Three attacks in SSL protocol and their solutions

Hong lei Zhang
Department of Computer Science
The University of Auckland
zhon003@ec.auckland.ac.nz

Abstract

Secure Socket Layer (SSL) and Transport Layer Security (TLS) is the protocol above TCP, which can protect user's privacy when they sending data from a client side to a web server, this is an important protocol due to the expansion of Internet. In fact, it is a long way to make the SSL/TLS protocol perfectly. However there are still shortcomings and problems during the development of SSL/TLS, and we cannot deny that there maybe some other potential security hole in the latest version. Successive attack is fatal for both the user and the company in using these protocols to establish a safe channel to transfer information. This article will introduce three typical attacks: Cipher suite rollback attack, version rollback attack and password interception in SSL/TLS channel.

1. Introduction

As the Internet and World Wide Web become popular, it is important to consider the system security. This is because the plaintext flowing through the Internet is unencrypted, it is for cracker or hacker, even a user without any programming knowledge, to intercept the message and modify it. So, How to protect personal privacy? How to ensure a safe online commerce? etc. These are the challenge for Information Technology.

SSL/TLS can set up a valid secure channel between server and client which can encode the plaintext, then the third party who intercept the message can not disclose the original message without decode it.

SSL consist of two phases: handshake and data transfer. During the handshake process, the client and server use a public-key encryption algorithm to determine secret-key parameters, during the data transfer process, both sides use the secret key to encrypt and decrypt successive data transmissions [1].

There are potential dangers both during handshake and data transfer state, although the latest TLS have fixed several secure hole of the old version, the successive attack in practice is not only a terrible thing for the user who trusts the SSL/TLS, but also a challenge for software security area. This article will introduce the birth and development of SSL in section 2 and give some background of SSL/TLS in section 3.Within the basic knowledge of the protocol, Section 4 will present some comment on an attack during data transfer in practice, followed by two attacks of the SSL handshake protocol from comparing old version SSL2.0 and new version SSL3.0. Finally, Section 5 concludes the strengths and weaknesses of SSL protocol.

2. The history of SSL/TLS

2.1 Birth of SSL

When Netscape developed their web browser, they discovered that during the transaction between web server and web browser, there should be some security method to protect the message alternated between client and server, especially for some commercial poppers. At first, Netscape established some encryption in their applications. However, they found that the encryption did not support non-HTTP applications. SSL (Secure Sockets Layer) protocol was developed which is just above the TCP (Transmission Control Protocol) to create this security.

2.2 Development of SSL/TLS

As other companies such as Microsoft began to develop their own transport security protocols, the Internet Engineering Task Force (IETF) intervened to define a standard for an encryption-layer protocol. [1] With the input of multiple vendors, the IETF created Transport Layer Security standard. In fact, TLS is a protocol based on SSL 3.0, some people believe TLS1.0 is the same as SSL3.1.Although there are some implement differences between TLS and SSL, application developer and user cannot detect any differences at all.

During the last few years, SSL developed rapidly, because there is always some secure holes and some previous versions of SSL cannot prevent attacker to eavesdropping or

intercept properly. For instance, Version 3.0 of SSL was designed to correct a flaw in previous versions 2.0, the cipher-suite rollback attack, whereby an intruder could get the parties to adopt a weak cryptosystem [2].

This article will put emphasis on comparing SSL version 2.0 and SSL version 3.0.

There are several security flaws in SSL 2.0[2]

1. SSL 2.0 unnecessarily weakens the authentication keys to 40 bits.
2. SSL2.0 has a weak MAC construction.
3. SSL2.0 puts padding bytes into the MAC in block cipher modes. But the padding length field are not authenticated, which may allow active attackers to selectively delete bytes from the end of messages.
4. During a cipher suite rollback attack, an active attacker edits the list of cipher suite preferences in the hello messages to force both endpoints to use a weaker form of encryption than they would choose.

Changes in SSL 3.0:

1. SSL 3.0 use 128 bits authentication keys.
2. SSL3.0 use stronger HMAC construction, a simple, fast hash-based construction with strong theoretical evidence for its security
3. Change the sequence of MAC and padding.
4. The change in SSL 3.0 will be introduced in section 4.

3. Basic knowledge and technology in SSL protocol

3.1 public key and private key

3.1.1What is public key and private key

The association of public key (sometimes called asymmetric key) and private key (also known as symmetric key or secret key) can ensure the safe transaction between Web server and web browser, the main function of this pair of key is encrypting and decrypting information. Private Key, which saves in web server, will decrypt the information being sent from web browser. However, web browser encrypts its information within public key.

3.1.2 public-key encryption

The sender uses public key to encrypt the message, and the receiver uses private key to decrypt it. So every public key owner can send his or her encoded message and only a private key owner can read the message within decoded the message.

However decrypting a message, which encrypted with public key, is a CPU intensive, such as RSA (Rivest-Shamir-Adelman)

Another usage of public key is the sender encodes the message within secret key, and the receiver use the private key to decode that message with the associated public key. This is efficient for message authentication, such as a bank server sending their digital signature encrypted with private key, and then any customer can decrypt the message within private key and verify the message.

3.1.3 What kind of key SSL use

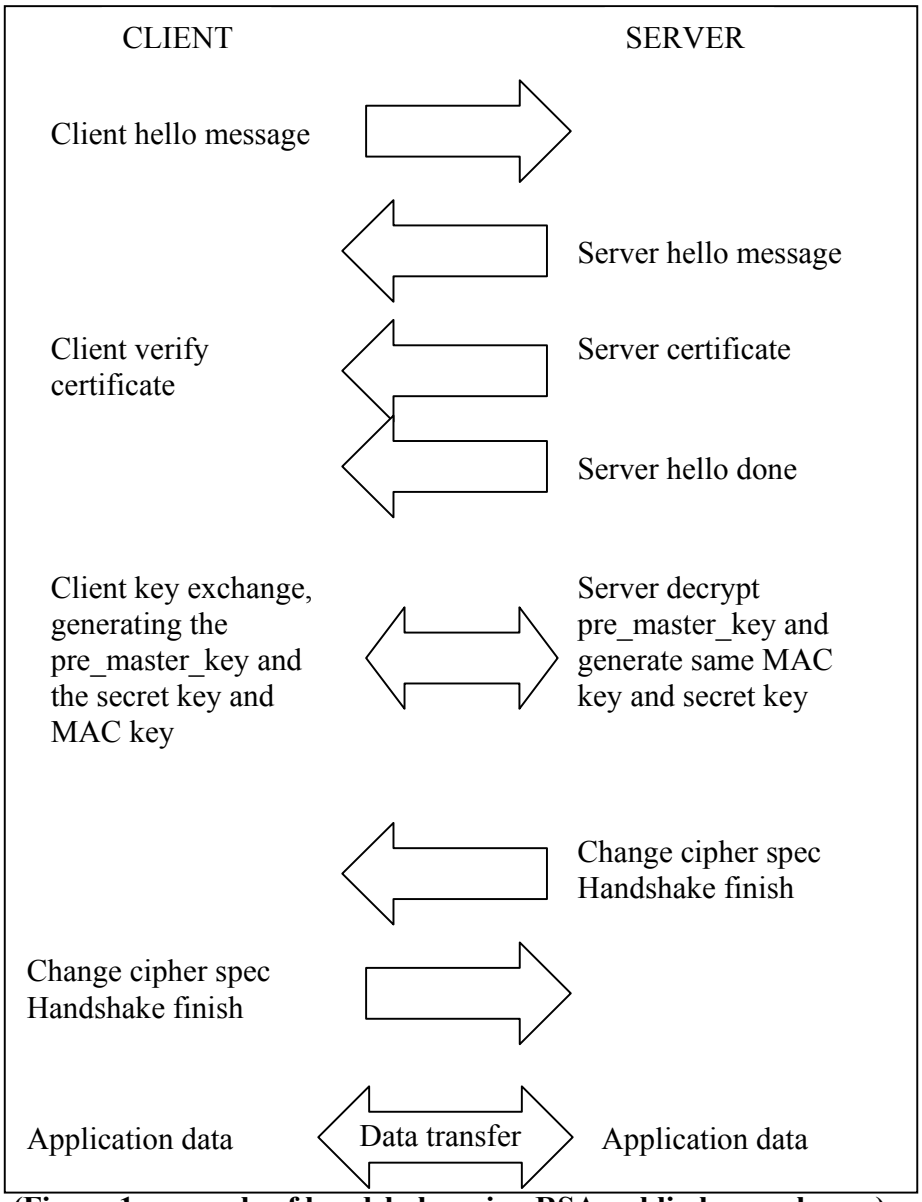
"Although SSL supports the Diffie-Hellman protocol, the majority of SSL transactions do not use this public-key agreement approach. Instead, they use the RSA public key algorithm to distribute secret-key parameters" [1]

