CS 315: Computer Networks Lab
Spring 2024-25, IIT Dharwad
Assignment-12

Wireshark Lab: 802.11 WiFi & TLS

April 7, 2025

Lab Instructions

- Mark your attendance in the attendance sheet before leaving the lab.
- Handle the lab resources with utmost care.
- Please go through the following exercises in today's lab.
- It is recommended that you complete all the following exercises during the lab slot itself.
- If you face any difficulties, please feel free to seek help online or from your peers or TAs.
- After finishing all exercises, please carry your solutions with you (via email/pen drive) for future reference, and delete the files from the desktop.

Part 0: Paste a screenshot of your system IP address, using ipconfig (on Windows) or ifconfig (on Mac and Linux), and fill out this Google form to submit the details of your system. The same system must be used to attempt all exercises of this lab.

Introduction

In this lab, we'll investigate the 802.11 wireless network protocol. In all of the Wireshark labs thus far, we've captured frames on a wired Ethernet connection. Here, since 802.11 is a wireless link-layer protocol, we'll be capturing frames "in the air." Unfortunately, some device drivers for wireless 802.11 NICs still don't provide the hooks to capture/copy received 802.11 frames for use in Wireshark. Thus, in this lab, we'll provide a trace of captured 802.11 frames for you to analyze and assume in the questions below that you are using this trace.

Getting Started

Download the file `WiFi_Trace.pcap` from the link. This trace was collected using AirPcap and Wireshark running on a computer in the home network of one of the authors, consisting of a Linksys 802.11g combined access point/router, with two wired PCs and one wireless host PC attached to the access point/router. The author is fortunate to have other access points in neighbouring houses available as well. In this trace file, we'll see frames captured on channel 6. Since the host and AP that we are interested in are not the only devices using channel 6, we'll see a lot of frames that we're not interested in for this lab, such as beacon frames advertised by a neighbour's AP also operating on channel 6. The recorded wireless host activities in the trace file are as follows:

- At the start of the trace, the host is already connected to the *30 Munroe St access point (AP)*.

- At t = 24.82, the host sends an HTTP request to `http://gaia.cs.umass.edu/wireshark-labs/alice.txt`, with the destination IP address 128.119.245.12.
- At t = 32.82, the host makes another HTTP request to `http://www.cs.umass.edu`, which resolves to 128.119.240.19.
- At t = 49.58, the host disconnects from the *30 Munroe St AP* and attempts to connect to *linksys_ses_24086*, a secured access point. However, the connection attempt is unsuccessful.
- At t = 63.0, after failing to associate with *linksys_ses_24086*, the host reconnects to the *30 Munroe St AP*.

Part-1: Beacon Frames

Recall that beacon frames are used by an 802.11 AP to advertise its existence. To answer some of the questions below, you'll want to look at the details of the "IEEE 802.11" frame and subfields in the middle Wireshark window.

1. What are the SSIDs of the two access points that are issuing most of the beacon frames in this trace?
2. What are the 'beacon interval' field values in the *linksys_ses_24086* access point and the *30 Munroe St*. access point?
3. What (in hexadecimal notation) is the source MAC address on the beacon frame from *30 Munroe St*?
4. What (in hexadecimal notation) is the destination MAC address on the beacon frame from *30 Munroe St*?
5. What (in hexadecimal notation) is the MAC BSS id on the beacon frame from *30 Munroe St*?
6. The beacon frames from the *30 Munroe St* access point advertise that the access point can support four data rates and eight additional "extended supported rates." What are these rates?

Part-2: Data Transfer

Since the trace starts with the host already associated with the AP, let's first look at data transfer over an 802.11 association before looking at AP association/disassociation. Recall that in this trace, at *t = 24.82*, the host makes an HTTP request to `http://gaia.cs.umass.edu/wireshark-labs/alice.txt`. The IP address of `gaia.cs.umass.edu` is 128.119.245.12. Then, at *t=32.82,* the host makes an HTTP request to `http://www.cs.umass.edu`.

