Realtime
publishers

The Shortcut Guide™ *To*

PCI Compliance and How SSL Certificates Fit

sponsored by
thawte™

Dan Sullivan

Copyright Statement

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers its web site sponsors. In no event shall Realtime Publishers or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com.

Chapter 2: Overview of SSL Certificates

SSL certificates are an important element of the security infrastructure that protects systems and communications. In that role, they also enable customers to trust businesses that customers might otherwise be unfamiliar with. What is it about SSL certificates that enable these properties? To answer this question, we must understand the components of an SSL certificate and how they are used for authentication and encryption. We also need to understand different uses of SSL certificates and how they enable the formation of trust. This chapter is organized into five sections that will address these issues:

- Components of an SSL certificate
- Authenticating servers with SSL certificates
- Encryption and SSL
- Different uses of SSL certificates
- Trust and SSL security

We start with the basic building blocks of SSL certificates.

Components of an SSL Certificate

An SSL certificate is a file. At least that is the basic implementation of an SSL certificate. The file is well structured and contains a variety of different types of information all of which are needed to meet the basic functional requirements of an SSL certificate. SSL certificates use a format outlined in the X.509 standard, and the main components include:

- Version and serial number
- Signature algorithm
- Issuer
- Validity dates
- Subject
- Subject's public key
- Digital signature of the issuing Certificate Authority (CA)

The X.509 standard was defined by the International Telecommunication Union (ITU) to provide a format for public key certificates, such as SSL certificates, as well as to define ancillary functions, such as certificate revocation lists. The version attribute specifies the version of the X.509 standard that was used to generate the certificate. The serial number is a unique number assigned by the issuer to the certificate. The combination of the issuer and the serial number is unique across all certificates.

The algorithm attribute is an identifier that indicates the algorithm used by the issuing party to digitally sign the certificate. The issuer attribute describes the CA that issued the certificate. The issuer attribute actually includes a number of features about an issuer, such as the name of the issuer, the country where it is located, and the organizational unit within the company that issued the certificate.

> **Note**
>
> Anyone or any organization can issue a certificate. All that is needed is some freely available software to generate the X.509 certificate file. In practice, however, most certificates used in online commerce are created by trusted organizations in the business of providing SSL certificates. Although we can all create certificates, far fewer people would actually trust our certificates than those generated by well-known certificate providers.

Validity dates are important attributes of an SSL certificate because these dates indicate the time period which a certificate is valid. Like credit cards and debit cards, SSL certificates have expiration dates. This feature ensures that a certificate cannot be used indefinitely without the holder of the certificate demonstrating that it still has legitimate use of the certificate. For example, a retailer that acquires an SSL certificate may go out of business. Once the validity date passes, the certificate will no longer be valid and the business would no longer exist to get another certificate from a legitimate certificate provider.

The subject attribute identifies the entity associated with the certificate. In some cases, this attribute is a server or a domain, but it could be a business or other organization. SSL certificates are sometimes categorized by the type of entity in the subject attribute. When an SSL certificate is issued for a server, the fully qualified domain name is specified, such as server1.example.com. If someone were to place this certificate on another server, a Web browser would be able to detect the mismatch between the subject name and the name of the server hosting the certificate.

The keys in an SSL certificate are cryptographic keys. Keys are binary strings. The types of keys used in SSL certificates are public key cryptography keys. These keys have a particularly useful property: They are created in pairs and one key is used to encrypt ("lock") data and the other is used to decrypt ("unlock") the encrypted data. As Figure 2.1 depicts, data, such as a message, is encrypted using one key from the key pair but is decrypted using the other. The key used to encrypt a message is known as the public key and the other is called the private key. Public keys, as the names implies, are freely shared.

Anyone who wants to send an encrypted message to the subject of the SSL certificate can encrypt a message using the public key that is included in SSL the certificate. The private key is kept secret by the subject, so only the subject can decrypt a message sent using its public key.

