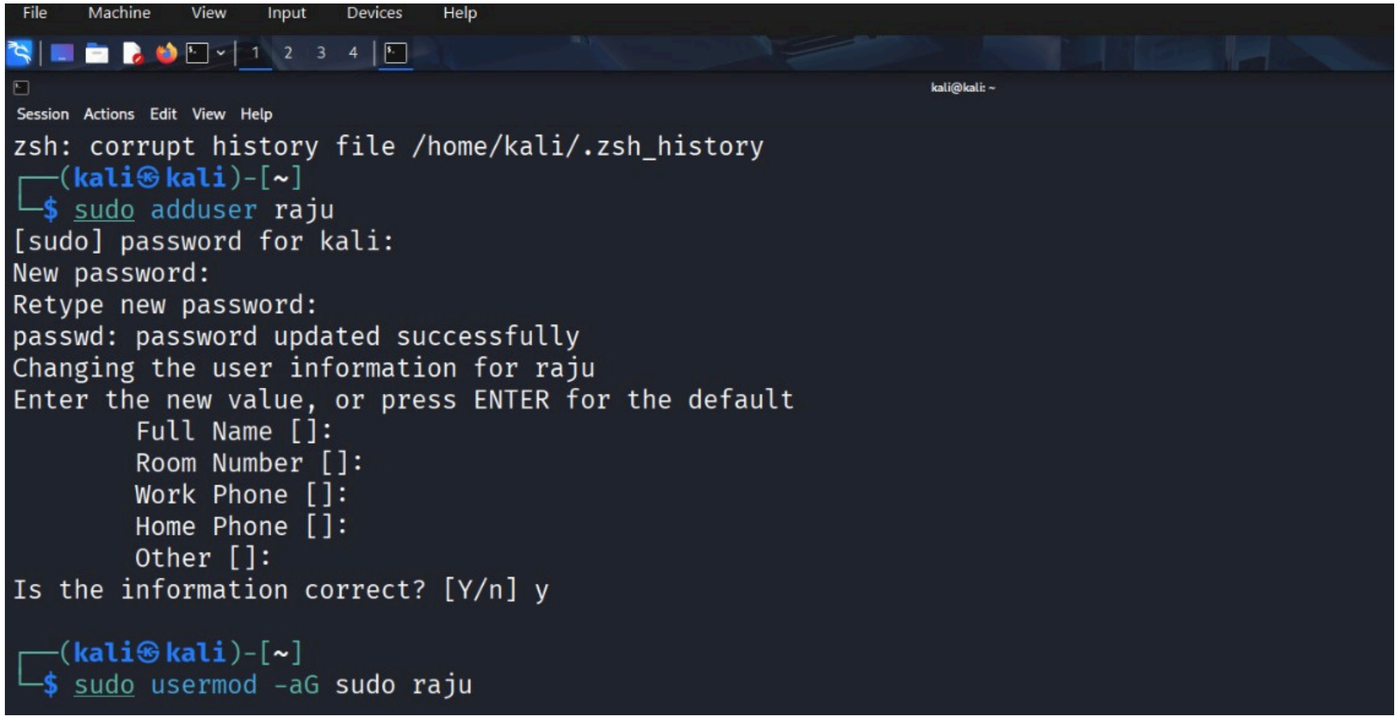
Creating User name

run : sudo adduser raju

this will create user with raju name

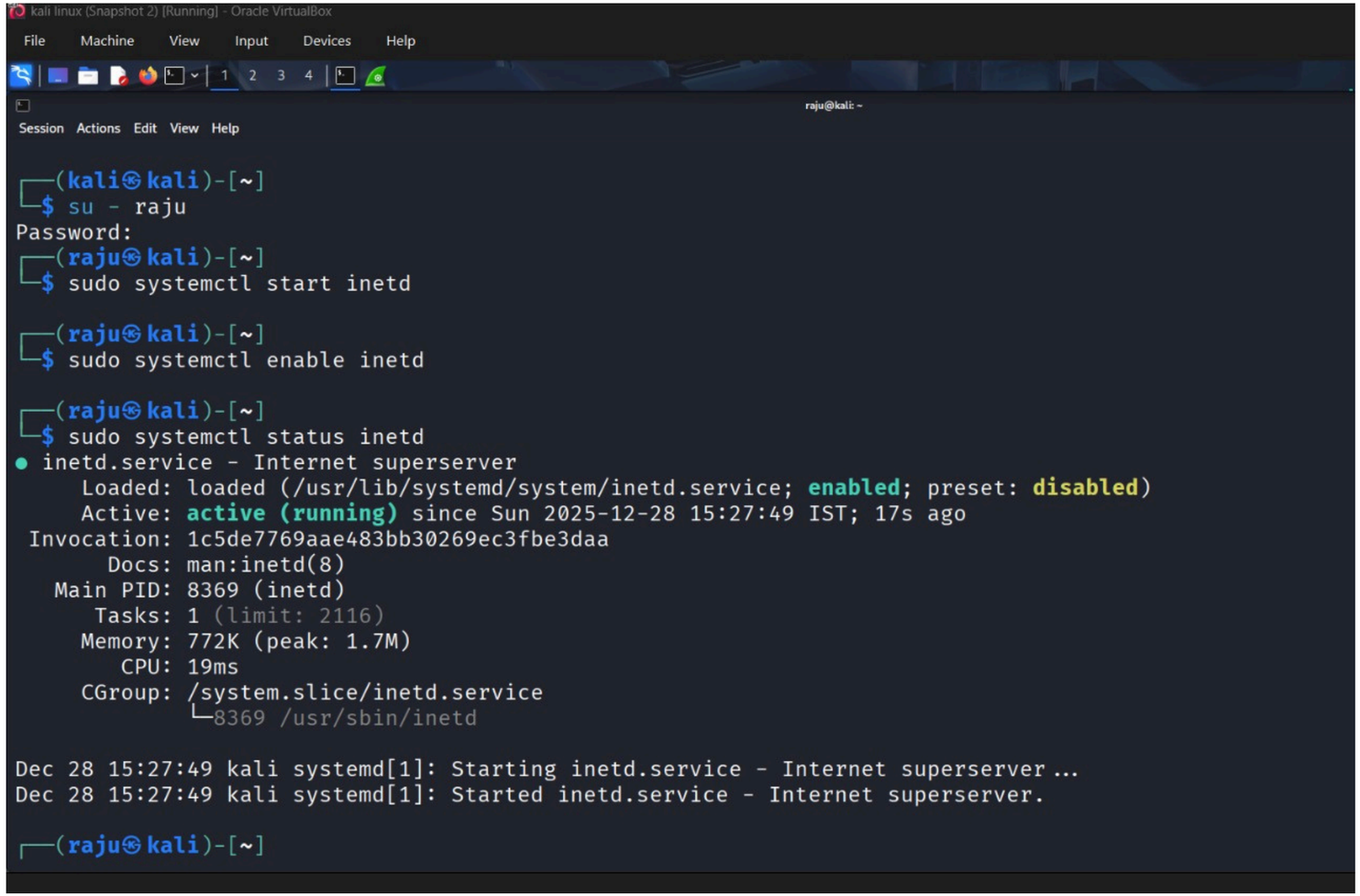
Run : sudo usermod -aG sudo raju

this will add this user to sudo file



Checking status running or not

Run : su- raju
Run : sudo systemctl start inetd
to start server.

