Realtime publishers

The Shortcut Guide™ To

Protecting Against Web Application Threats Using SSL

sponsored by

Symantec™

Dan Sullivan

Copyright Statement

Chapter 2: How SSL Certificates Can Protect Online Business and Maintain Customer Trust

What underlies SSL certificates is a well-established method for securing communication and authenticating services. To better understand how SSL certificates can protect online business, it helps to know something about the inner workings of SSL. Working with SSL certificates is a bit like driving a car—you do not need to be an auto mechanic to drive a car, but it can help to know the basics of how your engine and transmission work.

This chapter is organized into three sections:

- How SSL certificates work
- Web applications with and without SSL certificate protection
- Authentication and trust

The first section looks under the hood of an SSL certificate to describe its components and how they work to secure communications and support authentication. The second section continues the look-under-the-hood approach and considers how an application without SSL certificate protections operates differently than one using SSL certificates. In the third section, continuing our regimen of delving into the implementation details of SSL certificates, we look at how SSL certificates are created, the different types of SSL certificates, and the role of SSL certificate providers in establishing and maintaining a trust relationship between providers of SSL certificates, businesses that use them, and customers that expect the kinds of protections they provide.

How SSL Certificates Work

When we receive an SSL certificate from a provider, we receive a file. That may seem like a bit of a letdown at first. After all, this is something that will be used to encrypt communications and provide evidence for identity claims of servers. These are fairly important tasks, and they are all enabled because of one small file? Well, yes and no.

Yes, the SSL certificate file is essential for providing encryption and authentication services, but it is really just one part of a more complex set of protocols. Actually, an SSL certificate by itself would be of little use to you if it weren't for the established protocols that make use of the information stored within the SSL certificate file. The important security tasks are not enabled solely because of an SSL certificate file. It is the combination of the SSL certificate and the protocols that define how it is used that provide the security controls we seek. Let's take a look inside an SSL certificate and then examine the protocols that make use of it.

Components of an SSL Certificate

Figure 2.1 show the components of an SSL certificate. SSL certificates use the X.509 certificate structure, which includes information about the subject, such as a domain, and the encryption algorithm used to create encrypted data that can uniquely identify an entity (these are known as signatures):

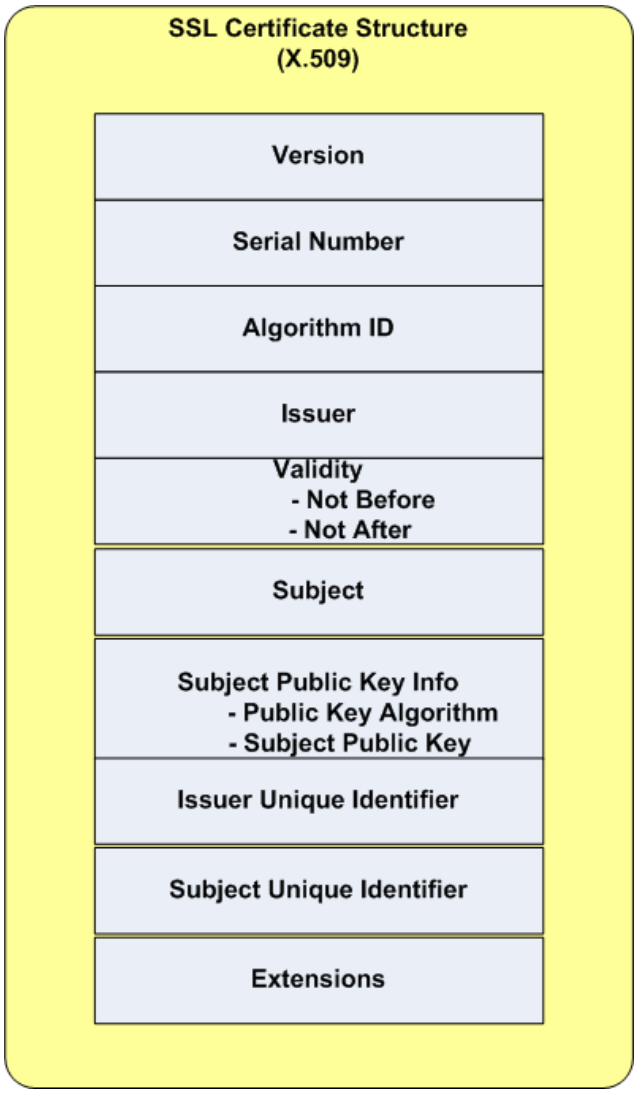

Figure 2.1: The data structure for representing an SSL certificate is based on the X.509 certificate standard.

