

Each group MUST complete and return this page to the tutor at the end of the lab.

GROUP NAME on Moodle: _____

GROUP Member 1

Student ID: _____

Full Name: _____

Signature: _____

GROUP Member 2

Student ID: _____

Full Name: _____

Signature: _____

If approved for a group of 3 fill below.

GROUP Member 3

Student ID: _____

Full Name: _____

*This is the **Instruction Sheet Document - Lab 3.***



COMP4337/9337

Lab 3: MITM Attack and Evil Twin AP

S1, 2019

Use eng.cse.COMP4337@unsw.edu.au for class communications
Students may use the Technical Questions Forum on Moodle to discuss this lab

Overview

This lab has two parts:

- **Part A:** MITM Attack on HTTP and HTTPS Traffic [Lab 2 Assignment 1]
- **Part B:** Evil Twin AP [Lab 2 Assignment 2]

During the lab:

- 1) Getting Started: We will be using Kali Linux VMWare images installed on lab machines.
 - a) To run Kali Linux, open a terminal, type “vm”, it will show you a list of VMs. Please choose the number for Comp4337
 - b) Login with user “root”, password “toor”
- 2) Each group is provided:
 - a) An ALFA wireless adapter which will be connected to your Kali Linux through a USB port on your laptop.
 - b) This “Instruction Sheet” document.
 - c) “Lab Assignment 1” and “Lab Assignment 2” will be available on Moodle.
- 3) **In the assignments you need to provide the commands you use in the lab. So, keep them. “history” command in terminal might come to help.**

At the end of the lab:

- 1) Return the first page, mentioning the Group and student Name and zID.
- 2) Students have 24 hours to complete Lab Assignment 1.
- 3) Students have 48 hours to complete Lab Assignment 2.

Marking of the labs

- 1) Labs are marked by lab tutors. The marks are submitted on Moodle and will be made available within 2 weeks of the lab date.
- 2) Breakdown:
 - a) Total mark for Lab 3 is 100. This lab combined with marks for other labs will be scaled to 20 out of 100.
 1. Part A: Lab Performance (25)
 2. Part B: Lab Assignment 1, submission on Moodle (25)
 3. Part B: Lab Assignment 2, submission on Moodle (50)
 - b) Students who do not attend the lab will lose ALL 100 marks for it.

Important:

Lab performance involves tutor asking question, feedback, and comment about the activity while the lab is in progress. Hence, if a group is found to be cheating or submitting a work for “Lab 3 Assignment 1” that does not match what the tutor observes of the team performance, then NO MARK will be awarded for Part A.

As mentioned, if a student submits “Lab 1 Assignment” after 15 minutes grace period, 5 marks for Lab performance is automatically deducted.

Part A –MITM Attack on HTTP and HTTPS Traffic [Lab 2 Assignment 1]

