Sri Lanka Institute of Information Technology

Specializing in Cyber Security

IE2012 - Systems and Network Programming

SNP Assignment

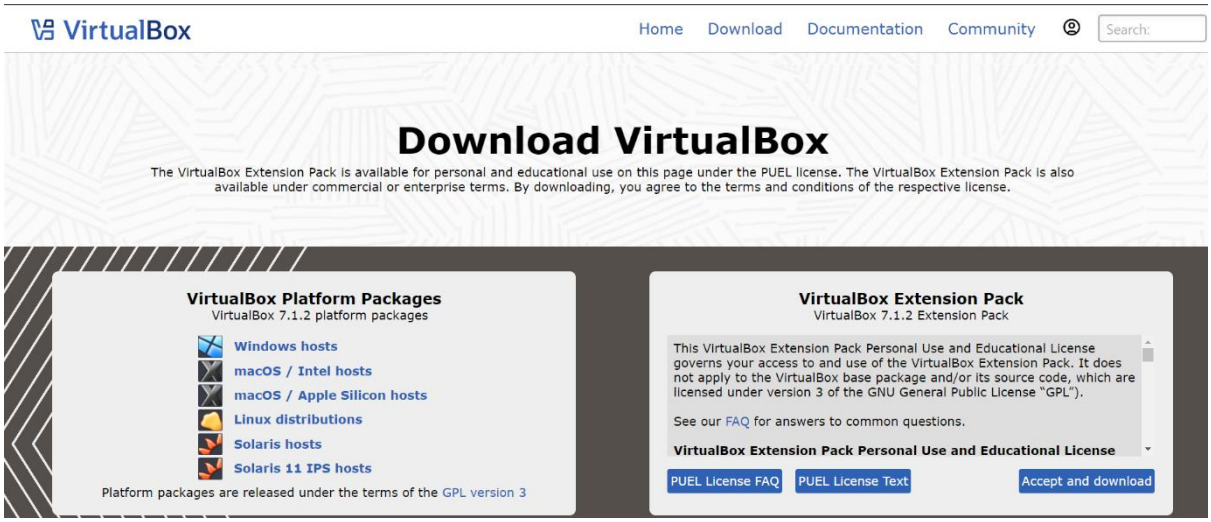
IT23238794 – DISSANAYAKE Y Y

Table of Contents.

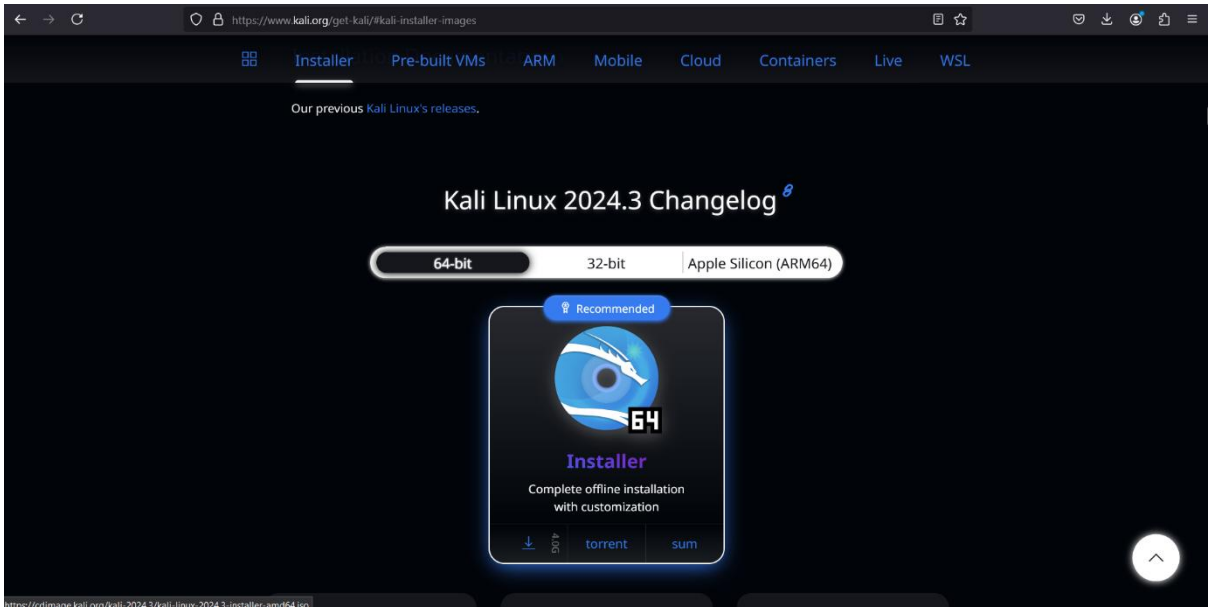
1. Basics of Linux Environment.

1.1 Virtual Machine Setup

Install a virtual emulator on your Mac, Windows, or Linux computer first. I like Oracle VirtualBox. Run the setup file and carry out the Windows system installation after the download is finished.

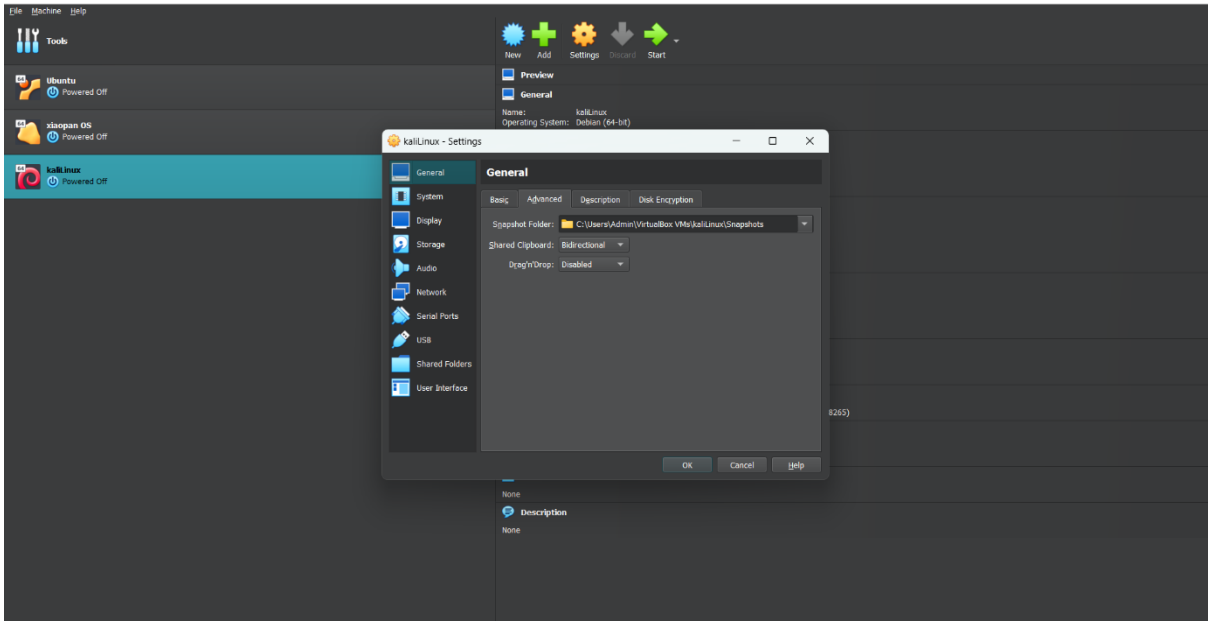


Go to the official website and download the Kali operating system.

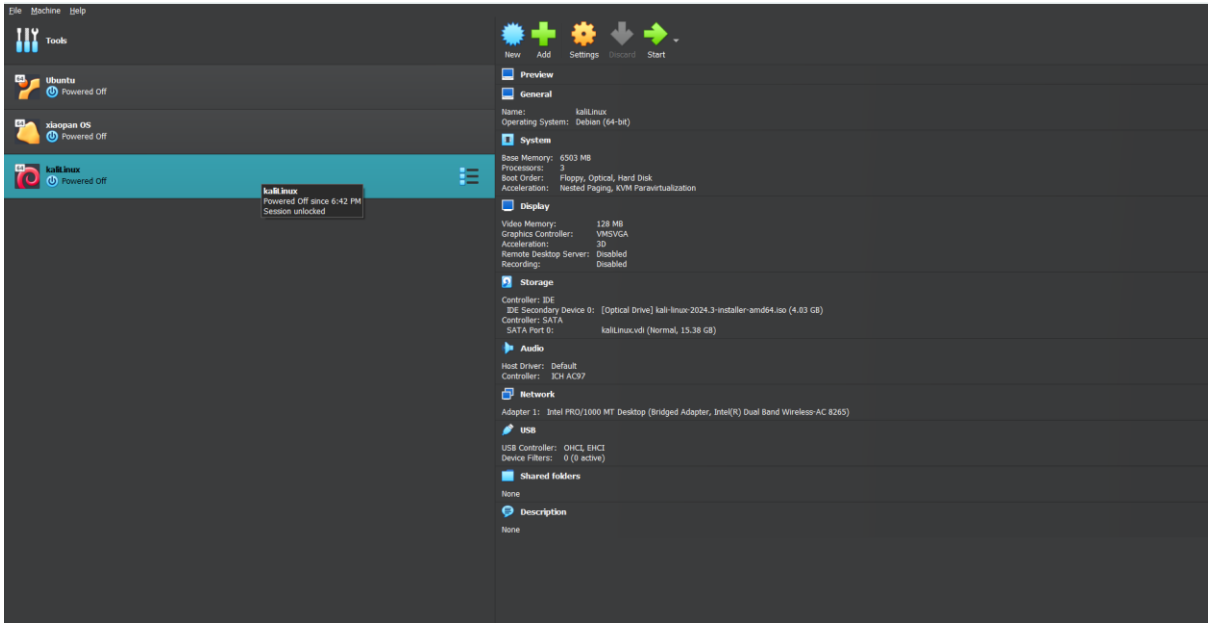


Install and configure VirtualBox. Click the New button and fill in the Create Virtual Machine. Then click Next button and type username and password. Create Virtual Machine.

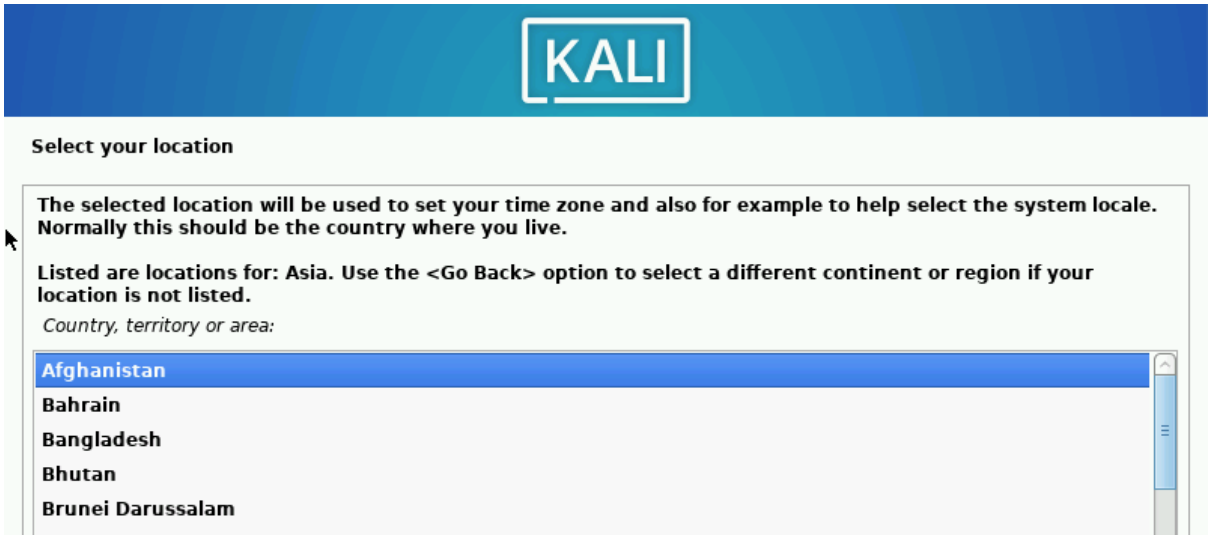
Then change settings in virtual box.



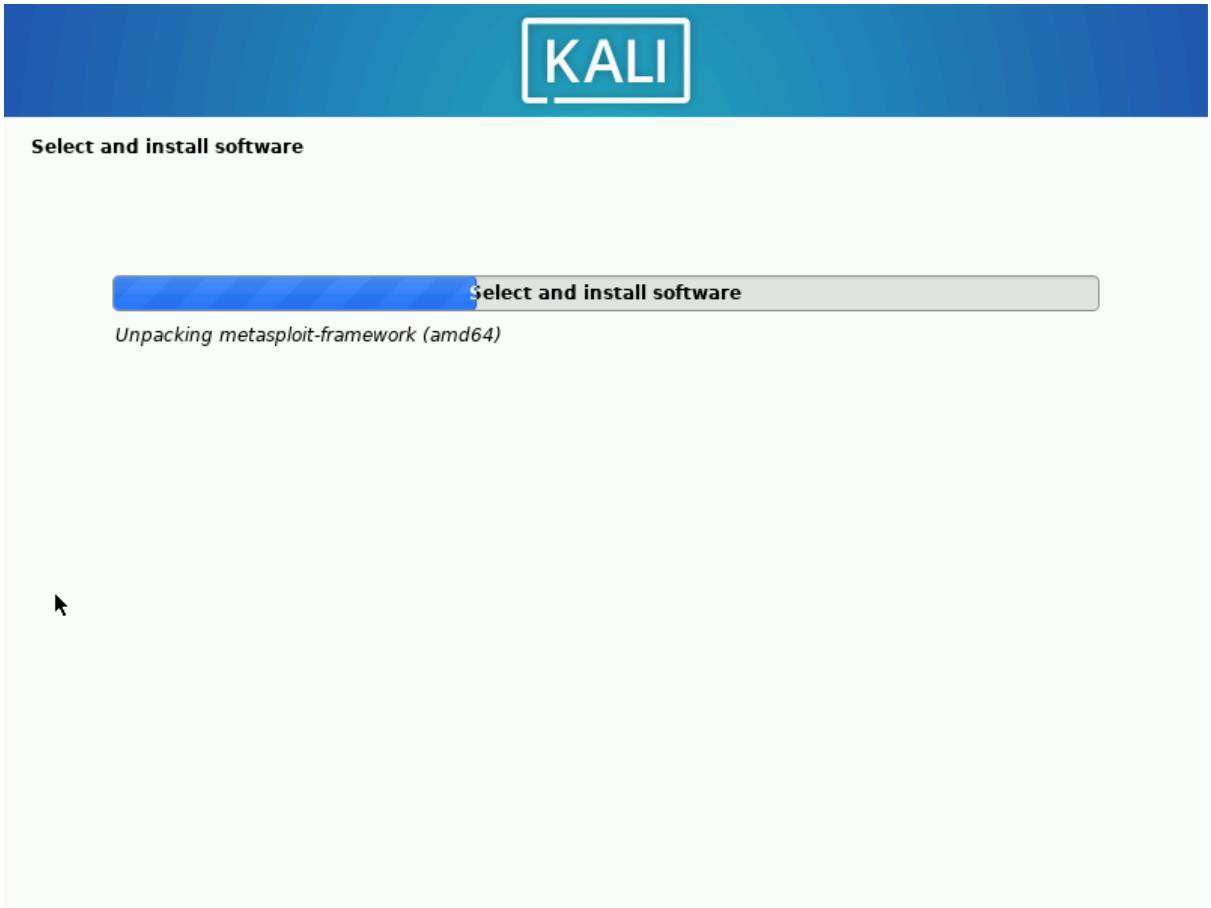
successfully installed Kali as a virtual machine.



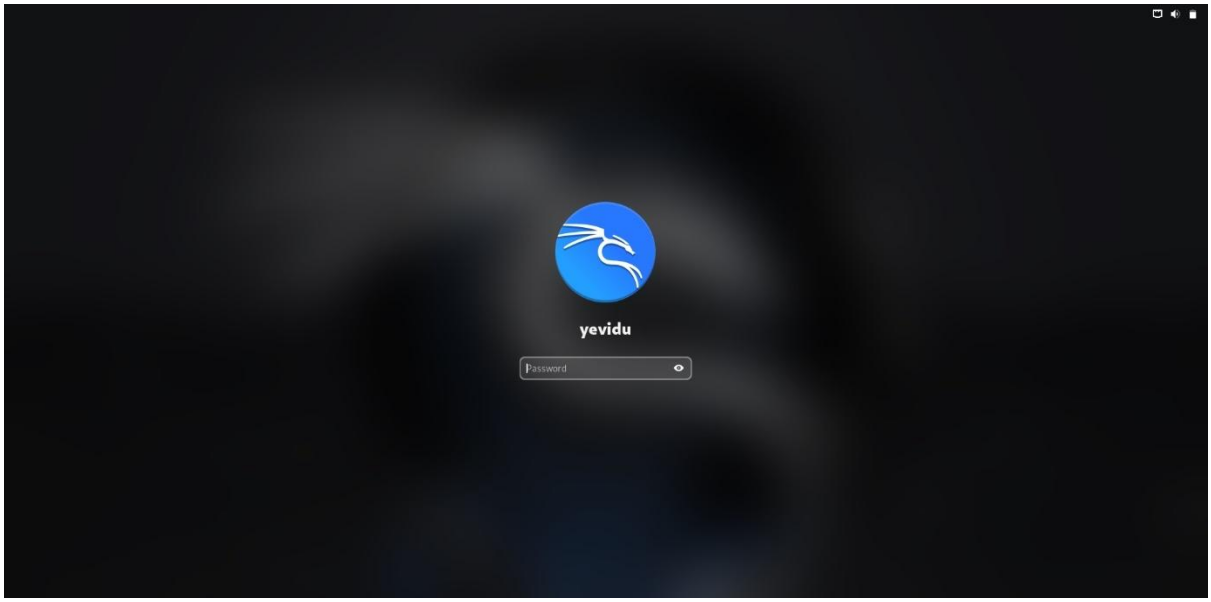
Then change the criteria after power on the kali Linux.



Then waiting for installing process.



Finally installing completed.

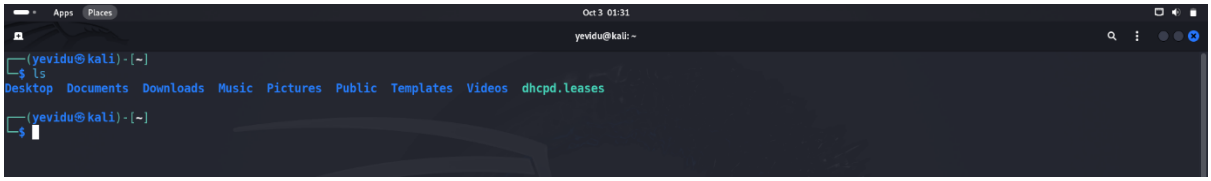


After type password navigate to kali interface.

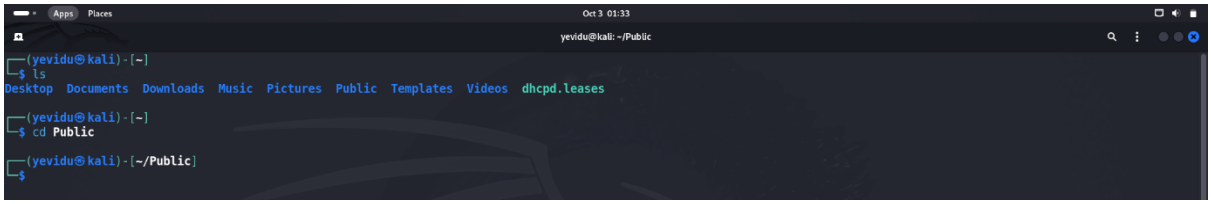
1.2 Command Line Introduction.

1.2.1 Basic Navigation Commands

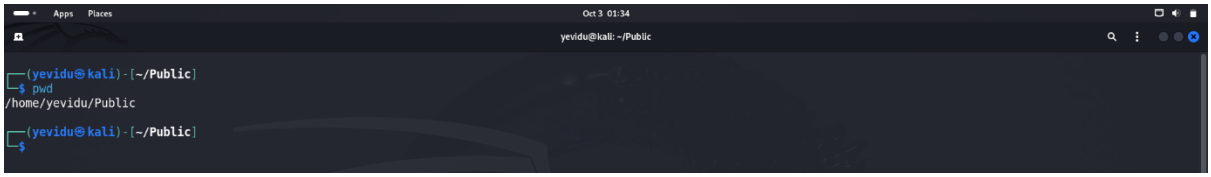
ls : Lists files and directories in the current or specified directory.



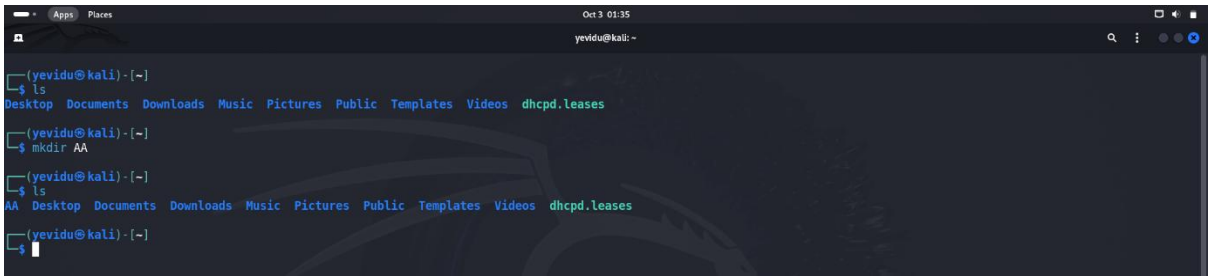
cd : Changes the active working directory.



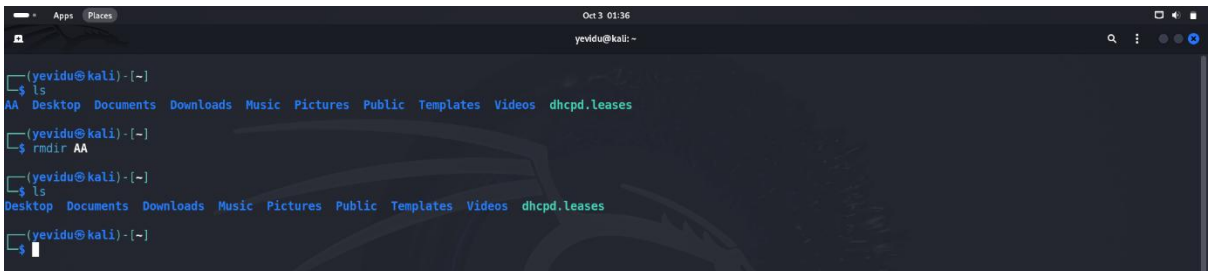
pwd : Displays the full path of the current directory.



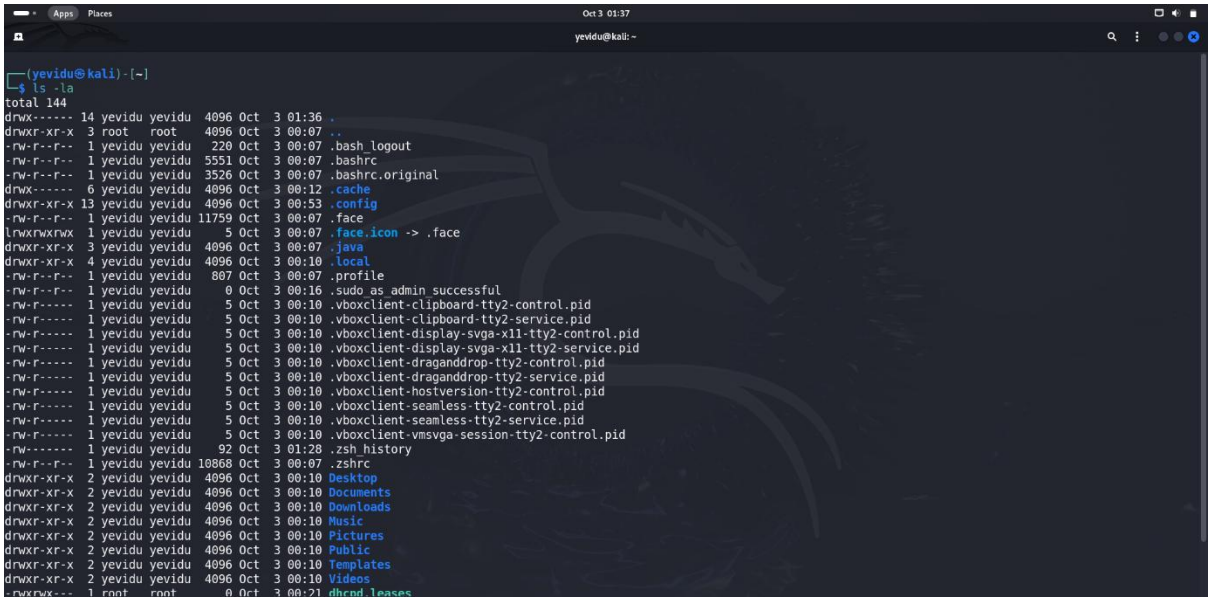
mkdir : Creates a new directory.



rmdir : Removes an empty directory.

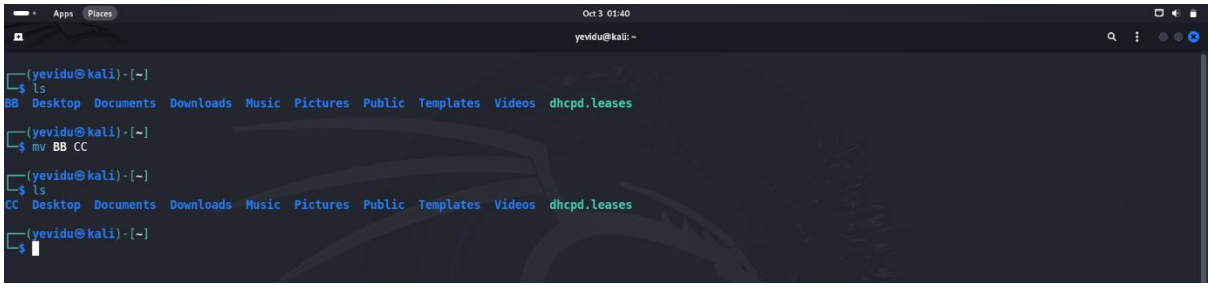


ls -la : provides detailed information about each file or directory.

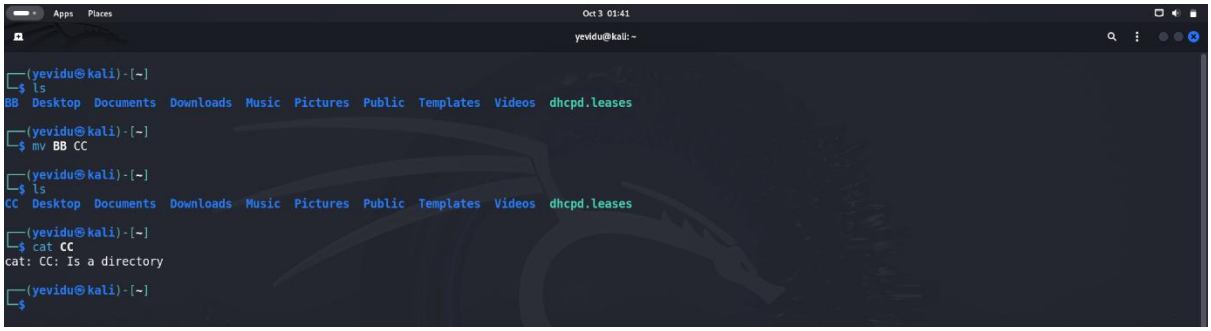


1.2.2 File Manipulation Commands

mv : Moves or renames files or directories.

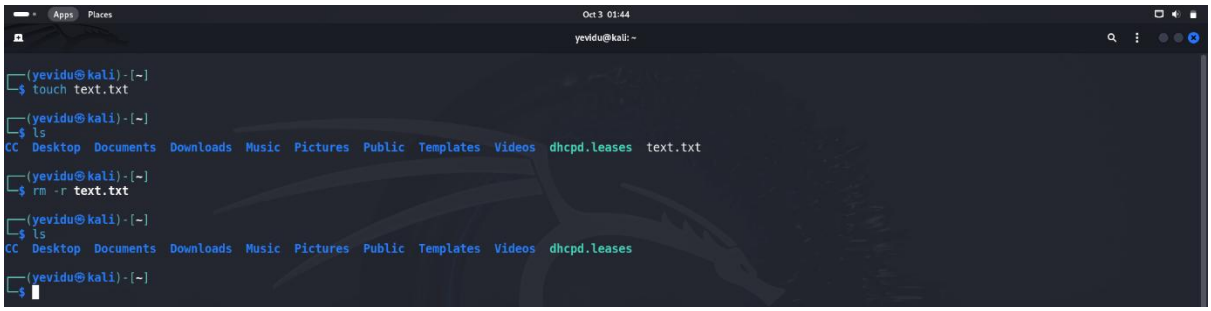


cat : Displays the content of a file.



touch : Creates a new, empty file.