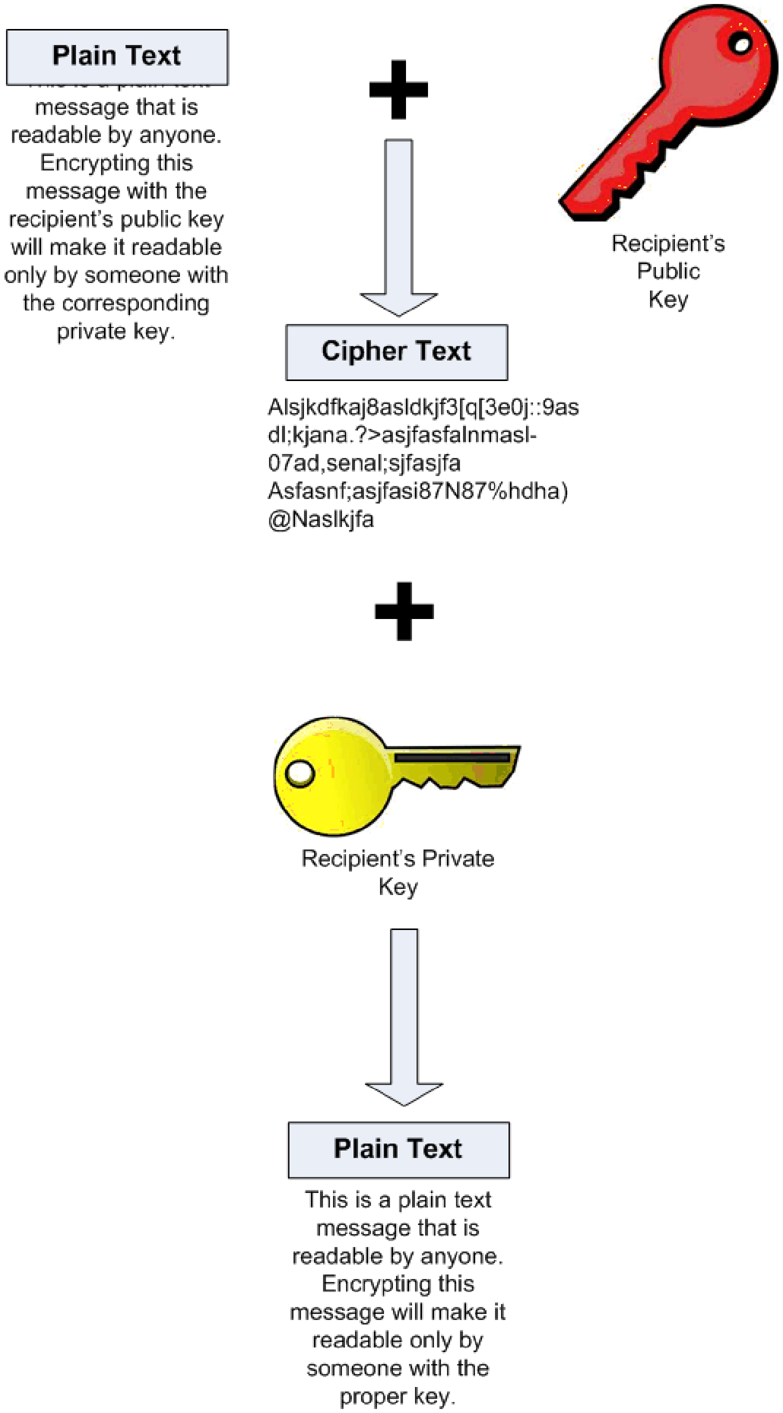

Plain Text
message that is readable by anyone. Encrypting this message with the recipient's public key will make it readable only by someone with the corresponding private key.
Recipient's Public Key
Cipher Text
Alsjkdfkaj8asldkjf3[q[3e0j::9as dl;kjana.?>asjfasfalnmasl-07ad,senal;sjfasjfa Asfasnf;asjfasi87N87%hdha) @Naslkjfa
Recipient's Private Key
Plain Text
This is a plain text message that is readable by anyone. Encrypting this message will make it readable only by someone with the proper key.

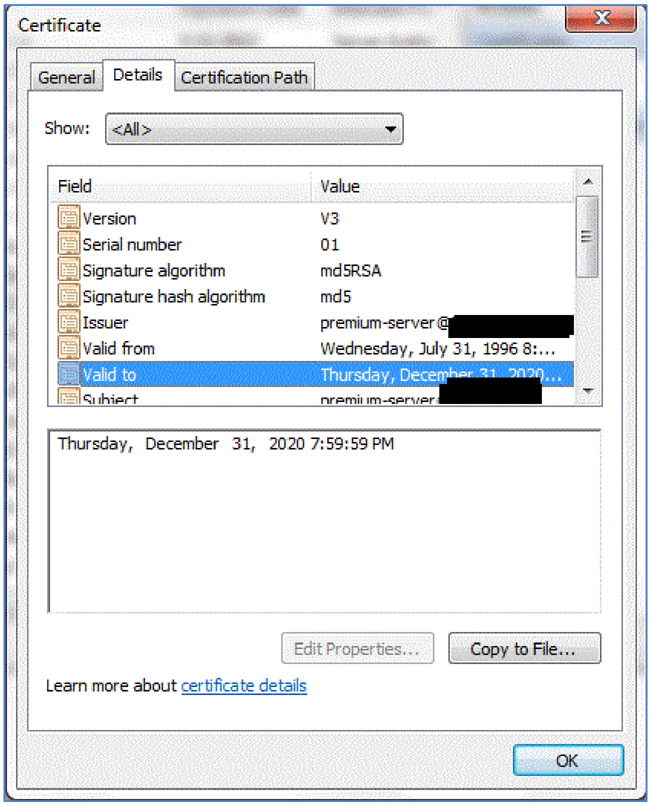
Figure 2.1: Public key cryptography uses two keys, one to encrypt data and one to decrypt it. It is not possible to decrypt a message using the same key that was used to encrypt it.

