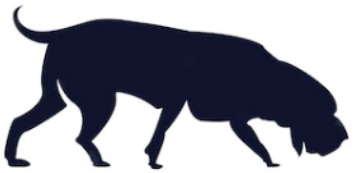

Operation Bloodhound - Cracking WiFi Passwords

SP2 - Student Project 2

Hegnes, Bjørn Martin

Abstract

This report goes through the vulnerabilities of WIFI Access Points in terms of acquisition and cracking passwords. With the ever-increasing use of WIFI devices in the daily life of the average citizen, many will own at least one device, if not several, that is connected to a WIFI Access Point.

The author wants to show that WIFI technology is vulnerable to different attacks that make it possible to crack the password and gain unauthorized access to the network without the individual's notice.

This report will show that it is possible to not only get the password of one WIFI Access Point but the passwords of a large city as well through “mass cracking”, i.e Oslo.

This report is the culmination of 5800 hashes as well as all of the 11 000 000 logged data points from WiFi Access Points, which in turn

- can show the top used passwords used on people's WIFI networks in Oslo
- map WiFi networks in the covered area in Oslo shown with their password pinpointed to their location.

This was done in 50 days, with a shoestring budget (4000 NOK) and the basic technical knowledge about exploiting vulnerabilities in WiFi Access Points.

The purpose of this report was to show the vulnerability of how easy it was to get the WIFI network password. The author did at no time login or performed any unauthorized access to the WIFI Access Points in the collected dataset. All logged data and an identifier of any personal information have been deleted by the author after the project's end. No personally identifiable information is presented in this report.

Content

Preface

This is my final Student Project for Network and IT-Security at Noroff – School of technology and digital media.

Firstly, I would like to thank my two partners for supporting me through my studies, especially the last semester, which did burn me out. And a big thanks to my brother for all of the brainstorming sessions during the last two years.

Secondly, I would like to give a big thank you to my teacher Mehdi Shadidi from the first year for all of the conversations and for thinking outside of the box, and a big thank you to my second-year teacher Justine Moodley for her great explanation which made the difficult syllabus more understandable.

How did the project come to be? It started a little before and during the two years of the study. Where I did learn to make a Pwnagotchi, by myself. This can be can compare with a Tamagotchi, as it is a Raspberry Pi zero running an A2C- based “AI” to learn to be better and better at capturing crackable WPA keys (Pwnagotchi.ai, 2022). And after reading that Ido Hoorvitch was able to crack 70% of the captured WIFI network hashes. Wanted to combine last year's project with this, crack the networks and show the locations of the networks with their password.

Figure 1. A Pwnagotchi in the wild. (Author, 2022).

1 - Introduction

In this report, the author will show how it was possible to crack the password of WPA and WPA2 WiFi Access Points at scale.

In the second part, the author will explain how a client connects to a WiFi Access Point, with a detailed visual explanation of how the authentication such as the 4-way handshake works and what the PMKID and PMK are.

In the third part of the report, the author will explain what attacks and vulnerabilities the project will use to get the password from a WiFi Access point.

In the fourth part, the authors will describe the equipment used to capture PMKID, briefly describe the different hardware chosen and explain the reasons for its use. In this part of the task, the authors will also outline the detailed configuration of the WIFI password capture device setup and images of the device.

In the fifth part, the author will go through Operation Bloodhound, with the structured way of collecting data from Oslo's WIFI Access Point.

In the sixth part, the author will use different software to convert the collected data and import the selected data into a database.

In the seventh part, the author will show the different attacks types used to crack the hashes collected, then the result from the cracking of the WIFI passwords, with the use of different parameters to crack most of them and displaying them in a table, such as how long the cracking was for each of the wordlists with different rules and how many passwords were cracked.

In the eighth part, the data points of the WiFi Access Points will be presented in a way that is shown as a colored dot on a map with its name and password over the location where it has been tracked. In addition, there will be WiFi Access Point graphs of the top-used WiFi passwords. More detailed information on how to interpret the visualization will be given where necessary.

In the ninth part, the authors explain what the collected datasets can be used for, and give examples of what others have already done in real-world scenarios, what new approaches are possible, and examples of doing similar things with other techniques. Additionally, the authors will present the inherent threats and dangers posed by creating large datasets for the cracked WiFi passwords, using scenarios and examples from real events.

The final section is a conclusion of the entire report, as well as the author's own reflections and reflections on the student project.

2 - Client and the Wireless AP

2.1 - PMK and PTK

When a client is going to connect to a WIFI Access Point, the WIFI Access Point must check if the client is allowed to join the network.

It will use the 4-way handshake network authentication protocol that is used in the IEEE-802.11i standard, this is used with the supplicants (clients) when establishing authentication with the wireless local area network (WIFI Access Point) to connect to the network. (Techopedia, 2013).

Every client that logs on to a wireless network using WPA2/PSK method and already knows the password or more technical word the Pre-Shared key (PSK). The PSK is used when generating the Pairwise Master Key (PMK), this gives the client access to the network as the PMK is the PTK and derives from the WIFI password (IEEE, 2001).

The PMK is 256 bits in size that contain the hash and SSID of the WIFI Access Point. The hash itself looks like this:

```
int pbkdf2_sha1 (constchar * passphrase,
                 constchar * ssid,
                 size_t      ssid_len,
                 int         iterations,
                 u8 *        buf,
                 size_t      buflen
)
```

In WPA2 PSK the Pre-Shared Key (PSK) is the same as the Pairwise Master key (PMK). The PSK is not used to encrypt data in each packet. The encryption key used to encrypt all of the data in transit between a client and an Access Point (Unicast) is called a Pairwise Transit Key (PTK) (wifi-professionals, 2019).

Below is the code box showing the PTK example:

```
PSK/PMK + Anonce + Snounce + MAC (authenticator) and MAC (supplicant)
```

2.2 - 4-way handshake

The authenticator is the WIFI Access Point, the supplicant is the client, anonce (Authenticator nonce) is a 1-time value for each packet generated by the access point and snonce (Supplicant nonce) is a 1-time value generated for each packet by the supplicant

The 4-way handshake starts when the access point sends “Message 1” see the Figure 2 with a one-time use-value anonce. The client creates its own PTK since it now has all the inputs (both MACs, PMK, snonce created by itself, and anonce). The WIFI Access Point sends out an RSN IE (PMKID).

Then the supplicant sends the “Message 2” back to the WIFI Access Point with its snonce so the WIFI Access Point can generate the same PTK. The message is sent with a MIC field set to 1 as a check to verify if this message is corrupted or not or if the key has been changed by a man in the middle. The supplicant also sends out an RSN IE (PMKID).

From there once the PTKs are verified, the WIFI Access Point derives GTK from GMK (For broadcast and multicast communication). Then the “Message 3” is sent with GTK which is encrypted with PTK and the message is also delivered with MIC. The message also instructs to install the temporal keys and an RSN IE packet is also sent in the frame.

And lastly “Message 4” sends a confirmation that the keys have been installed at the client/supplicant to the WIFI Access Point.

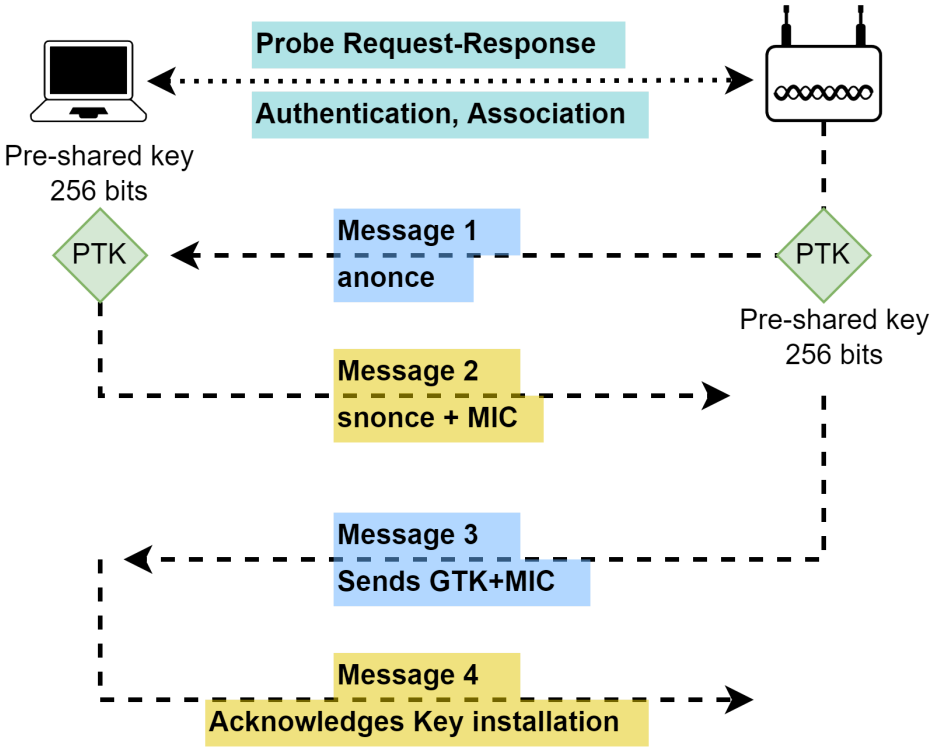

Figure 2. 4-way handshake simplified. (author 2022)

2.3 - WIFI Roaming and What is PMK Caching?

What is WiFi roaming? It can be compared to cell phones we use during our daily life, when we are at home going to the store, gym, or work, while moving around we seamlessly connect between different cell towers while moving without dropping the signal and need to reconnect to each new cell tower (Parsi, 2022).

So when a client happens to move from one part of an office to another part during a video call with the use of WiFi roaming enabled it makes the transition to move to the second access point seamlessly without the need to discontent the first Access Point and reconnect to the second Access Point in the traditional way such as a 4-way handshake.

PMKID is a unique key identifier that the WIFI Access Point uses to keep track of the PMK that is being used for the clients. “PMKID is a derivative of AP MAC, Client MAC, PMK, and PMK Name” (an, 2020)

As shown in the code box below, it shows the PMKID does use a hash-based message authentication code, and the HMAC does provide authentication using a shared secret:

```
PMKID = HMAC-SHA1-128(PMK, "PMK Name" | MAC (authenticator) | MAC
(supplicant))
```

PMK caching makes the client experience a smooth transition between WIFI Access Points when the client uses the PMKID caching, the client does not have to go through the entire authentication cycle and this reduces the time it takes for the client to authenticate to a new AP. See Figure 3, below for a visual expansion.

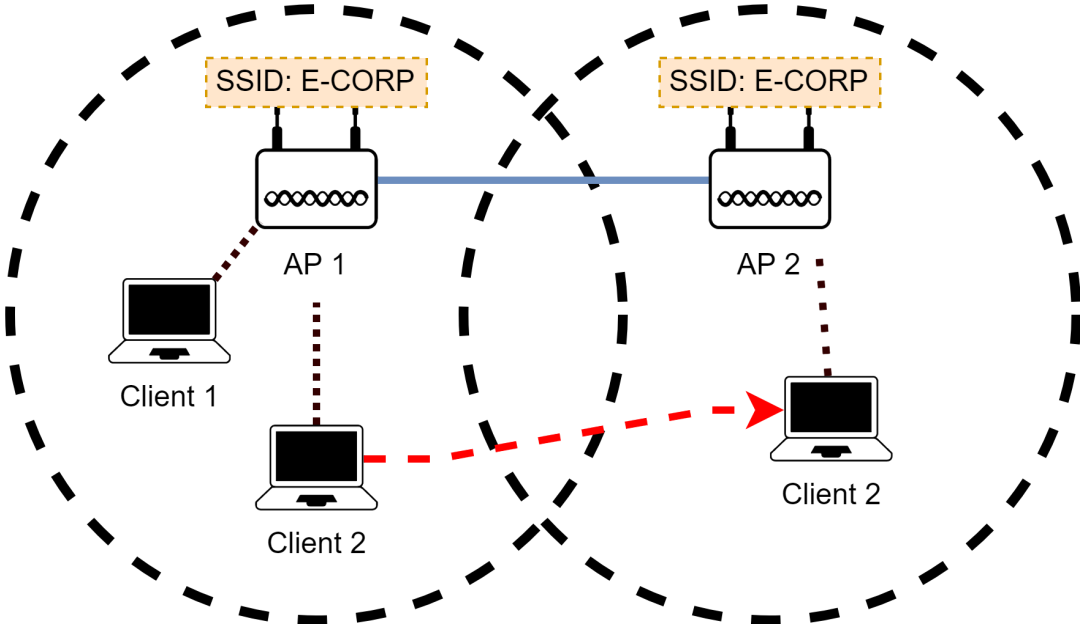

Figure 3. WiFi Roaming simplified. (Author, 2022).

3 - Attacks

In this section, the author will go through the deauthenticating attack which is a client attack, and the PMKID attack which is a client-less attack. Even though these attacks don't work on the new WP3 standard, the adaptation of WPA3 is still low since not all wifi devices support this standard. Making these attacks still relevant (Tay, 2020)

3.1 - Deauthenticating

This attack works by disconnecting a device from a WIFI Access Point, spoofing the client's MAC address then sending a deauthentication frame to the WIFI Access Point. From there the client must reauthenticate to the WIFI Access Point, by performing the 4-Way Handshake. These EAPOL frames of the handshake are then sniffed by the attacker, to later be used with hashcat to be cracked (Margaritelli, 2019).

