

Lab 7

Gunnar Yonker

I had trouble creating an SSL trace, so I used the one provided.

1.

Frame	Source	Destination	Time	SSL Count	SSL Type
106	128.238.38.162	216.75.194.220	21.805705	1	Client Hello
108	216.75.194.220	128.238.38.162	21.830201	1	Server Hello
111	216.75.194.220	128.238.38.162	21.853520	2	Certificate, Server Hello Done
112	128.238.38.162	216.75.194.220	21.876168	3	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
113	216.75.194.220	128.238.38.162	21.945667	2	Change Cipher Spec, Encrypted Handshake Message
114	128.238.38.162	216.75.194.220	21.954189	1	Application Data
122	216.75.194.220	128.238.38.162	23.480352	1	Application Data
149	216.75.194.220	128.238.38.162	23.559497	1	Application Data

Client: 128.238.38.162

Server: 216.75.194.220

2.

Content Type: 1 byte

Version: 2 bytes

Length: 2 bytes

ClientHello Record:

The image shows a Wireshark packet capture of an SSL handshake. The top pane displays a list of packets, with packet 106 (Client Hello) selected. The middle pane shows the details of the ClientHello record, including the version (3.0), cipher specifications, and other handshake parameters. The bottom pane shows the raw packet data in hexadecimal and ASCII.

ClientHello Record:

- 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)
 - Cipher Spec: TLS_RSA_WITH_RC4_128_MD5 (0x00000004)
 - Cipher Spec: TLS_RSA_WITH_RC4_128_SHA (0x00000005)
 - Cipher Spec: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x0000000a)
 - Cipher Spec: SSL2_RC4_128_WITH_MD5 (0x01000000)
 - Cipher Spec: SSL2_DES_192_EDE3_CBC_WITH_MD5 (0x0700c000)
 - Cipher Spec: SSL2_RC2_128_CBC_WITH_MD5 (0x03000000)
 - Cipher Spec: TLS_RSA_WITH_DES_CBC_SHA (0x00000009)
 - Cipher Spec: SSL2_DES_64_CBC_WITH_MD5 (0x00000040)
 - Cipher Spec: TLS_RSA_EXPORT1024_WITH_RC4_56_SHA (0x00000064)
 - Cipher Spec: TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA (0x00000062)
 - Cipher Spec: TLS_RSA_EXPORT_WITH_RC4_40_MD5 (0x00000003)
 - Cipher Spec: TLS_RSA_EXPORT_WITH_RC2_40_MD5 (0x00000006)
 - Cipher Spec: SSL2_RC4_128_EXPORT40_WITH_MD5 (0x02000000)
 - Cipher Spec: SSL2_RC2_128_CBC_EXPORT40_WITH_MD5 (0x04000000)
 - Cipher Spec: TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x00000113)
 - Cipher Spec: TLS_DHE_DSS_WITH_DES_CBC_SHA (0x00000112)
 - Cipher Spec: TLS_DHE_DSS_EXPORT1024_WITH_DES_CBC_SHA (0x00000063)

Lab 7

Gunnar Yonker

3.

The content type value in the Client Hello record is SSLv2 Record Layer, this means that SSLv2 protocol is being used for this handshake.

4.

Yes, it does contain a nonce(challenge).

66 df 78 4c 04 8c d6 04 35 dc 44 89 89 46 99 09

5.

Cipher Spec: TLS_RSA_WITH_RC4_128_MD5 (0x000004)

Public-Key Algorithm: RSA

Symmetric-Key Algorithm: RC4-128

Hash Algorithm: MD5

ServerHello Record:

The image shows a Wireshark packet capture of an SSLv2 handshake. The packet list on the left shows a ServerHello record (frame 165) from 128.238.38.162 to 216.75.194.220. The packet details pane on the right shows the following structure:

- SSLv2 Record Layer: Handshake Protocol: Server Hello
- Content Type: Handshake (22)
- Version: SSL 3.0 (0x0300)
- Length: 74
- Handshake Protocol: Server Hello
- Handshake Type: Server Hello (2)
- Length: 70
- Version: SSL 3.0 (0x0300)
- Random: 0000000042bed248b8831d04cc98c26e5bad4e267c391944f0f070ce57745
- Session ID Length: 32
- Session ID: 1bad05fab02eae92c64c54be4547c32f3e3c63d3a8c86ddad694b45682da22f
- Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)
- Compression Method: null (0)
- [JA3S Fullstring: 768,4,]
- [JA3S: 1f8f5a3d2fd435e36084db890693eaf]

The packet bytes pane on the right shows the raw data of the handshake record, including the random value and session ID.

6.

Yes, it does have a chosen cipher suite.

Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)

Public-Key Algorithm: RSA

Symmetric-Key Algorithm: RC4-128

Hash Algorithm: MD5

7.

Random: 0000000042dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745

Yes, it does include a nonce. It is a 32-byte value.

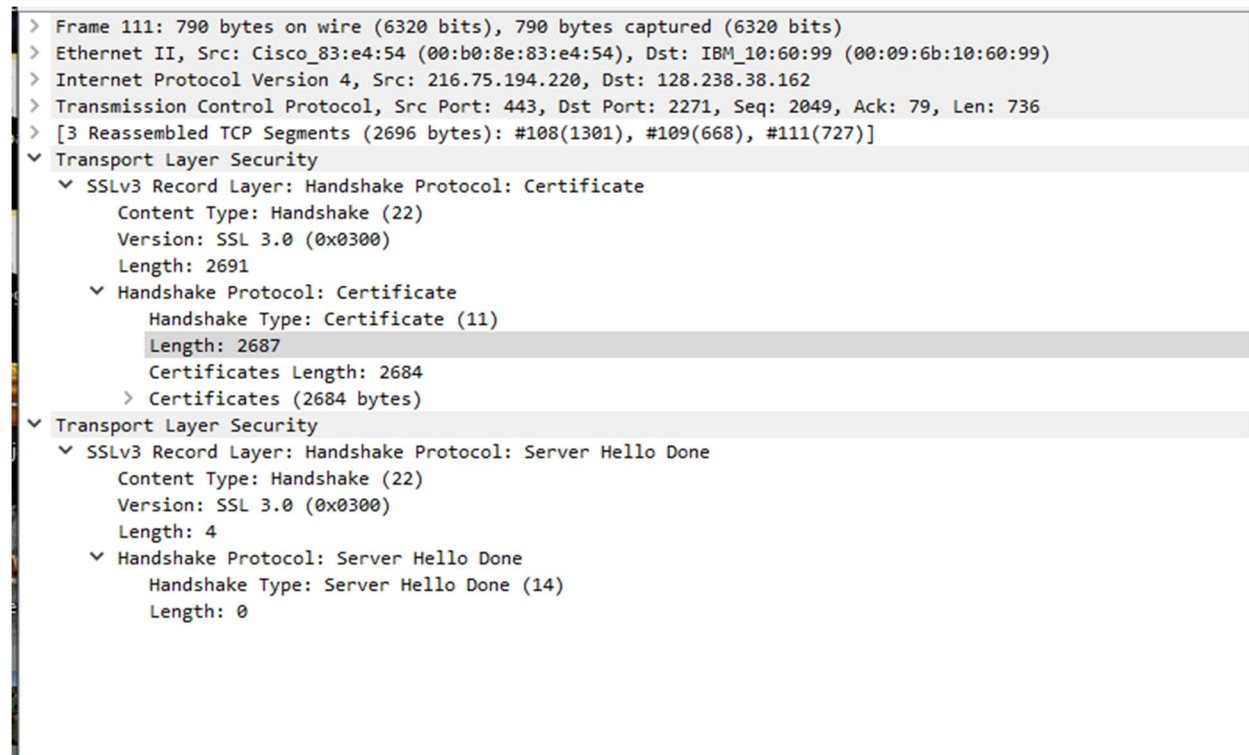
The purpose of the client and server nonces in SSL are to use them to establish a shared secret key that can then be used for the subsequent communication between them. The nonces are generated to be unique to that interaction as a random generated value and they are used so that the communication is not vulnerable to a replay attack.

