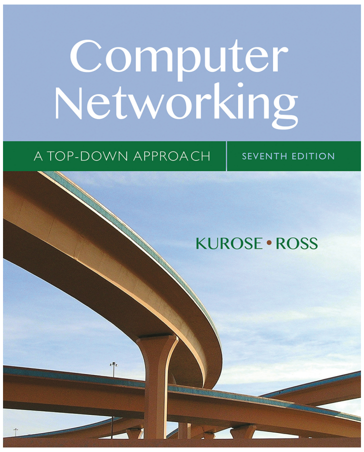
Wireshark Lab: SSL v7.0

Supplement to *Computer Networking: A Top-Down Approach, 7th ed.,* J.F. Kurose and K.W. Ross

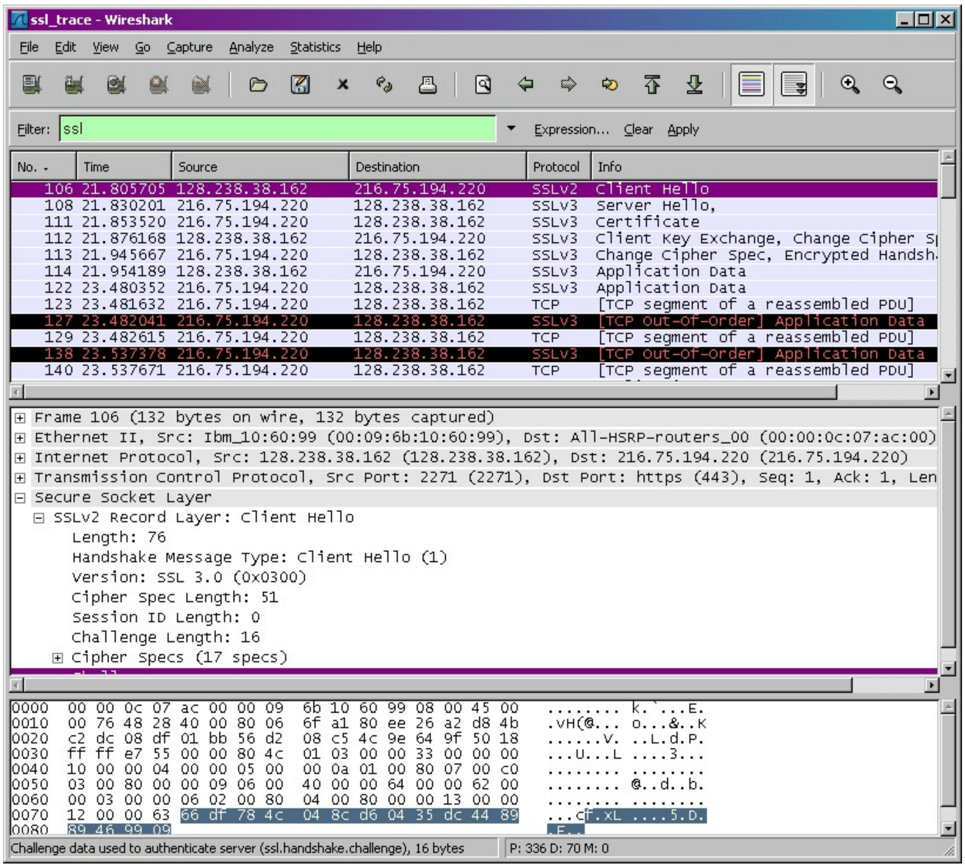


"Tell me and I forget. Show me and I remember. Involve me and I understand." Chinese proverb

In this lab, we'll investigate the Secure Sockets Layer (SSL) protocol, focusing on the SSL records sent over a TCP connection. We'll do so by analyzing a trace of the SSL records sent between your host and an e-commerce server. We'll investigate the various SSL record types as well as the fields in the SSL messages. You may want to review Section 8.6 in the text[1].

在本实验中，我们将研究安全套接层（SSL）协议，我们将会重点关注通过 TCP 连接发送的 SSL 记录。我们将会通过您的主机和电子商务服务器发送的 SSL 记录的跟踪来实现。 我们将研究各种 SSL 记录类型以及 SSL 消息中的字段。您可能会需要重新查看课本中的 8.6 节。

[1] References to figures and sections are for the 7th edition of our text, *Computer Networks, A Top-down Approach, 7th ed.,* J.F. Kurose and K.W. Ross, Addison-Wesley/Pearson, 2016.
课本：计算机网络 自顶向下方法第 7 版 中文版由机械工业出版社翻译发行


ssl_trace - Wireshark
File Edit View Go Capture Analyze Statistics Help
Expression... Clear Apply