1. Find the 802.11 frame containing the SYN TCP segment for this first TCP session (that downloads `alice.txt`). What are three MAC address fields in the 802.11 frame? Which MAC address in this frame corresponds to the wireless host (give the hexadecimal representation of the MAC address for the host)? To the access point? To the first-hop router? What is the IP address of the wireless host sending this TCP segment? What is the destination IP address?
2. Find the 802.11 frame containing the SYNACK segment for this TCP session. What are three MAC address fields in the 802.11 frame? Which of these are the MAC addresses corresponding to the host sending SYNACK, destination and BSS? What is the IP address of the server sending the TCP SYNACK?

Part-3: Association/Disassociation

In the text, a host must first *associate* with an access point before sending data. Association in 802.11 is performed using the ASSOCIATE REQUEST frame (sent from the host to AP, with a frame type 0 and subtype 0) and the ASSOCIATE RESPONSE frame (sent by the AP to a host with a frame type 0 and subtype of 1, in response to a received ASSOCIATE REQUEST).

1. What two actions are taken (i.e., frames are sent) by the host in the trace just after *t=49*, to end the association with the *30 Munroe St* AP that was initially in place when trace collection began? (Hint: one is an IP-layer action, and one is an 802.11-layer action). Looking at the 802.11 specification, is there another frame that you might have expected to see, but don't see here?
2. Examine the trace file and look for AUTHENTICATION frames sent from the host to an AP and vice versa. How many AUTHENTICATION messages are sent from the wireless host to the *linksys_ses_24086* AP (which has a MAC address of Cisco_Li_f5:ba:bb) starting at around *t=49?*
3. Does the host want the authentication to require a key or be open?
4. Do you see a reply AUTHENTICATION from the *linksys_ses_24086* AP in the trace?
5. Now let's consider what happens as the host gives up trying to associate with the *linksys_ses_24086* AP and now tries to associate with the *30 Munroe St* AP. Look for AUTHENTICATION frames sent from the host to an AP and vice versa. At what times is there an AUTHENTICATION frame from the host to *30 Munroe St.* AP, and when is there a reply AUTHENTICATION sent from that AP to the host reply? (Note that you can use the filter expression "wlan.fc.subtype == 11 && wlan.fc.type **==** 0 && wlan.addr **==** 00:13:02:d1:b6:4f" to display only the AUTHENTICATION frames in this trace for this wireless host.)
6. An ASSOCIATE REQUEST from the host to AP and a corresponding ASSOCIATE RESPONSE frame from AP to the host is used for the host to be associated with an AP. At what time is there an ASSOCIATE REQUEST from the host to *30 Munroe St* AP? When is the corresponding ASSOCIATE REPLY sent? (Note that you can use the filter expression "`wlan.fc.subtype < 2 && wlan.fc.type == 0 && wlan.addr == 00:13:02:d1:b6:4f`" to display only the ASSOCIATE REQUEST and ASSOCIATE RESPONSE frames for this trace.)
7. What transmission rates is the host willing to use? The AP? To answer this question, you will need to look into the parameter fields of the 802.11 wireless LAN management frame.

Part-4: Other Frame types

Our trace contains a number of PROBE REQUEST and PROBE RESPONSE frames.

1. What are the sender, receiver and BSS ID MAC addresses in these frames? What is the purpose of these two types of frames?

TLS

Introduction

In this lab, we'll investigate Transport Layer Security (known as TLS) and aspects of the authentication, data integrity, and confidentiality services provided by TLS. TLS is the successor to the now-deprecated Secure Sockets Layer (SSL).

We'll investigate TLS by analyzing a Wireshark packet trace captured during the retrieval of a web page via HTTPS - a secure version of HTTP, which implements TLS on top of HTTP. We'll look at TLS's client-server handshaking protocol in some detail since that's where most of the interesting action happens. You may use online resources for learning more about TLS here, here, and here; and, of course, in RFC 5246.

