Final Year B.Tech. (CSE) – VII [2024-25]

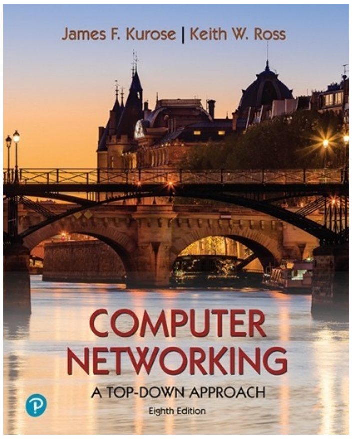
6CS451: Cryptography and Network Security Lab (C&NS Lab)

Date: 21/10/2024

Assignment10

PRN: 21510042 **Name:** Omkar Rajesh Auti

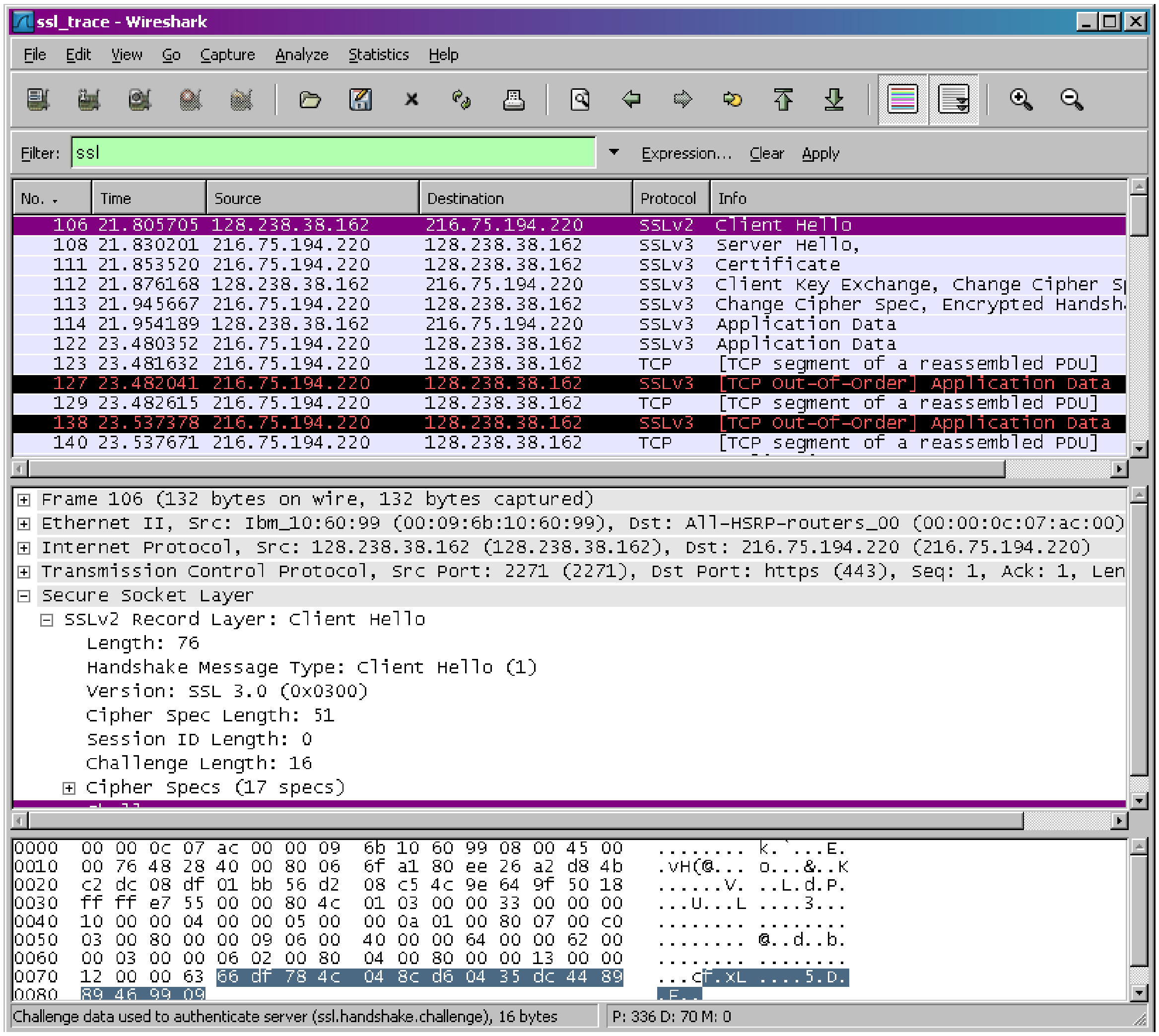
Wireshark Lab: SSL v8.0



Supplement to *Computer Networking: A Top-Down Approach, 8th 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 will investigate the Secure Sockets Layer (SSL) protocol, focusing on the SSL records sent over a TCP connection. We will do so by analyzing a trace of the SSL records sent between your host and an e-commerce server. We will investigate the various SSL record types as well as the fields in the SSL messages.


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

```
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 S
 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

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.

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.

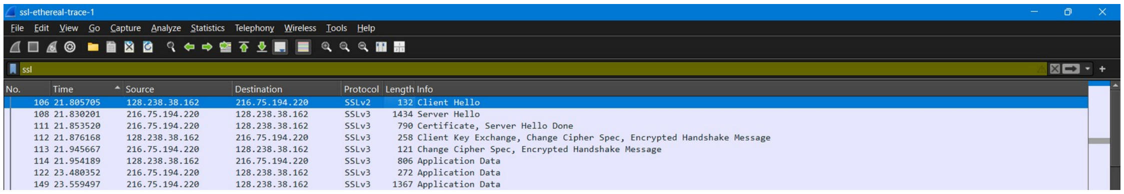
2. A look at the captured trace

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.

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

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.


ssl-ethereal-trace-1
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ssl