The disadvantage of the Deauthenticating attack is that it needs clients connected to the WIFI Access Point to be possible, this means that if there are no clients the attack cant be executed.

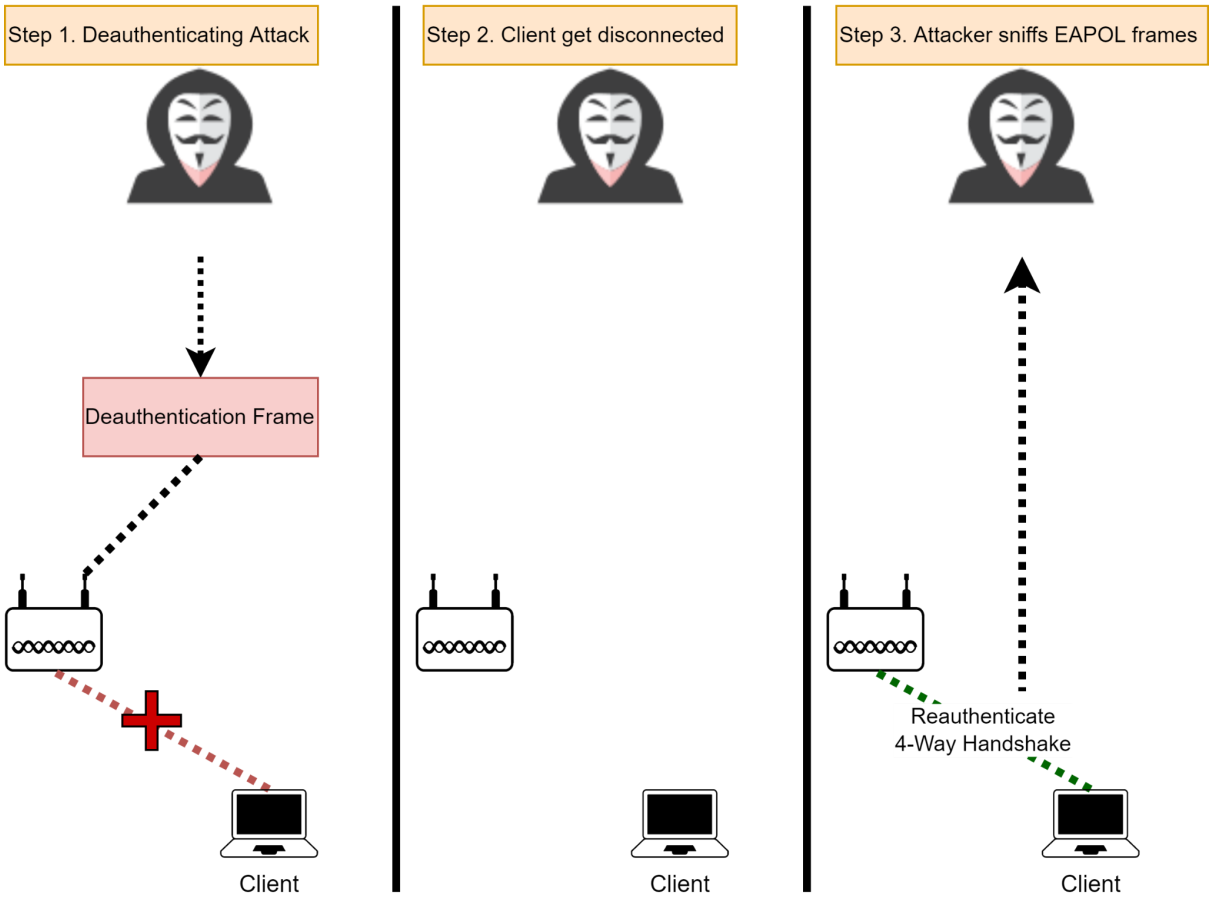

Figure 4. Deauthenticating attack simplified. (Author, 2022).

3.2 - Client-less PMKID Attack

For years, attacking a wpa2 Network meant kicking someone off the network, waiting for them to reconnect, then collecting the WPA handshake in order to attempt to crack it. A new attack based on the PMKID allows attacking networks even when no one's connected.

An attack technique to crack WPA PSK (Pre-Shared Key) in other words WiFi passwords, was discovered by researcher Jens “atom” Steube, this is performed on the RSN IE (Robust Security Network Information Element) and only needs a single EAPOL frame

This vulnerability can be used against most of the 802.11i/p/q/r networks that have roaming function enabled. The main advantages of this attack are as follows, there is no need for users since this is a client-less attack that communicates directly with the Access Point, and no need to wait for the complete 4-way handshake between the user and AP, as well as sending out deauthentication frames. (atom, 2018)

The attacker sends an association frame to the access points, so the WIFI Access Point sends out an EAPOL frame that contains the RSN IE which includes the PMKID, see the first step in section 3.2 (Kohlios and Hayajneh, 2018).

The reason these attacks work is because of the PMKID cache used during WIFI roaming, which many WIFI Access Points have enabled by default. See section 3.3 for an explanation.

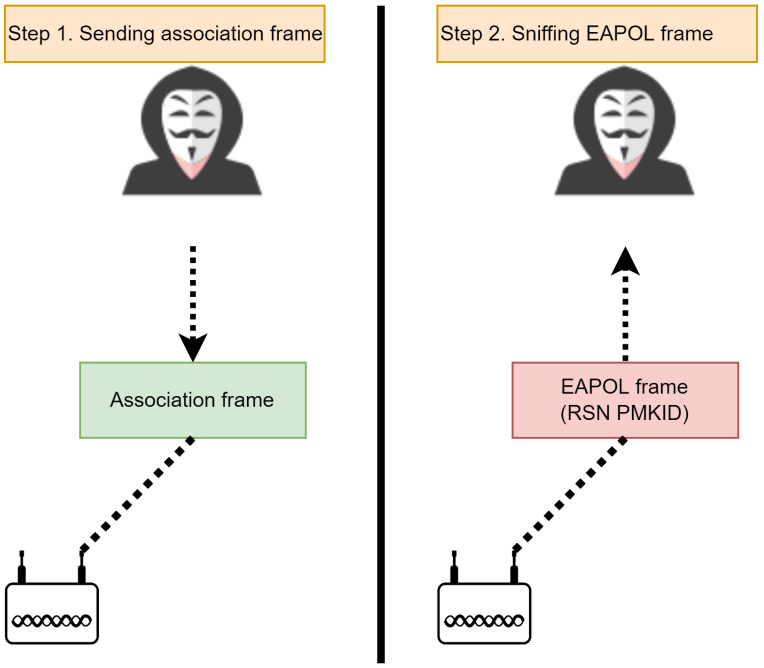

Figure 5. Client-less Attack simplified. (Author, 2022).

4 - How to capture PMKID

In this section, the author will go through the setup of the hardware and software that was used to capture PMKID WiFi Access Points that will be used to crack the passwords of the WiFi networks, in a step-by-step manner.

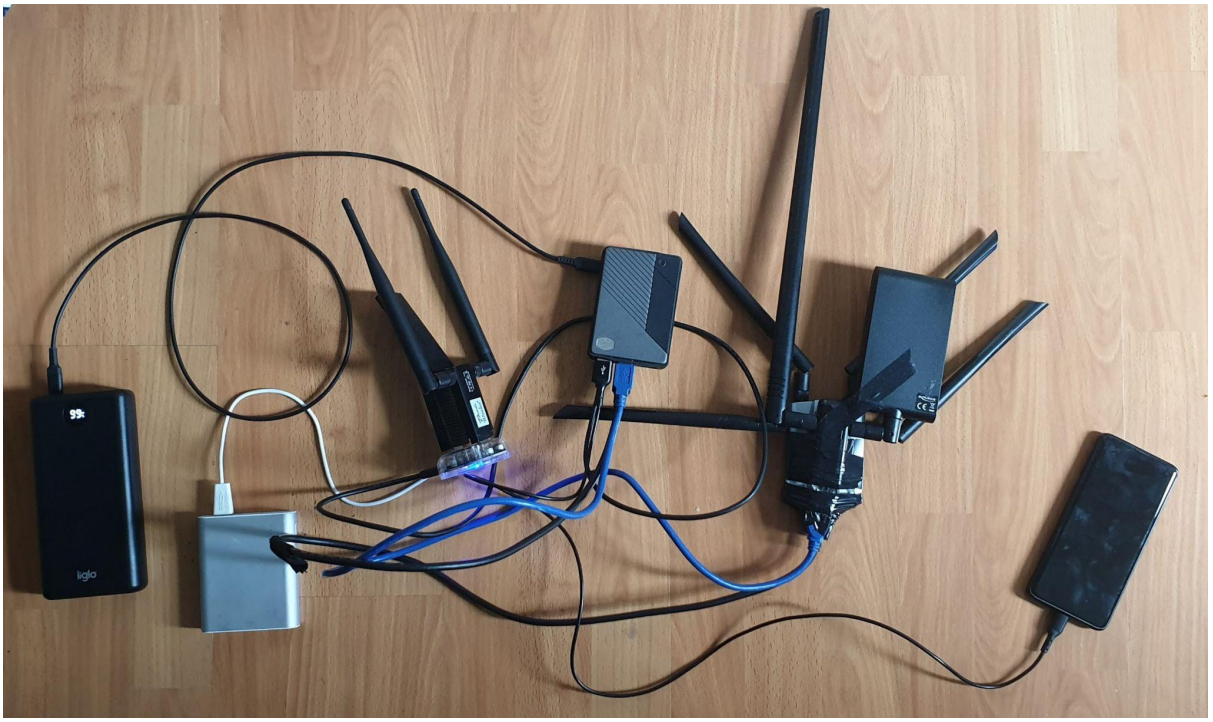

Figure 6. Hardware. (Author, 2022).

4.1 - Hardware

The Hardware chosen for this device in Figure 6 is the following:

Raspberry Pi 4 Model B 8GB is recommended as a good choice for a portable device because it is able to handle running Kismet in a moderate to a busy environment (Kismet, 2021)

Cooler Master Raspberry Pi Case 40, this was needed since

DS3231 Mini Hardware clock since the Raspberry Pi 4 does not come with an internal hardware clock and are relying on the Network Time Protocol, the author did choose to add a hardware clock to keep the time in case of the possibility of not having an internet connection (The Pi Hut, 2015).

SanDisk Micro SD-card 64 GB, was chosen so there will be no need to worry about storage.

Alfa Network AWUS1900, this network adapter was used since the author already had it before. But in retrospect, this specific adapter is not recommended due to unreliability.

Panda Wireless PAU09 N600, was chosen since I had it before and it supports monitor mode which is needed for hcxdumptool.

ALFA Network AWUS036ACM, was bought since this has the reliable driver mt76x2u and support monitor mode.

Netgear A6210 was bought since it has the reliable driver mt76x2u and support monitor mode.

Generic 4-port USB Hub, this is needed since all of the devices cant be powered by the raspberry pi 4 alone. This USB hub has a standalone USB-micro power input so it was connected to the Clas Ohlson Power Station.

An omnidirectional WiFi antenna 9 dBi and a Directional WiFi antenna 10 dBi, was attached to the ALFA Network AWUS036ACM adapter.

iiglo Powerbank 30000 mAh. This power bank was chosen because it will last 48 hours which leaves a 1-day power charge in reserve if one forgets to recharge daily. And since the Raspberry Pi 4 needs a 15-watt 3A power supply, this will provide it through the USB-C connector on the power bank.

Clas Ohlson Power Station 12000 mAh, this was the second power source to power all of the adapters.

Samsung Note 9 was used to share the GSP position with the Raspberry Pi 4 and as a backup solution for WiFi network capture.

pwnagotchi was used as a backup solution to collect PKMID.

4.2 - Software

Kismet

Kismet is a well-known wardriving tool. This tool is used by IT professionals, hackers, and researchers for wireless networks and device detection, and wireless sniffing, and can be used as a wireless intrusion detection system (WIDS) framework. (Kismet, 2022)

In this project, the software uses the case of collecting and logging WiFi networks with their location. This was done in the project “Location Tracking of WIFI Access Point and Bluetooth Devices” where the author did use Kismet to log all of the WiFi networks that were in the proximity when passing when bicycling through an area. (Hegnes, 2021)

Link to project: kismetwireless.net

GPSd Client

In this project, the author will use the GPS data from a smartphone and share it with the Raspberry Pi, the app used is GPSd Client on the play store. It works by forwarding the NMEA data from the smartphone GPS to a specific host over UDP (tiago shibata, 2021).

Link to project: play.google.com/store/apps/details?id=io.github.tiagoshibata.gpsdclient

gpsd

gpsd is a service demon that monitors one or more GPSes that can be connected through a USB, serial, or even set it up as a host that gets the GPS data from a client that sends the GPS data over UDP (GPSd, 2022).

