Available at: www.researchpublications.org

An Overview of Secure Sockets Layer

Ms.Dipti S.Charjan Ms.Priyanka S. Bochare Yogesh R.Bhuyar
Comp. Science & Engg. Department *Information Technology Department* *Information Technology Department*
P.R.M.I.T.R. Badnera,India *P.R.M.I.T.R Badnera,India* *P.R.M.I.T.R Badnera,India*
Address Address *Address*
charjandipu@gmail.compriyanka.bochare@gmail.comyogbhuyar@gmail.com

***Abstract*—In recent years, there has been tremendous development in Internet. Right from retrieving information about any subject, you can use Internet for many purposes. But when two parties like client and a server are communicating over the Internet, an attacker may interface the communication. Netscape came with 'SSL' i.e. Secure Socket Layer, to avoid this problem and to make the communication over the Internet secure. Today 'SSL' is widely used over the Internet for the secure communication.**

secure sockets layer (SSL) security protocol is largely used nowadays by the World Wide Web to secure Internet communications and an increasing number of handheld devices are provided by the manufacturers with preinstalled applications (e.g. Web browsers) and digital certificates in order to support SSL in wireless environments too. We present first a comprehensive analysis of SSL protocol and the cryptographic algorithms used as building blocks in the protocol. [2]

We demonstrate next that SSL support embedded also in other applications for some mobile devices does not impact significantly on overall application's performance when communicating with other parties. We wanted to use for our experimental setup, APIs and tools that are either available for the majority of the handheld devices or that can be downloaded and tested freely and without major modifications. This is the especially the case of application developers, who preferably would not use modified versions of cryptographic libraries and special drivers characteristic to academic/commercial testing environments. We also developed a tool named SSL performance for testing SSL security protocol's performance when transferring either small amounts of data (e.g. credit card number) or of medium dimension (e.g. a document in PDF format), with handheld devices. We've compared the results obtained with SSLperf on Windows CE-enabled handheld and pocket PCs and 'powerful' Windows 2000/XP-enabled platforms with other related works. [1]

***Keywords*— Secure sockets layer, hypertext transfer protocol,Transport Layer Security, Transmission Control Protocol/Internet Protocol, Internet Messaging Access Protocol.**

I. INTRODUCTION

SSL stands for Secure Sockets Layer protocol developed by Netscape and is the standard Internet protocol for secure communications. The secure hypertext transfer protocol (HTTPS) is a communications protocol designed to transfer encrypted information between computers over the World Wide Web. HTTPS is http using a Secure Socket Layer (SSL). A secure socket layer is an encryption protocol invoked on a Web server that uses HTTPS. SSL is a type of sockets communication and resides between TCP/IP and upper layer applications, requiring no changes to the application layer SSL is used typically between server and client to secure the connection. A common TCP/IP sockets call is substituted for a call to SSL sockets and a variety of application programming interfaces (APIs) are offered.

This approach of "plugging in" security at the socket layer can significantly reduce development time in contrast to building and incorrect Secure Socket Layer Regardless of where you access the Internet from, the connection between your Web browser and any other point can be routed through dozens of independent systems. Through snooping, spoofing, and other forms of Internet eavesdropping, unauthorized people can steal credit card numbers, PIN numbers, personal data, and other confidential information. The Secure Sockets Layer (SSL) protocol was developed to transfer information privately and securely across the Internet.

SSL is layered beneath application protocols such as HTTP, SMTP, and FTP and above the connection protocol TCP/IP. It is used by the HTTPS access method. Figure 1 illustrates the difference between a non-secure HTTP request and a secure SSL request. Transport Layer Security (TLS) is the successor of Secure Sockets Layer (SSL); they are both cryptographic protocols that provide secure communications on the Internet for such things as web browsing, e-mail, Internet faxing, instant messaging, and other data transfers. There are slight differences between SSL and TLS, but the protocol remains substantially the same.[5]