Signatures are encrypted forms of data in the certificate that are generated using message authentication functions. Signatures are used to help detect tampering. If someone were to try to alter some part of the certificate, such as the subject name, the tampering could be detected by recomputing the signatures. Message authentication functions are designed in such a way that even a small alteration, such as changing "server1.example.com" to "server2.example.com" would result in a different signature.

The data within an SSL certificate can be rendered in different forms. For example, a commonly used format is known as the privacy enhanced mail (PEM) Base 64 format, which is a text-based representation (see Figure 2.2). Such formats are useful for exchanging certificates but to view them it helps to use utilities such as the Microsoft Management Console (MMC) snap-in for certificates (see Figure 2.3).

```
-----BEGIN CERTIFICATE-----
HASDFasfjlksdjfui384389U;jJfEEoMA0GCSqGSIb3DQEBBQUAMEUxCzAJBgNV
BAYTAkFVMRMwEQYDVQQIEwpTb21lLVN0YXRlMSEwHwYDVQQKExhJbnRlcm5ldCBX
aWRnaXRzIFB0eSBMdGQwHhcNMDgwNjI2MTgxNTUyWhcNMDkwNjI2MTgxNTUyWjBF
+bo2UEYIzN7cIm5ImpmyW/2z0J1IDVDlvR2xJ659xrE0v5c2cB6Tf9lnNTwpSoeK
24Nd7Jwq4j9vk95fLrdqsBq0/KVlsCXeixS/CaqqduXfvwIDAQABo4GnMIGkMB0G
MQswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0ZTEhMB8GA1UEChMYSW50
ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQC89ZNxjTgWgq7Z1g0tJ65w+k7lNAj5IgjLb155UkUrz0XsHDnHFlbsVUg2Xtk6
A1UdDgQWBBTctMtI3EO9OjLI0x9Zo2ifkwIiNjB1BgNVHSMEbjBsgBTctMtI3EO9
OjLI0x9Zo2ifkwIiNqFJpEcwRTELMAkGA1UEBhMCQVUxEzARBgNVBAgTClNvbWUt
U3RhdGUxITAfBgNVBAoTGEludGVybmV0IFdpZGdpdHMgUHR5IEx0ZIIJAOqYOYFJ
fEEoMAwGA1UdEwQFMAMBAf8wDQYJKoZIhvcNAQEFBQADgYEAQwa7jya/DfhaDn7E
usPkpgIX8WCL2B1SqnRTXEZfBPPVq/cUmFGyEVRVATySRuMwi8PXbVcOhXXuocA+
43W+iIsD9pXapCZhhOerCq18TC1dWK98vLUsoK8PMjB6e5H/O8bqojv0EeC+fyCw
eSHj5jpC8iZKjCHBn+mAi4cQ514=
-----END CERTIFICATE-----
```

Figure 2.2: Example SSL certificate in PEM Base 64 format.



Show: <All>

Field	Value
Version	V3
Serial number	01
Signature algorithm	md5RSA
Signature hash algorithm	md5
Issuer	premium-server@[redacted]
Valid from	Wednesday, July 31, 1996 8:...
Valid to	Thursday, December 31, 2020 ...
Subject	premium-server[redacted]

Thursday, December 31, 2020 7:59:59 PM

Learn more about certificate details

Figure 2.3: The contents of SSL certificates can be displayed using the MMC Certificate Snap-in.

SSL certificates are frequently used for server authentication and encryption. Now that we have reviewed the structure of an SSL certificate, we can move on to review the process by which they are used for authentication and encryption.

Authenticating Servers with SSL Certificates

When a consumer visits a Web site to make a purchase, how would he know for sure he is dealing with the vendor he believes he is dealing with? After all, someone could conceivably copy the contents of a Web site, such as logos, images, text, etc. and masquerade as the legitimate business. Clearly, the looks of a Web site are not a good method to identify and authenticate a business. SSL certificates are not subject to simple copying attacks the way Web sites are, which is one of the reasons they are so important to online commerce.

Modern Web browsers are designed to work with servers that use SSL certificates for authentication. When a user navigates to a site and attempts to authenticate the site, the user's device, which we will refer to as the client, executes a handshake protocol, to establish secure communications with the server.

The process starts with the client device sending what is known as a ClientHello message. This message includes information about the highest version of the SSL protocol the client supports, a set of algorithms for authentication and encryption known as a cipher suite, a set of possible compression methods, and a random number.

Old Terminology for New Protocols

Once terminology is adopted, it is sometimes hard to change to new terminology. What we know of as the SSL standard today was formally known as the Transport Layer Security (TLS) protocol. TLS is the successor to the SSL protocols developed in the late 1990s. The protocol described here is actually the TLS version 1.2 protocol but we will use the term "SSL" throughout this e-book to refer to both the older SSL protocols and the TLS protocols.

When the server receives the ClientHello message, it selects the most secure protocol that both the server and the client support, the most secure cipher suite they both support, and a compression method from the set offered by the client. The server also generates a random number and sends it along with the other information in a ServerHello message.

Next, the server sends a Certificate message followed by a ServerHelloDone message to inform the client that the server is done with this initial step. The Certificate message contains a list of certificates that starts with its own certificate. Now, it's the client's turn to send information back to the server. Assuming the cipher suite selected by the client and server uses encryption and the server has not requested a certificate from the client, the next step is to send the Client Key Exchange Message. This message includes data known as the premaster secret. The TLS protocol supports multiple ways of creating a premaster secret; we will describe just one, the RSA-encrypted premaster secret.

One way to create a premaster secret is for the client to generate a 48-byte message and encrypt it with the server's public key. (The public key is provided by the server's SSL certificate). This premaster secret is then sent to the server using the RSA-Premaster Secret Message. The client and the server use the premaster secret along with random numbers to calculate a master secret, which is used to encrypt communications from this point forward.

The client then sends a ChangeCipherSpec message to the server indicating that all further communication will be encrypted and authenticated. The client sends a Finished message, which includes encrypted data that must be successfully decrypted by the server otherwise the handshake will fail. The server sends similar messages to the client to finish the handshake.

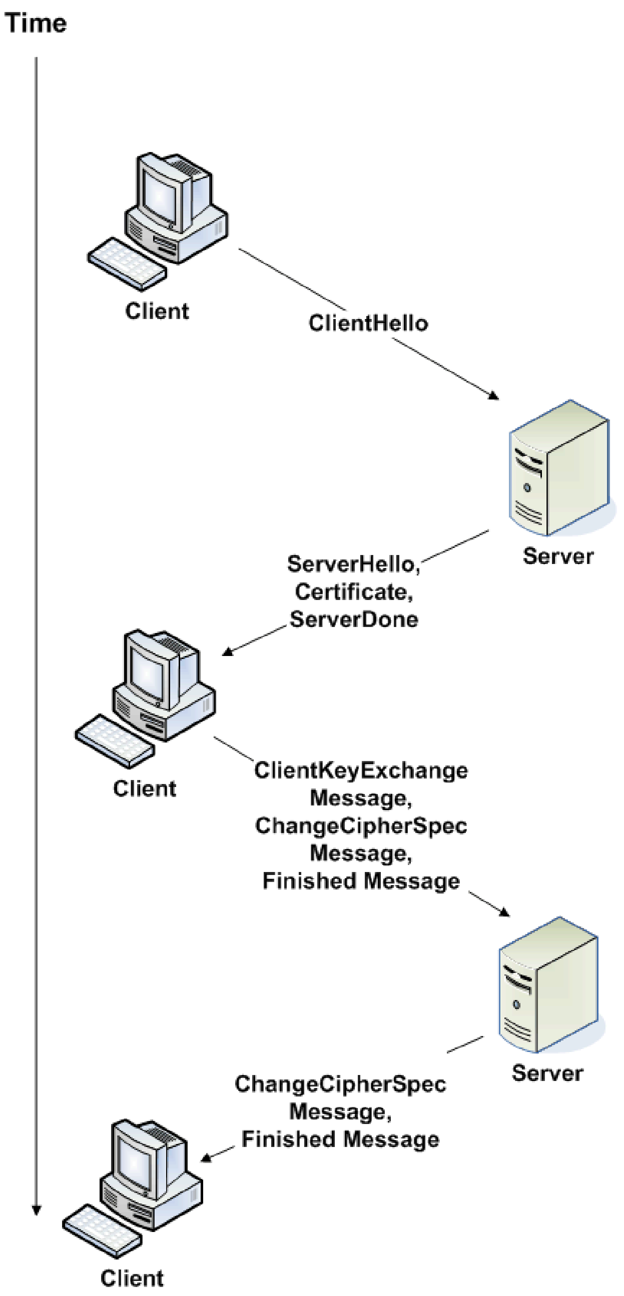
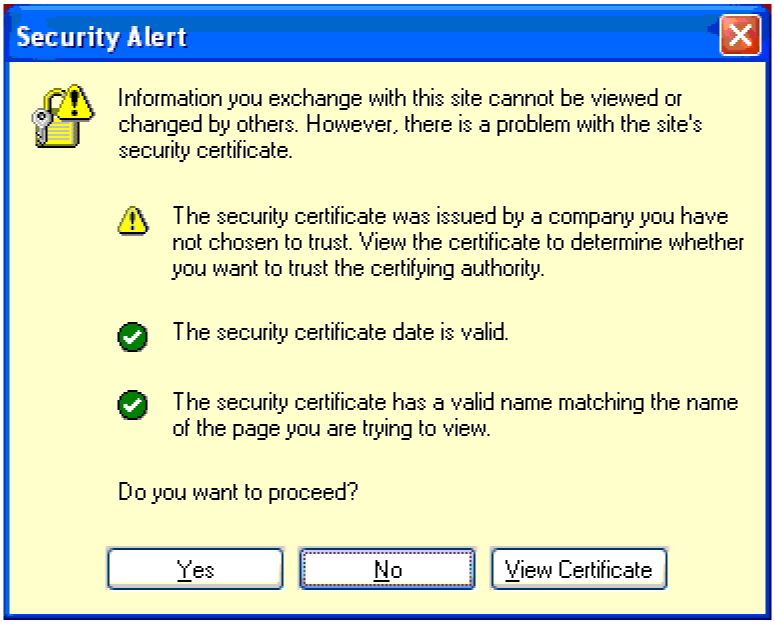


Figure 2.4: Steps in the SSL/TLS handshaking protocol to authenticate servers.

Note

When a server sends its certificate and the list of certifying certificates, that list is checked against CAs trusted by the client. A simple list may contain two certificates, the server's certificate and the certificate of the CA. If the CA certificate is known to the client as a trusted CA, the authentication will succeed (assuming all other steps succeed as described earlier.) If the CA is not recognized, the browser will display an error message such as that shown in Figure 2.5.



Information you exchange with this site cannot be viewed or changed by others. However, there is a problem with the site's security certificate.

⚠ The security certificate was issued by a company you have not chosen to trust. View the certificate to determine whether you want to trust the certifying authority.

✓ The security certificate date is valid.

✓ The security certificate has a valid name matching the name of the page you are trying to view.

Do you want to proceed?

Figure 2.5: A warning message generated when a server returns a certificate that was issued by an un-trusted CA.

The TLS Specification

The process discussed earlier describes the major steps in the handshake protocol. The TLS protocol supports a number of methods and levels of security for things such as exchanging keys and encrypting data. It also supports the use of client authentication as well as server authentication. For more information on these and other details about the protocols, see the Internet Engineering Task Force (IETF) Request for Comment (RCF) on TLS at http://tools.ietf.org/html/rfc5246.

In addition to the exchange between the client and the server, the client exchanges information with the CA to verify that the contents of the certificate are valid and apply to the particular server the client is communicating with.

Encryption and SSL

Part of the protocol described earlier included agreeing on a cipher suite. A cipher suite includes a bulk data encryption algorithm. The relative security of an encryption scheme is a function of both the algorithm you choose and the length of the key used for encryption. The TLS version 1.2 standard (the latest) supports the following algorithms:

- RC4
- 3DES
- AES

RC4 is a symmetric key algorithm, that is, it uses a single key for both encryption and decryption. The RC4 algorithm works by combining the plain text of a message with a random stream of bits. The RC4 algorithm is simple to implement and efficient and this makes it a common choice for bulk data exchange when encryption is required.

3DES, or Triple Data Encryption Standard (DES), differs from RC4 in that it works on blocks of data rather than streams of data. The original DES algorithm used a 56-bit key which was sufficiently long to protect messages at the time the algorithm was developed. As computing capabilities increased, so did the ability to break DES encryption. Triple DES improves on DES by using three 56-bit keys, which provides the equivalent protection of a 112-bit key.

The Advanced Encryption Standard (AES) replaced DES as the US government encryption standard in 2002. AES can use three different key lengths: 128 bits, 196 bits, or 256 bits. The 196-bit and 256-bit key length encryption is considered sufficient for top secret classified data of the US.

The SSL/TLS protocol supports different algorithms that have different properties. RC4 is a good choice for many applications, including online commerce. For the greatest level of security, the AES algorithm with a long key is a better option.