Link to project: gpsd.gitlab.io/gpsd

hcxdumptool

Will be used on three of the adapters to request the PMKIDs from the Access Points and to dump the received frames to a pcapng file.

Link to project: github.com/ZerBea/hcxdumptool

hcxtools

hcxpcaptool will be used to convert the captured data from pcapng format to a hash that is compatible with hashcat.

Link to project: github.com/ZerBea/hcxtools

hashcat

hashcat is well-known as one of the world's fastest and most advanced password recovery tools, this will be used to crack the PMK hashes (hashcat, 2021).

Link to project: github.com/hashcat/hashcat

MySQL

MySQL will be used on a VM to set up a database for all of the captured data..

Link to website: mysql.com

Tableau

Tableau will be used to see and understand the captured data. And this will be connected to the SQL database.

Link to website: tableau.com

Own Tools

wigle-kml-converter.sh, for parsing and converting Wigle KML to Wigle CSV format.

wigle-queries.sh, for queries to wigle.net API with a list of mac addresses for WiFi information such as WiFi name, location, and more to a wigle CSV format. This will be used to find WiFi networks that are missing in the project's dataset.

status.sh, for checking the status during the data collection, it displays CPU and GPU temperature, GPS location, lists all of the active screens from screen, displays kismet status, and lists all of the recently modified files in the WiFi hashes this makes it no need to switch to the screens with the hcxdumptool to see the status.

screen.sh, for starting the gpsd host service, stopping conflicting wifi processes, starting one screen with the status.sh, then three instances of screens with hcxdumptool for the different WiFi adapters, then the last screen is used for kismet.

SP2cat, for cracking all of the password hashes on Google colab cloud GPUS. And this is a fork of someshkar's colabcat, with the extra functionality of a GUI, able to choose wordlist, rules, and session.

Link to project: github.com/teddy-flow

4.3 - Raspberry Pi Setup

In this section, the author will install and configure everything needed to make the raspberry pi capture the WiFi networks as well as the PMKID. The operating system (OS) chosen to be installed on the Raspberry Pi 4 is Raspberry Pi OS Lite Buster, a lightweight operating system with no desktop or GUI, recommended as it is stable and designed for the Raspberry Pi (The Raspberry Pi Foundation, 2021).

Figure 7. Raspberry Pi 4 with Cooler Master Case. (Author, 2022).

After the installation of the operating system, it was needed to upgrade the system, this was done with the command below.

```
sudo apt-get update
sudo apt-get upgrade
```

Then it was needed to set static IP for ethernet and USB ethernet. Since the GPSD host server needs to get its GPS data from the smartphone over USB tethering. As well as a static IP for the home network to dump and backup the data.

```
sudo nano /etc/dhcpcd.conf
```

This was added to the dhcpcd.conf file.

```
# Home network
interface eth0
static ip_address=192.168.0.4/24
static routers=192.168.0.1
static domain_name_servers=192.168.0.1
# USB phone tethering network
interface usb0
static ip_address=192.168.42.100
static routers=192.168.42.1
static domain_name_servers=192.168.42.1
```

Then to install screen this gives the ability to launch and use multiple shell sessions from a single ssh session which is needed since there will be five shell sessions needed, this also prevents the sessions from being terminated and has the ability to resume the session (GeeksforGeeks, 2019).

```
sudo apt-get install screen
```

Then the GPSD tool is installed to get the GPS data from the smartphone, over UDP.

```
sudo apt-get install gpsd gpsd-clients
```

Disable the systemd service the GPSD installed.

```
sudo systemctl stop gpsd.socket
sudo systemctl disable gpsd.socket
```

Configure gpsd

```
sudo nano /etc/default/gpsd
```

This is the configuration used.

```
START_DAEMON="true"

USBAUTO="false"

DEVICES="udp://192.168.42.100:11123"

# Other options you want to pass to gpsd
GPSD_OPTIONS="-n -b"
```

This command is used to start the service.

```
sudo systemctl start gpsd.socket
```

To check if the Raspberry Pi gets the GPS data from the smartphone, the author did enable USB tethering, then opened the GPSd client app with the IP address of the Raspberry Pi. Then executed cgps.

```
cgps
```

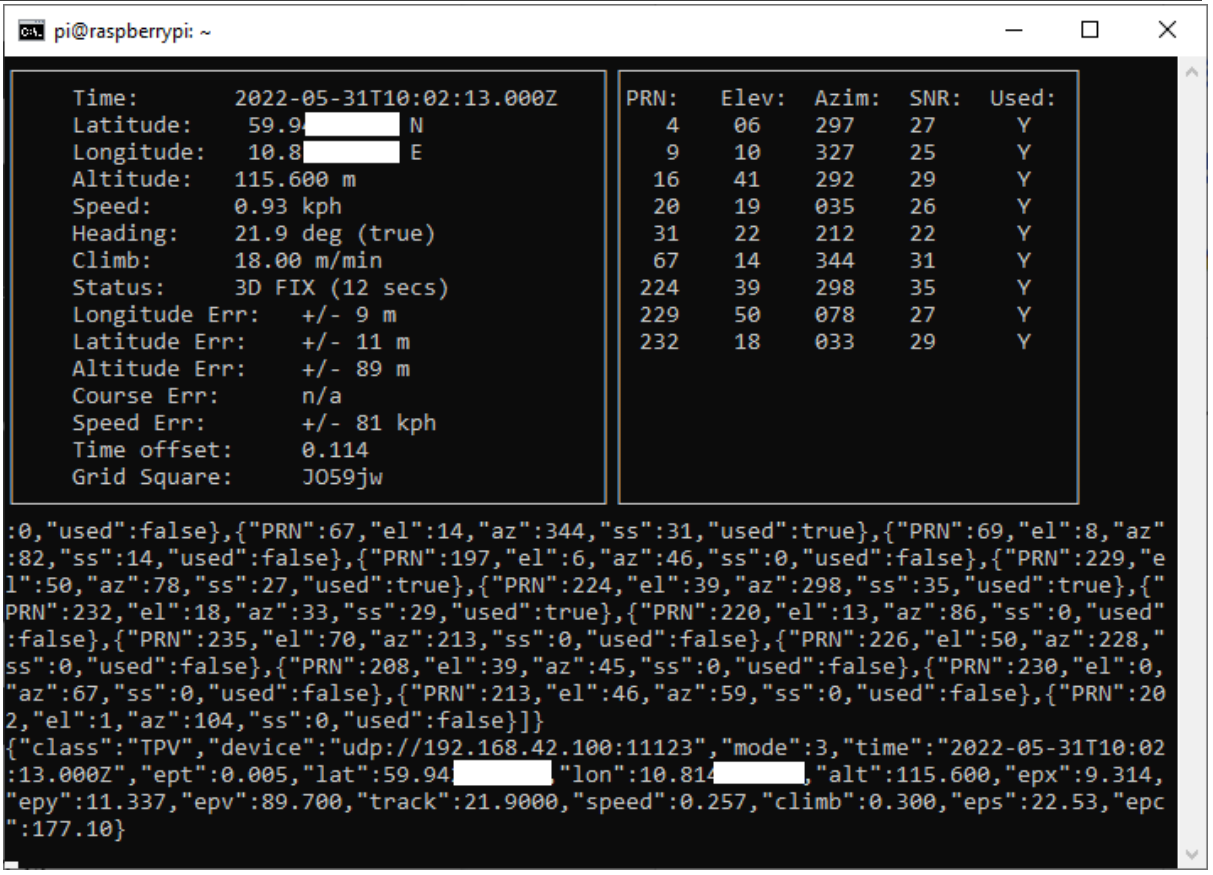

Figure 8. cgps reads the GPS data. (Author, 2022).

Then it is time to install the software to capture all of the WiFi networks, this program is called Kismet see section 4.2.

```
wget -O - https://www.kismetwireless.net/repos/kismet-release.gpg.key |
sudo apt-key add -
echo 'deb https://www.kismetwireless.net/repos/apt/release/bullseye
bullseye main' | sudo tee /etc/apt/sources.list.d/kismet.list
sudo apt update
sudo apt install kismet
```

Enable GPS in kismet by configure the kismet.conf

```
sudo nano /etc/kismet/kismet.conf
```

Delete # infront of the line below:

```
gps=gpsd:host=localhost,port=2947,reconnect=true
```

Then hcxdumptool is installed with the following commands

```
sudo apt-get install libcurl4-openssl-dev libssl-dev pkg-config
git clone https://github.com/ZerBea/hcxdumptool.git
cd hcxdumptool
make
sudo make install
```

4.4 - Database and Desktop

The MySQL server was installed on Ubuntu Server 22.04 Virtual Machine (VM).

With the following command.

```
sudo apt install mysql-server
```

After the installation a database called war was made with four tables, one for all of the captured WiFi networks with their location information, one for cracked passwords, one for all of the raw data of the hashes including those that were not cracked, and lastly one for vendors or owners of the OUI.

Collected WiFi networks table.

```
CREATE DATABASE IF NOT EXISTS war;

USE war;

CREATE TABLE `war_t` (
  `MAC` varchar(17) DEFAULT NULL,
  `SSID` varchar(101) DEFAULT NULL,
  `AuthMode` varchar(45) DEFAULT NULL,
  `FirstSeen` varchar(19) DEFAULT NULL,
  `Channel` int DEFAULT NULL,
  `RSSI` int DEFAULT NULL,
  `CurrentLatitude` decimal(12,10) DEFAULT NULL,
  `CurrentLongitude` decimal(12,10) DEFAULT NULL,
  `AltitudeMeters` decimal(9,6) DEFAULT NULL,
  `AccuracyMeters` int DEFAULT NULL,
  `Type` varchar(4) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

Password and hash table.

```
USE war;

CREATE TABLE `hashes` (
  `HASH` varchar(101) DEFAULT NULL,
  `MAC` varchar(17) DEFAULT NULL,
  `Client_Mac` varchar(17) DEFAULT NULL,
  `SSID` varchar(101) DEFAULT NULL,
  `Password` varchar(101) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

Raw hashes table.

```
USE war;

CREATE TABLE `raw` (
  `SIGNATURE` varchar(3) DEFAULT NULL,
  `TYPE` varchar(2) DEFAULT NULL,
  `PMKID` varchar(101) DEFAULT NULL,
  `MAC` varchar(17) DEFAULT NULL,
  `MACSTA` varchar(17) DEFAULT NULL,
  `ESSID` varchar(101) DEFAULT NULL,
  `some1` varchar(400) DEFAULT NULL,
  `some2` varchar(400) DEFAULT NULL,
  `some3` varchar(2) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

Vendor table.

```
USE war;

CREATE TABLE `vendor` (
  `MAC` varchar(17) DEFAULT NULL,
  `VENDOR` varchar(101) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

And lastly hcxtools was installed on the author's desktop with the following commands:

```
sudo apt-get install libcurl4-openssl-dev libssl-dev zlib1g-dev
git clone https://github.com/ZerBea/hcxtools.git
cd hcxtools
make
sudo make install
```

4.5 – Start Capture

In this section, the author will show how the data is captured with the Raspberry Pi. Before starting any of the bash scripts, the author had to plug in the smartphone to the Raspberry Pi via USB and enable USB tethering. Then navigate and open GPSd client using the proper IP and port to the Raspberry Pi and press start.

Figure 9. channel hopping. (Author, 2022)

Then the author did connect to the Raspberry Pi with SSH with the JuiceSSH that is available for the Android smartphone.

After login in, the author did run the screen.sh bash script, this will start the GPSD host that the smartphone shares its GPS data with. Then make a screen session for the status.sh bash script. And then three different screen sessions with hcxdumptool with the default command as shown in the example below.

```
sudo hcxdumptool -i wlan1 -o ./hashes/wlan1/1.pcapng --enable_status=3
```

Then start the last screen session for kismet with the following command.

```
sudo kismet -c hci0 -c wlan4 -p /home/pi/war-drive
```

And lastly connect to the first screen called status.

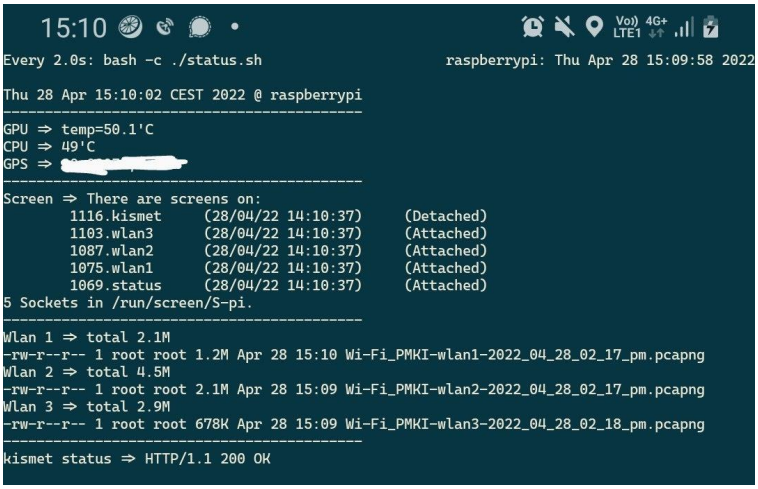

Figure 10. channel hopping. (Author, 2022)

5 - Operation Bloodhound 19.04.2022-13.05.2021

Operation Bloodhound was the nickname the author called the period of data gathering. As the Bloodhound breed has been honed to be masterful trackers and has a very good sense of smell. As they are used for sniffing and following air scents (Ned Hardy, 2020). And this gives an idea of how this data collection was done.