II.LITERATURE REVIEW

SSL (Secure Socket Layer) is a protocol developed by Netscape that enables a web browser and a web server to communicate securely; it allows the web browser to authenticate the web server. The SSL protocol requires the web server to have a digital certificate installed on it in order for an SSL connection to be made. SSL works by using a public key to encrypt data that's transferred over the SSL connection. Both Netscape Navigator and Internet Explorer support SSL, and many websites use the protocol to obtain confidential user information, such as credit card numbers. By convention, URLs that require an SSL connection start with https: instead of http. Secure Sockets Layer, or SSL, technology is one method for protecting Internet users from malicious groups or software. It is currently used in Web browsers, instant messaging programs, email clients and other software. You most likely use, or have used, severalapplications that depend on SSL to secure communications between you and another computer that you are connecting to, such as an email server. The Netscape Company created the SSL protocol in 1994. This technology allowed secure transmissions between computer applications on a remote server and the client's computer; however, it was never released to the public domain. SSL "provides privacy and □ Netscape continued to develop SSL technology for several years.

The first version of SSL was never released because of problems regarding protection of credit card transactions on the Web. In 1994, Netscape created SSLv2, which made it. possible to keep credit card numbers confidential and also authenticate the Web server with the use of encryption and digital certificates. In 1995, Netscape strengthened the cryptographic algorithms and resolved many of the security problems in SSLv2 with the release of SSLv3. SSLv3 now support more security algorithms than SSLv2.[4]

A.*SSL BASICS*

The main role of SSL is to provide security for Web traffic. Security includes confidentiality, message integrity, and authentication. SSL achieves these elements of security through the use of cryptography, digital signatures, and certificates.

1) Cryptography

SSL protects confidential information through the use of cryptography. Sensitive data is encrypted across public networks to achieve a level of confidentiality. There are two types of data encryption: symmetric cryptography and asymmetric cryptography.

SSL protects confidential information using cryptography. Sensitive data is encrypted across public networks to achieve a high level of confidentiality. Primarily, PKI utilizes asymmetric cryptography that is considered more secure than symmetric cryptography.

Simply, asymmetric algorithms use one key for encryption of data, and then a separate key for decryption. Asymmetric algorithms are stronger than symmetric algorithms because even if the encryption key is learned in one direction, the third party still needs to know the other key in order to decrypt the message in the other direction.

The primary benefit of asymmetric encryption is that both sides can spontaneously initiate a transaction without ever having met. This is achieved by the use of a public and private key pair. The public key of the entity is public knowledge and is used for encryption, whereas the private key of the entity remains secret and is used for decryption. Although PKI is more secure, it also is more expensive in terms of processing speed and encryption/ decryption (in PKI) can take up to 1000 times the processing than symmetric cryptography.

SSL Crypto Algorithms: SSL supports a variety of different cryptographic algorithms, or ciphers, that it uses for authentication, transmission of certificates, and establishing session keys. SSL-enabled devices can be configured to support different sets of ciphers, called cipher suites. If an SSL-enabled client and an SSL-enabled server support multiple cipher suites, the client and servernegotiate which cipher suites they use to provide the strongest possible security supported by both parties. A cipher suite specifies and controls the various cryptographic algorithms used during the SSL handshake and the data transfer phases.

There are two categories of cryptographic algorithms: conventional and public key.

Conventional cryptography: also known as symmetric cryptography requires the sender and receiver to share a key: a secret piece of information that may be used to encrypt or decrypt a message. As long as this key is kept secret, nobody other than the sender or recipient can read the message. If Alice and the bank know a secret key, then they can send each other private messages. The task of sharing a key between sender and recipient before communicating, while also keeping it secret from others, can be problematic.

2. Public key cryptography: also known as asymmetric cryptography solves the key exchange problem by defining an algorithm which uses two keys, each of which may be used to encrypt a message. If one key is used to encrypt a message then the other must be used to decrypt it. This makes it possible to receive secure messages by simply publishing one key (the public key) and keeping the other secret (the private key).Anyone can encrypt a message using the public key, but only the owner of the private key will be able to read it. In this way, for ex. Alice can send private messages to the owner of a key-pair (the bank), by encrypting them using their public key. Only the bank will be able to decrypt them.[7]

B.DIGITAL SIGNATURES

To ensure message integrity, each message exchanged in SSL has a digital signature attached to it. A digital signature is a hashed message digest with public key information. The message digest is based on the checksum of the message. The message digest is difficult to reverse. Both parties compute the message digest separately and compare the hashed results. Matching results means that the checksum was unaltered during transit, minimizing the chance of a compromised message (refer to Figure 1). Digital signatures are created by encrypting a digest of the message and other information (such as a sequence number) with the sender's private key. Though anyone can decrypt the signature using the public key, only the sender knows the private key. This means that only the sender can have signed the message. Including the digest in the signature means the signature is only good for that message; it also ensures the integrity of the message since no one can change the digest and still sign it.To guard against interception and reuse of the signature by an intruder at a later date, the signature contains a unique sequence number.