- The version number indicates which version of the X.509 specification is used. Newer versions support additional extensions and a unique identifier.
- The serial number is a unique number assigned by the certifying authority that issued the certificate. Certifying authorities are responsible for tracking these numbers so that the combination of issuer and serial number is unique across all X.509 certificates.
- The algorithm ID (referred to as a "signature" in the X.509 specification) is the identifier of the algorithm used by the certifying authority to generate the certificate.
- The issuer is the name of the certifying authority that issued the certificate. In addition to the name of the issuer, this field can contain the location of the issuer and the organizational unit within the issuing company that was responsible for creating the certificate.
- The validity section includes two dates, one marking the start period for which the certificate is valid and one indicating the end date that it is valid.
- The subject field is the name of the entity requesting the certificate. This name is in the form of a distinguished name that is unique to that entity within the certifying authority. Like the issuer field, this attribute can contain information about the subject's location and the organizational unit within the entity that requested the certificate.
- The subject public key field contains a public key, which is a string of characters, and the name of an algorithm with which the key is used. Why do we need this string of characters known as a public key? This key is part of the technology known as public key cryptography. We do not need to delve into too many details, but it is important to understand the basics. Here is how it works: When someone wants to send you an encrypted message that only you can read, that person would get your public key from your digital certificate. (Actually, she would use a program such as PGP to do this). With that key and the name of the encryption algorithm, the person can then encrypt the message. The public key is not like a key used to open and lock doors. The public key is a one-way key. It's only good for "locking" (that is, encrypting) but it cannot be used to "unlock" (that is, decrypt) the message. For that, we need a private key.
- The private key is created at the same time as the public key. You can share your public key with anyone who might want to send you an encrypted message and you do not have to worry about them reading an encrypted message someone else sent to you. The only way to decrypt a message encrypted with a public key is to use the corresponding private key. As long as no one else has your private key, they cannot read your encrypted messages.

With Enough Time and Resources

It is not theoretically impossible to read someone else's message without the private key. If you have enough cryptographic knowledge and access to large-scale computing resources (think large secret-government-agency-level resources), you could eventually decrypt a message without the private key. Unless you are passing around state secrets, the value of the decrypted message probably would not justify the time and expense necessary to try to crack the message. By one estimate, if you could check a billion (1,000,000,000,000,000,000) AES keys per section it would take 3,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000 to try all possible keys (Source: Wikipedia, Brute Force Attack).

- The issuer and subject unique identifiers are used to store identifiers that would uniquely identify an issuer or subject in cases where the name of either entity is reused.
- Extensions were added in version 3 of the X.509 standard and support the use of additional attributes that can be used to store several common extensions as well as private information used within a community of users.

X.509 Certificate Specification

For a more detailed and formal description of the X.509 certificate, see the Internet Engineering Task Force RFC at http://www.ietf.org/rfc/rfc2459.txt.

An SSL certificate contains three broad types of data:

- Information about the subject that owns the certificate and is identified by it
- Information about the certifying authority that issued the certificate
- Cryptographic information such as the subject key and algorithm

You can examine certificates on your Windows devices using the Microsoft Management Console (MMC) and the Certificates Management snap-in (see Figure 2.2). Even if you haven't installed any certificates yourself, you can still view certificates that are installed with the Windows operating system (OS). These are typically for trusted entities like certifying authorities. Your organization may also have installed additional certificates.

Resource

If you are not familiar with the MMC, see Microsoft Management Console 3.0 at Microsoft Technet.

Now that we've covered what is in an SSL certificate, let's take a look at how this information is used.

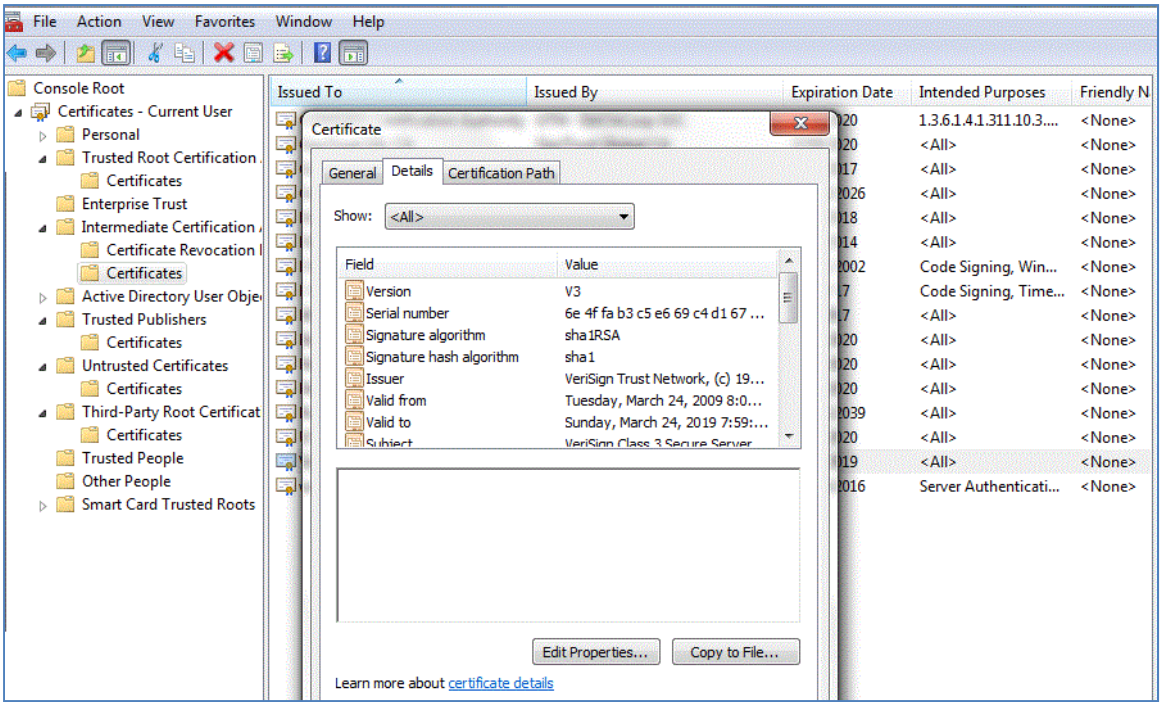

Figure 2.2: The MMC Certificates snap-in tool provides a viewer for reviewing the contents of SSL certificates.