```
No.   Time        Source          Destination     Protocol Length Info
  106 21.805705   128.238.38.162  216.75.194.220  SSLv2     132 Client Hello
  108 21.830201   216.75.194.220  128.238.38.162  SSLv3    1434 Server Hello
  111 21.853520   216.75.194.220  128.238.38.162  SSLv3     790 Certificate, Server Hello Done
  112 21.876168   128.238.38.162  216.75.194.220  SSLv3     258 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
  113 21.945667   216.75.194.220  128.238.38.162  SSLv3     121 Change Cipher Spec, Encrypted Handshake Message
  114 21.954189   128.238.38.162  216.75.194.220  SSLv3     806 Application Data
  122 23.480352   216.75.194.220  128.238.38.162  SSLv3     272 Application Data
  149 23.559497   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
```

first 8 Ethernet frames to determine:

1. Source (client or server)
2. Number of SSL records in the frame
3. SSL record types

Frame 1: (Frame 106)

- Source: Client (128.238.38.162)
- Number of SSL Records: 1
- SSL Record Type: Client Hello (SSLv2)

Frame 2: (Frame 108)

- Source: Server (216.75.194.220)
- Number of SSL Records: 1
- SSL Record Type: Server Hello (SSLv3)

Frame 3: (Frame 111)

- Source: Server (216.75.194.220)
- Number of SSL Records: 2
- SSL Record Types:
    1. Certificate (SSLv3)
    2. Server Hello Done (SSLv3)

Frame 4: (Frame 112)

- Source: Client (128.238.38.162)
- Number of SSL Records: 3
- SSL Record Types:
    1. Client Key Exchange (SSLv3)
    2. Change Cipher Spec (SSLv3)
    3. Encrypted Handshake Message (SSLv3)

Frame 5: (Frame 113)

- Source: Server (216.75.194.220)
- Number of SSL Records: 2
- SSL Record Types:
    1. Change Cipher Spec (SSLv3)
    2. Encrypted Handshake Message (SSLv3)

Frame 6: (Frame 114)

- Source: Client (128.238.38.162)
- Number of SSL Records: 1
- SSL Record Type: Application Data (SSLv3)

Frame 7: (Frame 122)

- Source: Server (216.75.194.220)
- Number of SSL Records: 1
- SSL Record Type: Application Data (SSLv3)

Frame 8: (Frame 149)

- Source: Server (216.75.194.220)
- Number of SSL Records: 1
- SSL Record Type: Application Data (SSLv3)

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.

Each SSL record starts with the following three fields:

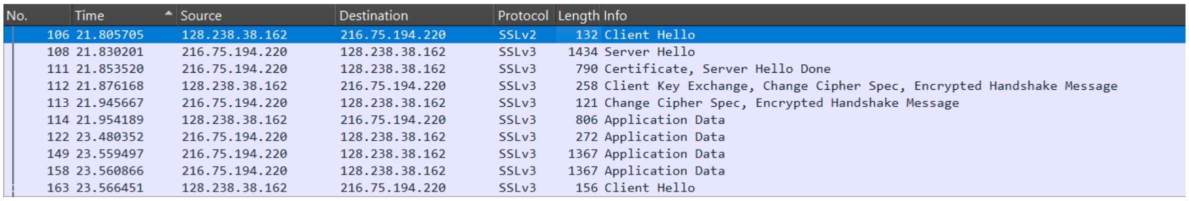
- **Content Type:** 1 byte
- **Version:** 2 bytes
- **Length:** 2 bytes

How to Find These Fields:

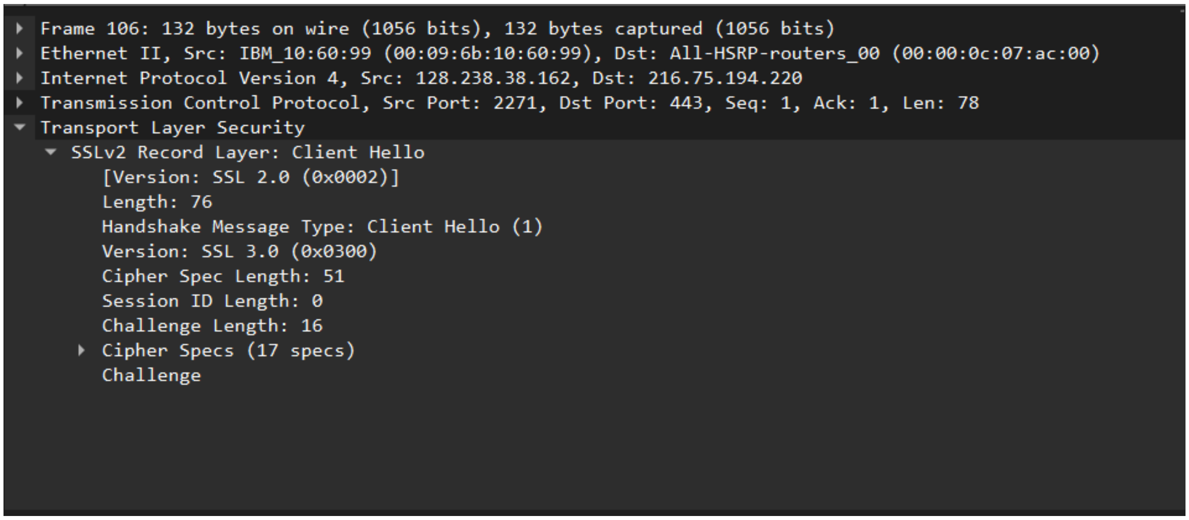
If you are using packet capture software like **Wireshark**, you can find these fields in the packet capture by:

1. **Open Wireshark** and load the captured SSL/TLS packet data (the one you listed).
2. **Select an SSL/TLS packet** from the list and expand the **"Secure Sockets Layer"** or **"Transport Layer Security"** section in the detailed packet view.

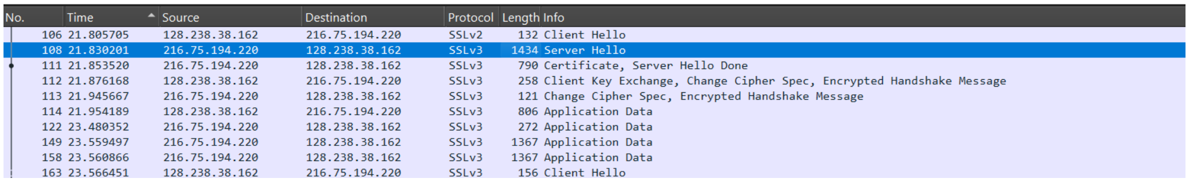
3. You will see the **Record Layer** header information, where these fields will be listed:
   - **Content Type:** Displays the type of SSL/TLS record (Handshake, Application Data, etc.)
   - **Version:** The protocol version (e.g., TLS 1.2)
   - **Length:** The size of the encrypted data.



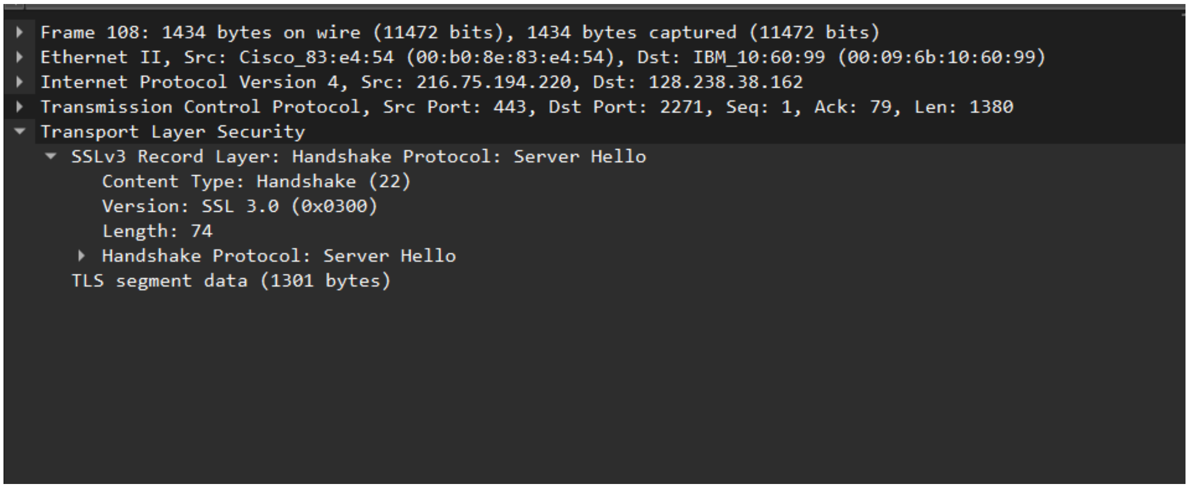
```
No.   Time        Source          Destination     Protocol Length Info
  106 21.805705   128.238.38.162  216.75.194.220  SSLv2     132 Client Hello
  108 21.830201   216.75.194.220  128.238.38.162  SSLv3    1434 Server Hello
  111 21.853520   216.75.194.220  128.238.38.162  SSLv3     790 Certificate, Server Hello Done
  112 21.876168   128.238.38.162  216.75.194.220  SSLv3     258 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
  113 21.945667   216.75.194.220  128.238.38.162  SSLv3     121 Change Cipher Spec, Encrypted Handshake Message
  114 21.954189   128.238.38.162  216.75.194.220  SSLv3     806 Application Data
  122 23.480352   216.75.194.220  128.238.38.162  SSLv3     272 Application Data
  149 23.559497   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
  158 23.560866   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
  163 23.566451   128.238.38.162  216.75.194.220  SSLv3     156 Client Hello
```



```
▸ Frame 106: 132 bytes on wire (1056 bits), 132 bytes captured (1056 bits)
▸ 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 Version 4, Src: 128.238.38.162, Dst: 216.75.194.220
▸ Transmission Control Protocol, Src Port: 2271, Dst Port: 443, Seq: 1, Ack: 1, Len: 78
▾ Transport Layer Security
    ▾ SSLv2 Record Layer: Client Hello
        [Version: SSL 2.0 (0x0002)]
        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)
        Challenge
```



```
No.   Time        Source          Destination     Protocol Length Info
  106 21.805705   128.238.38.162  216.75.194.220  SSLv2     132 Client Hello
  108 21.830201   216.75.194.220  128.238.38.162  SSLv3    1434 Server Hello
  111 21.853520   216.75.194.220  128.238.38.162  SSLv3     790 Certificate, Server Hello Done
  112 21.876168   128.238.38.162  216.75.194.220  SSLv3     258 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
  113 21.945667   216.75.194.220  128.238.38.162  SSLv3     121 Change Cipher Spec, Encrypted Handshake Message
  114 21.954189   128.238.38.162  216.75.194.220  SSLv3     806 Application Data
  122 23.480352   216.75.194.220  128.238.38.162  SSLv3     272 Application Data
  149 23.559497   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
  158 23.560866   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
  163 23.566451   128.238.38.162  216.75.194.220  SSLv3     156 Client Hello
```

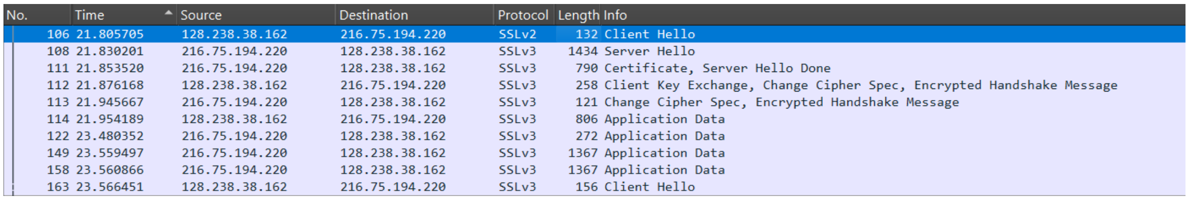