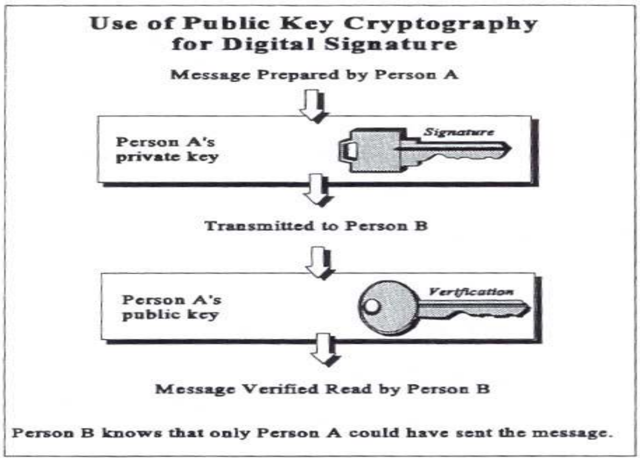
Use of Public Key Cryptography for Digital Signature
Message Prepared by Person A
Person A's private key
Signature
Transmitted to Person B
Person A's public key
Verification
Message Verified Read by Person B
Person B knows that only Person A could have sent the message.

Fig.1 Use of public key cryptography for digital signature

1. Client sends a message
2. Client has message and a public key
3. Client hashes message with public key
4. Server takes random message and knows public key
5. Server hashes message with public key
6. Server sends hashed message
7. Client compares its own hashed message to server's message
8. If the two match, then the message has not been tampered

C.SSL Certificate

SSL certificates becomes the "passport" or the digital document that verify that the security and authenticity of the interaction. The SSL certificate is installed on a web server to identify the business using it to encrypt sensitive data such as credit card information. SSL Certificates give a website the ability to communicate securely with its web customers. Without a certificate, any information sent from a user's computer to a website can be intercepted and viewed by hackers and fraudsters. It is similar to the difference between sending a post card and a tamper proof sealed envelope.

SSL Certificate interaction with the Browser and the Server

- Browser checks the certificate to make sure that the site you are connecting to is the real site and not someone intercepting.
- Determine encryption types that the browser and web site server can both use to understand each other.
- Browser and Server send each other unique codes to use when scrambling (or encrypting) the information that will be sent.
- The browser and server start talking using the encryption, the web browser shows the encrypting icon, and web pages are processed secured. Interaction Between Web Server and Web Browser.

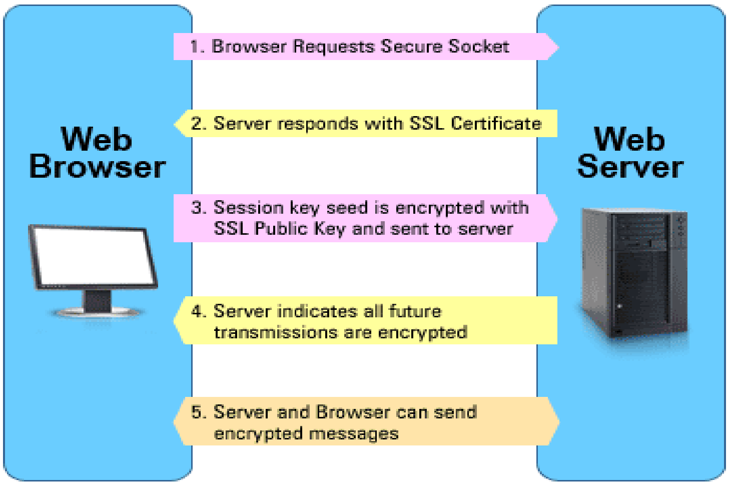
Web Browser
Web Server
1. Browser Requests Secure Socket
2. Server responds with SSL Certificate
3. Session key seed is encrypted with SSL Public Key and sent to server
4. Server indicates all future transmissions are encrypted
5. Server and Browser can send encrypted messages