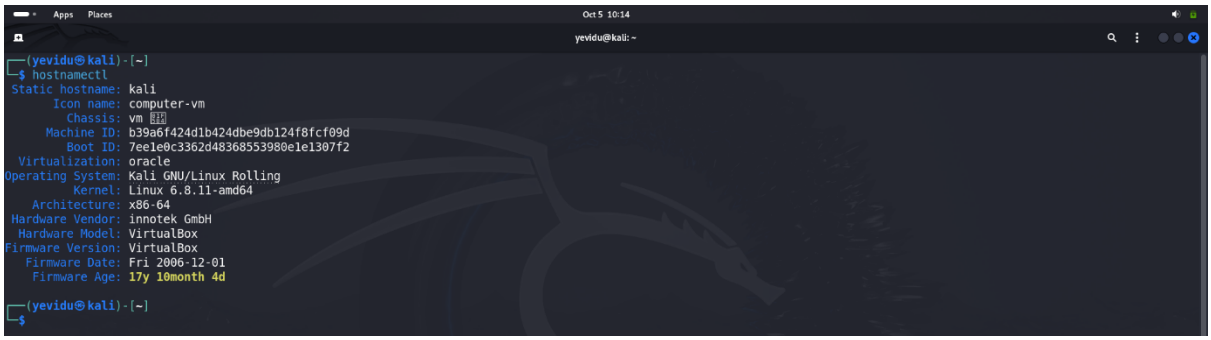
rm -r : Removes files or directories.



1.3 System Information and User Management.

1.3.1 System Information Commands.

hostnamectl : Display system hostname and OS details with kernel version and architecture.

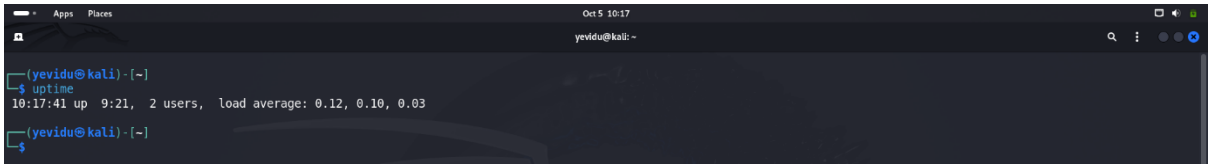


lscpu : Provide details about CPU architecture.

df -h : Show disk usage.

uname -a : Display kernel version and other information.

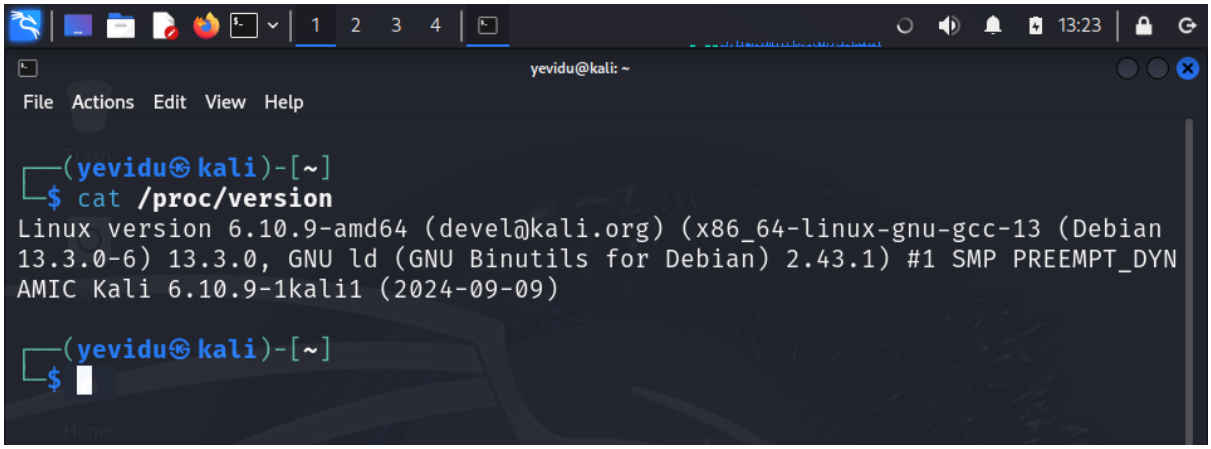
uptime : Show how long the system has been running with load average.



ip -a : Show detailed network interface information including IP addresses.

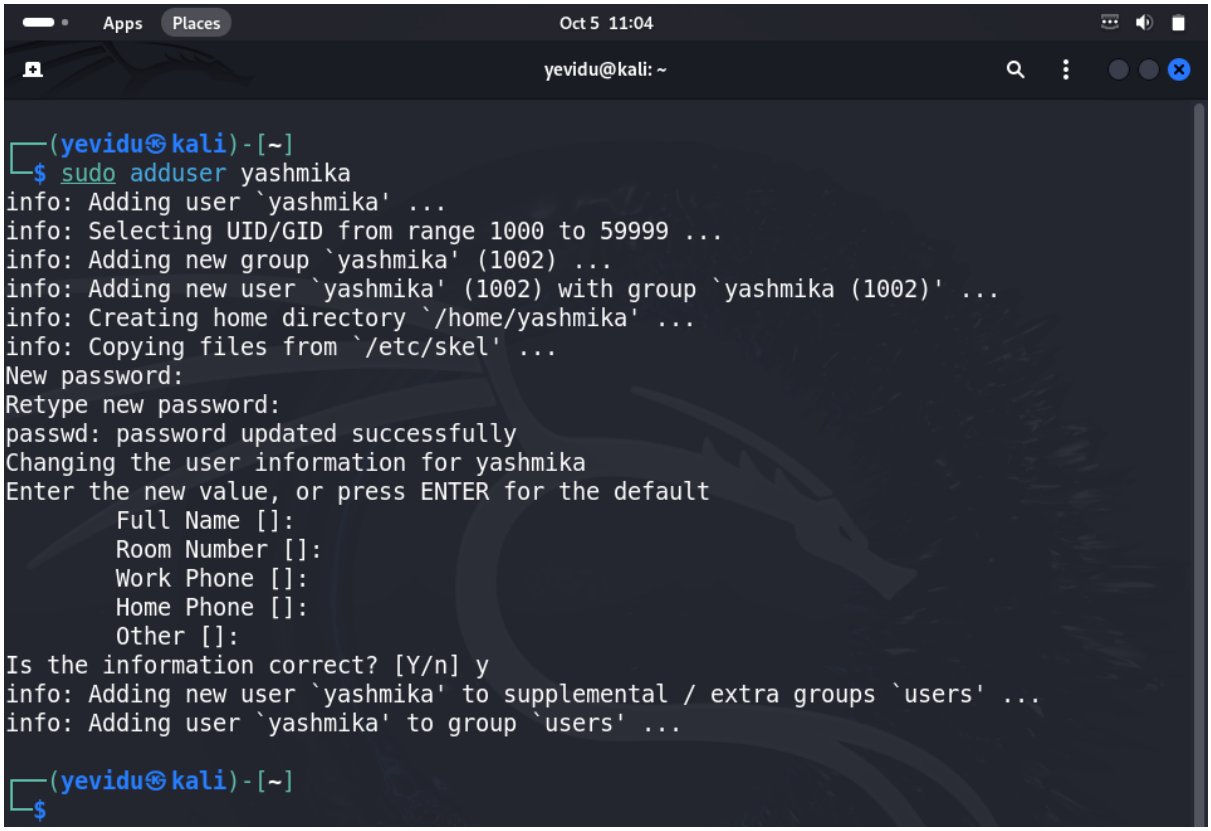
free -m : Display total memory, used memory, free memory, shared memory, buffers, and cache.

cat /proc/version : outputs detailed information about the Linux kernel version and system build.

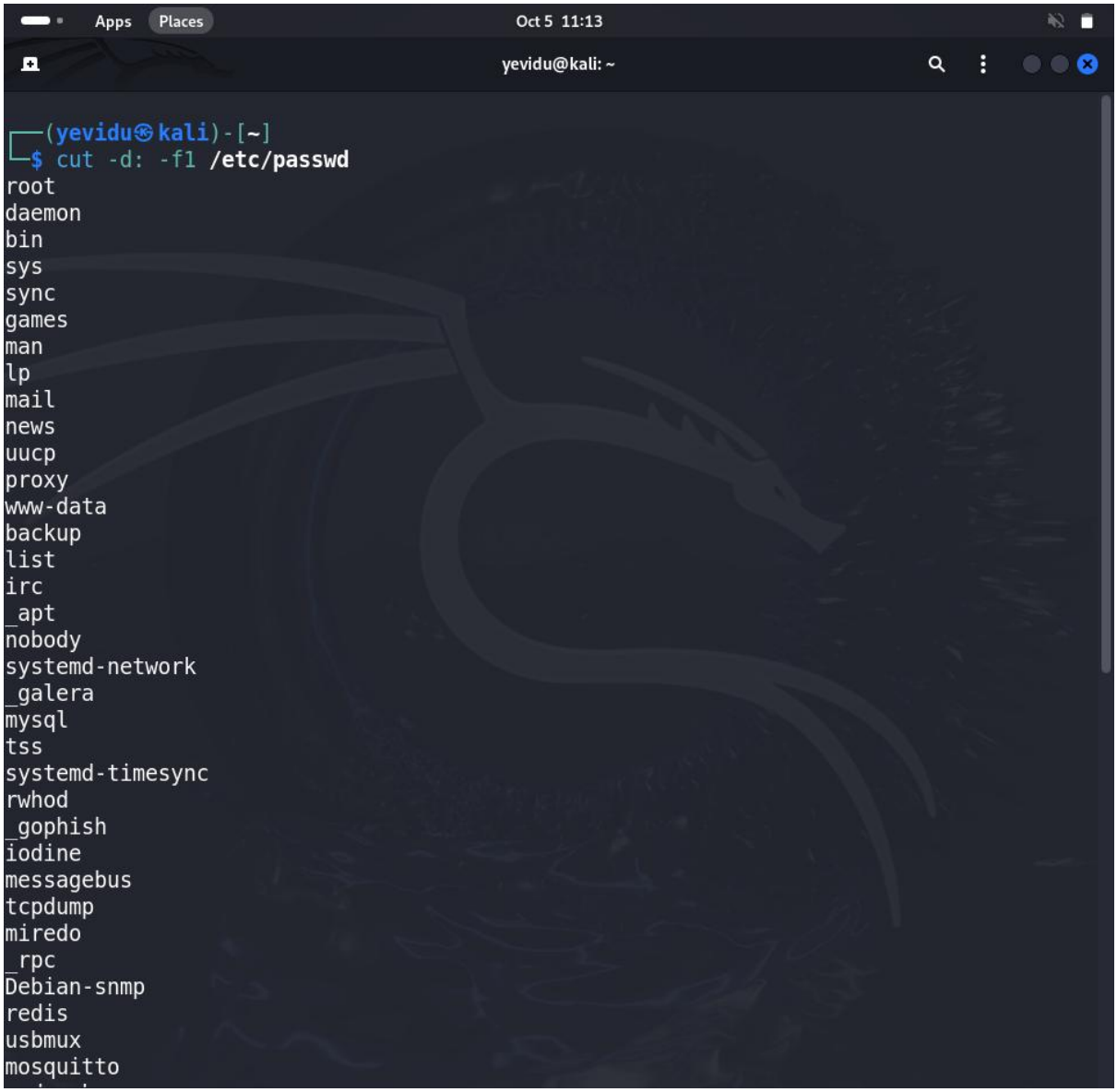


1.3.2 User Management Commands.

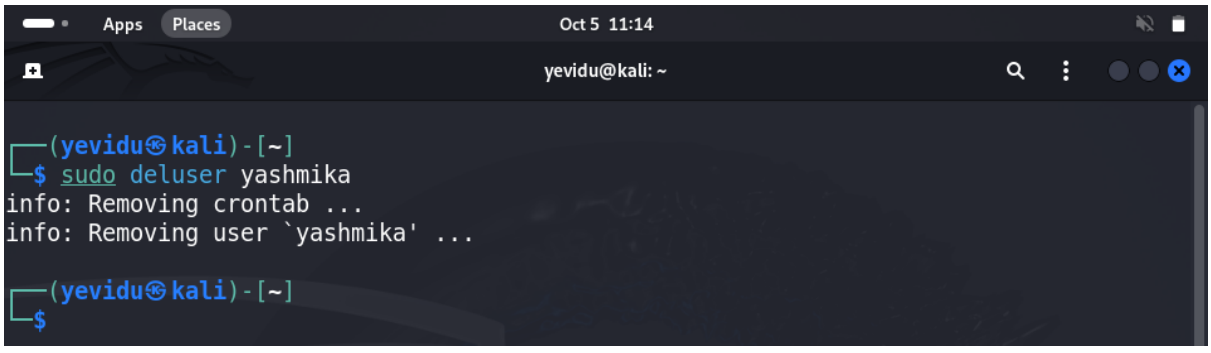
sudo adduser username : Add new user.



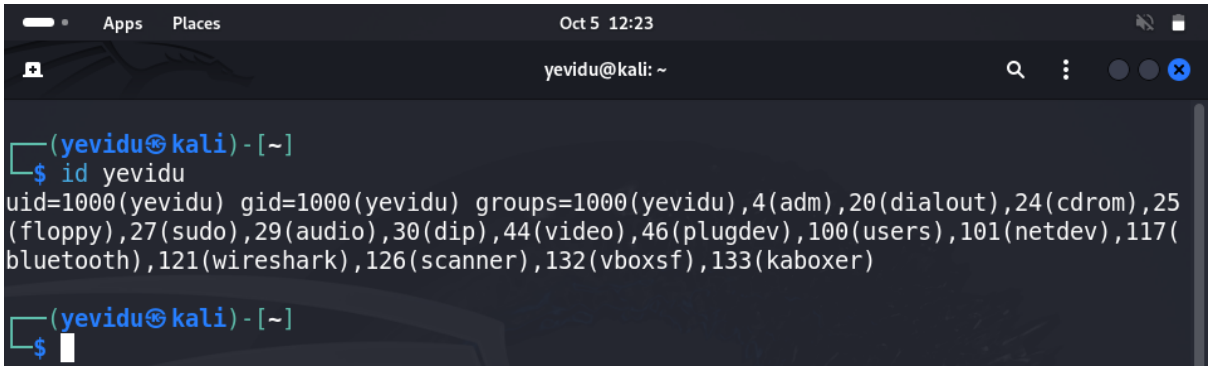
cut -d: -f1 /etc/passwd : View list of all users.



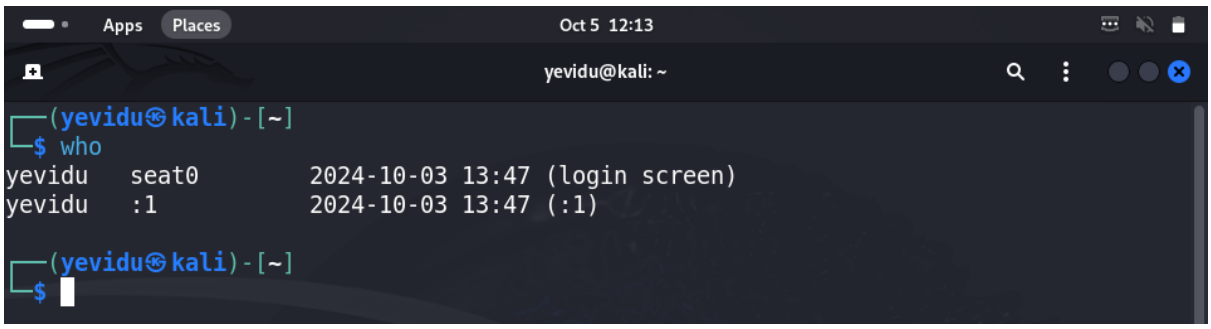
sudo deluser username : Delete a user.



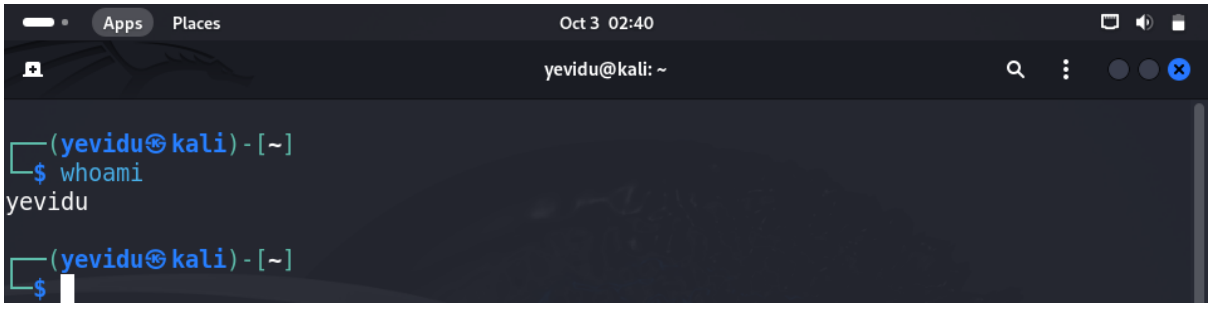
id : view a user's details.



who : Display users logged in to the system.



whoami : Display current user log in to the system.

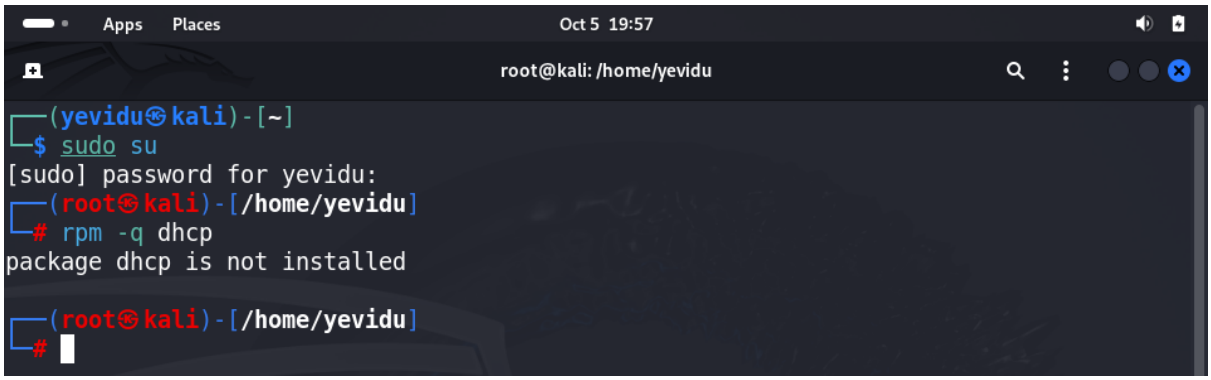


2. DHCP, DNS and NTP Services.

2.1 DHCP (Domain Host Configuration Management).

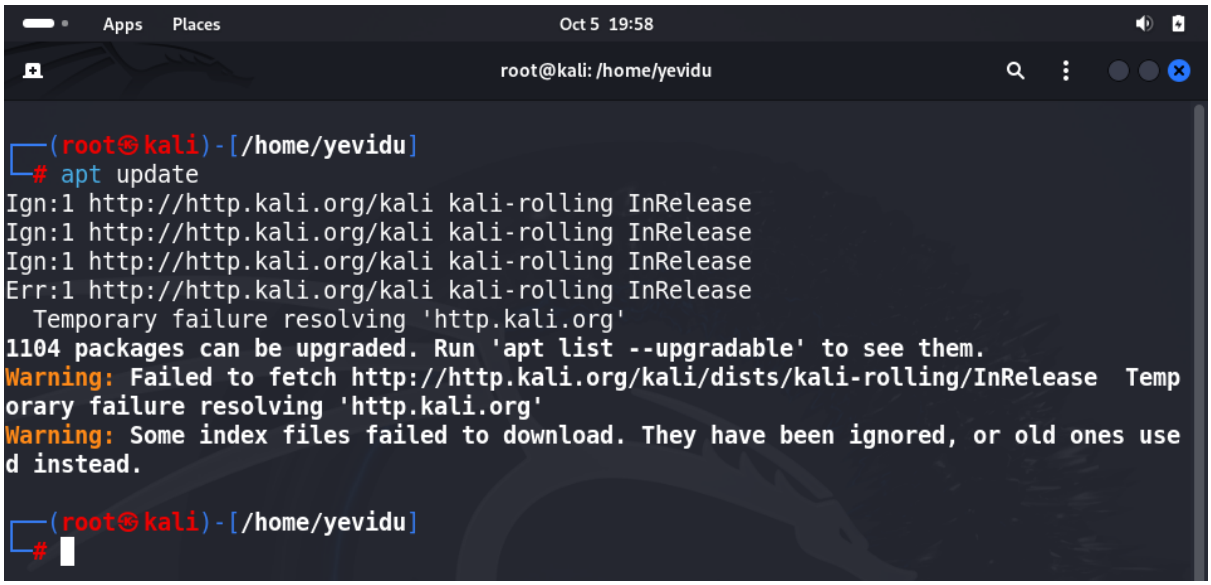
First check DHCP is already installed in the Linux.

Command **rpm -q dhcp**.



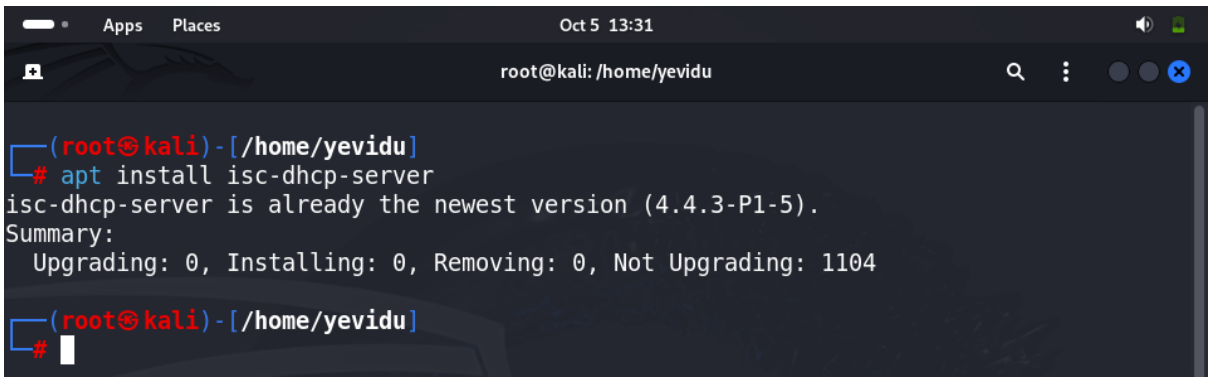
It does not install. So, we update the package.

Command **apt update**.



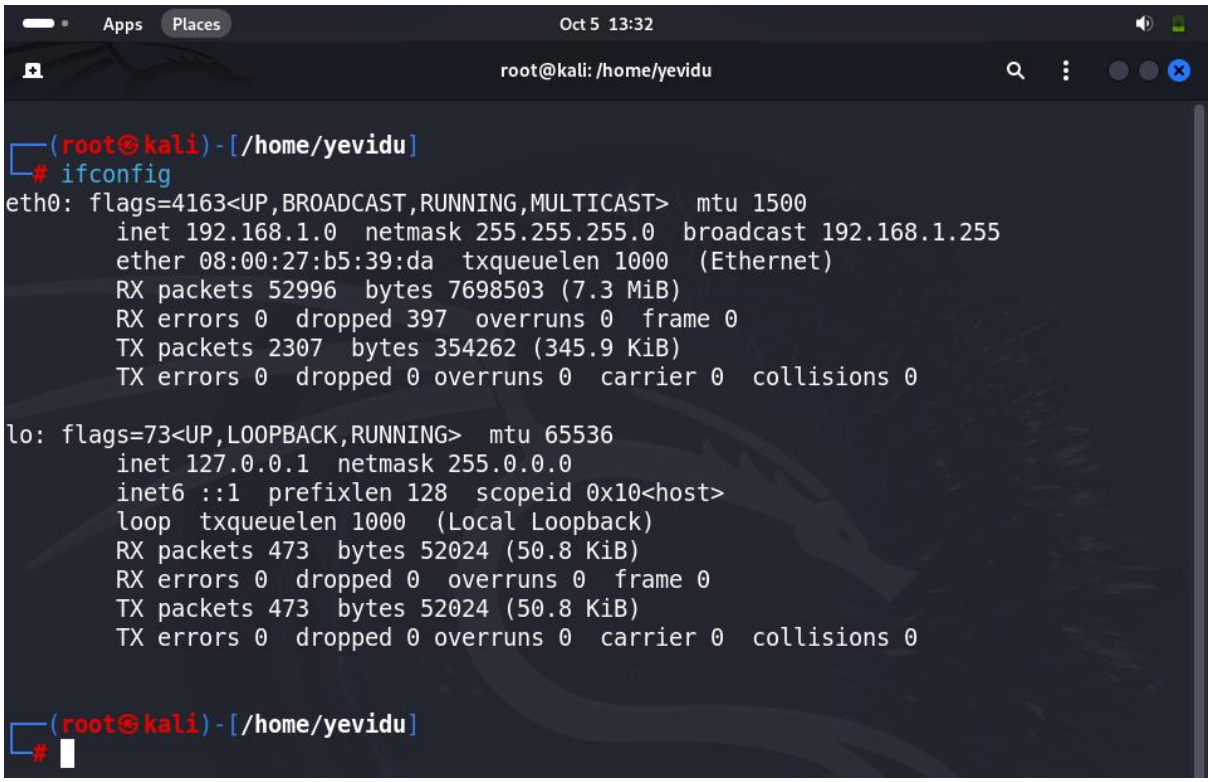
Then we install the DHCP server.

Command **apt install isc-dhcp-server.**



Check IP address.

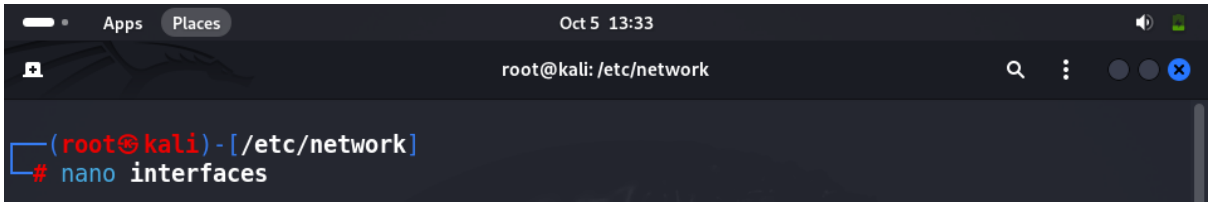
Command **ifconfig.**



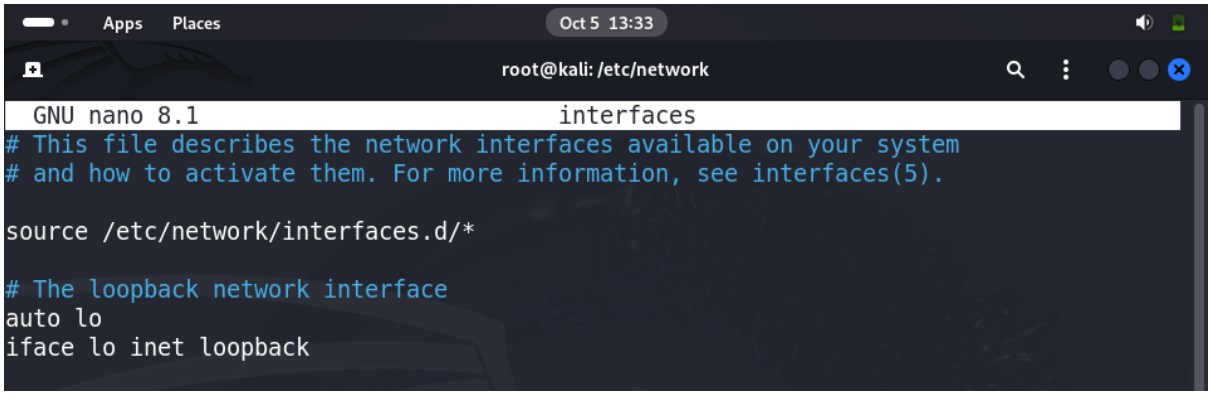
We need to edit interfaces.txt fie.

Then open interfaces.txt

Command **nano interfaces**.

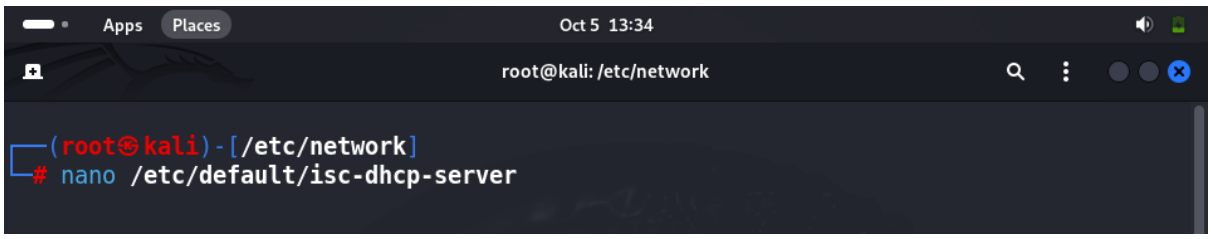


Then edit interfaces file.



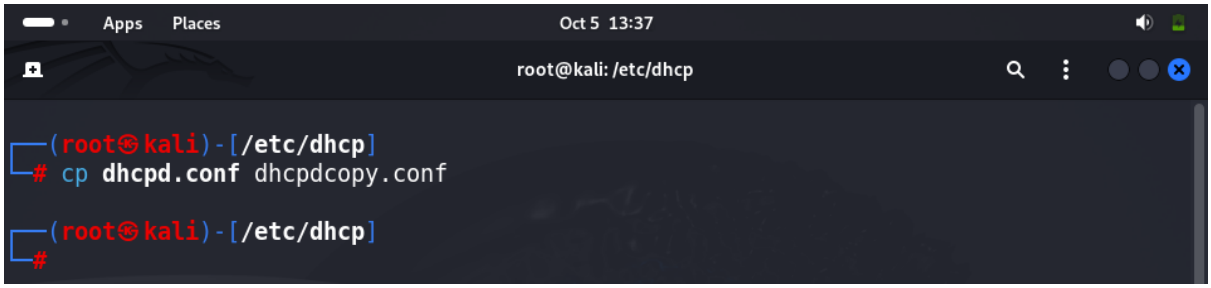
Open isc-dhcp-server file.

Command **nano /etc/default/isc-dhcp-server.**



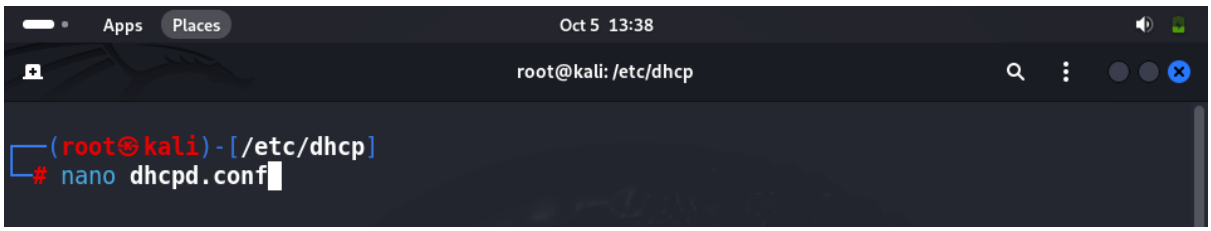
Edit the INTERFACEsv4 = “ eth0”.

Make backup of dhcp.conf file.

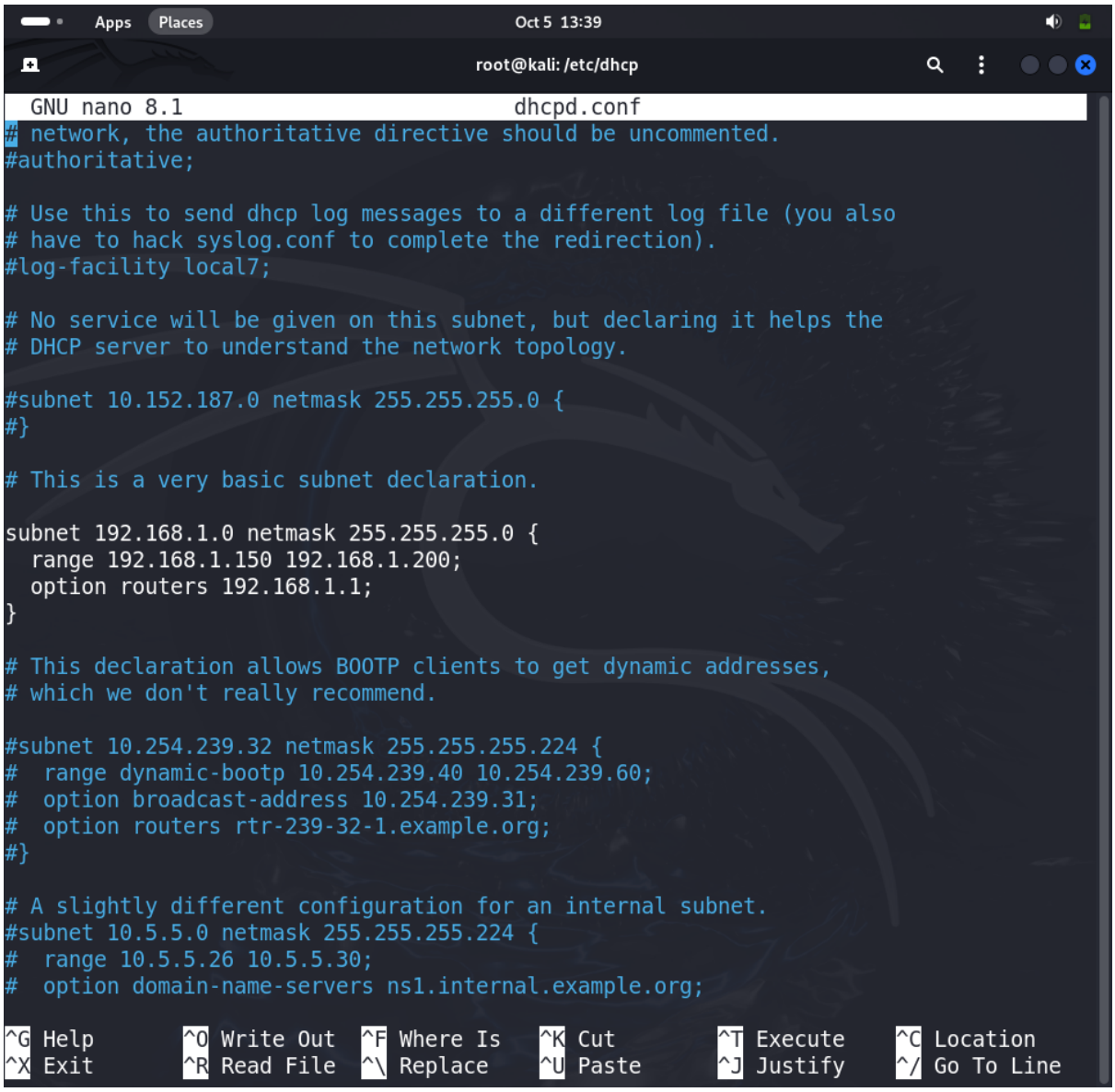


Open dhcp.conf file.

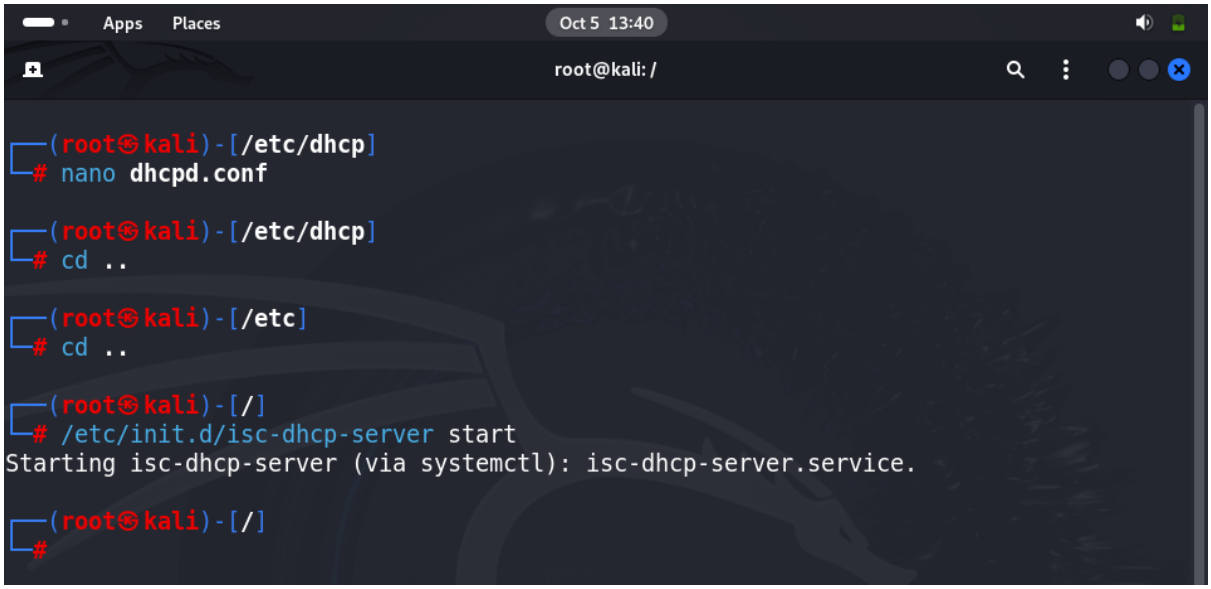
Command **nano dhcpd.conf.**



Edit dhcp.conf file.

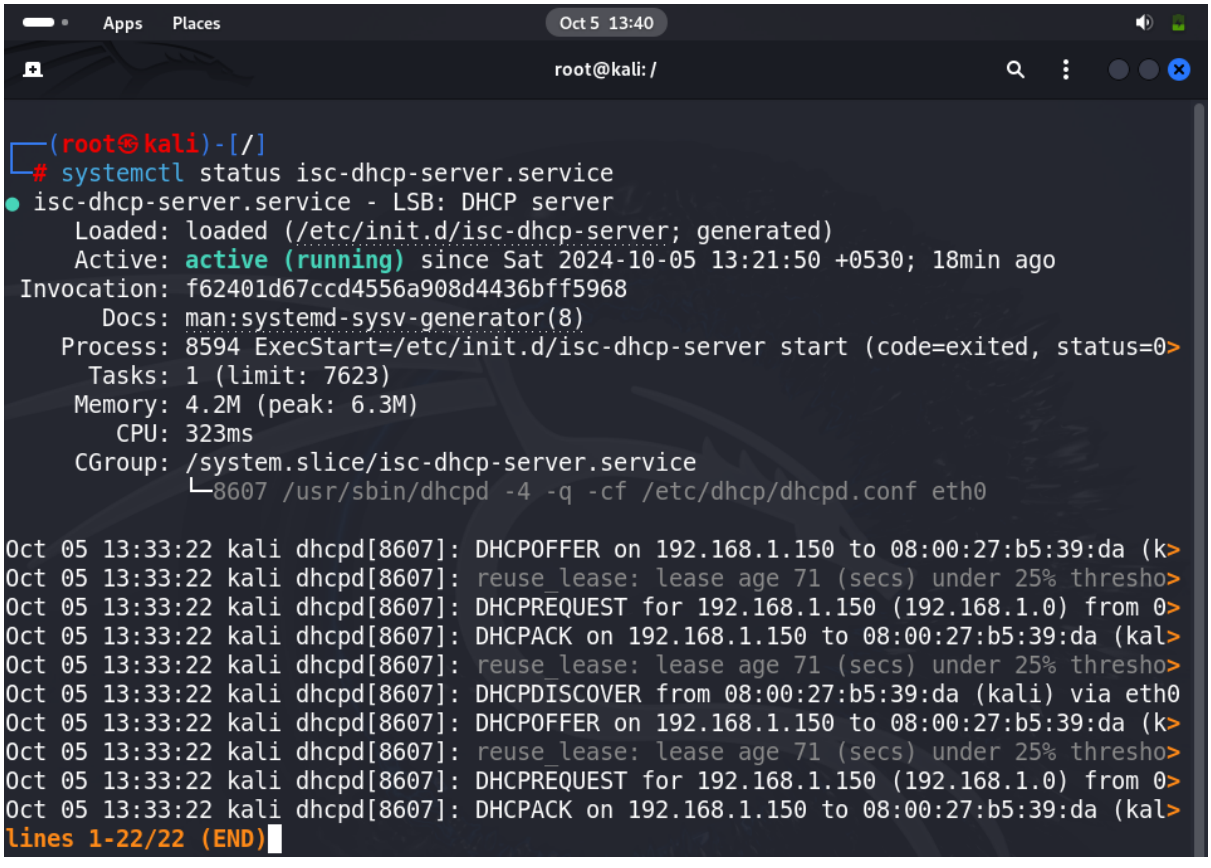


Start the dhcp server.



Now let's check the status to confirm whether DHCP is working properly.

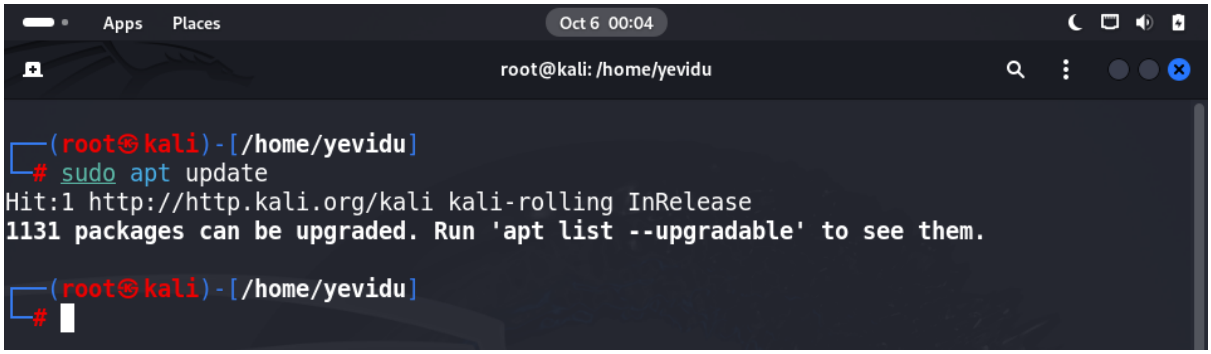
Command **systemctl status isc-dhcp-server.service.**



2.2 DNS (Domain Name System).

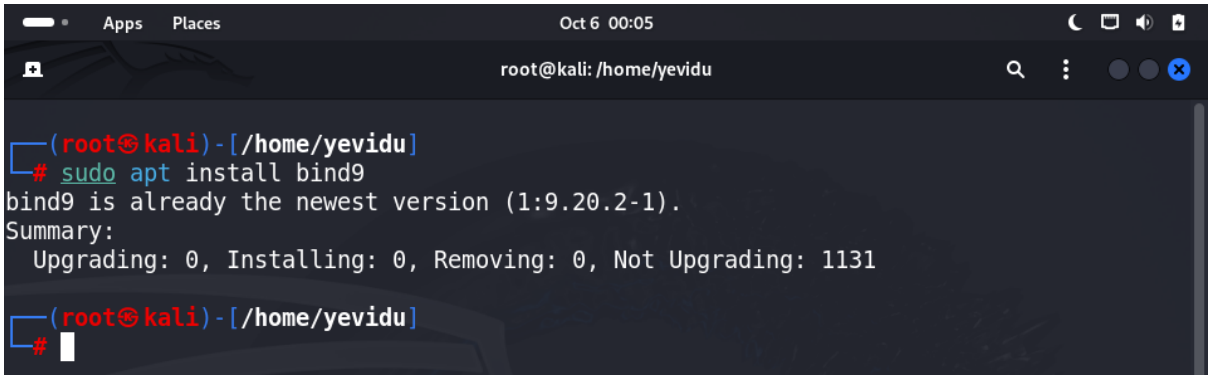
Before installing BIND9, ensure your package list is up to date. So 1st update package.

Command **sudo apt update.**



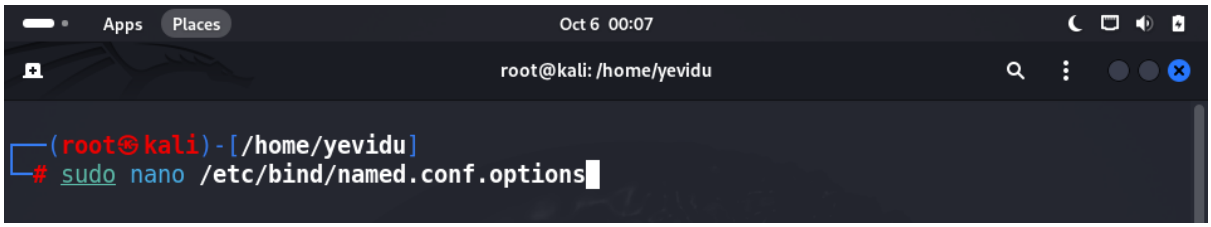
Install BIND9 package.

Command **sudo apt install bind9.**



After installation, you need to configure BIND9. So open main configuration file.

Command **sudo nano /etc/bind/named.conf.options**



Then edit main configuration file.

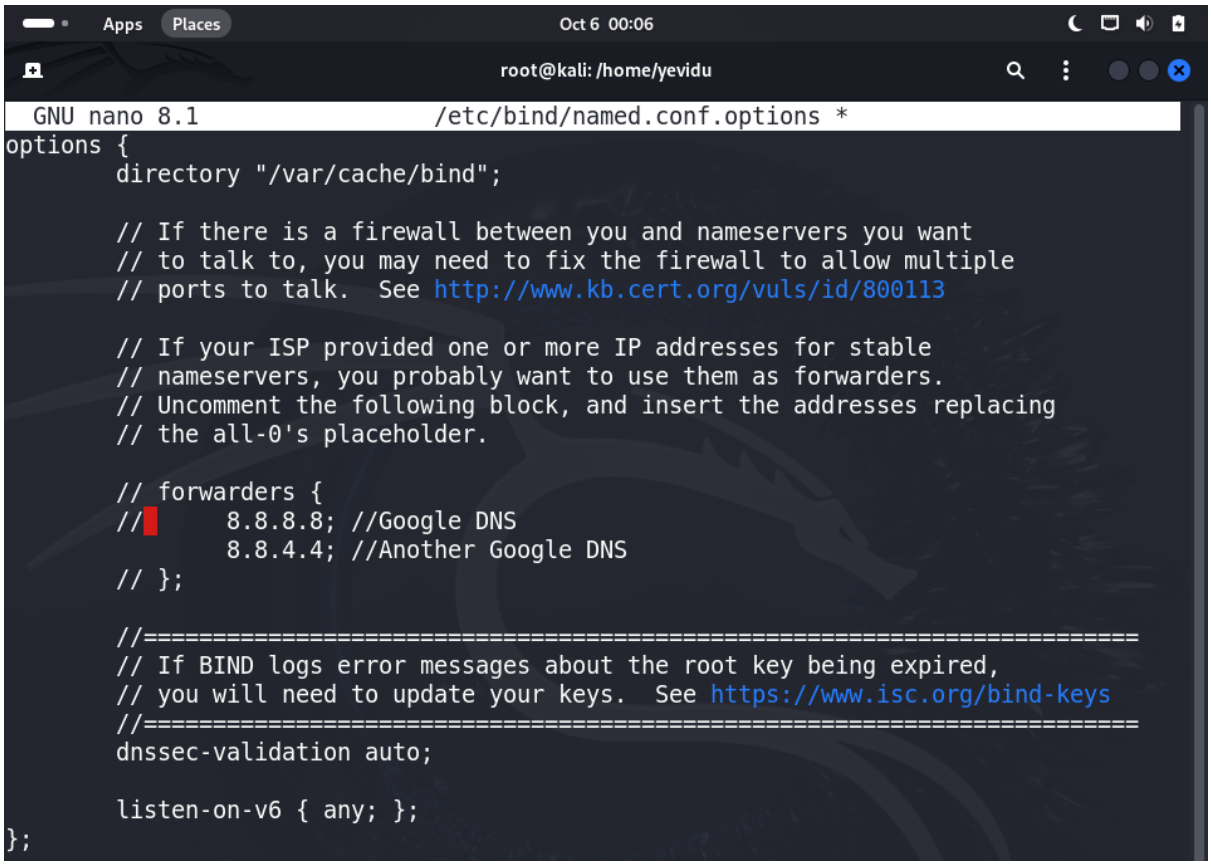
// Forwarders

forwarders {

8.8.8.8; // Google DNS

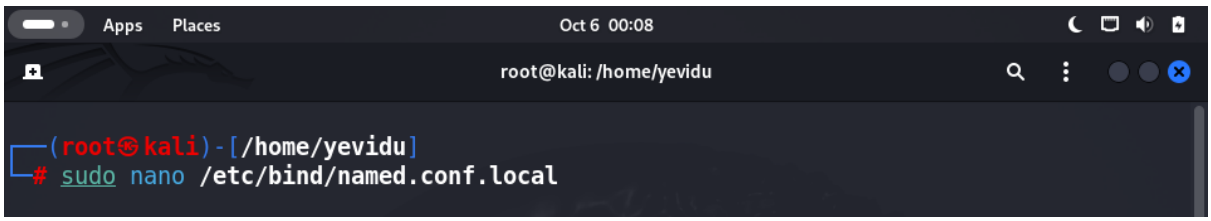
8.8.4.4; // Google DNS

};

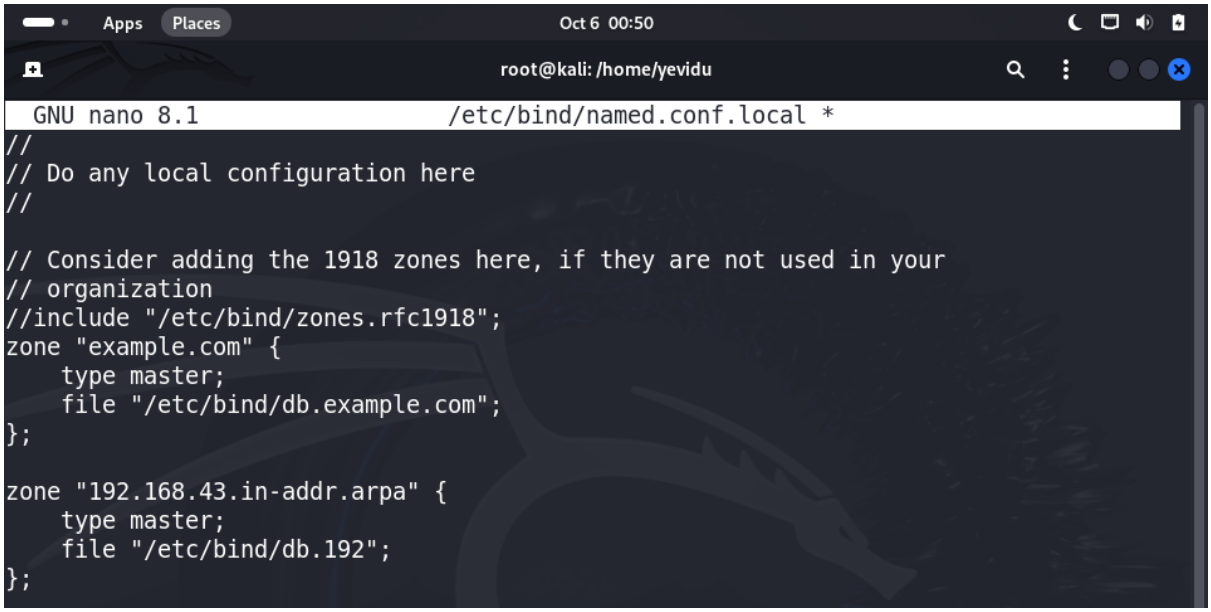


Next, define your zones by editing the named.conf.local file.

Command **sudo nano /etc/bind/named.conf.local**

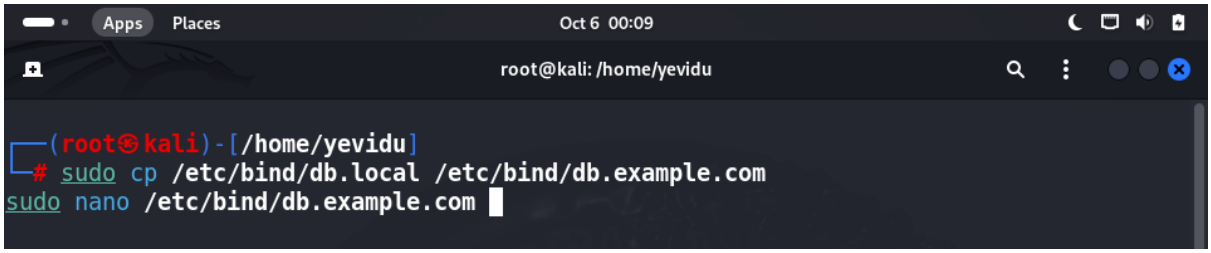


Edit local file.

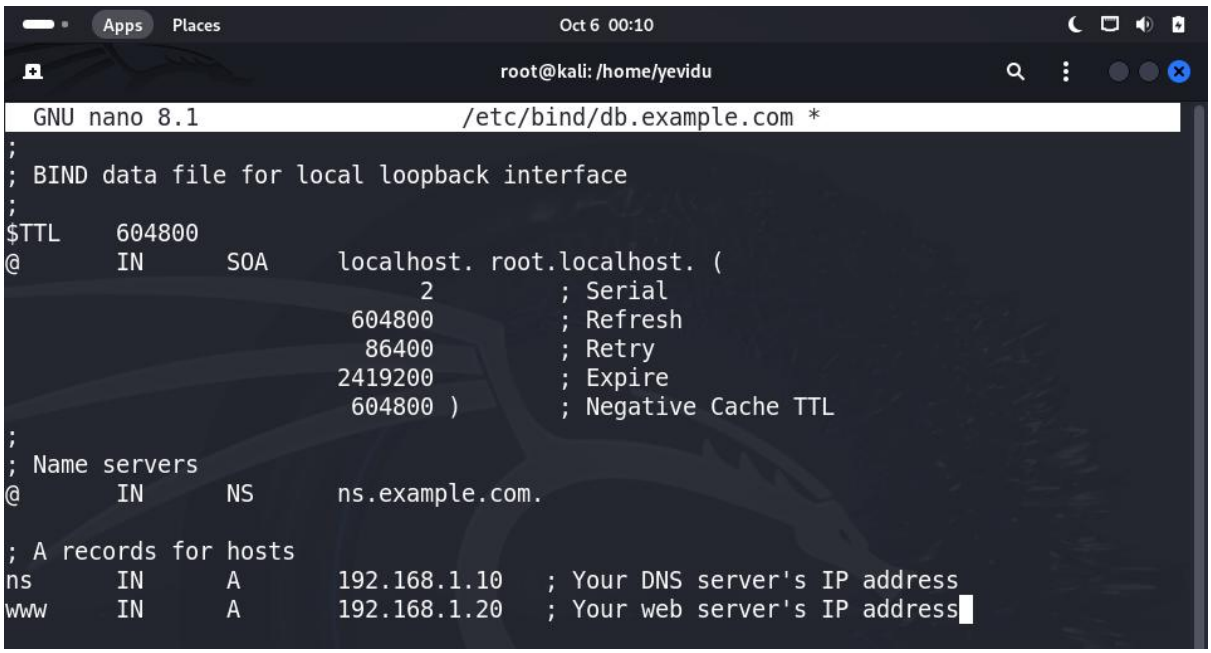


Create zone files for our domain.

Command **sudo cp /etc/bind/db.empty /etc/bind/db.example.com**

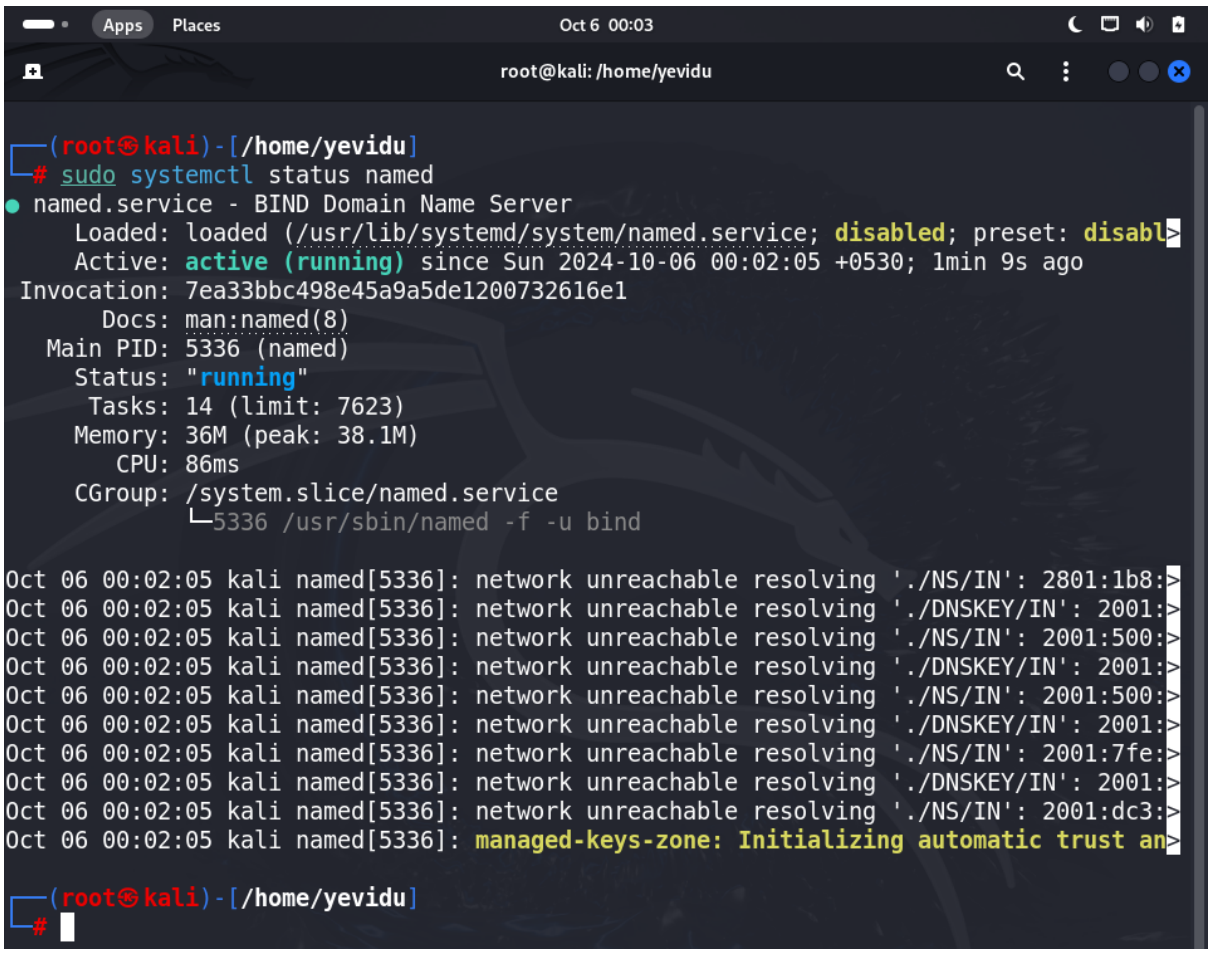


Edit new zone file.



Check whether the BIND9 server running correctly.

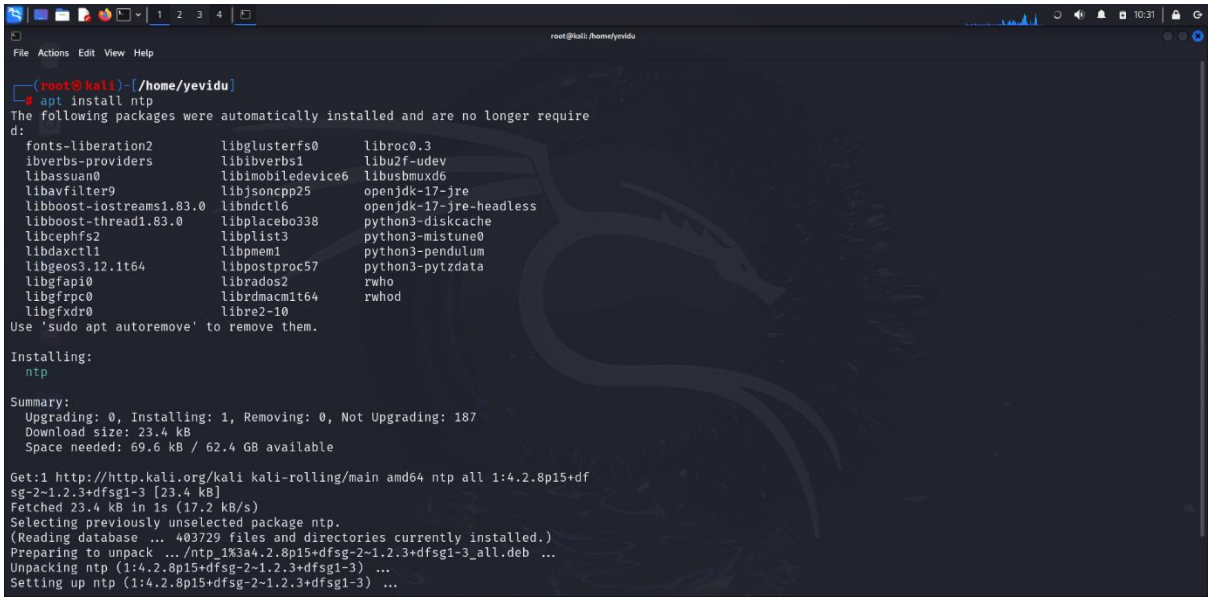
Command **sudo systemctl status bind9**



2.3 NTP (Network Time Protocol).

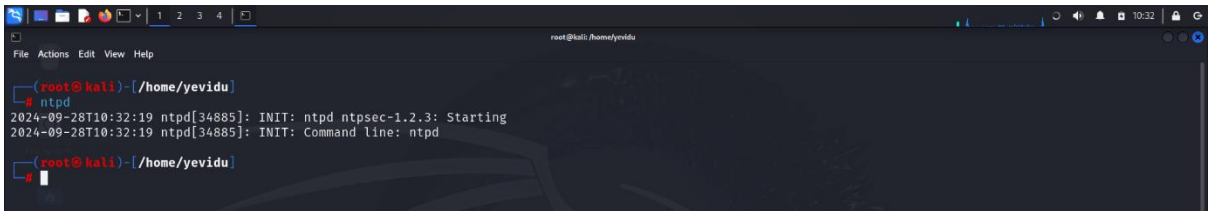
First install the NTP package.

Command **apt install ntp**



Once the installation is complete, the NTP service should start automatically. If not, you can start it manually.

Command ntpd

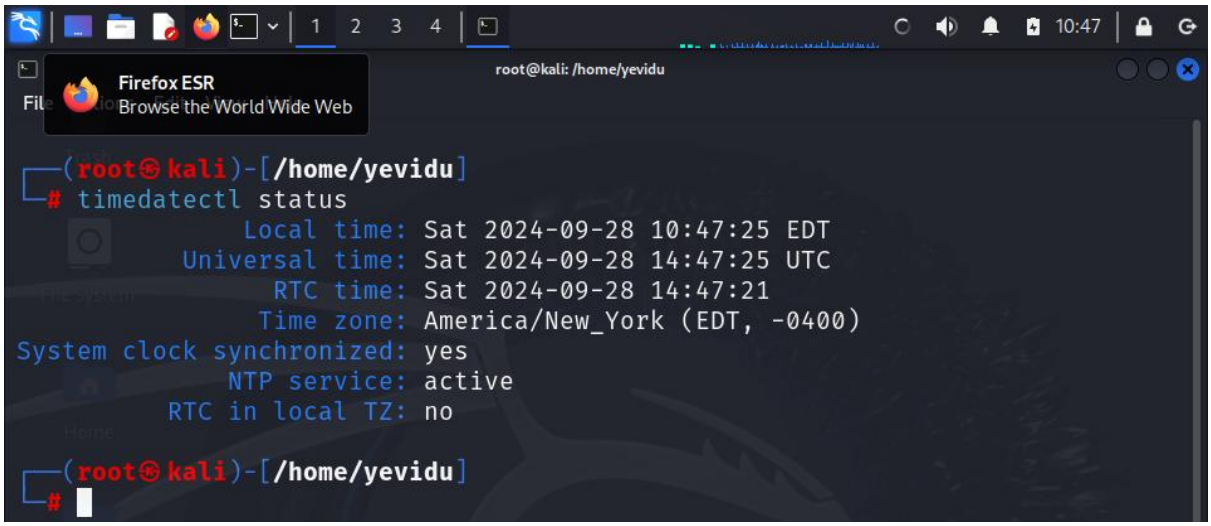


Now you can test our server.

Command **ntpq -c rv**

let's check if it is enabled on our server.

Command **timedatectl status.**



3. Shell Scripting and Security.

3.1 Shell Scripting.

a). Write a script to automate a report that captures key system details every day.

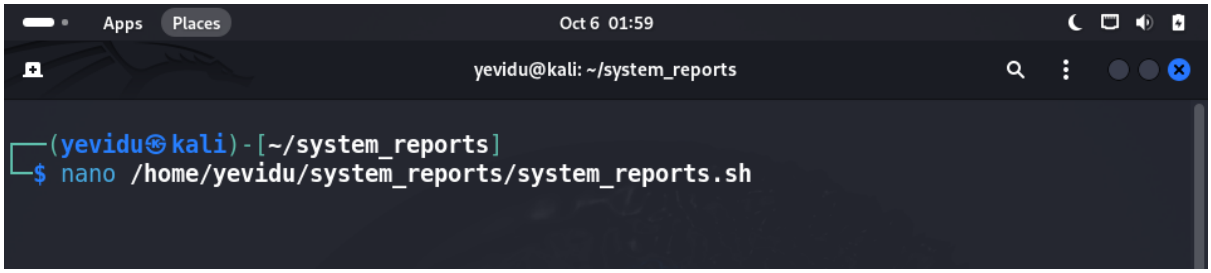
This script can be scheduled to run using cron jobs. Get System Information such as *Date, Uptime, Free memory and Disk Usage*. Create a report file at the location with the file name as mentioned below.

Destination directory: /home/user/system_reports

Open your terminal and create a new script file named system_report.sh.

Then view created file and navigate to that file.

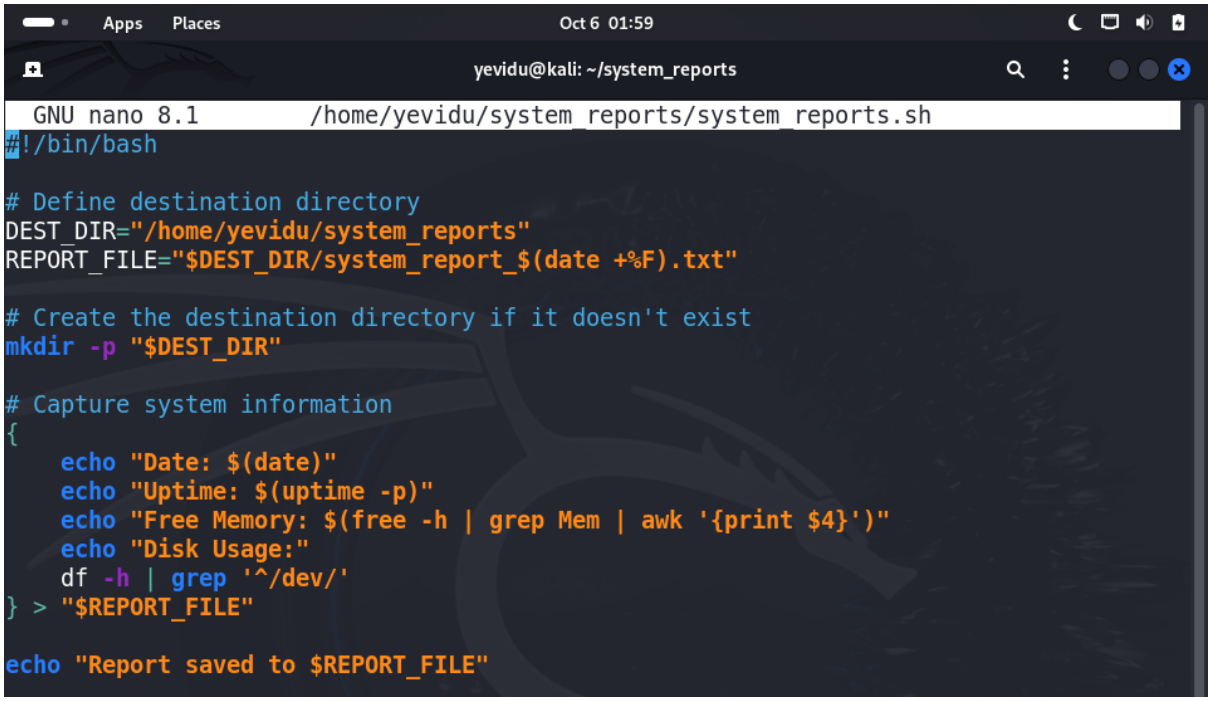
Go to new script file for editing.



Add the following lines to system_report.sh. This script gathers system information and writes it to a report file.

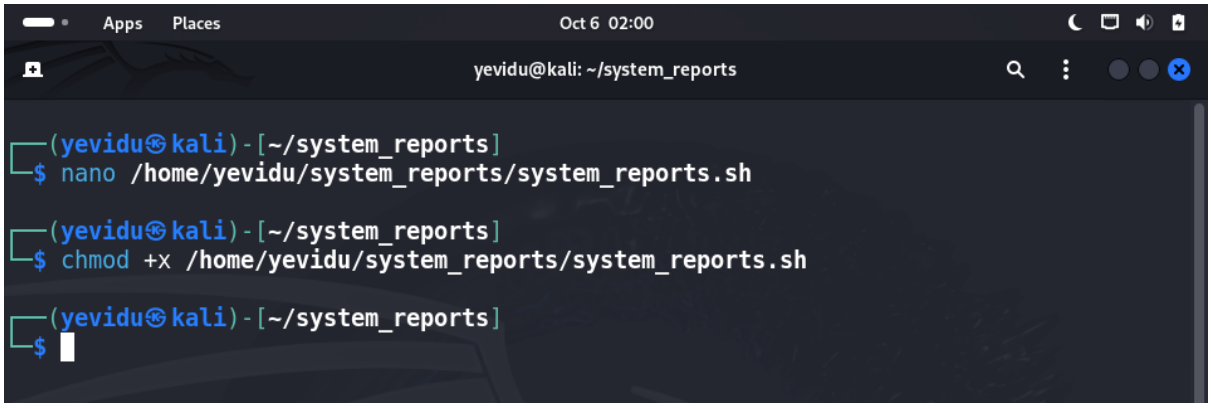
```
#!/bin/bash
DEST_DIR="/home/user/system_reports"
mkdir -p "$DEST_DIR"
DATE=$(date +'%Y-%m-%d')
REPORT_FILE="$DEST_DIR/system_report_$DATE.txt"
{
   echo "System Report for: $DATE"
   echo "============================="
   echo "Uptime: $(uptime -p)"
   echo "Free Memory: $(free -h | grep 'Mem' | awk '{print $4}')"
   echo "Disk Usage:"
   df -h | grep '^/dev/'
} > "$REPORT_FILE"

echo "Report generated at: $REPORT_FILE"
```



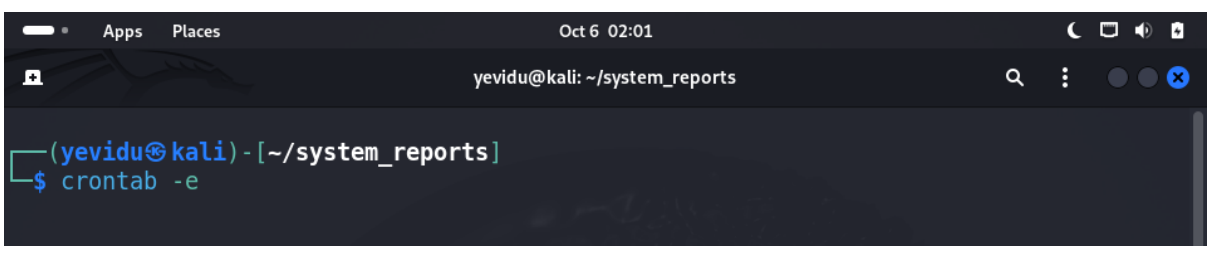
After saving the file.Then make it executable.

Command **chmod +x ~/system_report.sh**

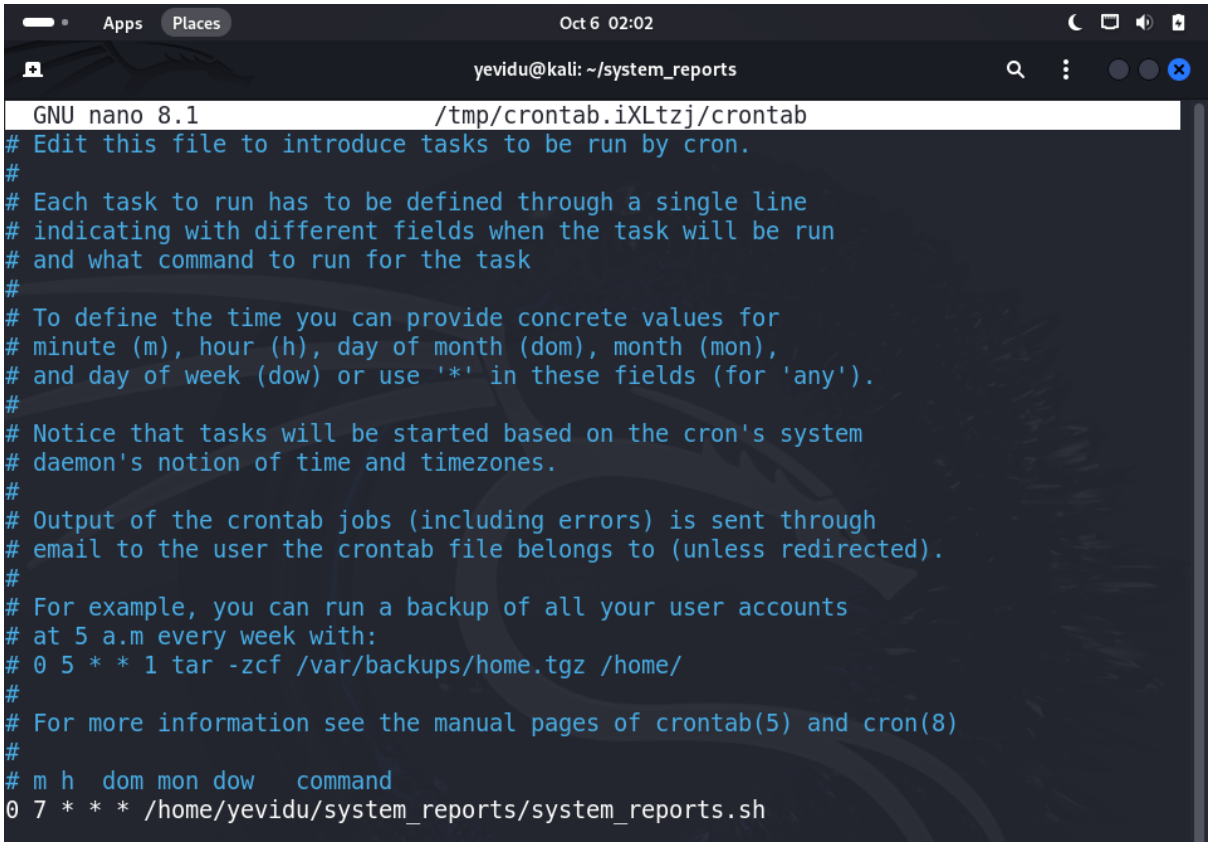


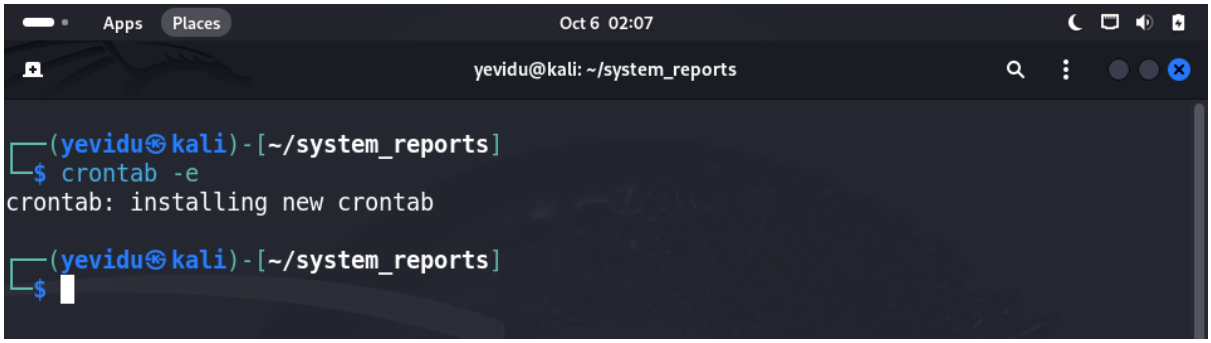
To Set Up a Cron Job lets Open the cron jobs configuration file.

Command **crontab -e**



Now let’s add a new line to schedule the script to run daily. to run the script every day at 7:00 PM.





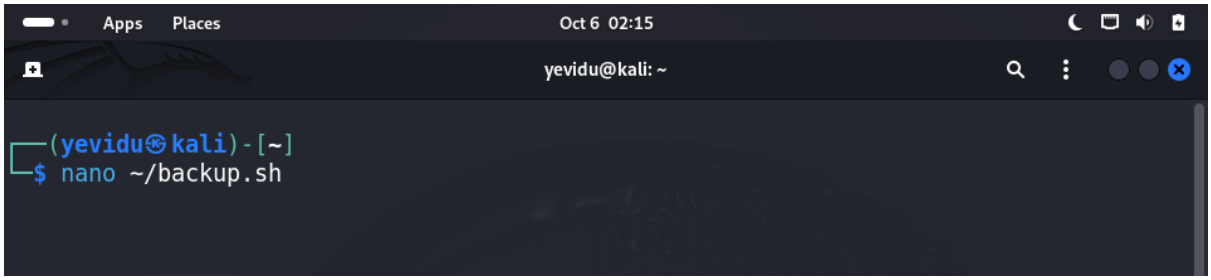
b). Write a script to automate the backup of a critical directory (/home/user/documents) containing important files. This script can be scheduled to run periodically. Make sure to name the backup file with the date.

Source directory : /home/user/documents

Destination directory : /home/user/backup/documents

Open your terminal and create a new file named backup.sh in preferred directory.

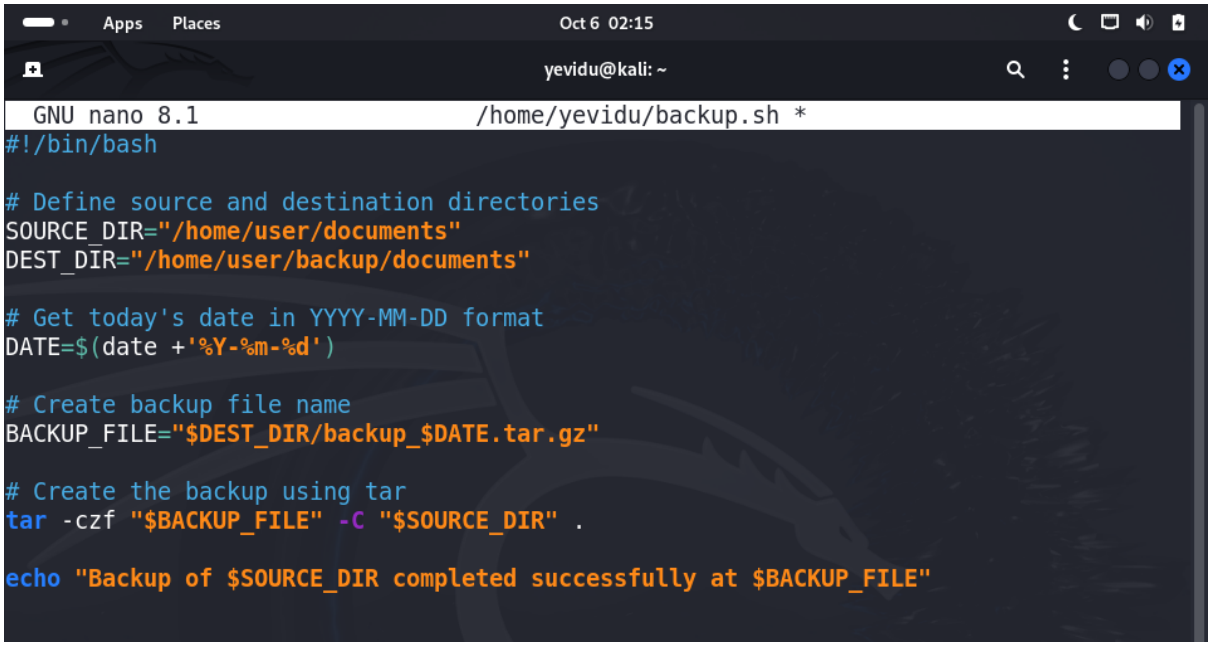
Command **nano ~/backup.sh**



Add the following lines to backup.sh.

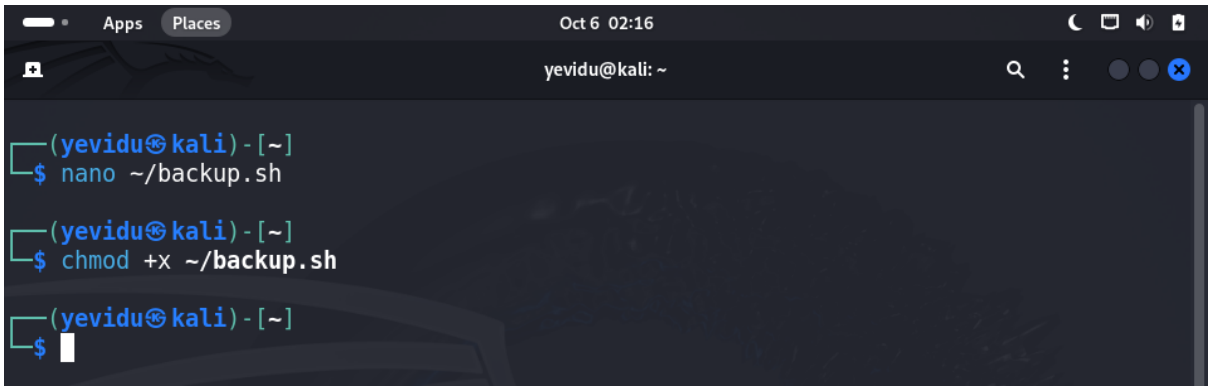
```
#!/bin/bash
SOURCE_DIR="/home/user/documents"
DEST_DIR="/home/user/backup/documents"
DATE=$(date +'%Y-%m-%d')
BACKUP_FILE="$DEST_DIR/backup_$DATE.tar.gz"
tar -czf "$BACKUP_FILE" -C "$SOURCE_DIR" .

echo "Backup of $SOURCE_DIR completed successfully at $BACKUP_FILE"
```



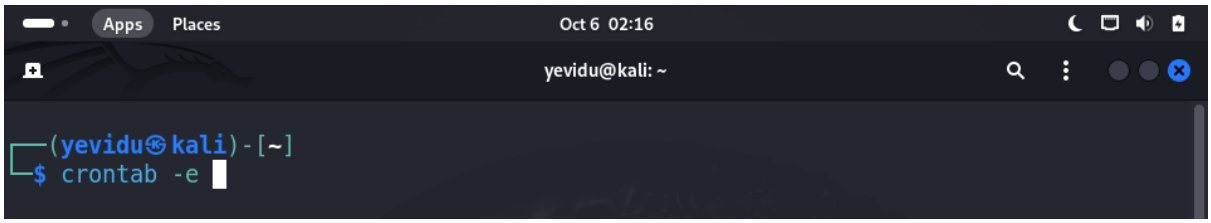
After saving the file.Then make it executable.

Command **chmod +x ~/system_report.sh**

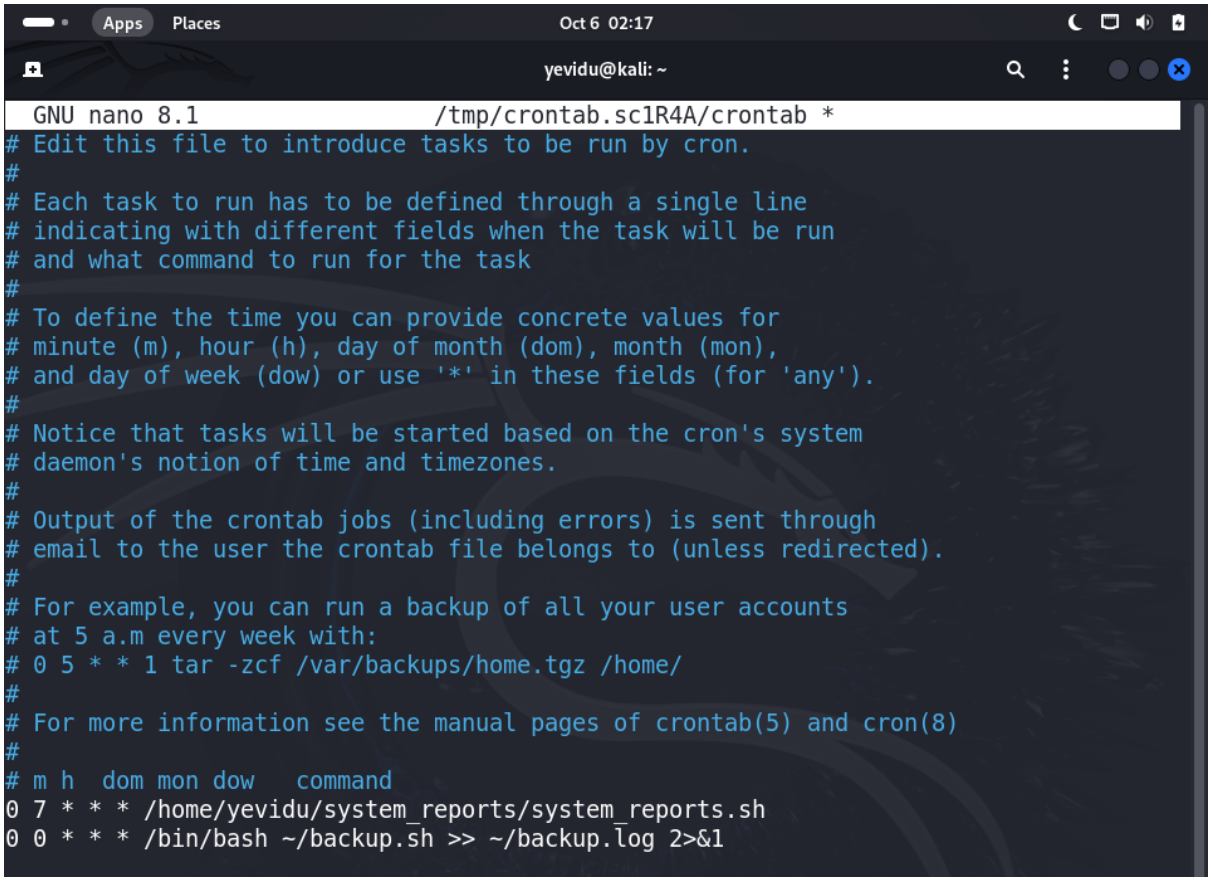


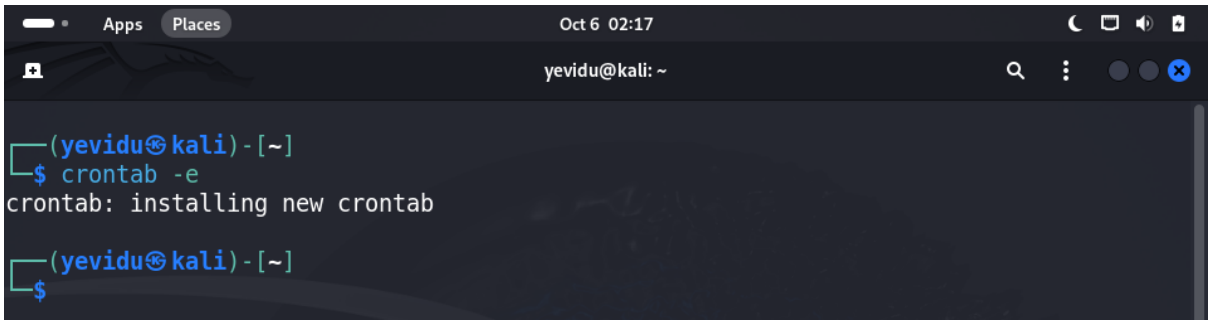
Open crontab**.**

Command **crontab -e**



To schedule the backup to run daily at midnight, added the following line to your crontab:





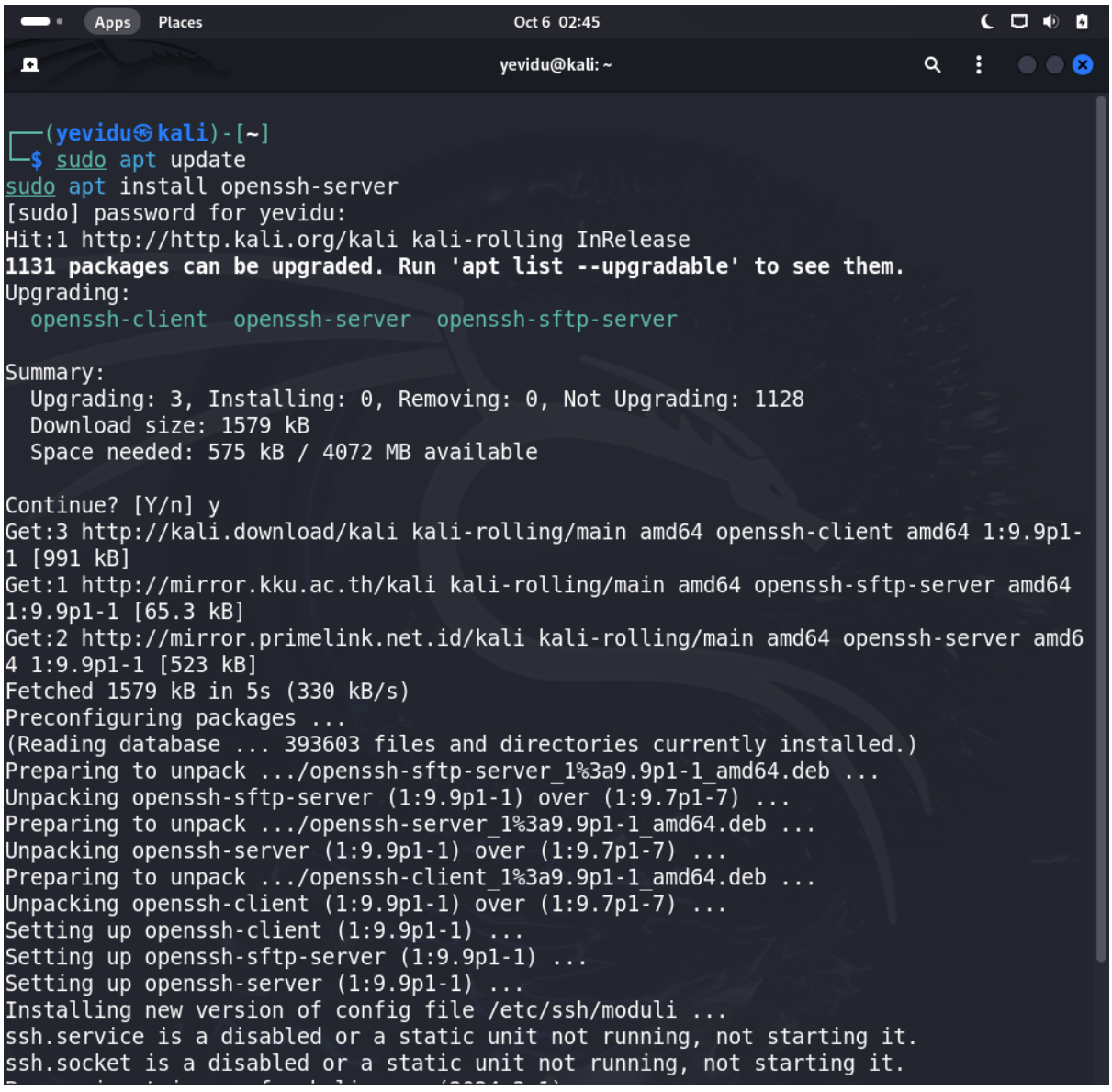
3.2 SSH (Secure Shell).

First install OpenSSH into Linux.

Command

sudo apt update

sudo apt install openssh-server

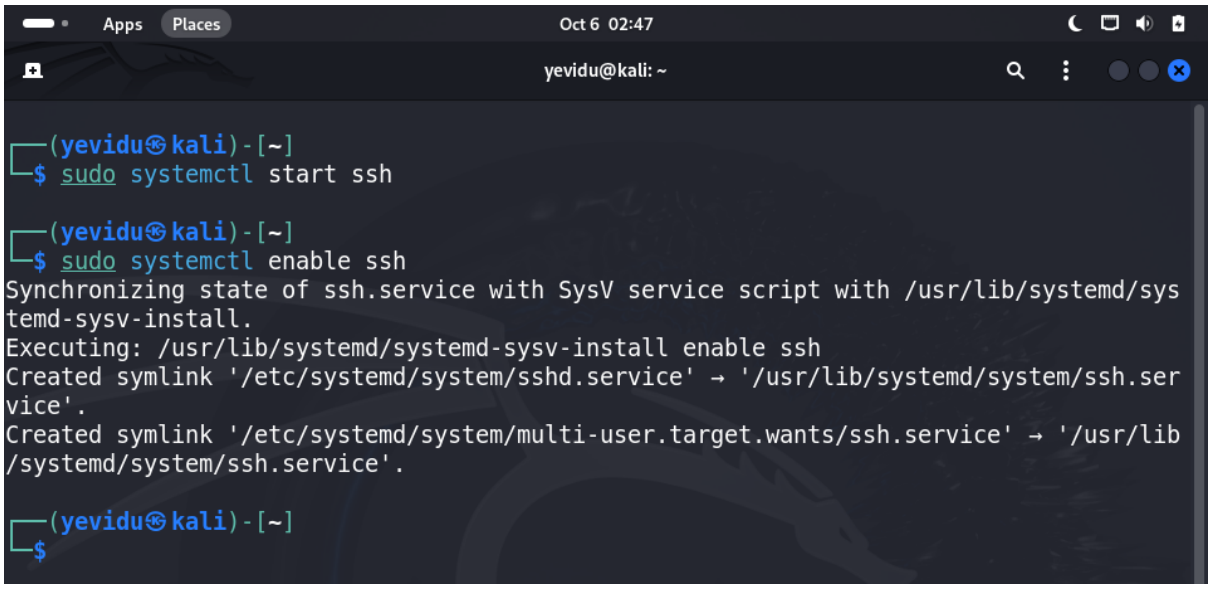


After installation, start the SSH service and enable it.

Command

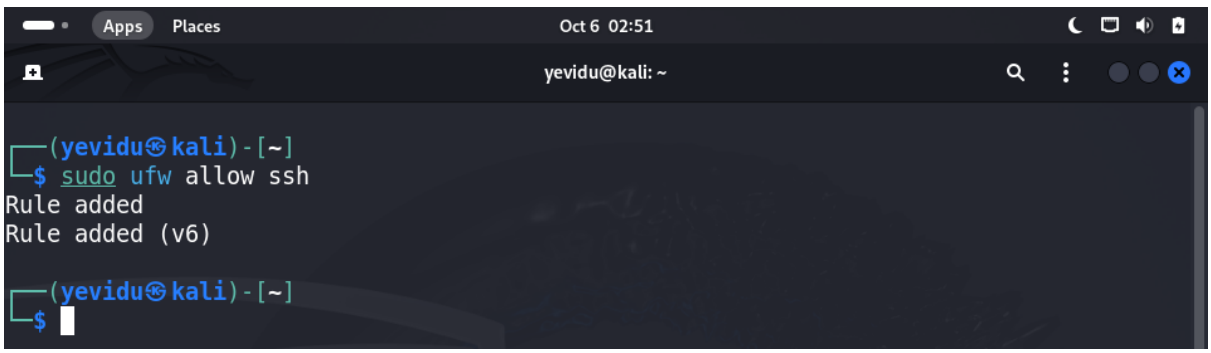
sudo systemctl start sshd

sudo systemctl enable sshd



Ensure that your firewall allows SSH connections on port 22.

Command **sudo ufw allow ssh**



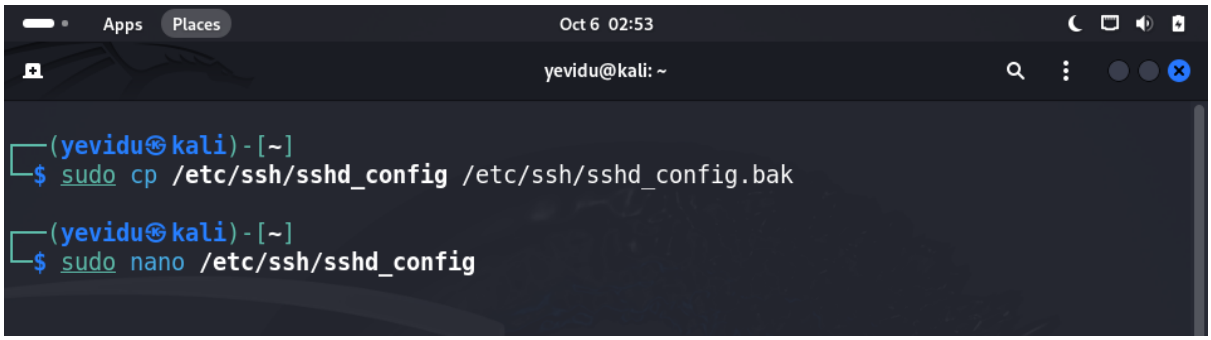
The main configuration file for SSH is located at /etc/ssh/sshd_config. Before making changes, back up the original file.

Then go to configuration file.

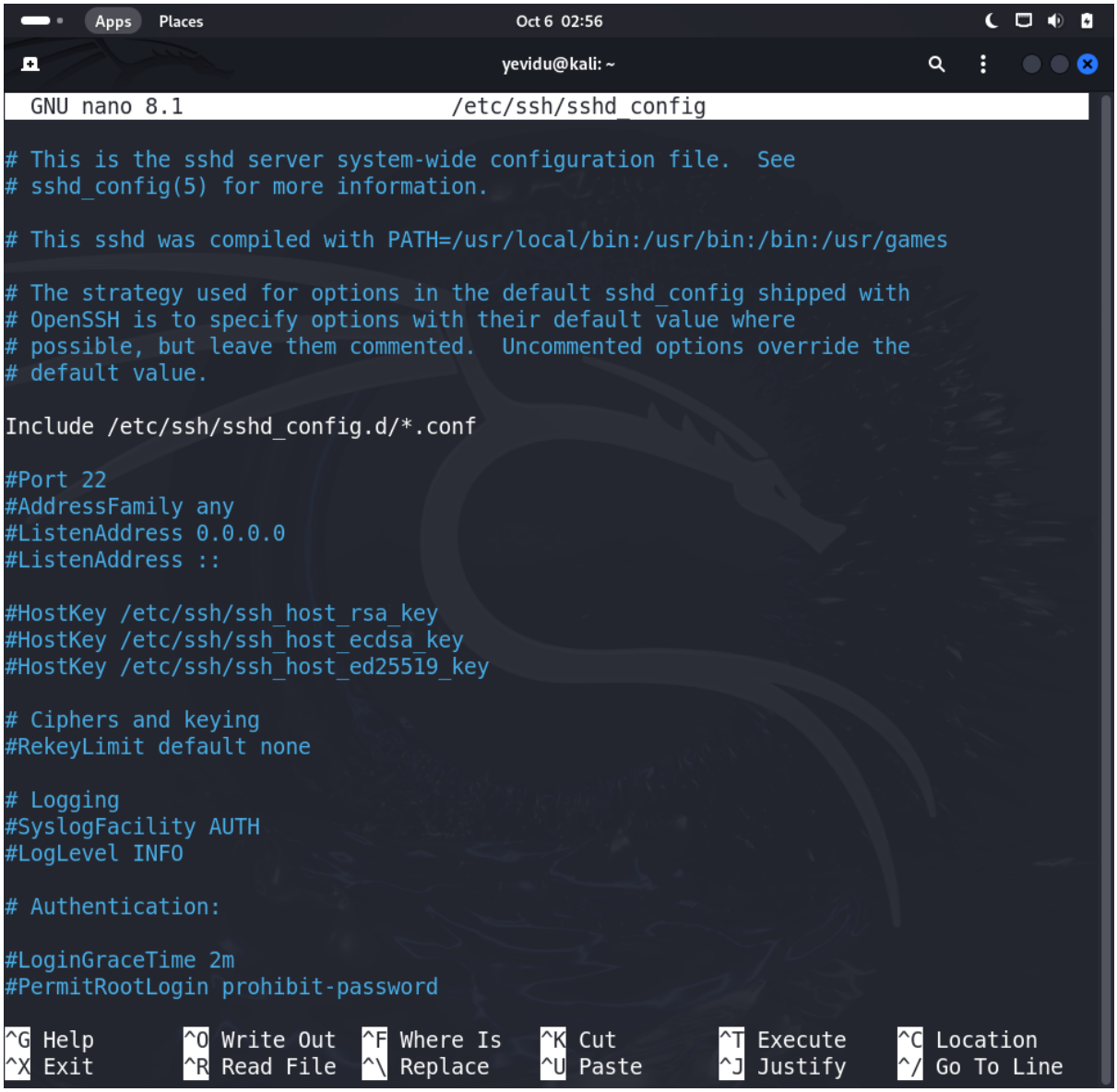
Commands

sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.bak

sudo nano /etc/ssh/sshd_config

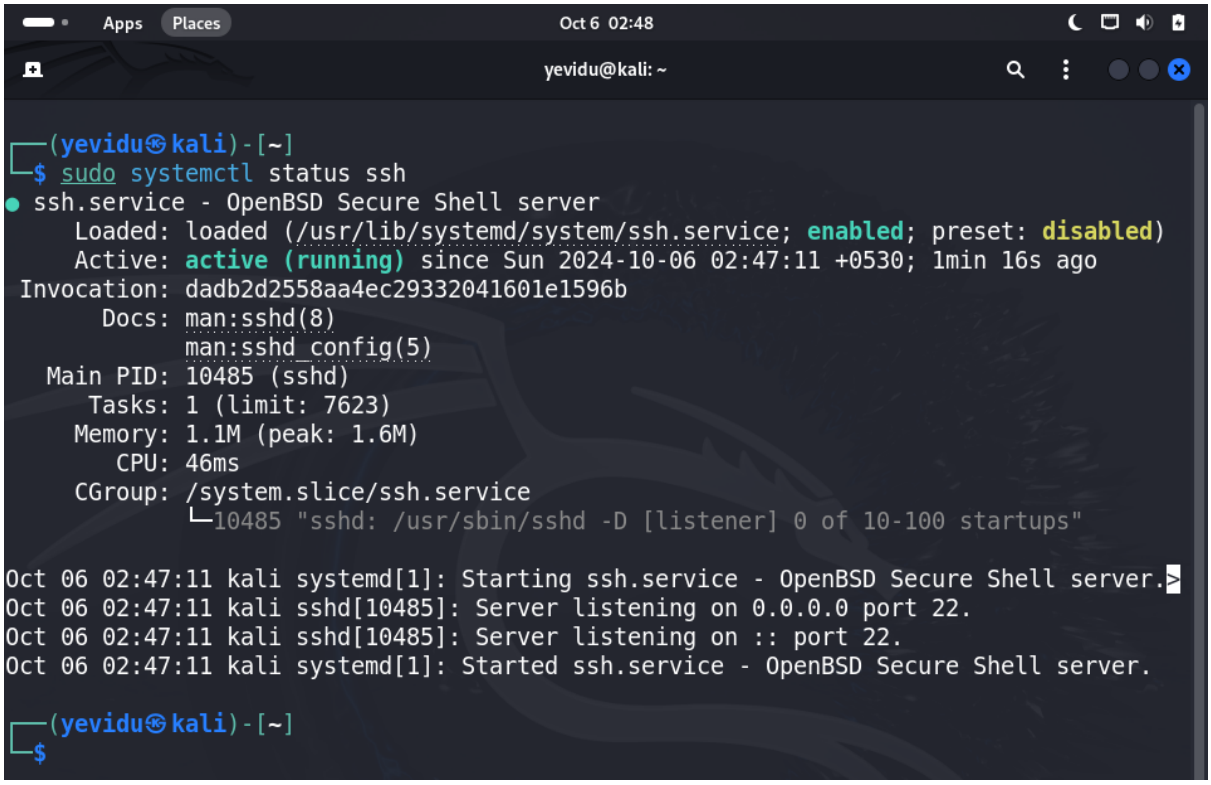


Edit the configuration file.



Check whether the SSH working properly.

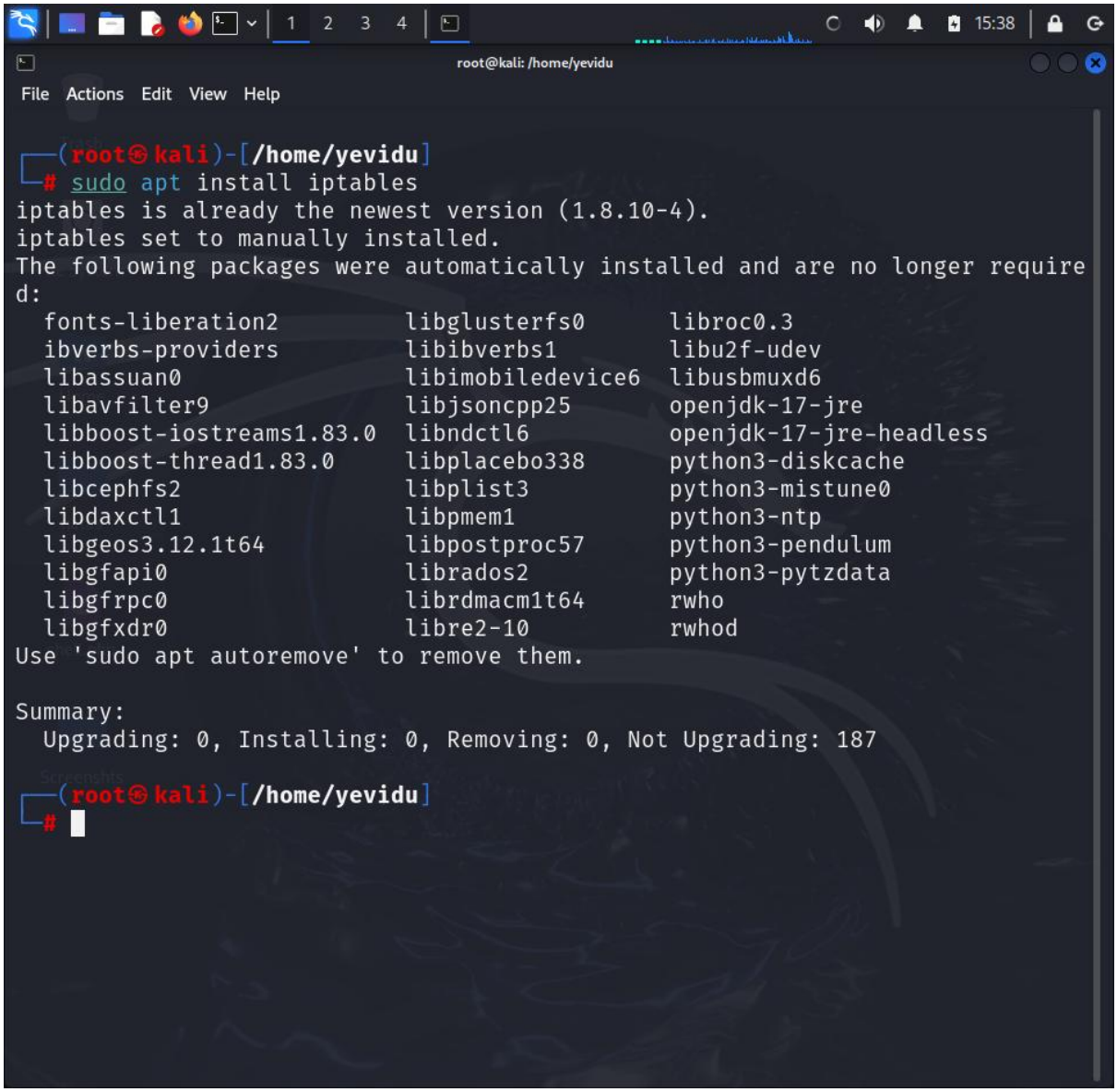
Command **sudo systemctl status ssh**



3.3 Iptables and ACLs

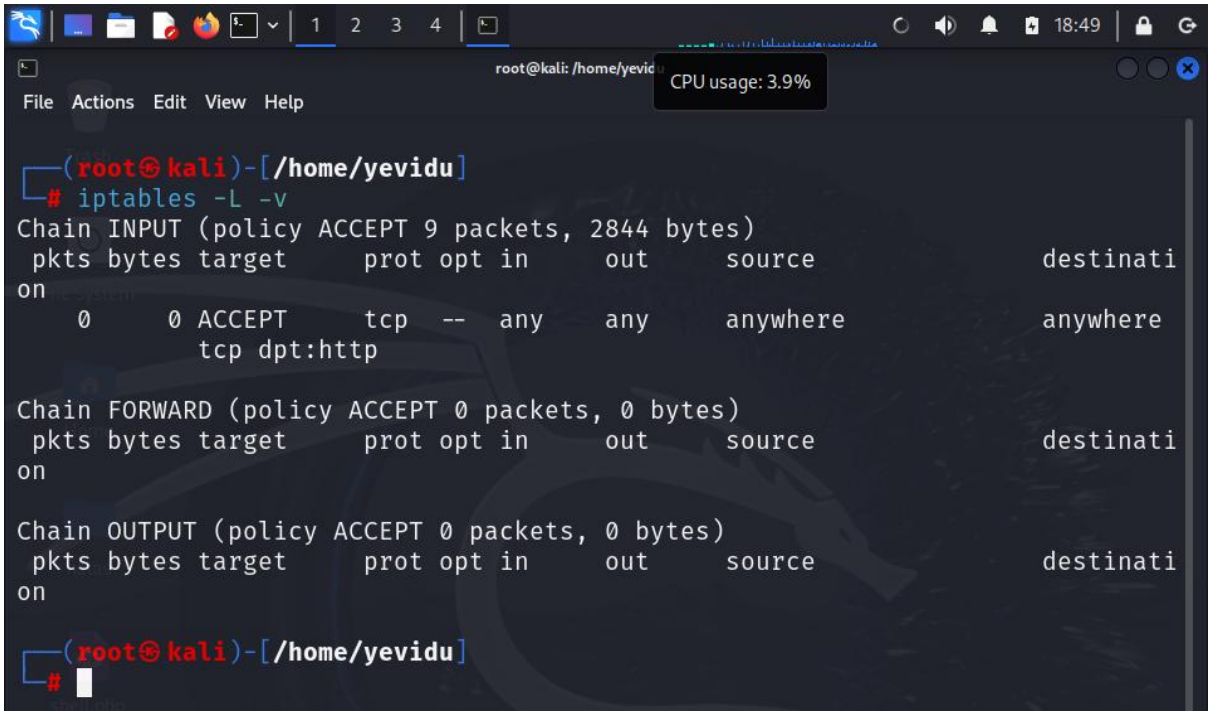
First, you need to install iptables. Most Linux operating systems come with this package pre-installed.

Command **sudo apt install iptables**



Check the status of your current iptables configuration.

Command **iptables -L -v**



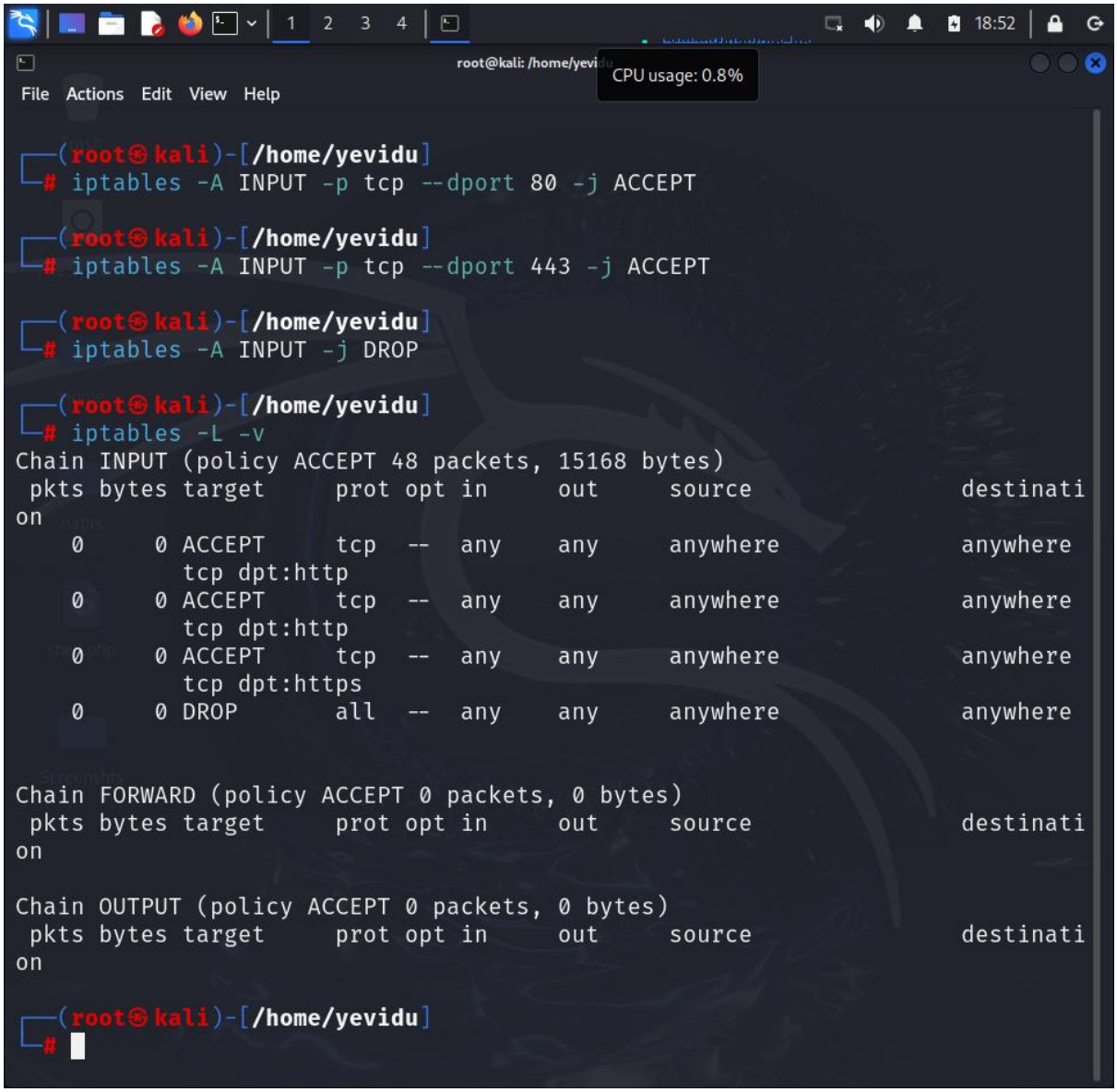
Web Server Security: Allow incoming traffic only on port 80 (HTTP) and port 443 (HTTPS) for your web server. Block all other incoming traffic by default.