```
▸ Frame 108: 1434 bytes on wire (11472 bits), 1434 bytes captured (11472 bits)
▸ Ethernet II, Src: Cisco_83:e4:54 (00:b0:8e:83:e4:54), Dst: IBM_10:60:99 (00:09:6b:10:60:99)
▸ Internet Protocol Version 4, Src: 216.75.194.220, Dst: 128.238.38.162
▸ Transmission Control Protocol, Src Port: 443, Dst Port: 2271, Seq: 1, Ack: 79, Len: 1380
▾ Transport Layer Security
  ▾ SSLv3 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: SSL 3.0 (0x0300)
      Length: 74
    ▸ Handshake Protocol: Server Hello
    TLS segment data (1301 bytes)
```

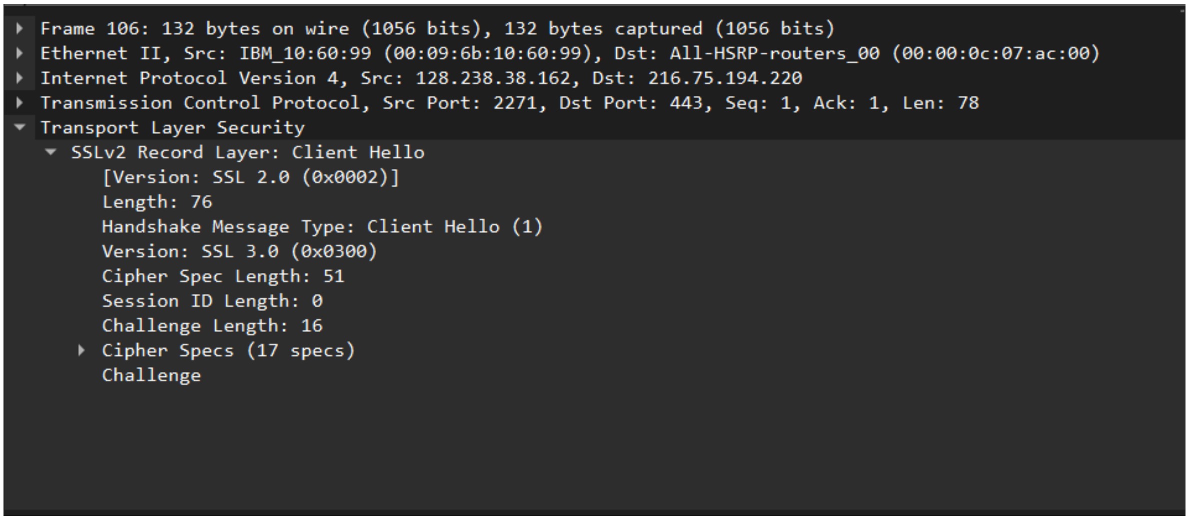
ClientHello Record:

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?

The **ClientHello** record in **Frame 106** is an SSLv2 message with a handshake message type of **Client Hello (1)**.



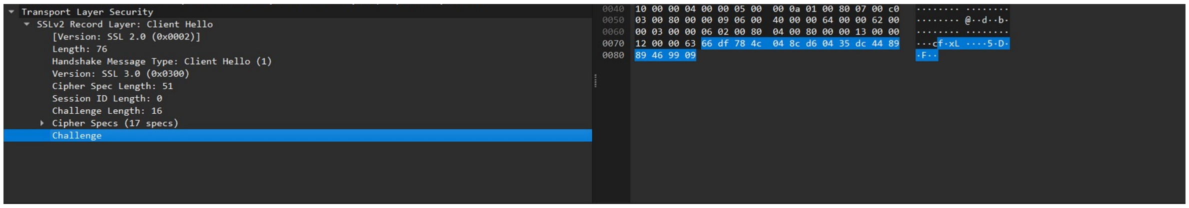
No.	Time	Source	Destination	Protocol	Length	Info
106	21.805705	128.238.38.162	216.75.194.220	SSLv2	132	Client Hello
108	21.830201	216.75.194.220	128.238.38.162	SSLv3	1434	Server Hello
111	21.853520	216.75.194.220	128.238.38.162	SSLv3	790	Certificate, Server Hello Done
112	21.876168	128.238.38.162	216.75.194.220	SSLv3	258	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
113	21.945667	216.75.194.220	128.238.38.162	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
114	21.954189	128.238.38.162	216.75.194.220	SSLv3	806	Application Data
122	23.480352	216.75.194.220	128.238.38.162	SSLv3	272	Application Data
149	23.559497	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
158	23.560866	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
163	23.566451	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello



```
▸ Frame 106: 132 bytes on wire (1056 bits), 132 bytes captured (1056 bits)
▸ 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 Version 4, Src: 128.238.38.162, Dst: 216.75.194.220
▸ Transmission Control Protocol, Src Port: 2271, Dst Port: 443, Seq: 1, Ack: 1, Len: 78
▾ Transport Layer Security
  ▾ SSLv2 Record Layer: Client Hello
      [Version: SSL 2.0 (0x0002)]
      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)
      Challenge
```

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?

Answer: YES

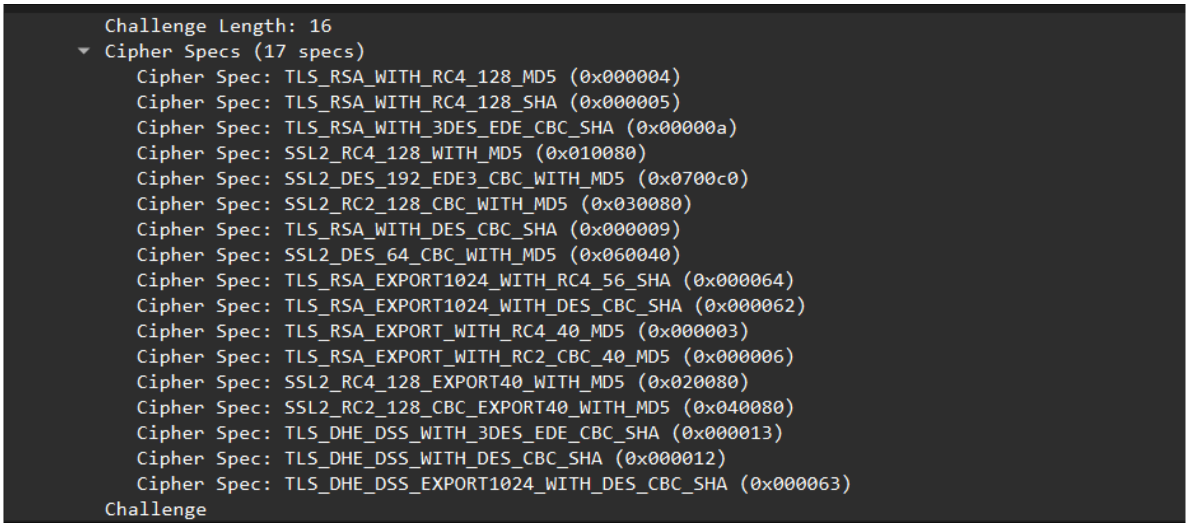


```
▾ Transport Layer Security
  ▾ SSLv2 Record Layer: Client Hello
      [Version: SSL 2.0 (0x0002)]
      Length: 76
      Handshake Message Type: Client Hello (1)
      Version: SSL 3.0 (0x0300)
      Cipher Spec Length: 51
      Session ID Length: 0
      Challenge Length: 16
    ▸ Cipher Specs (17 specs)
      Challenge
```