Fig. 2SSL certification

D.SSL Protocol

The Transmission Control Protocol/Internet Protocol (TCP/IP) governs the transport and routing of data over the Internet. Other protocols, such as the Hypertext Transport Protocol (HTTP), Lightweight Directory Access Protocol (LDAP), or Internet Messaging Access Protocol (IMAP), run "on top of" TCP/IP in the sense that they all use TCP/IP to support typical application tasks such as displaying web pages or running email servers. Figure 3, shows SSL runs above TCP/IP and below high-level application protocols. The SSL protocol includes two sub-protocols: the SSL record protocol and the SSL handshake protocol. The SSL record pro tocol defines the format used to transmit data. The SSL handshake protocol involves using the SSL record protocol to exchange a series of messages between an SSL-enabled server and an SSL-enabled client when they first establishan SSL connection. This exchange of messages is designed to facilitate the following actions:

- Authenticate the server to the client.
- Allow the client and server to select the cryptographic algorithms, or ciphers, that they both support.
- Optionally authenticate the client to the server.
- Use public-key encryption techniques to generate shared secrets.
- Establish an encrypted SSL connection.

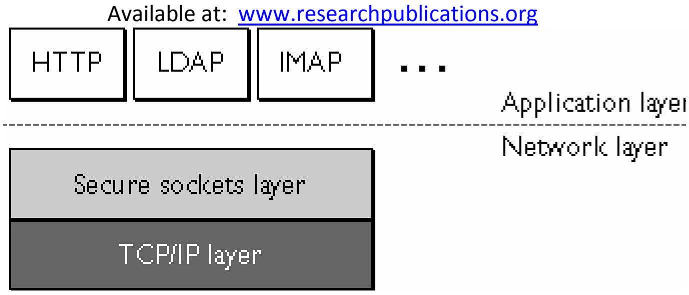

Fig.3 Location of secure socket layer

The SSL protocol supports the use of a variety of different cryptographic algorithms, or ciphers, for use in operations such as authenticating the server and client to each other, transmitting certificates, and establishing session keys. Clients and servers may support different cipher suites, or sets of ciphers, depending on factors such as the version of SSL they support, company policies regarding acceptable encryption strength, and government restrictions on export of SSL-enabled software. Among its other functions, the SSL handshake protocol determines how the server and client negotiate which cipher suites they will use to authenticate each other, to transmit certificates, and to establish session keys.[3]

III. WORKING OF SSL

When a client and server communicate, SSL ensures that the connection is private and secure by providing authentication, encryption, and integrity checks. Authentication confirms that the server, and optionally the client, is who they say they are. Encryption through a key-exchange then creates a secure “tunnel” between the two that prevents any unauthorized system from reading the data. Integrity checks guarantee that any unauthorized system cannot modify the encrypted stream without being detected.

SSL-enabled clients (such as a Mozilla™ or Microsoft Internet Explorer™ web browser) and SSL-enabled servers confirm each other’s identities using digital certificates. Digital certificates are issued by trusted third parties called Certificate Authorities (CAs) and provide information about an individual’s claimed identity, as well as their public key. Public keys are a component of public-key cryptographic systems. The sender of a message uses a public key to encrypt data. The recipient of the message can only decrypt the data with the corresponding private key. Public keys are known to everybody; private keys are secret and only known to the owner of the certificate. By validating the CA digital signature on the certificates, both parties can ensure that an imposter has not intercepted the transmission and provided a false public key for which they have the correct private key. SSL uses both public-key and symmetric key encryption. Symmetric key encryption is much faster than public-key encryption, but public-key encryption provides better authentication techniques. So SSL uses public key cryptography for authentication and for exchanging the symmetric keys that are used later for bulk data encryption. The secure tunnel that SSL creates is an encrypted connection that ensures that all information sent between an SSL-enabled client and an SSL-enabled server remains private. SSL also provides a mechanism for detecting if someone has altered the data in transit. This is done with the help of message integrity checks. These message integrity checks ensure that the connection is reliable. If, at any point during a transmission, SSL detects that a connection is not secure, it terminates the connection and the client and server establish a new secure connection. From a high-level point of view, the process of authenticating and establishing an encrypted channel using certificate-based mutual authentication involves the following steps:

- A client requests access to a protected resource.
- The server presents its certificate to the client.
- The client verifies the server’s certificate.
- If successful, the client sends its certificate to the server.
- The server verifies the client’s credentials.
- If successful, the server grants access to the protected resource requested by the client.

