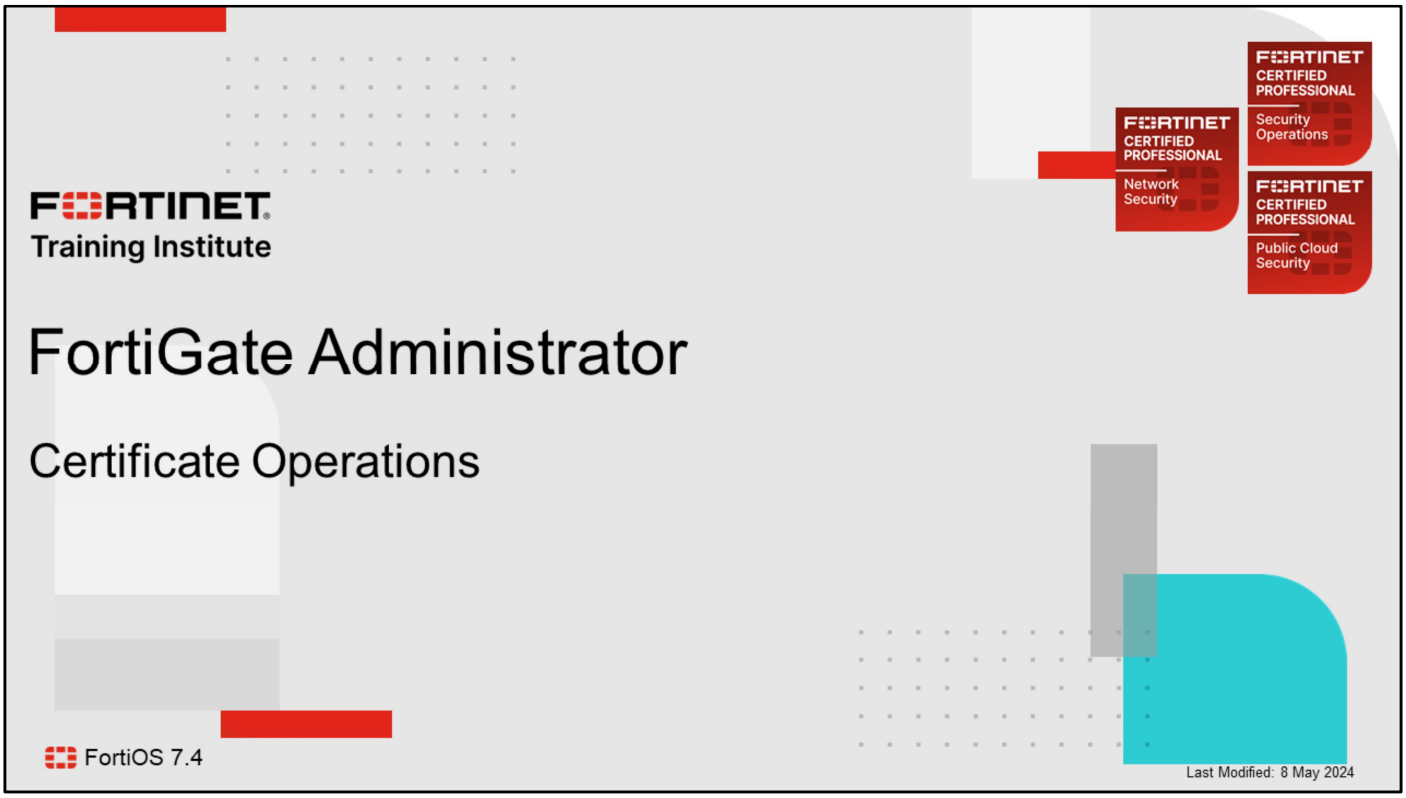
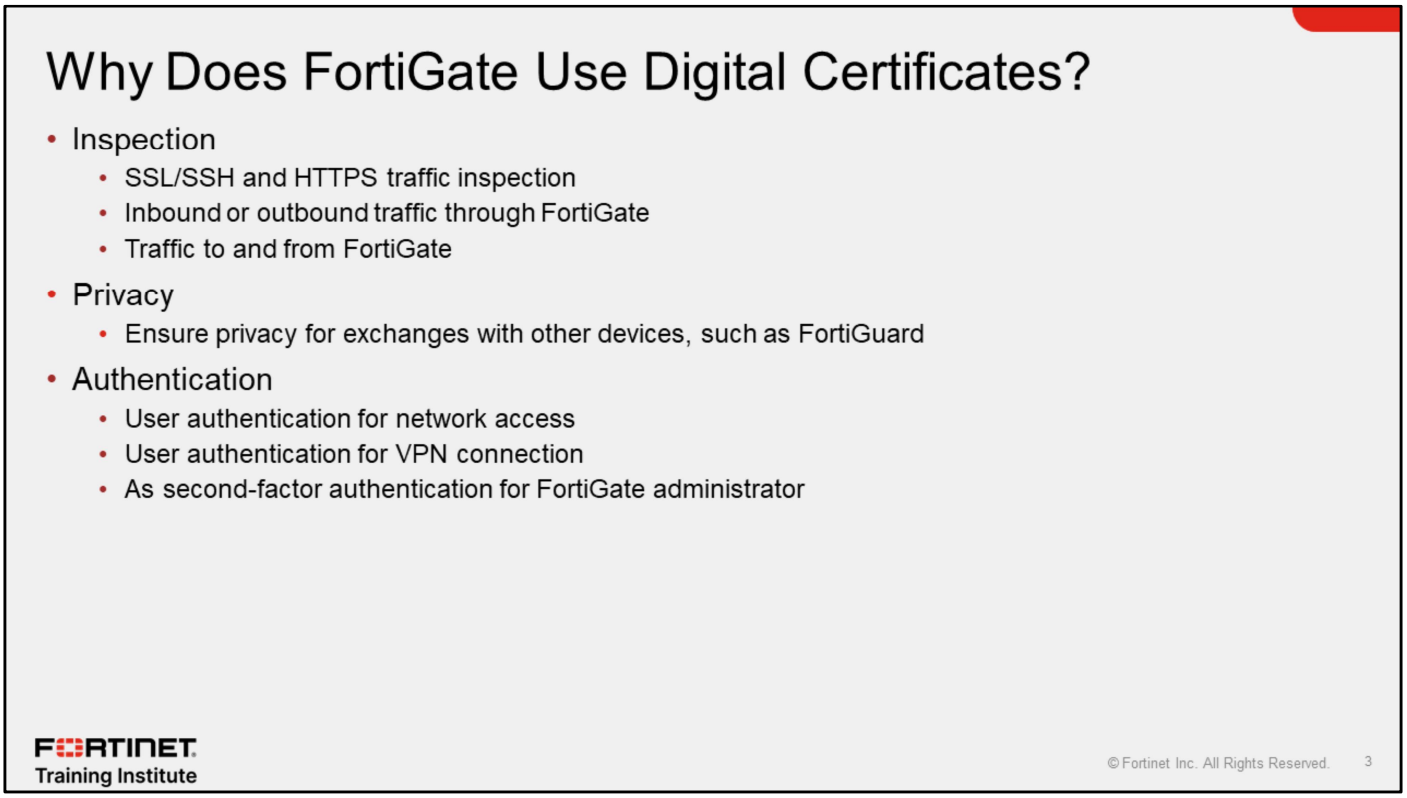


In this lesson, you will learn why FortiGate uses digital certificates, and how to configure FortiGate to use certificates for SSL and SSH traffic inspection.

Objectives

- Configure FortiGate for full SSL/SSH inspection
- Install private CA certificates on endpoints
- Troubleshoot certificate issues

After completing this section, you should be able to achieve the objectives shown on this slide.

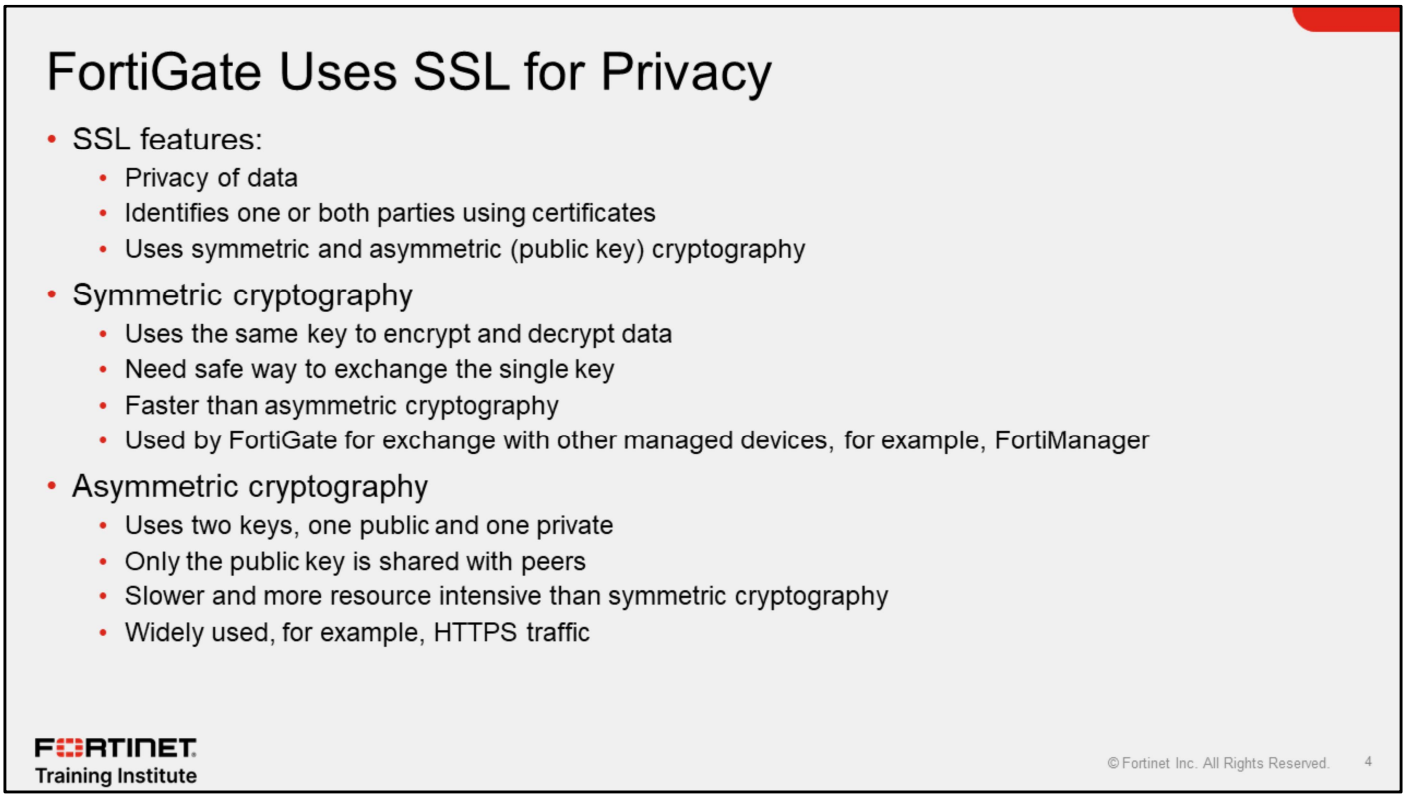


FortiGate uses digital certificates to enhance security in multiple areas.

FortiGate uses digital certificates for inspection, mainly outbound or inbound traffic inspection. If FortiGate trusts the certificate, it permits the connection. But if FortiGate does not trust the certificate, it can prevent the connection. How you configure FortiGate determines the behavior; however, other policies that are being used may also affect whether FortiGate accepts or rejects connection attempts. FortiGate can also inspect certificates to identify people and devices (in the network and on the internet), before it permits a person or device to make a full connection to the entity that it is protecting.

FortiGate uses digital certificates to enforce privacy. Certificates, and their associated private keys, ensure that FortiGate can establish a private SSL connection to another service, such as FortiGuard, or a web browser or web server.

FortiGate also uses certificates for authentication. Users who have certificates issued by a known and trusted CA can authenticate on FortiGate to access the network or establish a VPN connection. Administrator users can use certificates as a second-factor authentication credential to log in to FortiGate.



FortiGate uses SSL to ensure that data remains private when connecting with servers, such as FortiGuard, and with clients, such as a web browser. Another feature of SSL is that FortiGate can use it to identify one or both parties using certificates. SSL uses symmetric and asymmetric cryptography to establish a secure session between two points.

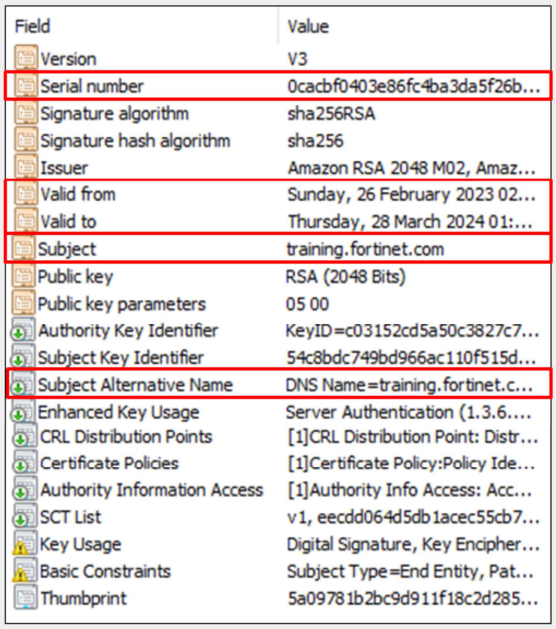
It is beneficial to understand the high-level process of an SSL handshake in order to understand how FortiGate secures private sessions.

For symmetric cryptography, the same key is used to encrypt and decrypt the traffic. This process requires fewer computing resources and is faster than asymmetric cryptography. However, one drawback is the requirement to share the key between participating devices in a safe way. When FortiGate establishes an SSL session between itself and another device, it must share the symmetric key (or rather the value required to produce it—usually the password you configure), so that data can be encrypted by one side, sent, and decrypted by the other side.

Asymmetric cryptography uses a pair of keys: One key performs one function, and the other key performs the opposite function. When FortiGate connects to a web server, for example, it uses the web server public key to encrypt a string known as the premaster secret. The web server private key decrypts the premaster secret.

Using Certificates to Identify a Person or Device

- What is a digital certificate?
  - A digital identity produced and signed by a certificate authority (CA)
  - Analogy: passport or driver's license
- How does FortiGate use certificates to identify devices and people?
  - The **Subject** and **Subject Alternative Name** fields in the certificate identify the device or person associated with the certificate
- FortiGate uses the X.509v3 certificate standard



FORTINET Training Institute

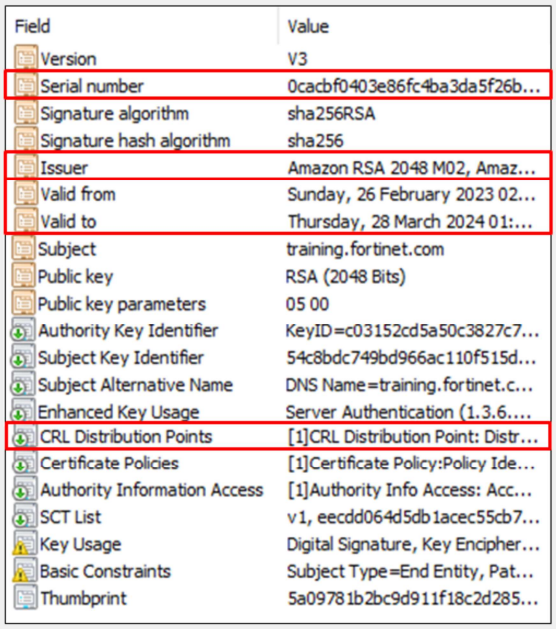
What is a digital certificate?

A digital certificate is a digital document produced, and signed by a certificate authority (CA). It identifies an end entity, such as a person (for example, Joe Bloggins), a device (for example, webserver.acme.com), or thing (for example, a certificate revocation list). FortiGate identifies the device or person by reading the common name (CN) value in the **Subject** field, which is expressed as a distinguished name (DN). FortiGate could also use alternate identifiers, shown in the **Subject Alternative Name** field, whose values could be a network ID or an email address, for example. FortiGate can use the **Subject Key Identifier**, and **Authority Key Identifier** values to determine the relationship between the issuer of the certificate (identified in the **Issuer** field), and the certificate.

FortiGate supports the X.509v3 certificate standard, which is the most common standard for certificates.

How Does FortiGate Trust Certificates?

- FortiGate does the following checks against a certificate before trusting it and using it:
  - Revocation check
  - CA certificate possession
    - FortiGate uses the **Issuer** value to determine if FortiGate possesses the corresponding CA certificate
    - Without the corresponding CA certificate, FortiGate cannot trust the certificate
  - Validity dates
  - Digital signature validation
    - The verification of the digital signature on the certificate must pass

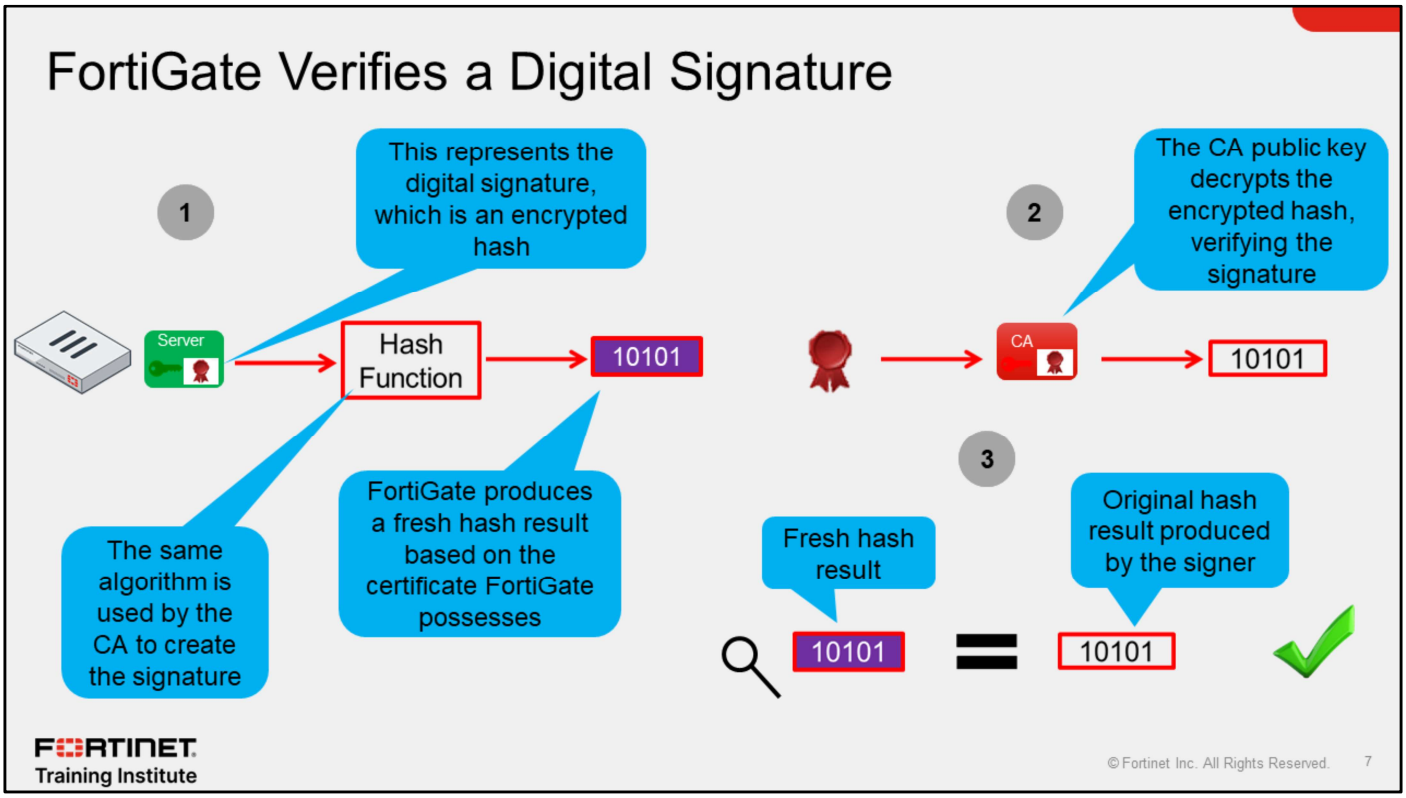


Field	Value
Version	V3
Serial number	0cacbf0403e86fc4ba3da5f26b...
Signature algorithm	sha256RSA
Signature hash algorithm	sha256
Issuer	Amazon RSA 2048 M02, Amaz...
Valid from	Sunday, 26 February 2023 02...
Valid to	Thursday, 28 March 2024 01:...
Subject	training.fortinet.com
Public key	RSA (2048 Bits)
Public key parameters	05 00
Authority Key Identifier	KeyID=c03152cd5a50c3827c7...
Subject Key Identifier	54c8bdc749bd966ac110f515d...
Subject Alternative Name	DNS Name=training.fortinet.c...
Enhanced Key Usage	Server Authentication (1.3.6....
CRL Distribution Points	[1]CRL Distribution Point: Distr...
Certificate Policies	[1]Certificate Policy:Policy Ide...
Authority Information Access	[1]Authority Info Access: Acc...
SCT List	v1, eecdd064d5db1acec55cb7...
Key Usage	Digital Signature, Key Encipher...
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint	5a09781b2bc9d911f18c2d285...

FORTINET Training Institute

FortiGate runs the following checks before it trusts the certificate:

- Checks the certificate revocation lists (CRLs) locally on FortiGate to verify if the certificate has been revoked by the CA.
  - FortiGate can download the relevant CRLs, and check if the serial number of the certificate is listed on the CRL. If the certificate is listed, it means that it has been revoked, and it is no longer trusted.
  - FortiGate also supports the Online Certificate Status Protocol (OCSP). When FortiGate uses the OCSP, it interacts with an OCSP responder (FortiAuthenticator acts as the OCSP responder) to check if the certificate is still valid.
- Reads the value in the **Issuer** field to determine if it has the corresponding CA certificate. Without the CA certificate, FortiGate does not trust the certificate.
- Verifies that the current date is between the **Valid From** and **Valid To** values. If it is not, the certificate is rendered invalid.
- Validates the signature on the certificate. The signature must be successfully validated.

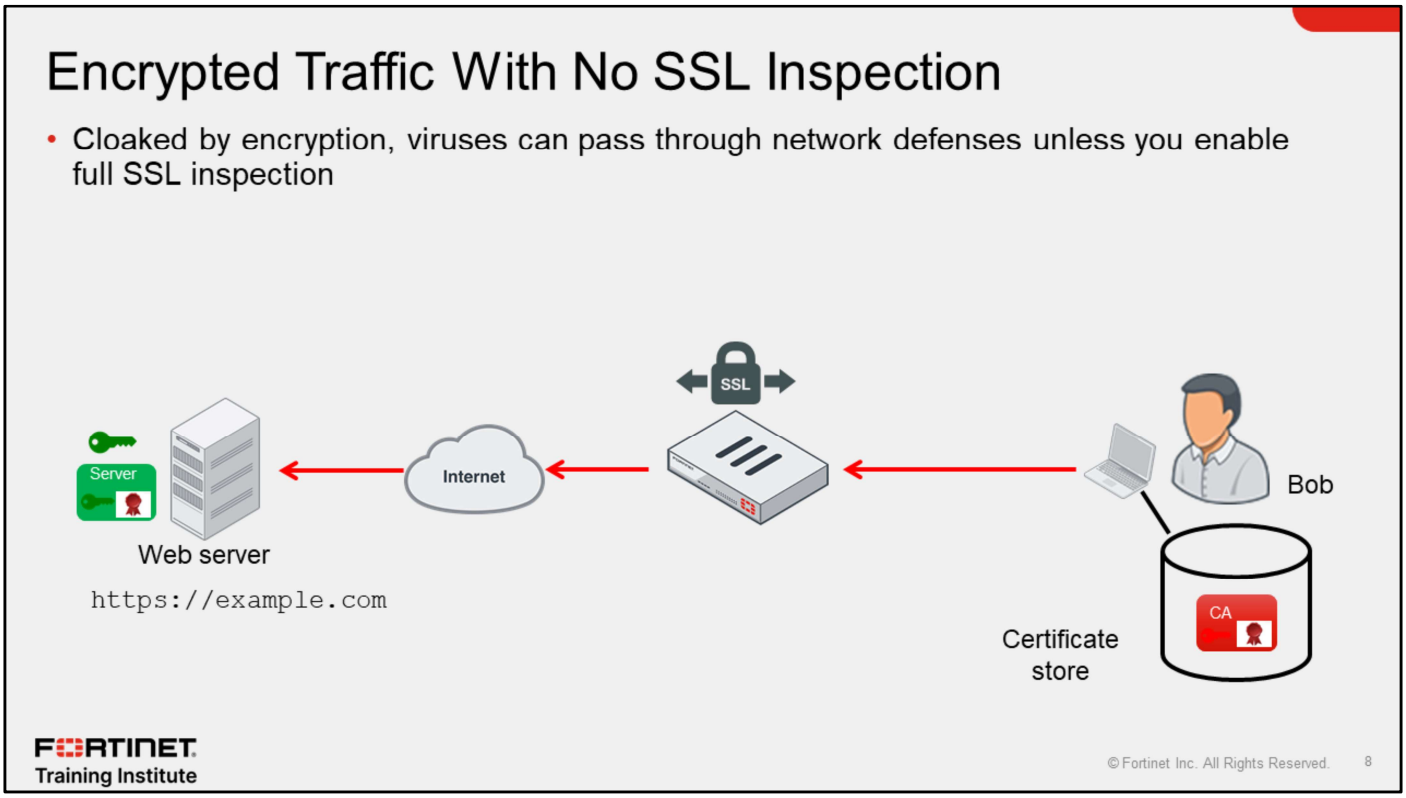


Before it generates a digital signature, the CA runs the content of the certificate through a hash function, which produces a hash result. The hash result, which is a mathematical representation of the data, is referred to as the *original hash result*. The CA encrypts the original hash result using its *private key*. The encrypted hash result is the digital signature.

When FortiGate verifies the digital signature, it runs the certificate through a hash function, producing a fresh hash result. FortiGate must use the same hash function, or hashing algorithm, that the CA used to create the digital signature. The hashing algorithm is identified in the certificate.

FortiGate decrypts the encrypted hash result (or digital signature) using the CA *public key* and applying the same algorithm that the CA used to encrypt the hash result. This process verifies the signature. If the key cannot restore the encrypted hash result to its original value, then the signature verification fails.

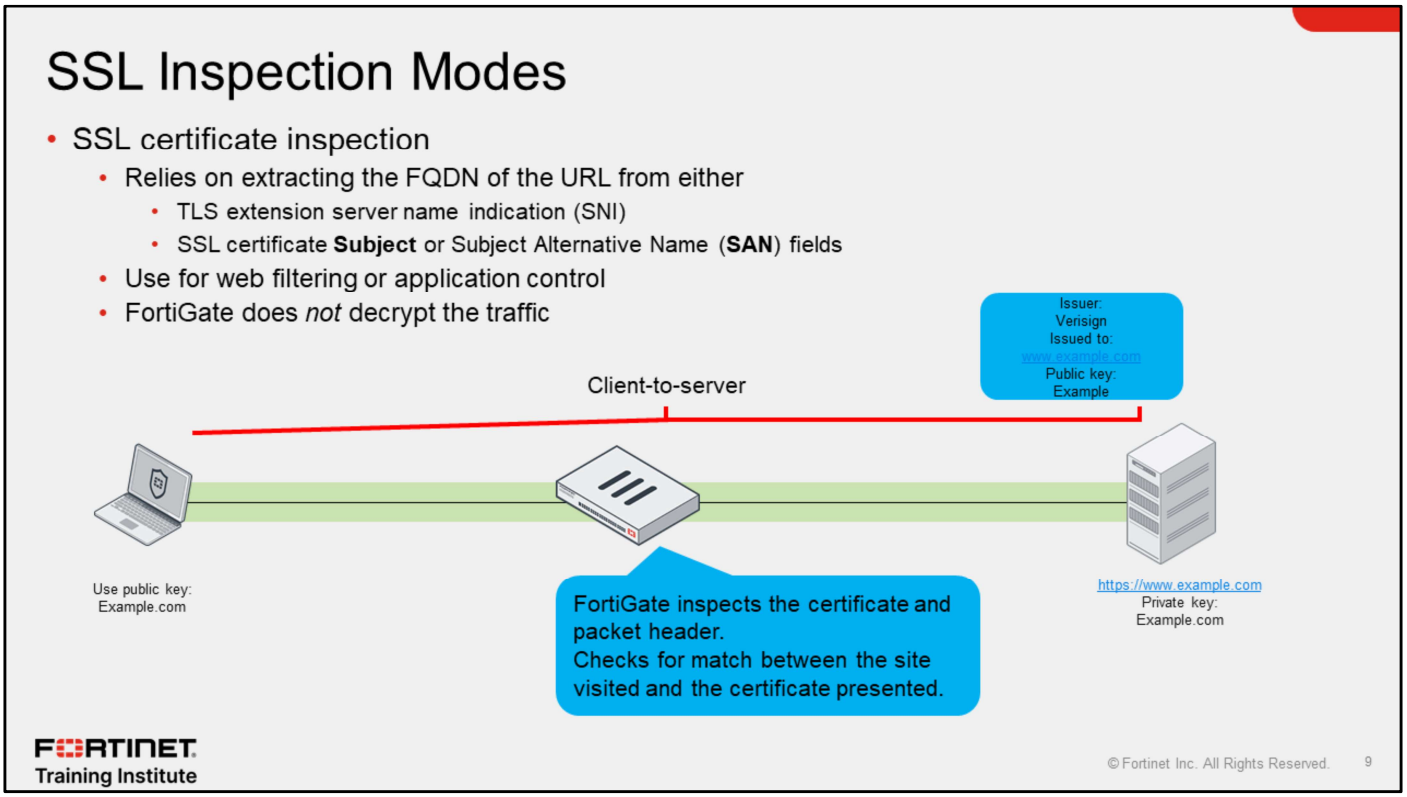
In the third, and final, part of the verification process, FortiGate compares the fresh hash result to the original hash result. If the two values are identical, then the integrity of the certificate is confirmed. If the two hash results are different, then the version of the certificate that FortiGate has is not the same as the one that the CA signed, and data integrity fails.



While there are benefits to using HTTPS, there are risks associated with its use as well, because encrypted traffic can be used to get around normal defenses. For example, if a session is encrypted when you download a file containing a virus, the virus might get past your network security measures.

In the example shown on this slide, Bob connects to a site with a certificate issued by a legitimate CA. Because the CA is an approved CA, the CA verification certificate is in Bob's certificate store, and Bob's browser is able to establish an SSL session with the `example.com` site. However, unknown to Bob, the example.com site has been infected with a virus. The virus, cloaked by encryption, passes through FortiGate undetected, and enters Bob's computer. The virus is able to breach security because full SSL inspection is not enabled.

You can use full SSL inspection, also known as deep inspection, to inspect encrypted sessions.



There are two SSL inspection modes: SSL certificate inspection, and full SSL inspection.

When using SSL certificate inspection, FortiGate is not decrypting the traffic. During the exchange of hello messages at the beginning of an SSL handshake, FortiGate parses the server name indication (SNI) from client Hello, which is an extension of the TLS protocol. The SNI tells FortiGate the hostname of the SSL server, which is validated against the DNS name before receipt of the server certificate. If there is no SNI exchanged, then FortiGate identifies the server by the value in the **Subject** field or **SAN** (subject alternative name) field in the server certificate.

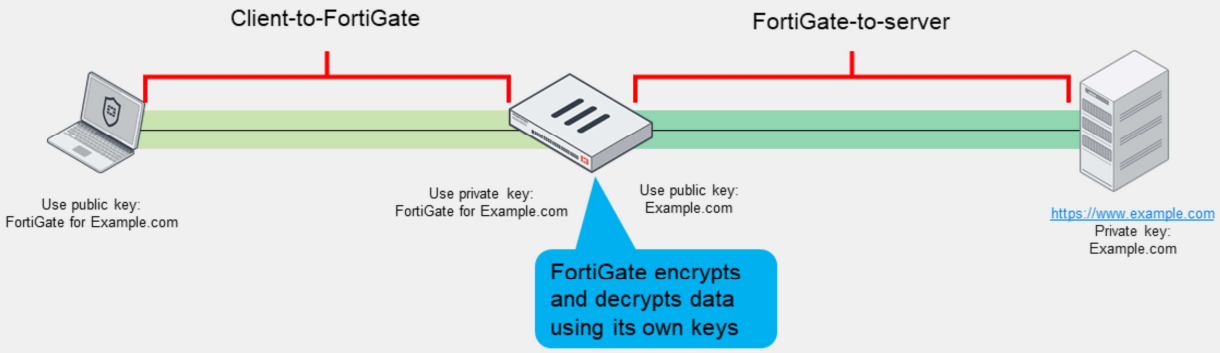
First, FortiGate tries to get the URL from the SNI field. The SNI field is a TLS extension that contains the complete URL that the user is connecting to. It is supported by most modern browsers.
If the SNI field is not present (because the web client may not support it), FortiGate proceeds to inspect the server digital certificate to get information about the URL or the domain.

The only security features you can apply using SSL certificate inspection mode are web filtering and application control. SSL certificate inspection allows FortiGate to identify the website visited or the application in use and categorize it. You can, therefore, use it to make sure that the HTTPS protocol isn't used as a workaround to access sites you have blocked using web filtering.

Note that while offering some level of security, certificate inspection does not allow FortiGate to inspect the flow of encrypted data.

SSL Inspection Modes (Contd)

- Full SSL Inspection
  - FortiGate acts as a man-in-the middle proxy
  - Maintains two separate SSL sessions—client-to-FortiGate, and FortiGate-to-server
  - FortiGate encrypts and decrypts packets using its own keys
  - FortiGate can inspect the traffic

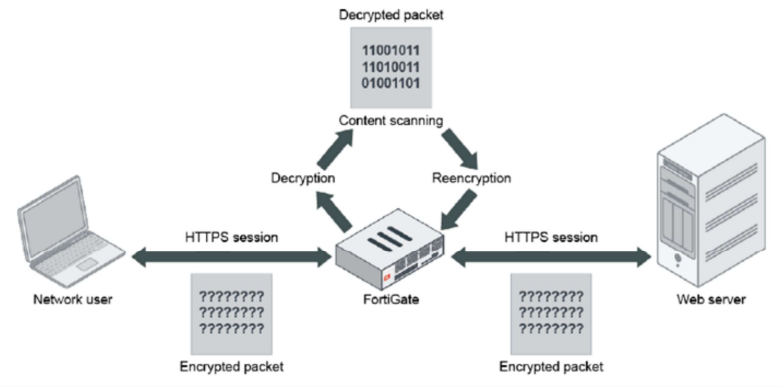


FORTINET Training Institute

You can configure full SSL inspection to inspect all of the packet contents, including the payload. FortiGate performs this inspection by proxying the SSL connection. Two SSL sessions are established—client-to-FortiGate and FortiGate-to-server. The two established sessions allow FortiGate to encrypt and decrypt packets using its own keys, which allows FortiGate to fully inspect all data inside the encrypted packets.

Full SSL Inspection

- Protect from attacks that use commonly used SSL-encrypted protocols
  - HTTPS
  - SMTPS
  - POP3S
  - IMAPS
  - FTPS
- FortiGate impersonates the recipient of the originating SSL session
  - Impersonates – decrypts
  - Inspects – blocks threats
  - Re-encrypts and sends to real recipient



FORTINET Training Institute

When you use deep inspection, FortiGate impersonates the recipient of the originating SSL session, and then decrypts and inspects the content to find threats and block them. It then re-encrypts the content and sends it to the real recipient. Deep inspection protects from attacks that use HTTPS and other commonly used SSL-encrypted protocols, such as SMTPS, POP3S, IMAPS, and FTPS.

Inbound or Outbound SSL/SSH Inspection

- SSL/SSH inspection for outbound traffic
  - Protecting internal users
  - Multiple clients connecting to multiple servers
    - External web servers
    - External mail servers
    - External FTPS servers

* → arrow direction represents traffic initialization

- SSL/SSH inspection for inbound traffic
  - Protecting a single company server
    - HTTPS server
    - Mail server
    - FTPS server
  - FortiGate use a server certificate
  - FortiGate as proxy server

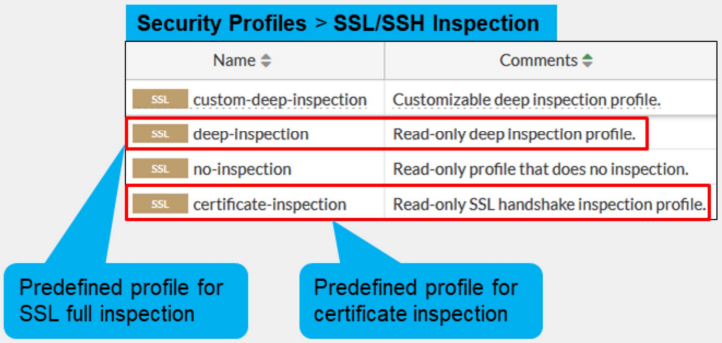
FORTINET Training Institute

FortiGate can proceed to SSL/SSH inspection for inbound traffic. Usually, this is the traffic initiated by local users bound for web servers on the internet. FortiGate protects users from traffic received from outside servers.

Conversely, you can use FortiGate to protect the company servers. Typically, you will protect the company web server from the outside world. For this purpose, FortiGate acts as a proxy server and presents the server certificate to internet users.

SSL Inspection Profile Configuration

- Ready-to-use profiles for inspection of outbound encrypted sessions
  - SSL certificate inspection
  - SSL full inspection
- Customizable profile
  - Outbound deep inspection with options
- User-defined profile
  - Inbound traffic
  - Outbound traffic

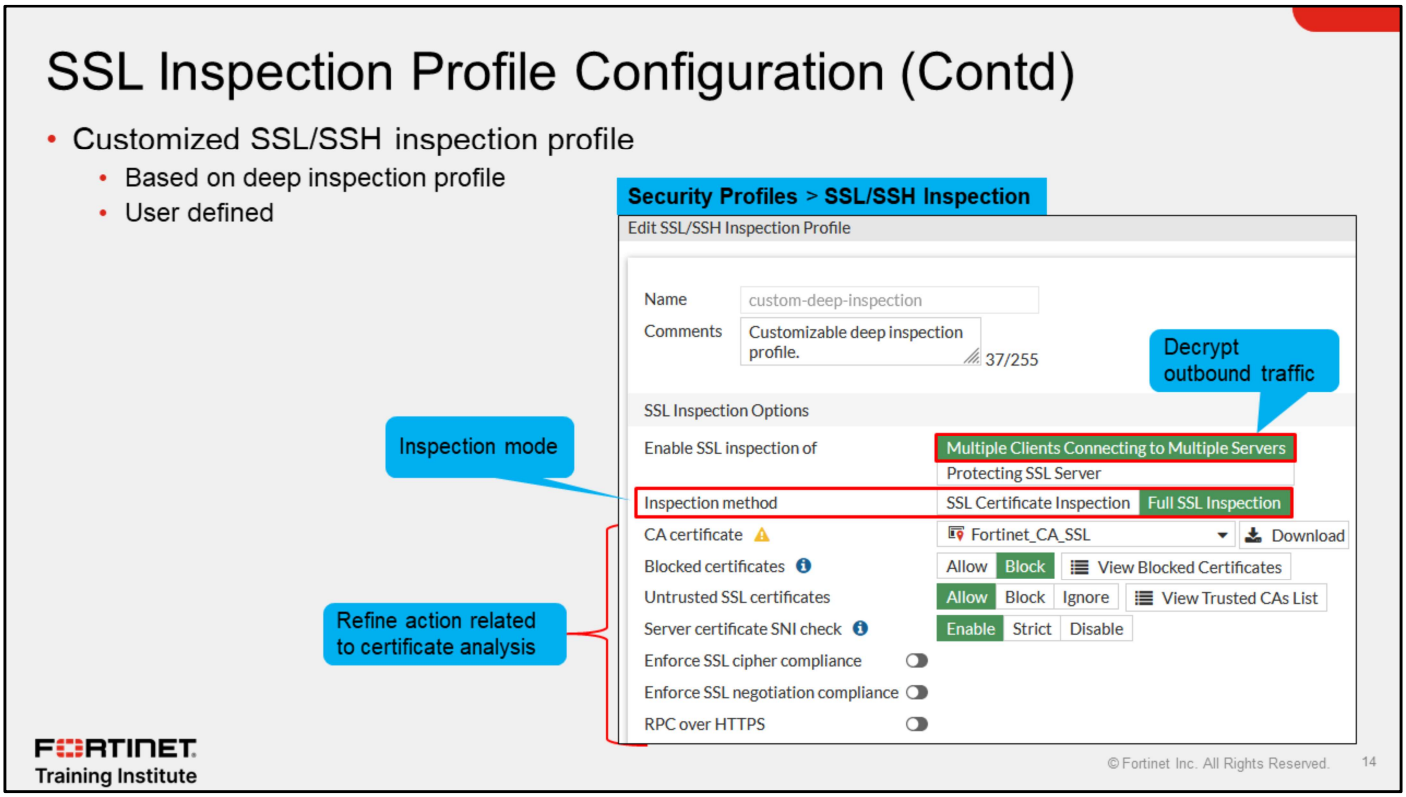


FORTINET Training Institute

On FortiGate, you can select the inspection mode applied at the firewall policy level. Three predefined SSL/SSH inspection profiles are available and correspond to the most common use cases.

The profile applied by default when you create a new firewall policy is the self-explanatory **no-inspection** profile. Other predefined profiles available are **certificate-inspection**, and **deep-inspection**, which applies full SSL inspection to the outbound traffic.

If you define an inspection profile for inbound traffic, or use some specific options for an outbound inspection profile, you can adjust the **custom-deep-inspection** profile or create your own profile.

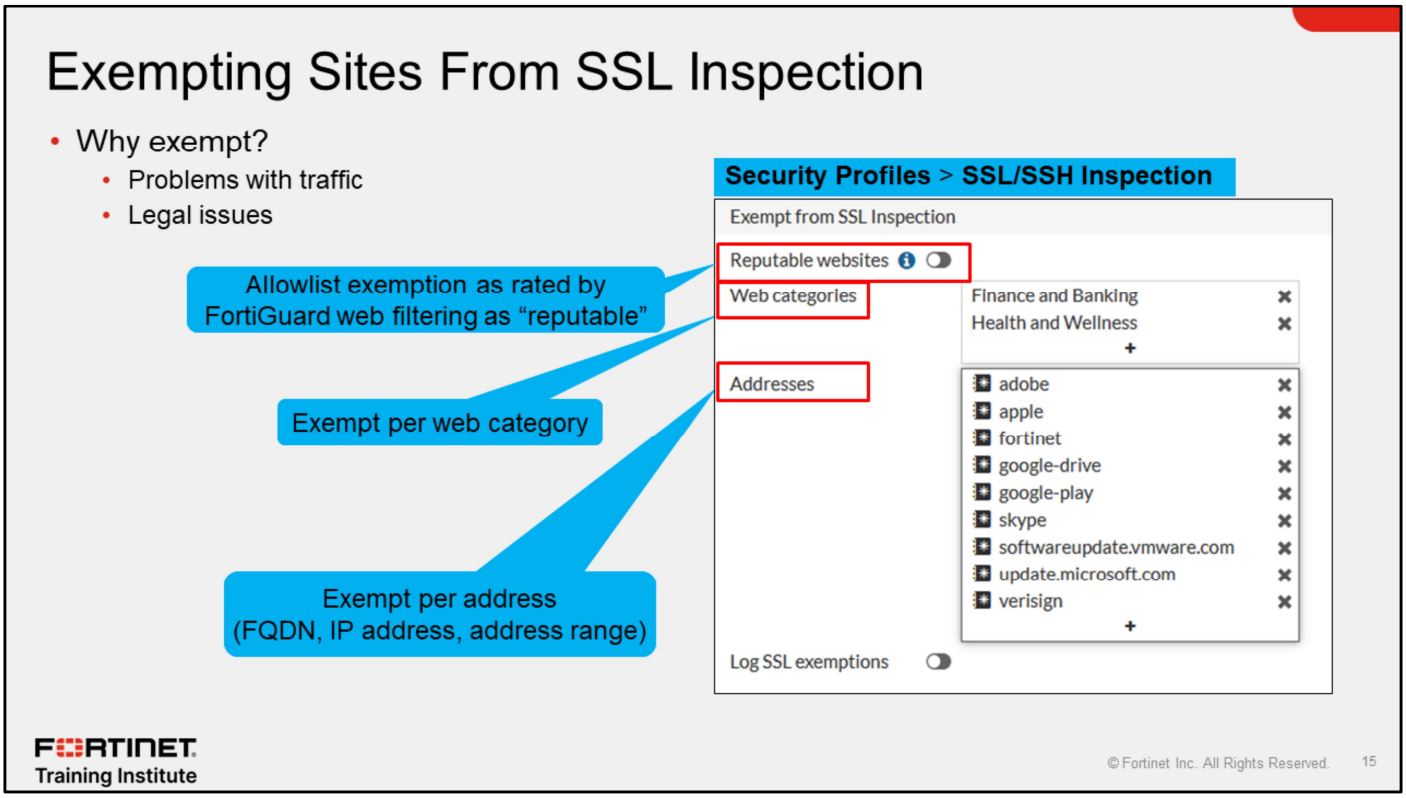


The predefined **certificate-inspection** and **deep-inspection** profiles are read-only. If you want to adjust the profile parameters, you can use the predefined **custom-deep-inspection** profile, or create a new, user-defined profile.

When you define a custom SSL/SSH profile, you can enable SSL inspection for output traffic with the parameter **Multiple Clients Connecting to Multiple Servers**, or for inbound traffic with the parameter **Protecting SSL Server**.

You can select the CA certificate used for traffic re-encryption between FortiGate and the destination. By default, FortiGate uses the preloaded `Fortinet_CA_SSL` certificate.

You can also specify the action that FortiGate takes according to some certificate parameters or status. For instance, you can define if you want to allow or block the traffic for untrusted or blocked certificates.



Within the full SSL inspection profile, you can also specify which SSL sites, if any, you want to exempt from SSL inspection. You may need to exempt traffic from SSL inspection if it is causing problems with traffic, or for legal reasons.

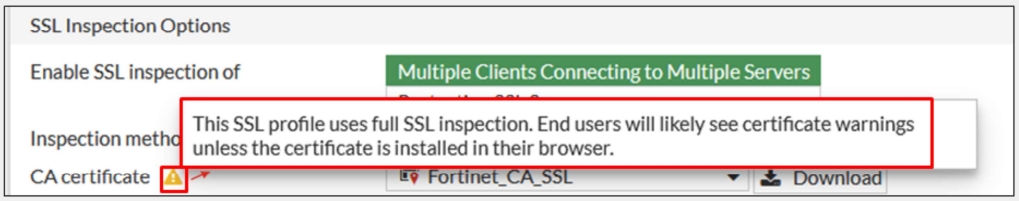
Performing SSL inspection on a site that is enabled with HSTS, for example, can cause problems with traffic. Remember, the only way for FortiGate to inspect encrypted traffic is to intercept the certificate coming from the server and generate a temporary one. After FortiGate presents the temporary SSL certificate, browsers that use HSTS refuse to proceed.

Laws protecting privacy might be another reason to bypass SSL inspection. For example, in some countries, it is illegal to inspect SSL bank-related traffic. Configuring an exemption for sites is simpler than setting up firewall policies for each individual bank. You can exempt sites based on their web category, such as **Finance and Banking**, or you can exempt them based on their address. Alternatively, you can enable **Reputable websites**, which excludes an allowlist of reputable domain names maintained by FortiGuard from full SSL inspection. This list is periodically updated and downloaded to FortiGate devices through FortiGuard.

The predefined **deep-inspection** and **custom-deep-inspection** profiles exclude some web categories—**Finance and Banking**, and **Heath and Wellness**—and some FQDN addresses such as google-play, skype, or verisign. When using the **custom-deep-inspection** profile, you can add or remove sites from this list.

FortiGate Self-Signed CA Certificates

- By default, FortiGate uses a self-signed encrypting SSL CA certificate
  - `Fortinet_CA_SSL`
  - Not listed with an approved CA, therefore, by default, not trusted



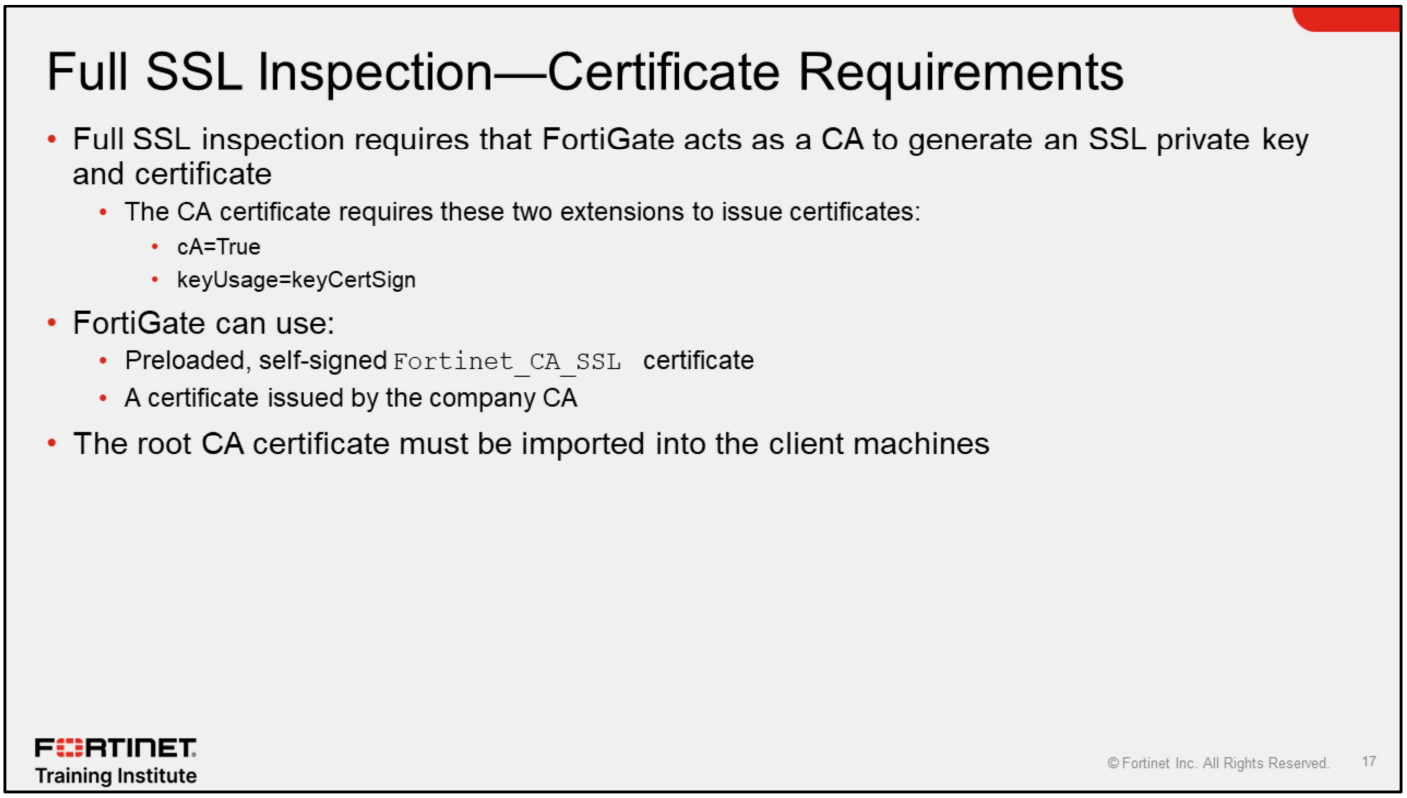
- To avoid warnings on user devices
  - Install CA certificate `Fortinet_CA_SSL` as trusted CA on user devices
  - Install a company CA certificate on FortiGate for SSL full inspection

FORTINET Training Institute

By default, FortiGate uses a self-signed CA certificate for the re-encryption required by the SSL full inspection. Because the corresponding CA is not prepopulated in client device certificate stores, users will likely see certificate warnings for traffic flows protected by the full SSL inspection.

To avoid the warning, you can install the `Fortinet_CA_SSL` certificate as trusted CA on the user devices. You can install it as part of the deployment process for all your company computers. Alternatively, you can install on FortiGate a CA certificate, used for traffic re-encryption, that is signed by your company CA. This certificate will already be recognized as valid by your company devices.

The certificate used to re-encrypt the traffic after the SSL full inspection must follow some specific requirements. You will discover them on the next slide.



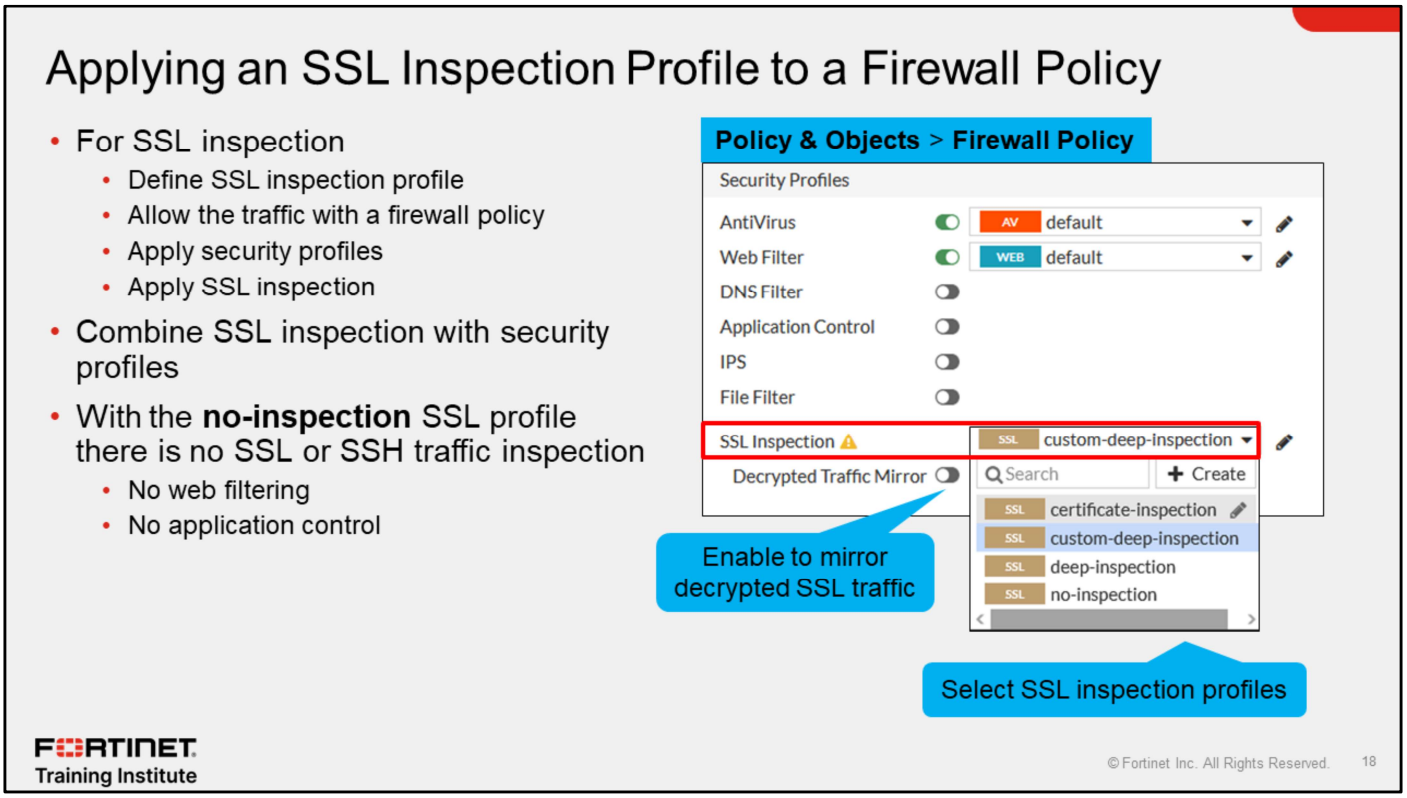
To perform full SSL inspection, FortiGate performs as a web proxy, and must act as a CA in order to re-encrypt the traffic. The FortiGate internal CA must generate an SSL private key and certificate each time it needs to re-encrypt a new traffic flow. The key pair and certificate are generated *immediately*, so the user connection with the web server is not delayed.

Although, from the user point of view, it appears as though the user browser is connected to the web server, the browser is in fact connected to FortiGate. To perform this proxy role, and generate a certificate that correspond to the server visited, the CA certificate must allow the generation of new certificates. To achieve this, it must have the following extensions: **cA** set to **True,** and the **keyUsage** extension set to **keyCertSign**.

The **cA=True** value identifies the certificate as a CA certificate. The **keyUsage=keyCertSign** value indicates that the certificate corresponding to the private key is permitted to sign certificates. For more information, see *RFC 5280 Section 4.2.1.9 Basic Constraints*.

All FortiGate devices come with the self-signed `Fortinet_CA_SSL` certificate that you can use for full SSL inspection. If your company has an internal CA, you can request the CA administrator to issue a certificate for your FortiGate device. The FortiGate device then acts as a subordinate CA.

If you use the `Fortinet_CA_SSL` certificate, or a certificate issued by your company CA, to trust FortiGate and accept re-encrypted SSL sessions without warning, you must import the root CA certificate used to your client devices.



To perform SSL inspection on traffic flowing through the FortiGate device, you must allow the traffic with a firewall policy and apply an SSL inspection profile to the policy. Note than an SSL inspection profile alone will not trigger a security inspection. You must combine it with other security profiles like **Antivirus**, **Web Filter**, **Application Control,** or **IPS**.

By default, firewall policies are set with the **no-inspection** SSL profile. Therefore, any encrypted traffic flows through uninspected. For instance, with the **no-inspection** profile, FortiGate cannot perform any web filtering for HTTPS traffic. To allow web filtering, DNS filtering, or application control for HTTPS traffic, you *must* select an SSL inspection profile with certificate inspection or a deep inspection enabled. For antivirus or IPS control you should use a deep-inspection profile.

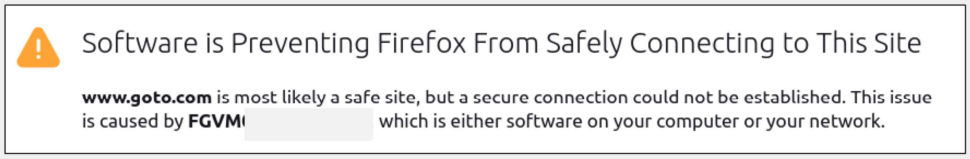
You can see a warning sign near the SSL inspection profile selection menu on the GUI. You will see this warning each time you select an SSL inspection profile with deep inspection. It is there to warn about the certificate warning that can appear on the user browser when traffic is allowed with this policy. If you hover over the warning sign you can read this message: “This SSL profile uses full SSL inspection. End users will likely see certificate warnings unless the certificate is installed in their browser.”

If you select a profile with full SSL inspection enabled, the option **Decrypted Traffic Mirror** appears. Enable this option if you want FortiGate to send a copy of the decrypted SSL traffic to an interface. It works only with flow-based inspection. When you enable **Decrypted Traffic Mirror**, FortiGate displays a window with the terms of use for this feature. The users must agree to the terms before they can use the feature.

You will apply an SSL profile to a firewall policy the same way for inbound or outbound traffic flow inspection. It is the SSL profile applied that specifies the certificate in use when the FortiGate device re-encrypts the traffic.

Certificate Warnings During Full SSL Inspection

- During full SSL inspection, browsers might display a warning because they do not trust the CA



- To enable a smooth user experience, and prevent certificate warnings, do one of the following:
  - Use the `Fortinet_CA_SSL` certificate
    - And import the FortiGate CA root certificate into all the browsers
  - Use an SSL certificate issued by a private CA
    - This CA may already be available in the device browsers
- This is not a FortiGate limitation, but a consequence of how SSL and digital certificates work

FORTINET Training Institute

When doing full SSL inspection using the FortiGate self-signed CA, your browser might display a certificate warning each time you connect to an HTTPS site. This is because the browser is receiving certificates signed by FortiGate, which is a CA it does not know and trust. This is not a limitation of FortiGate, but a consequence of how digital certificates are designed to work.

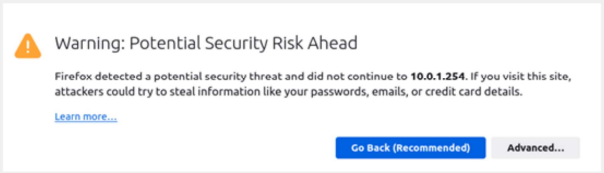
There are two ways to avoid those warnings:

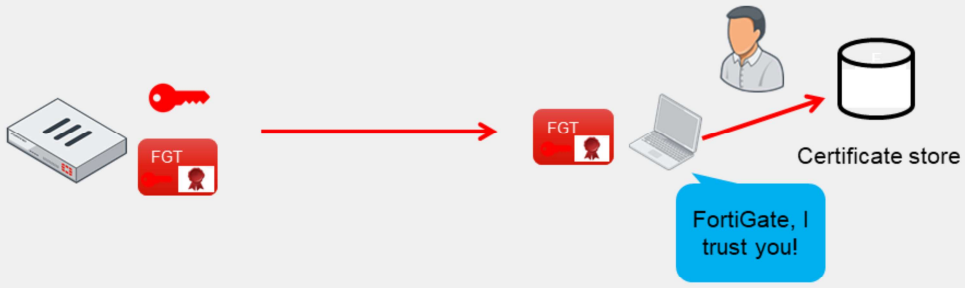
- The first option is to download the default FortiGate certificate for SSL proxy inspection and install it on all the workstations as a trusted root authority.
- The second option is to generate a new SSL proxy certificate from a private CA. In this case, the private CA certificate must still be imported into all the browsers.

If you use an SSL certificate signed by a subordinate CA, you must ensure that the entire chain of certificates—from the SSL certificate to the root CA certificate—is installed on FortiGate. Verify also that the root CA is installed on all client browsers. This is required for trust purposes. Because FortiGate sends the chain of certificates to the browser during the SSL handshake, you do not have to import the intermediate CA certificates into the browsers.

Certificate Warnings on the FortiGate GUI

- By default, FortiGate uses a self-signed SSL certificate
  - Not listed with an approved CA, therefore, by default, not trusted
  - Used for HTTPS GUI access
- Available options to avoid those warnings:
  - Accept the warning at first connection
  - Use the `Fortinet_GUI_Server` certificate and import the `Fortinet_CA_SSL` certificate
  - Use a certificate signed by a recognized CA





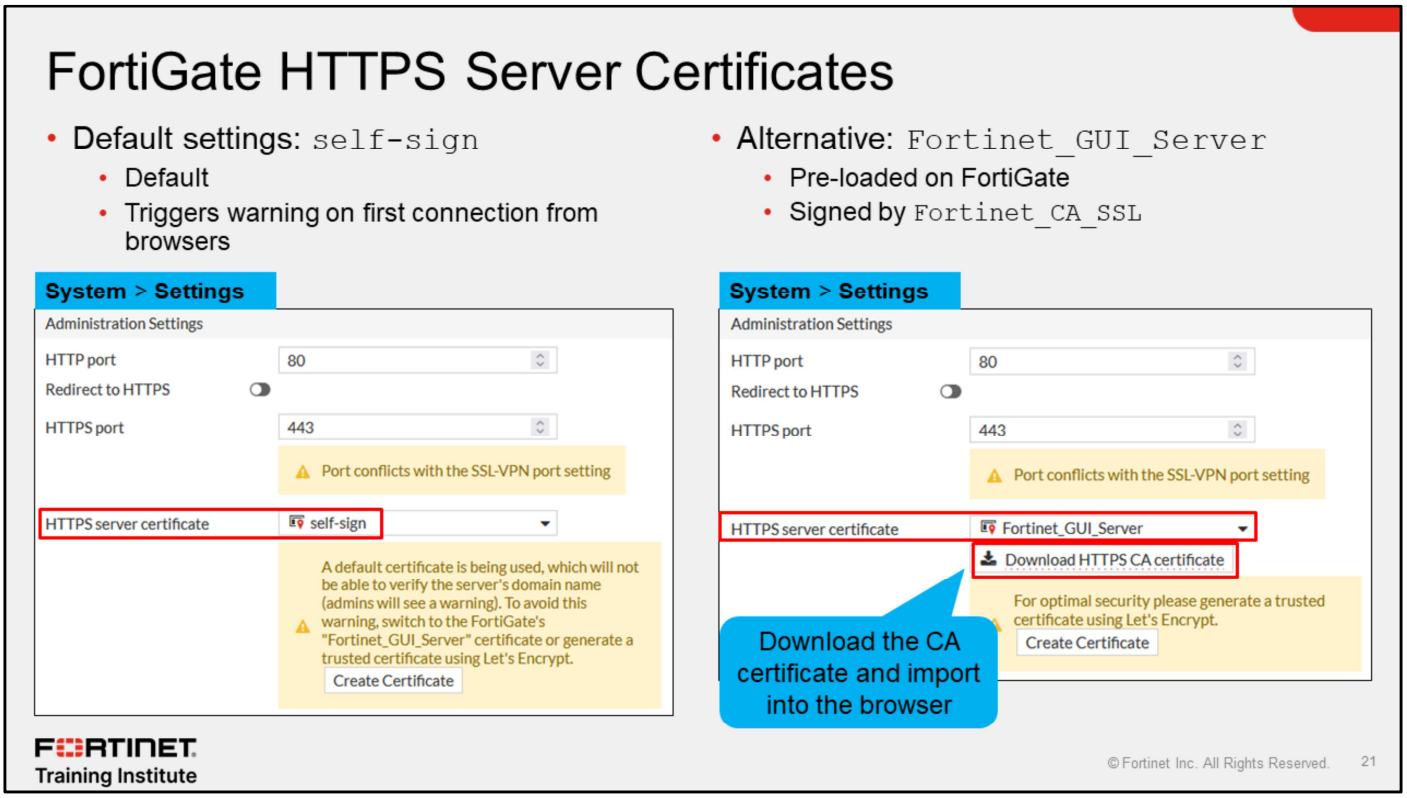
FORTINET Training Institute

By default, FortiGate uses a self-signed certificate to authenticate itself to HTTPS clients. Because the corresponding CA certificate is not prepopulated in the certificate stores of client devices, the first HTTPS connection to a FortiGate device triggers a security warning.

If you trust the FortiGate device and want to keep the self-signed certificate to establish SSL sessions, you can accept the warning and establish the connection. When you accept the warning, your browser imports the FortiGate self-signed certificate into its certificate store. So, the next time you connect to this FortiGate device, your browser already trusts the certificate presented.

Alternatively, you can configure FortiGate to use the `Fortinet_GUI_Server` certificate and add the FortiGate self-signed CA certificate—`Fortinet_CA_SSL`—to the local certificate store of any computer that needs to connect to the FortiGate device. For subsequent connections to the FortiGate GUI interface, those devices trust the certificate and allow connections without warning.

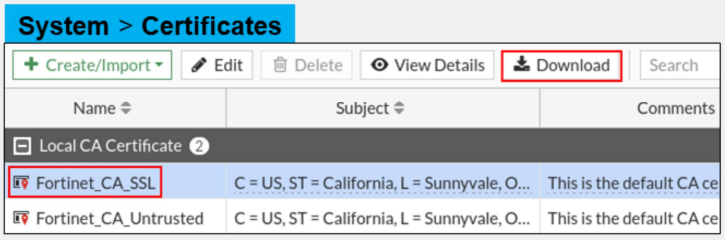
Another option for companies who manage their own CA is to generate a certificate for each of your FortiGate devices and use them to secure HTTPS connections.



You can select the certificate that FortiGate presents for HTTPS GUI access from the **Settings** menu. By default, FortiGate uses the `self-sign` certificate, which is not recognized as a trusted certificate by the browsers. Alternatively, you can select the `Fortinet_GUI_Server` certificate, which is signed by the `Fortinet_CA_SSL`. With this certificate, to avoid the browser warning on HTTPS access to the FortiGate GUI, you must import the `Fortinet_CA_SSL` certificate into your management devices.

Download Private CA Certificates From FortiGate

- Download `Fortinet_CA_SSL` private CA certificate



- Generate a file `Fortinet_CA_SSL.cer`
- Transfer to any computer that requires it

FORTINET Training Institute

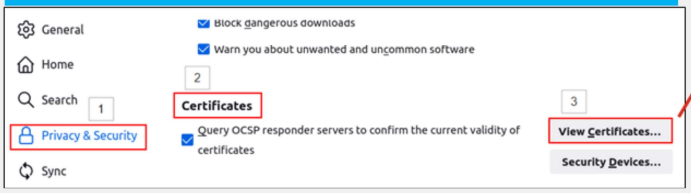
Before you can import the default CA certificate—`Fortinet_CA_SSL`— into the user devices, you must download it from FortiGate.

You can get it from the FortiGate certificate store available under the **System** menu. Upon download, FortiGate generates a `.cer` file that you can import into any device as required.

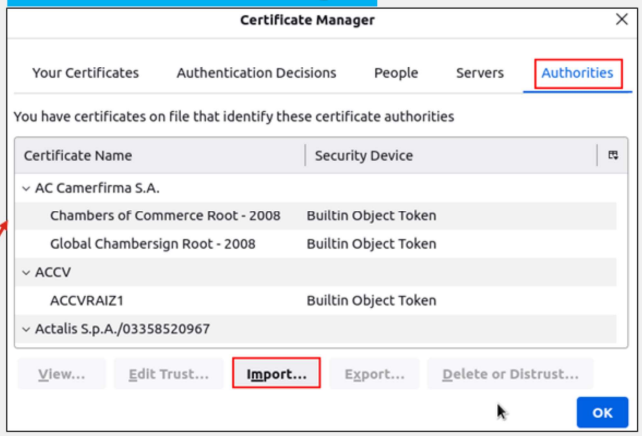
Import Private CA Certificates Into Endpoints

- Import `Fortinet_CA_SSL` private CA certificate into user device
  - Exact process depends on the operating system
  - Example for Linux and Firefox
    - Open the browser setting menu
    - Open the certificate store
    - Import the certificate as a CA authority

Firefox: **Settings** > **Privacy & Security** > **Certificates**



Firefox: **Certificate Manager**



FORTINET Training Institute

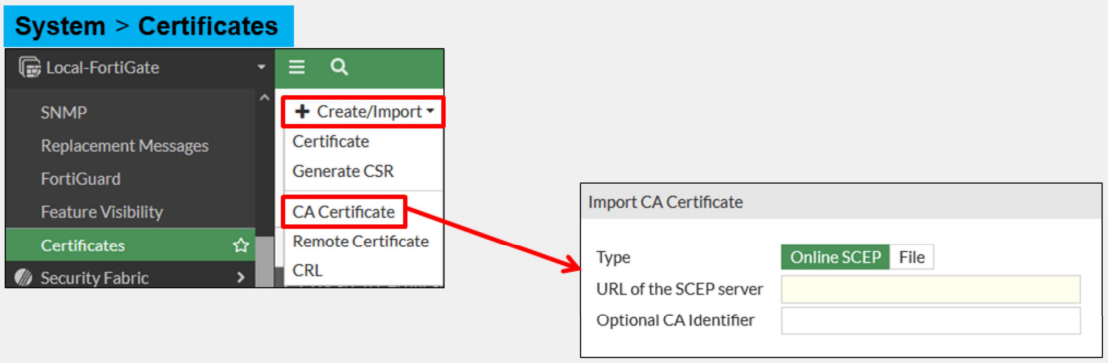
After you download the CA certificate from FortiGate, you can import it into any web browser or operating system. Not all browsers use the same certificate repository. For example, Firefox uses its own repository, while Internet Explorer and Chrome store certificates in a system-wide repository. In order to prevent certificate warnings, you must import the SSL certificate as a trusted root CA.

When you import the certificate, make sure that you save it to the certificate store for root authorities.

The example on this slide shows the menu you use to import a certificate into the Firefox browser.

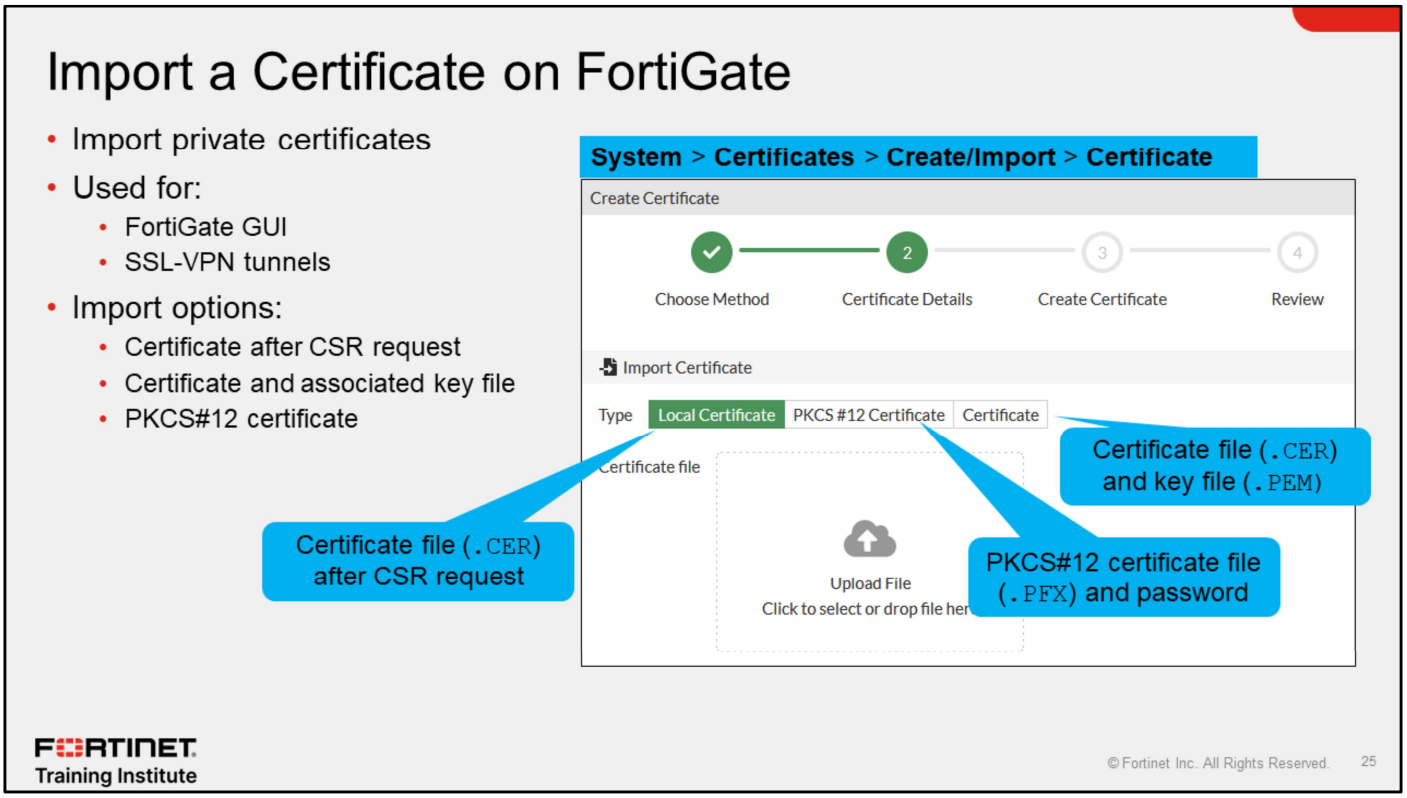
Import a CA Certificate on FortiGate

- Import company-owned private CA or CA signed by a certificate authority



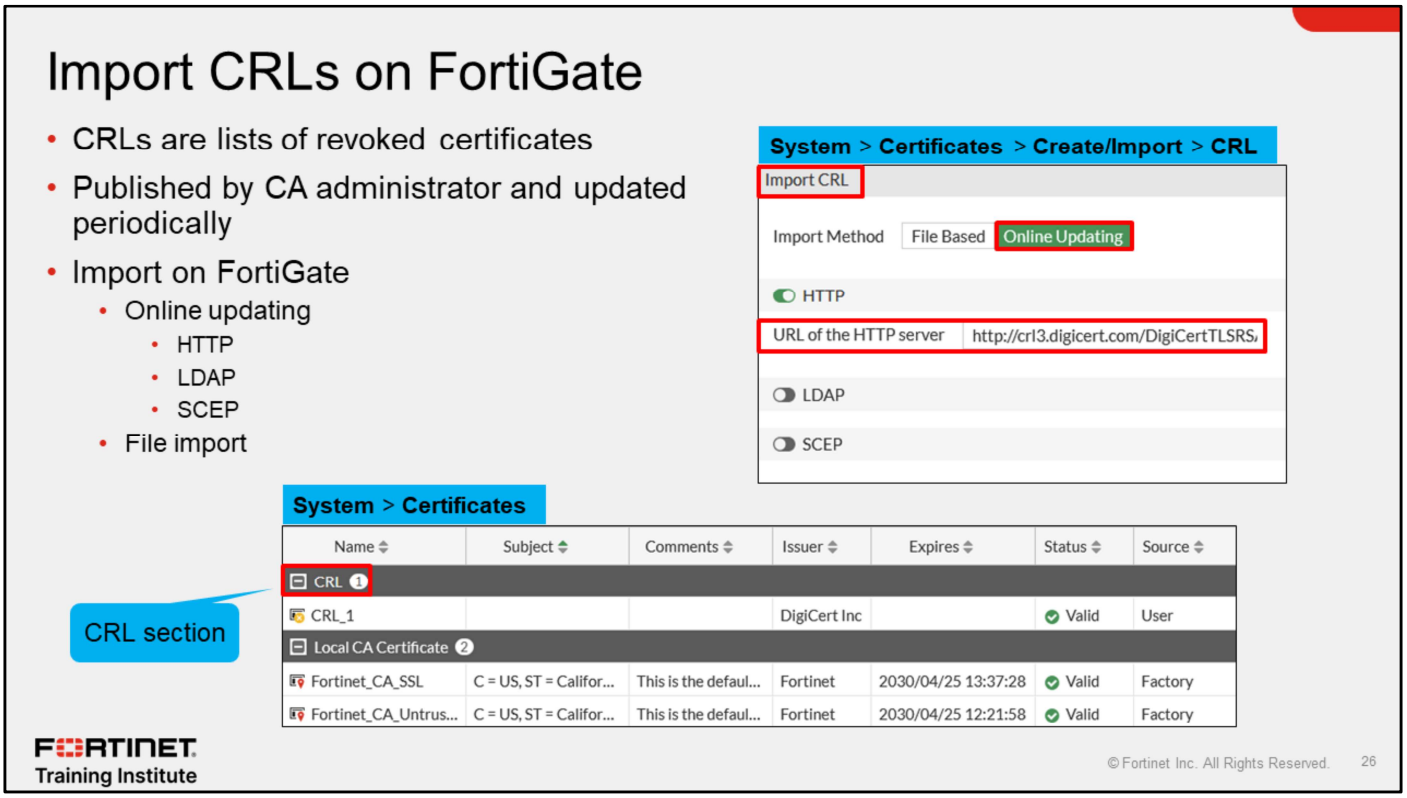
FORTINET Training Institute

If your company has a private signing CA or a signing CA signed by a certificate authority, you can import the corresponding certificate onto the FortiGate device as shown on this slide. Note that you can import the certificate by connecting to the SCEP server or as a file. SCEP stands for Simple Certificate Enrollment Protocol, and it is a popular and widely available certificate enrollment protocol.



If your company manages its own certificate authority, you can generate certificates for the FortiGate GUI or SSL access. You can also generate certificates that you will use for SSL-VPN tunnels.

FortiGate offers three options to import private certificates. You can first generate a certificate signing request (CSR) and submit it to the CA for certificate generation. With this process, the key file is automatically generated and stored on FortiGate when it generate the CSR. Later, you import only the certificate file (`.CER`) provided by the CA. Another option is to import the certificate file and the associated key into the FortiGate certificate store. Alternatively, you can load a PKCS#12 certificate file, which is identified as a `.PFX` file. It contains the certificate and associated private key.



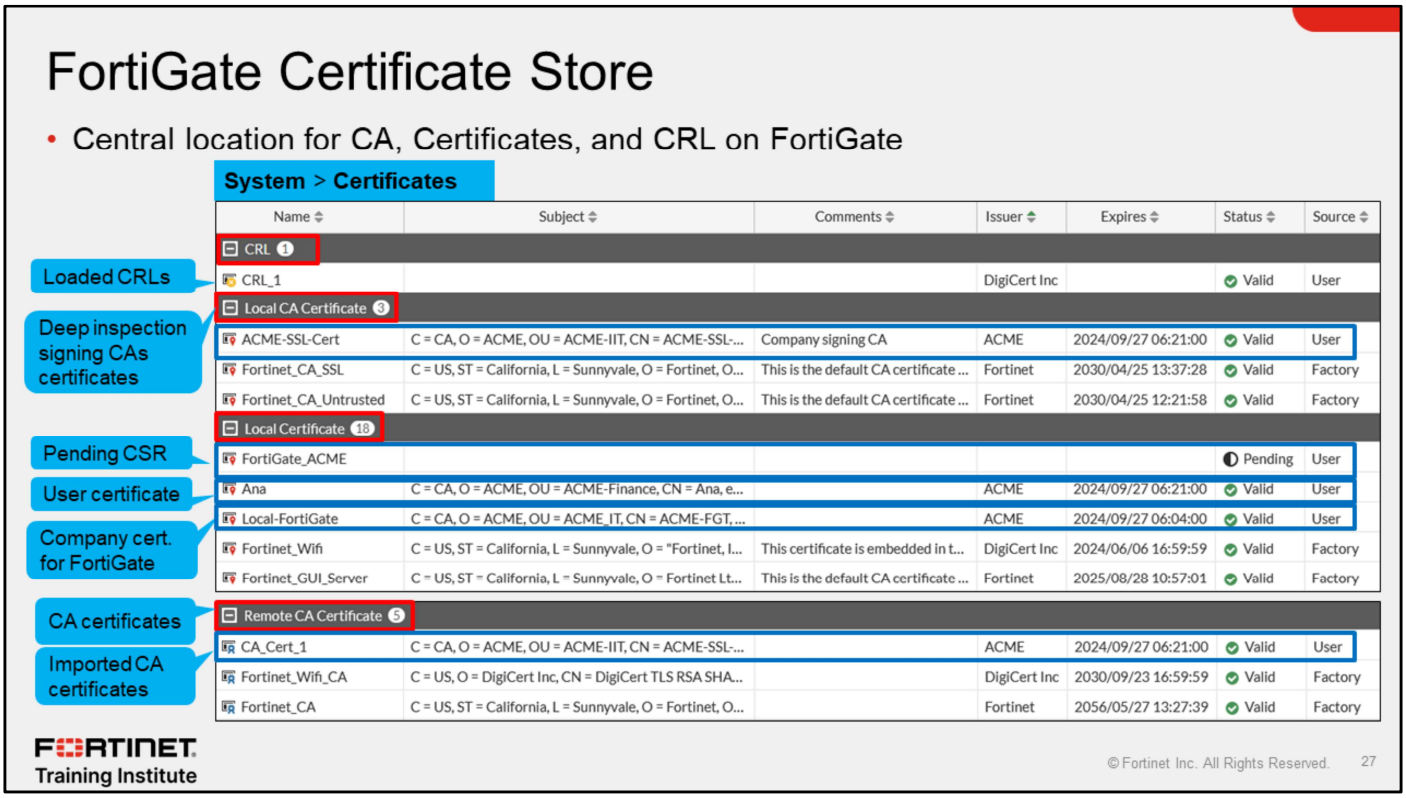
Because it is not possible to recall a certificate, the certificate revocation list (CRL) details certificates signed by valid CAs that should no longer be trusted. Certificates may be revoked for many reasons, such as if the certificate was issued erroneously, or if the private key of a valid certificate has been compromised.

CA administrators publish CRLs and periodically update them. You can load CRLs into the FortiGate device as files provided by CA administrators, or direct FortiGate to connect to the CRL repositories and load the corresponding list.

The recommended method to keep the list of revoked certificates up to date is to load them through one of the following available protocols: HTTP, LDAP, or SCEP. Alternatively, you can load the CRL list into the FortiGate certificate store by importing CRL files.

You can get the CRL distribution point associated with a certificate by editing it and navigating to the CRL endpoints information part.

Note that the CRL section on the FortiGate GUI **Certificates** menu is visible only after you have loaded at least one CRL.



The central location to review the certificates imported into a FortiGate device is the certificate list available in the **Certificates** section of the **System** menu.

In this table you can view:

- The **CRL** section, which contains all loaded CRLs.
- The **Local CA Certificate** section, which contains the FortiGate signing CA certificate. By default, it contains the `Fortinet_CA_SSL` and `Fortinet_CA_untrusted` certificates. If you import a signing CA certificate from your company, it will appear in this section.
- The **Local Certificate** section, which contains device and user certificates. In the example shown on this slide you can see a user certificate, `Ana`, and a device certificate, `Local-FortiGate`. For both, the issuer is ACME, which is the company private CA in this example.
- The **Remote CA** certificate section , which is section where FortiGate displays all imported CA certificates that are not signing CA certificates.

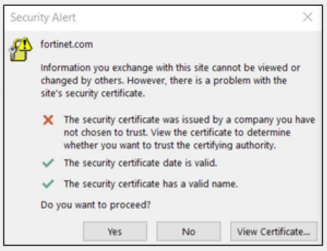
Note that:

- The **CRL** section is visible only after you have loaded at least one CRL.
- FortiGate displays CRLs only if the corresponding CA certificate is imported into the certificate store.
- FortiGate shows the certificate signing requests (CSR) in the **Local Certificate** section with the status **Pending**.
- The **Source** column indicates the origin of the certificate, either **Factory** for certificates always present or **User** for certificates imported by an administrator user.

Applications and SSL Inspection

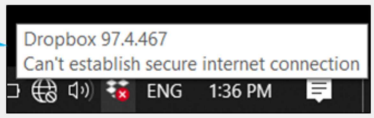
- Any SSL application might be impacted by SSL inspection (not just the browser)
  - The solution depends on the application security design
  - Consider other SSL-based protocols such as FTPS, SMTPS, and STARTTLS (not just HTTPS)
- Microsoft Outlook 365 for Windows error after enabling full SSL inspection:

Solution: Import the CA certificate into the Windows certificate store (FortiGate keeps inspecting SSL traffic)



- Dropbox for Windows error after enabling full SSL inspection:

Solution: Exempt Dropbox domains from SSL inspection (FortiGate no longer inspects SSL traffic)



FORTINET Training Institute

More and more applications are using SSL to securely exchange data over the internet. While most of the content in this lesson centers around the operation and impact of SSL inspection on browsers, the same applies to other applications using SSL as well. After all, the browser is just another application using SSL on your device.

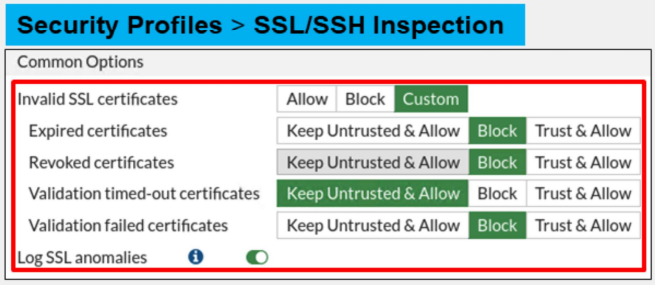
For this reason, when you enable SSL inspection on FortiGate, you need to consider the potential impact on your SSL-based applications. For example, Microsoft Outlook 365 for Windows reports a certificate error when you enable full SSL inspection because the CA certificate used by FortiGate is not trusted. To solve this issue, you can import the CA certificate into your Windows certificate store as a trusted root certificate authority. Because Microsoft Outlook 365 trusts the certificates in the Windows certificate store, then the application won't report the certificate error anymore. Another option is to exempt your Microsoft Exchange server addresses from SSL inspection. While this prevents the certificate error, you are no longer performing SSL inspection on email traffic.