Commands

iptables -A INPUT -p tcp --dport 80 -j ACCEPT

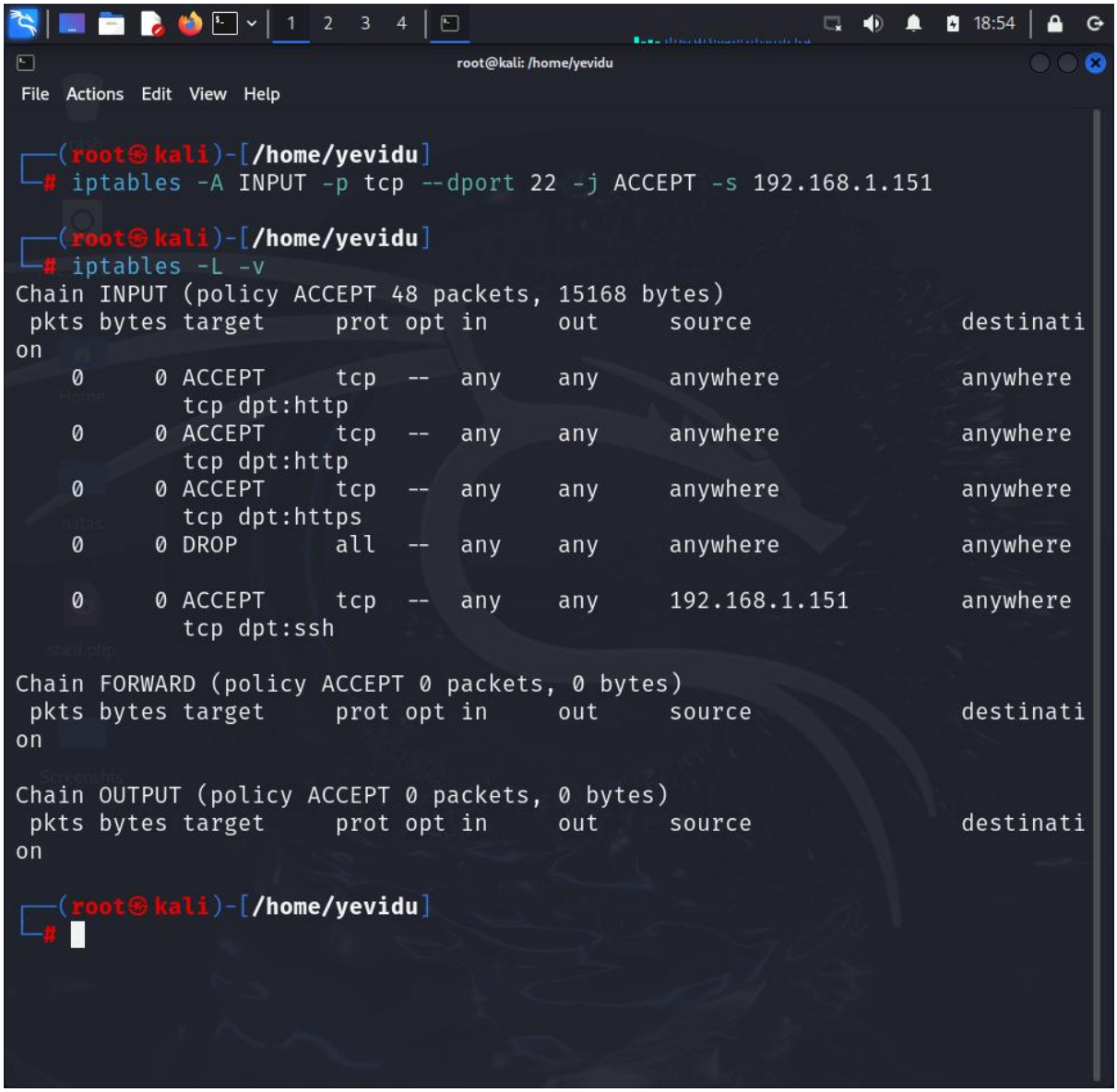
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

iptables -A INPUT -j DROP



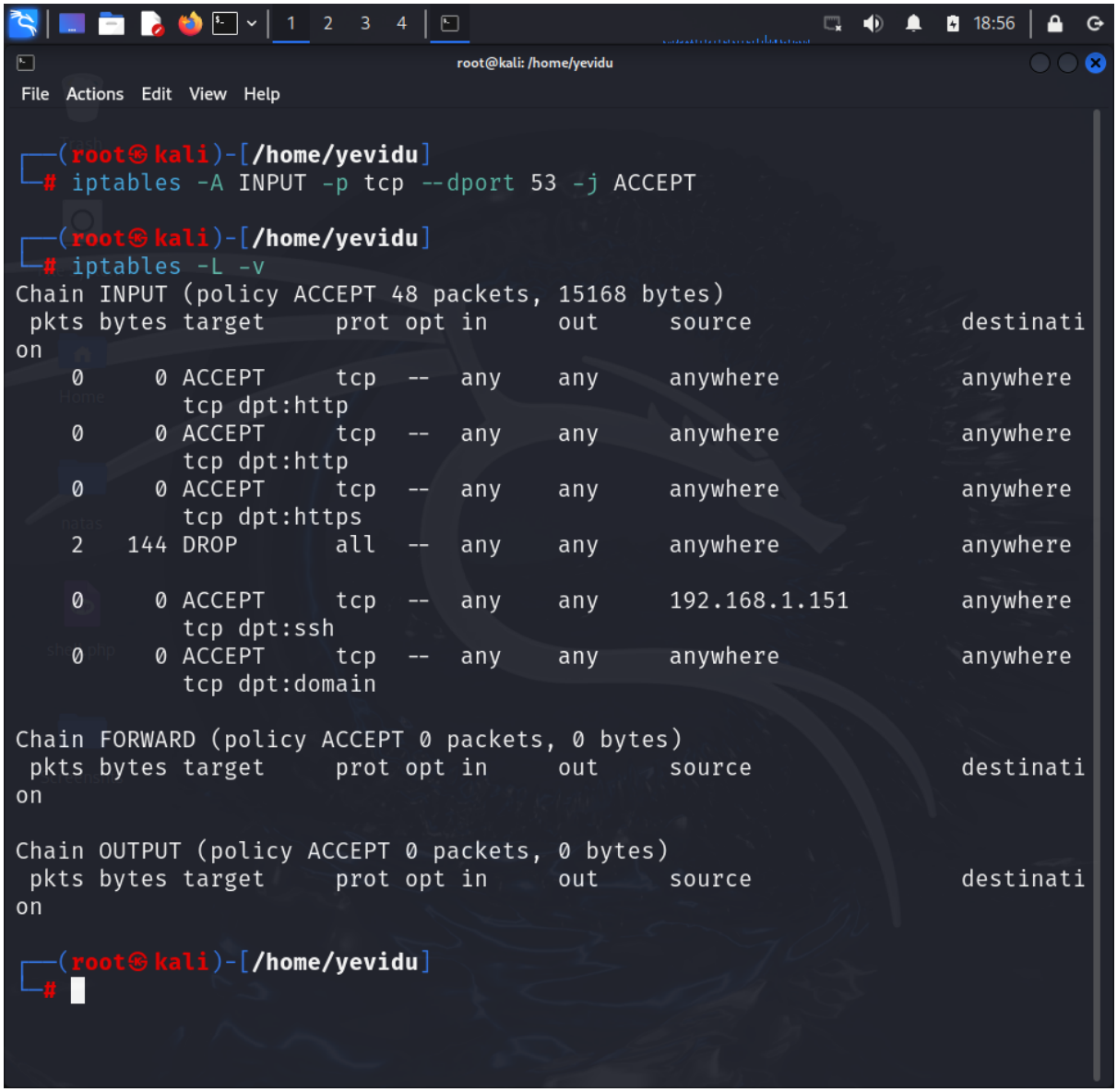
Remote Administration Access: Allow SSH access (port 22) only from specific IP addresses of your trusted machines used for administration. This restricts remote access attempts to authorized sources.

Commands iptables -A INPUT -p tcp --dport 22 -j ACCEPT -s 192.168.1.151



Allow Specific Applications: If you know the port numbers used by specific applications you want to allow (like a video conferencing app using port 443), you can create an ACL rule to permit traffic only for those ports.

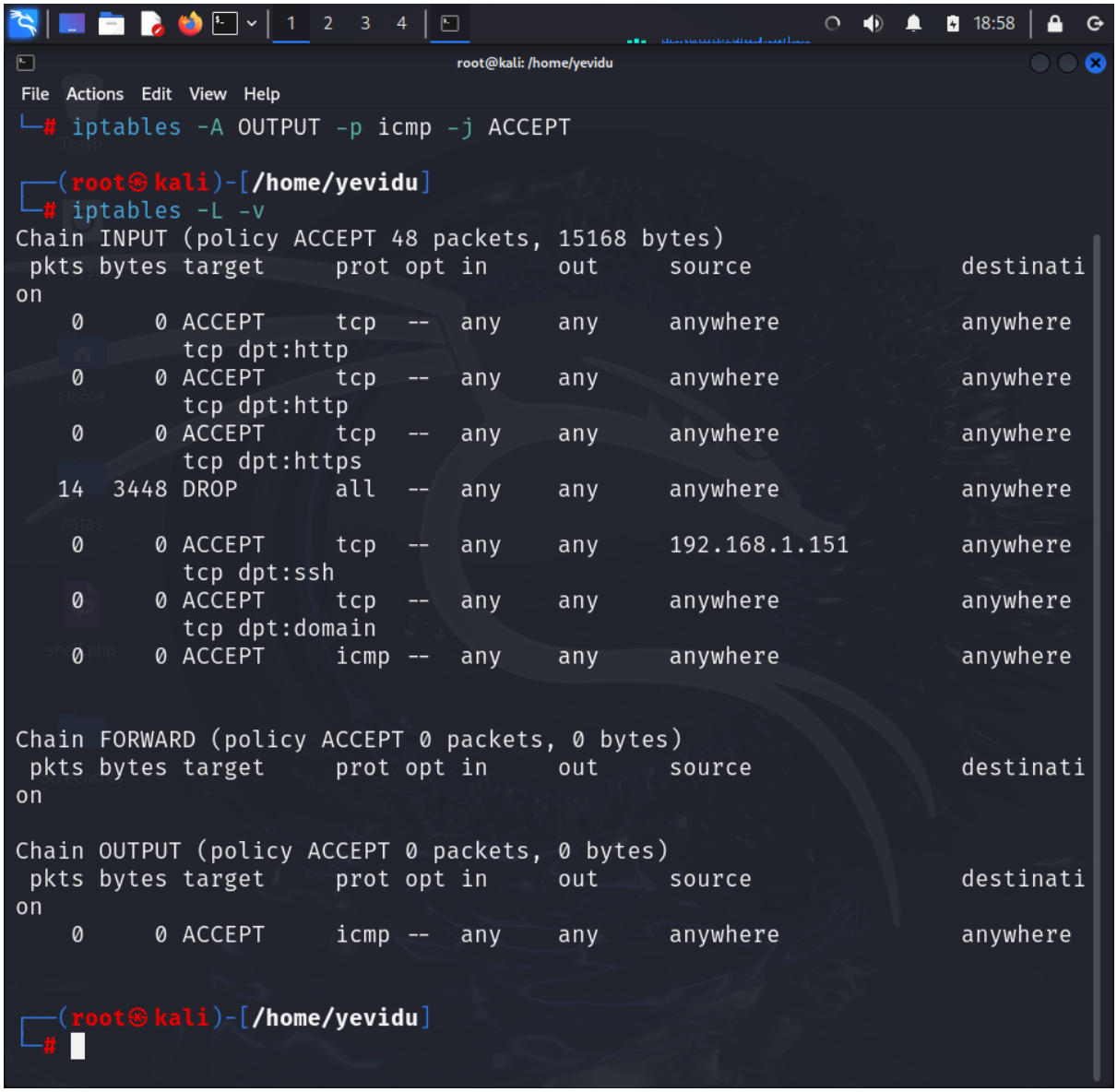
Commands **iptables -A INPUT -p tcp --dport 53 -j ACCEPT**



Allow Pings (ICMP Echo Request): This basic rule allows ping requests (ICMP Echo Request) to your machine, which can be helpful for troubleshooting network connectivity.

Commands

iptables -A OUTPUT -p icmp -j ACCEPT

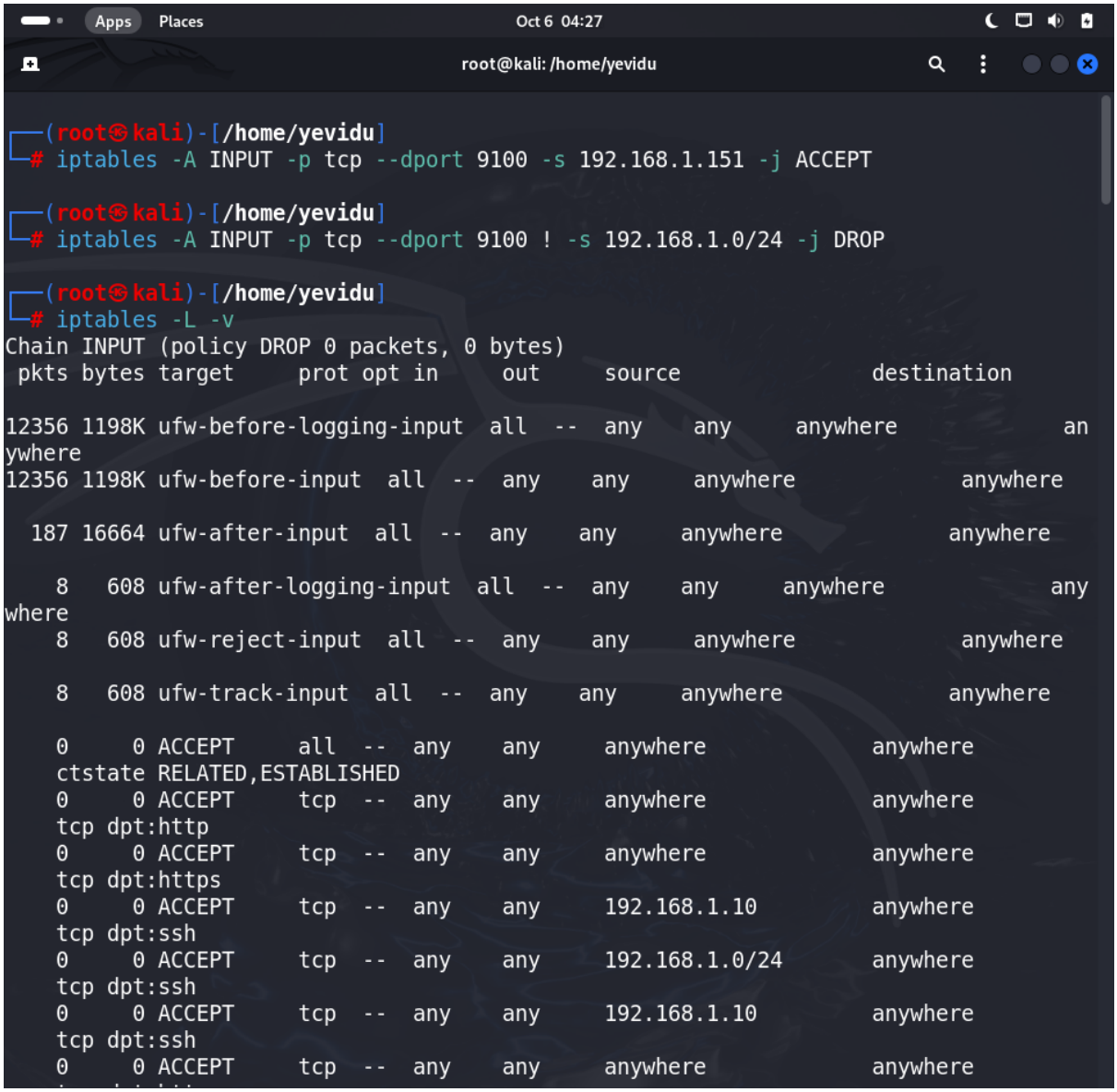


Printer Server Access: For a printer server, allow printing traffic (port 9100) only from specific IP addresses within your local network. Block all external access to the printer server to prevent unauthorized printing.

Commands

iptables -A INPUT -p tcp --dport 9100 -s 192.168.1.151 -j ACCEPT

iptables -A INPUT -p tcp --dport 9100! -s 192.168.1.0/24 -j DROP


Apps
Places
Oct 6 04:27
root@kali: /home/yevidu
┌──(root㉿kali)-[/home/yevidu]
└─# iptables -A INPUT -p tcp --dport 9100 -s 192.168.1.151 -j ACCEPT
┌──(root㉿kali)-[/home/yevidu]
└─# iptables -A INPUT -p tcp --dport 9100 ! -s 192.168.1.0/24 -j DROP
┌──(root㉿kali)-[/home/yevidu]
└─# iptables -L -v
Chain INPUT (policy DROP 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
12356 1198K ufw-before-logging-input all -- any any anywhere anywhere
12356 1198K ufw-before-input all -- any any anywhere anywhere
187 16664 ufw-after-input all -- any any anywhere anywhere
8 608 ufw-after-logging-input all -- any any anywhere anywhere
8 608 ufw-reject-input all -- any any anywhere anywhere
8 608 ufw-track-input all -- any any anywhere anywhere
0 0 ACCEPT all -- any any anywhere anywhere ctstate RELATED,ESTABLISHED
0 0 ACCEPT tcp -- any any anywhere anywhere tcp dpt:http
0 0 ACCEPT tcp -- any any anywhere anywhere tcp dpt:https
0 0 ACCEPT tcp -- any any 192.168.1.10 anywhere tcp dpt:ssh
0 0 ACCEPT tcp -- any any 192.168.1.0/24 anywhere tcp dpt:ssh
0 0 ACCEPT tcp -- any any 192.168.1.10 anywhere tcp dpt:ssh
0 0 ACCEPT tcp -- any any anywhere anywhere

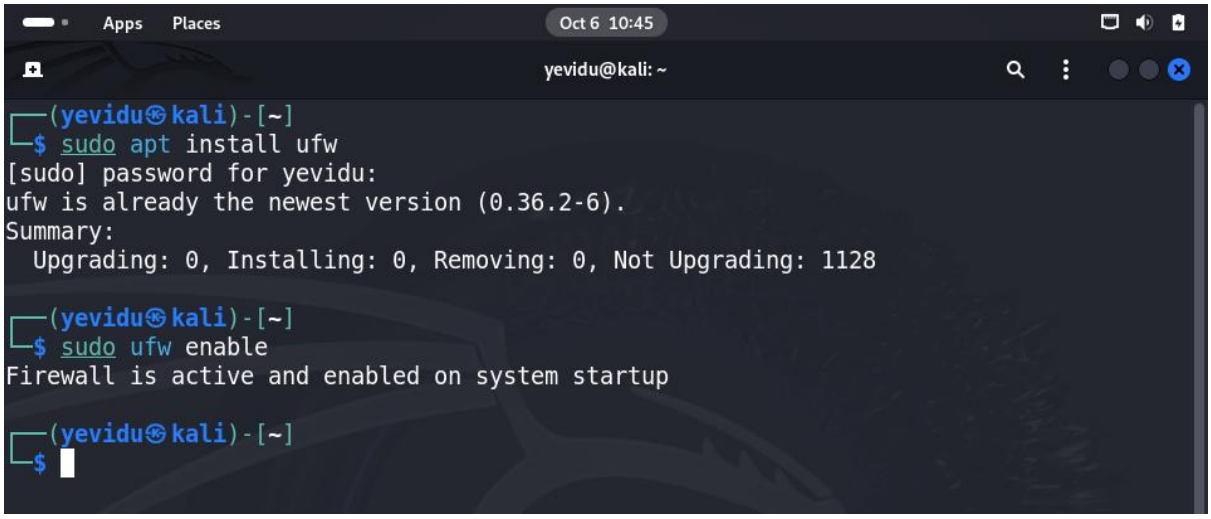
4.Best Practices.

4.1). Implementing a Firewall in Kali Linux.

Installing a firewall in Kali Linux is essential for protecting your system from threats and illegal access. This manual offers a methodical way to putting the Uncomplicated Firewall (UFW), which streamlines firewall rule management, into practice.

First need to install UFW if it's not already present on Linux.

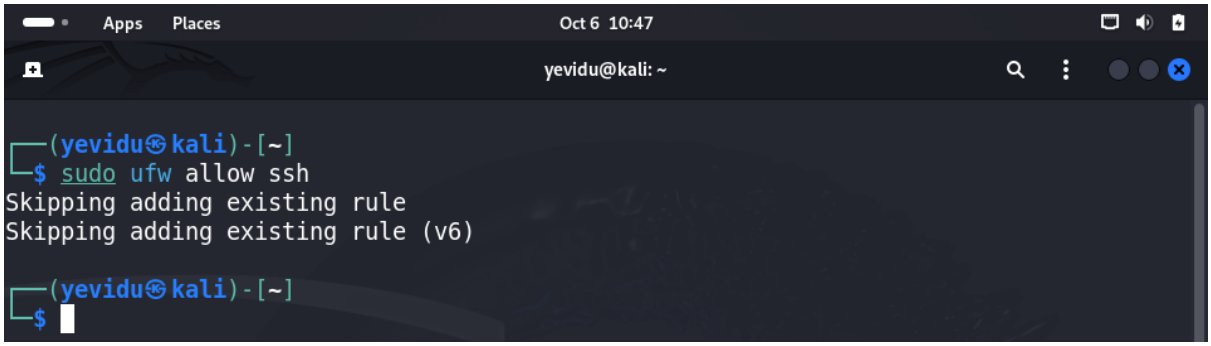
After installed, enable UFW to start protecting Linux.



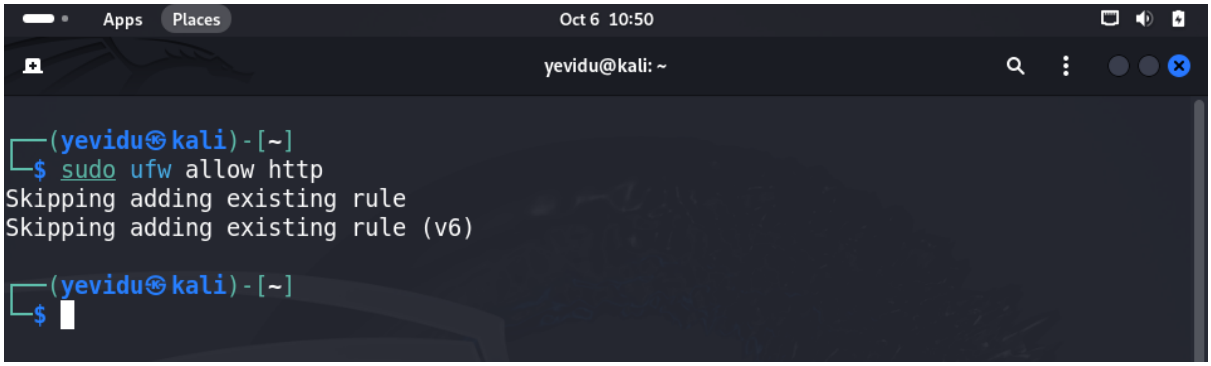
To improve security, set default rules for all incoming and outgoing traffic.

- **Deny Incoming:** This prevents all incoming connections unless explicitly allowed, reducing the risk of unauthorized access.
- **Allow Outgoing:** This permits all outgoing connections, which is generally safe for applications that need internet access.

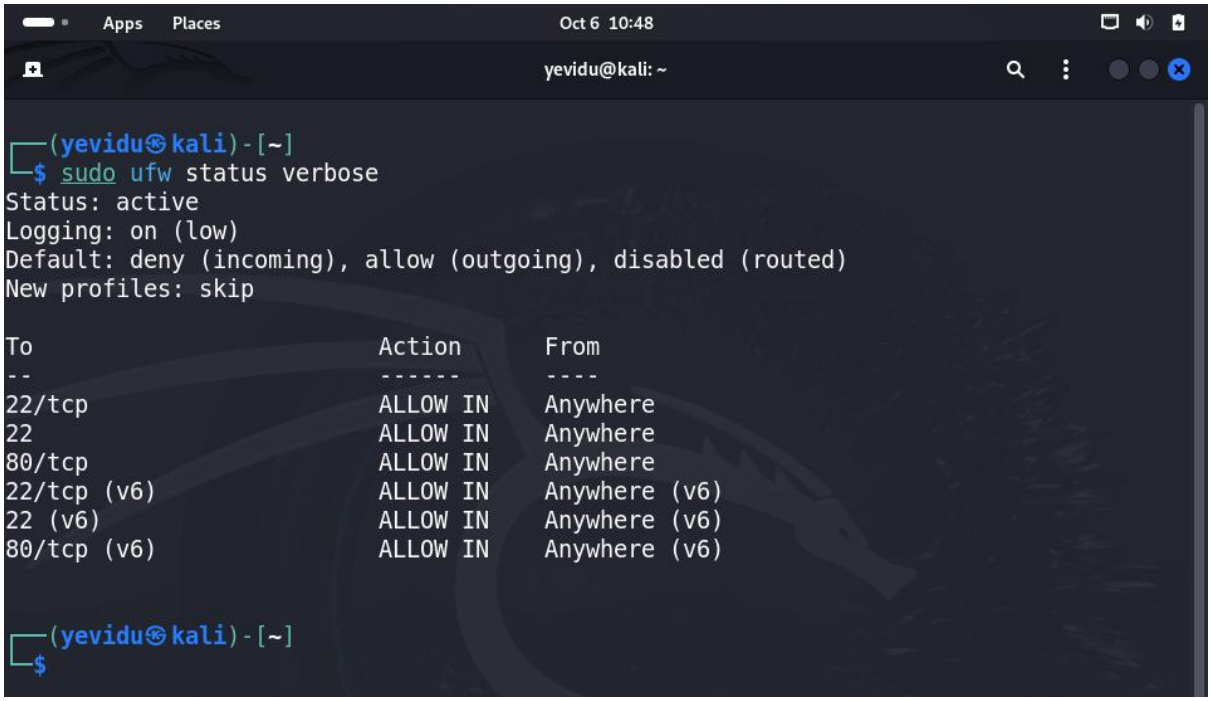
Then allow specific services through the firewall.



Allow HTTP traffic.

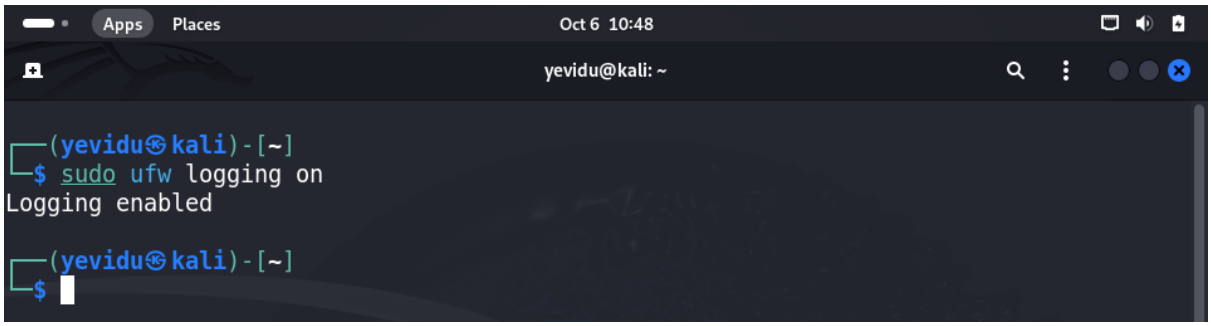


Check and verify the status of UFW and see whether which rules are active.

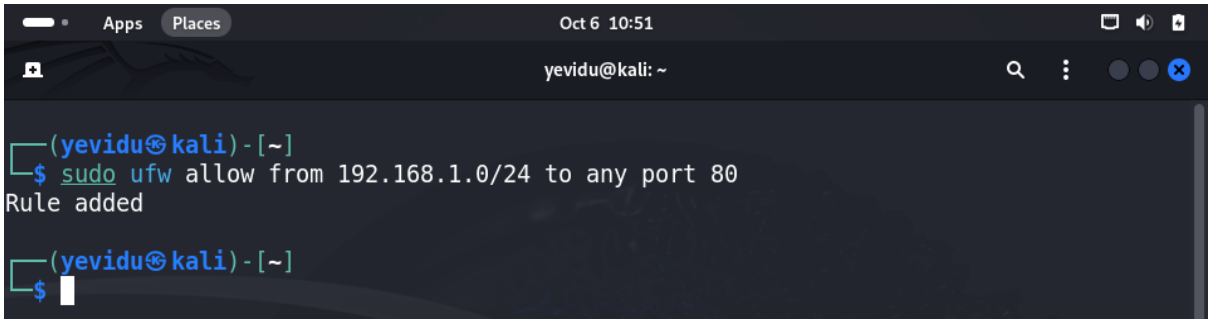


Enable logging.

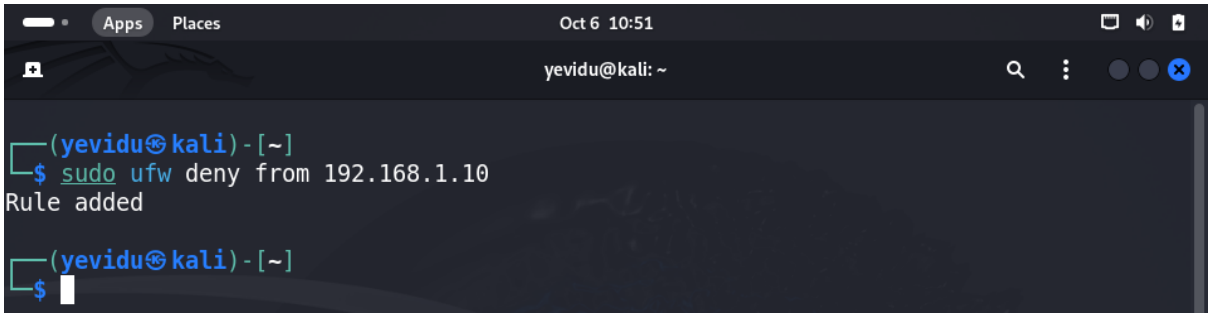
It monitor traffic and detect unauthorized access attempts.