Overview of How SSL Certificates Secure Communications

SSL certificates play a key role in establishing secure communications. They actually provide two services: identifying a party in the communication and providing a public key that can be used to encrypt messages sent back to the server. As we will see, the public key is used to set up a secure communication channel, which is then used to further exchange information and establish an efficient and secure channel for exchanging data.

> **SSL and TLS: A Rose by Any Other Name?**
>
> The Secure Sockets Layer (SSL) protocol is the predecessor of the Transport Layer Security (TLS) protocol. They both are used for securely communicating over the Internet. Although they are different protocols, the general descriptions here address concepts common to both. "SSL certificates" is a common term used to describe digital certificates used for encryption and authentication, so this guide will use the term "SSL" as synonymous with "TLS," as is typically done.

When you navigate to a server using a secure protocol, such as Hypertext Transfer Protocol over SSL (HTTPS), your computer, which we'll refer to as the client, will perform a handshaking protocol to set up a secure communication channel. The steps are as follows: The client requests a secure connection to a server and presents a list of security mechanisms it supports. These are known as encryption cipher suites that have functions that the client can work with. From the list, the server chooses the most secure option that it is able to support and sends its choice to the client. The server sends its SSL certificate, which includes the server's name, public key, and the identity of the certifying authority. Next, the client might send a message to the certifying authority to verify that the certificate is still valid. This option is available because it is possible for a certificate to be revoked during its valid period. Revoked SSL certificates can be checked using either the Online Certificate Status Protocol (OCSP) or certificate revocation lists (CLRs).

At this point, the client has authenticated the server and agreed on a cipher suite. The server may optionally request a client's certificate for mutual authentication. This is more likely in cases where the client should be known, such as when using a virtual private network (VPN); mutual authentication is less likely in cases where the client is contacting a public Web site set up for general commerce (see Figure 2.3).

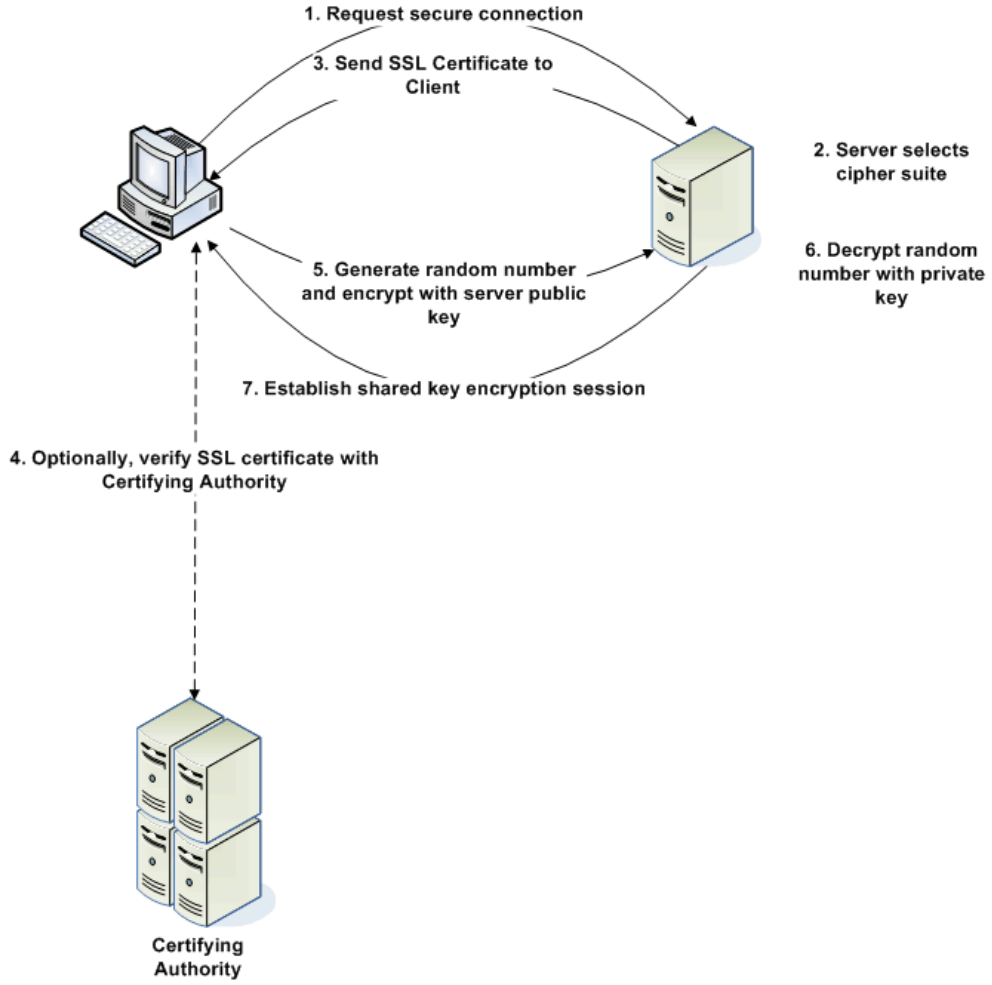

Figure 2.3: Steps to establish a secure connection using SSL certificates.

