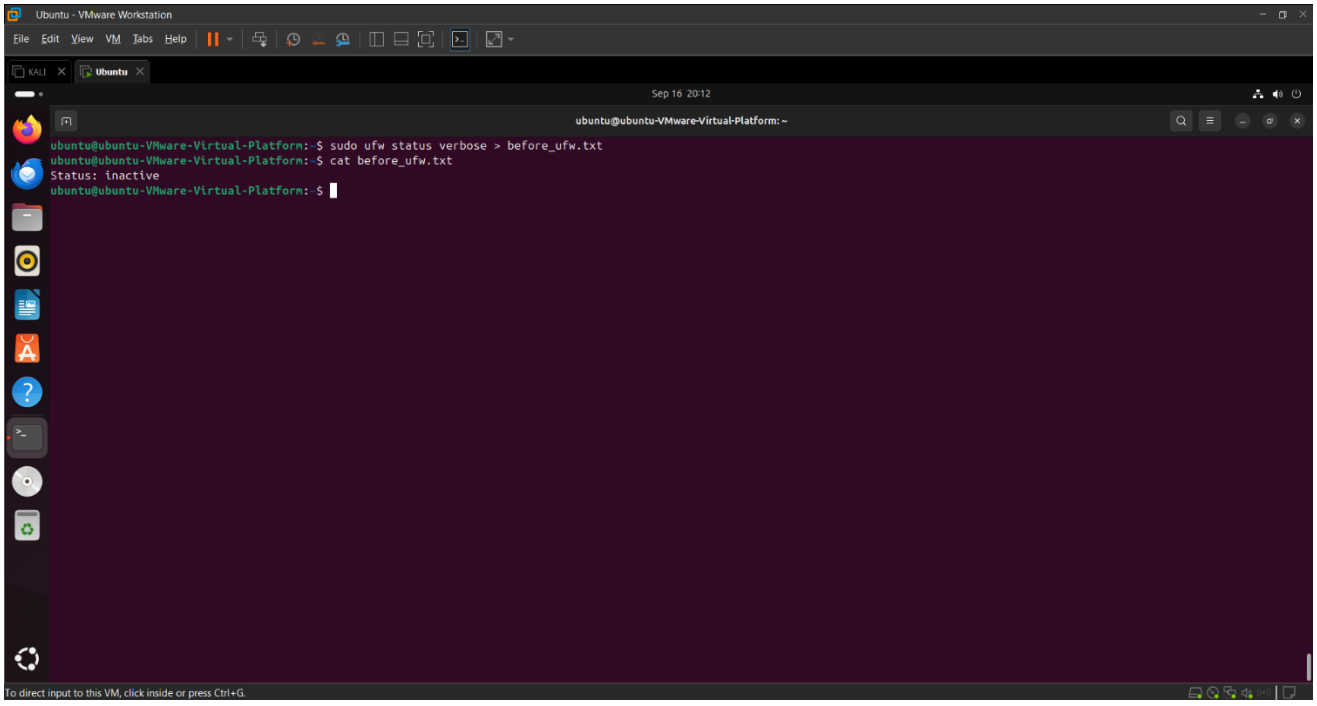
Linux Server Hardening

Objective

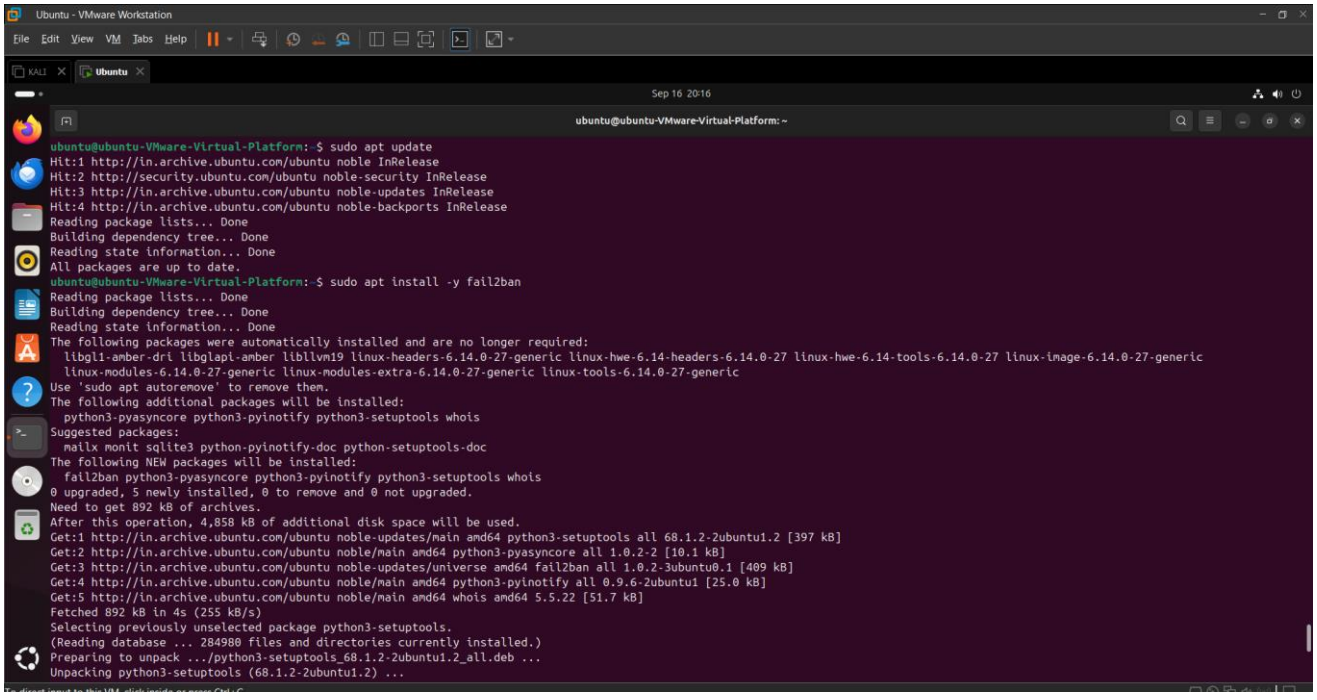
Capture and analyze live network traffic to identify credentials or suspicious activity. Apply Linux server hardening techniques using Ubuntu, UFW, Fail2ban, and SSH.

