PORTSPOOFING:-

STEPS-

1. **Iptables -F** :- After giving this command, if you want to see the current policy, you can check it with this command.

Command/Command Option	**Description**
Iptables	Linux default firewall.
-A	Appends the iptables rule to the end of the specified chain. This is the command used to add a rule when order in the chain does not matter.
-t	Specifies the table name which we are going to use.
-i	Selects the interface.
-m	Additional match options are also available through modules loaded by the `iptables` command. To use a match option module, load the module by name using the `-m` option, such as `-m <module-name>` (replacing *<module-name>* with the name of the module).
-p	Sets the default policy for the specified chain, so that when packets traverse an entire chain without matching a rule, they are sent on to the specified target, such as ACCEPT or DROP.
-dport	Sets a destination port
-j	Jump
-to-ports	Destination port to forward

After downloading portspoof we check it using portspoof -h and its working as you can see.

```
┌──(amritanshi㉿kali)-[~]
└─$ sudo iptables -F
[sudo] password for amritanshi:

┌──(amritanshi㉿kali)-[~]
└─$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

┌──(amritanshi㉿kali)-[~]
└─$ portspoof -h
Usage: portspoof [OPTION]...
Portspoof - service emulator / frontend exploitation framework.

-i                  ip : Bind to a particular  IP address
-p                  port : Bind to a particular PORT number
-s                  file_path : Portspoof service signature regex. file
-c                  file_path : Portspoof configuration file
-l                  file_path : Log port scanning alerts to a file
-f                  file_path : FUZZER_MODE - fuzzing payload file list
-n                  file_path : FUZZER_MODE - wrapping signatures file list
-1                  FUZZER_MODE - generate fuzzing payloads internally
-2                  switch to simple reply mode (doesn't work for Nmap)!
-D                  run as daemon process
-d                  disable syslog
-v                  be verbose
-h                  display this help and exit
```

Now it is time to forward those packets to portspoof in order to reply the client machine. To do so, use the following command:

```
┌──(amritanshi㉿kali)-[~]
└─$ sudo iptables -t nat -A PREROUTING -i eth0 -p tcp -mtcp --dport 1:65535 -j REDIRECT --to-ports 4444
```

first, it will collect all the packets accepted by iptables and then it will forward them to the 4444 port, which is by default a port of our portspoof tool.

Two mandatory options are needed to run the portspoof. The command to run portspoof is:

```
┌──(amritanshi㉿kali)-[~]
└─$ portspoof -c /usr/local/etc/portspoof.conf -s /usr/local/etc/portspoof_signatures
→ Using user defined configuration file /usr/local/etc/portspoof.conf
→ Using user defined signature file /usr/local/etc/portspoof_signatures
```

To check the ip address of our kali linux to scan from attacker:-

```
┌──(amritanshi㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.21.128  netmask 255.255.255.0  broadcast 192.168.21.255
        inet6 fe80::20c:29ff:fed2:9bbe  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:d2:9b:be  txqueuelen 1000  (Ethernet)
        RX packets 5318  bytes 348602 (340.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3214  bytes 213161 (208.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 6026  bytes 301324 (294.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 6026  bytes 301324 (294.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Now it is time to scan from our attacker machine (UBUNTU):-

```
amy99@ubuntu:~$ sudo nmap 192.168.21.128
Starting Nmap 7.80 ( https://nmap.org ) at 2021-04-10 17:25 IST
Nmap scan report for 192.168.21.128
Host is up (0.0015s latency).

PORT      STATE SERVICE
1/tcp     open  tcpmux
3/tcp     open  compressnet
4/tcp     open  unknown
6/tcp     open  unknown
7/tcp     open  echo
9/tcp     open  discard
13/tcp    open  daytime
17/tcp    open  qotd
19/tcp    open  chargen
20/tcp    open  ftp-data
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
24/tcp    open  priv-mail
25/tcp    open  smtp
26/tcp    open  rsftp
30/tcp    open  unknown
32/tcp    open  unknown
33/tcp    open  dsp
37/tcp    open  time
42/tcp    open  nameserver
43/tcp    open  whois
49/tcp    open  tacacs
53/tcp    open  domain
70/tcp    open  gopher
79/tcp    open  finger
80/tcp    open  http
81/tcp    open  hosts2-ns
82/tcp    open  xfer
83/tcp    open  mit-ml-dev
84/tcp    open  ctf
85/tcp    open  mit-ml-dev
88/tcp    open  kerberos-sec
50006/tcp open  unknown
50300/tcp open  unknown
50389/tcp open  unknown
50500/tcp open  unknown
50636/tcp open  unknown
50800/tcp open  unknown
51103/tcp open  unknown
51493/tcp open  unknown
52673/tcp open  unknown
52822/tcp open  unknown
52848/tcp open  unknown
52869/tcp open  unknown
54045/tcp open  unknown
54328/tcp open  unknown
55055/tcp open  unknown
55056/tcp open  unknown
55555/tcp open  unknown
55600/tcp open  unknown
56737/tcp open  unknown
56738/tcp open  unknown
57294/tcp open  unknown
57797/tcp open  unknown
58080/tcp open  unknown
60020/tcp open  unknown
60443/tcp open  unknown
61532/tcp open  unknown
61900/tcp open  unknown
62078/tcp open  iphone-sync
63331/tcp open  unknown
64623/tcp open  unknown
64680/tcp open  unknown
65000/tcp open  unknown
65129/tcp open  unknown
65389/tcp open  unknown
MAC Address: 00:0C:29:D2:9B:BE (VMware)

Nmap done: 1 IP address (1 host up) scanned in 6.06 seconds
amy99@ubuntu:~$
```

As you can see, starting from 1, it will show all 65535 ports open. Actually these ports are not actually open and some don't eve exist, but this is how we are fooling the attacker to make him see all 65535 ports are opened. If you want to scan that host with any signature within nmap then it will show as below. I am using nmap with the –v and –A options. Then the result, will be as shown below: