Dated: 25 January,2024

Information Security

Lab Assignment 2

“Ipconfig and ifconfig(linux) commands used to see the IP’s connect to Operating System”
“Ipconfig /all”

Private IP Address:
A private IP address is an address used within a private network, such as a local area network (LAN) or an internal network within an organization. These addresses are not routable on the public internet and are reserved for use within private networks to facilitate communication between devices within the same network. Private IP addresses are defined by RFC 1918 and include the following ranges:
- 10.0.0.0 to 10.255.255.255 (10.0.0.0/8)
- 172.16.0.0 to 172.31.255.255 (172.16.0.0/12)
- 192.168.0.0 to 192.168.255.255 (192.168.0.0/16)

These addresses are commonly used for devices like computers, printers, routers, and other networked devices within a local network environment.

Public IP Address:
A public IP address is an address that is globally routable on the public internet. It uniquely identifies a device or network on the internet, allowing communication between devices across different networks. Public IP addresses are assigned by Internet Service Providers (ISPs) or network administrators and are used to access resources and services on the internet. These addresses are typically visible to other devices on the internet and can be used to communicate with servers, websites, and other internet-connected devices.

Public IP addresses can be static or dynamic. Static public IP addresses remain the same over time, while dynamic public IP addresses may change periodically, especially for residential internet connections.

When you access a website, for example, your device communicates with the website's server using your public IP address, enabling data exchange between your device and the server over the internet.

(It’s for our device)
A **gateway** in networking is a device or software component that acts as an entry and exit point between two different networks, allowing them to communicate with each other. Here's a simple breakdown:

Function: The gateway facilitates communication between devices on different networks by forwarding data packets between them.

Types:
- Default Gateway: This is the gateway through which devices on a local network connect to devices on external networks, such as the internet. It typically has an IP address assigned within the local network.

- Router: A common example of a gateway is a router, which connects a local network to the internet. It manages traffic between devices within the local network and devices on the internet.
- Firewall: Gateways can also include firewall functionality, providing security by filtering incoming and outgoing traffic based on predefined rules.
- Proxy Server: In some cases, a gateway can be a proxy server that acts as an intermediary between clients and servers, caching and filtering web content.

IP Address: Gateways have their own IP address, which is used by devices on the local network to send data outside the network.

Routing: Gateways use routing tables to determine the best path for forwarding data packets to their intended destinations. They may employ protocols such as IP routing (for IPv4) or ICMPv6 routing (for IPv6) to manage routing decisions.

In summary, a gateway serves as a bridge between different networks, enabling communication and traffic management between them.

Registry.in - Domain naim registration
Google Public DNS Server IP - 8.8.8.8
If i want to change the server then
Nslookup
> server 8.8.8.8
> website name

- **DNCP server provides IP to Machines**
- **Lease Expired and Lease Time**
- **DNS Server translates domain names to IP address (DNS Client is available to all devices)**
- **A MAC (Media Access Control) address, also known as a hardware or physical address, is a unique identifier assigned to network interfaces for communication on a network.**

1. Security Concepts

Security and OS Hardening:

Security: In the realm of computing, security refers to measures taken to protect systems, data, and networks from unauthorized access, misuse, or damage. It encompasses various techniques, protocols, policies, and tools designed to ensure confidentiality, integrity, and availability of resources.

OS Hardening: OS hardening involves implementing security measures to reduce the vulnerability of an operating system (OS) by eliminating potential attack vectors, minimizing the system's exposure to threats, and enhancing its resistance to unauthorized access and exploitation.

Differentiation between Windows and Linux OS Architecture:

Windows OS: Windows is a proprietary operating system developed by Microsoft. Its architecture is primarily based on a monolithic kernel design, where the kernel provides extensive services and runs in kernel mode, handling system calls, memory management, and hardware interactions directly.

Linux OS: Linux, on the other hand, is an open-source operating system kernel developed by Linus Torvalds and a large community of contributors. Linux follows a modular design with a monolithic kernel at its core but allows for dynamic loading and unloading of kernel modules. It also supports various kernel architectures, making it highly customizable and adaptable to different hardware platforms.

Importance of Linux Security:

- Linux is widely used in servers, embedded systems, and critical infrastructure due to its stability, performance, and open-source nature.
- Many internet-facing services and websites are hosted on Linux servers, making them prime targets for cyber attacks.
- As an open-source platform, Linux benefits from rapid security patching and community-driven development, but this also means vulnerabilities are quickly disclosed, requiring proactive security measures.
- Securing Linux systems is crucial to protect sensitive data, ensure uninterrupted service availability, and safeguard against cyber threats such as malware, ransomware, and unauthorized access.

Types of Security Breach:

1. Unauthorized Access: This occurs when an attacker gains entry to a system or network without proper authorization, often through exploiting vulnerabilities or using stolen credentials.

2. Data Breach: A data breach involves unauthorized access, disclosure, or theft of sensitive or confidential information, such as personal data, financial records, or intellectual property.

3. Malware Infection: Malware, including viruses, worms, trojans, and ransomware, can compromise system security by infecting and manipulating files, stealing data, or disrupting normal operations.

4. Denial of Service (DoS): DoS attacks aim to disrupt or disable services, applications, or networks by overwhelming them with excessive traffic, requests, or malicious packets, rendering them inaccessible to legitimate users.

5. Social Engineering: Social engineering techniques exploit human psychology to manipulate individuals into divulging confidential information, clicking on malicious links, or performing actions that compromise security.

6. Insider Threats: Insider threats involve individuals with authorized access to systems or networks intentionally or unintentionally abusing their privileges, stealing data, or sabotaging operations.

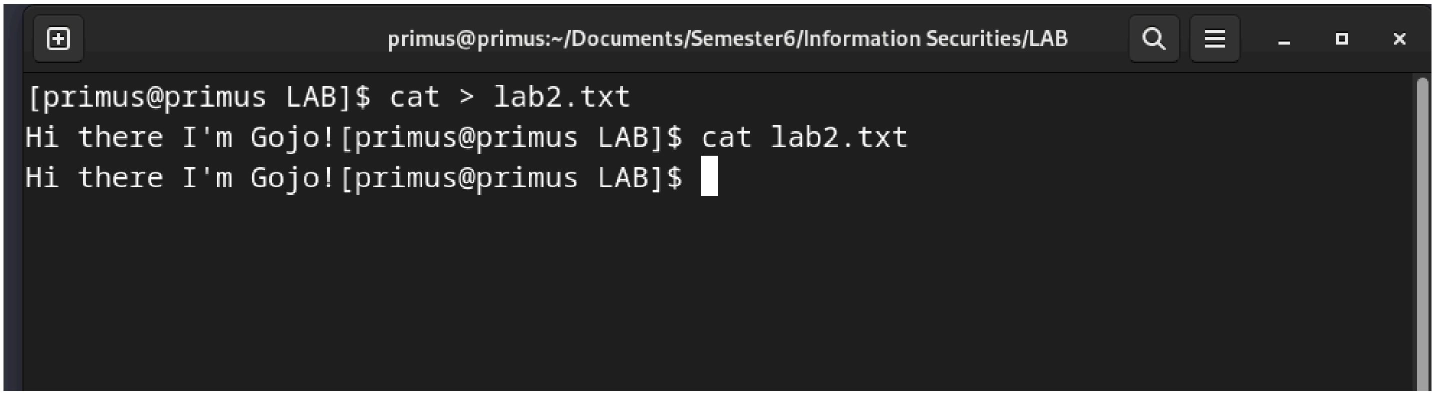
7. Phishing: Phishing attacks use deceptive emails, messages, or websites to trick users into revealing sensitive information, such as login credentials or financial details, or downloading malicious software.

