Cryptosystems that Secure Web Browsers

E. Craig Luther

Introduction

The need to secure Web browsers from eavesdropping of unauthorized parties or stealing of data pushed the development of cryptosystems that work to secure Web browsers. This was of particular importance for users trust of any Web based business such as electronic commerce sites as internet users wouldn't purchase goods over the web unless the technology existed for transmitting data securely via a Web browser. [1] There are numerous cryptosystems that provide this technology, including Secure Socket Layer (SSL), Secure Electronic Transaction (SET), Secure Hypertext Transfer Protocol (SHTTP), Secure Shell (SSH) and IP Security (IPSec).

Secure Socket Layer (SSL)

Netscape Communications was the first to introduce a communications protocol for securing Web transactions back in the summer of 1994. Secure Socket Layer, or commonly known as SSL, outlined a method of authenticating and encrypting communication between clients and servers. The first version of the protocol was designed for use in the Mosaic browser and later the 2nd version was integrated into the original Netscape Navigator Web browser near the end of 1994. Microsoft was developing it's own web browser and released it's first version of Internet Explorer less then a year later in 1995 and soon after released Private Communication Technology which improved upon some of the weaknesses in the 2nd version of SSL Netscape Communications had released. [1] Finally in May of 1996, the Internet Engineering Task Force (IETF) agreed to make SSL an international standard. Today Secure Socket Layer is the most widely used protocol to secure data transactions over the web.

SSL can best be described as a handshake protocol and is designed to negotiate encryption keys and to authenticate the server before data is exchanged. The integrity of the transmission channel is maintained using encryption, authentication and MACs. [2] It runs above TCP/IP and below high-level protocols such as HTTP or IMAP and in the process allows an SLL-enabled server to authenticate itself to an SSL-enabled client, allows the client to authenticate itself to the server, and allows both machines to establish an encrypted connection. SSL server authentication allows a user to confirm a server's identity, SSL client authentication allows a server to confirm a users identity and encrypted SSL connection requires all information sent between a client and a server to be encrypted by the sending software and decrypted by the receiving software. All of this provides a high degree of confidentiality and security. [3]

Secure Electronic Transactions (SET)

In 1996, Visa and MasterCard (including Microsoft, IBM, Netscape and others) announced the development of a technical standard for securing credit card purchases made over open networks, such as the Internet. This standard was called the Secure Electronic Transaction, or SET. In mid December 1997, Visa and MasterCard created a corporate entity that would direct the future development of the SET protocol. [4]

Secure Electronic Transaction, or SET, is an open encryption and security specification. SET is a set of security protocols and formats that allowed the existing credit card payment infrastructure to operate on an open network, like the Internet, in a secure manner. SET provides privacy, integrity and authentication in its protocol. Privacy is maintained via cryptography, which makes intercepted messages unreadable. Two forms of cryptography are used, RSA and DES. RSA is used for signatures and public key encryption of symmetric encryption keys and bankcard numbers while DES encrypts the data that is going to be transmitted during the transaction. Hashing and signing, ensure integrity by making sure messages sent are received without being altered. A hashing algorithm is a function that calculates a unique integrity value from the original message. Finally, through digital certificates, authentication assures that the parties involved in the transaction are indeed who they say they are. A trusted 3rd party known as a Certification Authority, which gives assurances that the identity of the certificate holder is legitimate, issues the certificates. Each of the digital certificates has owner identification information and a copy of one of the owner's public keys. [4]

Secure Electronic Transaction is very secure, and is one of the strongest encryption applications for public use and is designed to be used with 1,024-bit cipher keys. To put that in perspective, it would take roughly 2,800,000,000,000 years (give or take a few), using 100 computers each able to process 10,000,000 instructions per second to break the encryption. SET can be used for export from the US, but only in financial transactions and cannot be used to pass secret or sensitive information to anyone outside the US. [4]

Secure HTTP

E. Rescorla and A. Schiffman of Enterprise Integration Technologies developed this protocol in 1994. SHTTP is simply an extension of HTTP, which you may recognize as the protocol of the World Wide Web. S-HTTP is related to SSL as it is created by SSL running under HTTP. The securing of end-to-end transactions is accomplished by S-HTTP adding cryptography to messages at the application layer. SSL, which operates at the transport layer, encrypts the entire communication between client and server, while S-HTTP encrypts each message on an individual basis. [6]

HTTP message, sender's cryptographic preferences, and receiver's preferences make up the 3 parts of an SHTTP message. The sender will integrate both preferences to the message, which in turn results in multiple cryptographic enhancements applied to the message. When the recipient attempts to decrypt the message, the message headers are evaluated to determine which cryptographic methods were used during the encryption process. Once this is determined a combination of the cryptographic preferences outlined earlier from both the sender and recipient is used to decrypt the message. Secure transactions can occur at anytime without the need of a key, as SHTTP does not require the client to possess a public key certificate. [6]