And in Operation Bloodhound, the author will just walk for the duration of the data collection. And the reasoning for walking rather than driving, bicycling, or any other transportation that is faster than walking is the following.

Since there is only one WiFi adapter used for the collection of the WiFi networks, it has to do something that's called channel hopping to scan all of the WiFi channels if a Wifi network is on that channel. And as shown in Figure 11 below. There are 28 channels that the adapter has to channel hop between.

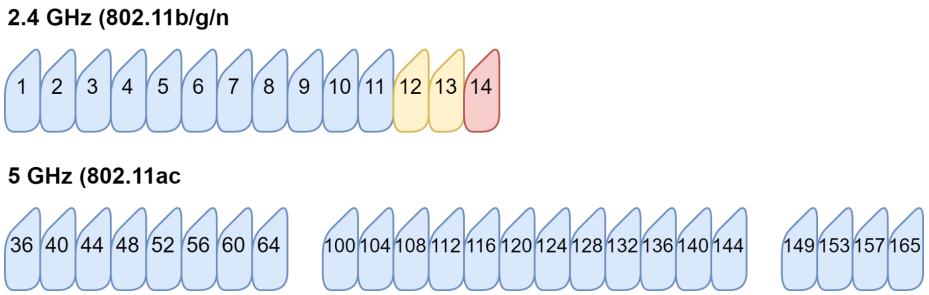

Figure 11. WiFi channels. (Author, 2022)

The figure below shows the channel hopping the WiFi adapter does on kismet.

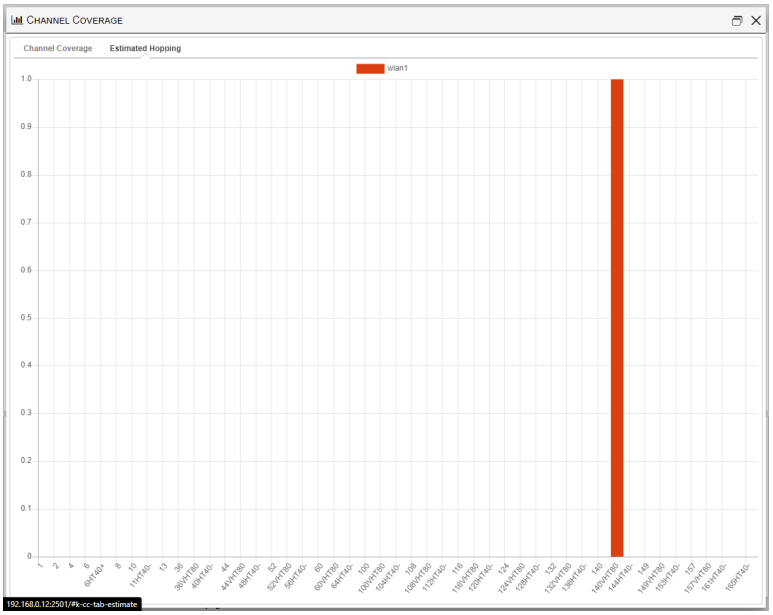

Figure 12. channel hopping. (Author, 2022)

And from the paper “Accuracy Performance Analysis between War Driving and War Walking in Metropolitan WiFi Localization”, the researchers found that when walking they collected around 6.5 million signal samples while driving they only collected around 233,089 signal samples (Tsui et al., 2010).

And with this in mind, the author has set the average speed to walk to around 4 km/h to 6 km/h during the data collection, in other words, a light walk, so kismet can scan and discover most of the WiFi networks in the range of the capture device.

In addition, the routes that were planned were chosen because those have the highest population density within ring 3 of Oslo, this data was obtained from Statistisk sentralbyrå or Statistics Norway in English. As you can see in Figure 13 the darker an area is the more populated it is.

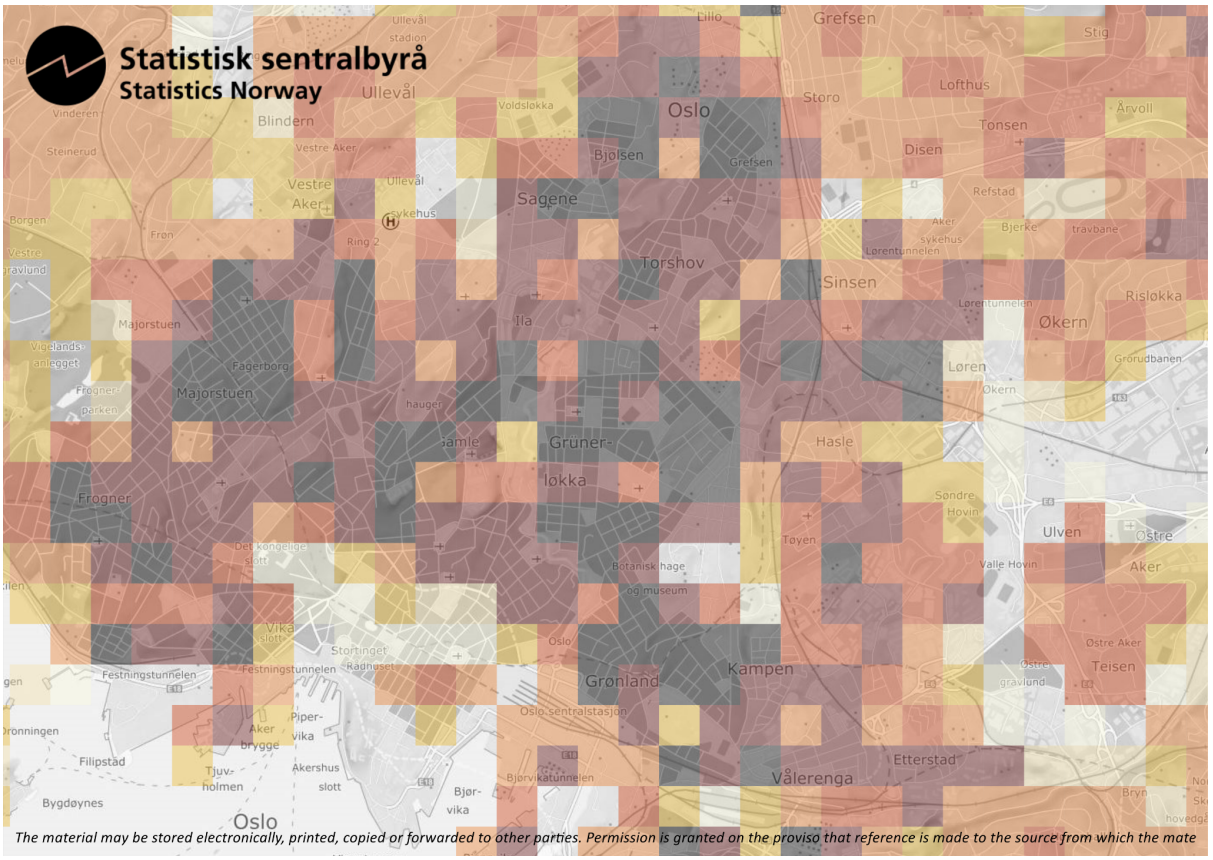

Figure 13. Oslos population density. (Author, 2022)

Then the data collection started with the author’s own two legs. Seventeen working days (19.04.2022-13.05.2021) were allocated to Operation Bloodhound, this period consisted of walking through different highly populated parts of Oslo.

At the end of the data collection phase, was the total distance of 200 km covered. The entire route is shown in Figure 15, this is the logged GPS data from the data collection.

And the author did use the same technique from “Location Tracking of WIFI Access Point and Bluetooth Devices”, by structuring the data collection by dividing the city up into parts. Then collecting the data by walking Oslo part by part and avoiding going through one are more than necessary (Hegnes, 2021). As shown in Figure 14 below.

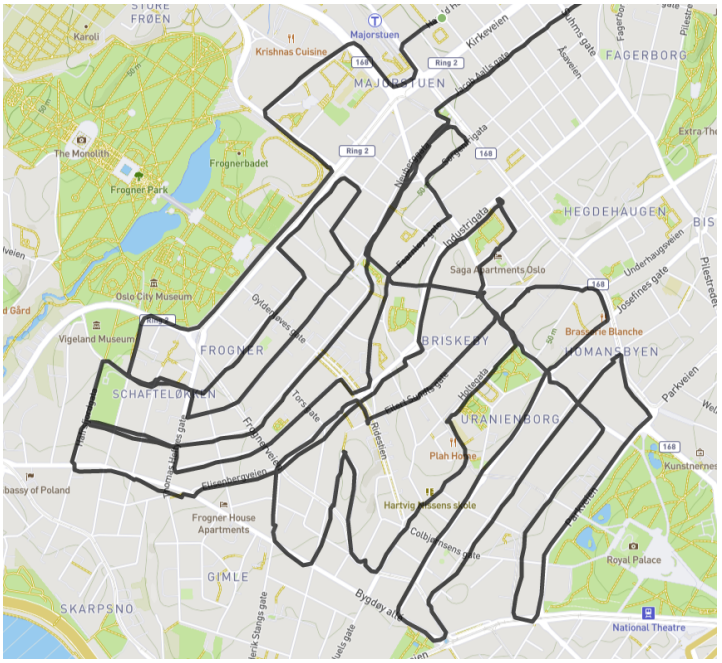

Figure 14. One part. (Author, 2021).

The entire route the author walked is shown in Figure 15, which is 17 days of raw data gpx files combined to visualize the city's itinerary.

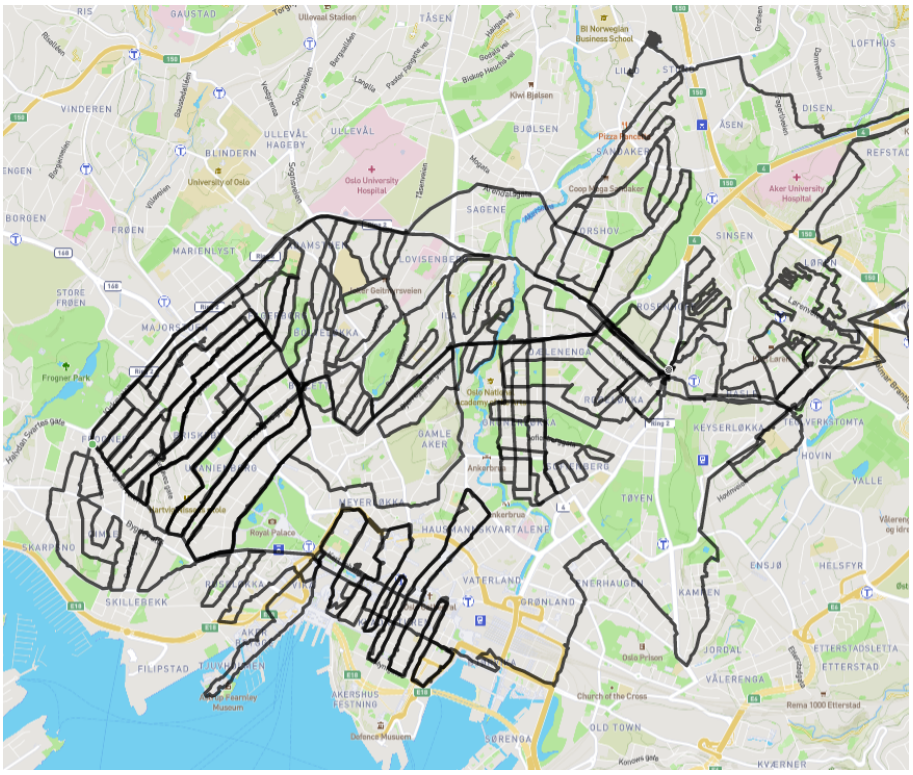

Figure 15.Operation Bloodhound In Oslo. (Author, 2022).

One of the first observations was that walking did collect more data in the kismet files as shown in the table below. The first one in the table with the cycling was from the author's student project “Location Tracking of WIFI Access Point and Bluetooth Devices” from 2021. As seen it shows that walking actually collects three times more data. walking collected more data.

Collection type	Distance traveled in km	Total data in MB	Data per km in MB
Cycling	300	6070	20
Walking	200	11941	60

Table 1. Collection Type.

And 518 MB of .pcapng files was captured by the three adapters running hcxdumptool during the time of Operation Bloodhound.

6 - Convert and data import

In this section, the author will show how the captured data from Operation Bloodhound was converted and imported to the database. As seen in Figure 16, show the process from the raw data to the import of the selected data that is intended for this project.