Different Uses of SSL Certificates

CAs have adopted SSL certificates to meet a range of requirements that businesses face. The most basic use of an SSL certificate is to authenticate a single server. Such a simple model is easy to understand but as is often the case with information technology, simple things can be difficult to manage. In other cases, simple solutions may not provide all the functionality that businesses or other organizations may require. This section will look at several uses and specializations of the SSL certificate:

- Domain-level certificates
- Organization-level certificates
- Server Gated Cryptography (SGC) certificates
- Extended validation (EV) certificates

The underlying SSL technology and protocols are the same, but the way these function with browsers can vary.

Domain-Level Certificates

Earlier, we noted that a certificate includes a subject identifier, such as the fully qualified name of a server, for example, server1.example.com. The specification of a single server name is not required by the SSL certificate standard. In fact, CAs are able to generalize the subject of a certificate so that a single SSL certificate can be valid for multiple subdomains. For example, a basic SSL certificate must specify that it is valid only for www.example.com. Such a certificate can only be used on a Web server for the example.com organization. If example.com also supported an ftp site at ftp.example.com or a mail server at mail.example.com, each of these sites would need their own certificate unless the certificate provider generated subject alternate names (SAN) in the certificate. Domain-level certificates, sometimes called wildcard certificates, allow organizations to purchase a single SSL certificate and use it for multiple subdomains.

Organization-Level Certificates

Different CAs may restrict wildcard SSL certificates to work with a single level of subdomains, such as:

- www.example.com
- ftp.example.com
- collaboration.example.com
- db.example.com

The same domain-level certificate may not work with multiple subdomain levels that might be found in large organizations. For example, a fully qualified server name might include multiple levels such as:

- ftp.marketing.example.com
- ftp.research.example.com
- mail.newyork.example.com
- mail.london.example.com

When choosing a single SSL certificate option, be sure to determine whether the CA provides wildcard certificates that meet your subdomain requirements.

Wildcard certificates help reduce management overhead because a single certificate can be used for multiple domains. This means only a single certificate needs to be renewed. It also makes it easier to deploy additional virtual machines in the domain with SSL certificates. If you determine you need more virtual servers in your environment, you can rapidly deploy another virtual machine because you do not have to wait for approval to procure, download, and install a new SSL certificate.