Before State

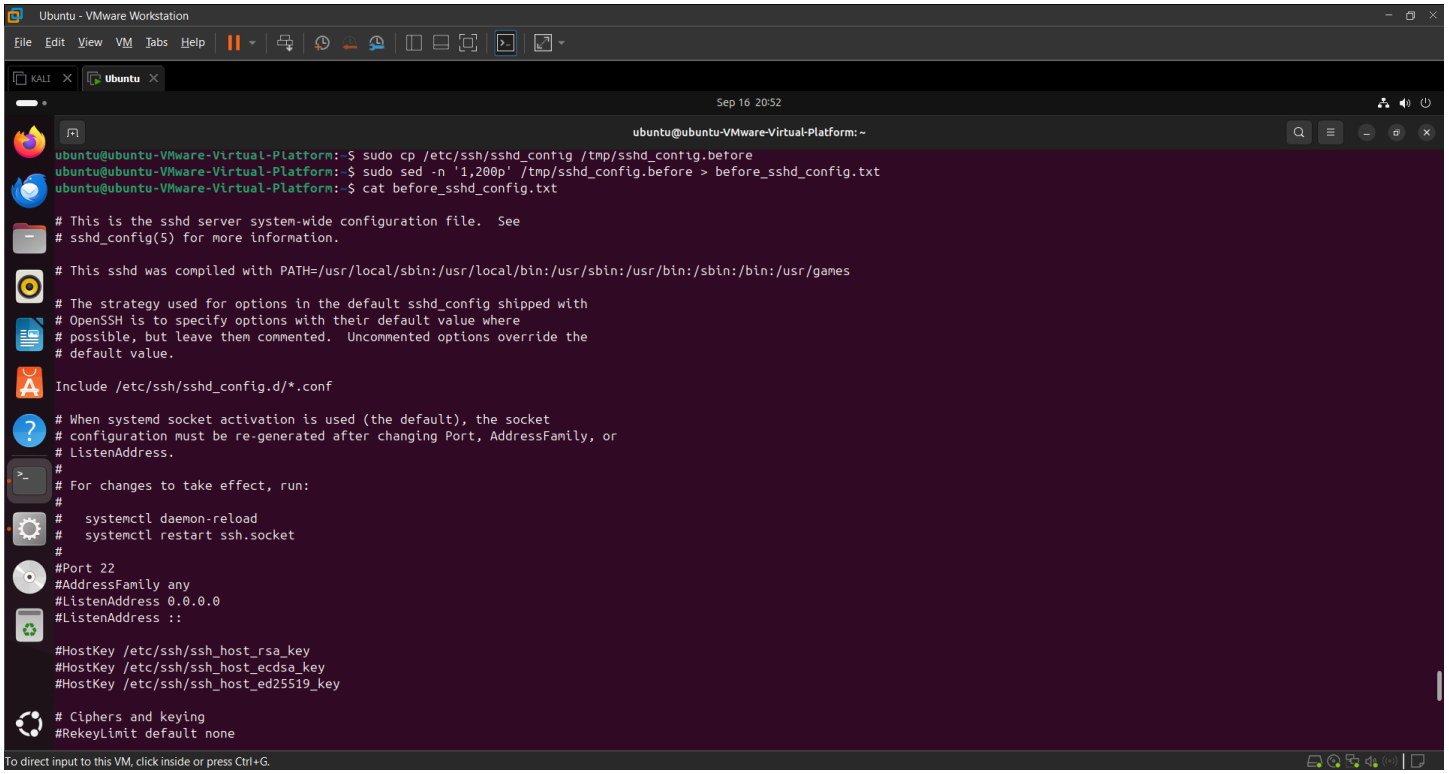
- Firewall (UFW): Inactive



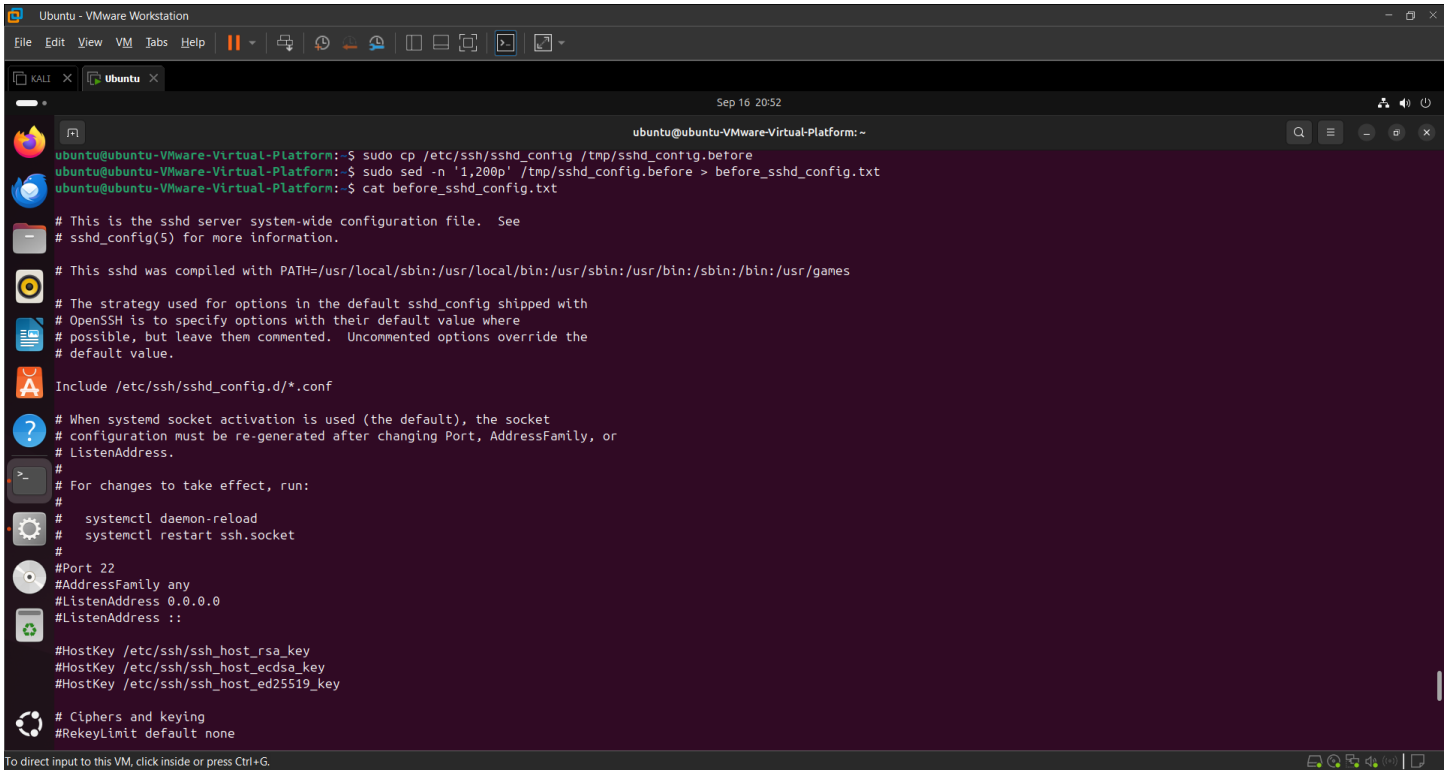
- Fail2ban: Not installed ("Unit fail2ban.service could not be found")



- SSH configuration: Root login allowed, password authentication enabled



- Open Ports: 22 (SSH), 80 (HTTP), 3306 (MySQL – exposed)



Legal / Ethical reminder

Only capture and analyze traffic on systems/networks you own or for which you have written permission. Finding credentials on other people's networks without permission is illegal and unethical.