After the client and server have a secure channel, they can securely exchange information that allows them to create a secure session that is more computationally efficient. The more efficient methods, known as symmetric key cryptography, are faster but require both the client and server to know about a shared key. The next steps allow the client and server to securely exchange such a shared key:

- The client generates a random number and encrypts it with the server's public key.
- The server decrypts the encrypted random number using its private key.
- The client and server establish a secure communication using a shared key and an encryption method that requires only one key for both encryption and decryption.

After completing these steps, the client and server are ready to securely exchange data.

Overview of How SSL Certificates Support Authentication

Peter Steiner's iconic 1993 New Yorker cartoon of a couple of dogs in front of a computer with the caption "On the Internet, nobody knows you're a dog" captures a fundamental problem with the Internet: How do we know who we are interacting with? Let's skip the philosophical issues about how we can know something and settle for trusting that someone (or something like a server) is who or what it purports to be.

We have a bit of a circular problem here. We want to know how we can trust someone online when we don't trust them in the first place when they assert to be someone or something. Any of us can set up a server and put up a Web page proclaiming to be a bank. We might even produce an authentic-looking site by copying pages from a real bank. How will customers know the difference? They will know because we will not be able to get an SSL certificate from a trusted certifying authority that vouches for our identity. The major browsers change the display of the navigation bar when displaying content from a site that uses SSL for identification and encryption (see Figure 2.4). Locks are used to indicate encrypted communication. The "green bar" indicates the use of a special type of SSL certificate known as Extended Validation (EV) SSL certificate, which we'll talk about a bit later in this chapter.

Figure 2.4: Browsers automatically change the navigation bar display when rendering content from a site with a trusted SSL certificate using encrypted communication.

The changes in the browser display are a visual cue that the site has an SSL certificate that has been provided by a trusted certifying authority. Browsers come preconfigured with a set of trusted certifying authorities. When a connection is made to a server, the server sends its SSL certificate to the browser. The browser then makes a number of checks:

- Verifying that the domain name of the site matches the domain name of the SSL certificate
- Verifying the current date is within the valid date ranges
- Checking the issuer and verifying it is one of the trusted certifying authorities known to the browser

When a certificate is issued by a certifying authority that is not trusted by the browser, most browsers will display a warning message (see Figure 2.5).

Warning messages such as the one that Figure 2.5 shows as a rule should not occur when working with trusted commercial or government sites. You are likely to see a warning if you navigate to a site that is using an invalid certificate or a certificate that was generated by an untrusted authority. Certificates may be invalid because they have expired or the domain name of the site does not match the subject name on the certificate. You may also see such messages when using self-signed certificates, which we create for ourselves, for example, in a development environment.

