SSL Acceleration and Offloading: What Are the Security Implications?

Date: Jun 02, 2004
Section: Articles :: Web Server Security
Author: Deb Shinder
Printable Version
Rating: 4.3/5 - 14 Votes

Secure Sockets Layer (SSL) is a popular method for encrypting data transferred over the Internet. It is commonly used to provide secure transfer of credit card information and other sensitive data in an e-commerce situation. SSL can also be used to create a virtual private networking (VPN) tunnel, as an alternative to "old standbys" IPSec and PPTP. I will discuss SSL VPNs in next month's article titled VPN Options.

If you would like to receive an email when the article SSL VPNs is released, subscribe to the WindowSecurity.com Real-Time Article Updates from our Newsletter subscriptions page.

SSL uses symmetric encryption (a single shared key for both encryption and decryption) to provide data confidentiality. Although this is considered less secure than asymmetric (public key) encryption that uses a matched key pair, that disadvantage is offset somewhat by the fact that symmetric encryption is much faster (something that is important in e-commerce transactions) and requires less processing. SSL encryption is strengthened by the use of a longer key; it can use DES, 3DES, RC2 and RC4, with key length up to 168 bits.

Note
Transport Layer Security (TLS) is an extension of and the successor to SSL and you will often see them discussed as "SSL/TLS." However, the two are not interoperable. Most modern Web browsers support both.

Despite the fact that it uses faster symmetric encryption for confidentiality, SSL still causes a performance slowdown. That's because there is more to SSL than the data encryption. The "handshake" process, whereby the server (and sometimes the client) is authenticated uses digital certificates based on asymmetric or public key encryption technology. Public key encryption is very secure, but also very processor-intensive and thus has a significant negative impact on performance. E-commerce sites are especially prone to SSL bottlenecks, and companies may lose business when customers encounter slow response and long waits.

In this article, we will take a look at some of the solutions that can be implemented to address this performance and processor-load problem. Specifically, we'll discuss the concepts of SSL acceleration and two different SSL offloading techniques: SSL termination and SSL bridging (also referred to as SSL initiation). We'll also look at possible security implications involved in deploying these solutions.

How SSL Works

SSL uses a "handshake" protocol to negotiate and establish a session between the client and server computers. During the handshake sequence, digital certificates are used to authenticate identity, and the communicating computers agree upon a hash algorithm (such as MD5 or SHA-1) for ensuring data integrity.

An SSL session is initiated by a message sent to the server by the client computer (called a Client Hello message). The server responds with a Server Hello message. These messages establish parameters for the communication, including what version of SSL will be used, a session ID (if the client is continuing a previous session), the "cipher suite" that will be used (this identifies the key exchange algorithm, encryption algorithm and hash function), and the compression algorithm that will be used.

The client is able to authenticate the server's identity because the server sends its digital certificate containing its public key. In some cases, two-way authentication is necessary. That is, the server must verify the identity of the client in addition to the client verifying the server's identity. Internet banking is a good example of such a situation. In this case, the server sends a client certificate request.

The client responds with its own certificate in two-way authentication situations. The client also sends a key exchange message with the premaster secret, which it encrypts with the server's public key.

The server decrypts the message with its private key, which authenticates its identity since only the private key that belongs to the same key pair as the public key used to encrypt the message will be able to decrypt it.

The client may send a hash of the foregoing messages encrypted with its private key (in two-way authentication situations). The server can then verify the client's identity by decrypting this with the client's public key. The client will then send a message to let the server know that subsequent messages will be encrypted with the negotiated algorithms, and finally, the client sends a "Client Finished" message, which is encrypted and hashed.

The server will also send a message to tell the client that its subsequent messages will be encrypted, and then sends a "Server Finished" message that is encrypted. If the client is able to decrypt it, the handshake has succeeded. Communications between client and server are encrypted using the keys and algorithms that were negotiated in the handshake process.

What is SSL Acceleration?

One of the first methods used to address the SSL performance problem was the hardware accelerator. This is a card that plugs into a PCI slot or SCSI port and contains a co-processor that performs part of the SSL processing, relieving the load on the Web server's main processor. SSL hardware accelerators are made by a number of vendors, including nCipher (www.ncipher.com/nfast).

Typically, only the RSA operations that use public key cryptography are offloaded to a hardware accelerator. That's because the symmetric encryption is much faster and doesn't need to be offloaded; in fact, offloading those operations could actually result in degrading performance. So the accelerator card performs the asymmetric cryptography operations and the symmetric cryptography operations are performed by the server's main processor.

The level of performance improvement that you get with a hardware accelerator varies from one vendor to another. Some vendors claim an increase in SSL processing capacity of 500% or more. You can add more than one card to the same server to increase capacity even more, and you can install dual cards for high availability and failover. Some cards also include additional functions such as key management.

Some accelerators, called network accelerators as opposed to server-side accelerators, are designed to work with network switches and intercept and decrypt SSL traffic before it reaches the server. This goes beyond mere acceleration and gets into the area of SSL offloading.