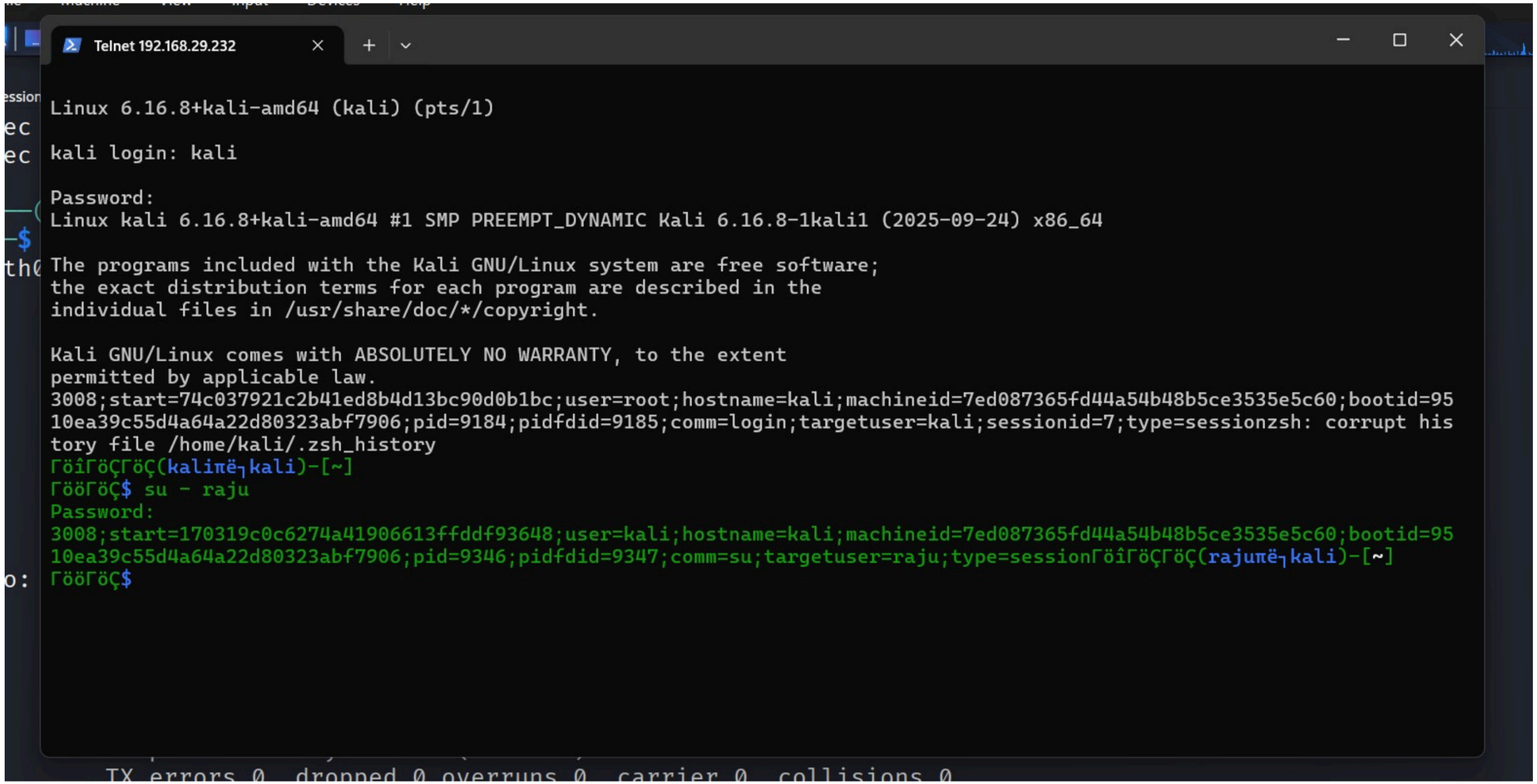


Getting ip address

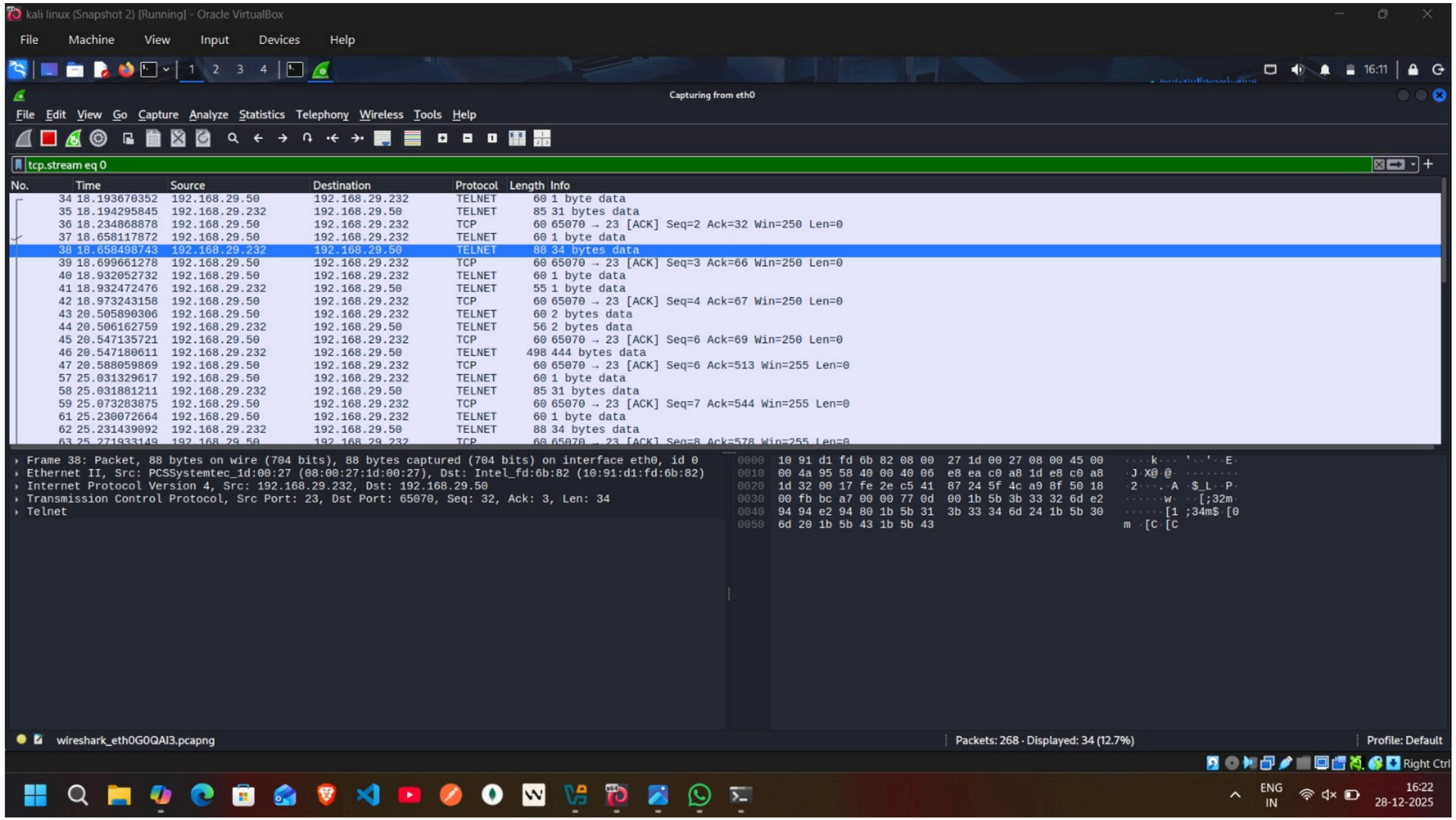
```
┌──(raju㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.29.232  netmask 255.255.255.0  broadcast 192.168.29.255
        inet6 2405:201:3006:58d7:a00:27ff:fe1d:27  prefixlen 64  scopeid 0×0<global>
        inet6 fe80::a00:27ff:fe1d:27  prefixlen 64  scopeid 0×20<link>