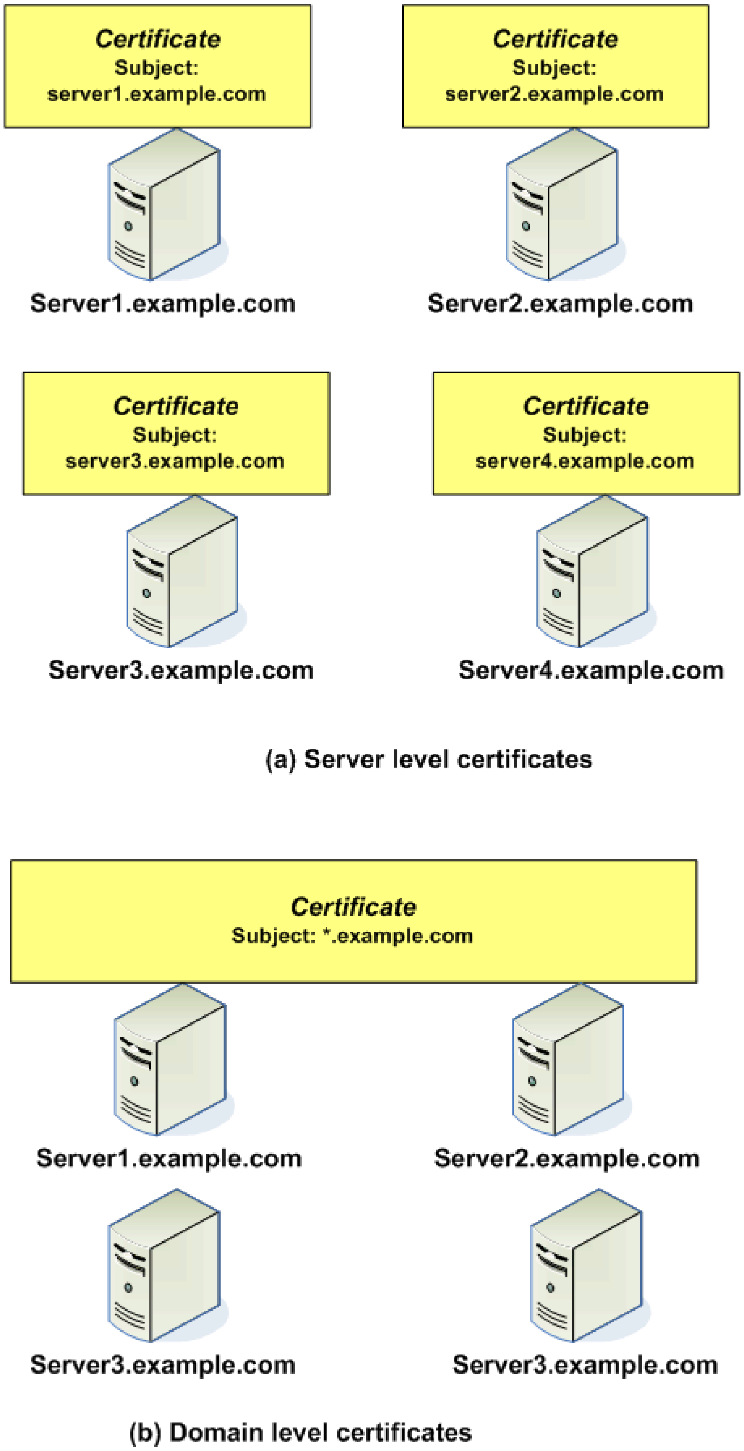


Figure 2.6: When server-level certificates are used, each server must have an individual SSL certificate (a); when domain level certificates, or wildcard certificates, are used, a single certificate can be used for multiple servers (b).

Server Gated Cryptography Certificates

In the 1990's, the US government limited the export of strong cryptography. Encryption technology that was destined for export outside the US was required to use weak encryption algorithms and short keys. As a result of this restriction, there existed users who were limited to 40-bit encryption that needed to connect to servers using 128-bit encryption. Server Gated Cryptography (SGC) certificates were created to allow browsers with the weaker encryption to connect to such servers. The limits on exporting strong cryptography in browsers have been removed, so there is less need for SGC certificates.

EV SSL Certificates

Sometimes getting an SSL certificate requires just some basic checks, for example, a CA might verify that you are the owner of a domain before issuing you a certificate for that domain. This level of identity verification is sufficient for many purposes, such as setting up an email server or a collaboration server for a group of remote telecommuters. In both cases, there is relatively low risk that someone would be motivated to spoof users into working with an attacker-controlled email or collaboration server. After all, what would be the value? The attacker may hope to get some emails and documents. Unless those documents contain significant intellectual property or have insights to high-value competitive bids, it probably isn't worth the effort that would be required to successfully set up a site masquerading as a legitimate site. That is not always the case, though.

Consider a bank or an online payment service. These are high-value targets. If someone could successfully spoof users into working with fake Web sites for even a short period of time, the attacker could collect authentication information and account data that could ultimately result in identify theft, credit card theft, or other forms of financial gain for the attacker. When there is that much motivation to masquerade as a legitimate site, there is comparable motivation to protect against it and this is why EV SSL certificates were created.

Contents of an EV SSL Certificate

EV SSL certificates include several pieces of information about the entity receiving the certificate:

- Organization name—The full legal name of the organization as it would appear in legal documents, such as incorporation records. It may also include additional information such as a doing business as (DBA) name.
- Domain name—One or more domain names controlled by the entity named in the certificate subject. It should be noted that domain names must be explicitly listed; wildcards are not allowed in EV SSL certificates.

- Jurisdiction of incorporation—This is name of the government jurisdiction, such as a state or country, in which the entity was incorporated.
- Registration number—A registration number assigned to the entity by the incorporating agency, such as a state corporation commission.
- Address of place of business—This is the address of the physical location of the business.

Before an EV SSL certificate is issued, the CA must perform various checks over and above those required for typical organization-level certificates to verify the identity of the organization receiving the certificate. These steps are defined by an industry working group, the CA/Browser Forum (http://www.cabforum.org/). All CAs issuing EV SSL certificates are expected to follow these guidelines.

Requirements to Acquiring an EV SSL Certificate

The guidelines for private organizations include:

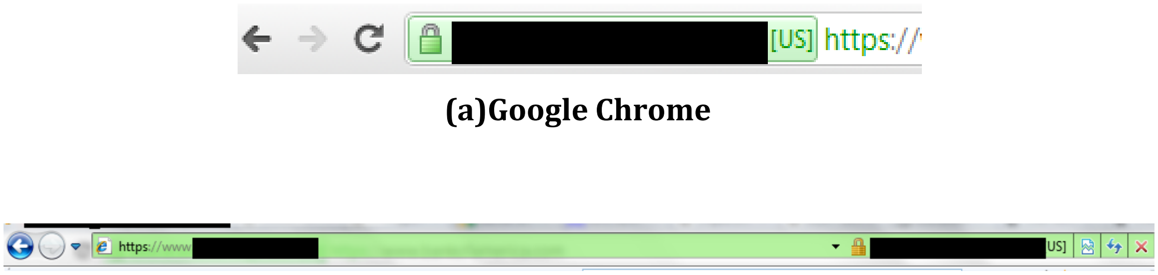
- The organization must be recognized and created by an incorporating or registering agency. The incorporating or registering entity must be chartered by a state or federal government.
- The organization must have named a registered agent or registered office or an equivalent physical facility.
- The organization must not be designated as inactive, invalid, not current, or similar designation by the incorporating or registering entity.
- The organization must have a physical place of business that can be verified by the CA.
- The CA must be able to legally conduct business in the location where the business is located.
- The business must not be listed on a government denial list or prohibit list, such as a trade embargo.

These guidelines ensure that the organization receiving an EV SSL certificate is at least recognized by some level of government as a legitimate organization and that the organization has a physical presence. Under these guidelines, for example, someone interested in committing fraud could not simply incorporate a shell company and get an EV SSL certificate without also investing in a physical presence somewhere. Comparable guidelines exist for issuing EV SSL certificates to government agencies or other subdivisions.

User Experience with an EV SSL Certificate

The additional information tracked in an EV SSL certificate and the additional steps taken to verify the identity of a subject both contribute to reducing the risk that an illegitimate operation would be able to acquire an EV SSL certificate. So far, we have not described how a typical user would benefit from an EV SSL certificate.

Today's major browsers can recognize EV SSL certificates and change their displays to give users visual cues that they are working with a site that has an EV SSL certificate.



(a)Google Chrome

(b) Internet Explorer

Figure 2.7: Browsers provide visual cues, such as the "green bar" to indicate when a user is viewing a site authenticated by an EV SSL certificate.

Choosing the Right Type of SSL Certificate

SSL certificates are used in many situations and with varying requirements. Some requirements are relatively simple and are based on the needs of a single application running on a single server. In other cases, a business may have a large number of servers that are fairly generic, in which case a key consideration is ease of management. In other cases, businesses want to convey to customers an additional level of security while at the same time mitigating the risk of an attacker spoofing their site with additional controls, such as the browser green bar.

Trust and SSL Security

SSL certificates promote trust between people and businesses that might otherwise not trust each other, or more specifically, not trust their Web sites. Many of us who shop online trust online retailers because we see the lock icon in the Web browser that indicates our communications are encrypted (and presumably, our credit card information is safe from simple eavesdropping). We trust that we are at our bank's Web site and not some spoofed version of the site because we see the green bar display indicating the use of a valid EV SSL. We trust these things because we trust the CAs behind them.

Now imagine if we did not trust the CAs. So what if a business has an EV SSL from Certificates-Are-Us, a fly-by-night operation that was just set up by a teenager in her parent's basement—would that be of much value or assurance to you? Probably not.

Our trust in CAs comes with a number of assumptions about how they will behave:

- CAs will protect encryption keys and their own root certificates so that others cannot generate fake certificates as if they were from the CA
- CAs will follow reasonable practices to verify the identity of an entity seeking an SSL certificate with additional protections for EV SSL certificates
- CAs will protect their information infrastructure to prevent breaches that would compromise the integrity of their certificate generation process

Those of us that use SSL certificates also bear some responsibility. We need to manage our certificates and ensure that private keys are kept private. We should use the appropriate type of certificate to meet our requirements, but also control them so that they are not abused. This can require additional controls and management protocols to track the number and types of servers using a domain certificate.

A word of caution: CAs are targets for cyber attacks. A European CA was recently compromised, which ultimately led to fake certificates being issued to several domains, including Google and a US government agency. (For more details, see Mike Lennon, "Infrastructure Compromise Put Fraudulent SSL Certificates in the Hands of Attackers," Security Week, August 30, 2011.)

Summary

SSL certificates are essential to the way we secure online business and support trust between businesses and their customers. SSL certificates use well-established technologies and protocols, such as the X.509 standard and strong encryption algorithms, to support server authentication and encryption. CAs have adapted to the needs of businesses, particularly with regard to easing the management burden of multiple SSL certificates with the introduction of domain-level certificates, and for the need for additional levels of verification with EV SSL certificates. The trust engendered with the use of SSL certificates is based on trust of CAs as well as the ability of those who use SSL certificates to protect them from improper use.

Download Additional Books from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this book to be informative, we encourage you to download more of our industry-leading technology books and video guides at Realtime Nexus. Please visit http://nexus.realtimepublishers.com.