```
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··
```

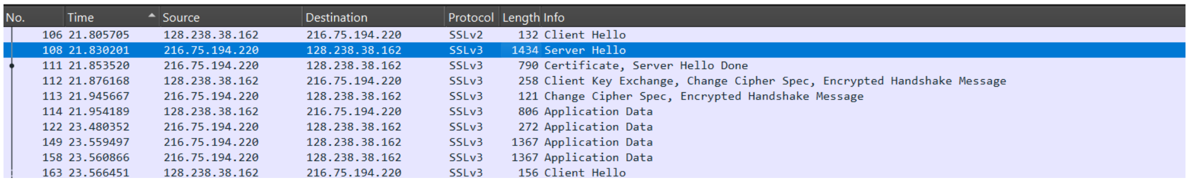
5. Does the ClientHello record advertise the cyber suites it supports? If so, in the first listed suite, what are the public-key algorithm, the symmetric-key algorithm, and the hash algorithm?

Answer: YES

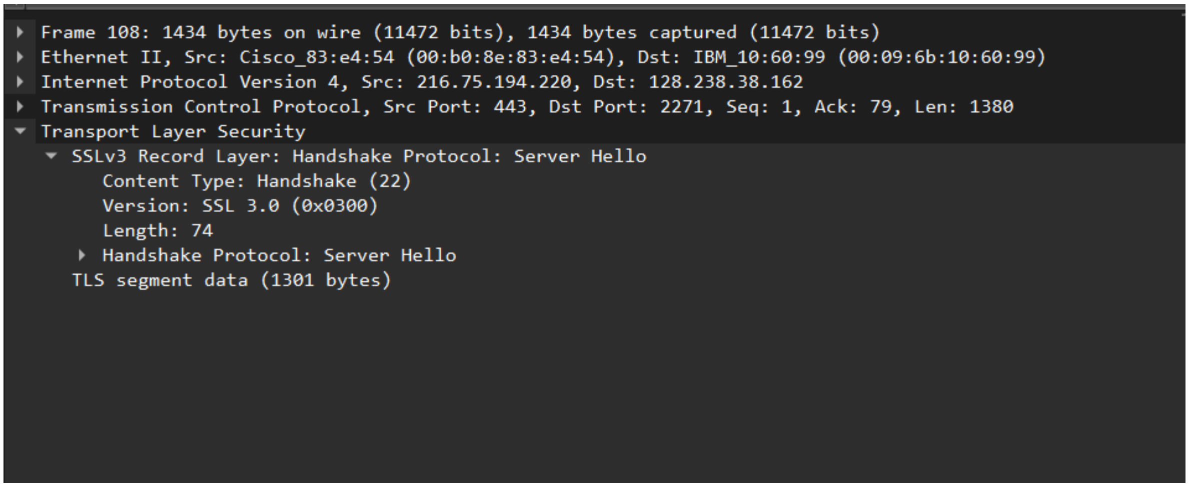


```
    Challenge Length: 16
  ▾ Cipher Specs (17 specs)
      Cipher Spec: TLS_RSA_WITH_RC4_128_MD5 (0x000004)
      Cipher Spec: TLS_RSA_WITH_RC4_128_SHA (0x000005)
      Cipher Spec: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x00000a)
      Cipher Spec: SSL2_RC4_128_WITH_MD5 (0x010080)
      Cipher Spec: SSL2_DES_192_EDE3_CBC_WITH_MD5 (0x0700c0)
      Cipher Spec: SSL2_RC2_128_CBC_WITH_MD5 (0x030080)
      Cipher Spec: TLS_RSA_WITH_DES_CBC_SHA (0x000009)
      Cipher Spec: SSL2_DES_64_CBC_WITH_MD5 (0x060040)
      Cipher Spec: TLS_RSA_EXPORT1024_WITH_RC4_56_SHA (0x000064)
      Cipher Spec: TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA (0x000062)
      Cipher Spec: TLS_RSA_EXPORT_WITH_RC4_40_MD5 (0x000003)
      Cipher Spec: TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 (0x000006)
      Cipher Spec: SSL2_RC4_128_EXPORT40_WITH_MD5 (0x020080)
      Cipher Spec: SSL2_RC2_128_CBC_EXPORT40_WITH_MD5 (0x040080)
      Cipher Spec: TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x000013)
      Cipher Spec: TLS_DHE_DSS_WITH_DES_CBC_SHA (0x000012)
      Cipher Spec: TLS_DHE_DSS_EXPORT1024_WITH_DES_CBC_SHA (0x000063)
    Challenge
```

ServerHello Record:



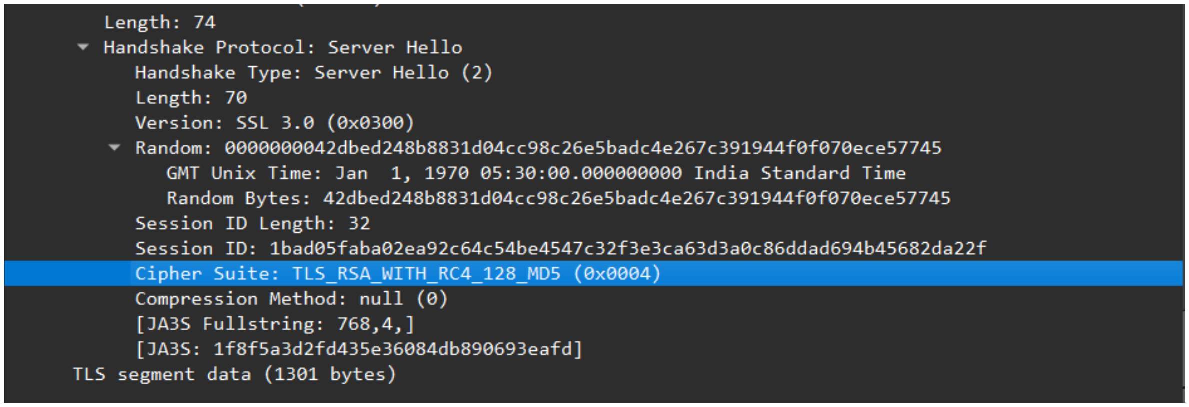
No.	Time	Source	Destination	Protocol	Length	Info
106	21.805705	128.238.38.162	216.75.194.220	SSLv2	132	Client Hello
108	21.830201	216.75.194.220	128.238.38.162	SSLv3	1434	Server Hello
111	21.853520	216.75.194.220	128.238.38.162	SSLv3	790	Certificate, Server Hello Done
112	21.876168	128.238.38.162	216.75.194.220	SSLv3	258	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
113	21.945667	216.75.194.220	128.238.38.162	SSLv3	121	Change Cipher Spec, Encrypted Handshake Message
114	21.954189	128.238.38.162	216.75.194.220	SSLv3	806	Application Data
122	23.480352	216.75.194.220	128.238.38.162	SSLv3	272	Application Data
149	23.559497	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
158	23.560866	216.75.194.220	128.238.38.162	SSLv3	1367	Application Data
163	23.566451	128.238.38.162	216.75.194.220	SSLv3	156	Client Hello



```
▸ Frame 108: 1434 bytes on wire (11472 bits), 1434 bytes captured (11472 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: 1, Ack: 79, Len: 1380
▾ Transport Layer Security
  ▾ SSLv3 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: SSL 3.0 (0x0300)
      Length: 74
    ▸ Handshake Protocol: Server Hello
    TLS segment data (1301 bytes)
```

6. Locate the ServerHello SSL record. Does this record specify a chosen cipher suite? What are the algorithms in the chosen cipher suite?

Answer: YES

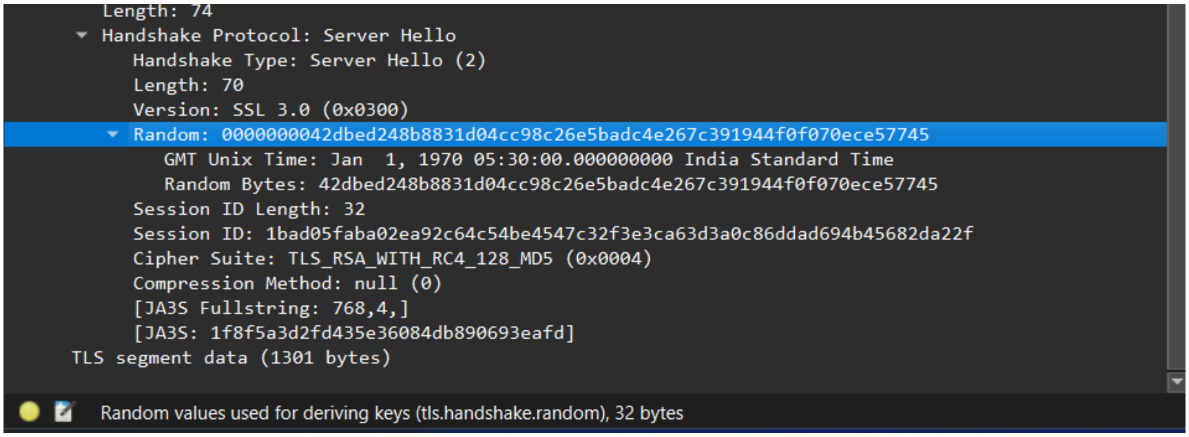


```
      Length: 74
    ▾ Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 70
        Version: SSL 3.0 (0x0300)
      ▾ Random: 0000000042dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745
          GMT Unix Time: Jan  1, 1970 05:30:00.000000000 India Standard Time
          Random Bytes: 42dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745
        Session ID Length: 32
        Session ID: 1bad05faba02ea92c64c54be4547c32f3e3ca63d3a0c86ddad694b45682da22f
        Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)
        Compression Method: null (0)
        [JA3S Fullstring: 768,4,]
        [JA3S: 1f8f5a3d2fd435e36084db890693eafd]
    TLS segment data (1301 bytes)
```

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?

Locate the Nonce:

- The **ServerHello** response may not explicitly list a nonce like the **ClientHello** does, but it usually includes a **Session ID** and potentially a **Server Random** value (which acts similarly to a nonce).
- Look for fields labeled **Session ID Length**, **Session ID**, and **Random**.



```
         Length: 74
       ▾ Handshake Protocol: Server Hello
           Handshake Type: Server Hello (2)
           Length: 70
           Version: SSL 3.0 (0x0300)
         ▾ Random: 0000000042dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745
               GMT Unix Time: Jan  1, 1970 05:30:00.000000000 India Standard Time
               Random Bytes: 42dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745
           Session ID Length: 32
           Session ID: 1bad05faba02ea92c64c54be4547c32f3e3ca63d3a0c86ddad694b45682da22f
           Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)
           Compression Method: null (0)
           [JA3S Fullstring: 768,4,]
           [JA3S: 1f8f5a3d2fd435e36084db890693eafd]
     TLS segment data (1301 bytes)

Random values used for deriving keys (tls.handshake.random), 32 bytes
```

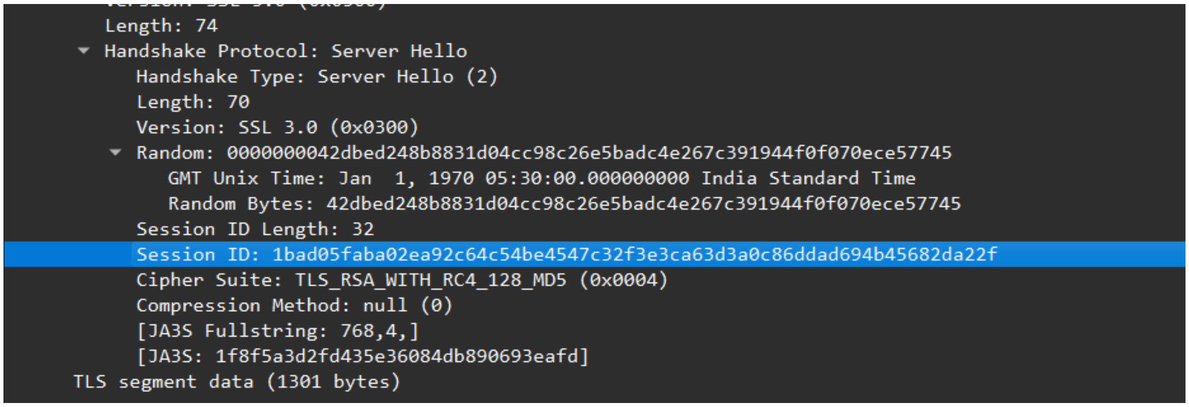
Purpose of Nonce in the ServerHello Record

1. **Session Uniqueness**:
    - Similar to the **ClientHello**, the **Server Random** value helps ensure that the session is unique. It differentiates this session from previous ones.
2. **Key Derivation**:
    - The **Server Random** value is combined with the **Client Random** value (from the **ClientHello**) during the key derivation process to create session keys for encrypting the data exchanged in the session. This ensures that the keys are unique for each session.

3. **Preventing Replay Attacks**:
    - Just as with the client, the server's nonce (or **Server Random**) helps protect against replay attacks, ensuring that each session is independent and cannot be reused maliciously.

8. Does this record include a session ID? What is the purpose of the session ID?

Answer YES



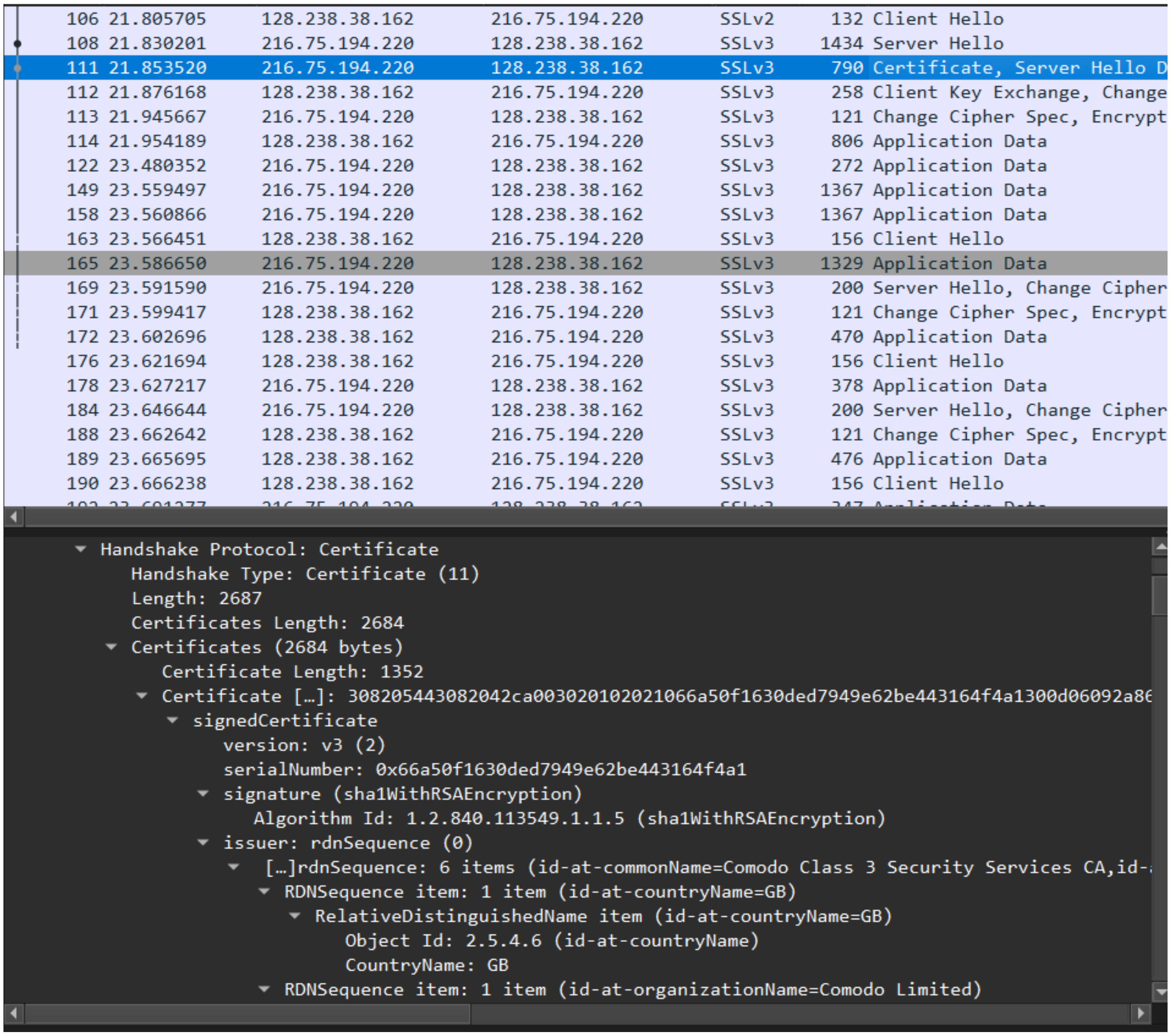
```
        Length: 74
      ▾ Handshake Protocol: Server Hello
          Handshake Type: Server Hello (2)
          Length: 70
          Version: SSL 3.0 (0x0300)
        ▾ Random: 0000000042dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745
            GMT Unix Time: Jan  1, 1970 05:30:00.000000000 India Standard Time
            Random Bytes: 42dbed248b8831d04cc98c26e5badc4e267c391944f0f070ece57745
          Session ID Length: 32
          Session ID: 1bad05faba02ea92c64c54be4547c32f3e3ca63d3a0c86ddad694b45682da22f
          Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)
          Compression Method: null (0)
          [JA3S Fullstring: 768,4,]
          [JA3S: 1f8f5a3d2fd435e36084db890693eafd]
    TLS segment data (1301 bytes)
```

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?

Answer: YES

After the **ServerHello** frame, there should be another frame labeled something like **Certificate**. This frame contains the actual server certificate sent by the server.

If the certificate size is less than or equal to 1500 bytes, it will fit into a single Ethernet frame. If it exceeds this size, it will be fragmented across multiple frames.

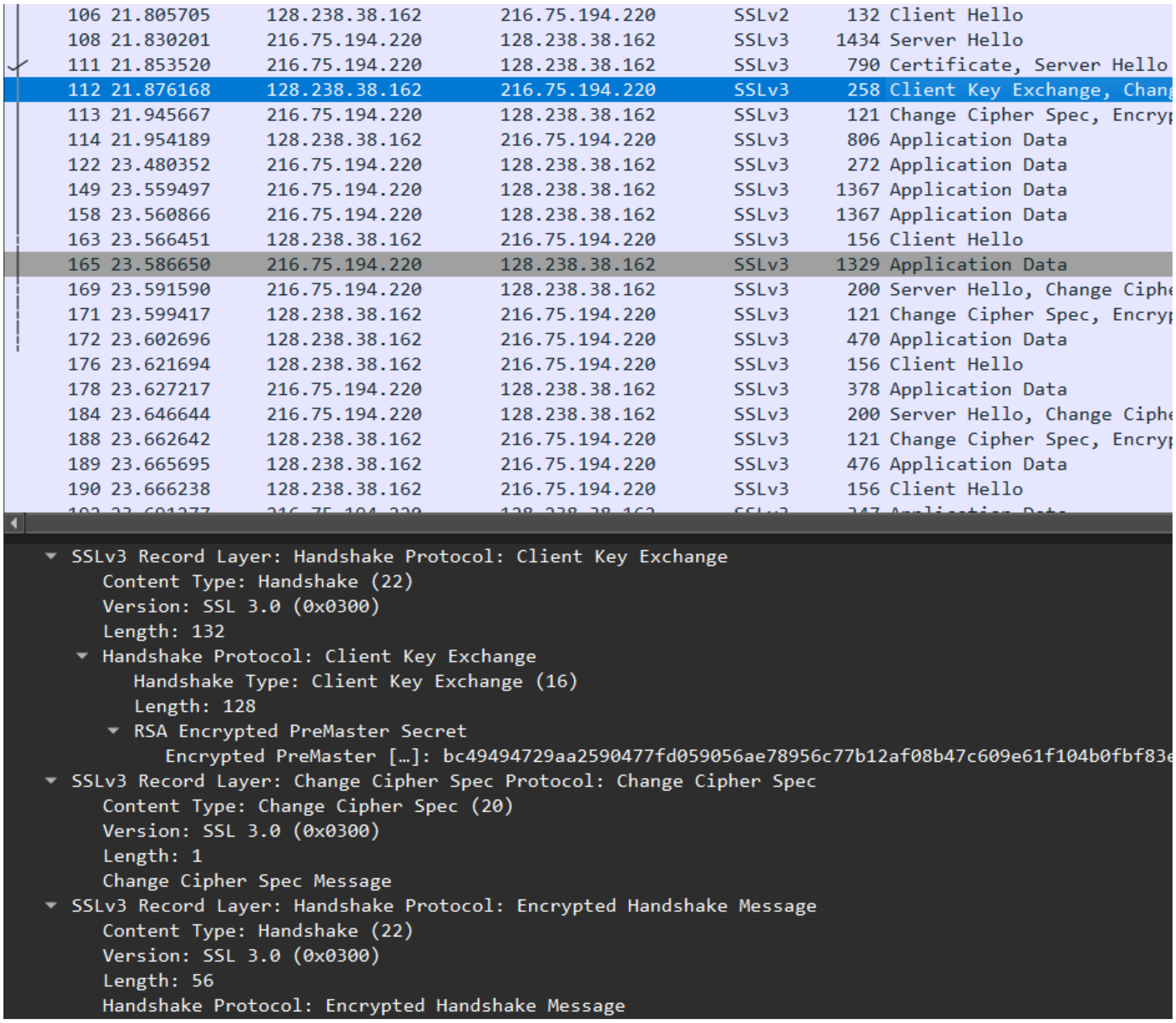