        inet6 2405:201:3006:58d7:c7a3:b975:4cb7:4b94  prefixlen 64  scopeid 0×0<global>
        ether 08:00:27:1d:00:27  txqueuelen 1000  (Ethernet)
        RX packets 1226  bytes 97939 (95.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 304  bytes 34442 (33.6 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 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 8  bytes 480 (480.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 480 (480.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

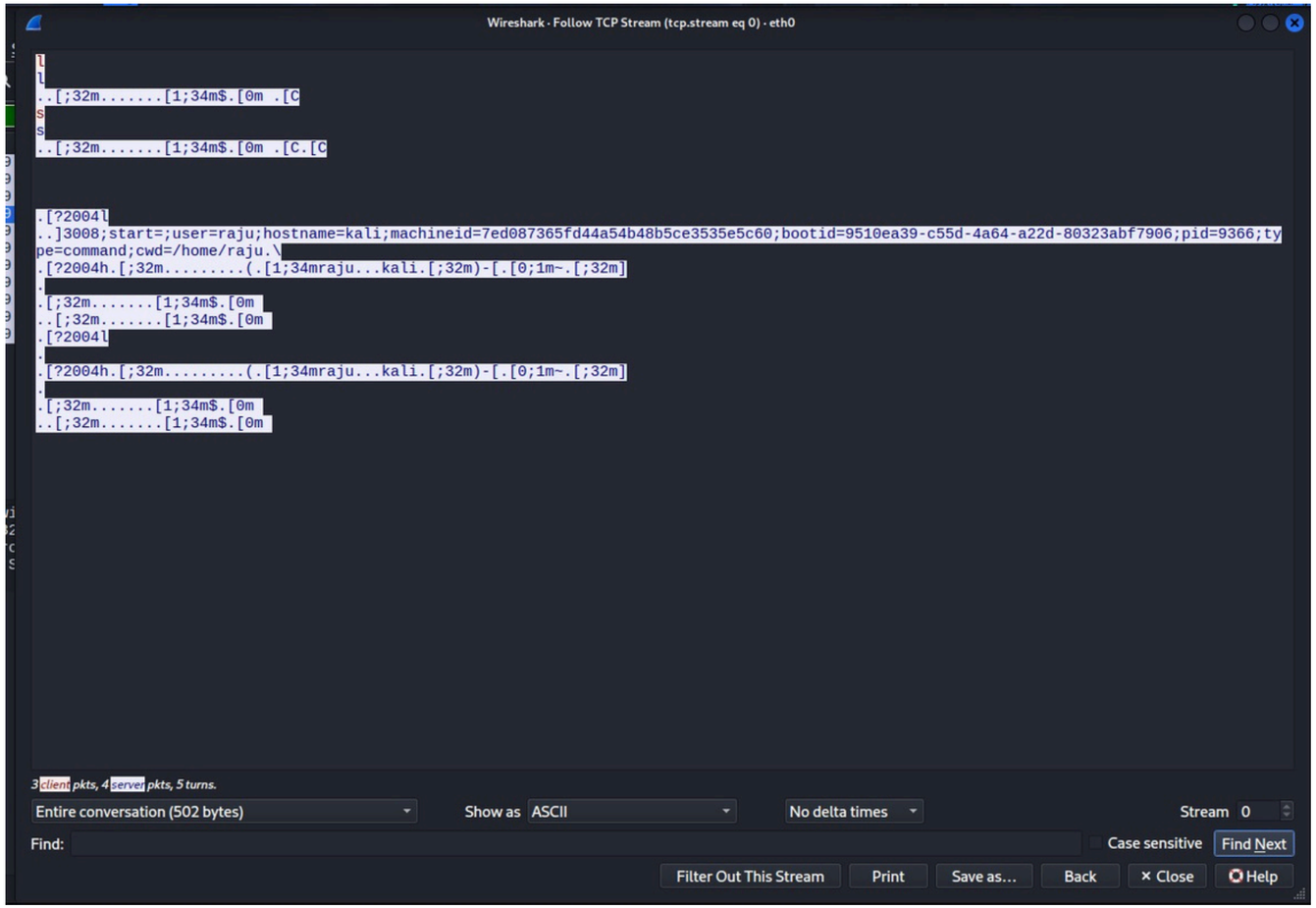
Getting access of victim os



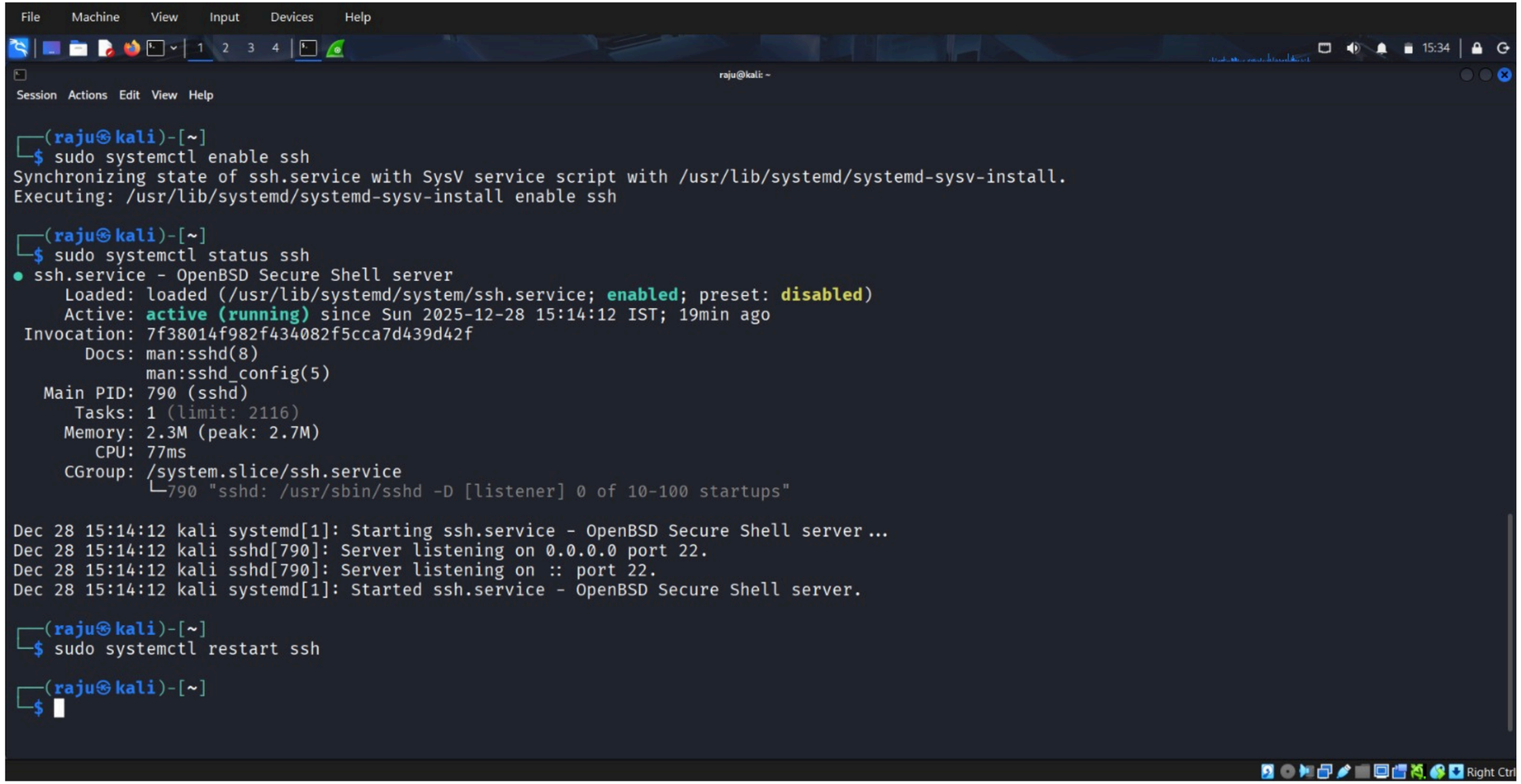
Observing telnet through wireshark



All file in redable format



Starting SSH server and Getting it running

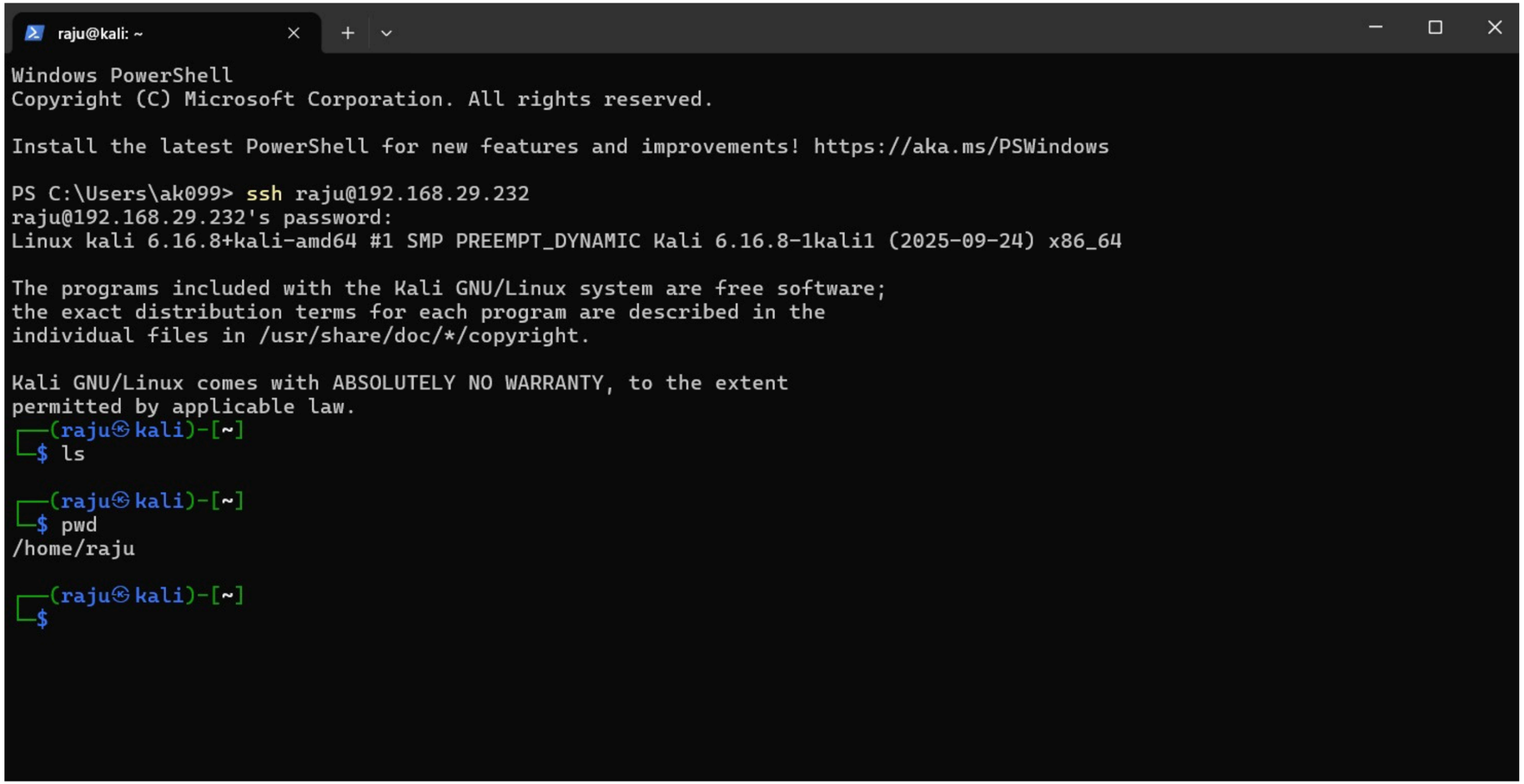


Getting ip Address

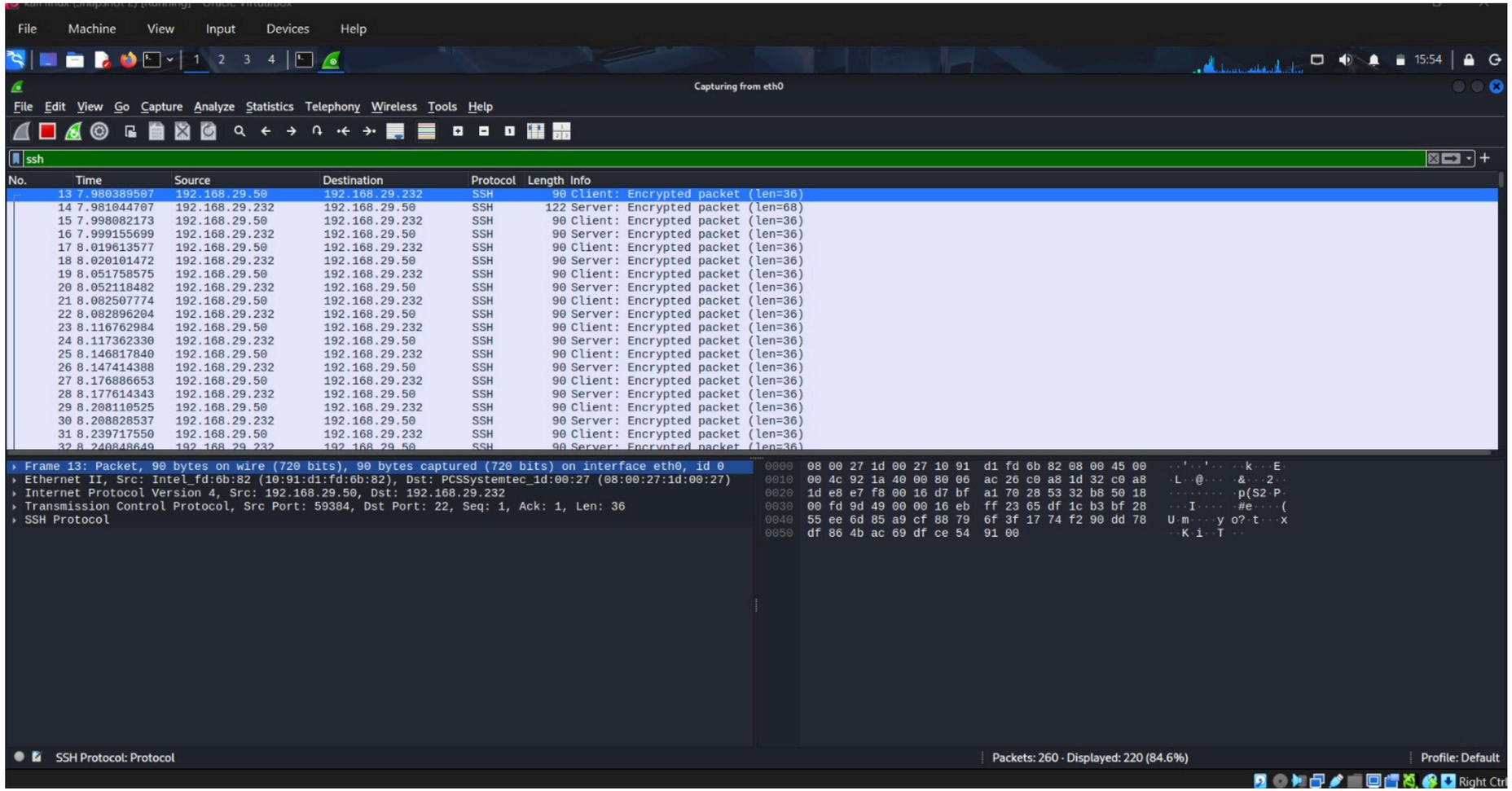
```
┌──(raju㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.29.232  netmask 255.255.255.0  broadcast 192.168.29.255
        inet6 2405:201:3006:58d7:a00:27ff:fe1d:27  prefixlen 64  scopeid 0×0<global>
        inet6 fe80::a00:27ff:fe1d:27  prefixlen 64  scopeid 0×20<link>
        inet6 2405:201:3006:58d7:c7a3:b975:4cb7:4b94  prefixlen 64  scopeid 0×0<global>
        ether 08:00:27:1d:00:27  txqueuelen 1000  (Ethernet)
        RX packets 1226  bytes 97939 (95.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 304  bytes 34442 (33.6 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 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 8  bytes 480 (480.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 480 (480.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