2. Linux Commands

a. File Commands:

i. Creating Files:

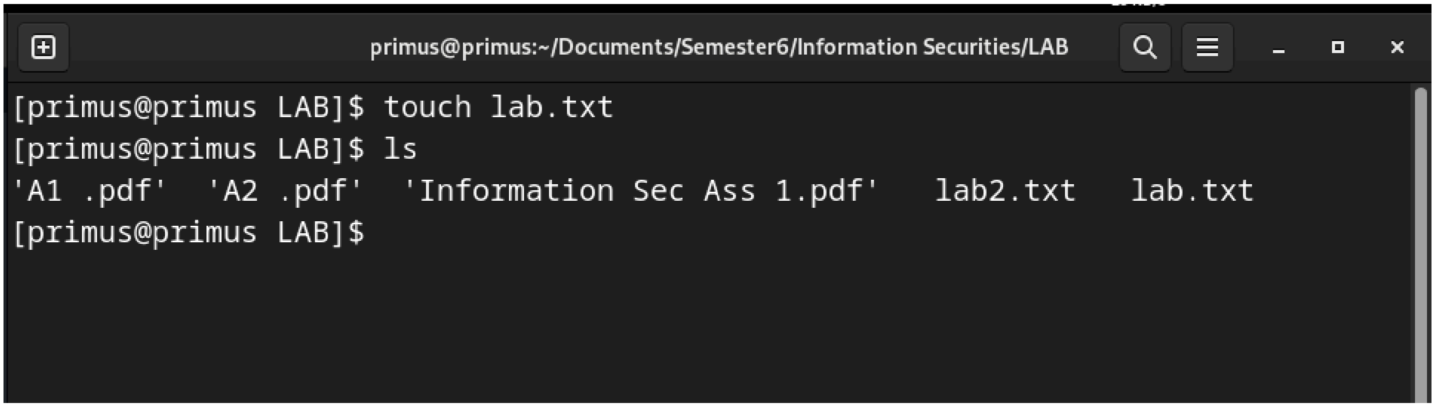
1. cat: Creates a file or concatenates files. Example: `cat > filename`



primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ cat > lab2.txt
Hi there I'm Gojo![primus@primus LAB]$ cat lab2.txt
Hi there I'm Gojo![primus@primus LAB]$ 
```

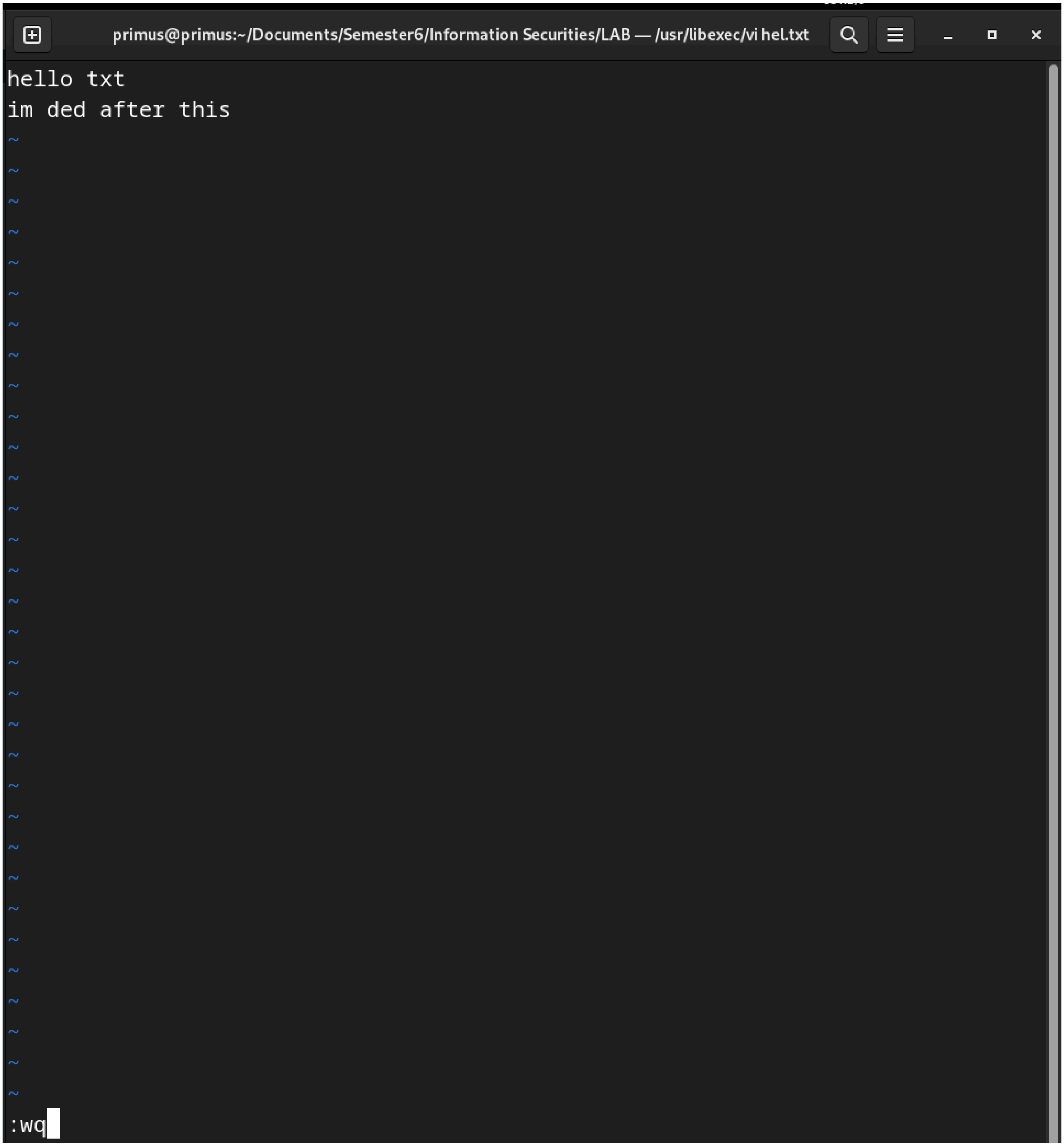
2. touch: Creates an empty file or updates the access and modification times of an existing file. Example: `touch filename`



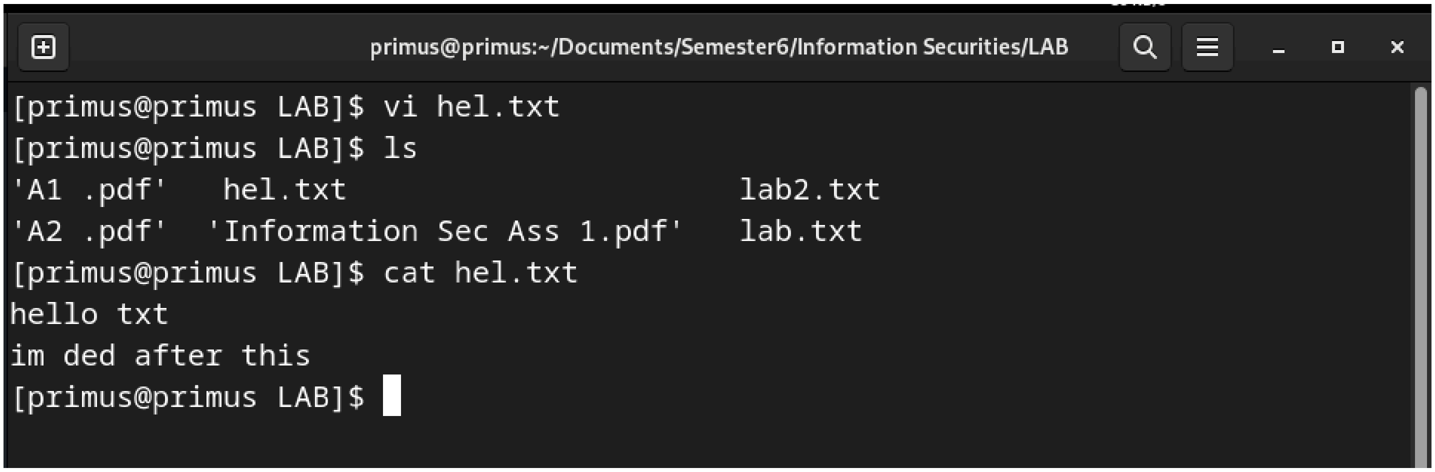
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ touch lab.txt
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'  'Information Sec Ass 1.pdf'   lab2.txt   lab.txt
[primus@primus LAB]$
```