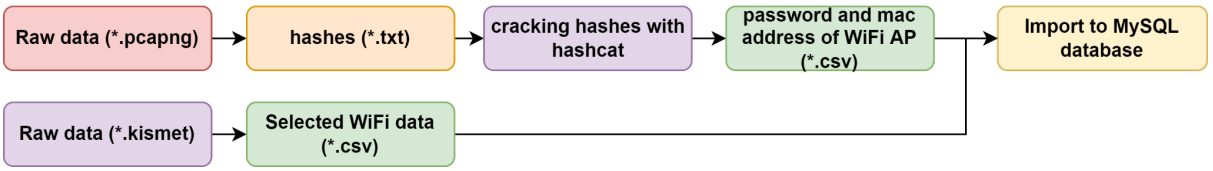

Figure 16. Convert Prosses. (Author, 2022).

kismetdb_to_wiglecsv

One of the export tools for Kismet is called kismetdb_to_wiglecsv, this one is chosen to be used since it makes a CSV file that is easily human-readable and can be imported into a SQL database. The Command used to convert the kismet file is the following.

```
kismetdb_to_wiglecsv --in captured-data.kismet --out captured-data.csv
```

hcxpcaptool

Then convert all of the pcapng files into a hash file that is compatible with hashcat. The command below was used to merge all of the pcapng files into one hash file.

```
hcxpcapngtool -o all-hashes.txt *.pcapng
```

The result was 5881 password hashes that were captured during the data collection period.

Hashcat

The author did use Google colab for hashcat for the cracking of the hashes see Figure 17.

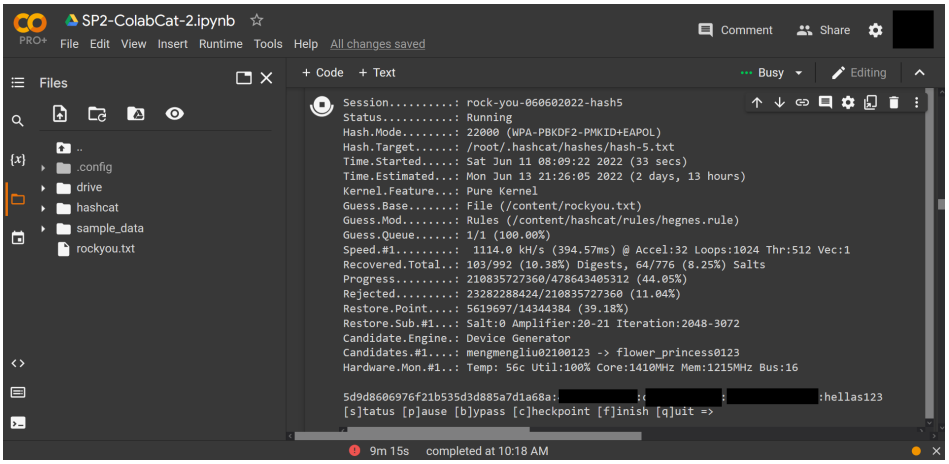

Figure 17. SP2Cat. (Author, 2022).

The command below is an example of the command used to crack the hashes.

```
hashcat -m 22000 all-hashes.txt rockyou.txt -w 4
```

MySQL

After converting the kismet file it was imported into the SQL database table war_t by this command.

```
LOAD DATA LOCAL INFILE 'captured-data.csv' INTO TABLE war_t FIELDS
TERMINATED BY ',' LINES TERMINATED BY '\n' IGNORE 2 LINES;
```

The selected data that was parsed from the raw kismet data file was for all of the captured devices and will be added row by row in a CSV file with the following structure MAC, SSID, AuthMode, FirstSeen, Channel, RSSI, CurrentLatitude, CurrentLongitude, AltitudeMeters, Accuracy Meters, Type. This imported data have all of the mentioned above as columns and all of the devices as rows as shown in Figure 18 below for a visual explanation.

MAC ...	SSID ...	AuthMode ...	FirstSeen ...	Channel...	RSSI int	CurrentLatitude ...	CurrentLongitude...	AltitudeMeters ...	AccuracyMeters...	Type ...
	Uncategoriz...	2021-04-05 1...	7936	-94	60	10.7208637778	45.4301897651	4.000000	0	(NULL)
	Uncategoriz...	2021-04-05 1...	7936	-88	60	10.7209512880	43.6290841807	6.000000	0	(NULL)
5f:e1:97:b9:a...	(EMPTY)	Misc	2022-05-13 ...	0	-92	59.9429382544	10.8140534445	220.591821	12	BLE
74:1b:b2:d9:...	Airport hje...	[WPA2-PSK-...	2022-05-13 ...	36	-85	59.9429382544	10.8140534445	220.591821	12	WIFI
88:ac:c0:91:...	Telia-2G-Sk...	[WPA2-PSK-...	2022-05-13 ...	6	-90	59.9429382544	10.8140534445	220.591821	12	WIFI
76:1b:b2:d9:...	Airport gjest	[WPA2-PSK-...	2022-05-13 ...	6	-94	59.9429382544	10.8140534445	220.591821	12	WIFI
28:6d:97:4e:...	[washer] Sa...	[WPA2-PSK-...	2022-05-13 ...	1	-87	59.9429382544	10.8140534445	220.591821	12	WIFI
88:ac:c0:91:...	Telia-1A0338	[WPA2-PSK-...	2022-05-13 ...	64	-77	59.9429382544	10.8140534445	220.591821	12	WIFI
88:ac:c0:91:c...	Telia-2G-C9...	[WPA2-PSK-...	2022-05-13 ...	6	-84	59.9429382544	10.8140534445	220.591821	12	WIFI
48:f2:54:b4:8...	(EMPTY)	Misc	2022-05-13 ...	0	-96	59.9429382544	10.8140534445	220.591821	12	BLE
24202_2805...	TeliaSonera...	LTE;no	2022-05-13 ...	0	-109	59.9429382544	10.8140534445	220.591821	12	GSM

Figure 18. war_t table. (Author, 2022).

After the password hashes were cracked were they formated to fit the table structure as seen in section 4.4. The “all-hashes.txt” was then imported into the hashes table of the database, with the command below.

```
LOAD DATA LOCAL INFILE 'all-hashes.txt' INTO TABLE hashes FIELDS
TERMINATED BY ',' LINES TERMINATED BY '\n';
```

The table structure is the following HASH, MAC, Client_Mac, SSID, Password. The reason for this is that the MAC collum from both tables will be used to link the WIFI networks with the passwords since they are relational data.

HASH varchar	MAC varchar	Client_Mac varchar	SSID varchar	Password varchar
00073752a4c03b51739e3a0378d0ed31	88:ac:c0:	5c:aa:fd:f		Revolver
0036da95aefc9469352394c57b687483	fc:7f:f1:f0	ac:de:48:		not4you1
0066807a3b57987b7176aa4fcaac4c0c	0c:81:12:	f0:fe:6b:e		222222222
00a1c1306502e86efbcc1dec448f986b	c8:1f:be:	b4:e1:eb:		14171303
010edf1384680eab2b2c6e4fadf6a21b	00:05:78:	86:fc:88:e		Energized

Figure 19. hashes table. (Author, 2022).

Then all of the raw hashes were imported. This will be used to correlate collected hashes with the vendor table.

```
LOAD DATA LOCAL INFILE 'all-raw.txt' INTO TABLE hashes FIELDS TERMINATED
BY ',' LINES TERMINATED BY '\n';
```

WPA	01	016169cf58bb58a511a9e...	5c:f4:ab:dc...	e0:cb:1d:...	54656c656e6f7233373533616e65
WPA	01	4cfcec9fa245ab4c60b544...	5c:f4:ab:dd...	e0:cb:1d:...	54656c656e6f72343938346c6976
WPA	01	a02813e60a8fa6ded838c...	60:31:97:1...	e0:cb:1d:...	54656c656e6f72323132376d696c
WPA	01	2c69b56d09ed47d68bbb8...	60:31:97:1...	e0:cb:1d:...	54656c656e6f7237303036757465
WPA	01	98e147678c2cd50bc269c...	60:31:97:1...	e0:cb:1d:...	41426f67454a
WPA	01	5bc540a2e6344bb79fa02...	34:21:09:2...	ac:67:84:...	4169724c696e6b326237333530
WPA	02	b73eabfa3d7f64e320886...	00:23:f7:1...	f0:fe:6b:b...	313131313131
WPA	02	df271a1f2c9385e1621c5...	00:23:f7:1...	2c:44:fd:...	4b4742

Figure 20. war_t table. (Author, 2022).

And lastly, all of the vendor names and the OUI of the MAC were imported into the vendor table. This will be used to correlate a vendor or company of manufacture with a WIFI Acces Point.

```
LOAD DATA LOCAL INFILE 'vendors.txt' INTO TABLE hashes FIELDS TERMINATED
BY ',' LINES TERMINATED BY '\n';
```

MAC varchar	VENDOR varchar
00:00:07	XEROX CORPORATION
00:00:08	XEROX CORPORATION
00:00:09	XEROX CORPORATION
00:00:0A	OMRON TATEISI ELECTRONICS CO.
00:00:0B	MATRIX CORPORATION
00:00:0C	Cisco Systems
00:00:0D	FIBRONICS LTD.
00:00:0E	FUJITSU LIMITED
00:00:0F	NEXT

Figure 21. war_t table. (Author, 2022).

7 - The Hash Data

In this section the author will show the passwords that were cracked with the different wordlist and rule combinations that were used in a dictionary attack against the hashes as well as a brute force attack and rule-attack. See figure 22 below for an example of the different attacks.

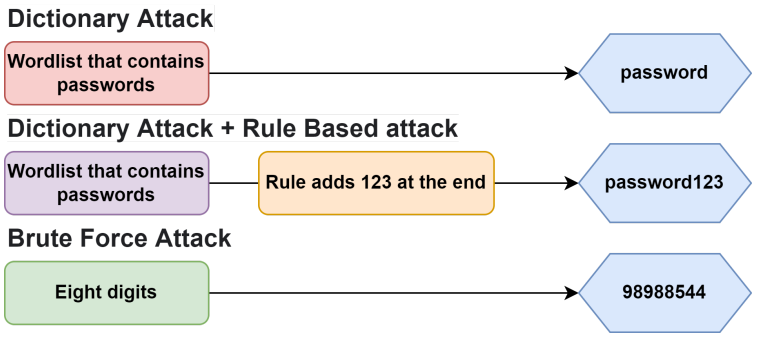

Figure 22. hashcat attacks. (Author, 2022).

In terms of hacking, a wordlist can be referred to as a dictionary of different words and well-known passwords that are stored in a plain text file (Abraham, 2021). And this wordlist is used in dictionary attacks and is also known as a “Wordlist attack”, as it reads the wordlist line by line and tries each line as a password candidate (Hashcat.net, 2022). The brute force attack tries a combination that is given such as an eight-digit number and it will go through all of the combinations from “00000000” to “9999999” before completing.

The rule-based attack works with the dictionary and brute force attacks, by adding rules such as adding 123 after all of the password candidates or making the password candidate full uppercase for lowercase such as “PASSWORD” or password.

In the password cracking phase different wordlist was used, where the top 10 passwords the cybernews did list the most common passwords (CyberNews, 2022)

Then the top 100 passwords from Wikipedia (Wikipedia Contributors, 2022).

The author made the hegnes wordlist with Norwegian words, names, last names, swear words, and all of the streets for the biggest Norwegian cities and more.

The rockyou wordlist was from a company that did develop widgets for MySpace, and in December of 2009, the company did have a data breach that did result in the exposure of personal data for their users. The company did store the passwords in plain text and the results were the rockyou wordlist.

And lastly, the brute force attack that tries every eight-digit combination, the reason for this is the Security Researcher Ido Hoorvitch did crack 44% of the collected hashes with this exact attack since a lot of WIFI networks did use a phone number for passwords (Hoorvich, 2021).

In the table below are the total hashes that are cracked with the different attack methods. And by the look of it, it shows that the rockyou wordlist is the wordlist that does recover most passwords, as well as the best64.rule. But due to the limited time of the project, there was not enough time to run best64.rule on rockyou as it would take over 40 days to complete. And that applies also to the brute force attack with the eight digits, as the rule will make the run time take too long. But by the look of the result from top 10 to hegnes wordlist, they all did recover 2.6% more passwords on average than “hegenes.rule”. With this in mind, it's possible to recover more passwords with the use of best64.rule on rockyou wordlist.

Then all of the cracked passwords were merged into one file and the duplicates were removed, making 549 password hashes cracked. And the same was done with the 5881 raw hashes, after removing duplicate lines it ended on 3962 hashes. It shows that 13.86% of the collected 3962 hashes were cracked.

	Rules			
wordlist	None	low-Up-Ti.rule	hegnes.rule	best64.rule
top 10	61	61	62	64
top 100	62	62	64	66
hegnes	132	169	210	213
rockyou	421	496	510	na
Eight digits	234	na	na	na

Table 2. Hashes cracked.

The below table shows the runtime of the different attack methods used. As seen in the runtime of hegnes.rule and best.64 shows that even though the best64.rule recovers 2.6% more passwords on average it's not efficient. The increase in the time to perform the attack from hegnes. rule to best64.rule it shows it increased by 102% on average. By the look of it, the best64.rule may crack 2.6% more passwords on average but the rule is not as power-efficient as the hegnes.rule. The Eight digits brute force attack is excluded from the total rule runtime.