```
106 21.805705   128.238.38.162   216.75.194.220   SSLv2    132 Client Hello
108 21.830201   216.75.194.220   128.238.38.162   SSLv3   1434 Server Hello
111 21.853520   216.75.194.220   128.238.38.162   SSLv3    790 Certificate, Server Hello D
112 21.876168   128.238.38.162   216.75.194.220   SSLv3    258 Client Key Exchange, Change
113 21.945667   216.75.194.220   128.238.38.162   SSLv3    121 Change Cipher Spec, Encrypt
114 21.954189   128.238.38.162   216.75.194.220   SSLv3    806 Application Data
122 23.480352   216.75.194.220   128.238.38.162   SSLv3    272 Application Data
149 23.559497   216.75.194.220   128.238.38.162   SSLv3   1367 Application Data
158 23.560866   216.75.194.220   128.238.38.162   SSLv3   1367 Application Data
163 23.566451   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
165 23.586650   216.75.194.220   128.238.38.162   SSLv3   1329 Application Data
169 23.591590   216.75.194.220   128.238.38.162   SSLv3    200 Server Hello, Change Cipher
171 23.599417   128.238.38.162   216.75.194.220   SSLv3    121 Change Cipher Spec, Encrypt
172 23.602696   128.238.38.162   216.75.194.220   SSLv3    470 Application Data
176 23.621694   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
178 23.627217   216.75.194.220   128.238.38.162   SSLv3    378 Application Data
184 23.646644   216.75.194.220   128.238.38.162   SSLv3    200 Server Hello, Change Cipher
188 23.662642   128.238.38.162   216.75.194.220   SSLv3    121 Change Cipher Spec, Encrypt
189 23.665695   128.238.38.162   216.75.194.220   SSLv3    476 Application Data
190 23.666238   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
```

```
▾ Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 2687
    Certificates Length: 2684
  ▾ Certificates (2684 bytes)
      Certificate Length: 1352
    ▾ Certificate […]: 308205443082042ca003020102021066a50f1630ded7949e62be443164f4a1300d06092a86
      ▾ signedCertificate
          version: v3 (2)
          serialNumber: 0x66a50f1630ded7949e62be443164f4a1
        ▾ signature (sha1WithRSAEncryption)
            Algorithm Id: 1.2.840.113549.1.1.5 (sha1WithRSAEncryption)
        ▾ issuer: rdnSequence (0)
          ▾  […]rdnSequence: 6 items (id-at-commonName=Comodo Class 3 Security Services CA,id-
            ▾ RDNSequence item: 1 item (id-at-countryName=GB)
              ▾ RelativeDistinguishedName item (id-at-countryName=GB)
                  Object Id: 2.5.4.6 (id-at-countryName)
                  CountryName: GB
            ▾ RDNSequence item: 1 item (id-at-organizationName=Comodo Limited)
```

Client Key Exchange Record:

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?



