.
- Some customers start by compiling a list of business critical applications within their organization and test those with HTTPS inspection enabled for a group of test users.
- Other customers take the approach that with the increasing number of HTTPS sites, the risk of malware within commonly used sites is also growing and therefore the HTTPS inspection policy should cover a broader spectrum of content. In these cases, the admin should be prepared to consider excluding only a few sensitive categories and specific hosts from the inspection.

Frequently Asked Questions for HTTPS Inspection

Should users expect any performance impact when HTTPS Inspection is enabled?

When HTTPS Inspection is enabled, a small performance overhead is introduced at the time of the initial SSL request, but this should not be noticeable by end-users. Once the SSL handshake has been completed, the decrypted traffic is inspected by the CWS scanning engines and at this point the latency will be the same as for any other traffic that gets scanned by the cloud proxies.

Should HTTPS Inspection be used for inspecting guest traffic?

Due to the procedures involved in importing trusted certificates, it is usually not practical to apply HTTPS inspection in environments where guests' devices are protected by CWS, and in some cases the same challenges may also apply in BYOD environments.

Should organizations inspect all HTTPS traffic?

As is the case with web filtering policies, the inspection requirements for HTTPS traffic also differ from company to company and there is no "one size fits all" here either. This requires admins and security professionals to create, customize, and fine-tune their HTTPS Inspection policies according to their needs and their legal and privacy considerations.

Are there ways to simplify the task of rolling out certificates to all browsers?

In many cases Group Policy Objects (GPO) and other software distribution tools can assist admins with the task of rolling out certificates to browsers of all their users' machines.

What are the advantages of generating a Certificate Signing Request (CSR) instead of using Cisco's trusted certificate?

Generating a CSR eliminates the necessity of certificate distribution as it leverages SSL certificates that are already installed and trusted by the browsers. The process required for generating these certificates is more complex and it is recommended for the admin to have an understanding of SSL software before performing this procedure. Admins should note that when creating a CSR, there is a restriction of 30 minutes maximum between the time that the initial CSR is generated to the time that the signed certificate is uploaded back into the CWS admin portal (this is due to the 30 minutes' timeout that will be incurred on a static page in the interface without activity).

Can non-browser traffic also be inspected with HTTPS Inspection?

While HTTPS is used primarily for browser-based traffic, there are a number of applications that may also use the HTTPS protocol to communicate with web servers and some may follow the browser settings and use the browser's trusted certificate store, but others may not. Admins should check whether these applications can be manually configured to trust certificates. Various browsers have different procedures for importing trusted certificates and some simply follow the settings of Microsoft's Internet Explorer browser on Windows platforms.

Can some level of control or protection be applied to HTTPS traffic without using HTTPS Inspection?

Yes! CWS extracts the requested host name from the SNI or the CONNECT headers upon the initial SSL handshake, enabling the application of URL Filtering and Web Reputation to HTTPS traffic without the need to decrypt. The verdict that these engines return is only as accurate as the top level host name that is extracted from the SNI and does not take into consideration the exact URL (host + path) that is encrypted within the SSL thread, which cannot be seen without enabling HTTPS Inspection. As a result, this process provides only basic web

filtering and reputation services and the content is still encrypted from end to end, so nothing is scanned by the cloud proxy. It should also be noted that in case of any policy blocks through this method, the requested page will fail to load but the end-user will not be presented with a notification page.

Is cloud-based authentication supported with HTTPS traffic?

When using cloud-based authentication with cookie surrogates, the cookies will not be seen unless HTTPS Inspection is enabled. Even with HTTPS Inspection enabled there may still be embedded content in web pages that cannot always be authenticated due to Cross Origin Resource Sharing (CORS) restrictions. These limitations can be overcome by using IP surrogates, but even then the initial web request that triggers the authentication process should be HTTP-based.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Printed in USA 05/16