3. vi/vim: Text editors available in all Linux distributions. Example: `vi filename` or `vim filename`

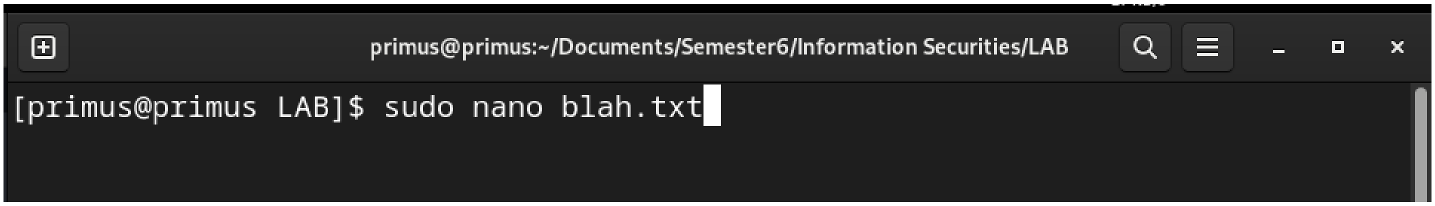

primus@primus:~/Documents/Semester6/Information Securities/LAB — /usr/libexec/vi hel.txt

```
hello txt
im ded after this
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

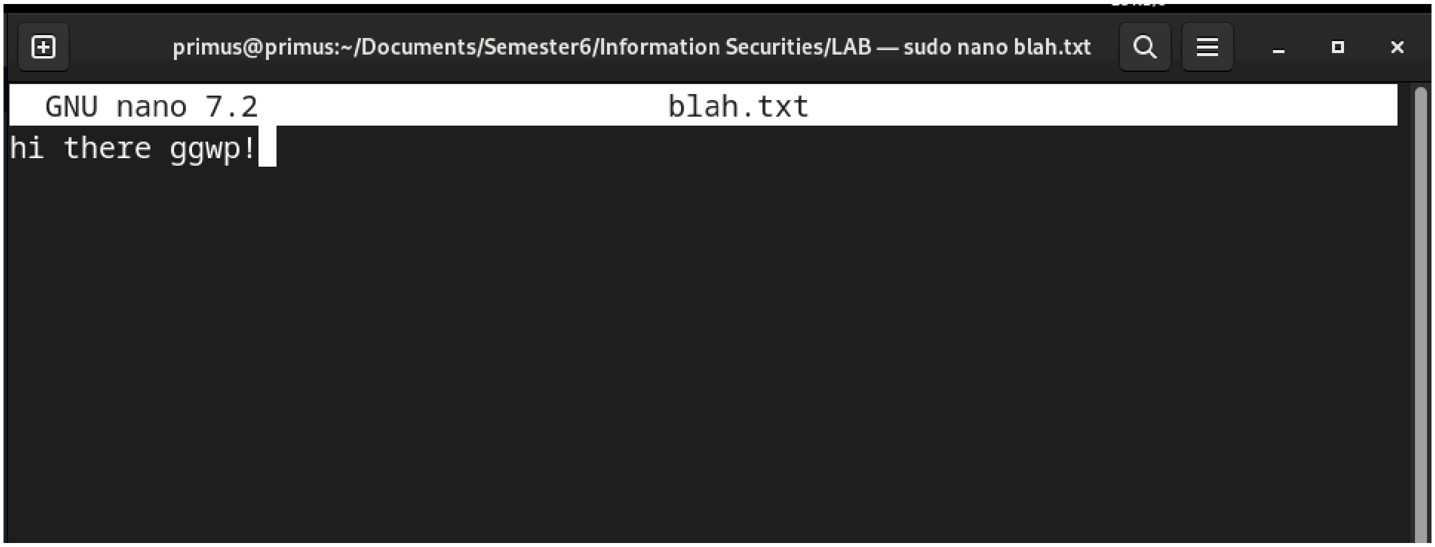

primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ vi hel.txt
[primus@primus LAB]$ ls
'A1 .pdf'    hel.txt                       lab2.txt
'A2 .pdf'   'Information Sec Ass 1.pdf'   lab.txt
[primus@primus LAB]$ cat hel.txt
hello txt
im ded after this
[primus@primus LAB]$ 
```

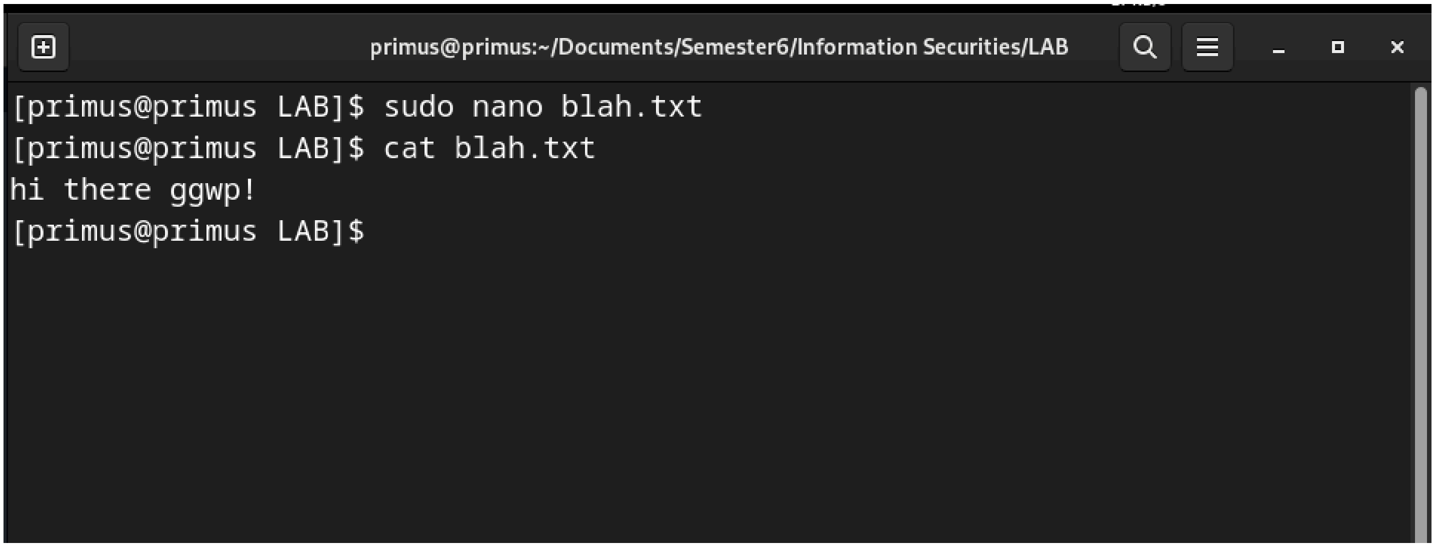
4. nano: A text editor not always built-in but commonly available. Example: `nano filename`


primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ sudo nano blah.txt
```


primus@primus:~/Documents/Semester6/Information Securities/LAB — sudo nano blah.txt

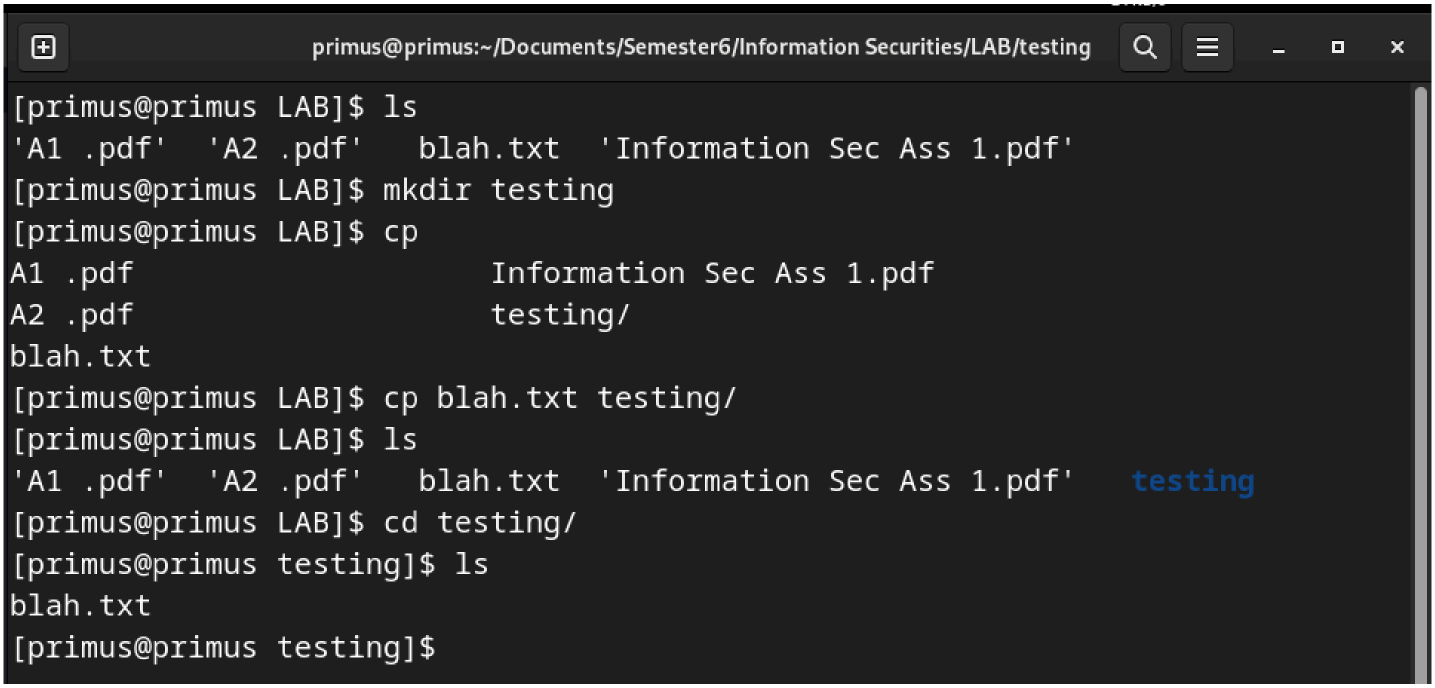
```
  GNU nano 7.2                       blah.txt
hi there ggwp!
```


primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ sudo nano blah.txt
[primus@primus LAB]$ cat blah.txt
hi there ggwp!
[primus@primus LAB]$
```

b. Copying Files:

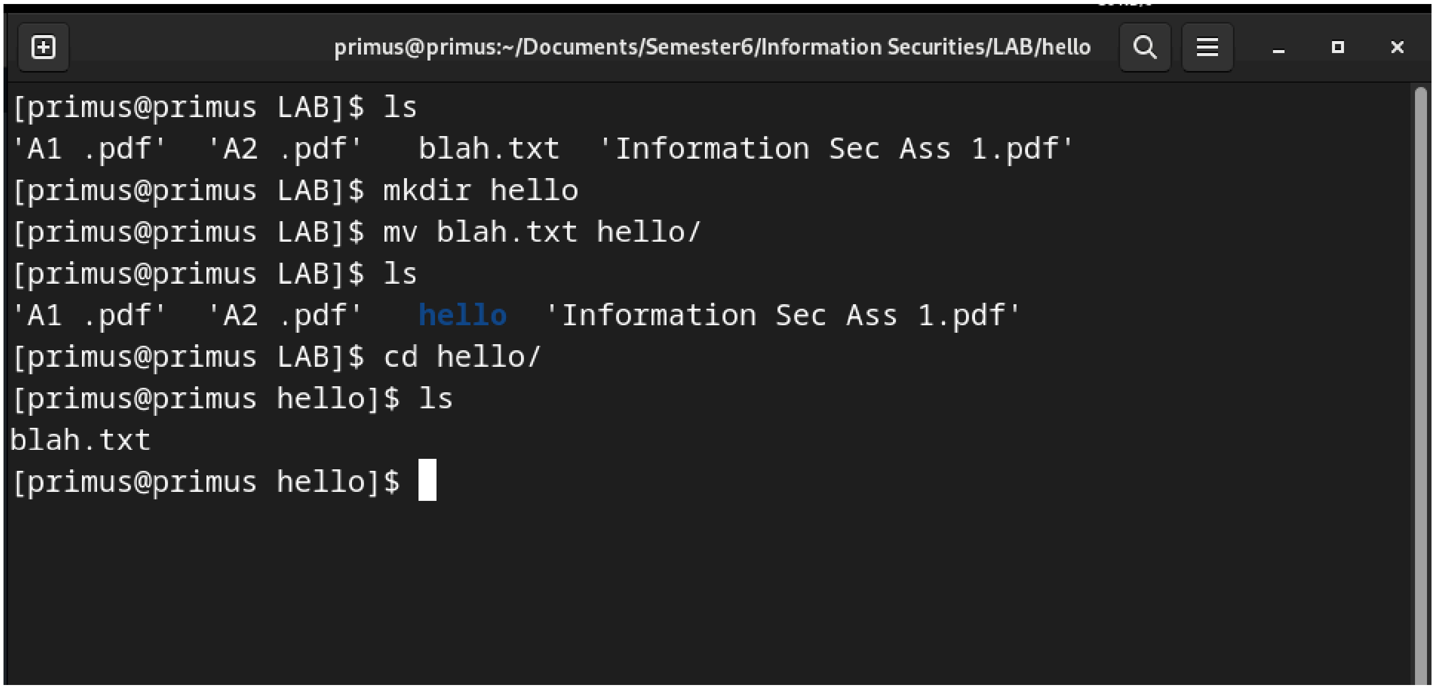
i. cp: Copies files or directories. Example: `cp source_file destination_file`


primus@primus:~/Documents/Semester6/Information Securities/LAB/testing

```
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   blah.txt  'Information Sec Ass 1.pdf'
[primus@primus LAB]$ mkdir testing
[primus@primus LAB]$ cp
A1 .pdf                    Information Sec Ass 1.pdf
A2 .pdf                    testing/
blah.txt
[primus@primus LAB]$ cp blah.txt testing/
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   blah.txt  'Information Sec Ass 1.pdf'   testing
[primus@primus LAB]$ cd testing/
[primus@primus testing]$ ls
blah.txt
[primus@primus testing]$
```