```
106 21.805705   128.238.38.162   216.75.194.220   SSLv2    132 Client Hello
108 21.830201   216.75.194.220   128.238.38.162   SSLv3   1434 Server Hello
111 21.853520   216.75.194.220   128.238.38.162   SSLv3    790 Certificate, Server Hello
112 21.876168   128.238.38.162   216.75.194.220   SSLv3    258 Client Key Exchange, Chang
113 21.945667   216.75.194.220   128.238.38.162   SSLv3    121 Change Cipher Spec, Encryp
114 21.954189   128.238.38.162   216.75.194.220   SSLv3    806 Application Data
122 23.480352   216.75.194.220   128.238.38.162   SSLv3    272 Application Data
149 23.559497   216.75.194.220   128.238.38.162   SSLv3   1367 Application Data
158 23.560866   216.75.194.220   128.238.38.162   SSLv3   1367 Application Data
163 23.566451   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
165 23.586650   216.75.194.220   128.238.38.162   SSLv3   1329 Application Data
169 23.591590   216.75.194.220   128.238.38.162   SSLv3    200 Server Hello, Change Ciphe
171 23.599417   128.238.38.162   216.75.194.220   SSLv3    121 Change Cipher Spec, Encryp
172 23.602696   128.238.38.162   216.75.194.220   SSLv3    470 Application Data
176 23.621694   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
178 23.627217   216.75.194.220   128.238.38.162   SSLv3    378 Application Data
184 23.646644   216.75.194.220   128.238.38.162   SSLv3    200 Server Hello, Change Ciphe
188 23.662642   128.238.38.162   216.75.194.220   SSLv3    121 Change Cipher Spec, Encryp
189 23.665695   128.238.38.162   216.75.194.220   SSLv3    476 Application Data
190 23.666238   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
192 23.691277   216.75.194.220   128.238.38.162   SSLv3    347 Application Data

▾ SSLv3 Record Layer: Handshake Protocol: Client Key Exchange
    Content Type: Handshake (22)
    Version: SSL 3.0 (0x0300)
    Length: 132
  ▾ Handshake Protocol: Client Key Exchange
      Handshake Type: Client Key Exchange (16)
      Length: 128
    ▾ RSA Encrypted PreMaster Secret
        Encrypted PreMaster […]: bc49494729aa2590477fd059056ae78956c77b12af08b47c609e61f104b0fbf83e
▾ 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
```

Presence of Pre-Master Secret: The **Client Key Exchange** record does contain the pre-master secret, which is crucial for establishing session keys.

Purpose of the Pre-Master Secret: The pre-master secret is used to derive symmetric session keys that will encrypt the data exchanged between the client and server after the handshake is complete.

Encryption: The pre-master secret is typically encrypted with the server's public key, ensuring that only the server can decrypt it using its private key.

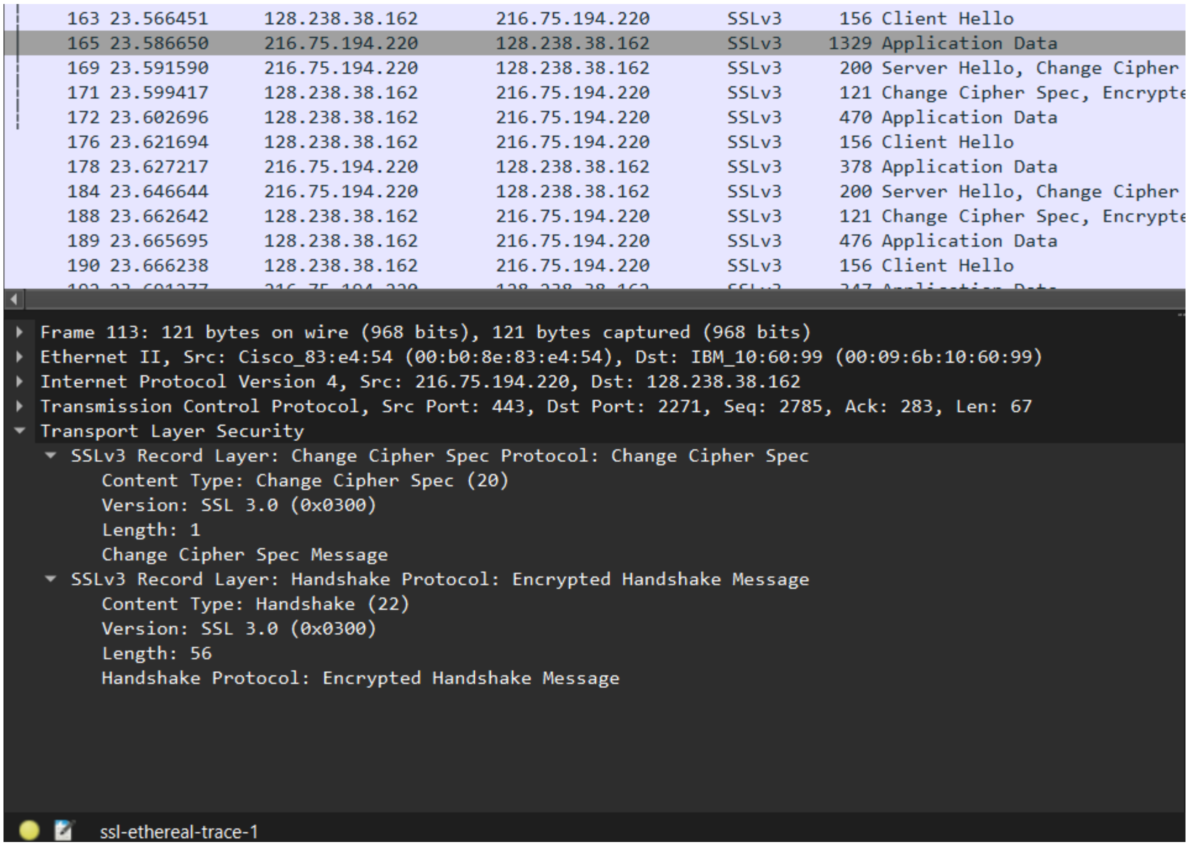
Length of the Encrypted Secret: The length of the encrypted pre-master secret is usually around 128 bytes but can vary based on the cipher suite and specific implementation.

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

11. What is the purpose of the Change Cipher Spec record? How many bytes is the record in your trace?

Purpose: The Change Cipher Spec record indicates that the sender is ready to switch to encrypted communication using the new cipher suite and keys.

Size: The record is generally **2 bytes** in total (1 byte for the content type and 1 byte for the Change Cipher Spec message itself).



```
163 23.566451   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
165 23.586650   216.75.194.220   128.238.38.162   SSLv3   1329 Application Data
169 23.591590   216.75.194.220   128.238.38.162   SSLv3    200 Server Hello, Change Cipher
171 23.599417   128.238.38.162   216.75.194.220   SSLv3    121 Change Cipher Spec, Encrypte
172 23.602696   128.238.38.162   216.75.194.220   SSLv3    470 Application Data
176 23.621694   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
178 23.627217   216.75.194.220   128.238.38.162   SSLv3    378 Application Data
184 23.646644   216.75.194.220   128.238.38.162   SSLv3    200 Server Hello, Change Cipher
188 23.662642   128.238.38.162   216.75.194.220   SSLv3    121 Change Cipher Spec, Encrypte
189 23.665695   128.238.38.162   216.75.194.220   SSLv3    476 Application Data
190 23.666238   128.238.38.162   216.75.194.220   SSLv3    156 Client Hello
```