3.2 How SSL secure a transaction

Handshake

1. Client Hello: client side send hello message including a list of cipher suite which client supported, in this step client also create a random number: ClientHello.random.
2. Server Hello: server sends response of client hello message. Server will select a cipher suite, generate random number: ServerHello.random and session id.
3. Certificate Server sending certificate then client verifies it. Followed by server hello done.
4. Client generates another random number: pre_master_secret (encrypted with server's public key), then produce a master_key (structure in [3]).The master_key and two random number generated during the Hello procedure are used to create the secret key and MAC key.
5. Server decrypt the pre_master_key transmitted from client and generate a same master key as client.

6. Change cipher specification: send by client then client copies the pending cipher spec into the current cipher spec. at this point, client sends the finished message.
7. At same time, server is ready to transmit data encrypted with created secret key and also send a handshake finished message to client.



(Figure 1: example of handshake using RSA public-key exchange)

Data transfer:

After the handshake, two sides cut up their messages into fragment and append a Message Authentication Code (MAC). The MAC is a message digest of the message itself plus material derived from the master key [5]. As mentioned that MAC is computed during the handshake phrase. “When transmitting, the sender combines the data fragment, MAC, and a record header and encrypts them with the secret key to produce the completed SSL packet” [1]. When receiving, the receiver decodes the packet, and then compares the computed MAC to the received MAC.

4. Example of attacking previous SSL version and solutions

In this section, three typical attacks will be presented. Cipher suite rollback attack and version rollback attack are more theoretical, and the Password Interception in a SSL/TLS is a man-in-the-middle attack which is a successful attack performed by a student of LASEC.

4.1 cipher suite rollback attack

According to Figure1, the initial handshake messages are not protected and that is why each endpoint sends a “change cipher spec” before finishing the handshake. The utility of “change cipher spec” is alert the other side to change their pending session state to current. In fact, “change cipher spec” message is not protected by the authentication during transfer. It is a quirk of the SSL protocol which an attacker can delete the “change cipher spec” message so that the server and client will never upgrade their current cipher suite.

One solution of this attack is including the “change cipher spec” in the finished message’s message authentication calculation. However, this would need a change to SSL specification. Another solution is to generate a warning message for this kind of error caused by an attacker.

4.2 Version rollback attacks

Although SSL 2.0 has many flaws, SSL 3.0 supports to accept SSL 2.0 connections. This acceptance make the protocol have vulnerable to the possibility of being attacked by version rollback attacks.

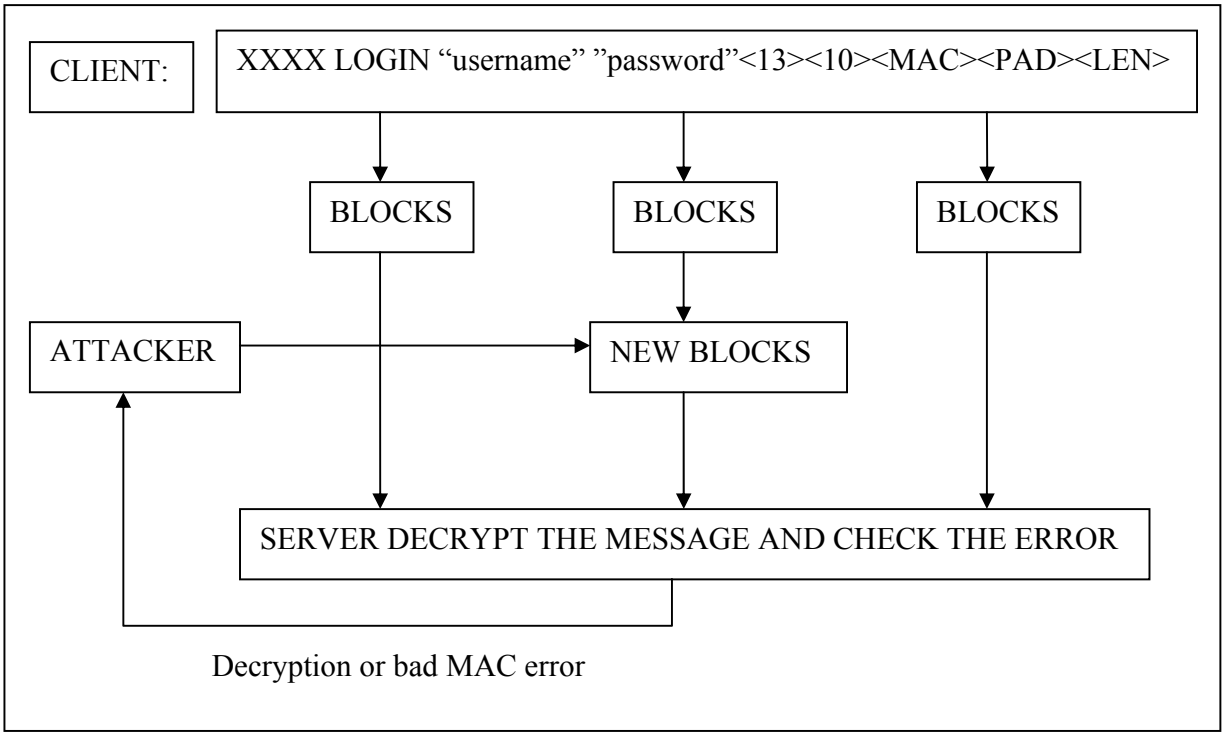
Paul Kocher designed an excellent strategy to detect the attack in different version [8]. He makes the client that supports SSL 3.0 embed several fixed redundancy in the RSA PKCS padding bytes. In the server side, if RSA encryption includes the RSA PKCS padding bytes, the server will not permit RSA-encrypted key-exchange over version 2.0.

This makes the endpoints which support SSL version 3.0 to have the ability to detect version rollback attacks. Furthermore, old SSL 2.0 clients will be using random PKCS padding, so they will still work with servers that support SSL 2.0[2].

However, this solution still has some vulnerability. Because the detect method depends on the assumption that SSL 2.0 only support RSA key-exchange, there still exist some non-RSA public key algorithm exist. Thus the version rollback attacks have no effect if the server supports non-RSA key-exchange while working in SSL 2.0 mode. In this case, two side which support both SSL 2.0 and SSL 3.0 will be forced to revert to version 2.0 protocols.

4.3 Password Interception in a SSL/TLS

A successful attack performed by Martin Vuagnoux in 2002 for intercepting password sent to an IMAP server when checking emails with MS Outlook Express 6.x client using a secure connection.