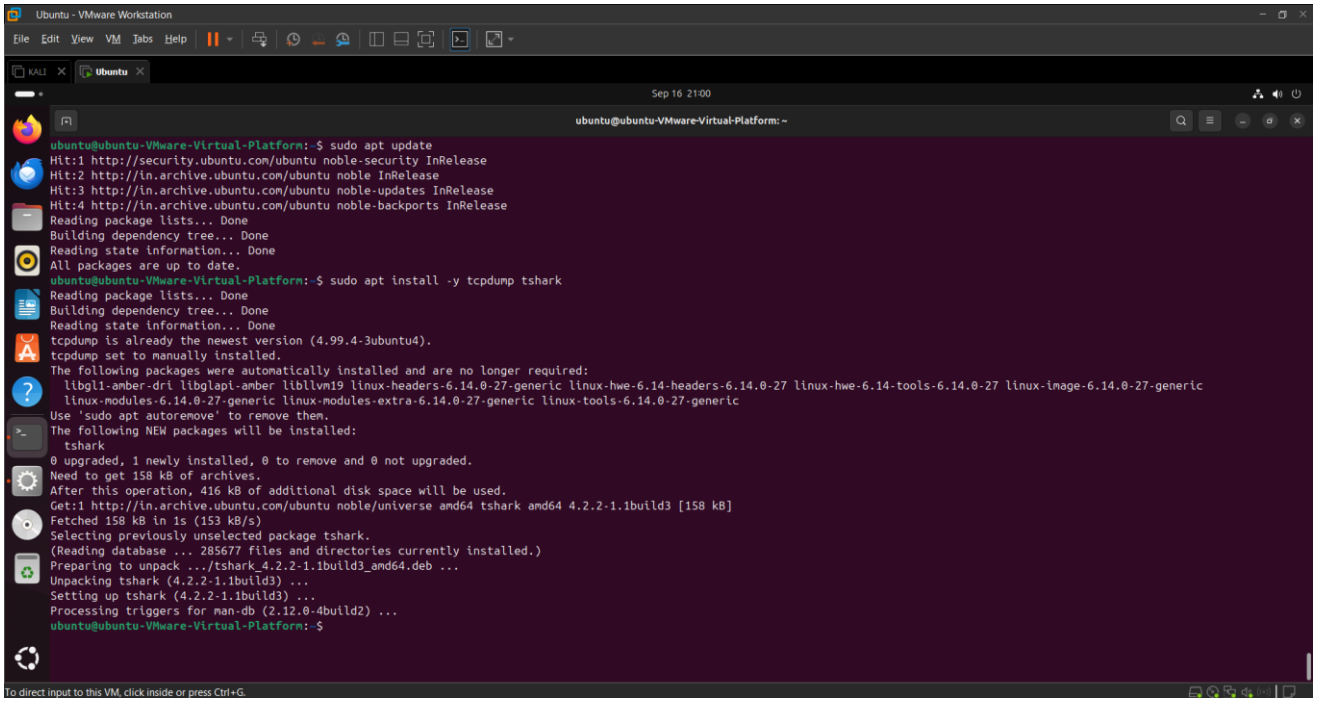
Live capture — safe, targeted capturing:

Important: Capturing everything can expose sensitive data. Only capture what you are authorized to. Use filters to limit scope to suspicious protocols or hosts.

Update and install the tcpdump and tshark by running below commands

-sudo apt update

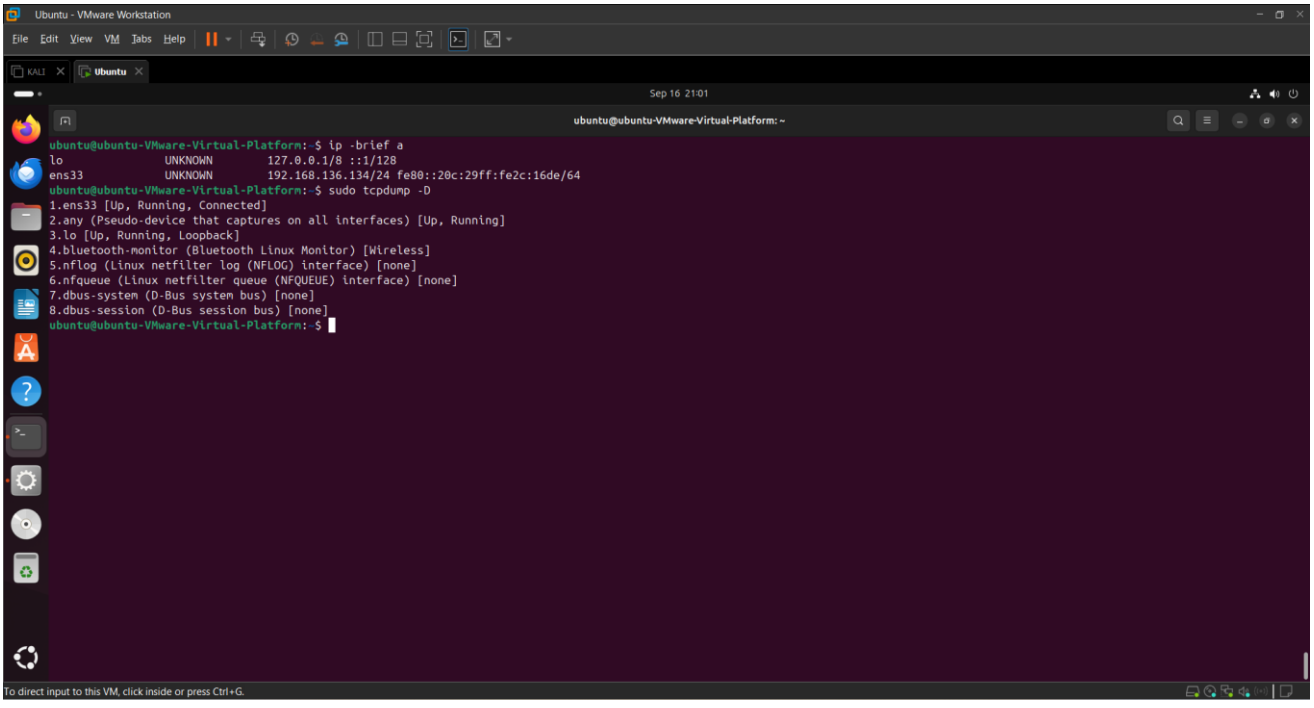
-sudo apt install -y tcpdump tshark



To List network interfaces commands used are :