	Rules				
wordlist	none	low-UP-Ti.rule	hegnes.rule	best64.rule	Total Runtime
top 10	48 sec	2 min	33 min	1 hour	1 hour, 38 min
top 100	1 min, 20 sec	4 min	1 hour	2 hours	3 hours, 17 min
hegnes	45 min	2 hours, 12 min	13 hours, 24 min	2 days, 14 hours	3 days, 13 hours
rockyou	8 hours, 7 min	1 day, 17 hours	20 days, 12 hours	40 days	22 days, 13 hours
Eight digits	4 days, 10 hours	-	-	-	4 days, 10 hours
Total Runtime	8 hours, 54 min	1 day 19 hours	21 days 3 hours	2 days 17 hours	

Table 3. Crack Runtime.

8 - Data Visualization

In this section the author will present the selected data with the use of Tableau which is a data visualization software, see section 4.2.

The Visualization data here will be the data from “Operation Bloodhound” which are the WIFI networks, and the password hashes. See figure 23 for the process.

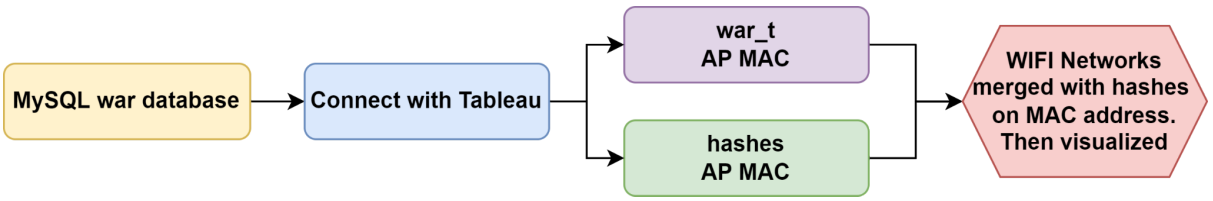

Figure 23. SQL DB to Visualization. (Author, 2022).

The selected data that are imported into the SQL database has relational data that it will be joined on and that is the MAC column in both tables, the reasoning for this is that the MAC address is a physical address, where the first three octets are assigned by the Institute of Electrical and Electronics Engineers (IEEE) to a vendor, manufacturer or an organization. And the last is assigned by the owner of the first three octets, combined, this makes it a unique identifier.

8.1 - Top 10 Passwords

As shown below in figure 24, the most common password in the dataset is “222222222” and occurs 111 times. And then the more common password combination where the numbers start at one and are ascending with such as “1234567890”, “12345678”.

There were also different networks that used phone numbers, and checking yellow pages shows that the WIFI Access Point was within the address registered. As well as WIFI networks that use the street number as the name and the street name as the password.

And the WIFI networks that did have GET and TP-Link in their SSID that was in the data set collected used an eight-digit combination, in total there were 149 WIFI networks with this combination.

Another finding was that many WIFI networks do have unique passwords, but many of them are just using a name or a word and adding the following “123” or “1234”.

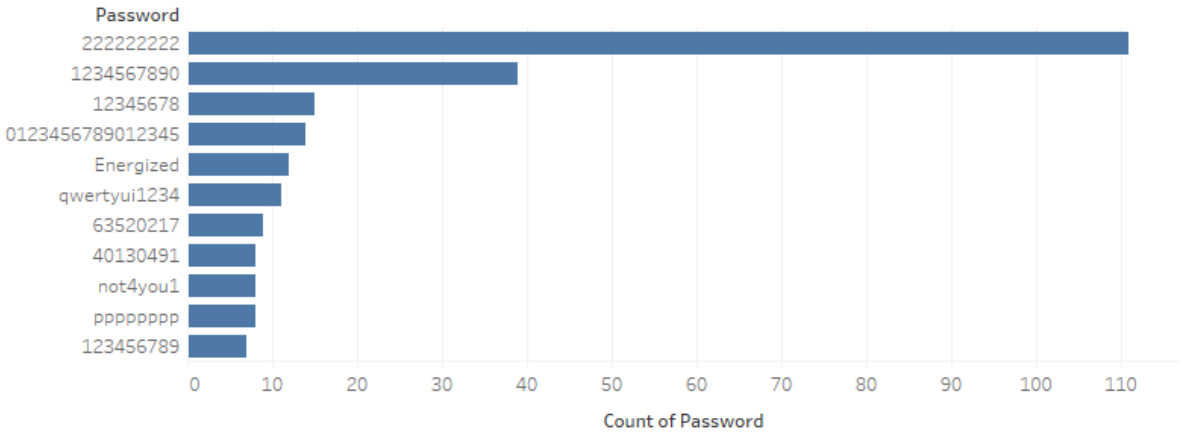

Figure 24. Top 10 Passwords. (Author, 2022).

Top 10 WIFI Password	Count of Password
222222222	111
1234567890	39
12345678	15
0123456789012345	14
Energized	12
qwertyui1234	11
63520217	9
pppppppp	8
not4you1	8
40130491	8
123456789	7

Table 4. Top 10 Passwords.

And figure 25 below shows a sample from the dataset of the different so-called “unique” password that doesn't repeat.

Abc123456	1	51094690	1
Adidas81	1	51212193	1
Admin1234	1	51652702	1
aftenposten	1	51758776	1
akersgata	1	53276605	1
Albert21	1	53835205	1
alexander	1	53878668	1
alohamora	1	54048414	1
Andreas1	1	5454545454	1
angeliukas	1	54682580	1
Apeldoorn	1	56615937	1
appelsin	1	57946313	1
arsenewenger	1	58500835	1
arubadub	1	59600466	1
asdf1234	1	60244974	1
askepott	1	60437524	1
Astrid14	1	60609934	1
aviation	1	60639073	1

Figure 25. Top 10 Passwords. (Author, 2022).

8.2 - Cracked WIFI networks visualized

As shown below in figure 26, those are the WIFI networks that were cracked, these are the cracked passwords that are matched with the WIFI networks that have a GPS location recorded. The names of the WIFI Access Points are removed due to privacy in figure 26.

This shows how easy it is to crack and not only make a list of the cracked network but also geo-locate the WIFI network to be able to go back at a later time.

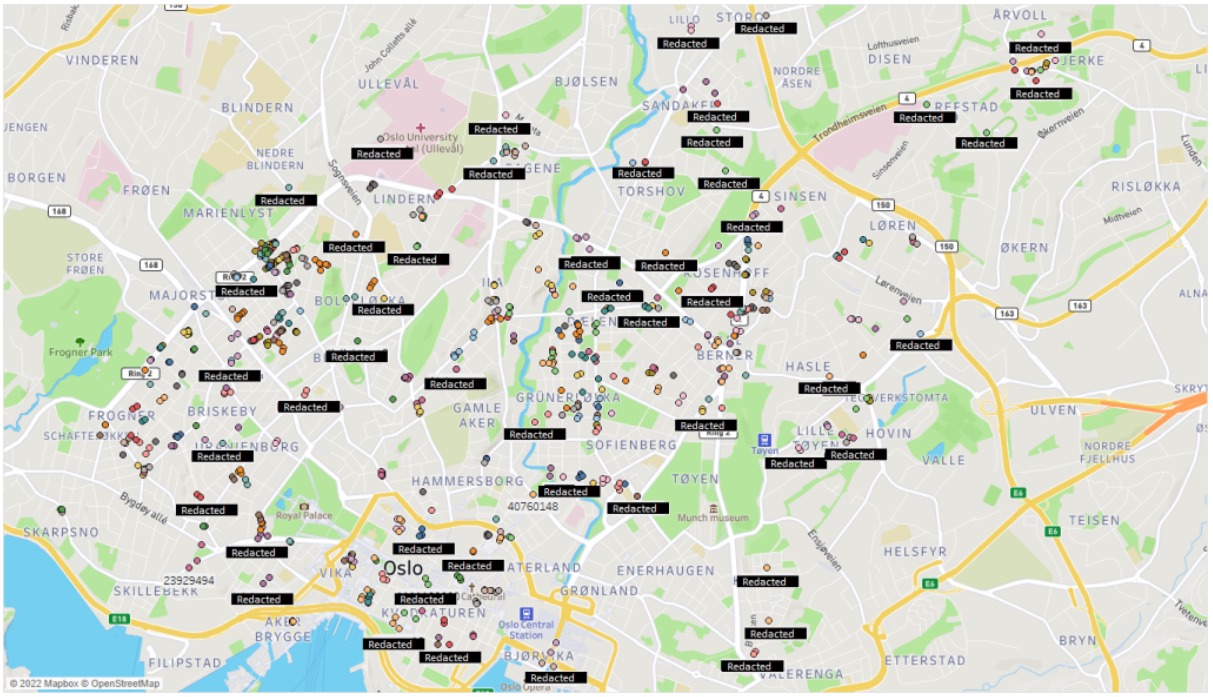

Figure 26. Cracked WIFI networks visualized. (Author, 2022).

This method can also be used in checking what passwords are most used in different areas as shown in figure 27, (due to geo-location accuracy passwords cant be shown).

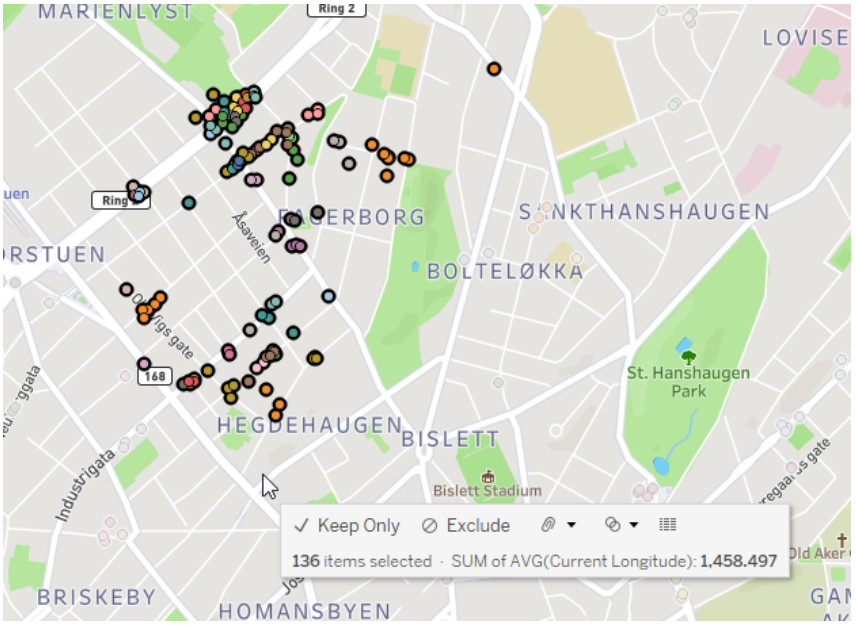

Figure 27. Cracked WIFI networks visualized. (Author, 2022).

It is even possible to see what vendor of the cracked WIFI Access Points. This was done by joining the data from two tables, one containing all of the vendor's OUI and vendor names joined with the hashes table.

```
SELECT hashes.MAC,vendor.VENDOR
FROM vendor, hashes
where left(hashes.MAC,8) = vendor.MAC
```

And below are the results from the SQL query.

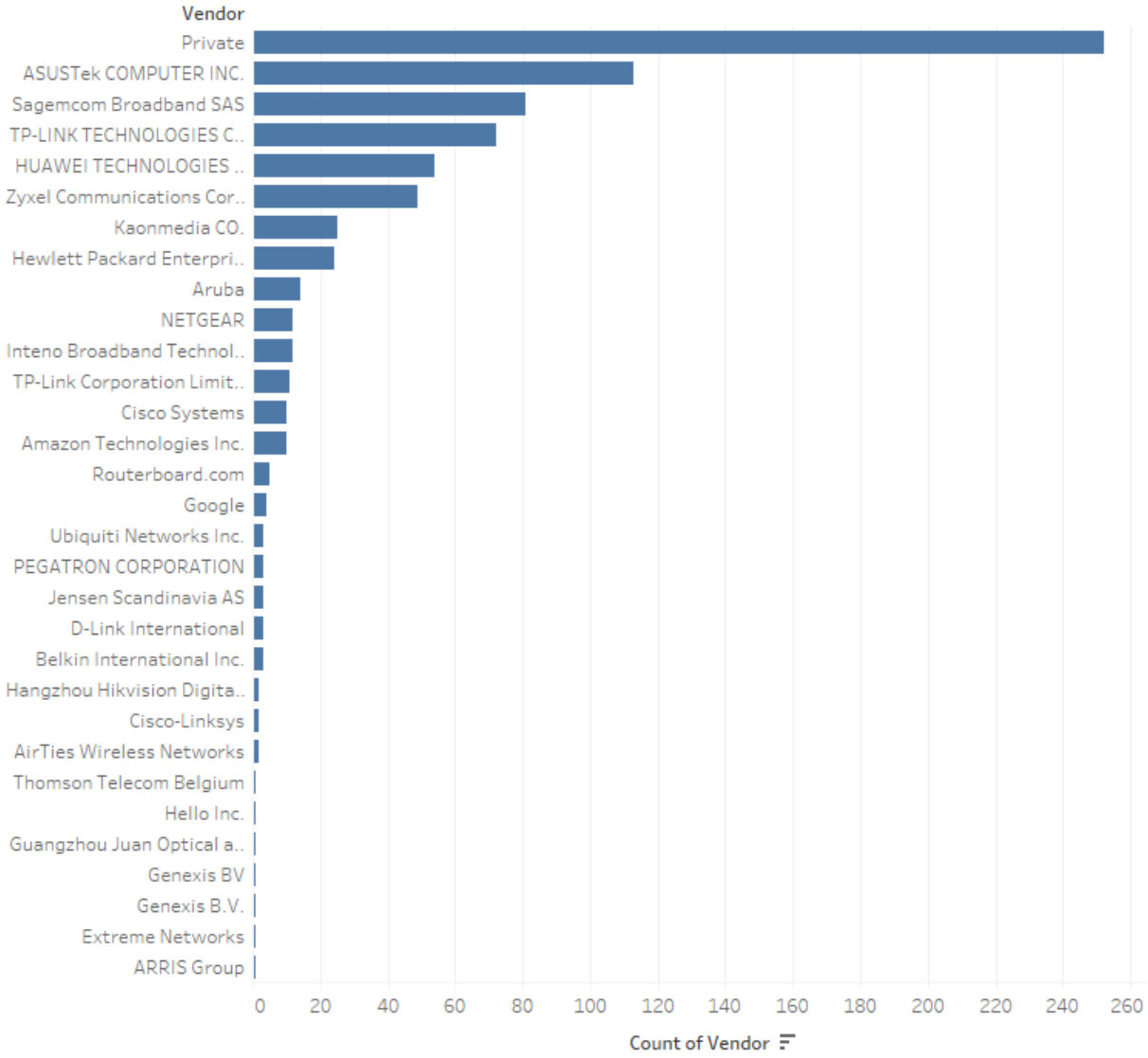

Figure 28. Vendors of cracked WIFI AP. (Author, 2022).