c. **Moving Files:**

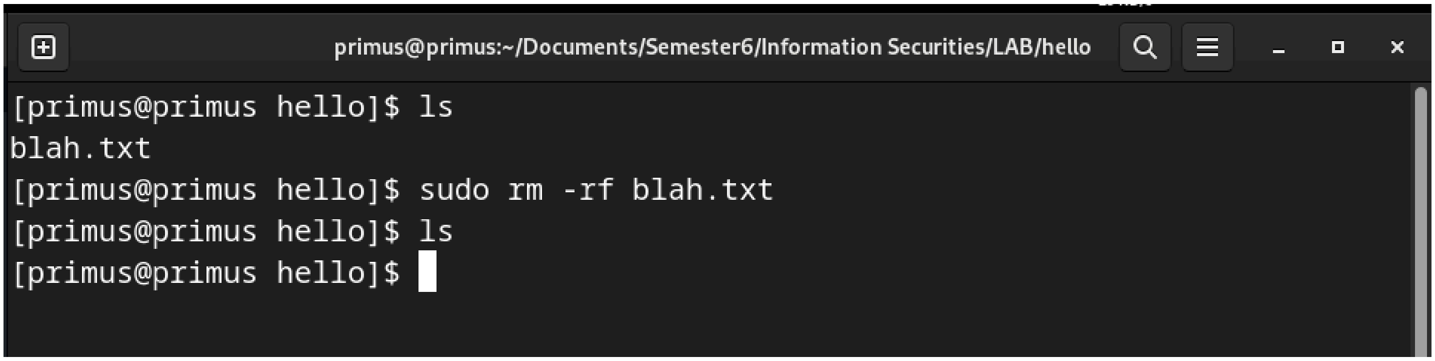
   i. mv: Moves or renames files or directories. Example: `mv source_file destination_file`


primus@primus:~/Documents/Semester6/Information Securities/LAB/hello

```
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   blah.txt  'Information Sec Ass 1.pdf'
[primus@primus LAB]$ mkdir hello
[primus@primus LAB]$ mv blah.txt hello/
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   hello  'Information Sec Ass 1.pdf'
[primus@primus LAB]$ cd hello/
[primus@primus hello]$ ls
blah.txt
[primus@primus hello]$
```

d. **Removing Files:**

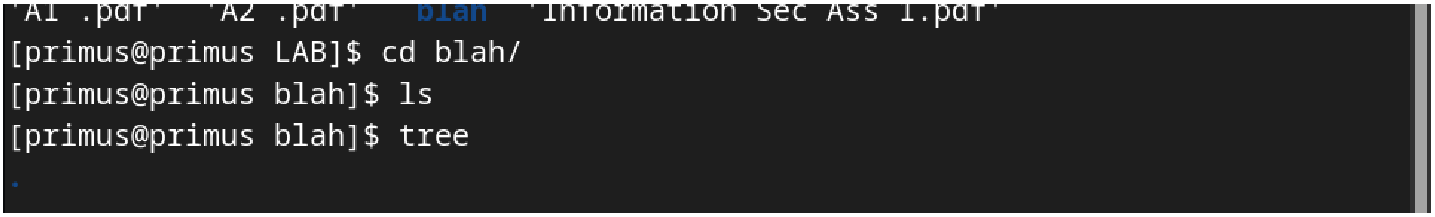
i. rm: Removes files or directories.
ii. p: Removes parent directories as well if empty.
iii. -pv: Shows each file as it's being removed.
iv. -rf: Forcefully removes files or directories recursively without prompting.
v. -rp: Removes directories and their contents recursively.
vi. -r: Removes directories and their contents recursively.


primus@primus:~/Documents/Semester6/Information Securities/LAB/hello

```
[primus@primus hello]$ ls
blah.txt
[primus@primus hello]$ sudo rm -rf blah.txt
[primus@primus hello]$ ls
[primus@primus hello]$ 
```

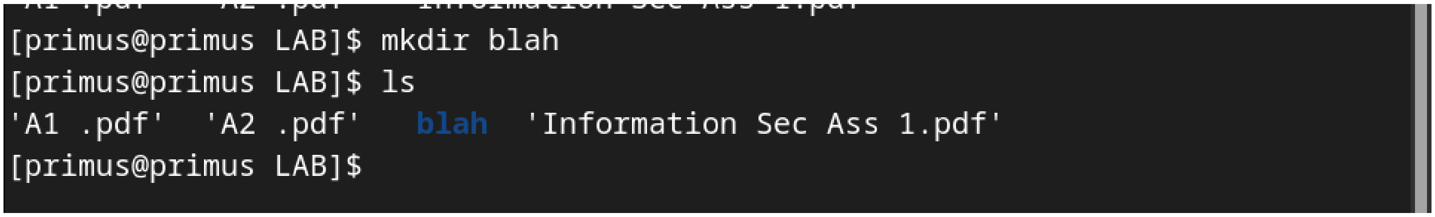
e. **Directory Commands:**

i. sudo: Executes a command with superuser privileges
ii. sudo su: Switches user to root.
iii. cd: Changes the current directory. Example: `cd directory_path`



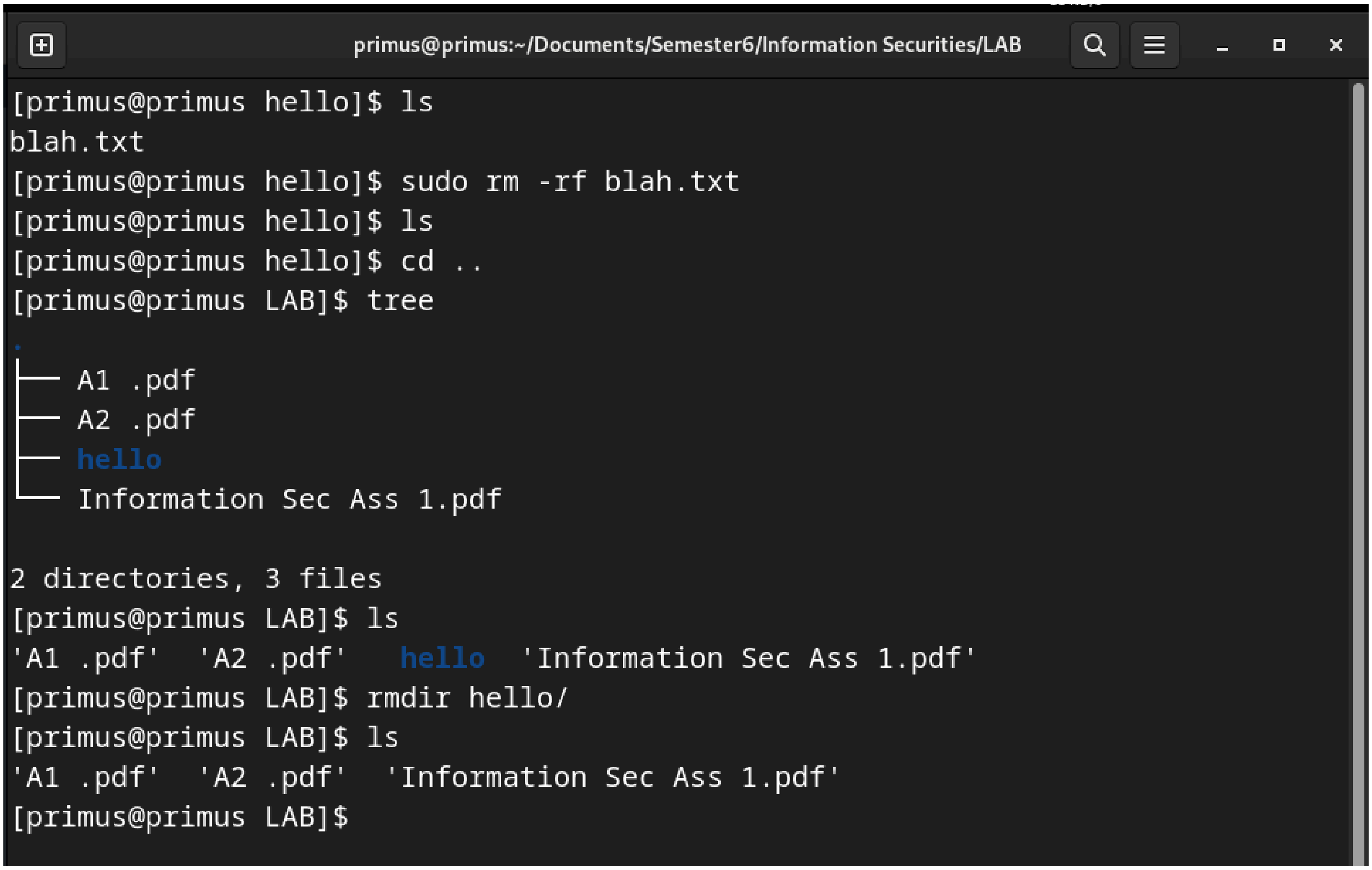
```
'A1 .pdf'   'A2 .pdf'    blah   'Information Sec Ass 1.pdf'
[primus@primus LAB]$ cd blah/
[primus@primus blah]$ ls
[primus@primus blah]$ tree
.
```

