TLS:
WIRESHARK LABS

Corso di Introduzione alla Sicurezza Informatica

Chiara Luchini

Anno Accademico 2023-2024

How to

- Download Wireshark: https://www.wireshark.org/download.html
- Download pcap files:

  http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces-8.1.zip
- Follow TLS Lab:

  http://www-net.cs.umass.edu/wireshark-labs/Wireshark_TLS_v8.1.doc
- Open pcap file *tls-wireshark-trace1.pcapng* in Wireshark
- Enter *ip.addr == 128.119.240.84* in Wireshark's display filter window

TCP connection

1. **What is the packet number in your trace that contains the initial TCP SYN message?** (By “packet number,” we meant the number in the “No.” column at the left of the Wireshark display, not the sequence number in the TCP segment itself)

The packet number that contains the initial TCP SYN is No. 17

TCP connection

2. **Is the TCP connection set up before or after the first TLS message is sent from client to server?**

The TCP connection is set before the first TLS message because HTTPS implements TLS running “over” TCP. That means that a TCP connection **must first be established** between your browser and the web server for www.cics.umass.edu before TLS and HTTP messages can be exchanged.

The TLS Handshake: Client Hello message

3. **What is the packet number in your trace that contains the TLS Client Hello message?**

The packet number that contains the TLS Client Hello message is No. 28

The TLS Handshake: Client Hello message

4. **What version of TLS is your client running, as declared in the Client Hello message?**

The TLS version is TLS 1.2

The TLS Handshake: Client Hello message

5. **How many cipher suites are supported by your client, as declared in the Client Hello message?**

A cipher suite is a set of related cryptographic algorithms that determine how session keys will be derived, and how data will be encrypted and be digitally signed via a HMAC algorithm

There are 17 cipher suites supported by the client.

The TLS Handshake: Client Hello message

6. **Your client generates and sends a string of “random bytes” to the server in the Client Hello message. What are the first two hexadecimal digits in the random bytes field of the Client Hello message?**

The first two hexadecimal digits in the random bytes field of the Client Hello message are:
4b909a780b955b1a679367e8af0312ec2362979794c50c162089004b

The TLS Handshake: Server Hello message

7. **What is the purpose(s) of the “random bytes” field in the Client Hello message?**

The purpose of the client and server **nonces** in TLS is to prevent attacker from **replaying or reordering records**.
The randomness helps in this to prevent attackers from simulating a bunch of sessions beforehand and then picking the relevant one for you.
Moreover, client/server generate a **master secret** from the **premaster secret** and exchanged random values.

The TLS Handshake: Server Hello message

8. **What is the packet number in your trace that contains the TLS Server Hello message?**

The packet number that contains the TLS Server Hello message is No. 32

The TLS Handshake: Server Hello message

9. **Which cipher suite has been chosen by the server from among those offered in the earlier Client Hello message?**

The cipher suite chosen by the server is:
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)

The TLS Handshake: Server Hello message

10. **Does the Server Hello message contain random bytes, similar to how the Client Hello message contained random bytes? And if so, what is/are their purpose(s)?**

Yes, it contains a Random Bytes field. Its value is:
5c08b35ca6b696fcd26eaf9a275f67f37730fa82d5a570809ef8ab9f
The reason and purpose is the same as the client.

The TLS Handshake: Server Hello message

11. **What is the packet number in your trace for the TLS message part that contains the public key certificate for the www.cics.umass.edu server (actually the www.cs.umass.edu server)?**

The packet number that contains the public key certificate is No. 37

The TLS Handshake: Server Hello message

12. **A server may return more than one certificate. If more than one certificate is returned, are all of these certificates for www.cs.umass.edu? If not all are for www.cs.umass.edu, then who are these other certificates for?**

There are 3 certificates returned by the server and they are for:

- www.cs.umass.edu
- InCommon RSA Server CA
- USERTrust RSA Certification Authority

The TLS Handshake: Server Hello message

13. **What is the name of the certification authority that issued the certificate for id-at-commonName=www.cs.umass.edu?**

The CA is InCommon RSA Server CA

The TLS Handshake: Server Hello message

14. **What digital signature algorithm is used by the CA to sign this certificate?**

The digital signature algorithm is sha256WithRSAEncryption

The TLS Handshake: Server Hello message

15. **Let's take a look at what a real public key looks like! What are the first four hexadecimal digits of the modulus of the public key being used by www.cics.umass.edu?**