There are other applications that have built-in extra security checks that prevent MITM attacks, such as HSTS. For example, Dropbox uses certificate pinning to ensure that no SSL inspection is possible on user traffic. As a result, when you enable full SSL inspection on FortiGate, your Dropbox client stops working and reports that it can't establish a secure connection. In the case of Dropbox, the only way to solve the connection error is by exempting the domains Dropbox connects to from SSL inspection.

In addition, remember that SSL is leveraged by different protocols, not just HTTP. For example, there are other SSL-based protocols such as FTPS, POP3S, SMTPS, STARTTLS, LDAPS, and SIP TLS. If you have an application using any of these SSL-based protocols, and you have turned on SSL inspection along with a security profile that inspects those protocols, then the applications may report an SSL or certificate error. The solution depends on the security measures adopted by the application.

Invalid Certificates

- FortiGate can detect invalid certificates for a variety of reasons
  - Invalid certificates produce security warnings due to problems with the certificate details
- FortiGate can **Keep Untrusted & Allow**, **Block**, or **Trust & Allow** invalid certificates
- Selecting **Custom** allows the user to select the action for each reason



FORTINET Training Institute

FortiGate can detect certificates that are invalid for the following reasons:

- Expired: The certificate is expired.
- Revoked: The certificate has been revoked based on CRL or OCSP information.
- Validation timeout: The certificate could not be validated because of a communication timeout.
- Validation failed: FortiGate could not validate the certificate, or it is not yet valid.

When a certificate fails for any of the reasons above, you can configure any of the following actions:

- **Keep untrusted & Allow**: FortiGate allows the website and lets the browser decide the action to take. FortiGate takes the certificate as *untrusted*.
- **Block**: FortiGate blocks the content of the site.
- **Trust & Allow**: FortiGate allows the website and takes the certificate as *trusted*.

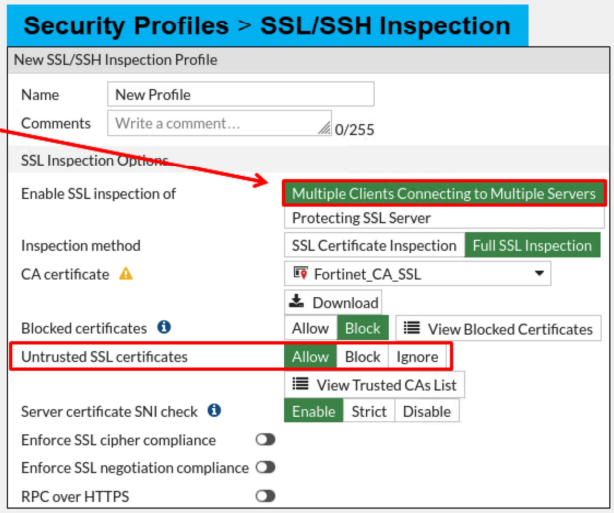
The certificate check feature can be broken down into two major checks, which are done in parallel:

- FortiGate checks if the certificate is invalid because of the four reasons described on this slide.
- FortiGate performs certificate chain validation based on the CA certificates installed locally and the certificates presented by the SSL server.

Based on the actions configured, and the check results, FortiGate presents the certificate as either trusted (signed by `Fortinet_CA_SSL`) or untrusted (signed by `Fortinet_CA_Untrusted`), and either allows the content or blocks it. You can also track certificate anomalies by enabling the **Log SSL anomalies** option.

Untrusted SSL Certificates Setting

- Allow, block, or ignore untrusted certificates (only available if **Multiple Clients Connecting to Multiple Servers** is selected)
  - **Allow**: sends the browser an untrusted temporary certificate when the server certificate is untrusted
  - **Block**: blocks the connection when an untrusted server certificate is detected
  - **Ignore**: uses a trusted FortiGate certificate to replace the server certificate always, even when the server certificate is untrusted



FORTINET Training Institute

The browser presents a certificate warning when you attempt to access an HTTPS site that uses an untrusted certificate. Untrusted certificates include self-signed SSL certificates, unless the certificate is imported into the browser-trusted certificate store. FortiGate has its own configuration setting on the **SSL/SSH Inspection** profile, which includes options to **Allow**, **Block**, or **Ignore** untrusted SSL certificates.

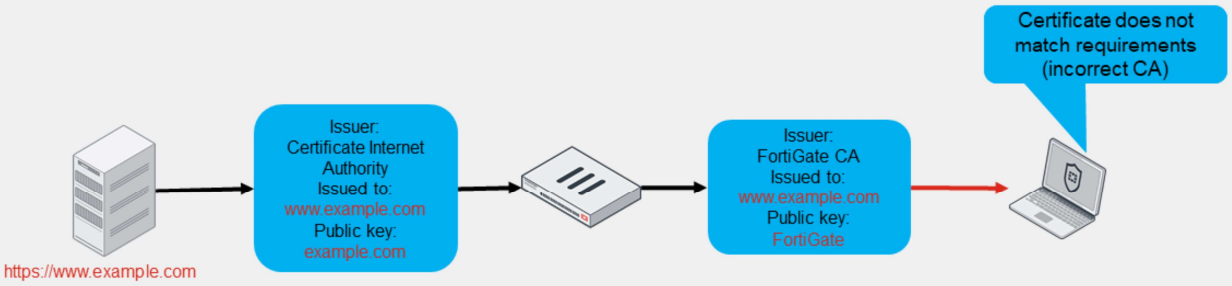
When you set the **Untrusted SSL certificates** setting to **Allow**, and FortiGate detects an untrusted SSL certificate, FortiGate generates a temporary certificate signed by the built-in `Fortinet_CA_Untrusted` certificate. FortiGate then sends the temporary certificate to the browser, which presents a warning to the user indicating that the site is untrusted. If FortiGate receives a trusted SSL certificate, then it generates a temporary certificate signed by the built-in `Fortinet_CA_SSL` certificate and sends it to the browser. If the browser trusts the `Fortinet_CA_SSL` certificate, the browser completes the SSL handshake. Otherwise, the browser also presents a warning message informing the user that the site is untrusted. In other words, for this function to work as intended, you must import the `Fortinet_CA_SSL` certificate into the trusted root CA certificate store of your browser. The `Fortinet_CA_Untrusted` certificate must *not* be imported.

When the setting is set to **Block**, and FortiGate receives an untrusted SSL certificate, FortiGate blocks the connection outright, and the user cannot proceed.

When the setting is set to **Ignore**, FortiGate sends the browser a temporary certificate signed by the `Fortinet_CA_SSL` certificate, regardless of the SSL certificate status—trusted or untrusted. FortiGate then proceeds to establish SSL sessions.

Full SSL Inspection and HSTS

- Some clients have specific requirements for SSL
  - HSTS: HTTPS Strict Transport Security
    - Example: Chrome requires a Google certificate when accessing any Google site
- HSTS common error message
  - "Privacy error: Your connection is not private" (NET::ERR_CERT_AUTHORITY_INVALID)

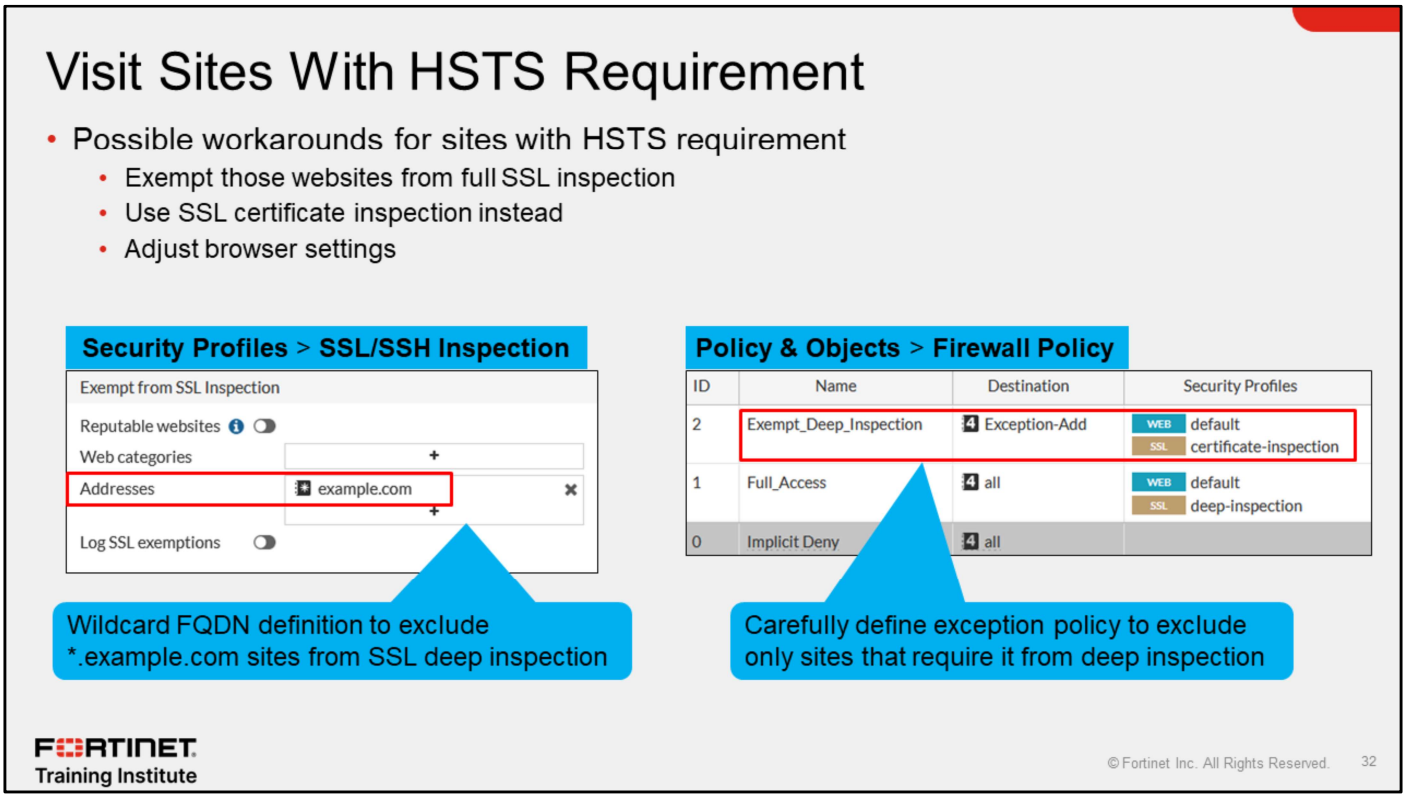


FORTINET Training Institute

Replacing the certificate for the traffic can cause problems. Some software and servers have specific limitations on the certificates that are allowed to be used.

HSTS is a security features designed to detect man-in-the-middle SSL attacks by making sure that any certificate presented when accessing a server resource is signed by a specific CA.

If the browser detects any other CA, it simply refuses to continue the SSL handshake, and prevents access to the website. If you are using a Chrome browser, for such sites, you will get the privacy error message "Your connection is not private" this slide shows.



When replacing the certificate for the traffic causes problems and prevents users from accessing some websites, the solutions available are limited. You can select one of the following workarounds according to the level at which you can act.

At the FortiGate level, you can exempt the affected websites from full SSL inspection and use certificate inspection instead. If you can act at the browser level, you can disable HSTS validation per website or globally (refer to the browser manual for the process).

If you want to use certificate inspection instead of deep inspection only for a few sites, you must be careful when defining the policy. It must be restrictive enough to match exclusively the sites that you want to allow, and which do not support deep inspection. Otherwise, you might get sites allowed to pass through with only certificate inspection instead of deep inspection.

Knowledge Check

1. Which attribute or extension identifies the owner of a certificate?

✓ A. The subject name in the certificate
B. The unique serial number in the certificate

2. Which configuration requires FortiGate to act as a CA for full SSL inspection?

✓ A. Multiple clients connecting to multiple servers
B. Protecting the SSL server

3. Which inspection mode can protect your LAN devices from encrypted malware?

A. Certificate inspection
✓ B. Deep inspection

Review

- ✓ Describe certificate inspection and full SSL inspection
- ✓ Configure FortiGate for full SSL inspection
- ✓ Identify obstacles to implementing full SSL inspection and possible remedies

This slide shows the objectives that you covered in this lesson.

By mastering the objectives covered in this lesson, you learned how FortiGate uses certificates, and how to manage and work with certificates in your network.