iv. mkdir: Creates a new directory. Example: `mkdir directory_name`



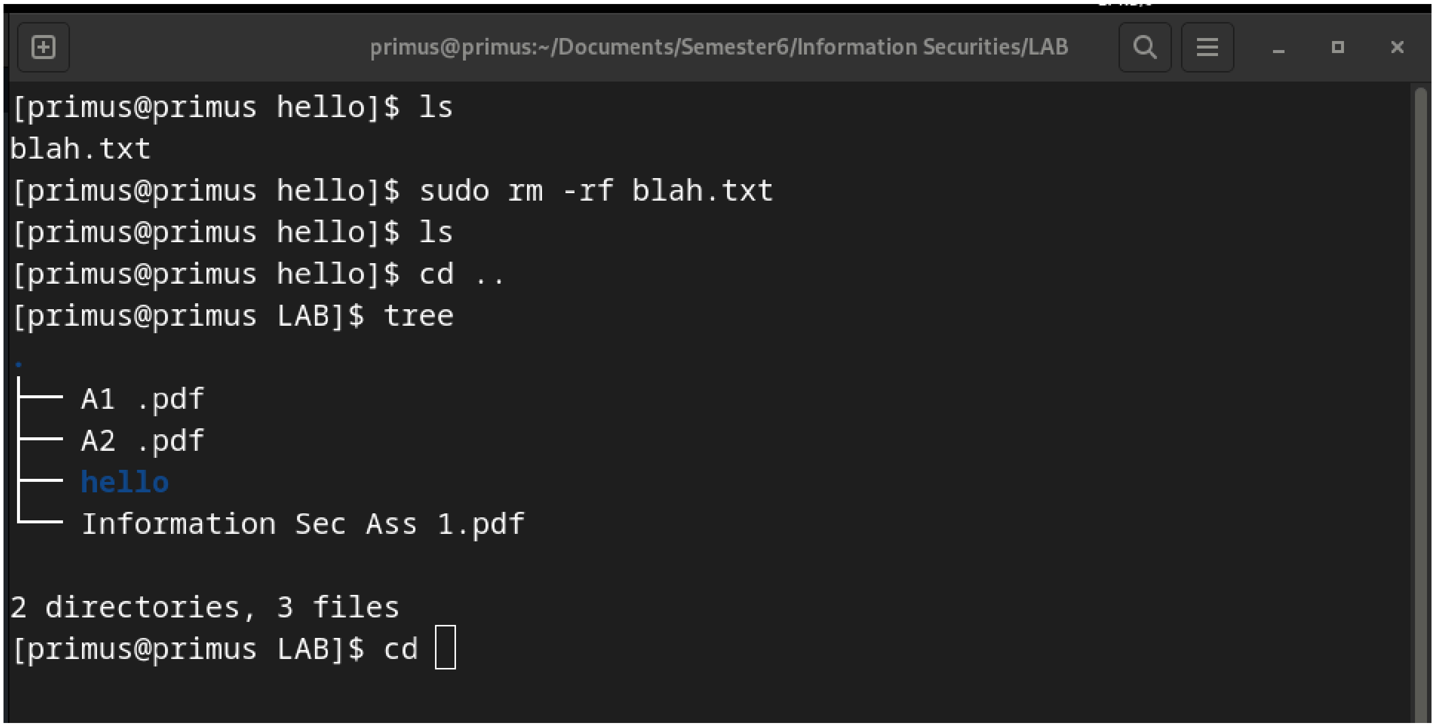
```
'A1 .pdf'   'A2 .pdf'    Information Sec Ass 1.pdf'
[primus@primus LAB]$ mkdir blah
[primus@primus LAB]$ ls
'A1 .pdf'   'A2 .pdf'    blah   'Information Sec Ass 1.pdf'
[primus@primus LAB]$
```

v. rmdir: Removes empty directories. Example: `rmdir directory_name`


primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus hello]$ ls
blah.txt
[primus@primus hello]$ sudo rm -rf blah.txt
[primus@primus hello]$ ls
[primus@primus hello]$ cd ..
[primus@primus LAB]$ tree
.
├── A1 .pdf
├── A2 .pdf
├── hello
└── Information Sec Ass 1.pdf

2 directories, 3 files
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   hello  'Information Sec Ass 1.pdf'
[primus@primus LAB]$ rmdir hello/
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'  'Information Sec Ass 1.pdf'
[primus@primus LAB]$
```

vi. tree: Displays the directory tree structure. Example: `tree directory_path`

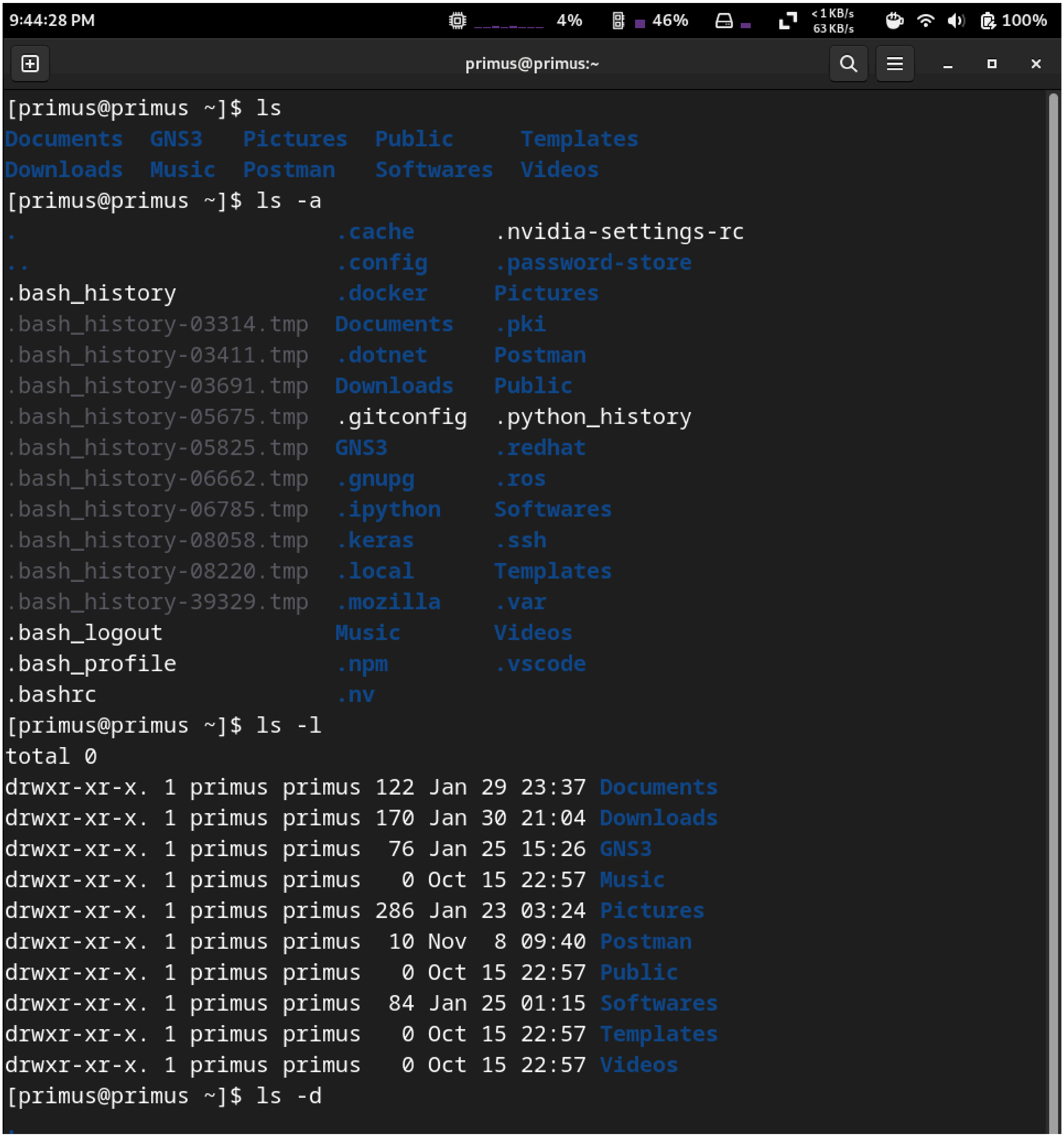

primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus hello]$ ls
blah.txt
[primus@primus hello]$ sudo rm -rf blah.txt
[primus@primus hello]$ ls
[primus@primus hello]$ cd ..
[primus@primus LAB]$ tree
.
├── A1 .pdf
├── A2 .pdf
├── hello
└── Information Sec Ass 1.pdf

2 directories, 3 files
[primus@primus LAB]$ cd
```