8.

Session ID: 1bad05faba02ea92c64c54be4547c32f3e3ca63d3a0c86ddad694b45682da22f

Yes, it does include a session ID. The purpose of the session ID is to enable the client and server to resume a previous session. When the client initiates a new session it can then include the session ID in the Client Hello message and if the server has the matching session ID then the session can be resumed. This helps to save time and resources by avoiding having to re-complete the SSL handshake.

9.



```
> Frame 111: 790 bytes on wire (6320 bits), 790 bytes captured (6320 bits)
> Ethernet II, Src: Cisco_83:e4:54 (00:b0:8e:83:e4:54), Dst: IBM_10:60:99 (00:09:6b:10:60:99)
> Internet Protocol Version 4, Src: 216.75.194.220, Dst: 128.238.38.162
> Transmission Control Protocol, Src Port: 443, Dst Port: 2271, Seq: 2049, Ack: 79, Len: 736
> [3 Reassembled TCP Segments (2696 bytes): #108(1301), #109(668), #111(727)]
▼ Transport Layer Security
  ▼ SSLv3 Record Layer: Handshake Protocol: Certificate
    Content Type: Handshake (22)
    Version: SSL 3.0 (0x0300)
    Length: 2691
    ▼ Handshake Protocol: Certificate
      Handshake Type: Certificate (11)
      Length: 2687
      Certificates Length: 2684
      > Certificates (2684 bytes)
  ▼ Transport Layer Security
    ▼ SSLv3 Record Layer: Handshake Protocol: Server Hello Done
      Content Type: Handshake (22)
      Version: SSL 3.0 (0x0300)
      Length: 4
      ▼ Handshake Protocol: Server Hello Done
        Handshake Type: Server Hello Done (14)
        Length: 0
```

The certificate is included in a separate record in this case, the record is “Certificate, Server Hello Done”. This certificate does fit into a single Ethernet frame in this case, this usually depends on the MTU of the

Lab 7

Gunnar Yonker

network and the size of the certificate, but this certificate fits into one frame. This certificate has a length of 2687 in the packet.

Client Key Exchange Record:

The image shows a Wireshark packet capture of an SSLv3 handshake. The packet list on the left shows packet 112, which is the Client Key Exchange record. The packet details pane on the right shows the structure of the record:

- SSLv3 Record Layer: Handshake Protocol: Client Key Exchange (22)
 - Content Type: Handshake (22)
 - Version: SSL 3.0 (0x0300)
 - Length: 132
- Handshake Protocol: Client Key Exchange (16)
 - Length: 128
 - RSA Encrypted PreMaster Secret
- SSLv3 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec (20)
 - Content Type: Change Cipher Spec (20)
 - Version: SSL 3.0 (0x0300)
 - Length: 1
 - Change Cipher Spec Message
- SSLv3 Record Layer: Handshake Protocol: Encrypted Handshake Message (22)
 - Content Type: Handshake (22)
 - Version: SSL 3.0 (0x0300)
 - Length: 56
 - Handshake Protocol: Encrypted Handshake Message

The packet bytes pane on the right shows the raw data of the record, starting with 0000 00 00 0c 07 ac 00 00 09 60 10 60 99 08 00 45 00.

10.

This record does contain a pre-master secret. The secret is used to establish the shared secret key between the client and the server to subsequently encrypt data exchanged between them. It is encrypted using RSA encryption. The length of the pre-master secret is 128 bytes.

Encrypted PreMaster:

bc49494729aa2590477fd059056ae78956c77b12af08b47c609e61f104b0fbf83e41c08d...