Filter: ssl

```
No. -  Time      Source          Destination     Protocol Info
   106 21.805705 128.238.38.162  216.75.194.220  SSLv2    Client Hello
   108 21.830201 216.75.194.220  128.238.38.162  SSLv3    Server Hello,
   111 21.853520 216.75.194.220  128.238.38.162  SSLv3    Certificate
   112 21.876168 128.238.38.162  216.75.194.220  SSLv3    Client Key Exchange, Change Cipher Sp
   113 21.945667 216.75.194.220  128.238.38.162  SSLv3    Change Cipher Spec, Encrypted Handsh.
   114 21.954189 128.238.38.162  216.75.194.220  SSLv3    Application Data
   122 23.480352 216.75.194.220  128.238.38.162  SSLv3    Application Data
   123 23.481632 216.75.194.220  128.238.38.162  TCP      [TCP segment of a reassembled PDU]
   127 23.482041 216.75.194.220  128.238.38.162  SSLv3    [TCP Out-Of-Order] Application Data
   129 23.482615 216.75.194.220  128.238.38.162  TCP      [TCP segment of a reassembled PDU]
   138 23.537378 216.75.194.220  128.238.38.162  SSLv3    [TCP Out-Of-Order] Application Data
   140 23.537671 216.75.194.220  128.238.38.162  TCP      [TCP segment of a reassembled PDU]
```

```
⊞ Frame 106 (132 bytes on wire, 132 bytes captured)
⊞ Ethernet II, Src: Ibm_10:60:99 (00:09:6b:10:60:99), Dst: All-HSRP-routers_00 (00:00:0c:07:ac:00)
⊞ Internet Protocol, Src: 128.238.38.162 (128.238.38.162), Dst: 216.75.194.220 (216.75.194.220)
⊞ Transmission Control Protocol, Src Port: 2271 (2271), Dst Port: https (443), Seq: 1, Ack: 1, Len
⊟ Secure Socket Layer
  ⊟ SSLv2 Record Layer: Client Hello
      Length: 76
      Handshake Message Type: Client Hello (1)
      Version: SSL 3.0 (0x0300)
      Cipher Spec Length: 51
      Session ID Length: 0
      Challenge Length: 16
    ⊞ Cipher Specs (17 specs)
```

```
0000  00 00 0c 07 ac 00 00 09  6b 10 60 99 08 00 45 00   ........ k.`...E.
0010  00 76 48 28 40 00 80 06  6f a1 80 ee 26 a2 d8 4b   .vH(@... o...&..K
0020  c2 dc 08 df 01 bb 56 d2  08 c5 4c 9e 64 9f 50 18   ......V. ..L.d.P.
0030  ff ff e7 55 00 00 80 4c  01 03 00 00 33 00 00 00   ...U...L ....3...
0040  10 00 00 04 00 00 05 00  00 0a 01 00 80 07 00 c0   ........ ........
0050  03 00 80 00 00 09 06 00  40 00 00 64 00 00 62 00   ........ @..d..b.
0060  00 03 00 00 06 02 00 80  04 00 80 00 00 13 00 00   ........ ........
0070  12 00 00 63 66 df 78 4c  04 8c d6 04 35 dc 44 89   ...cf.xL ....5.D.
0080  89 46 99 09                                        .F..
```

Challenge data used to authenticate server (ssl.handshake.challenge), 16 bytes   P: 336 D: 70 M: 0

1. Capturing packets in an SSL session 在 SSL 会话中抓包

The first step is to capture the packets in an SSL session. To do this, you should go to your favorite e-commerce site and begin the process of purchasing an item (but terminating before making the actual purpose!). After capturing the packets with Wireshark, you should set the filter so that it displays only the Ethernet frames that contain SSL records sent from and received by your host. (An SSL record is the same thing as an SSL message.) You should obtain something like screenshot on the previous page.

第一步是在 SSL 会话中捕获数据包。要做这一步，您应该去你最喜欢电子商务网站开始购买物品（但是请勿真的购买）。使用 Wireshark 捕获数据包后，应设置过滤器，使其仅显示包含主机发送和接收的 SSL 记录的以太网帧。（SSL 记录就是 SSL 消息）您应该获得如上屏幕截图所示的内容。

If you have difficulty creating a trace, you should download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the *ssl-ethereal-trace-1* packet trace.