-ip -brief a

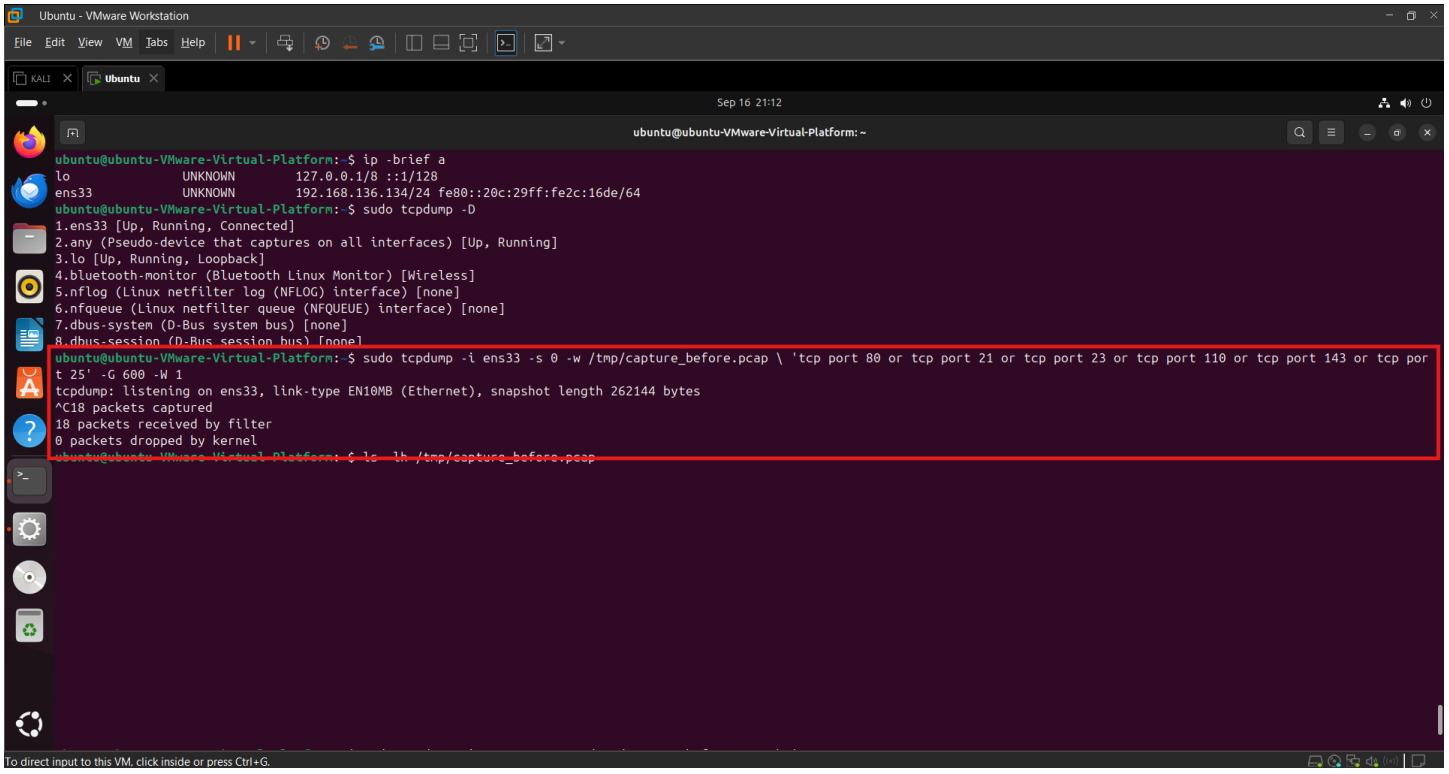
-sudo tcpdump -D



Capture common cleartext authentication protocols (FTP, Telnet, POP3, IMAP, SMTP) for 10 minutes by using below command:

replace ens33 with your interface from tcpdump -D

-sudo tcpdump -i ens33 -s 0 -w /tmp/capture_before.pcap \ 'tcp port 80 or tcp port 21 or tcp port 23 or tcp port 110 or tcp port 143 or tcp port 25' -G 600 -W 1

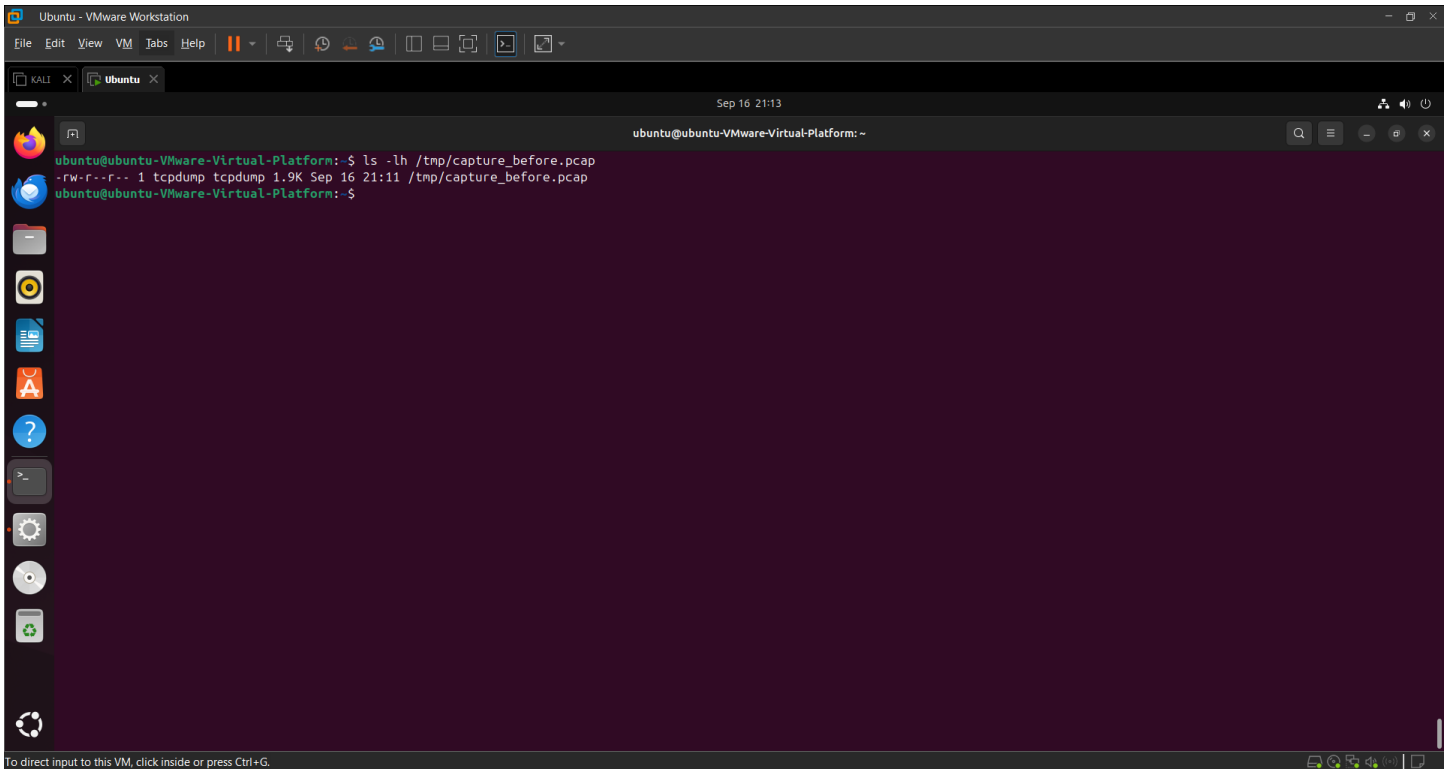


Notes:

- -s 0 captures full packet.
- -w writes pcap for later analysis.
- Use -G with -W to rotate files by seconds if long run needed.
- It will listen for 10 minutes (-G 600).
- It will write all captured packets that match your filter into /tmp/capture_before.pcap
- If there’s no traffic on those ports (80, 21, 23, 110, 143, 25), the file might remain very small or even empty.

Verify capture file by using this below command:

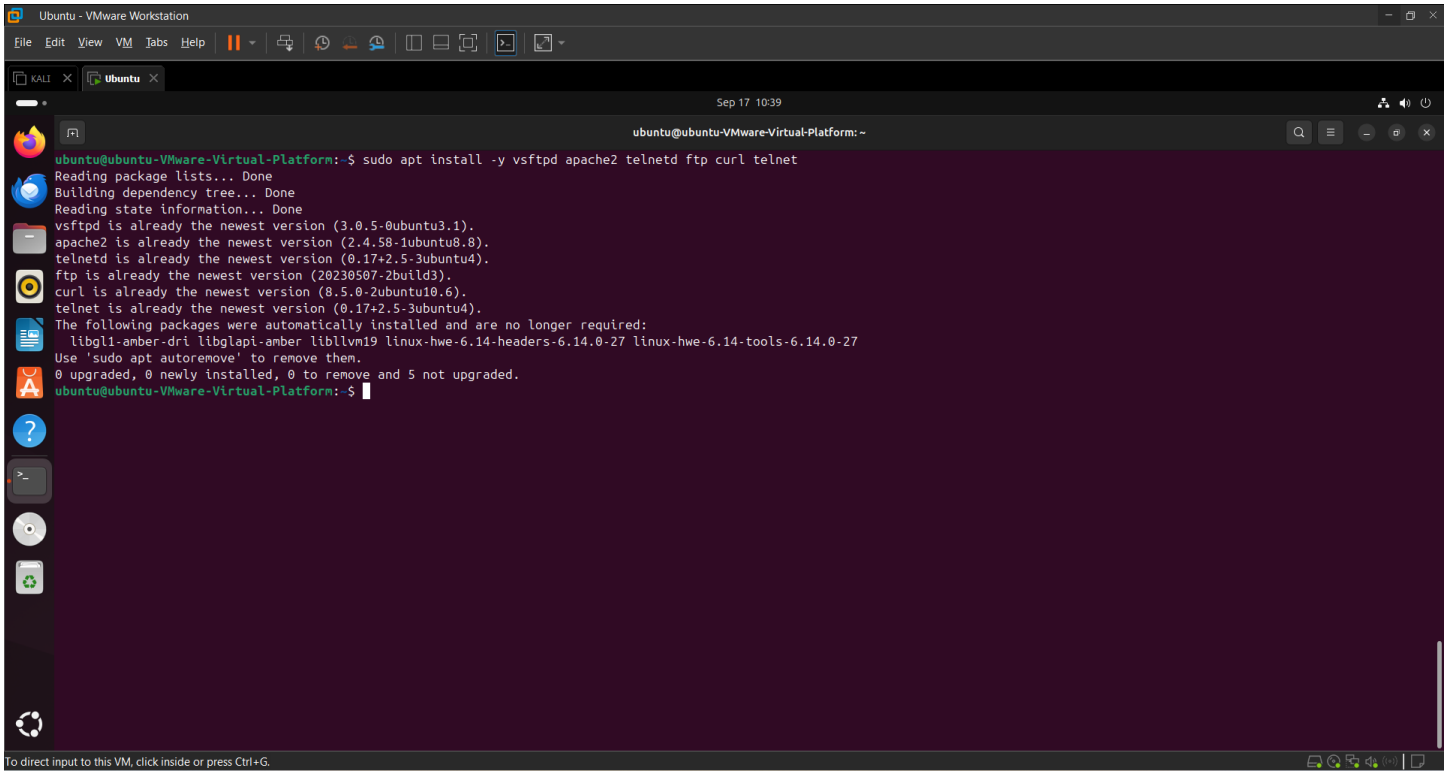
ls -lh /tmp/capture_before.pcap



Install basic servers & clients by below commands

-sudo apt update

-sudo apt install -y vsftpd apache2 telnetd ftp curl telnet

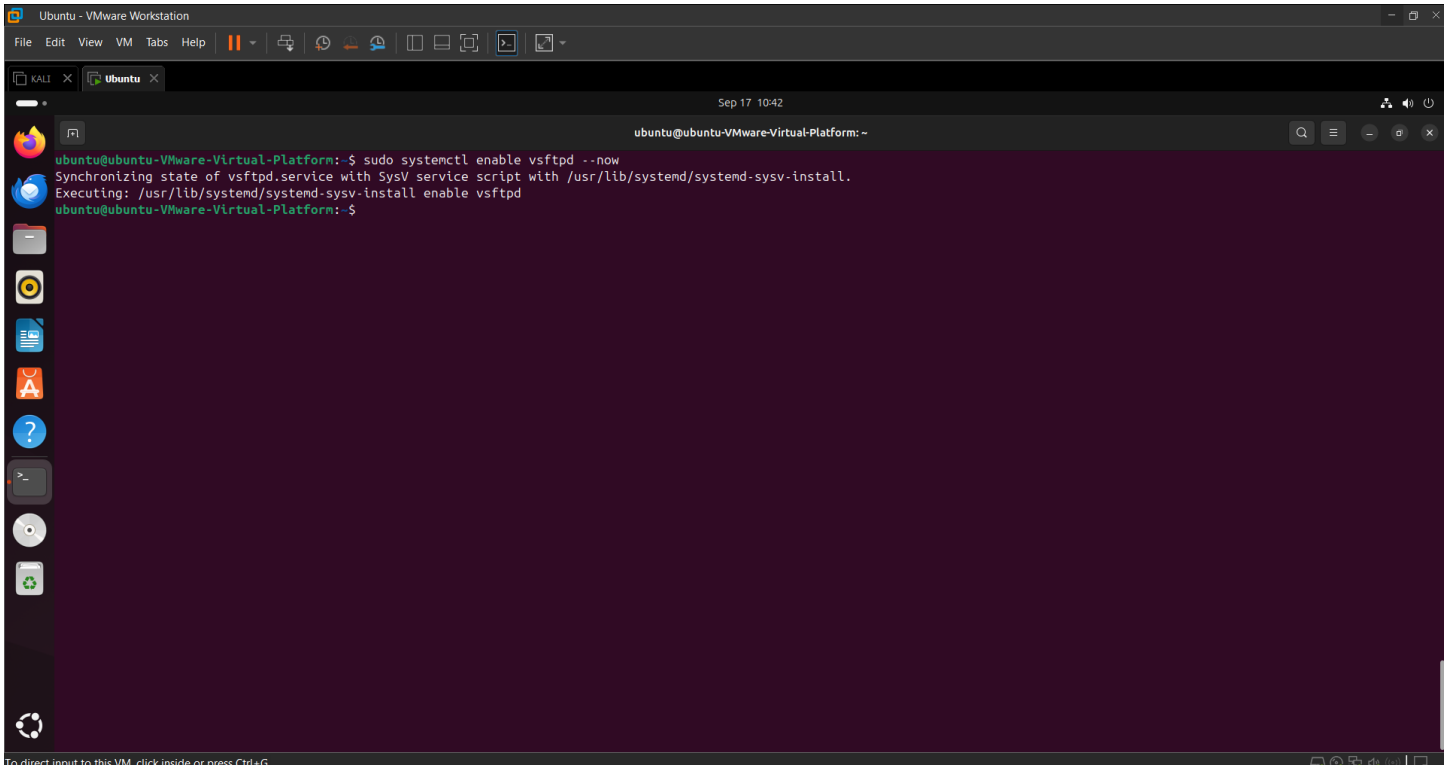


This gives:

- **vsftpd** → FTP server
- **apache2** → HTTP server
- **telnetd** → Telnet server
- **ftp, curl, telnet** → clients to test

now start FTP (vsftpd) service by default, it allows anonymous or local logins. You can test with your Ubuntu username/password. Use this command:

-sudo systemctl enable vsftpd --now



Start Telnet service

-sudo systemctl enable inetd –now

-sudo systemctl start telnetd

Start Apache (HTTP)

-sudo systemctl enable apache2 –now

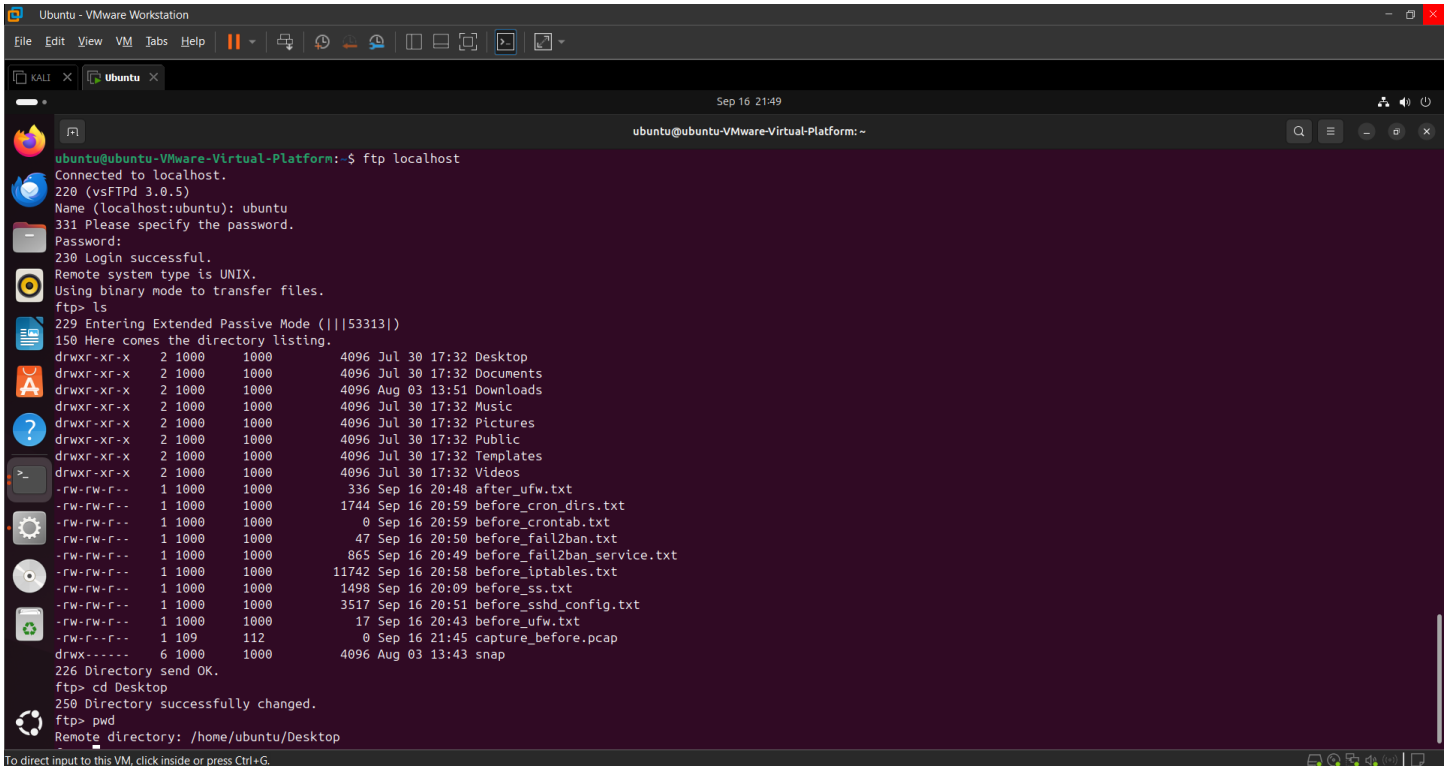
Generate traffic with credentials

FTP test From the same VM or another machine:

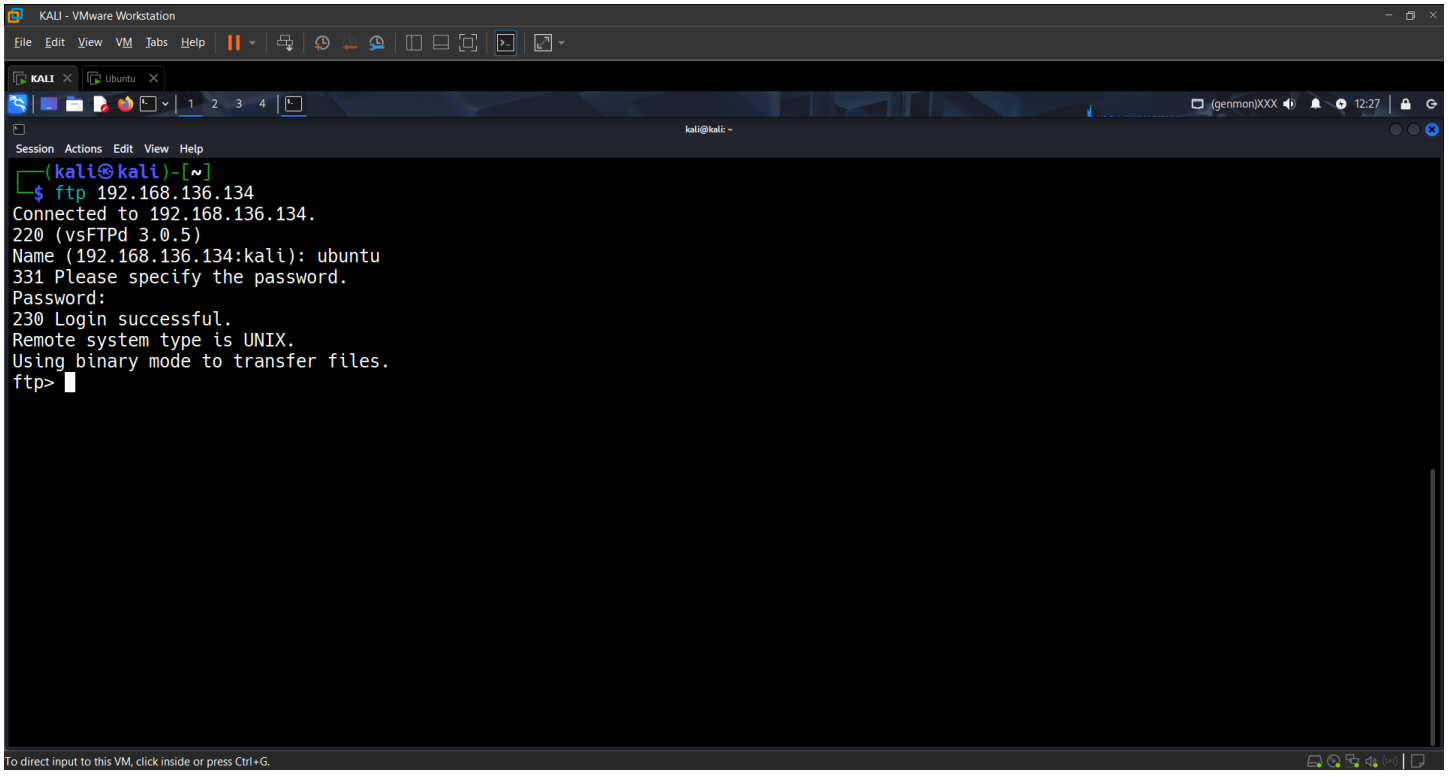
-ftp (IP of ubuntu)

Enter username (your ubuntu username)

Enter password (your ubuntu password)