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

The image shows a Wireshark packet capture of an SSL/TLS handshake. The packet list on the left shows several records, including '121 Change Cipher Spec, Encrypted Handshake Message' at time 21.945667. The packet details pane on the right shows the structure of this record: 'SSLv3 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec', 'Content Type: Change Cipher Spec (20)', 'Version: SSL 3.0 (0x0300)', 'Length: 1', 'Change Cipher Spec Message', 'SSLv3 Record Layer: Handshake Protocol: Encrypted Handshake Message', 'Content Type: Handshake (22)', 'Version: SSL 3.0 (0x0300)', 'Length: 56', 'Handshake Protocol: Encrypted Handshake Message'. The packet bytes pane on the right shows the raw data in hexadecimal and ASCII.

11.

The Change Cipher Spec record's purpose is to signal a switch in the encryption algorithm used for the communication between the client and the server. This message informs the other party that any further data will be encrypted with a different key than was previously being used. This is usually done during the handshake process once the client and server agree on what encryption algorithms will be used. In this record trace, just the Change Cipher Spec record has a length of 1 byte.

12.

The record is encrypted and contains a series of handshake messages using the established session key. It is encrypted so we do not know the exact contents but most likely contained is the server's certificate, cryptographic parameters such as the algorithm being used, and any other optional handshake messages between the client and server such as a client key exchange method. The handshake message is encrypted using the session keys that are used to generate a symmetric encryption key using an encryption method such as AES. The length of the Encrypted Handshake Message is 56 bytes which means that there are multiple handshake messages that have been encrypted using the established session key.

13.

Yes, the server also sends a Change Cipher Spec record and a Handshake Message to the client during the handshake process. The Change Cipher Spec record sent by the server has the same format as the one sent by the client so those do not differ other than being sent by the server instead of the client. The Handshake Message is in the same format as well containing the handshake messages that are encrypted using the established session key. The encryption format will be the same as the client record, the only difference is that the handshake messages contained will be different since it is coming from the server not the client. Even though they are pretty much the same, it is necessary to ensure that the same encryption parameters are being used by both parties to have secure communication.

Bonus Question:

The image shows a Wireshark packet capture of a TLSv1.2 connection. The packet list on the left shows packets 9051 through 9106. Packet 9092 is selected, showing details for a TLSv1.2 Record Layer: Handshake Protocol: Certificate. The packet bytes pane on the right shows the raw data of the selected packet, including the TLSv1.2 record structure and the certificate data.

No.	Time	Source	Destination	Protocol	Length	Info
9051	15.555757	192.168.0.168	192.225.158.132	TLSv1.2	571	Client Hello
9056	15.584044	192.225.158.132	192.168.0.168	TLSv1.2	96	Application Data
9057	15.584221	192.225.158.132	192.168.0.168	TLSv1.2	1514	Application Data
9063	15.584414	192.225.158.132	192.168.0.168	TLSv1.2	1505	Application Data
9064	15.584414	192.225.158.132	192.168.0.168	TCP	1514	443 → 4520 [ACK] Seq=28355 Ack=17999 Win=69120 Len=1448 TSval=3252151000 TSecr=1 [TCP segment of a reassembled PDU]
9070	15.584499	192.225.158.132	192.168.0.168	TLSv1.2	1514	Application Data
9078	15.584625	192.225.158.132	192.168.0.168	TLSv1.2	1514	Application Data
9085	15.597901	192.225.158.132	192.168.0.168	TLSv1.2	313	Application Data
9086	15.615363	192.225.158.132	192.168.0.168	TLSv1.2	1514	Server Hello
9092	15.615830	192.225.158.132	192.168.0.168	TLSv1.2	702	Certificate, Server Key Exchange, Server Hello Done
9094	15.616157	192.168.0.168	34.149.211.227	TLSv1.2	457	Application Data
9095	15.616192	192.168.0.168	34.149.211.227	TLSv1.2	3808	Application Data
9096	15.617334	192.168.0.168	192.225.158.132	TLSv1.2	100	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
9097	15.618278	192.168.0.168	192.225.158.132	TLSv1.2	3618	Application Data
9099	15.622692	192.225.158.132	192.168.0.168	TLSv1.2	1514	Server Hello
9103	15.623171	192.225.158.132	192.168.0.168	TLSv1.2	702	Certificate, Server Key Exchange, Server Hello Done
9105	15.623646	192.168.0.168	34.149.211.227	TLSv1.2	457	Application Data
9106	15.623684	192.168.0.168	34.149.211.227	TLSv1.2	3808	Application Data