Secure Shell

Secure Shell, or SSH, was developed by SSH Communications Security Ltd. SSH is a program to log into another computer over a network, execute commands in a remote machine and to move files from one machine to another using tunneling. [7] SSH software is made up of a SSH server and client and there are two versions, SSH1 and SSH2. The secure channel is established first by a protocol version exchange phase, then a key exchange phase and finally a user authentication phase between the SSH Server and client (in both versions). SSH uses strong authentication and secure communications over insecure channels. The secure channel is considered to be guaranteed that a user can communicate with the SSH server secretly. [8]

During the key exchange phase, the SSH Server and client exchange a session key that they use to encrypt the communication channel. The SSH server then, will authenticate the SSH client with the session key over the secure channel. Since the exchange of the session key occurs before user authentication there is a slight possibility that a malicious client could trick the server. [8]

SSH uses for types of keys. The **Host Key** is considered a public key and all SSH servers will have one. It is used to verify the client is really actually talking to the intended server, so the client has to have prior knowledge of the server's host key. The **Server Key** is a temporary public key that is only used in SSH1. Its purpose is to protect the Session Key (described below) and is created on a defined regular interval. The **User Key** is used by the SSH server to authenticate the public key and both versions of SSH must support the public key authentication. The **Session Key** is symmetric (secret key) and is generated each session and is used to encrypt the communication between the client and server. [8]

IPSec

IPSec, or Internet Protocol Security, is a set of protocols developed by the Internet Engineering Task Force. IPSec provides cryptographic-based protection mechanisms for IP packets. The protection mechanisms include packet confidentiality, packet integrity, packet origin authentication and protection against replay. Packet confidentiality means the packets are encrypted before they are sent over the network. This keeps any unauthorized people from reading them. If any of the packets are altered during the data transmission this will be detected which provides a level of integrity. The IP address of the sender is included in the source address of the IP header within the packet thereby ensuring they are from the sender who claims them. Finally the packets are protected from being captured and then resent at a later time. [9]

IPSec can be implemented on routers, hosts, gateways or basically anywhere a secure IP connection is required. It is used to create Virtual Private Networks or VPNs. A VPN is a network within a network, which allows users to set up a tunnel and send encrypted data back and forth using an IP-packet-within-an-IP-packet method. The IPSec is made up of the protocol, which defines how the data is encrypted and what information to add, and the Internet Key Exchange. The key exchange negotiates the security associations using an asymmetric-based key exchange. [10]

Conclusion

As the ongoing trend for businesses to conduct more and more of their critical business online continues, the need for cryptosystems to secure internet based transactions will continually evolve. Organizations must take this task seriously and determine which of the particular security protocols described in this paper best meet their needs for a particular task at hand. If an e-business runs a website selling computer parts they would obviously need to have data security and authentication for transactions, such as SSL, while at the same time allowing users easy access to their site. On the other hand perhaps an IPSec solution might make more sense when the task is to provide for strict access control to protect critical data. In any event an organization must implement the right protocol to meet the needs of the business. [11]

References

[1] Rainbow Technologies Inc.; “The Secure Sockets Layer Protocol – Enabling Secure Web Transactions” February 3, 2002; URL: http://www.itsecurity.com/papers/rainbow3.htm

[2] Mactaggar, Murdoch; “Introduction to cryptography, Part 4: Cryptography on the Internet” March 1, 2001; URL: http://www-106.ibm.com/developerworks/library/s-crypt04.html

[3] Introduction to SSL; URL: http://docs.sun.com/source/816-6156-10/contents.htm

[4] Wolrath, Carl Eric; “Secure Electronic Transaction: a market survey and a test implementation of SET technology” September 27, 1998; URL: http://www.wolrath.com/set.html#3_Secure_Electronic_Transaction

[5] Stallings, William; “Introduction to Secure Electronic Transaction (SET)” May 17, 2002; URL: http://www.informit.com/articles/article.asp?p=26857

[6] Karve, Anita; “SSL and SHTTP – Secure Communication over the Internet” January 1, 1997; URL: http://www.network-mag.com/article/NMG20000727S0002

[7] FAQs; URL: http://www.onsight.com/faq/ssh/ssh-faq.html

[8] Saito, T.; Kito, T.; Umesawa, K.; Mizoguchi, F.; “Architectural Defects of the Secure Shell”; Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on, Vol., Iss., 2-6 Sept. 2002 Pages: 22- 28

[9] Li, Man; “Policy-based IPsec Management”; Network, IEEE, Vol.17, Iss.6, Nov.-Dec. 2003 Pages: 36- 43

[10] Whitman, Michael E.; Mattord, Herbert J.; “Principles of Information Security”; 2003 Course Technology

[11] Chou, W.; “Inside SSL: The Secure Sockets Layer Protocol”; IT Professional, Vol.4, Iss.4, Jul/Aug 2002 Pages: 47- 52