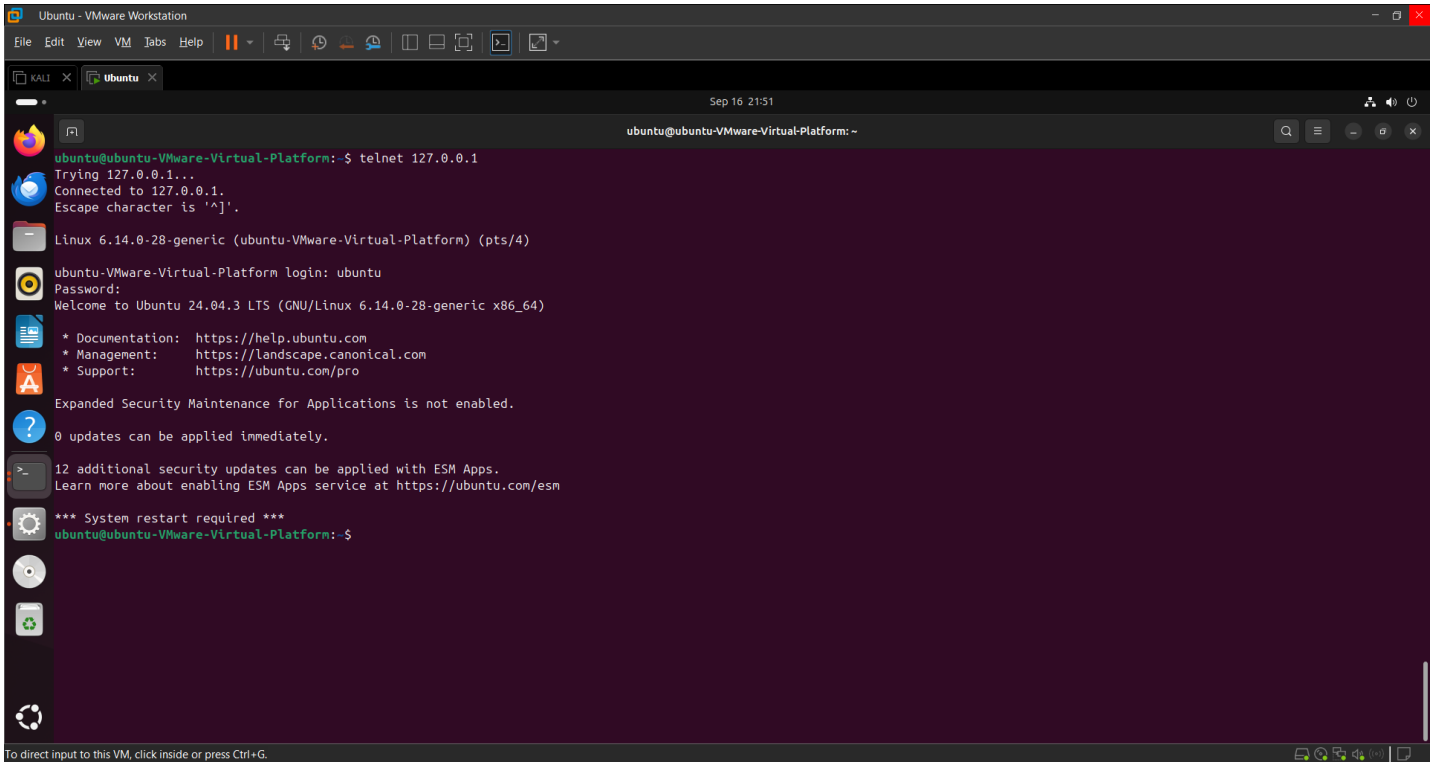
Other machine – Kali Linux



Telnet test From the same VM

-telnet 127.0.0.1

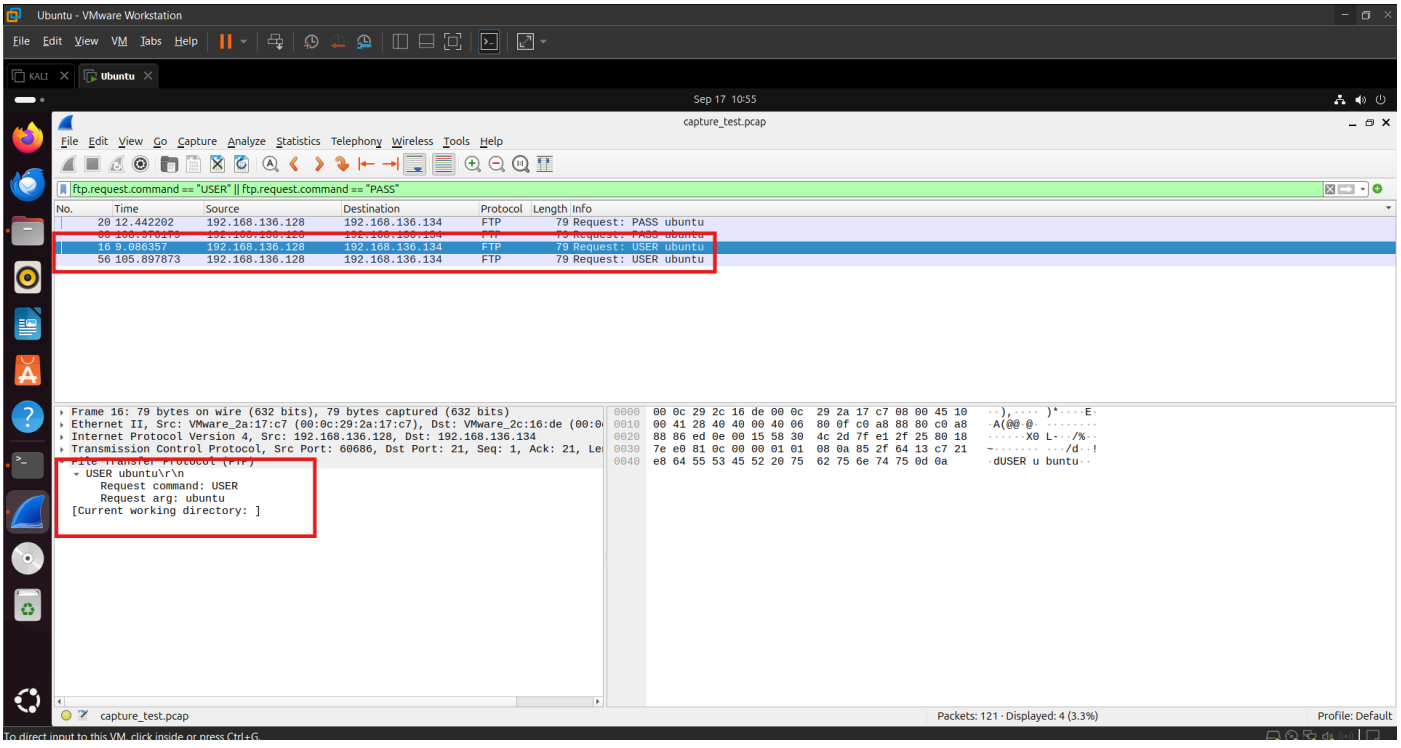
Login with your ubuntu username & password

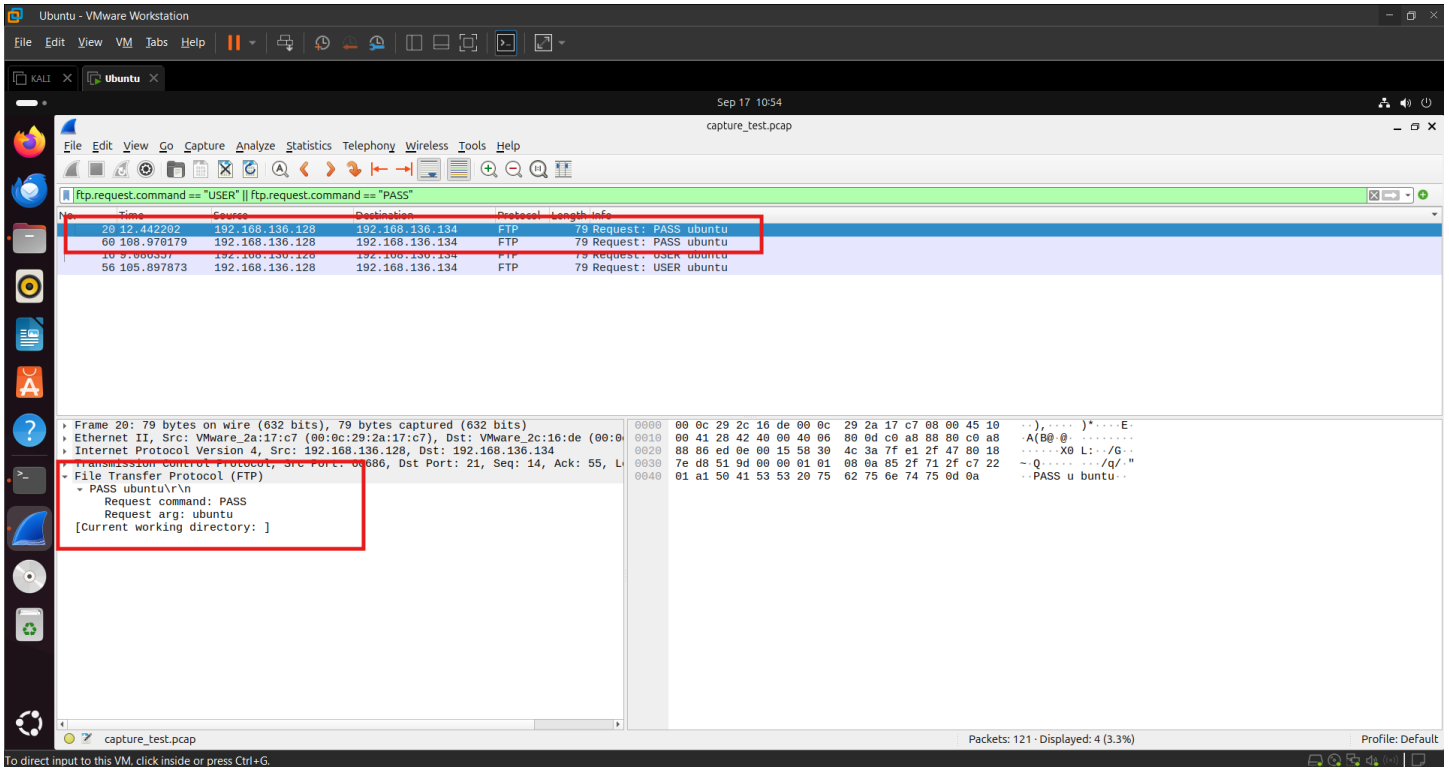


Analyse the Captured traffic in Wireshark

Open /tmp/capture_test.pcap in Wireshark and use filters:

- **FTP**:
  ftp.request.command == "USER" || ftp.request.command == "PASS"





Here we can see the user name and password in plain text.

- **Telnet**:
  telnet
  → Right-click a packet → **Follow → TCP Stream**. You'll see your username/password typed.

- **HTTP**:
  http.authorization
  → Look for Authorization: Basic ... → Base64 decode to get username:password.

Deliverables for report

- **Before**: Show tcpdump running (listening on ens33…)
- **After**: Show .pcap analysis in Wireshark with highlighted credentials
- **Commands used** (services installed, tcpdump, test logins)