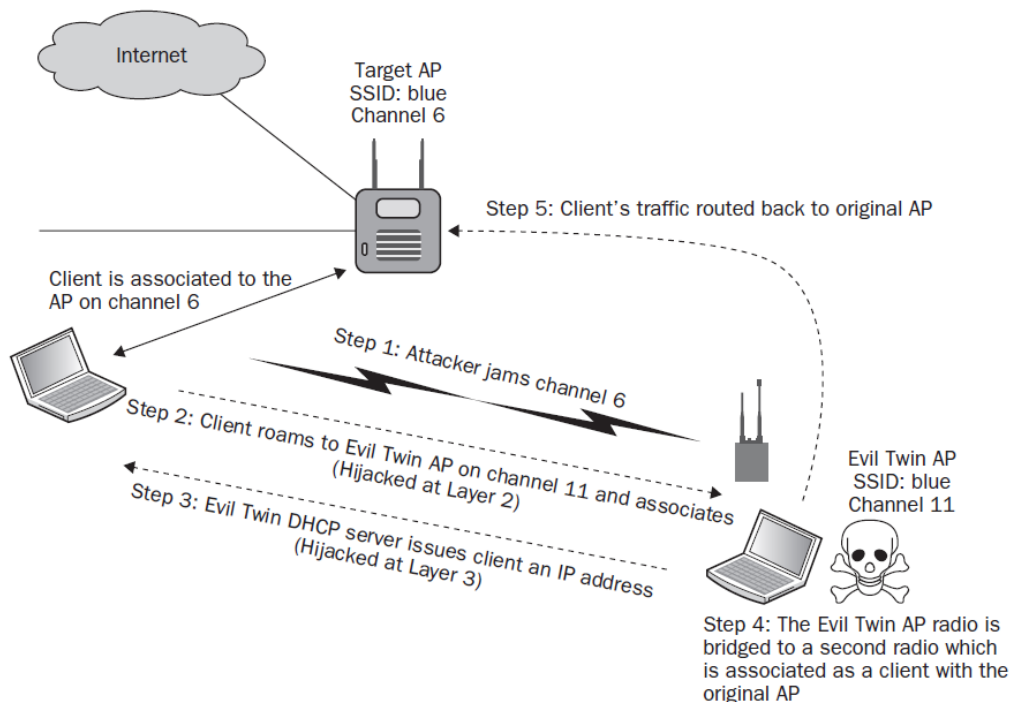


Figure 1: Wireless Hijacking/MITM Attack (Source: CWSP Book, Page: 318)

1. Prior to the attack the attacker configures access point on his laptop with client-card, with the same SSID as that of the Target AP (“blue” in above example). This way, the attacker’s access point is now functioning as an evil twin AP of the Target AP.
 - Note that, the evil twin will be transmitting on a different channel than the public hotspot. (Channel 11 vs Channel 6 in above example)
2. **(Step 2)** In order to force clients to leave the Target AP and join this new evil twin, attacker then sends spoofed disassociation or de-authentication frames, forcing client stations associated with the Target AP to roam to the evil twin access point
 - The attacker has hijacked the client stations at Layer 2.
 - Also note that, although de-authentication frames are usually used as one way to start a hijacking attack, **(Step 1)** RF jammers can also be used to force any clients to roam to an evil twin AP
3. **(Step 3)** The evil twin AP will typically be configured with a Dynamic Host Configuration Protocol (DHCP) server available to issue IP addresses to the clients.
 - The attacker has hijacked the client stations at Layer 3.
 - The user’s computer could, during the process of connecting to the evil twin, fall victim to the DHCP attack (An attack that exploits the DHCP process to dump root kits or other malware onto the victim’s computer in addition to giving them an IP address)
4. **(Step 4)** The attacker may also be using a second wireless card with their laptop to execute what is known as a man-in-the-middle attack, as shown in Figure 1.
 - The second WLAN card is associated with the original access point as a client.
 - The attacker has bridged together their second wireless card with the Wi-Fi card that is being used as the evil twin access point.
5. **(Step 5)** The traffic from the client is now routed from the evil twin access point through the second Wi-Fi card, right back to the original access point from which the users have just been hijacked.
 - The result is that the users remain hijacked; however, they still have a route back through the gateway to their original network, so they never know they have been hijacked.

The attacker can therefore sit in the middle and execute peer-to-peer attacks indefinitely while remaining completely unnoticed.

Mimproxy Overview:

Here is our network topology:

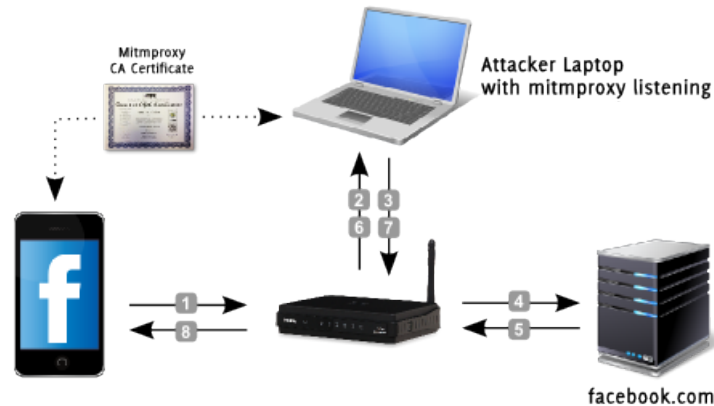


Figure 2: Our Network Topology

Mitmproxy is an open source proxy application that allows intercepting HTTP and HTTPS connections between any HTTP(S) client (such as a mobile or desktop browser) and a web server using a typical man-in-the-middle attack (MITM). Like other proxies (such as Squid), it accepts connections from clients and forwards them to the destination server. However, while other proxies typically focus on content filtering or speed optimization through caching, the goal of mitmproxy is to let an attacker monitor, capture and alter these connections in real-time.

1. HTTP Connection: For unencrypted HTTP connections, this is quite simple: mitmproxy accepts a connection from the HTTP client, say a mobile browser, displays the request to the attacker on the screen, and forwards the request to the destination web server as soon as the attacker confirms — maybe after adjusting the request payload a bit. mitmproxy simply acts as a middle man:
 - To the client, it looks like as if the mitmproxy server was simply relaying its connection (like your router or your ISP's servers do).
 - To the server, it looks like the mitmproxy server is the client.
2. HTTPS Connection: Unlike unencrypted HTTP traffic, here the transferred data is encrypted with a shared secret, a middle man (or a proxy) cannot decipher the exchanged data packets. When the client opens an SSL/TLS connection to the secure web server, it verifies the server's identity by checking two conditions:
 1. Trusted CA: it checks whether its certificate was signed by a CA known to the client
 2. Same CA: it makes sure that the common name (CN, also: host name) of the server matches the one it connects to.

If both conditions are true, the client assumes the connection is secure.

In order to be able to sniff into the connection, mitmproxy acts as a certificate authority (not a trustworthy one though). Instead of issuing certificates to actual persons or organizations, mitmproxy dynamically generates certificates to whatever hostname is needed for a connection.

- For example, if a client wants to connect to <https://www.facebook.com>, mitmproxy generates a certificate for "www.facebook.com" and signs it with its own CA.
- Provided that the client trusts this CA, both of the above-mentioned conditions are true (Trusted CA, same CN) — meaning that the client believes that the mitmproxy server is in fact "www.facebook.com". Figure 2 shows the request/response flow for this scenario.

For this attack to work, there are a few conditions that must be met:

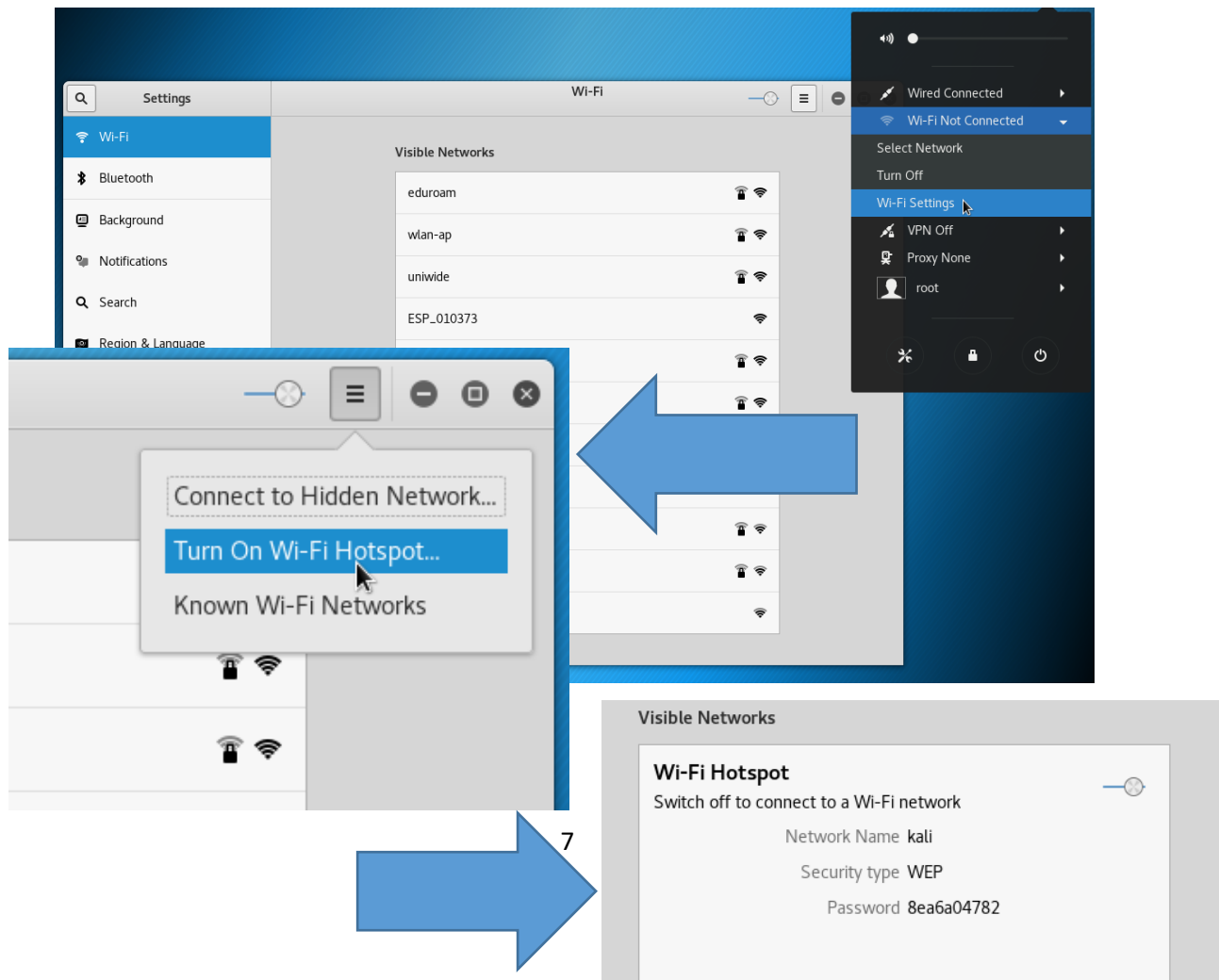
- **Mitmproxy as standard gateway (HTTP and HTTPS):** For both HTTP and HTTPS proxying, the server running mitmproxy must be able to intercept the IP packets — meaning that it must be somewhere along the way of the packet path. The easiest way to achieve this is to change the default gateway in the client device to the mitmproxy server address.
- **Trusted mitmproxy CA (HTTPS only):** For the HTTPS proxying to work, the client must know (and trust!) the mitmproxy CA, i.e. the CA key file must be added to the trust store of the client.

Tools we will be using for this lab:

1. Kali Linux on CSE Machines,
 - a. Mitmproxy (included in Kali),
 - b. ALFA wireless device,
2. Your own Andorid/iOS device.

Step 1: Quickest way to setup WiFi access point on Kali is through visual interface of Network Manager.

Plug-in your ALFA device and then “Turn on Wi-Fi Hotspot” (see below on how to do this).



Step 2: Transparent proxy-ing: When a transparent proxy is used, traffic is redirected into a proxy at the network layer, without any client configuration being required. This makes transparent proxy-ing ideal for those situations where you can't change client behaviour - proxy-oblivious mobile applications being a common example.