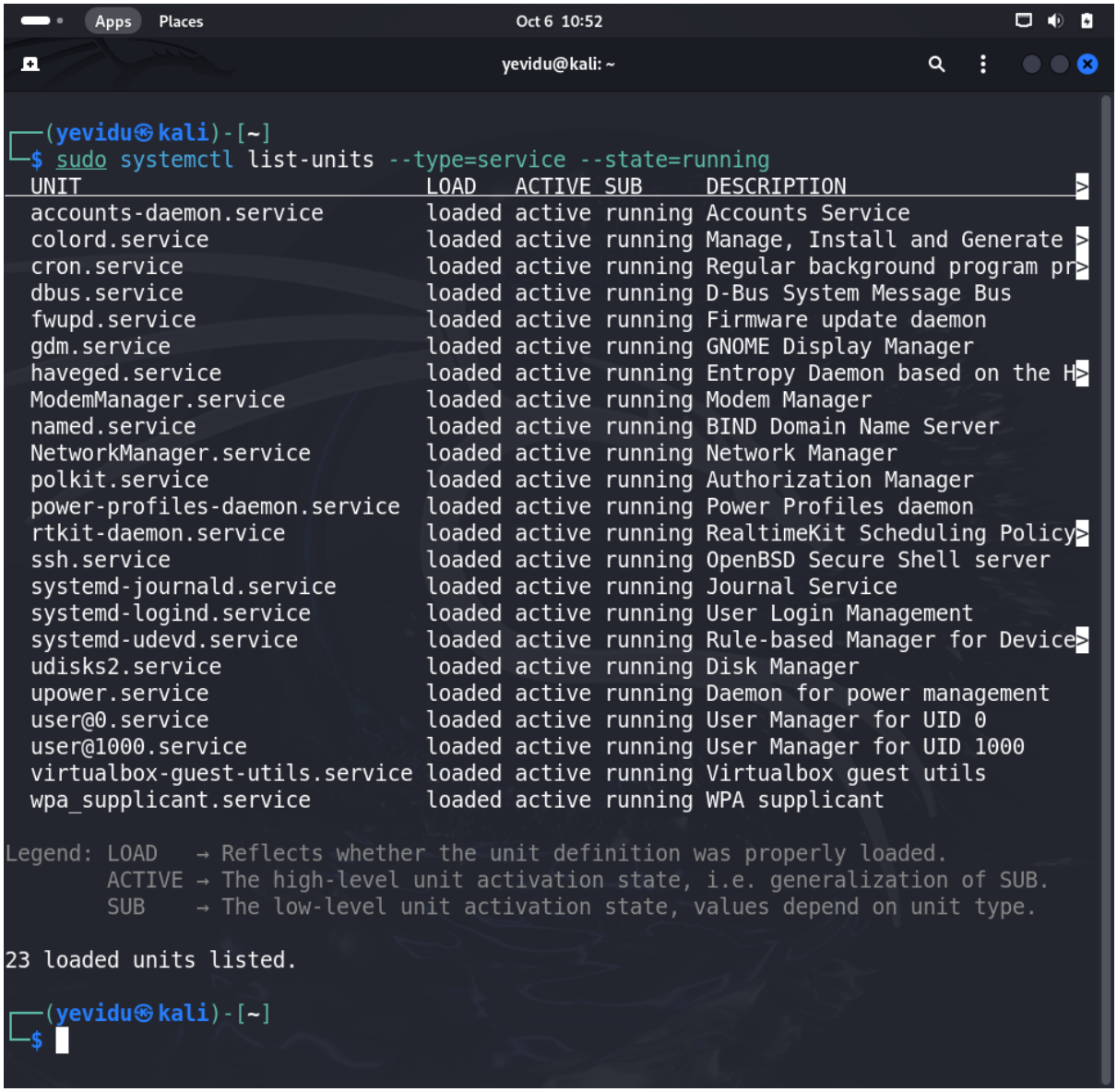
Your firewall can be further customized by permitting or prohibiting IP addresses or ranges. To permit access, for example, from a particular IP address



Deny access from a specific IP address.

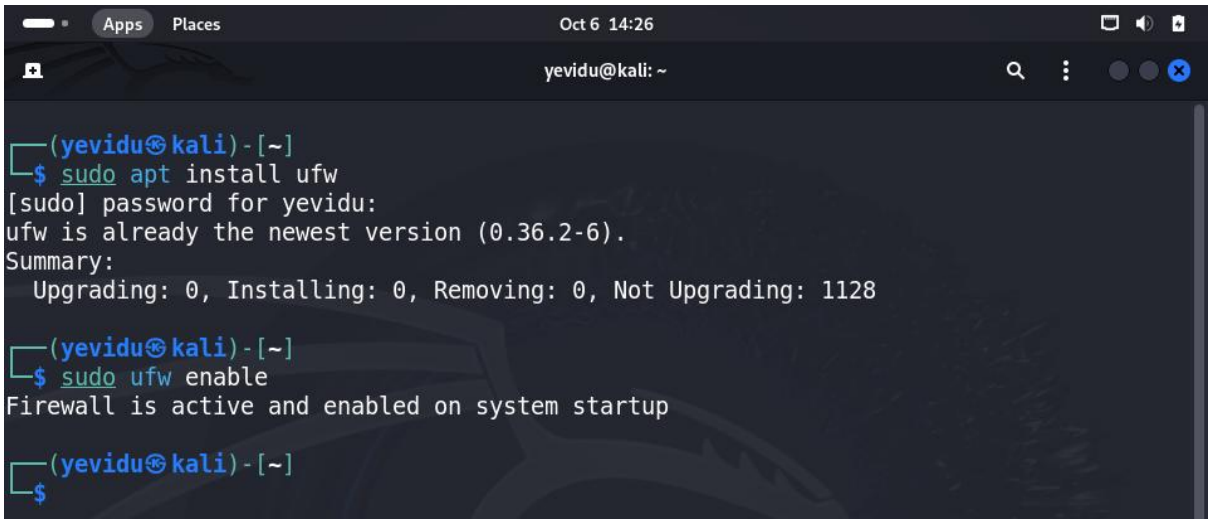


Now check active services using.



4.2). Minimize Open Ports and Services.

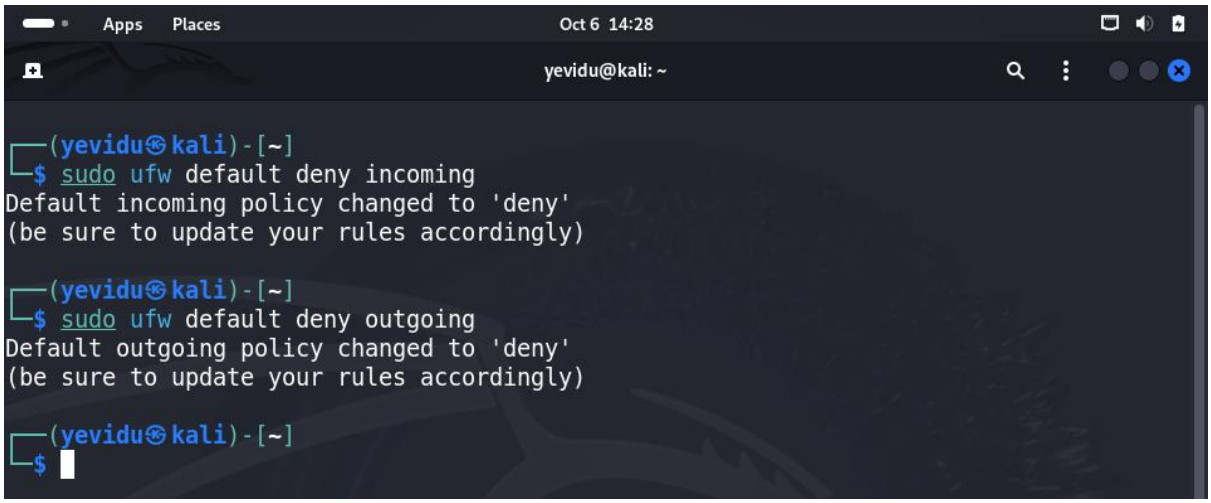
First install and Enable UFW.



Then Set Default Policies.

Deny Incoming Connections: This prevents all incoming connections by default.

Allow Outgoing Connections: This allows all outgoing connections.



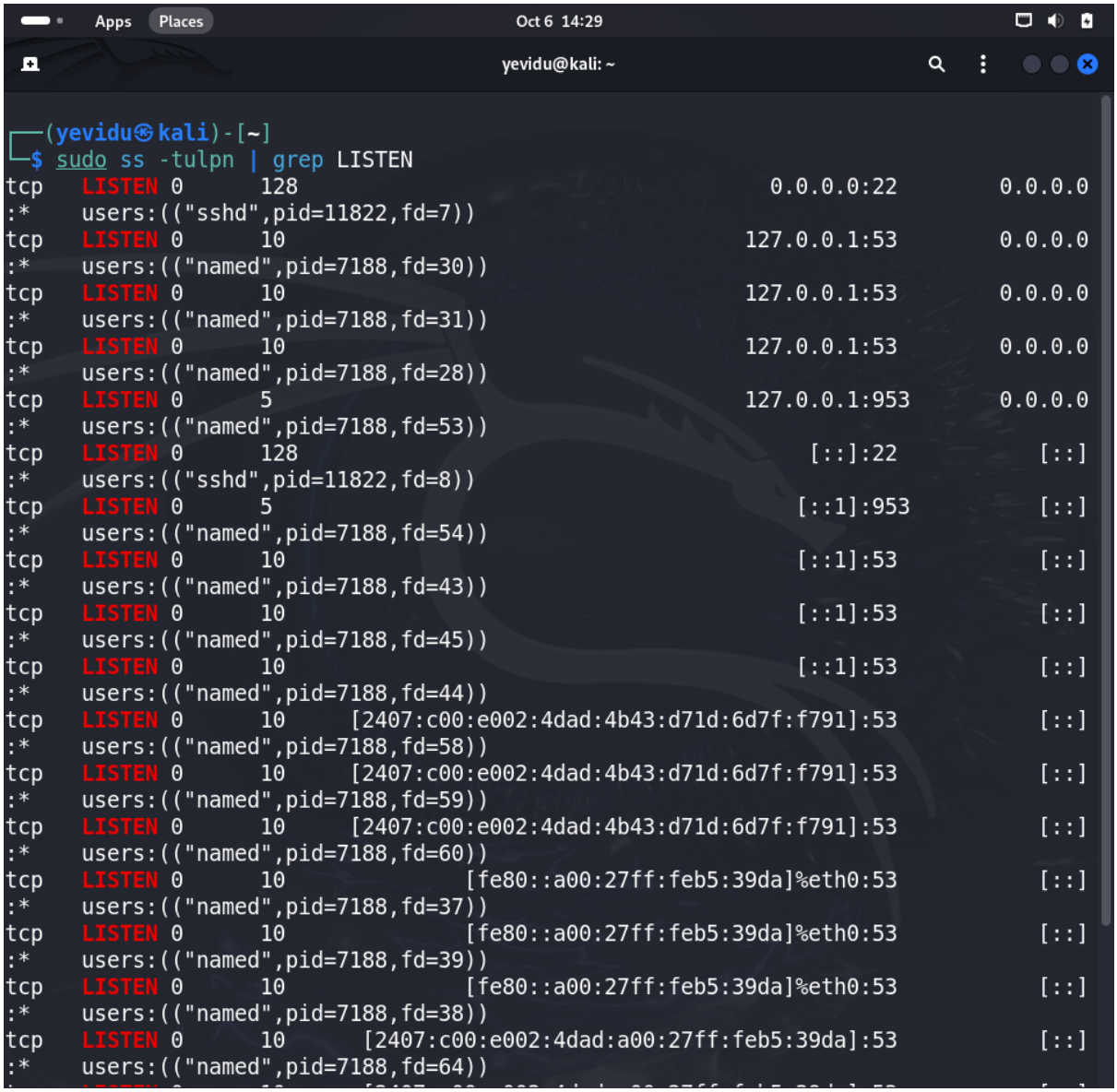
Allow SSH Access.

Then Open Other Specific Ports,

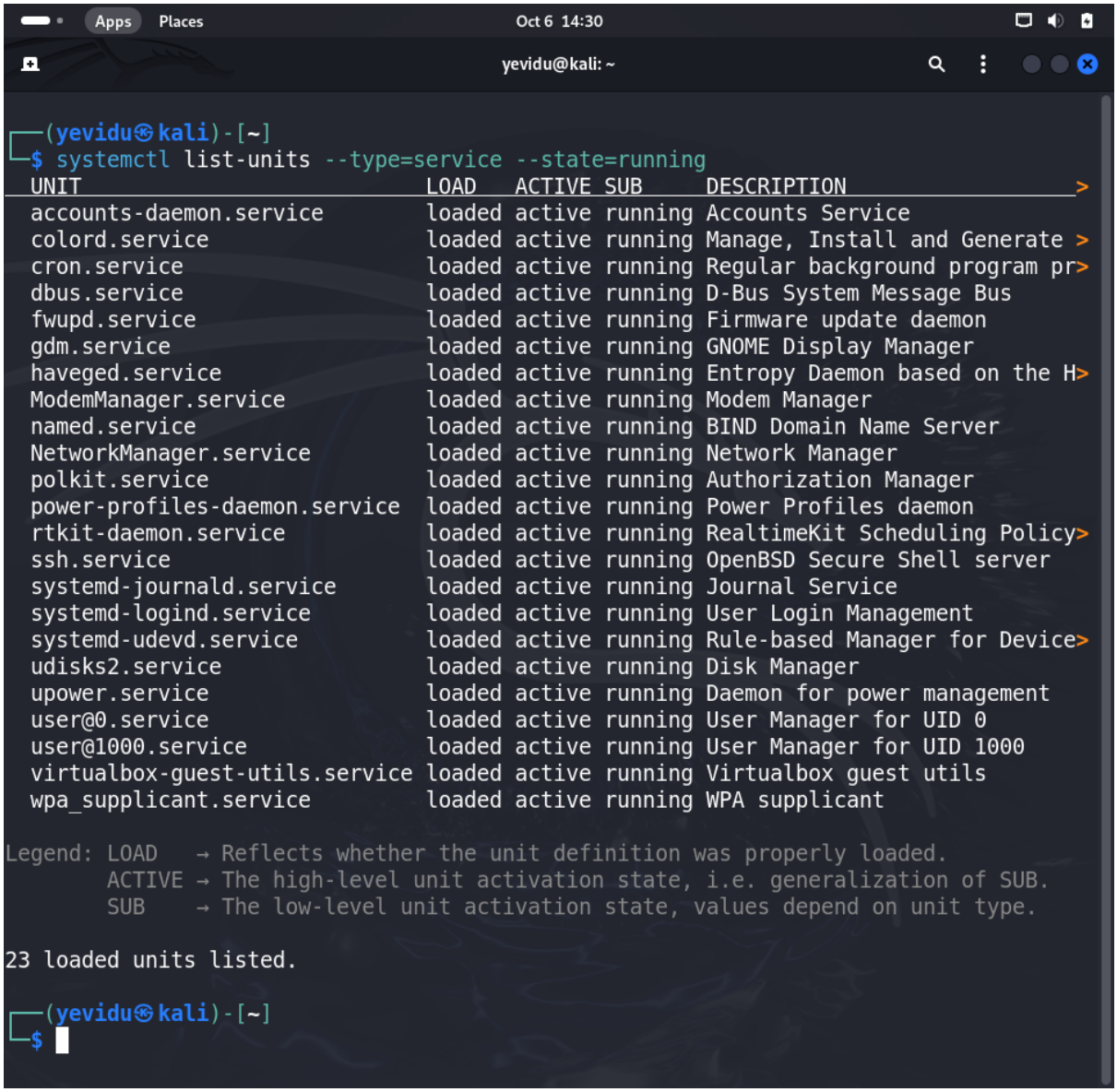
HTTP (port 80) ; **sudo ufw allow 80/tcp**

HTTPS (port 443) ; **sudo ufw allow 443/tcp**

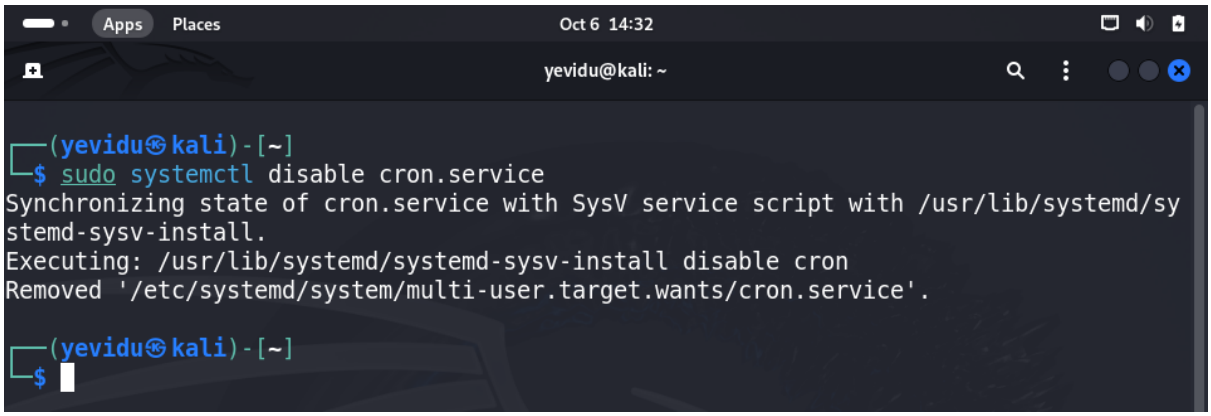
Check Open Ports.



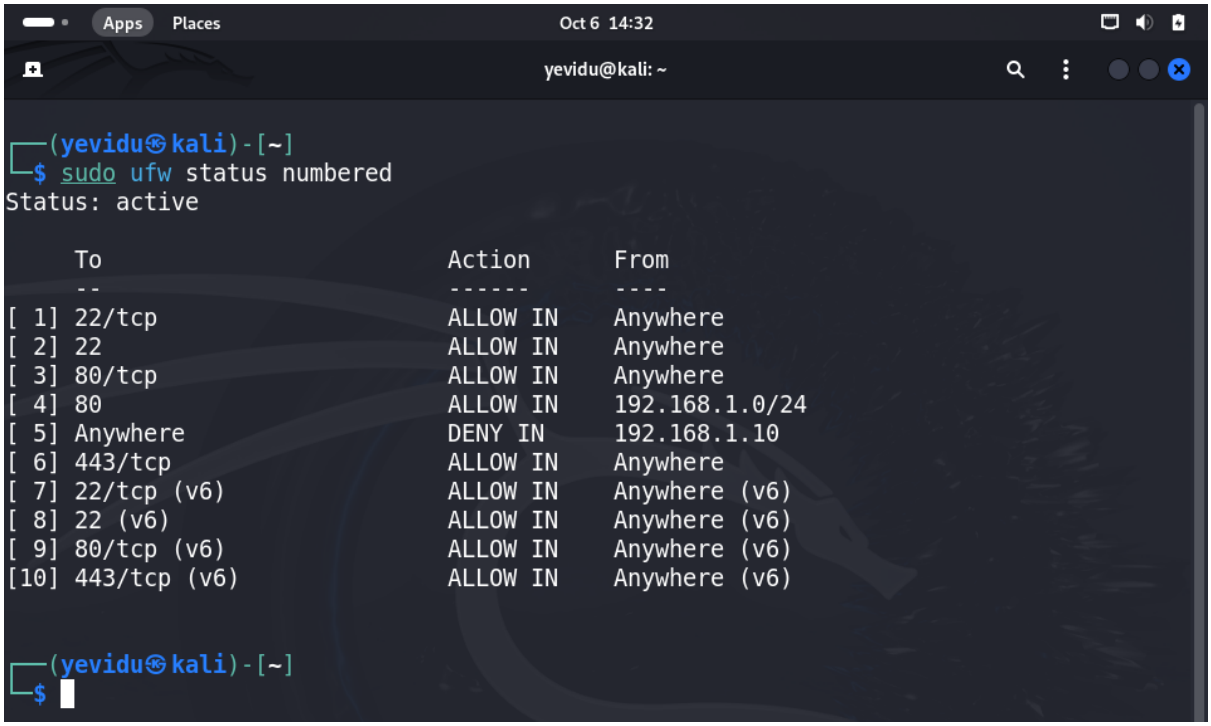
Identify Running Services and list all.



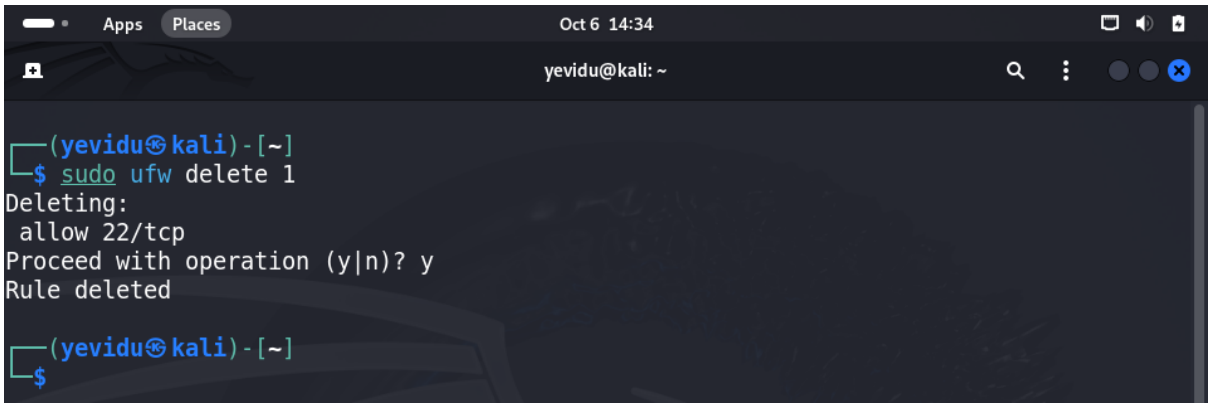
Disable Unused Services.



Regularly review the firewall status and rules.

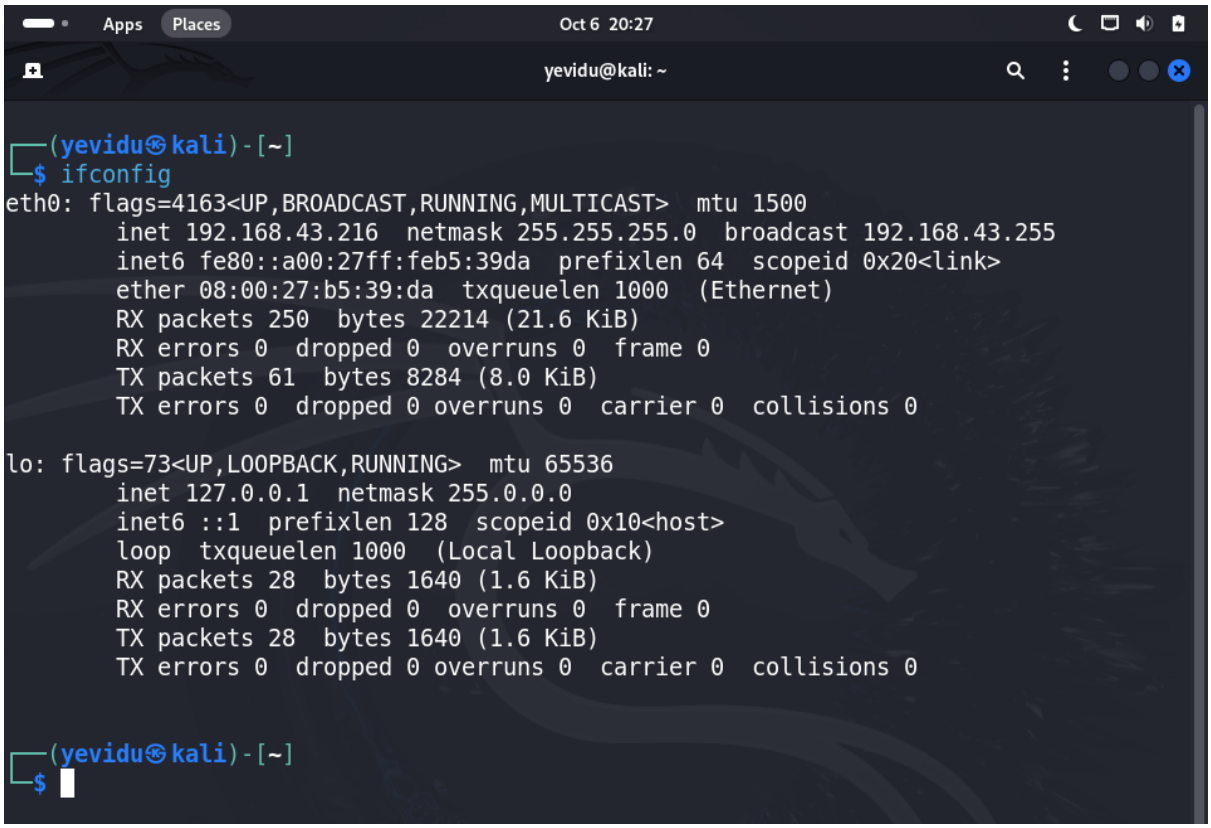


If you find any unnecessary rules, delete them by their number.

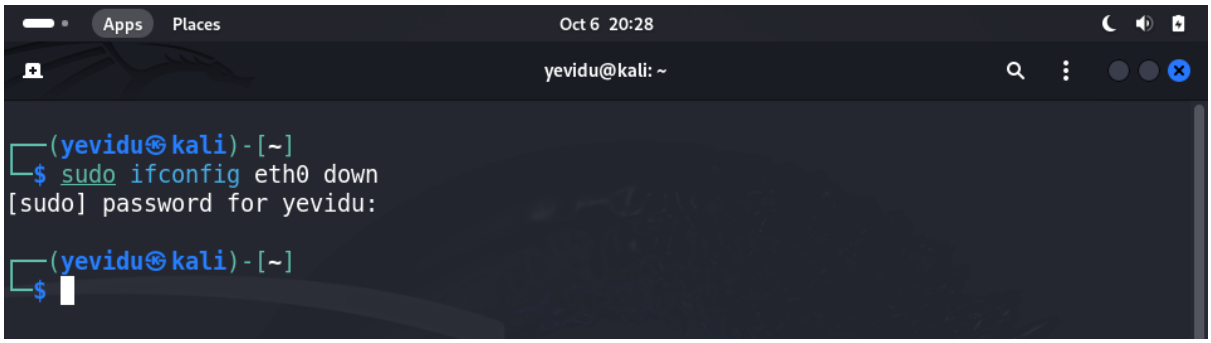


4.3). Disable Unused Network Interface.

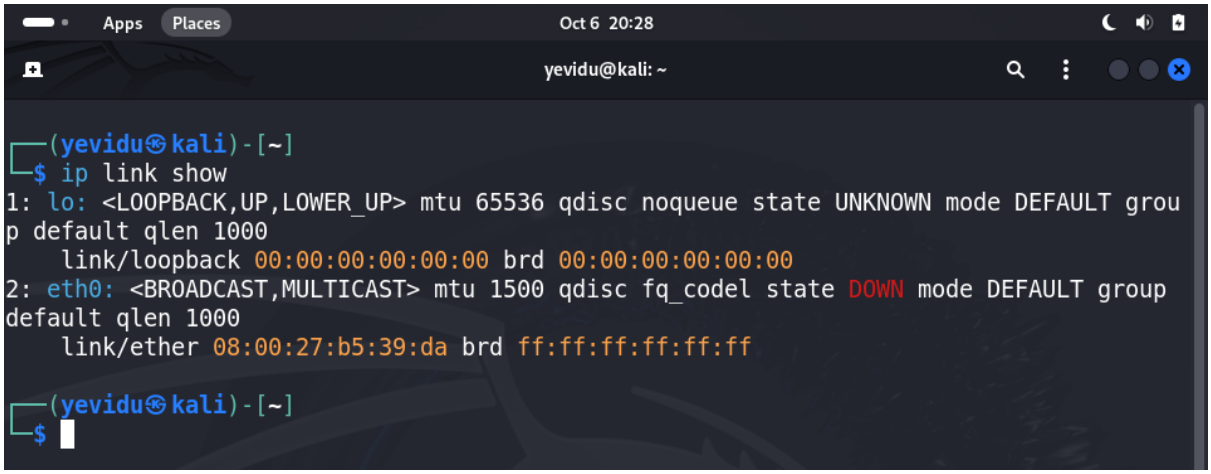
First view the list of all active network interfaces.



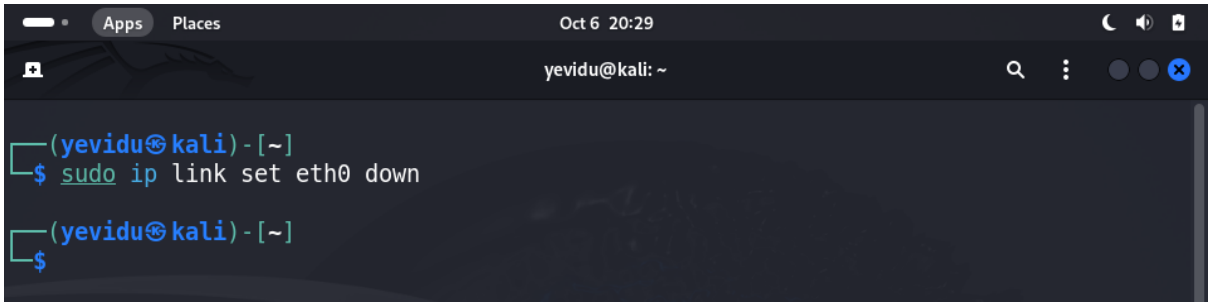
Disable the interface with name of the interface that you wish to disable.