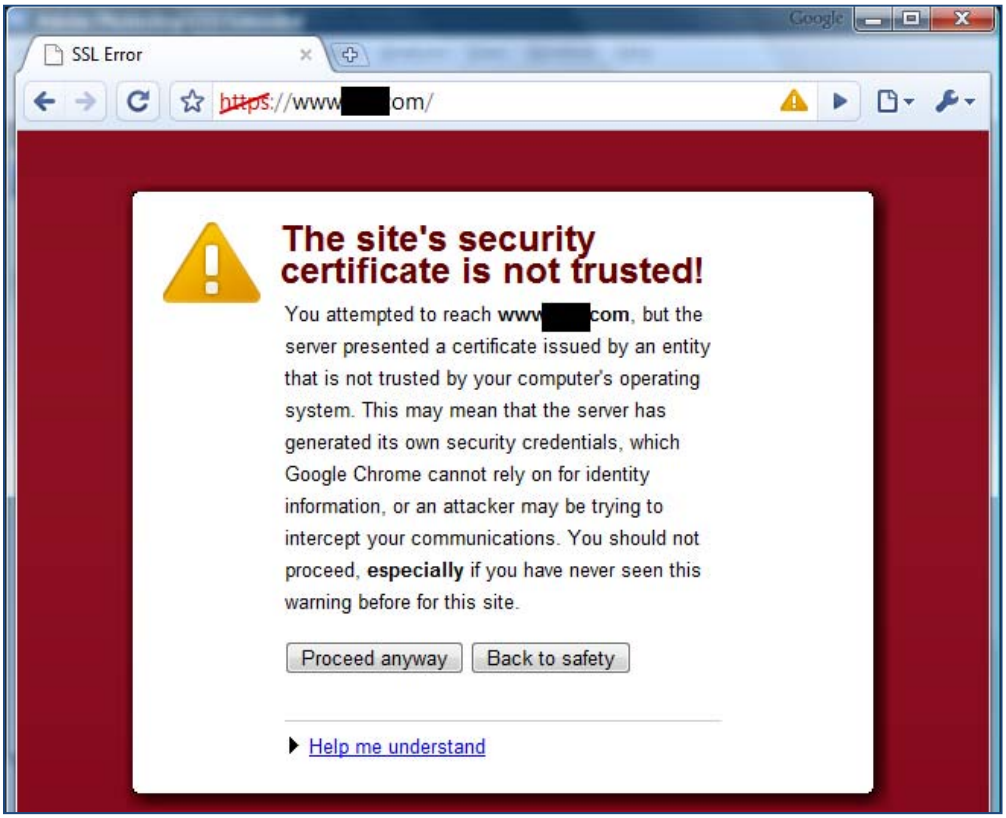

Figure 2.5: An example warning message presented by a browser when an SSL certificate is used by a certifying authority that is not trusted by the browser.

Now that it has been established that SSL certificates provide the means to encrypt communications and authenticate servers, it is time to consider how these capabilities work with Web applications.

Web Applications Without and With SSL Certificate Protection

Let's consider two scenarios: Web applications without SSL certificate protection and Web applications with their security benefits. We'll start with the unsecured examples.

Scenario 1: Web Applications Without SSL Certificate Protection

Consider an executive working with a Web collaboration application. The application supports common functions needed for group work including the ability to upload files, search collections of documents, and add notes and other metadata about the documents. The collaboration application does not use SSL certificates and instead relies on other security measures, such as access controls and network security, to protect its users.

The executive in our scenario is working on a proposal for a new client. The value of the potential contract is substantial, and there are multiple competitors vying for the work. Today, the client decides to get away from the office to work on the proposal. She heads to the coffee shop down the street and sets to work. After a couple of hours, the executive is ready to upload the proposal to the collaboration server. She connects to the coffee shop's WiFi, starts the collaboration application, and uploads the proposal. Unknown to her, someone else in the coffee shop was monitoring network traffic in search of some useful competitive intelligence. Figure 2.6 illustrates this scenario.

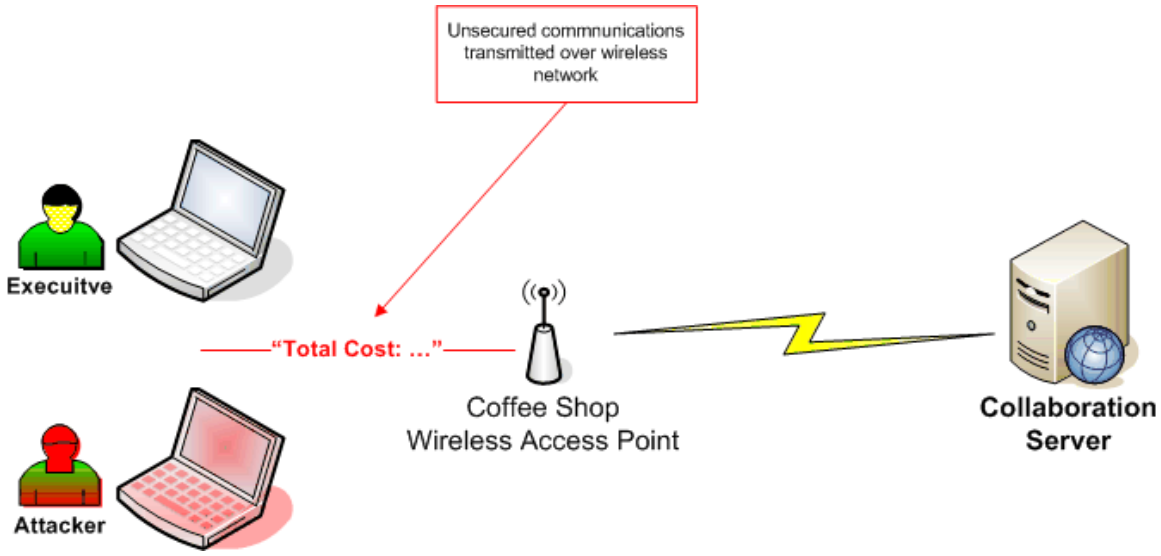

Figure 2.6: Unsecured communications can be detected and captured by others.

The communication was not encrypted by the application server or on the WiFi network, so the document was sent as clear text. This allowed a third party to pick up the network traffic and discover the contents of the document. Whatever competitive advantage the executive's firm had could have been undermined by this data leak.

> **Note**
>
> Although this example is fictitious, this kind of attack is not. See, for example, cyberattacks on energy companies for proposal data.

Unauthorized monitoring of communication is only one problem with not using SSL certificates. Another problem is the potential for someone creating a server that appears to be legitimate but is actually only *masquerading* as a legitimate server. This is known as spoofing.