(1)

The key exchange is using TLSv1.2 as the protocol and starts with the Client Hello message then continues similarly compared the previous SSL examples above. There is the Server Hello packet, Certificate, Server Key Exchange, Server Hello Done packet where the server sends the certificate to the client(me). Then the Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message packet from the client to the server followed by the same packet coming from the server later on in packet number 9128.

9126	15.685066	192.225.158.132	192.168.0.168	TLSv1.2	482	Application Data
9128	15.687059	192.225.158.132	192.168.0.168	TLSv1.2	117	Change Cipher Spec, Encrypted Handshake Message
9129	15.690471	192.225.158.132	192.168.0.168	TLSv1.2	117	Change Cipher Spec, Encrypted Handshake Message

Lab 7

Gunnar Yonker

(2)

Yes a digital certificate was used and it was the digital certificate of the server.

9092 15.615830	192.225.158.132	192.168.0.168	TLSv1.2	702	Certificate, Server Key Exchange, Server Hello Done
9094 15.616157	192.168.0.168	34.149.211.227	TLSv1.2	457	Application Data
9095 15.616192	192.168.0.168	34.149.211.227	TLSv1.2	3808	Application Data
9096 15.617334	192.168.0.168	192.225.158.132	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
9097 15.618278	192.168.0.168	192.225.158.132	TLSv1.2	3618	Application Data
9099 15.622692	192.225.158.132	192.168.0.168	TLSv1.2	1514	Server Hello
9103 15.623171	192.225.158.132	192.168.0.168	TLSv1.2	702	Certificate, Server Key Exchange, Server Hello Done
9105 15.623646	192.168.0.168	34.149.211.227	TLSv1.2	457	Application Data
9106 15.623684	192.168.0.168	34.149.211.227	TLSv1.2	3808	Application Data
9107 15.624783	192.168.0.168	192.225.158.132	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
9114 15.640467	34.149.211.227	192.168.0.168	TLSv1.2	449	Application Data
9116 15.648499	34.149.211.227	192.168.0.168	TLSv1.2	449	Application Data

> Frame 9092: 702 bytes on wire (5616 bits), 702 bytes captured (5616 bits) on interface \Device\NPF_{0BE8C525-6815-4239-91E0-89F0045...}

> Ethernet II, Src: TP-Link 61:13:c0 (28:87:ba:61:13:c0), Dst: ASUSTek_4f:ca:22 (5c:7c:3f:4f:ca:22)

> Internet Protocol Version 4, Src: 192.225.158.132, Dst: 192.168.0.168

> Transmission Control Protocol, Src Port: 443, Dst Port: 4521, Seq: 4897, Ack: 518, Len: 636

> [4 Reassembled TCP Segments (4272 bytes): #9086(1335), #9089(1448), #9091(1200), #9092(289)]

Transport Layer Security

- TLSv1.2 Record Layer: Handshake Protocol: Certificate
 - Content Type: Handshake (22)
 - Version: TLS 1.2 (0x0303)
 - Length: 4267
 - Handshake Protocol: Certificate
 - Handshake Type: Certificate (11)
 - Length: 4263
 - Certificates Length: 4260
 - Certificates (4260 bytes)
- TLSv1.2 Record Layer: Handshake Protocol: Server Key Exchange
 - Content Type: Handshake (22)
 - Version: TLS 1.2 (0x0303)
 - Length: 333
 - Handshake Protocol: Server Key Exchange
 - Handshake Type: Server Key Exchange (12)
 - Length: 329
 - EC Diffie-Hellman Server Params
 - Curve Type: named_curve (0x03)
 - Named Curve: secp256r1 (0x0017)
 - Pubkey Length: 65

0000 3c 7c 3f 4f ca 22 28 87 ba 61 13 c0 08 00 45 00 <[70]T(0).....E:

0010 02 b0 95 5d 40 00 38 06 89 f4 c0 e1 9e 64 c0 a0@S.....