f. **Miscellaneous Commands:**

   i. -ls: Lists files and directories.
      1. -a: Lists all files, including hidden ones.
      2. -l: Displays detailed information about files.

   ii. -d: Lists only directories.



```
[primus@primus ~]$ ls
Documents  GNS3   Pictures  Public     Templates
Downloads  Music  Postman   Softwares  Videos
[primus@primus ~]$ ls -a
.                        .cache      .nvidia-settings-rc
..                       .config     .password-store
.bash_history            .docker     Pictures
.bash_history-03314.tmp  Documents   .pki
.bash_history-03411.tmp  .dotnet     Postman
.bash_history-03691.tmp  Downloads   Public
.bash_history-05675.tmp  .gitconfig  .python_history
.bash_history-05825.tmp  GNS3        .redhat
.bash_history-06662.tmp  .gnupg      .ros
.bash_history-06785.tmp  .ipython    Softwares
.bash_history-08058.tmp  .keras      .ssh
.bash_history-08220.tmp  .local      Templates
.bash_history-39329.tmp  .mozilla    .var
.bash_logout             Music       Videos
.bash_profile            .npm        .vscode
.bashrc                  .nv
[primus@primus ~]$ ls -l
total 0
drwxr-xr-x. 1 primus primus 122 Jan 29 23:37 Documents
drwxr-xr-x. 1 primus primus 170 Jan 30 21:04 Downloads
drwxr-xr-x. 1 primus primus  76 Jan 25 15:26 GNS3
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Music
drwxr-xr-x. 1 primus primus 286 Jan 23 03:24 Pictures
drwxr-xr-x. 1 primus primus  10 Nov  8 09:40 Postman
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Public
drwxr-xr-x. 1 primus primus  84 Jan 25 01:15 Softwares
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Templates
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Videos
[primus@primus ~]$ ls -d
.
```

iii. history: Displays a list of previously executed commands.

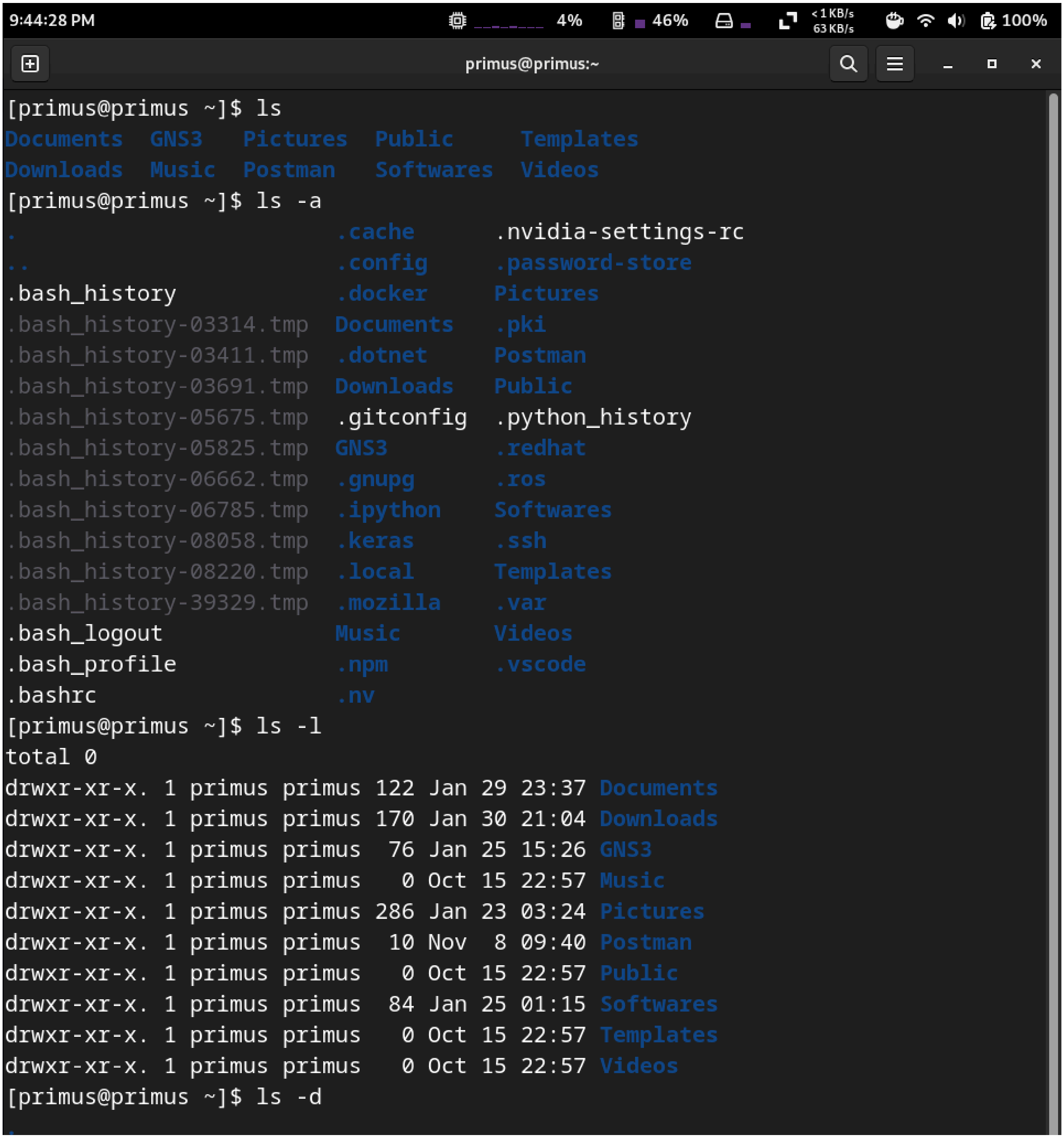
iv. pwd: Prints the current working directory.