Consider another scenario. One of your regular customers decides to come to your company site to place an order. She has done this dozens of times and doesn't think much about it. She types in your site's domain name and sees the usual order page. She tries to start a new order but receives an error message. It seems, according to the Web page displayed, that your company has lost some customer data including hers. She is prompted to enter her name and bank account information. The problem is, this is not your business' site and your customer has no way to tell.

Unknown to the customer, the service that translates domain names into Internet addresses (domain name system—DNS) for her has been compromised. It seems her company has been the victim of a DNS cache poisoning attack. DNS servers translate domain names, such as www.example.com, into a numeric address, such as 192.169.0.1. When a DNS cache is poisoned, someone changes the legitimate numeric address to one assigned to an attacker-controlled server. Your customer's traffic is routed to the attacker's server with no obvious indication something is wrong as Figure 2.7 shows.

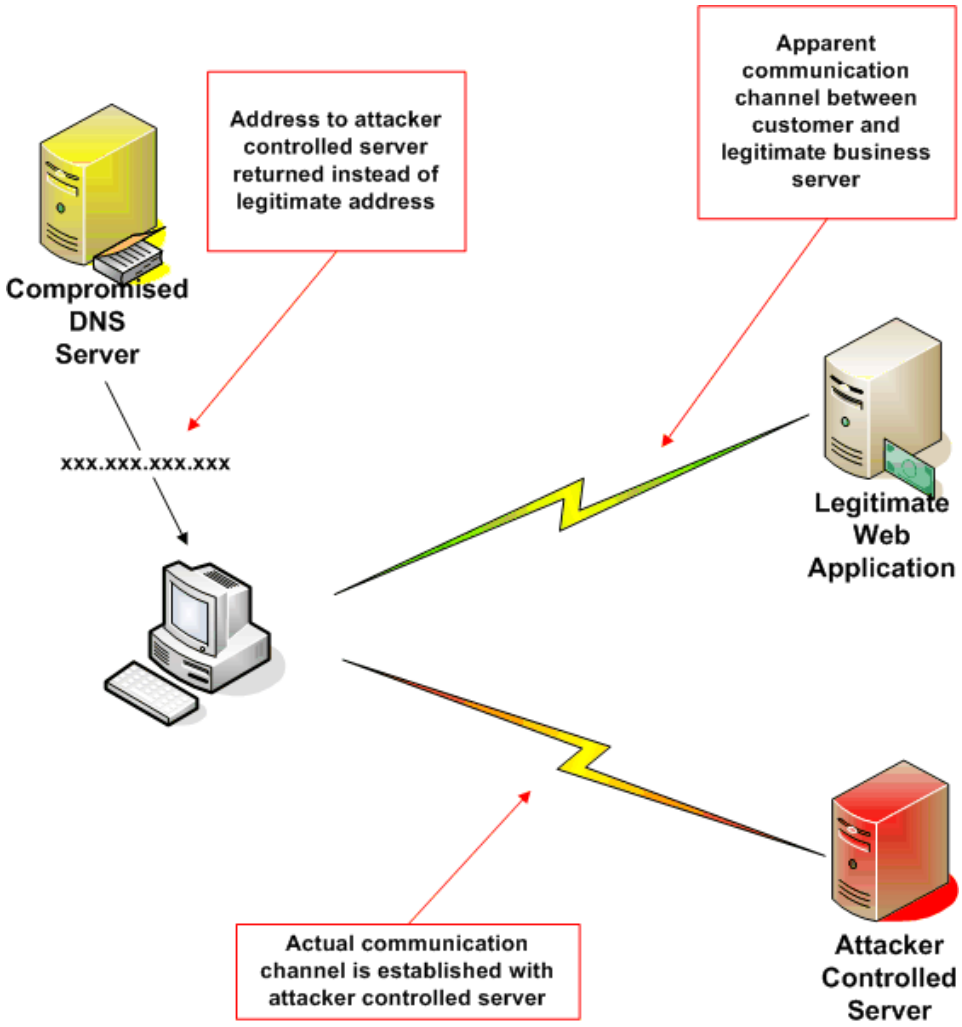

Figure 2.7: Without authentication provided by SSL certificates, users can be lured to use spoofed servers and applications that appear to be legitimate servers and applications.

In case you might be tempted to think that eavesdropping on your communications or server spoofing is only a theoretical problem that is not likely to affect you, consider these additional points:

- Sidejacking attacks involve using unencrypted data to allow an attacker to steal your session information and interact with a Web site as if the attacker were you. See the Firesheep tool for a demonstration of how this can be done.
- Attackers can find wireless networks with tools like NetStumbler, and even if the networks are not broadcasting identification data, tools like Kismet can be used to get that data.
- Auditing and testing tools, such as DSNiff can be used to scan network traffic—great for testing weakness in your network but these tools are just as useful to attackers with malicious intent.

Without the encryption and authentication protections enabled by SSL certificates, we and our customers and collaborators are vulnerable to a variety of attacks. Let's consider the earlier scenarios but with SSL certificates in place.

Scenario 2: With SSL Certificate Protection

In the case of the executive working in the coffee shop, had the collaboration server used SSL certificates, the executive could send secure communications to the server. In the event that an attacker intercepted the traffic, it would appear to be a random stream of data, not a valuable and confidential business proposal (see Figure 2.8).

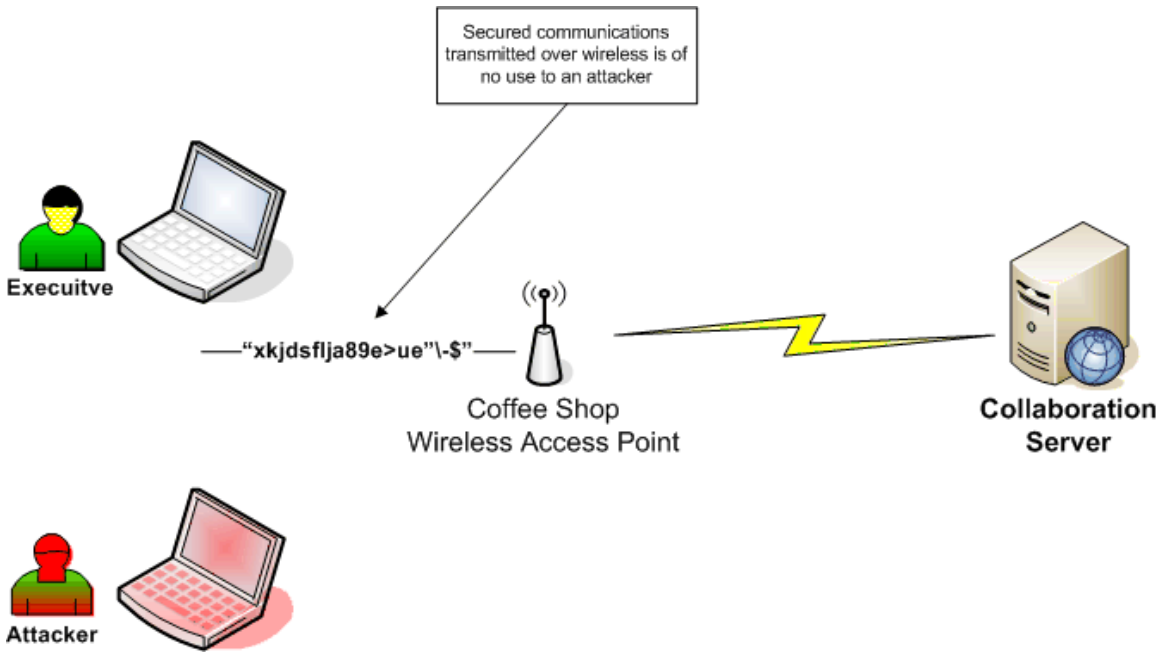

Figure 2.8: With SSL certificate-based encryption, data transmitted over wireless networks will appear to be more like random data than what it actually represents.

The case of the customer who maliciously redirected from her intended target to an attacker-controlled Web site would turn out differently as well if SSL certificates were used. One of the problems for the customer was that there was no indication that she was at a malicious site. With SSL certificate authentication, she would have received a warning from her browser that something was not consistent with the malicious site.

If the malicious site was using an SSL certificate, it would have inconsistent information because either the certificate subject entity would be something the attacker could get a certificate for, which would not match the spoofed domain name, or the attacker acquired an SSL certificate from an untrusted provider. In either case, the user would be alerted to the fact that something was not as it usually is.