0020 00 a8 01 bb 11 a9 ae 05 87 6e 44 63 65 cb 00 18ndce.....

0030 00 3b c9 74 00 00 01 01 08 0a c1 d7 e2 f8 00 00t.....

0040 00 01 7d e7 3b 09 51 04 6c 8d 9f 90 12 66 ab 30} ;1Q-1----f-0

0050 0d 06 09 2a 06 48 06 f7 0d 01 01 0b 05 00 83 82-P-H.....

0060 01 01 00 79 9f 1d 96 c6 b6 79 3f 22 8d 87 d3 87y?.....

0070 03 04 60 6a 6b 9a 2e 59 89 73 11 ac 43 d1 f5 13Jk..Y..s:..C...

0080 ff 8d 39 2b c0 f2 bd 4f 70 8c a9 2f ea 17 c4 0b-9---0 p-/-...

0090 54 9e d4 1b 96 08 33 c0 a8 ad e2 a2 00 76 ab 59 T-----3c---b-v-Y

00a0 69 6e 06 1d 7e c4 b9 44 8d 98 af 12 d4 61 db 0a in-----D-----

00b0 19 46 47 f3 eb f7 63 c1 40 05 40 a5 d2 b7 f4 b5 -FG---c-@-@-----

00c0 9a 36 bf a9 88 76 88 04 55 84 2b 9c 87 7f 1a 37 -6---v---U+---7

00d0 3c 7e 2d a5 1a d8 44 89 5e ca bd ac 3d 6c d8 6d <-----A---1-m

00e0 af d5 f3 76 0f cd 3b 88 38 22 9d 6c 93 9a c4 3d8"-l-==

00f0 bf 82 1b 65 3f a6 0f 5d aa fc e5 b2 15 ca b5 ade?..].....

0100 c5 bc 3d d0 04 e8 ea 06 72 10 ad 39 32 78 bf 3e-P-H2a->

0110 11 9c 0b a4 9d 9a 21 f3 f0 9b 0b 30 78 db c1 dc-Bx-----

0120 87 43 fe bc 63 9a ca c5 c2 1c c9 c7 8d ff 3b 12 -C-----:-----

0130 58 08 e6 b6 3d ec 7a 2c 4e fb 83 96 ce 0c 3c 69 X-----z, H-----i

0140 87 54 73 a4 73 c2 93 ff 51 10 ac 15 54 01 68 fc 'Ts-s---Q---T---

0150 05 b1 09 a1 7f 74 83 9a 49 07 dc 4e 7b 8a 48 6ft---I-N{Ho

0160 8b 45 f6 16 03 03 01 4d 0c 00 01 49 03 00 17 41 -E-----H-----A

0170 04 ef 93 2d a9 84 08 95 c3 ea e6 75 48 b4 2c 38spht,8

0180 09 36 18 bf 2a 9b 5c 7f 72 c6 fd 63 88 3d 6b d4 -6---A---p---k-

0190 30 f2 43 b7 fc 9c 6d 1e c0 6b 0e 97 ab 20 68 35 0-C-----k-----h5

01a0 b0 ac e2 71 9e b4 32 4d d8 73 18 e8 95 8e 8b 99-q-2M-s-----

01b0 9c 06 01 01 00 b8 71 8a 87 80 bf a3 cd b2 ec 32-q-----2

01c0 2b 6d df a4 93 3a f3 56 2b 45 4f 9c 73 8a 10 5f 4m-----V-EO-s---

01d0 c8 fe bc f8 02 e3 90 d9 40 49 44 85 9a 2c a4 9e@ID-----