```
▸ Frame 113: 121 bytes on wire (968 bits), 121 bytes captured (968 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: 2785, Ack: 283, Len: 67
▾ Transport Layer Security
  ▾ 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
```

12. In the encrypted handshake record, what is being encrypted? How?

What is Encrypted: Handshake messages exchanged during the SSL/TLS handshake process.

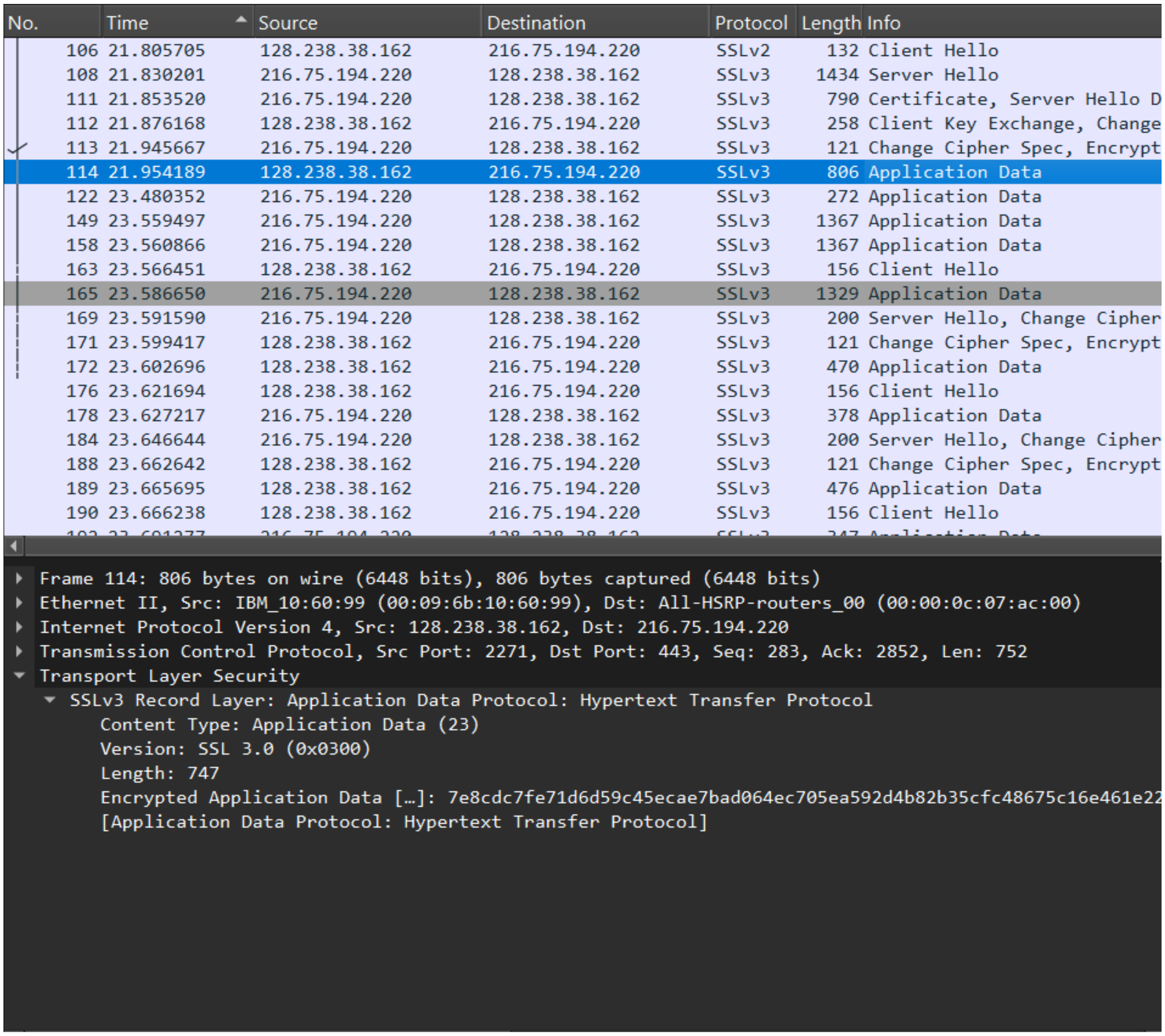
How it is Encrypted: Using symmetric-key algorithms determined by the negotiated cipher suite, leveraging session keys derived from the pre-master secret. The messages are encrypted and often accompanied by a MAC for integrity and authenticity.

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?

Both the client and server send Change Cipher Spec and encrypted handshake records.

The **Change Cipher Spec records** signify readiness for encrypted communication.

The **encrypted handshake records** finalize the handshake and vary in content based on whether they originate from the client or the server, with each party indicating completion of their Application Data



```
No.    Time        Source          Destination     Protocol Length Info
   106 21.805705   128.238.38.162  216.75.194.220  SSLv2     132 Client Hello
   108 21.830201   216.75.194.220  128.238.38.162  SSLv3    1434 Server Hello
   111 21.853520   216.75.194.220  128.238.38.162  SSLv3     790 Certificate, Server Hello D
   112 21.876168   128.238.38.162  216.75.194.220  SSLv3     258 Client Key Exchange, Change
   113 21.945667   216.75.194.220  128.238.38.162  SSLv3     121 Change Cipher Spec, Encrypt
   114 21.954189   128.238.38.162  216.75.194.220  SSLv3     806 Application Data
   122 23.480352   216.75.194.220  128.238.38.162  SSLv3     272 Application Data
   149 23.559497   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
   158 23.560866   216.75.194.220  128.238.38.162  SSLv3    1367 Application Data
   163 23.566451   128.238.38.162  216.75.194.220  SSLv3     156 Client Hello
   165 23.586650   216.75.194.220  128.238.38.162  SSLv3    1329 Application Data
   169 23.591590   216.75.194.220  128.238.38.162  SSLv3     200 Server Hello, Change Cipher
   171 23.599417   128.238.38.162  216.75.194.220  SSLv3     121 Change Cipher Spec, Encrypt
   172 23.602696   128.238.38.162  216.75.194.220  SSLv3     470 Application Data
   176 23.621694   128.238.38.162  216.75.194.220  SSLv3     156 Client Hello
   178 23.627217   216.75.194.220  128.238.38.162  SSLv3     378 Application Data
   184 23.646644   216.75.194.220  128.238.38.162  SSLv3     200 Server Hello, Change Cipher
   188 23.662642   128.238.38.162  216.75.194.220  SSLv3     121 Change Cipher Spec, Encrypt
   189 23.665695   128.238.38.162  216.75.194.220  SSLv3     476 Application Data
   190 23.666238   128.238.38.162  216.75.194.220  SSLv3     156 Client Hello
   192 23.691277   216.75.194.220  128.238.38.162  SSLv3     347 Application Data
```

```
> Frame 114: 806 bytes on wire (6448 bits), 806 bytes captured (6448 bits)
> 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 Version 4, Src: 128.238.38.162, Dst: 216.75.194.220
> Transmission Control Protocol, Src Port: 2271, Dst Port: 443, Seq: 283, Ack: 2852, Len: 752
v Transport Layer Security
  v SSLv3 Record Layer: Application Data Protocol: Hypertext Transfer Protocol
      Content Type: Application Data (23)
      Version: SSL 3.0 (0x0300)
      Length: 747
      Encrypted Application Data […]: 7e8cdc7fe71d6d59c45ecae7bad064ec705ea592d4b82b35cfc48675c16e461e22
      [Application Data Protocol: Hypertext Transfer Protocol]
```

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?

Encryption of Application Data:

- In SSL/TLS, application data is encrypted using the symmetric encryption algorithms agreed upon during the handshake process. After the Change Cipher Spec record has been exchanged, both the client and server use the session keys derived from the pre-master secret to encrypt and decrypt application data.
- The specific symmetric encryption algorithm (such as AES, DES, etc.) is part of the cipher suite chosen during the handshake.

Inclusion of MAC (Message Authentication Code):

- Yes, the records containing application data include a MAC. The MAC is calculated over the plaintext data (the application data) along with additional information like sequence numbers and the session keys.
- The MAC serves as a form of integrity check, ensuring that the data has not been tampered with during transmission.

Wireshark Distinction:

- In Wireshark, encrypted application data and the MAC are typically bundled together in the same record. However, Wireshark does display a breakdown of the decrypted application data, allowing users to view the plaintext contents after decryption.
- If the application data is decrypted (for instance, if the session keys are available to Wireshark), the MAC may not be separately shown in the decrypted data, as it is used internally to verify integrity but does not need to be displayed in the application layer.

15. Comment on and explain anything else that you found interesting in the trace.

Use of Different SSL Versions:

The trace indicates a transition from SSLv2 to SSLv3. It's interesting to note the evolution of the SSL protocol versions, as SSLv2 is considered outdated and insecure. Modern applications primarily use TLS, which is the successor to SSL. The presence of SSLv2 could indicate compatibility settings or legacy systems.

Cipher Suite Negotiation:

The ClientHello message lists multiple cipher suites supported by the client. The server chooses one from this list for the session, which can reveal insights into the security posture and configurations of both the client and server. Observing this negotiation process can be critical for understanding potential vulnerabilities.

Challenge and Nonce Usage:

The ClientHello message includes a nonce (challenge), which is a random value used to prevent replay attacks. This is an interesting feature of SSL/TLS that enhances security by ensuring that each session is unique. The presence of nonces shows the protocols' design to handle specific security threats effectively.

Certificate Exchange:

The certificate exchange step during the ServerHello message and subsequent records is crucial for establishing trust. This trace shows the server providing its certificate, which may be signed by a trusted Certificate Authority (CA). The ability to verify this certificate is essential for the client to ensure that it is communicating with the legitimate server.

Packet Sizes and Performance:

Analyzing the sizes of the packets in the trace could provide insights into network performance. Larger packets may indicate bulk data transfers, while smaller packets might signify many small transactions. Identifying patterns in packet sizes could help in optimizing application performance and network resource utilization.

Timing of Records:

Observing the timing between records can provide insights into latency and performance issues. For example, if there are significant delays between the ClientHello and ServerHello messages, it could indicate network congestion or processing delays.

Application Data Records:

The presence of application data records after the handshake signals that secure communication has commenced. Analyzing the types of application data exchanged can provide insights into the nature of the application traffic, whether it's HTTP requests, file transfers, etc.

Network Security Considerations:

The trace can help identify potential security concerns, such as unencrypted traffic, or weak cipher suites. It is important to ensure that strong encryption practices are followed, as vulnerabilities in these areas could lead to exposure of sensitive data.

These points provide a deeper understanding of the SSL handshake process and the resulting secure communication, illustrating both the complexity and importance of cryptographic protocols in modern network security.