Capturing packets in a TLS session

The first step in this lab is to capture the packets in a TLS session. To do this, you should start Wireshark and begin packet capture, retrieve the homepage from https://www.cics.umass.edu using the browser of your choice, and then stop Wireshark packet capture. The 's' after 'http' will cause the **Hypertext Transfer Protocol Secure** (**HTTPS**) – an extension of HTTP – to be used to securely retrieve the homepage from `www.cics.umass.edu`. Here, "securely" means that the `www.cics.umass.edu` server will be authenticated by your web browser, that the transmission of your client HTTP GET request and the server's reply will be encrypted, and the integrity of all message content will be cryptographically verified. Of course, the authentication, integrity and encryption of a computer science department's web page may not be as critical as that for Internet commerce and banking sites, but the same TLS protocol and TLS messages are used in all cases.

Part-5: A first look at the captured trace

Let's first set Wireshark's display so that only the packets to and from `www.cics.umass.edu`, are displayed.

It's important to keep in mind that an Ethernet frame (containing an IP datagram containing a TCP segment) may contain one or more TLS records. (This is very different from HTTP, for which each frame contains either one complete HTTP message or a portion of an HTTP message.) Also, a TLS record may not completely fit into an Ethernet frame, in which case multiple frames will be needed to carry the record.

We've said earlier that 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, just as we saw with the vanilla (non-TLS) HTTP protocol.

Answer the following questions:

1. What is the IP address of the domain `www.cics.umass.edu`?
2. Does your system set up a TCP connection with the server of `www.cics.umass.edu`? Provide the packet numbers of the three-way handshake, with the corresponding screenshot.
3. Is the TCP connection set up before or after the first TLS message is sent from the client to the server?

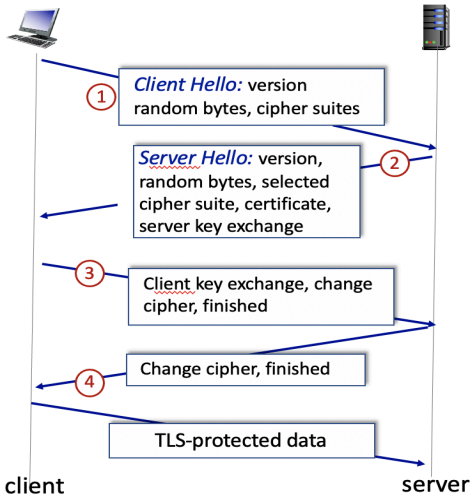

Figure 2: TLS handshake

4. What TLS version is used by the `www.cics.umass.edu` server?
5. List the various TLS messages between the client and the server?
6. In the `Client Hello` message, what are the two versions of the TLS, and are they different and why?
7. Which field in the `Client Hello` message has the value of the domain? Is it encrypted or in plaintext?
8. Traverse through the '`Extension:`' parameters of the Client Hello message and provide the field name that contains the version of the HTTP, and what is the HTTP version?
9. List all the cipher suites the client offers the server to choose from for further communication.

Now, analyse the corresponding `Server Hello` message and answer the following questions.

10. Which TLS version has the server agreed on to set the TLS handshake?
11. Which cipher suite has the server agreed on for further communication?

Analyse the `Certificate`, `Server Key Exchange`, `Server Hello Done` packet to answer the following questions.

12. What is the size (in bytes) of the `Certificate`, `Server Key Exchange`, `Server Hello Done` messages?
13. Which certificate is agreed upon by the server?
14. Which handshake protocol is used in the `Server Key Exchange` message?

15. Which signature algorithm does the server send to the client?

Analyse the `Client Key Exchange`, `Change Cipher Spec`, `Finished` message and answer the following questions.

16. Does the client agree on the same handshake protocol?

Submission Details

- Write your answers in a single doc/tex file, and submit its PDF named after your IIT Dharwad roll number, which contains all answers (with screenshots, if necessary).