Then it's possible to see what vendor has the most vulnerable devices to these attacks in total. This was aggregated from all of the sample sizes of the raw data 5800 hashes. The data were joined on this SQL query. After importing the raw data into a table called raw.

```
SELECT raw.MAC,vendor.VENDOR
FROM vendor, raw
where left(raw.MAC,8) = vendor.MAC;
```

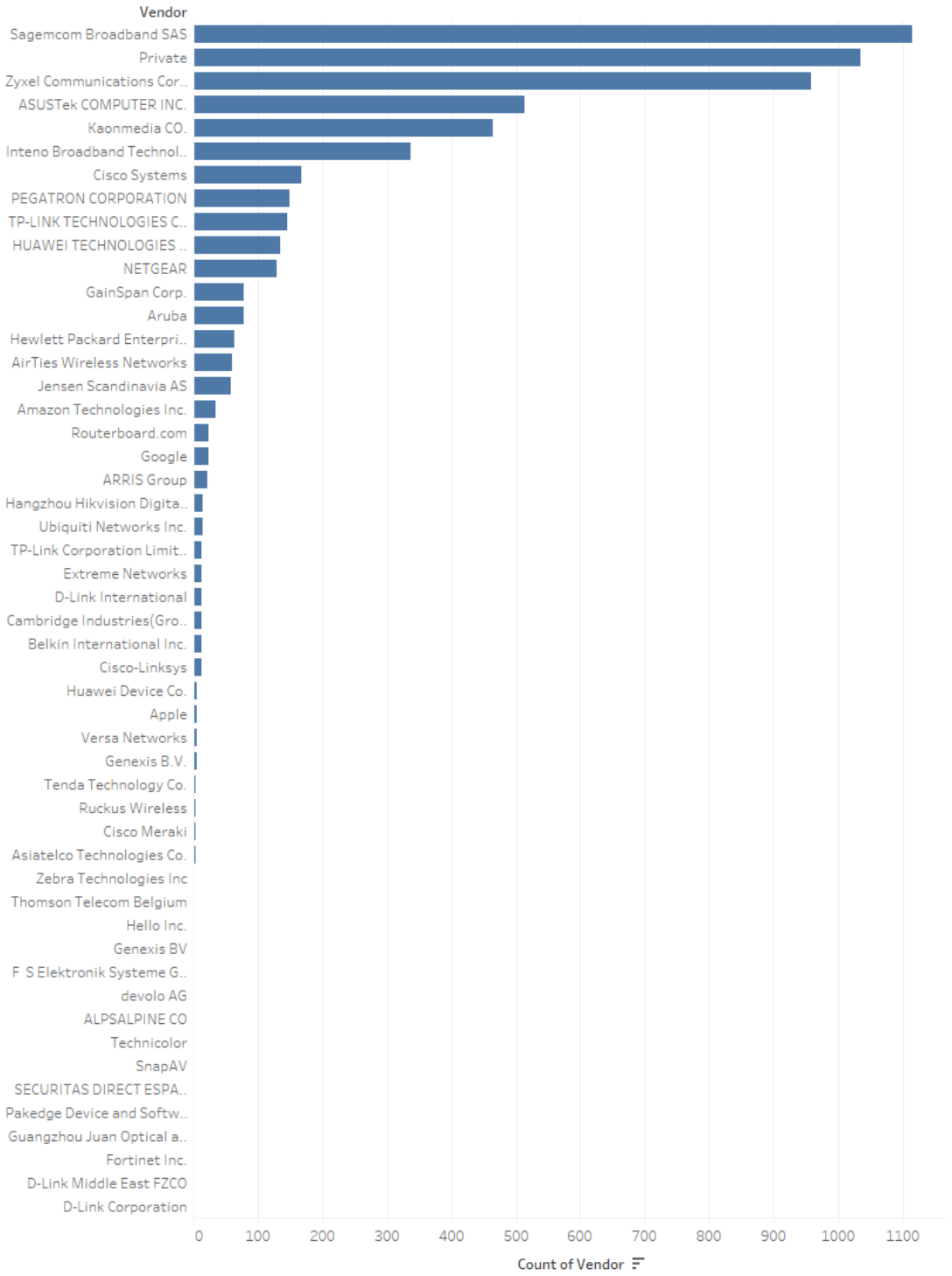

Figure 29. Vendors with vulnerable WIFI AP. (Author, 2022).

After finding what MAC address most of them used at the top, the author did use the SQL query of one sample from the Sagemcom Broadband MAC OUI.

```
SELECT SSID FROM war_t WHERE MAC LIKE 'ac:3b:77_________';
```

The result was the following: most of the WIFI Access Points name (SSID) did contain GET. This also shows that this attack can be used to find out what ISP provider that has the most vulnerable WIFI Access Point.

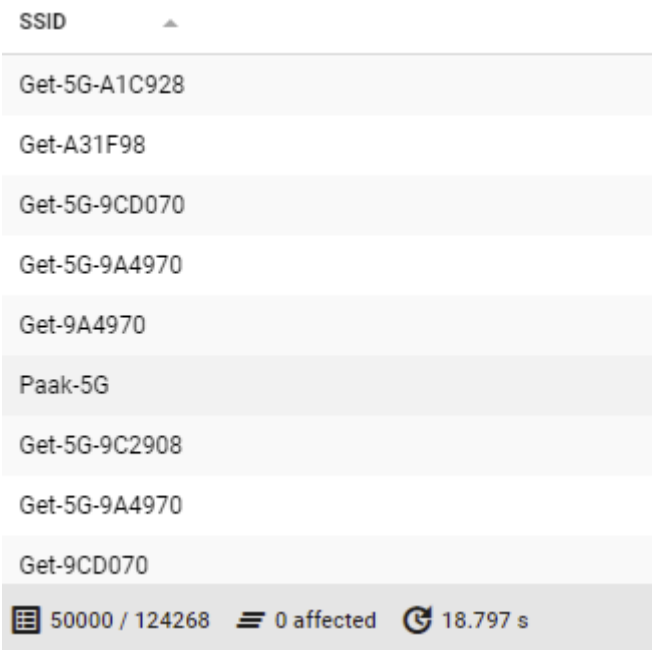

Figure 30.OUI join on SSID. (Author, 2022).

09 - What can the dataset be used for?

Sections five and seven showed how easy it was for the author to collect and crack the sample password hashes from a whole city within a small amount of time. The effort to gain unauthorized access to a network is quite low, compared to other methods.

This method can be used for Security Research and password analysis to check for the inhabitants of a whole city of a selected area what their password is for in the use of statistics of the most common password or password combination used. And this can be used to create stronger passwords.

The more insight the manufacturer has for its user regarding the creation of passwords in their product, the better safety parameters, and the better the security the manufacturer can provide for its users.

This data can be used in the research and development of new equipment, where the manufacturer can block the use of well known passwords that are in different wordlist “password lists” and password combinations. making the user of the product choose a good password. And block the user from setting up the equipment with easily crackable passwords such as “password123”.

This can also be used for nation-states to survey the security of their own infrastructure as well as for the population, and organizations. Such as locating WIFI networks that have outdated password security. Such as the FBI did in 2021 when they were authorized by the Department of Justice to remove scripts and web shells that did allow remote access that was infected on Exchange servers belonging to different organizations in eight states (Bankinfosecurity.com, 2013). As one of the biggest cyber crimes is industrial espionage, it's estimated that the USA may lose $200 billion to $600 billion dollars a year due to intellectual property theft by China (Cbsnews.com, 2021).

As more and more businesses and companies do offer remote work, it makes the organization have even more attack vectors. The number of professionals jobs hiring remotely has risen from 4% to 24% by the first quarter of 2022, showing that there might be an even higher percentage of remote workers in the future (Hughes, 2022) Not only is it for the potential threats from other devices on the home networks such as the family computer, but with this method, the authors have shown. It makes it possible to combine it with Open-source intelligence (OSINT), find out where the target lives, hobbies, and interests,s and make a specialized wordlist that is intended to crack the network of the target.

Security risks can be that the attacker can check for exploitable devices on the WIFI network and exploit these. Furthermore, they can have access to all shared drives on the network, which its possible to copy all confidential files, and private pictures, and even install malware. As did happen with the uber hack where the attacker did gain access to the company network via VPN. And from there did find a PowerShell script that included the username and password of a system administrator which was located on a network share. (Lily Hay Newman, 2022)

And another attack that is possible when being on the same network is a man-in-middle (MITM) attack. This makes it possible for the attacker to intercept and/or alter the data traffic between the victim and the destination such as a website (CSRC Content Editor, 2015).

With the MITM attack, the attacker gains access to your online accounts and gets bank and financial information, or worst case access to these accounts, and other accounts that have sensitive documents or personal photo albums where people store private photos.

The Seattle Police Department investigated a criminal hacking group driving through the city looking for business WiFi networks using insecure WEP encryption and compiled a list of locations. After finding a business Wi-Fi network with vulnerable WEP encryption, the hacker group began to crack the encryption to infiltrate the business network and then moved laterally to high-value applications such as a cashier system and install a credit card stealer. The hacking group ran the campaign from 2005 to 2007 and stole as many as 130 million credit card numbers before being caught and convicted (Hegnes, 2021).

For companies and big organizations that have remote workers, the issue can be as written before where the attacker can locate employees where they live with OSINT and crack and infiltrate their home WIFI networks. From there the attacker can gain access to the victim's computer and wait for a VPN connection to the organization's network and move laterally from there and infiltrate the network.

And in a time where more and more metadata about internet traffic for ISP subscribers gets collected, in other words, internet traffic history makes the ISP Subscriber prone to scrutiny about their internet activity. Such as one ISP subscriber did experience in 2009, he did get arrested on the suspicion of downloading illicit, illegal material from the internet. After three days the authorities did find out it was downloaded by a neighbor that did have access to his WIFI (McKay, 2022).

On another occasion, a person repeatedly hacked into his next-door- neighbor's wifi network in 2009. He tried to frame the couple with illegal material such as child pornography, sexual harassment and sending death threats to politicians, such as at the time vice president Joe Biden. His reasoning for this was to “get even” by launching computer attacks in a letter that was sent to the judge after the couple did tell the police that he did kiss their 4-year-old son on the mouth. An investigation into this led to that there was an undefined device on their network and after packet sniffing they found the name of the neighbor that did hack, after a search warrant was performed they did find manuals such as Simple WEP Crack Aircracking and Cracking WEP and the information about the couples WIFI Access Point (Kravets, 2011).

This shows that even if the people that own the WIFI Access Point are innocent, there can be a lot of legal problems for them, as well as slandering their reputation when an attacker does this form of extreme digital harassment.

10 - Conclusion

There was a lot of good data captured during the collection phase, but the hardware chosen for this project was not the most compatible and was a little flawed. The first issue the author did have was being able to power everything, a workaround was found by using two power banks and a USB hub with micro USB power input. Then the next issue was overheating, which made the author get the Cooler Master Raspberry Pi Case 40 and in addition put two ice packs in the backpack.

The next issue was the USB cable was of low quality and made the USB adapter crash constantly, this was fixed by using higher-quality cables.