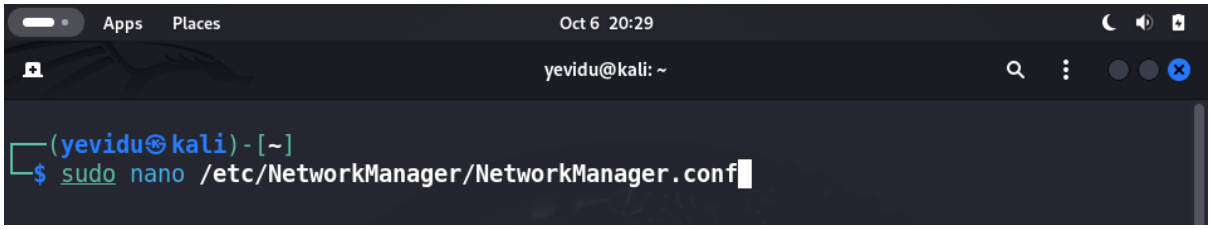
Check the current network interfaces with.



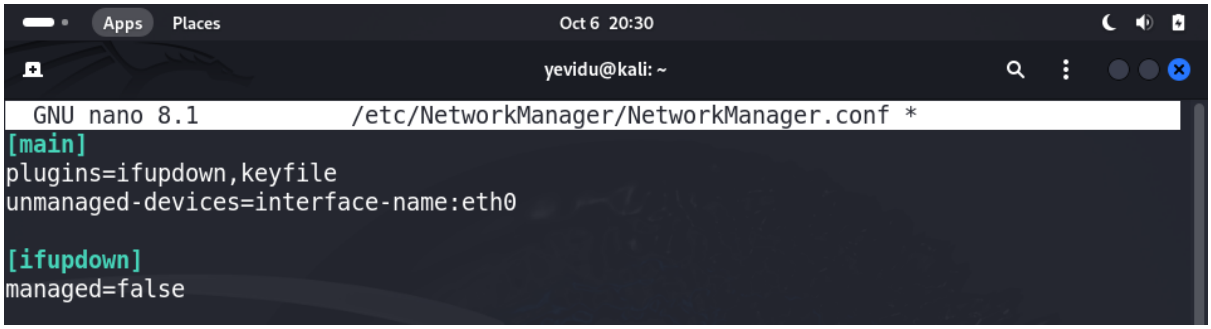
Again, replace eth0 with your target interface name.



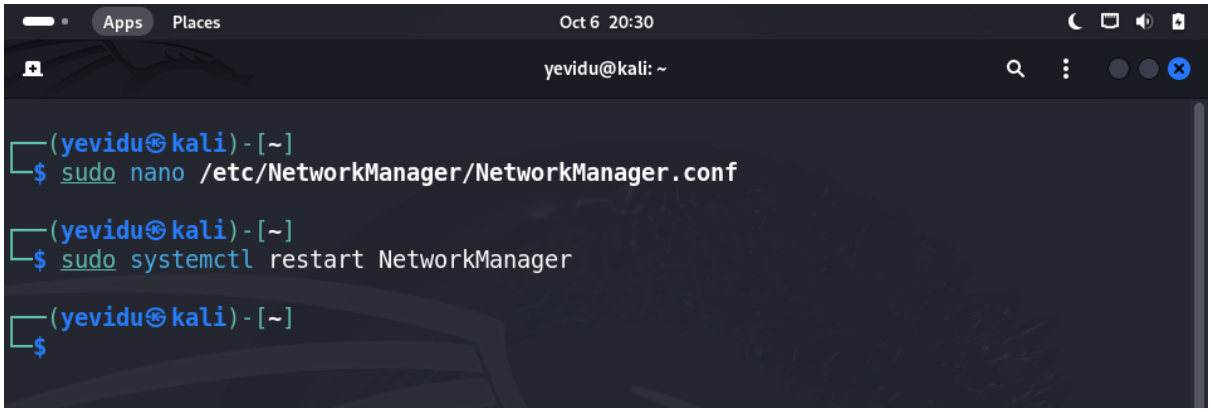
Open the configuration file with a text editor.



Edit the file. Under the [main] section, add or modify the line to include your interface.



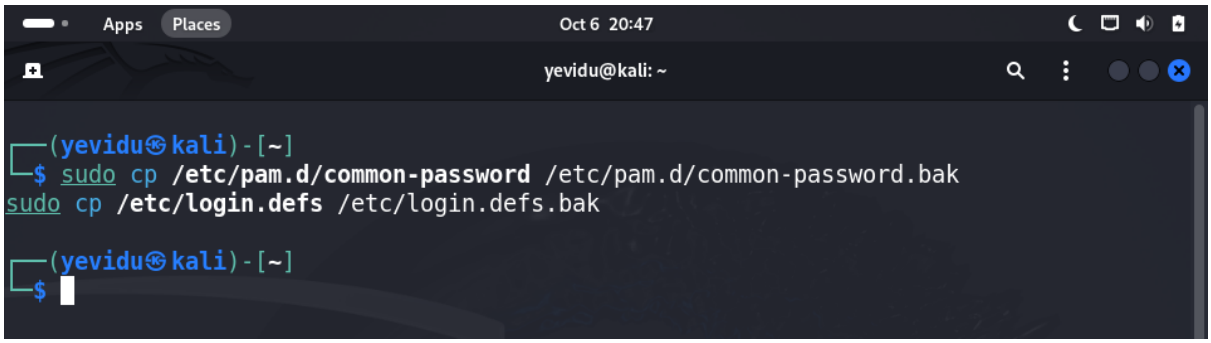
After saving changes, restart NetworkManager.



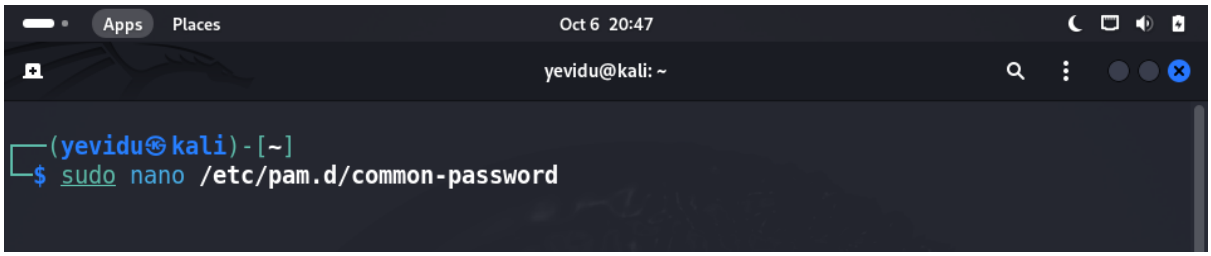
To re-enable a disabled interface, simply reverse the commands.

4.4). Modify the Default Password Policy Settings.

First, make backup relevant to the configuration file.

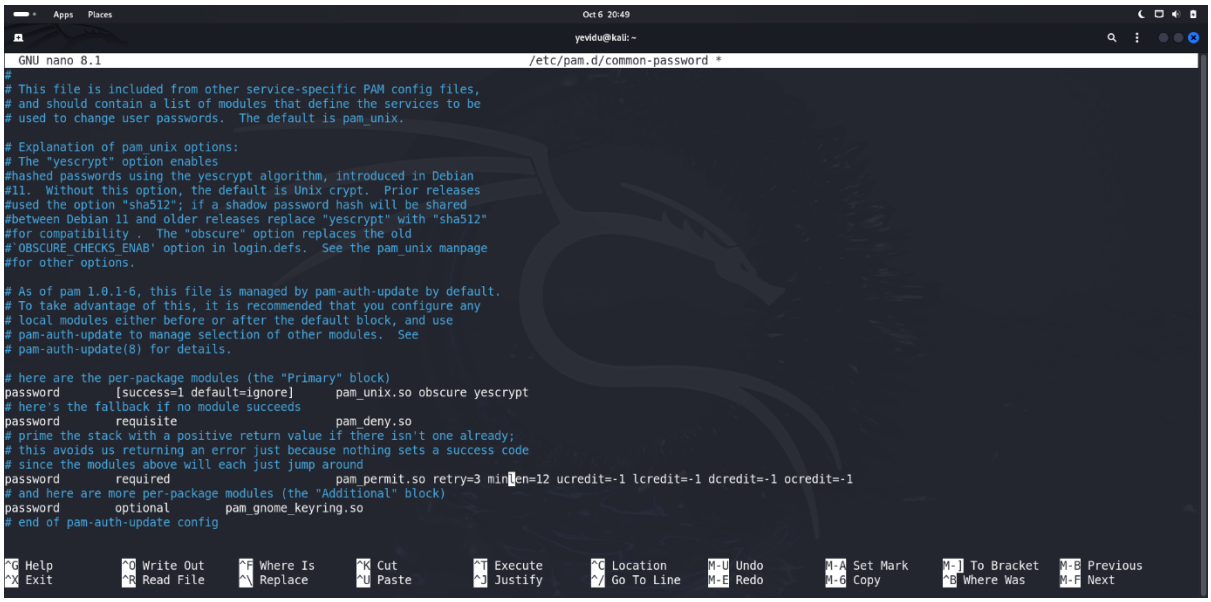


Open the file using a text editor.

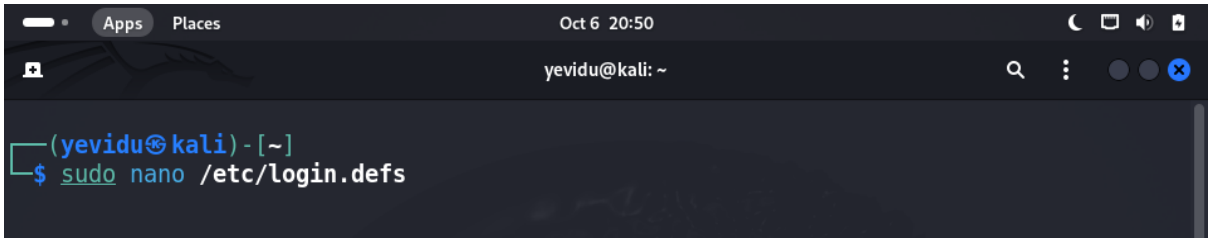


find the line that starts with password requisite pam_pwquality.so or similar and append your desired settings.

Then edit it.



Open the login.defs file.

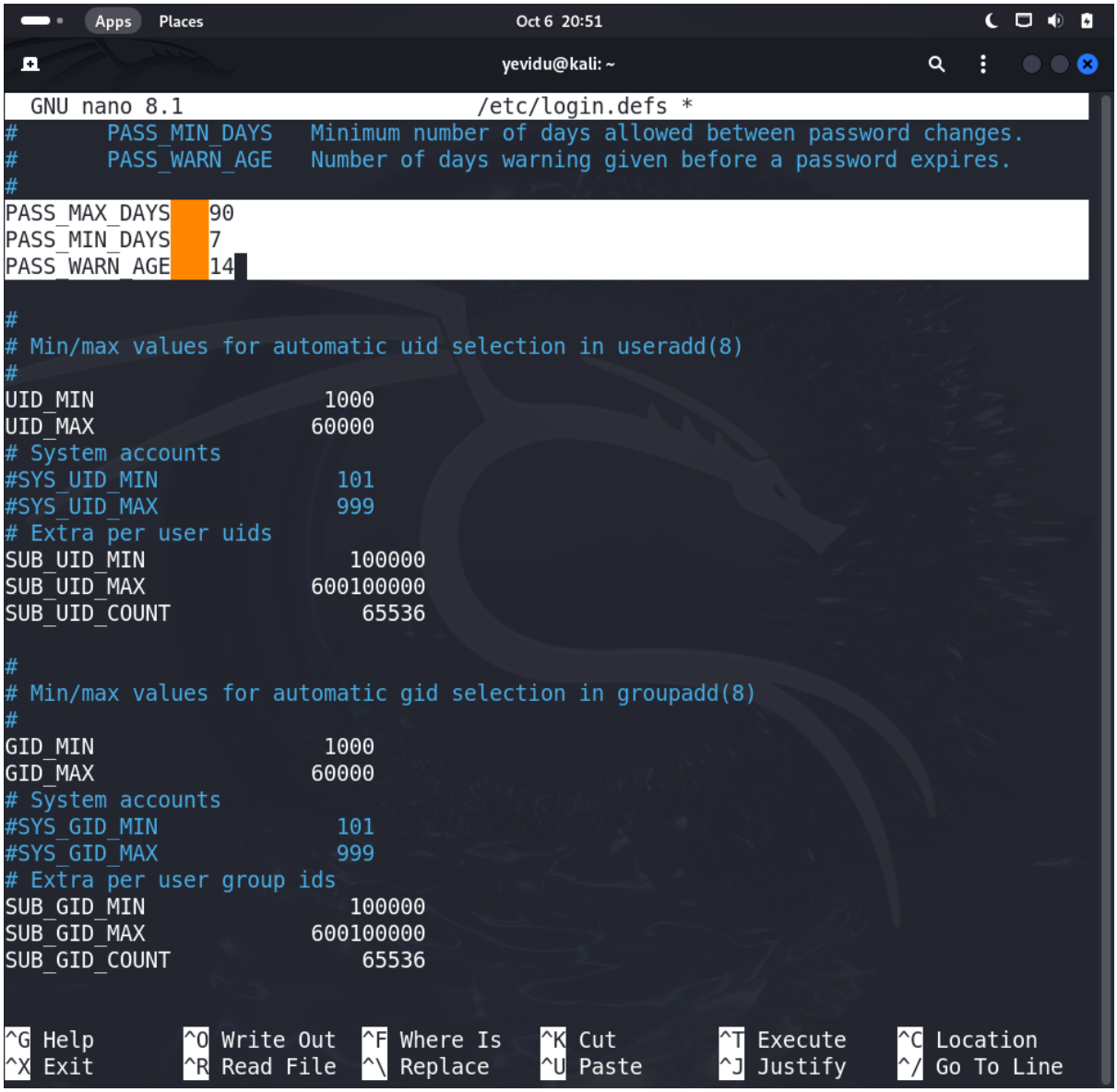


Modify or add the following lines according to your requirements.

PASS_MAX_DAYS 90 # Maximum days before password must be changed

PASS_MIN_DAYS 7 # Minimum days before a password can be changed

PASS_WARN_AGE 14 # Days before expiration to warn user



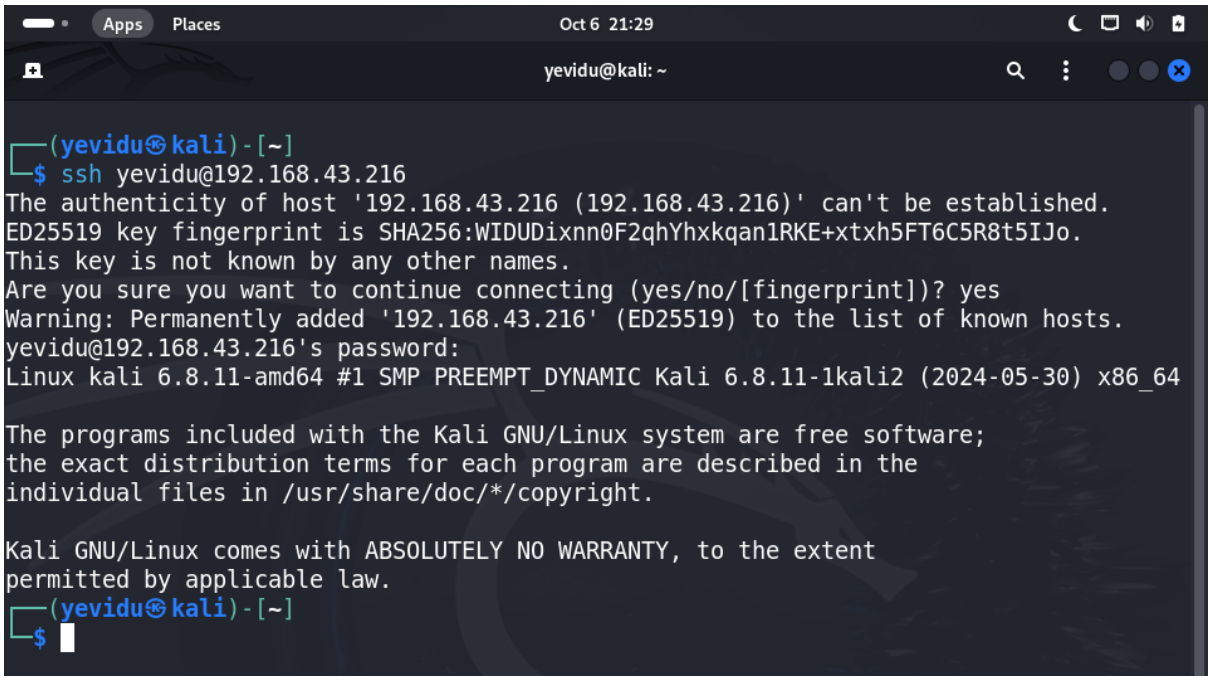
After making all necessary changes in both files, save and exit the text editor.

4.5). Change Default SSH Port.

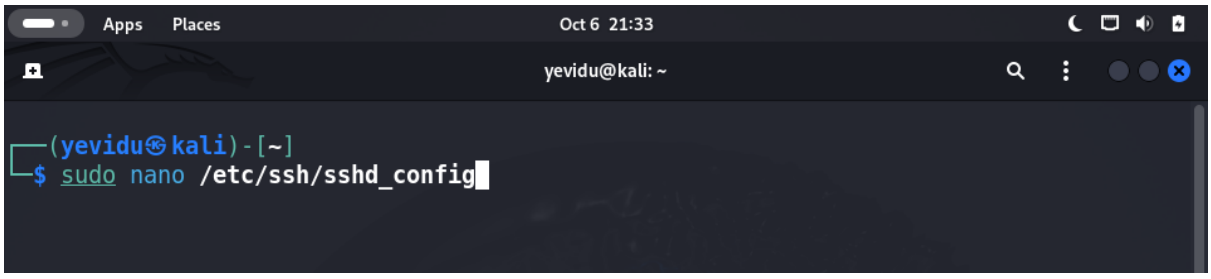
First connect to your server using the following command.

ssh username@server_ip

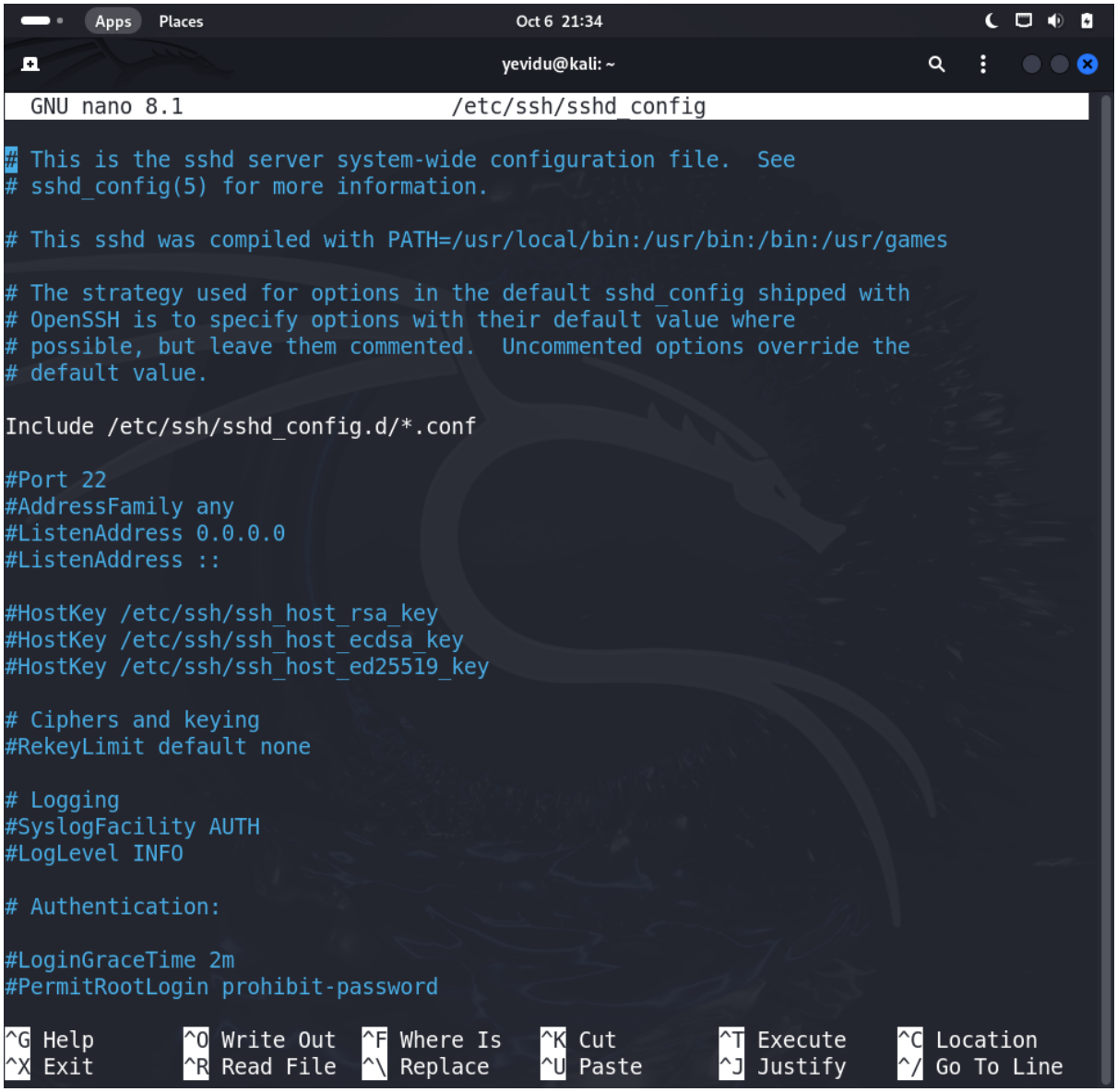
Replace username with your actual username and server_ip with the IP address of your server.

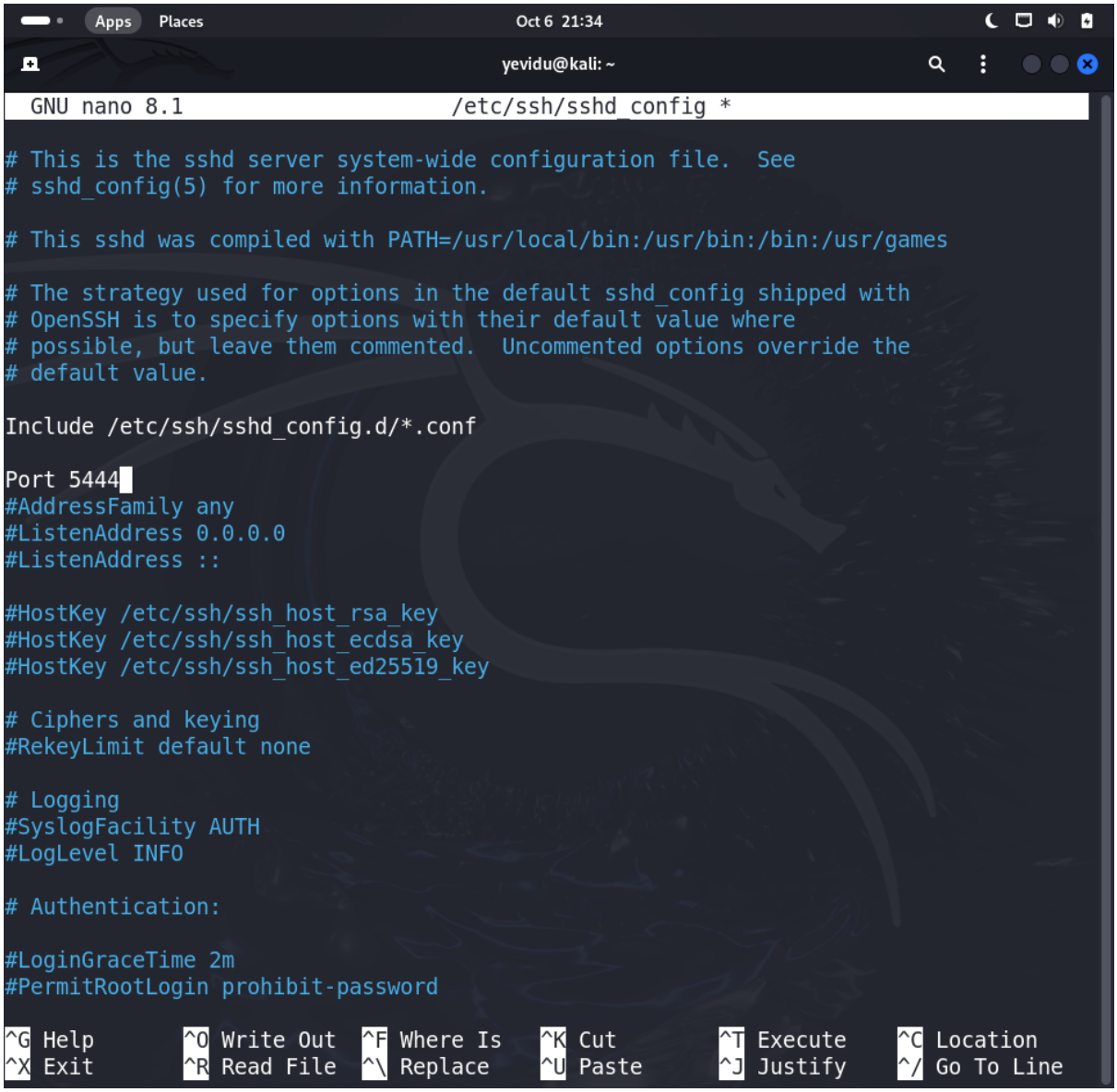


Open the SSH daemon configuration file using a text editor.



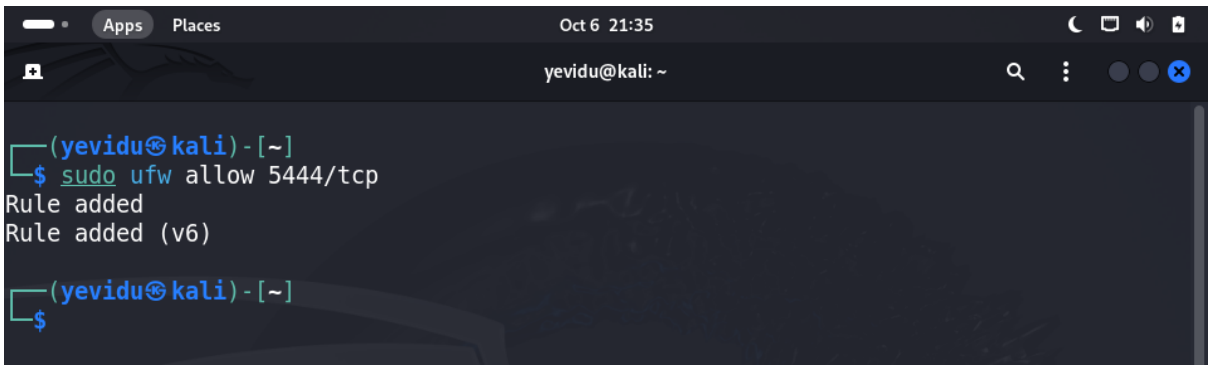
Uncomment this line by removing the # symbol and change 22 to your desired port number.



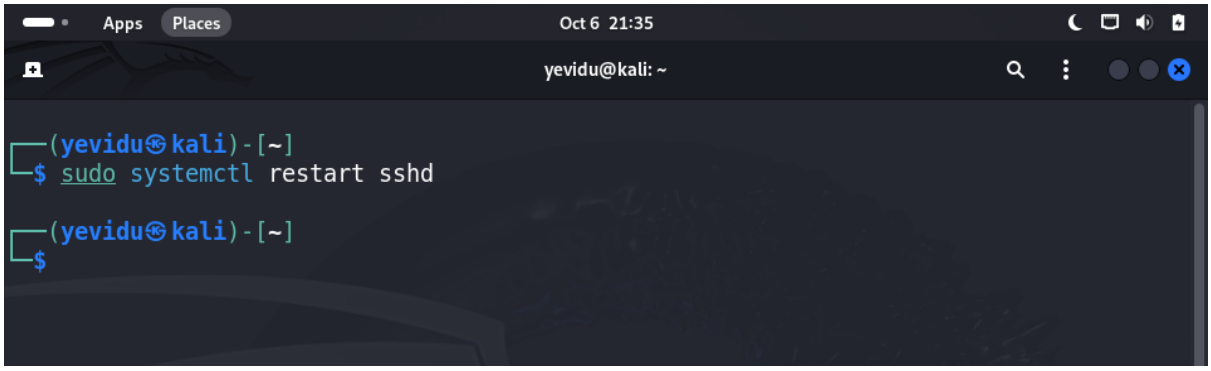


Then save and exit from the editor.

Configure Your Firewall.



Restart the SSH Service.



Test the New Port.