(Figure 2: Process of Martin’s attack)

There are several flaws of SSL found from this attack:

1. When SSL and TLS use block ciphers in Cipher Block Chaining Mode (CBC).After the message and MAC generated, there is always a padding and length of padding. Those data was cut into blocks of the fixed (always eight bytes) length have potential vulnerability which the cipher text length can disclose the length of plaintext. This can be easily attacked by an attacker by constructing the block and make a guess of messages in those blocks.
2. Using some traffic analysis or through timing attack can easily get some useful information attacker needed. Pascal Junod has been conducting some research on sequential distinguishers [6] .Within this method, LASEC get the distribution of decryption failed error and bad MAC error. These two kinds of error are important for an attacker to tell whether his guessing at a particular byte of the block is right or wrong, and determine discover the real transfer message. Bennet Yee [7] also noted examination of cipher text lengths can reveal information about URL requests in SSL-encrypted traffic.

3. I think one disadvantage of Outlook is it connects to the server every five minutes. These reconnections give the attacker many free sessions every five minutes. So it is important to design the client applications carefully. If a user open the browser directly and connect their mail box, the attack of Martin's will not succeed. Nowadays, many people use on line banking and shopping software to transmit their personal information such as bank account password and credit card password. It is vital that a more secure channel is created to transfer this information.
4. One solution to protect Martin's attack is to check the message authentication code even when the padding is not correct. However the only difference is no padding error generated or much complex error report to the client side. In my point of view, another solution is in the server side, according to John's anomaly-based intrusion detection, the server should have the ability to scan its log file and find the repeated generation of decryption error and bad_MAC error, then make some decision against this kind of attack.

   They also can perform an attack by analyzing these error messages. Furthermore, after the server found the fatal error and generates a new session, the clear text in the new session is always the same as the old one. LASEC also recommends [3] "changing the order in which the padding and the MAC are added to the plaintext in CBC mode". First appending padding to the message then adding the MAC can enable the receiver of the cipher text can first check the MAC, and then treat the padding when the message is authenticated.

5. Conclusion:

From researching the difference between SSL version 2.0 and 3.0, and the attack in practice. It is easy to find although SSL is a long developed protocol, there are quite a lot of vulnerabilities with the protocol. These secure problems are fatal for the information transfer in the Internet. And the solution of these potential dangerous is not easy to implement and change. Some of solutions are related with protocol itself, such as version rollback attack, which should take a long time to replace all SSL 3.0 with version 3.0. Others are easy to improve, as Martin's attack to IMAP server, but people should aware the essential problem within that attack, not only stop the attacker, but also prevent the

possibility in the future. In fact, SSL/TLS protocol is including complex algorithm and calculations. It is impossible to solve all the problems by changing the protocol itself; other intrusion detection in the server side is a good method to prevent potential attack from man-in-the-middle attack and repeated attack. Because most attack presented in this paper should be executed several times to find the correct result. There is a long way for the development of SSL/TLS.

Reference

[1] Wesley Chou "Inside SSL: The secure sockets layer protocol" IEEE IT Professional Volume: 4 Issue: 4, Jul/Aug 2002 Page(s): 47 -52
Available at URL: http://www.computer.org/itpro/it2002/f4047abs.htm

[2] David Wagner, Bruce Schneider. "Analysis of the SSL 3.0 protocol" University of California, Berkeley and Counterpane Systems, April 15 1997 Available at URL http://www.counterpane.com/ssl.html

[3]Brice Canvel. "Password interception in a SSL/TLS channel" Available at URL http://lasecwww.epfl.ch/memo_ssl.shtml

[4] Alan O. Freier, Philip Karlton, Paul C. Kocher, Independent Consultant "The SSL Protocol 3.0" Available at URL http://www.netscape.com/eng/ssl3/ssl-toc.html March 1996

[5] T. Dierks C. Allen. "TLS document" Available at URL http://www.ietf.org/rfc/rfc2246.txt January 1999

[6] Pascal Junod. "On the optimality of linear, differential and sequential distinguishers" EPFL I&C Technical Report 2003

[7] Bennet Yee, from the writer of [2]'s personal communication, June 1996

[8] Paul kocher, from the writer of [2]'s personal communication, 1996