SSL lab

Adapted from wireshark lab by J.F. Kurose and K.W. Ross

.cap file:

1. you can use the cap file in snakeoil2\_070531.zip if you want to do the last question in this lab.
2. or you can use the cap file provided by Kurose and Ross.
3. or you can capture your own SSL packets.

the lab:

open your .cap file and answer the questions below:

1. For each of the first 8 Ethernet frames, specify the source of the frame (client or server), determine the number of SSL records that are included in the frame, and list the SSL record types that are included in the frame. Draw a timing diagram between client and server, with one arrow for each SSL record. （有时一个数据包含有多个SSL包）
2. Each of the SSL records begins with the same three fields (with possibly different values). One of these fields is “content type” and has length of one byte. List all three fields and their lengths.

ClientHello Record:

1. Expand the ClientHello record. (If your trace contains multiple ClientHello records, expand the frame that contains the first one.) What is the value of the content type?
2. Does the ClientHello record contain a nonce (also known as a “challenge”)? If so, what is the value of the challenge in hexadecimal notation?
3. Does the ClientHello record advertise the cyber suites it supports? If so, in the first listed suite, what are the public-key algorithm, the symmetric-key algorithm, and the hash algorithm?

ServerHello Record:

1. Locate the ServerHello SSL record. Does this record specify a chosen cipher suite? What are the algorithms in the chosen cipher suite?
2. Does this record include a nonce? If so, how long is it? What is the purpose of the client and server nonces in SSL?
3. Does this record include a session ID? What is the purpose of the session ID?

When the server sends the “Server Hello” message, it can include a session identifier. The client should store it and present it in the “Client Hello” message of the next session. If the server finds the corresponding session in its cache and accepts to resume the session, it will send back the same session identifier and will continue with the abbreviated SSL handshake.

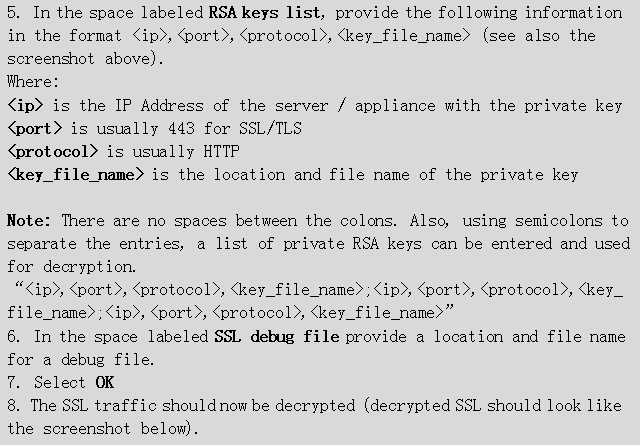
1. Does this record contain one certificate or several certificates? What is the relationship between these certificates? Could you find the corresponding certificate in your browser?

Client Key Exchange Record:

1. Locate the client key exchange record. Does this record contain a pre-master secret? What is this secret used for? Is the secret encrypted? If so, how? How long is the encrypted secret? （128个字节）

Change Cipher Spec Record (sent by client) and Encrypted Handshake Record:

1. What is the purpose of the Change Cipher Spec record? How many bytes is the record in your trace?
2. In the encrypted handshake record, what is being encrypted? How?
3. Does the server also send a change cipher record and an encrypted handshake record to the client? How are those records different from those sent by the client?
4. Broke the SSL stream by providing the private key of the server. Set RSA keys list to **127.0.0.1,443,http,/path/to/rsasnakeoil2.key** to decrypt. Think about why it can decrypt the encrypted message.



Further questions and references:

1. Very interesting article on SSL：

<http://www.moserware.com/2009/06/first-few-milliseconds-of-https.html>

1. broke the SSL stream method:

<https://wiki.wireshark.org/SSL/>