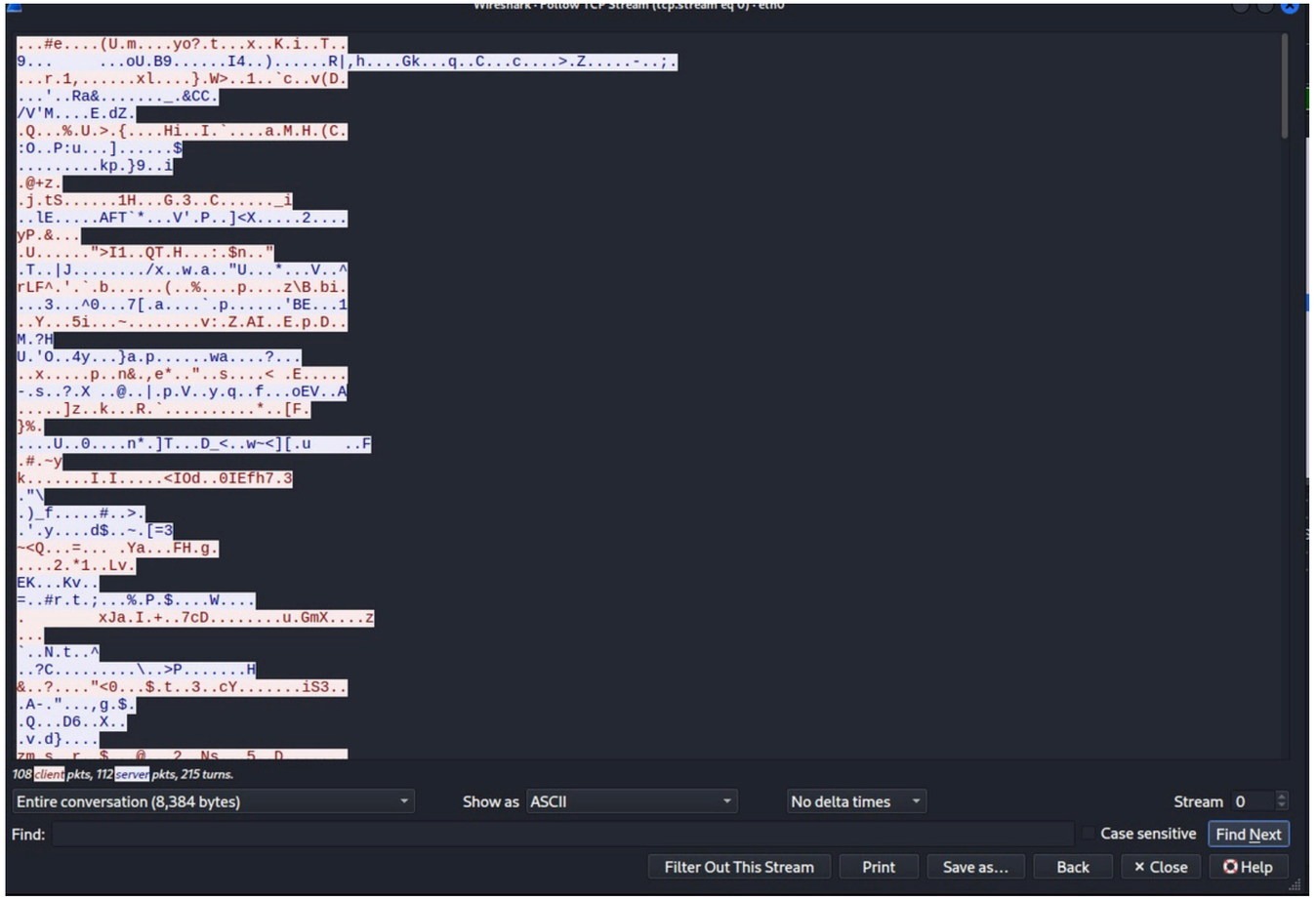
Getting access of Victim machine



Observing ssh through wireshark



All data in encrypted form



SSH and Telnet Explained

- SSH hides your messages so no one can see them.
- Telnet shows your messages to anyone watching.
- SSH asks for safe keys or passwords before opening a connection.
- Telnet only uses a simple password that can be stolen easily.
- SSH works well on modern systems and keeps things private.
- Telnet is only used for simple testing because it is not safe.

Secure SSH vs Open Telnet

- SSH keeps your login private using strong protection.
- Telnet sends your login openly across the network.
- SSH stops others from changing your messages.
- Telnet lets attackers read or change your messages.
- SSH supports safe remote work.
- Telnet is unsafe for real-world use.

SSH vs Telnet

- SSH keeps your data safe by locking everything with strong protection.
- Telnet sends everything in plain form, so anyone can read it.
- SSH checks who you are before letting you in, which makes it safer.
- Telnet has no safety checks, so it’s easy to break into.
- SSH is used today for servers because it protects both login and data.
- Telnet is mostly gone now because it cannot keep data private.