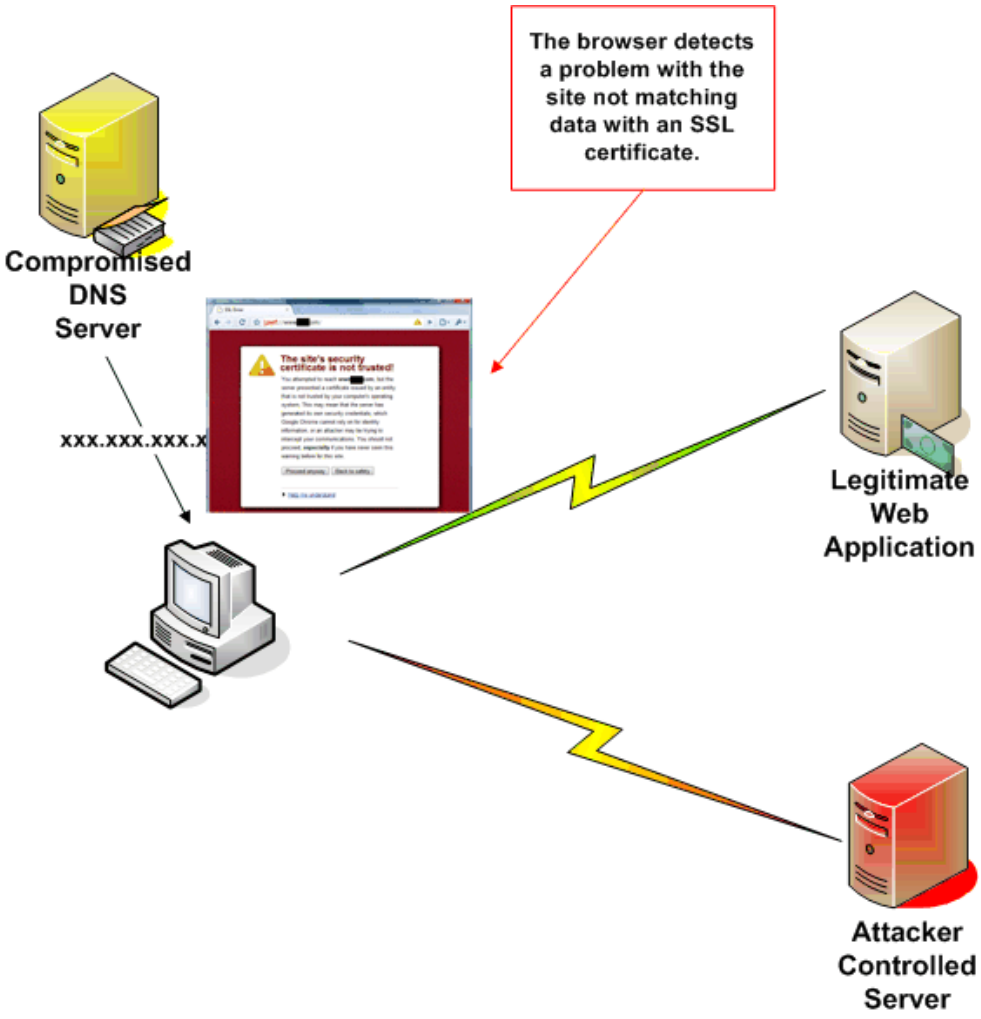

Figure 2.9: A spoofing attack would trigger an error on the client browser and alert the customer to the fact that there is some kind of problem with the site.

SSL certificates enable encryption and authentication, but businesses need more than that. Businesses need to know they can trust who they are dealing with. That is the ultimate reason we deploy SSL certificates.

Authentication and Trust

Trust cannot be reduced to digital certificates or encrypted messages. Trust is established over time and requires one party to be confident that another party will function as expected. We can't have trust with businesses or individuals we never met or have not heard of. We can, however, establish a trust relationship with an unknown party when we trust a third party and that third party assures us that the unknown party is trustworthy. This role of trusted third party is played by certifying authorities. These are companies that have built a business and a reputation around the business of verifying identities.

How Certifying Authorities Authenticate

The Internet community has different levels of need when it comes to verifying identities. For example, we might be ready to put information about our calendar into a site established to schedule company softball games with minimal verification but we are much more careful about our online banking practices. Certifying authorizes have created different procedures for verification, depending on the level of trust that is needed:

- Domain-level verifications are used when the certifying authority needs to establish that the requestor of a certificate is the owner of a domain name. Checking the domain registry may be sufficient for this. (See whois.net or any one of many other services that provide details about domain owners.)
- Business verification is used when a certificate is to be provided to a business and more evidence than domain ownership is required to establish identity.
- Extended validation (EV) certificates require the most comprehensive verification, including legal documentation and checks on the physical location of the business.

Certifying authorities go through varying levels of due diligence to verify the identity of domains or businesses that receive their certificates. That is only one part of the process for establishing trust. Another part is educating users about these practices and providing information on how to ensure that legitimate certificates are in place.

Developing Trust

Businesses have long used marks to indicate a product or service is trustworthy. Marks ranging from the Underwriter's Laboratories "UL" symbol to the Better Business Bureau logo have been used to indicate the safety of products and the trustworthiness of businesses. With the emergence of online business activity, it would help to have trust marks suitable for the Internet. We have trust indicators with SSL certificates, which use a lock in the browser address bar to indicate a secure communications channel. Green bar indicators are used with EV SSL certificates. Businesses can help promote knowledge about these trust marks by educating customers about their use and by using them on business sites as well as promoting other safe online practices. Trust can be further reinforced with trust marks such as a trusted seal from a certifying authority or an established organization such as the Better Business Bureau.

Businesses should also use the appropriate type of SSL certificate for their needs. When low trust is required by users, a simple domain certificate can be used. Sites that do not collect confidential or private information, do not require financial information or credit card data, and do not deal with other highly-valued data may be well served by conventional domain- or business-level certificates. When additional verification is required to help assure users that the site is legitimate, EV certificates should be considered because they provide highly-visible trust indicators such as the green bar and the display of the organization name.

Also, to develop trust, try to avoid situations in which your SSL certificates will generate error messages on customer browsers. These can occur for a number of reasons, so be sure to follow basic guidelines for good SSL certificate management:

- Do not use self-signed certificates for customer or other externally-accessed servers
- Use certificates from certifying authorities recognized by all major browsers
- Keep certificates up to date and renew them before they expire

A combination of factors goes into establishing trust: working with known and trusted certifying authorities, using the appropriate types of SSL certificates, and using trust marks and educating users about risks.

Summary

SSL certificates enable encryption and authentication. These are essential for securing Web applications and protecting customers from eavesdropping, data leaks, and spoofing attacks. SSL certificates enable key functionality required to build a trust relationship between business partners that might not have a pre-existing relationship. The best-designed application can have all the features and capabilities that users want, but if users do not trust the application, those features may not be used.