The first four hexadecimal digits of the modulus of the public key are:
00b39e7296158da80176a2f1035c7c61f06120f9852aad0d20d4931a30842fecec1b8724a7315b71
bd377bd80da05ca9babe115b1e498e8c66e2db98964d859af88fceee08e13679106029d868a3a
637d75148360d9c436743ce6d3344903df8bd5358df61d8f7ecfc906880df668fc06090391f35b
00111f8fca3ae8594db305ce83a8861161ec3e66f2558d0ee0d9dc7ca92ac3265cc2893edf25562
a6190ce4af95cbac219761b7b8ab695837280c4005e6d4b42cc88a73a556c2cb136ef35c5ad10e
adce6662a637dbdd0f4b9e37b2ebf8b8fc02c9969fae7ad1a7407f7bc071eb17a1ec9b0bbde45a
40ab304893a367b2dbda3504056b6c3ad506dc879d29

The TLS Handshake: Server Hello message

16. **Look in your trace to find messages between the client and a CA to get the CA's public key information, so that the client can verify that the CA-signed certificate sent by the server is indeed valid and has not been forged or altered. Do you see such message in your trace? If so, what is the number in the trace of the first packet sent from your client to the CA? If not, explain why the client did not contact the CA.**

No there isn't a message between client and CA. This is because usually a list of CA is included inside the web browser for example this is a list of CA for Mozilla: https://ccadb.my.salesforce-sites.com/mozilla/IncludedCACertificateReport
A client can find CA's public key info in the **Subject + Subject Public Key Info (SPKI) SHA256** field.

The TLS Handshake: Server Hello message

17. **What is the packet number in your trace for the TLS message part that contains the Server Hello Done TLS record?**

The same as the Certificate one so No.37

The TLS Handshake: wrapping up the handshake

18. **What is the packet number in your trace for the TLS message that contains the public key information, Change Cipher Spec, and Encrypted Handshake message, being sent from client to server?**

The packet number is No.39

The TLS Handshake: wrapping up the handshake

19. **Does the client provide its own CA-signed public key certificate back to the server? If so, what is the packet number in your trace containing your client’s certificate?**

No there isn’t.

The TLS Handshake: wrapping up the handshake

20. **What symmetric key cryptography algorithm is being used by the client and server to encrypt application data (in this case, HTTP messages)?**

The application data is encrypted using the specified algorithms in the chosen cipher suite:

- RSA (public-key),
- 128-bit GCM AES (symmetric)
- SHA-256 (hash algorithm)

The TLS Handshake: wrapping up the handshake

21. **In which of the TLS messages is this symmetric key cryptography algorithm finally decided and declared?**

The symmetric key cryptography algorithm is decided and declared in the "ClientKeyExchange" message.

The TLS Handshake: wrapping up the handshake

22. **What is the packet number in your trace for the first encrypted message carrying application data from client to server?**

The packet number is No. 41

The TLS Handshake: wrapping up the handshake

23. **What do you think the content of this encrypted application-data is, given that this trace was generated by fetching the homepage of www.cics.umass.edu?**

The application-data could be the html of the web page, images inside the web page and so on.

The TLS Handshake: wrapping up the handshake

24. **What packet number contains the client-to-server TLS message that shuts down the TLS connection?**

The way a TLS connection is closed normally is by sending an alert record of type **close_notify** in at least one direction, followed by a normal TCP close which sends and **ACKs FIN** in both directions using a variable number and sequence of packets depending on the dynamic state and timing. Unless some other fatal alert has been transmitted, **each party** is required to send a **close_notify alert** before closing the write side of the connection. The other party MUST respond with a close_notify alert of its own and close down the connection immediately, discarding any pending writes. It is not required for the initiator of the close to wait for the responding close_notify alert before closing the read side of the connection.

All alerts after the handshake completes are encrypted, so wireshark only decodes as **Encrypted Alert.** In principle several alerts are possible, but in practice if you see an alert after normal data exchange it's extremely likely to be close_notify.

References

- Kurose, J. F., & Ross, K. W. (2021). *Computer networking : a top-down approach* (8th ed.). Pearson.
- Rescorla, E., & Dierks, T. (2008, August). *The Transport Layer Security (TLS) Protocol Version 1.2*. IETF. https://datatracker.ietf.org/doc/html/rfc5246