To set up transparent proxy-ing, we need two new components. The first is a redirection mechanism that transparently reroutes a TCP connection destined for a server on the Internet to a listening proxy server. This usually takes the form of a firewall on the same host as the proxy server - iptables on Linux. When the proxy receives a redirected connection, it sees a vanilla HTTP request, without a host specification. This is where the second new component comes in - a host module that allows us to query the redirector for the original destination of the TCP connection.

```
root@kali:~# sysctl -w net.ipv4.ip_forward=1

root@kali:~# iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 80 -j REDIRECT --to-port 8080

root@kali:~# iptables -t nat -A PREROUTING -i wlan0 -p tcp --dport 443 -j REDIRECT -to-port 8080
```

Step 3: Run mitmproxy. It has been added as a standard tool to Kali recently so no need for installation.

```
root@kali:~# mitmproxy -T --host
```

(For newer versions use *mitmproxy --mode transparent --showhost*) Keep the terminal window open and do not close it.

Step 4: Connect to the hotspot you have created using your Andorid/iOS device. Browse the following page using a browser (Google Chrome is suggested):

```
http://aavtrain.com
```

Try to login using the you group name as username and "kali" as password. (Don't be upset if you are not able to login :-))

Step 5: Return to Kali and revisit mitmproxy terminal window. (You'll see a list of http messages). Type "?" in the window and find the command to "set interception pattern". Use the relevant command and set "aavtrain" as the "intercept filter". Press 'q' to return back. Browse through the POST and GET commands. Press ENTER to view in detail and press 'q' to return back to the list. Try to find the POST command containing the username and password you have provided.

**** Important: Assessment point ****

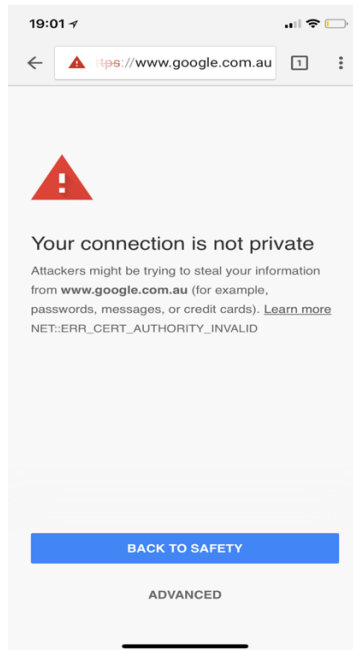
Raise your hand for the attention of you tutor who will come and verify that you have correctly located the username and password exchanged using HTTP.

**** Important: Assessment point ****

Step 6: Try to access the following page using a browser on your Andorid/iOS device.