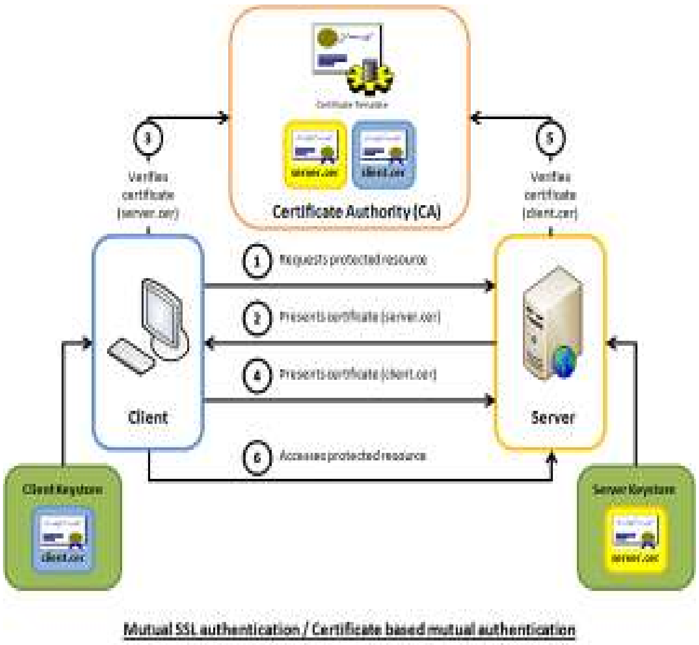
Certificate Authority (CA)
server.cer
client.cer
3
Verifies certificate (server.cer)
5
Verifies certificate (client.cer)
Client
Server
1 Requests protected resource
2 Presents certificate (server.cer)
4 Presents certificate (client.cer)
6 Accesses protected resource
Client Keystore
client.cer
Server Keystore
server.cer
Mutual SSL authentication / Certificate based mutual authentication

Fig.4 Mutual SSLauthentication/certificate based mutual authentication

IV. SSL AND THE OSI MODEL

To use ISA server to request a secure SSL (Secure Sockets Layer) channel between the client and the ISA server when making and sustaining the connection. SSL is one of the specialized methods used by websites and firewalls to do authentication. The SSL protocol is a security protocol that sits on top of TCP at the transport layer. In the OSI model, application layer protocols such as HTTP or IMAP, handle user application tasks such as displaying web pages or running email servers. Session layer protocols establish and maintain communications channels. Transport layer protocols such as

TCP and UDP,handle the flow of data between two hosts. Network layer protocols such as IP and ICMP provide hop-by-hop handling of data packets across the network. SSL operates independently and transparently of other protocols so it works with any application layer and any transport layer protocol. This allows clients and servers to establish secure SSL connections without requiring knowledge of the oth party's code.

An application layer protocol hands unencrypted data to the session/transport layer, SSL encrypts the data and hands it down through the layers. When the server receives the data atthe other end, it passes it up through the layers to the session layer where SSL decrypts it and hands it off to the application layer. Since the client and the server have gone through the key negotiation handshake, the symmetric key used by SSL is the same at both ends. [3] [4] [5]