What is SSL Offloading?

In a sense, an SSL hardware accelerator is performing SSL offloading, because part of the SSL processing is "offloaded" from the server's CPU to the card's co-processor. The term "offloading," however, is generally used to describe an appliance or a completely separate computer that performs *all* SSL processing, so that the SSL load is taken off of the Web server completely.

An advantage of an offloader, as opposed to the typical accelerator, is that it can do SSL processing for more than one Web server, whereas the accelerator card is tied to a single server.

SSL offloaders can greatly enhance the effectiveness of intrusion detection systems, virus detection systems, etc. These systems are unable to detect attack signatures and virus signatures that are contained in data that's SSL encrypted, but the offloader can decrypt the data so the IDS, virus software or application layer firewall can examine its contents and block suspicious packets.

There are two basic ways of doing this: the offloader can perform SSL termination or SSL bridging (sometimes called SSL initiation).

SSL Termination

An SSL offloader that acts as an SSL terminator decrypts the SSL-encrypted data and then sends it on to the server in an unencrypted state, so that the server does not have to perform decryption and the burden on its processor is relieved.

The unencrypted data may pass through an IDS, virus detection system and/or application layer firewall on its way to the server.

SSL termination increases the performance at the server level, but also poses a security problem: data is traveling from the offloader to the server without the protection of encryption.

SSL Bridging (Initiation)

There is a method for allowing inspection of SSL-encrypted data before it reaches the server to prevent application layer attacks hidden inside, without compromising the end-to-end security of the data. Microsoft calls this technology SSL bridging. Other vendors use different terminology; for example, SonicWall calls it SSL initiation.

Regardless of the name, here's how it works: the application layer aware firewall intercepts and decrypts SSL-encrypted traffic, examines the contents to ensure that it doesn't contain malicious code, then re-encrypts it before sending it on to the server. Although the data is temporarily in a decrypted state at the firewall, it is protected when it is sent across the network.

However, this means that the server will have to decrypt the data again, thus negating the performance advantage of SSL offloading.

What are the Security Implications of Offloading SSL?

SSL offloading can greatly increase the performance of your secure Web servers, thus increasing customer satisfaction. However, offloading means the SSL connection extends only from client to offloader, not from client to server. Data passes across the network unencrypted from offloader to server.

Granted, this data is moving across your internal network, not the public Internet. Thus the question becomes: how secure is that internal network? That depends to a great extend on your network topology. If you have the offloader and server deployed behind a department firewall on a secure subnet where only critical servers are located and to which users don't have direct access, you might be confident in allowing unencrypted data to pass from offloader to server. If you are offloading SSL processing to a firewall located on the network edge, the exposure and risk of compromise after the data is decrypted is much greater.

Finally, you should consider customer perception and expectations. Do customers expect that, when they are told they are making a secure encrypted connection, the secure tunnel extends all the way from client to server? That would be a logical assumption for the typical user, who isn't aware of technologies such as SSL offloading. What liability issues would you be subject to if confidential customer data were accessed by unauthorized persons while traveling over the network in a decrypted state, and then misused?

Summary

SSL encryption presents a dilemma wherein performance and security are at opposite ends of a continuum and more of one results in less of the other. Ultimately, you must evaluate your own network structure, the nature of the data that travels over it, and how much of a performance tradeoff is worth extra security or vice versa. The purpose of this article is to make you aware that once again the old maxim holds true: TANSTAAFL ("There Ain't No Such Thing As A Free Lunch"). While SSL offloading offers distinct advantages to your business, those advantages come with a price.

If you would like us to email you when Deb Shinder releases another article on WindowSecurity.com, subscribe to our 'Real-Time Article Update' by clicking here. Please note that we do NOT sell or rent the email addresses belonging to our subscribers; we respect your privacy!

DEBRA LITTLEJOHN SHINDER, MCSE, is a technology consultant, trainer and writer who has authored a number of books on computer operating systems, networking, and security. These include *Scene of the Cybercrime: Computer Forensics Handbook,* published by Syngress, and *Computer Networking Essentials,* published by Cisco Press. She is co-author, with her husband, Dr. Thomas Shinder, of *Troubleshooting Windows 2000 TCP/IP,* the best-selling *Configuring ISA Server 2000,* and *ISA Server and Beyond.* Deb is also a tech editor, developmental editor and contributor to over 15 books on subjects such as the Windows 2000 and Windows 2003 MCSE exams, CompTIA Security+ exam and TruSecure's ICSA certification. She formerly edited the Brainbuzz A+ Hardware News and currently edits Sunbelt Software's WinXP News, and is regularly published in TechRepublic's TechProGuild and Windowsecurity.com. Deb currently specializes in security issues and Microsoft products and writes marketing material for Microsoft Corp. She lives and works in the Dallas-Ft Worth area and occasionally teaches computer networking and security courses at Eastfield College (Mesquite, TX). Her personal web site is at www.shinder.net.