`https://www.google.com`

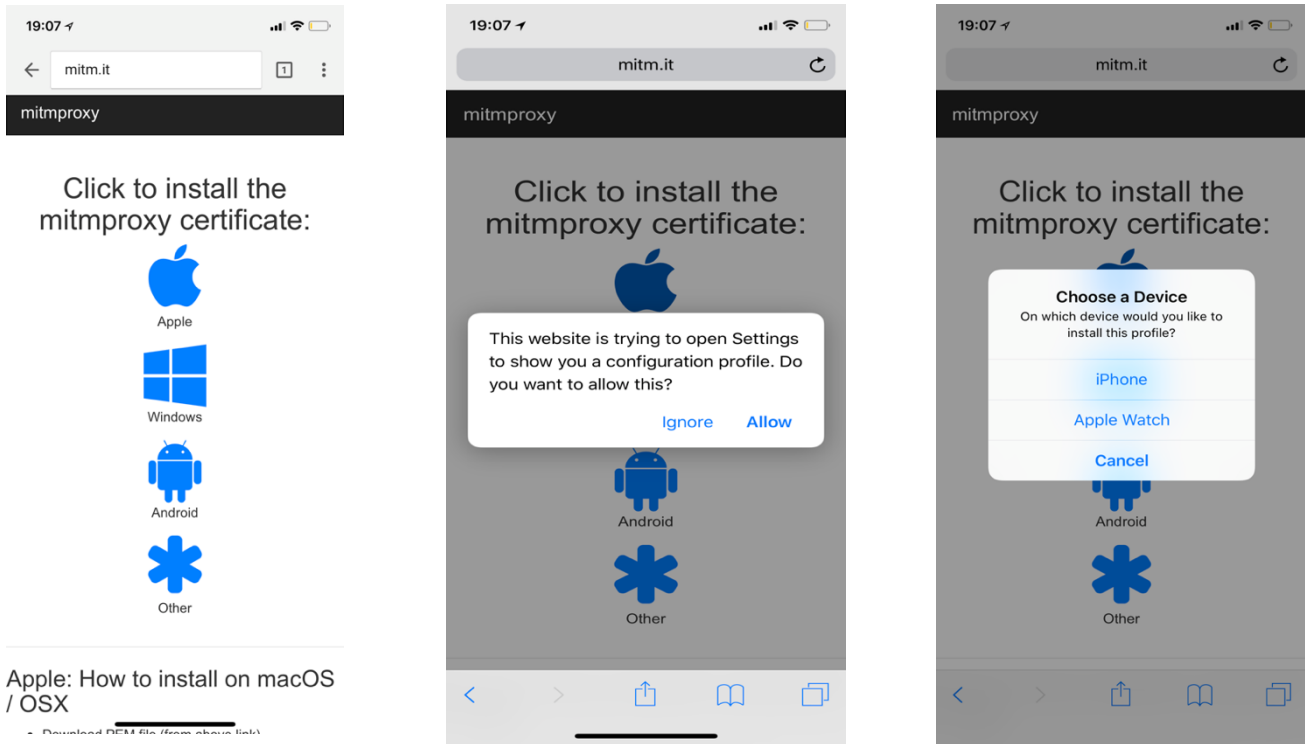
You should see an error like this or something similar to this.



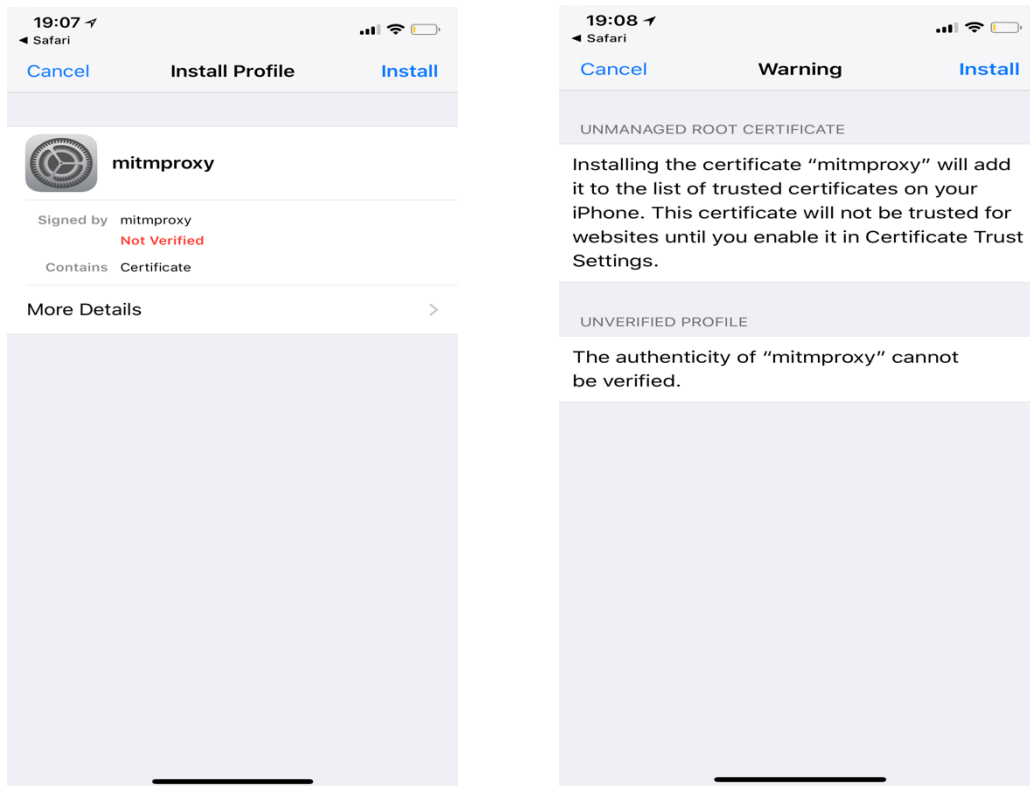
To fix this issue, do the following. On your browser (if iOS use Safari) and browse the following link:

`mitm.it`

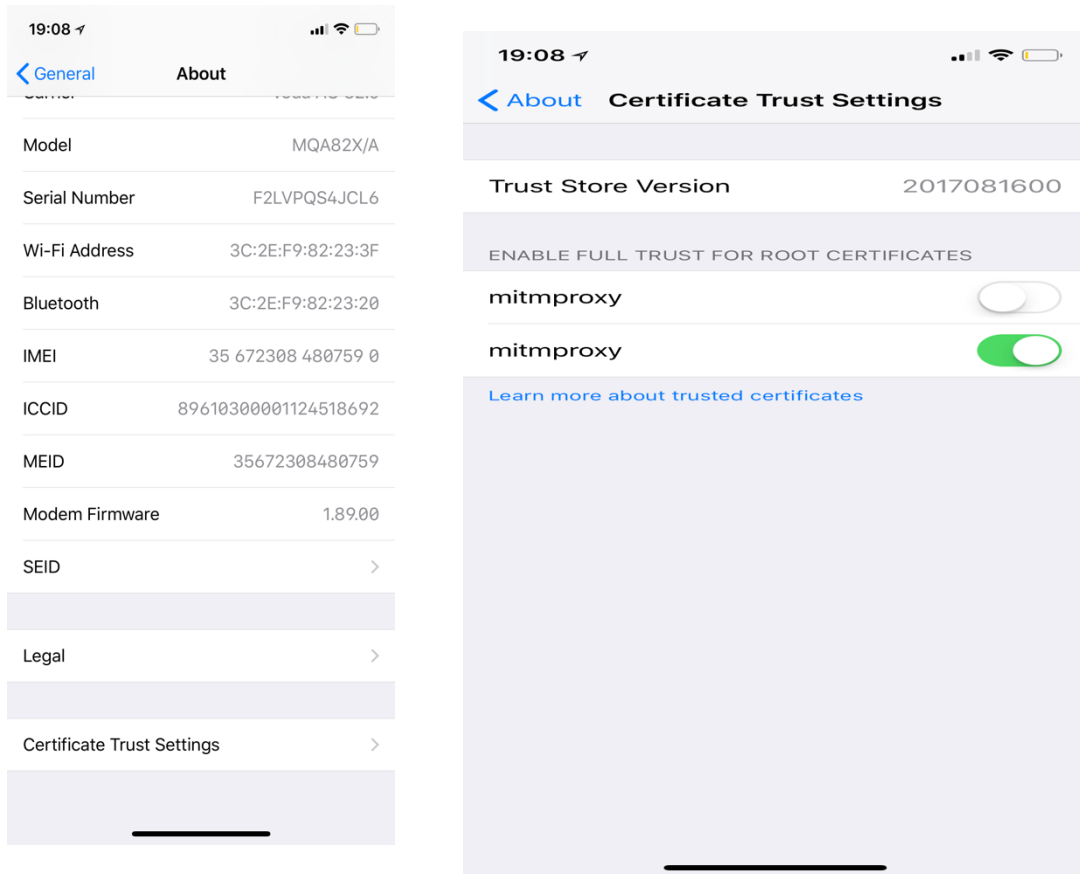
Download the mitm certificate matching your device operating system. Here, we assume you are using an iOS device. The process is very similar for Android.



If you have an Apple Watch then the third image may show as well. Please choose iPhone and proceed. Click on “Install” twice.