如果您抓包跟踪遇到困难，建议下载作者的抓包结果并且解压 *ssl-ethereal-trace-1 进行分析（*http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip*）。*

2. A look at the captured trace 分析抓包结果

Your Wireshark GUI should be displaying only the Ethernet frames that have SSL records. It is important to keep in mind that an Ethernet frame may contain one or more SSL records. (This is very different from HTTP, for which each frame contains either one complete HTTP message or a portion of a HTTP message.) Also, an SSL record may not completely fit into an Ethernet frame, in which case multiple frames will be needed to carry the record.
您使用的 Wireshark 界面应该仅仅显示含有 SSL 记录的以太网帧。建议您记住：每个以太网帧可能包含一个或多个的 SSL 记录，这点很重要。（这与 HTTP 消息不同，每个以太网帧包含一个完整的 HTTP 消息或者仅仅包含 HTTP 消息的一部分）因此，SSL 记录不仅仅需要一个以太网帧，这样的话，将有多个以太网承载。

Whenever possible, when answering a question below, you should hand in a printout of the packet(s) within the trace that you used to answer the question asked. Annotate the printout[2] to explain your answer. To print a packet, use *File->Print*, choose *Selected packet only*, choose *Packet summary line,* and select the minimum amount of packet detail that you need to answer the question
请尽量清晰的展示您的答案，必要时您可以在图中用标记辅以说明。您的答案应该简单可读。

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.
   对于前 8 个以太网帧，请分别指出每一个帧的来源（客户端和服务器），确定每个帧包含的 SSL 记录的数量，并且列出包含 SSL 记录的类型。绘制客户端和服务器含有箭头指向的时序图。
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.
   每个 SSL 记录都以相同的三个字段（可能具有不同的值）开头。 其中一个字段是“内容类型”，长度为一个字节。 请列出所有三个字段及其长度。

ClientHello Record: 客户端发出请求 (ClientHello) 记录

[2] What do we mean by “annotate”? If you hand in a paper copy, please highlight where in the printout you’ve found the answer and add some text (preferably with a colored pen) noting what you found in what you ‘ve highlight. If you hand in an electronic copy, it would be great if you could also highlight and annotate.
请善用标记展示你的实验结果。

3. 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?
展开 ClientHello 记录（如果您的跟踪包含多个 ClientHello 记录，请展开包含第一个记录的以太网帧），内容类型的值是多少？
4. Does the ClientHello record contain a nonce (also known as a “challenge”)? If so, what is the value of the challenge in hexadecimal notation?
ClientHello 记录是否包含随机数（也称为“挑战码”（challenge））？ 如果是这样，十六进制的挑战码值是多少？
5. 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?
ClientHello 记录是否通知了它所支持密码加密套件（suite）？ 如果是这样，请在第一个密码套件，分别指出非对称密钥加密算法，对称密钥加密算法，哈希算法分别都是什么？

ServerHello Record: 服务器回应(ServerHello) 记录

RC4是对称密钥加密体系

6. Locate the ServerHello SSL record. Does this record specify a chosen cipher suite? What are the algorithms in the chosen cipher suite?
找到 ServerHello SSL 记录。 此记录是否指定了之前的密码套件之一？ 选择的密码套件中有哪些算法？
7. Does this record include a nonce? If so, how long is it? What is the purpose of the client and server nonces in SSL?
此记录是否包含随机数？ 如过有，它有多长？ SSL 中客户端和服务器段随机数用来干什么？
8. Does this record include a session ID? What is the purpose of the session ID?
此记录是否包含会话 ID？ 会话 ID 的目的是什么？
9. Does this record contain a certificate, or is the certificate included in a separate record. Does the certificate fit into a single Ethernet frame?
此记录是否包含证书，或者证书是否包含在单独的记录中。 证书是否适合一个单独的以太网帧传输？

Client Key Exchange Record: 客户端密钥交换记录

10. 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?
找到客户端密钥交换记录。 此记录是否包含前主密钥(pre-master secret)？ 这个前主密钥用于什么？ 前主密钥加密了吗？ 如果是这样，为什么？ 加密的前主密钥有多长？

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

由客户端发送编码改变记录和加密握手记录：

11. What is the purpose of the Change Cipher Spec record? How many bytes is the record in your trace?
    通知服务器后面的通信使用协商好的加密方法和密钥进行通信
    编码改变记录目的是什么？在您的跟踪中本记录有多少字节。
12. In the encrypted handshake record, what is being encrypted? How?
    在加密的握手记录中，什么是加密的？ 为什么？
13. 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?
    服务器是否还向客户端发送更改编码记录和加密的握手记录？ 这些记录与客户发送的记录有何不同？

Application Data 应用数据

14. How is the application data being encrypted? Do the records containing application data include a MAC? Does Wireshark distinguish between the encrypted application data and the MAC?
    如何加密应用程序数据？ 包含应用程序数据的记录是否包含消息认证码 MAC？ Wireshark 是否区分加密的应用程序数据和消息认证码 MAC？
15. Comment on and explain anything else that you found interesting in the trace.
    请您指出和解释您在跟踪中发现的任何其他内容。