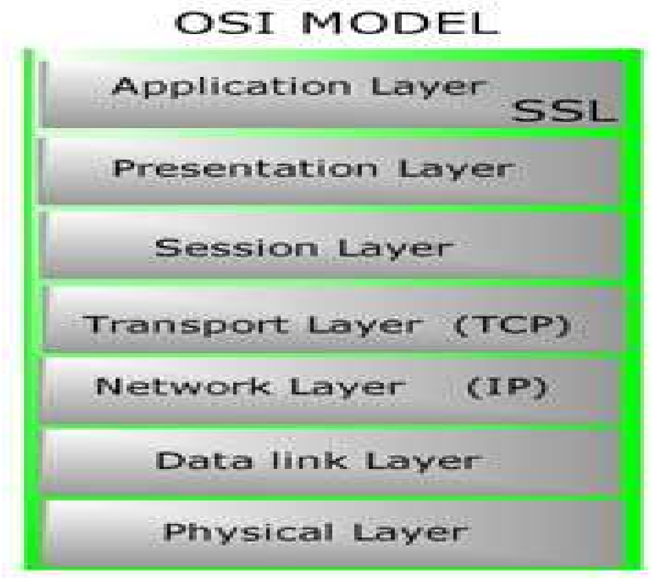

Fig:5 SSL functions at the Application layer of the OSI model

Facts about SSL that we should know

- SSL runs above TCP/IP and below high-level application protocols
- SSL allows a user to confirm a server's identity
- SSL allows a server to confirm a user's identity
- A SSL connection requires all information sent between a client and a server to be encrypted
- Data sent over an SSL connection is protected with a mechanism for detecting tampering. Any suspicion of data tampering requires a retransmit.
- SSL uses Algorithms and ciphers to perform encryption
- SSL come in two flavours 128 bit encryption and 40 bit encryption In some countries 128 bit is illegal.

V.USES OF SSL

SSL is the de facto standard for encrypted and authenticated communications between clients and servers on the Internet. Virtually all online purchases and browser-based monetary transactions that occur on the Internet are secured by SSL. However, SSL is not just limited to securing e-commerce transactions; the following are a few other examples of SSL use:

Financial institutions implement SSL to secure the transmission of PIN numbers and other confidential account information.

Insurance companies implement SSL to secure transmission of confidential policy information.

Organizations who have established Business-to-Business (B2B) extranets implement SSL to secure transactions between the company and its partners, suppliers, and customers.

Private organizations implement SSL in their intranets to confidentially transfer information to and from employees.

Email providers implement SSL to secure webmail for users

VI.SSL BENEFITS

A customer connecting to a secure website is assured of three things:

1.Authentication: The company that installed the certificate really owns the website.

2.Message privacy: Using a unique "session key", SSL encrypts all information exchanged between your web server and your customers, such as credit card numbers and other personal data. This ensures that personal information cannot be viewed if it is intercepted by unauthorized parties.

3.Message integrity: The data cannot be tampered with over the Internet.

4.Increasing Business: Certificates let you securely exchange sensitive information online and increase business by giving your customers confidence that their transactions are safe. [1]

VII.CONCLUSIONS

SSL is vital to Web security. It provides a strong sense of confidentiality, message integrity, and server authentication to users. The business of e-commerce is tied closely to consumer confidence in the operation of SSL across the net. In the future, SSL termination devices will be able to handle more transactions at a faster rate. The encryption of key lengths and the cipher suites used will also continue to evolve in order to ensure the security of sensitive information over the Web. This way, e-commerce will be able to continue to grow in popularity as users grow more confidants in shopping and banking online, and embracing new online applications. Our future work involves investigating the design and performance of architectural support for security protocols further.

REFERENCES

[1] A. Freier, P. Karlton, P. Kocher "The Secure Sockets Layer (SSL) Protocol Version 3.0", August 2011.

[2]Thomas Y. C. Woo, Raghuram Bindignavle, Shaowen Su and Simon S. Lam, "*SNP: An interface for secure network programming* Proceedings USENIX Summer Technical Conference", June 1994

[3]T. Dierks, E. Rescorla "The Transport Layer Security (TLS) Protocol, Version 1.2", August 2008.

[4] Steve Lloyd et al, *Understanding Certificate Path Construction*, 2002, PKI Forum, 2002,

[5] Krishna Kant, Ravi Iyer and Prasant Mahapatra, " Architectural Impact of Secure Socket Layer on Internet Servers", Sep. 2000

[6] Ruby B. Lee, Zhijie Shi and Xiao Yang, "Efficient Permutation Instructions for Fast Software Cryptography", IEEE Micro,December 2001

[7] A.J. Menezes, P.C. Van Oorschot, et al., "Handbook of Applied Cryptography", Oct. 1996