You need to trust the certificate for websites you will be browsing. Go to “Settings”>“General”>”About”>”Certificate Trust Settings”. Find “mitmproxy” and change its status to green to enable it.



Step 7: Browse to Facebook.com and enter fake username and password to login.
WARNING: DO NOT ENTER YOUR REAL USERNAME and PASSWORD for Facebook.

Step 8: Return to Kali and revisit mitmproxy terminal window. Type “?” in the window and find the command to “set interception pattern”. Use the relevant command and set “facebook” as the “intercept filter”. Browse through the POST and GET commands and find the POST command containing the username and password you have provided.

**** Important: Assessment point ****

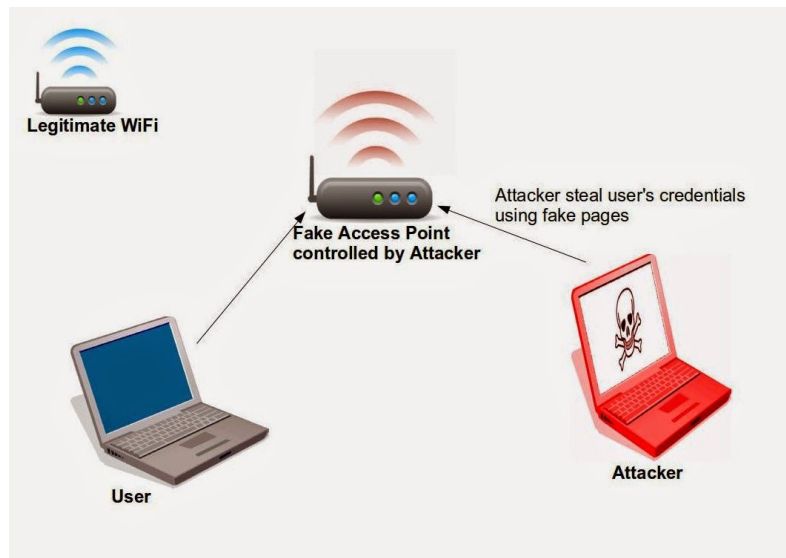
Raise your hand for the attention of you tutor who will come and verify that you have correctly located the username and password exchanged using HTTPS.

**** Important: Assessment point ****

You just extracted username and passwords exchanged over an HTTPS connection ;-)

Now, please go to “Lab Assignment 1” on Moodle and answer the questions. Remember for a full mark you must do this during the lab time (see instructions at the beginning of this document).

Part 2– Evil Twin Attack [Lab 2 Assignment 2]



In Part A, we used a hotspot as our AP. As you may have noticed this is easily noticeable as not being an actual AP and is hard to convince our target to connect to it. So, in Part 2 your task is to find a way to setup your very own AP. This AP must route the user traffic to Internet so the target would not notice connection to fake AP.

AP name: Starbucks

Channel: 2

Encryption: None

You will also need to disconnect clients connected to the real “Starbucks” AP so on reconnect they associate to your Evil Twin AP.

You will be required to provide all commands used to setup this working Evil Twin AP as part of one of the “Lab 2 Assignment 2” questions on Moodle.

Acknowledgements & Version History:

This lab was originally developed by Arash Shaghaghi, PhD Candidate at CySPri Lab of UNSW Sydney in 2015. It has undergone major refinement in 2018 by Arash Shaghaghi. It has benefited from revision and improvements by other CySPri Lab students including Chitra Javali, Girish Revadigar and Mohsen Rezvani.

Content in Part A has been extracted from the following sources:

- CWSP – Certified Wireless Security Professional Official Study Guide
- <https://blog.heckel.xyz/2013/07/01/how-to-use-mitmproxy-to-read-and-modify-https-traffic-of-your-phone/>
- <https://www.trustwave.com/Resources/SpiderLabs-Blog/Intercepting-SSL-And-HTTPS-Traffic-With-mitmproxy-and-SSLsplit/>
- <https://docs.mitmproxy.org/stable/howto-transparent/>

Document Version is 1.0.