Then the next issue was the Globalsat BU-353S4 GPS used by the author in the 2021 student project “Location Tracking of WIFI Access Point and Bluetooth Devices”, which had major issues getting GPS signals when there were buildings around in this project even on the same routes from last year, where it did get signal. The author did fix this by forwarding the GPS location from an Android smartphone to the Raspberry PI and that did fix it. As seen below highlighted in red is the area that is missing the GPS location data when the author did use the Globalsat BU-353S4 GPS.

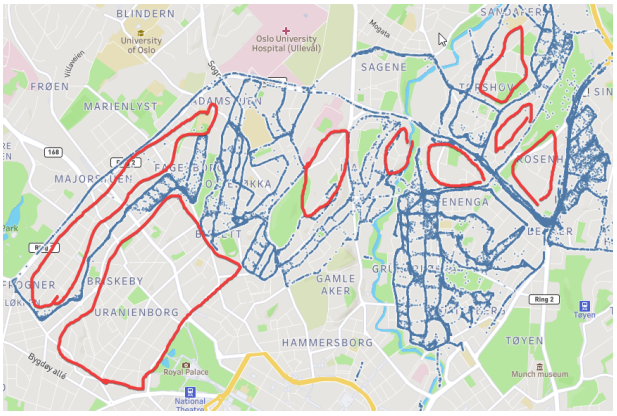

Figure 28. Missing GPS locations. (Author, 2022).

If the author had a bigger hardware budget, the following would be preferred. A Raspberry Pi 4 with a case that actively cools, four of the ALFA Network AWUS036ACM adapter, a dedicated smartphone for sharing GPS, a USB hub for powering the adapters, and lastly only one powerbank that have enough Watt to power all of it, and all of this is intended for a city-scale attack. For smaller attacks or a specific network, the budget that is needed is about 300 NOK, as the attacker only needs an adapter with monitor mode, a laptop, and a couple of hours to spare.

Of the sample size of 5881 (3962 unique, see section 7) hashes, there were 15.27% that were crackable using the rule with the commonly used passwords and number combinations, this was after removing duplicate rows in the result. As this made it easy for the author to crack the easiest password of the WIFI Access Points with roaming enabled and were in proximity during the data collection phase. In a 2019 Google study about passwords, the findings show that 24% of Americans have used common passwords that the author did use, as well as the use of a pet's, partner, child's, or own name (Comparitech.com, 2022).

This shows in the dataset that there are several names with combinations of numbers such as 123, making it likely this study about passwords does correlate to WIFI networks passwords, see section 8.1.

The preventive measure that can be used to protect the user's WIFI Access Point is by choosing a complex password with a combination of numbers, symbols, uppercase characters, and lowercase characters with a minimum of ten characters as this example “N8!yc7FB4E”. When trying to use a brute force attack on the collected hashes, it is estimated that it will be completed in 191884 years, 116 days.

Other preventive measures are to enable only WP3 for the WIFi if the connected devices support it and not to use WIFI for life-critical use cases.

One other finding was that the author did notice that of those WIFI Access Points that were cracked many of them had changed their default name (SSID), so in this example when a WIFI Access Point has an SSID that is “Tylder-Durden-WIFI”, it can be assumed that the password is also changed and be guessed as ”FightClub”.

It would seem that the greatest danger towards security is human nature: a sense of complacency and “slight aversion” against learning something new and complex.
More precisely, the average human mind cannot easily remember a 16-character alfa-numerical password, composed of upper -and lower case letters, symbols, and numbers, let alone several of such passwords. Secondly, in terms of accessibility for persons who are connecting to a network in a public space (ie gym, cafe, etc.), any customers who would be required to input such a password, would most likely not see the security value of such a password, but more of a hassle in trying to write it in on their smartphones/laptops.

Not all WIFI Access Points have the WIFI roaming functionality and are not vulnerable to PMKID attacks. But the author's research found that WIFI routers made by many of the world's largest vendors are vulnerable, but the most vulnerable were the proprietary WIFI Access Points provided by the ISP, such as Sagemcom Broadband SAS. See section 8.2.

Other findings were during Operation Bloodhound, where the author did walk rather than bicycling like last year's project “Location Tracking of WIFI Access Point and Bluetooth Devices”, with the use of only one Wireless adapter with Kismet. It shows that walking did collect three times more data per 1 km than bicycling.

In hindsight after data collection and data preparation, the author did find it was scaringly easy to gain unauthorized access (if that were the end goal) to people's private WIFI networks, and even different business networks, such as one big company that has several locations around Oslo, that used a simple password on their IoT WIFI Access Point that can be found in the rockyou wordlist. Looking at the data, it is even easy to identify people who own the WIFI Access Point, as some chose to use their phone numbers. It is the author’s opinion that this is a tech area where many do know about these vulnerabilities and know how to avoid them, but downright choose to ignore them.

11 - References

Abraham (2021). Kali Linux Wordlist: What you need to know. [online] FOSS Linux. Available at: https://www.fosslinux.com/48115/kali-linux-wordlist-what-you-need-to-know.htm [Accessed 2 Jun. 2022].
Hashcat.net. (2022). dictionary_attack [hashcat wiki]. [online] Available at: https://hashcat.net/wiki/doku.php?id=dictionary_attack [Accessed 2 Jun. 2022].

an (2020). What is PMKID? Why would even a router give away the PMKID to an unauthorized stranger? [online] Super User. Available at: https://superuser.com/questions/1547307/what-is-pmkid-why-would-even-a-router-give-away-the-pmkid-to-an-unauthorized-st [Accessed 9 Jun. 2022].

area networks- Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Medium Access Control (MAC) Security Enhancements IEEE Computer Society IEEE Standards. IEEE Stds, [online] 802, pp.11–1999. Available at: https://paginas.fe.up.pt/~jaime/0506/SSR/802.11i-2004.pdf [Accessed 11 Jun. 2022].

atom (2018). New attack on WPA/WPA2 using PMKID. [online] Hashcat.net. Available at: https://hashcat.net/forum/thread-7717.html [Accessed 27 May 2022].

Bankinfosecurity.com. (2013). FBI Removing Web Shells From Infected Exchange Servers. [online] Available at: https://www.bankinfosecurity.com/fbi-removing-web-shells-from-infected-exchange-servers-a-16399 [Accessed 7 Jun. 2022].

Cbsnews.com. (2021). Top counterintelligence official Mike Orlando on foreign espionage threats facing U.S. - ‘Intelligence Matters’. [online] Available at: https://www.cbsnews.com/news/foreign-espionage-threats-u-s-intelligence-matters-podcast/ [Accessed 7 Jun. 2022].

Cisco Meraki. (2020). PMKID Vulnerability FAQ - WPA/WPA2-PSK and 802.11r. [online] Available at: https://documentation.meraki.com/MR/Other_Topics/PMKID_Vulnerability_FAQ_-_WPA%2F%2FWPA2-PSK_and_802.11r [Accessed 16 May 2022].

Comparitech.com. (2022). 25+ Password Statistics that may change your password habits. [online] Available at: https://www.comparitech.com/blog/information-security/password-statistics/ [Accessed 9 Jun. 2022].

CSRC Content Editor (2015). man-in-the-middle attack (MitM) - Glossary | CSRC. [online] Nist.gov. Available at: https://csrc.nist.gov/glossary/term/man_in_the_middle_attack [Accessed 8 Jun. 2022].

CyberNews. (2022). Best Password Manager Review: Dashlane, LastPass, 1Password, NordPass, RememBear. [online] Available at: https://cybernews.com/best-password-managers/most-common-passwords/ [Accessed 2 Jun. 2022].

GeeksforGeeks. (2019). screen command in Linux with Examples - GeeksforGeeks. [online] Available at: https://www.geeksforgeeks.org/screen-command-in-linux-with-examples/ [Accessed 31 May 2022].

GPSd. (2022). GPSd — Put your GPS on the net! [online] Available at: https://gpsd.gitlab.io/gpsd/ [Accessed 30 May 2022].

hashcat (2021). GitHub - hashcat/hashcat: World's fastest and most advanced password recovery utility. [online] GitHub. Available at: https://github.com/hashcat/hashcat [Accessed 30 May 2022].

Hegnes.tech. (2021). Location Tracking of WIFI Access Point and Bluetooth Devices – Hegnes. [online] Available at: https://www.hegnes.tech/2021/09/01/location-tracking-of-wifi-access-point-and-bluetooth-devices/#htoc-4-1-hardware [Accessed 30 May 2022].

Hoorvich, I. (2021). Cracking WiFi at Scale with One Simple Trick. [online] Cyberark.com. Available at: https://www.cyberark.com/resources/threat-research-blog/cracking-wifi-at-scale-with-one-simple-trick [Accessed 2 Jun. 2022].

Hughes, O. (2022). Is remote working here to stay? The data might surprise you. [online] ZDNet. Available at: https://www.zdnet.com/article/is-remote-working-here-to-stay-the-data-might-surprise-you/ [Accessed 7 Jun. 2022].

IEEE (2001). IEEE Standards IEEE Standard for Information technology-Telecommunications and information exchange between systems- Local and metropolitan area networks- Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Medium Access Control (MAC) Security Enhancements IEEE Computer Society IEEE Standards. IEEE Stds, [online] 802, pp.11–1999. Available at: https://paginas.fe.up.pt/~jaime/0506/SSR/802.11i-2004.pdf [Accessed 11 Jun. 2022].

Kismet. (2022). Kismet. [online] Available at: https://www.kismetwireless.net/ [Accessed 30 May 2022].

Kohlios, C. and Hayajneh, T. (2018). A Comprehensive Attack Flow Model and Security Analysis for Wi-Fi and WPA3. Electronics, [online] Available at: https://www.mdpi.com/2079-9292/7/11/284/htm

Kravets, D. (2011). Wi-Fi–Hacking Neighbor From Hell Sentenced to 18 Years. [online] Wired. Available at: https://www.wired.com/2011/07/hacking-neighbor-from-hell/ [Accessed 9 Jun. 2022].

Lily Hay Newman (2022). The Uber Hack's Devastation Is Just Starting to Reveal Itself. [online] WIRED. Available at: https://www.wired.com/story/uber-hack-mfa-phishing/ [Accessed 16 Oct. 2022].

Margaritelli, S. (2019). Pwning WPA/WPA2 Networks With Bettercap and the PMKID Client-Less Attack. [online] evilsocket. Available at: https://www.evilsocket.net/2019/02/13/Pwning-WiFi-networks-with-bettercap-and-the-PMKID-client-less-attack/ [Accessed 8 Jun. 2022].

McKay, T. (2022). Risks of Hosting an Open Wi-Fi Network. [online] Rittercommunications.com. Available at: https://blog.rittercommunications.com/risks-of-hosting-an-open-wi-fi-network [Accessed 5 Jun. 2022].

Ned Hardy. (2020). Why Are Bloodhounds Such Good Tracking Dogs? [online] Available at: https://nedhardy.com/2020/02/29/bloodhounds/ [Accessed 11 Jun. 2022].

Parsi, N. (2022). How Roam , PMK caching, OKC and Pre-auth works. [online] Blogspot.com. Available at: http://ilovewifi.blogspot.com/2012/10/how-roam-pmk-cachingokc-and-pre-auth.html [Accessed 27 May 2022].

PCWorld. (2011). Seattle Police Say "wardrivers" Are Hitting Small Businesses. [online] Available at: https://www.pcworld.com/article/226086/article.html [Accessed 29 May 2021].

Pwnagotchi.ai. (2022). Pwnagotchi - Deep Reinforcement Learning instrumenting bettercap for WiFi pwning. [online] Available at: https://pwnagotchi.ai/ [Accessed 6 Jun. 2022].

Tay, K. (2020). Wi-Fi de-authentication attacks and how you can prevent them using 802.11w or WPA3. [online] Medium. Available at: https://x4bx54.medium.com/use-802-11w-or-wpa3-to-prevent-de-authentication-attacks-in-your-wi-fi-network-4ce63ab20033 [Accessed 16 Oct. 2022].

Techopedia (2013). Four-Way Handshake. [online] Techopedia.com. Available at: https://www.techopedia.com/definition/27188/four-way-handshake [Accessed 16 May 2022].

The Raspberry Pi Foundation (2021). Operating system images – Raspberry Pi. [online] Raspberry Pi. Available at: https://www.raspberrypi.org/software/operating-systems/ [Accessed 7 May 2021].

tiagoshibata (2021). GPSd Client. [online] Google.com. Available at: https://play.google.com/store/apps/details?id=io.github.tiagoshibata.gpsdclient&hl=en&gl=US [Accessed 30 May 2022].

Tsui, A.W.T., Lin, W.-C., Chen, W.-J., Huang, P. and Chu, H.-H. (2010). Accuracy Performance Analysis between War Driving and War Walking in Metropolitan Wi-Fi Localization. IEEE Transactions on Mobile Computing, 9(11), pp.1551–1562. doi:10.1109/tmc.2010.121.
wifi-professionals (2019). 4-Way Handshake - WiFi. [online] WiFi. Available at: https://www.wifi-professionals.com/2019/01/4-way-handshake [Accessed 11 Jun. 2022].

Wikipedia Contributors (2022). Wikipedia:10,000 most common passwords. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Wikipedia:10,000_most_common_passwords [Accessed 2 Jun. 2022].