v. head: Outputs the first part of files. Example: `head filename`

vi. tail: Outputs the last part of files. Example: `tail filename`

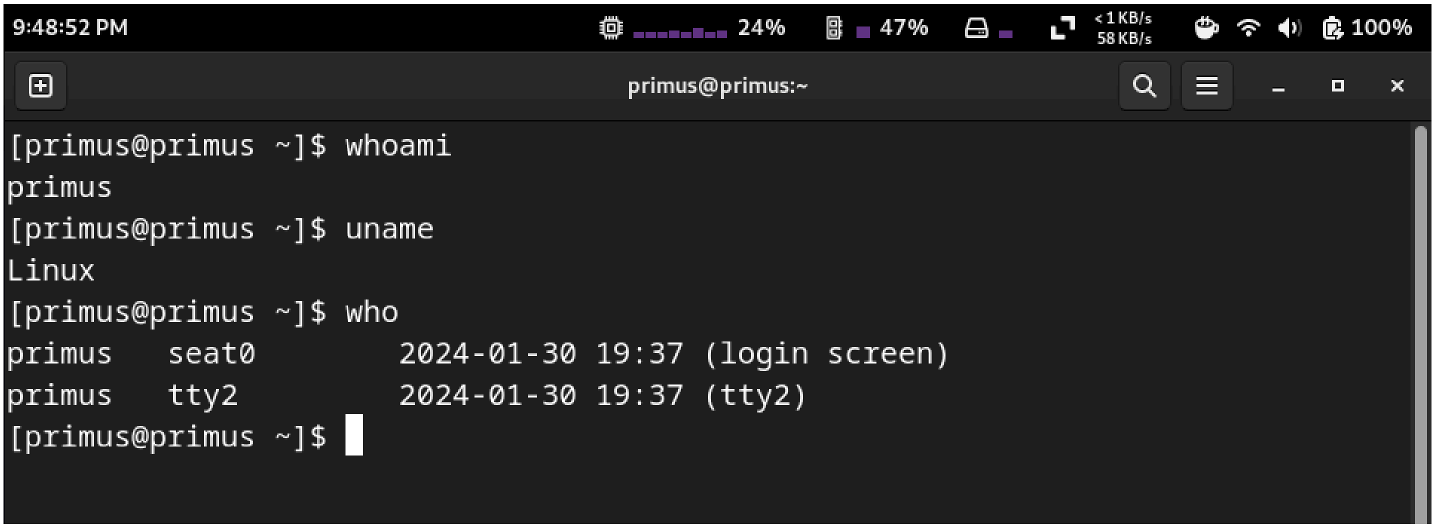
vii. ps: Displays information about active processes. Example: `ps aux`

viii. kill: Terminates processes. Example: `kill process_id



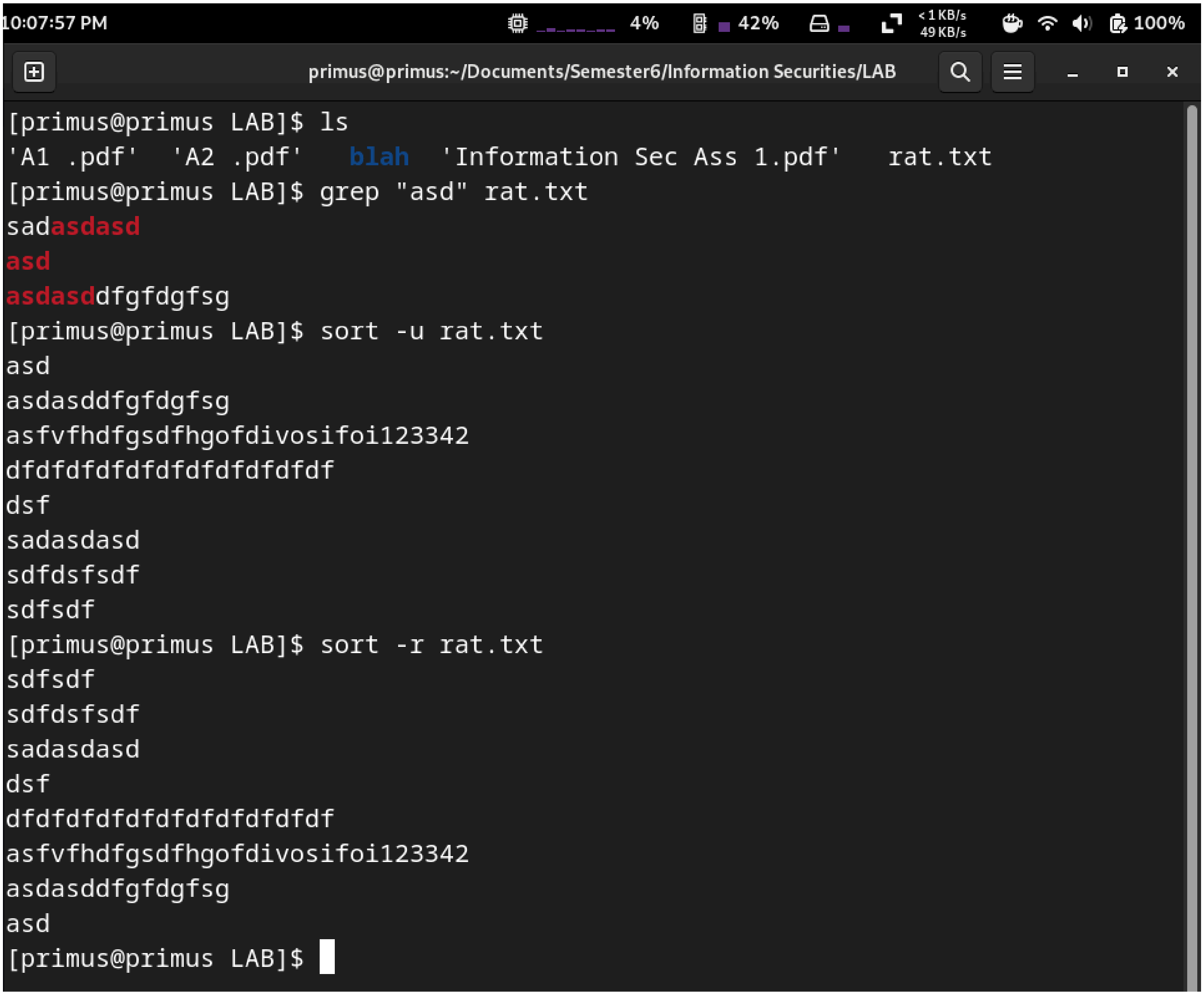
```
[primus@primus ~]$ ls
Documents  GNS3   Pictures  Public     Templates
Downloads  Music  Postman   Softwares  Videos
[primus@primus ~]$ ls -a
.                        .cache      .nvidia-settings-rc
..                       .config     .password-store
.bash_history            .docker     Pictures
.bash_history-03314.tmp  Documents   .pki
.bash_history-03411.tmp  .dotnet     Postman
.bash_history-03691.tmp  Downloads   Public
.bash_history-05675.tmp  .gitconfig  .python_history
.bash_history-05825.tmp  GNS3        .redhat
.bash_history-06662.tmp  .gnupg      .ros
.bash_history-06785.tmp  .ipython    Softwares
.bash_history-08058.tmp  .keras      .ssh
.bash_history-08220.tmp  .local      Templates
.bash_history-39329.tmp  .mozilla    .var
.bash_logout             Music       Videos
.bash_profile            .npm        .vscode
.bashrc                  .nv
[primus@primus ~]$ ls -l
total 0
drwxr-xr-x. 1 primus primus 122 Jan 29 23:37 Documents
drwxr-xr-x. 1 primus primus 170 Jan 30 21:04 Downloads
drwxr-xr-x. 1 primus primus  76 Jan 25 15:26 GNS3
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Music
drwxr-xr-x. 1 primus primus 286 Jan 23 03:24 Pictures
drwxr-xr-x. 1 primus primus  10 Nov  8 09:40 Postman
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Public
drwxr-xr-x. 1 primus primus  84 Jan 25 01:15 Softwares
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Templates
drwxr-xr-x. 1 primus primus   0 Oct 15 22:57 Videos
[primus@primus ~]$ ls -d
.
```

ix. **whoami -** to see the user name



```
[primus@primus ~]$ whoami
primus
[primus@primus ~]$ uname
Linux
[primus@primus ~]$ who
primus   seat0        2024-01-30 19:37 (login screen)
primus   tty2         2024-01-30 19:37 (tty2)
[primus@primus ~]$ 
```

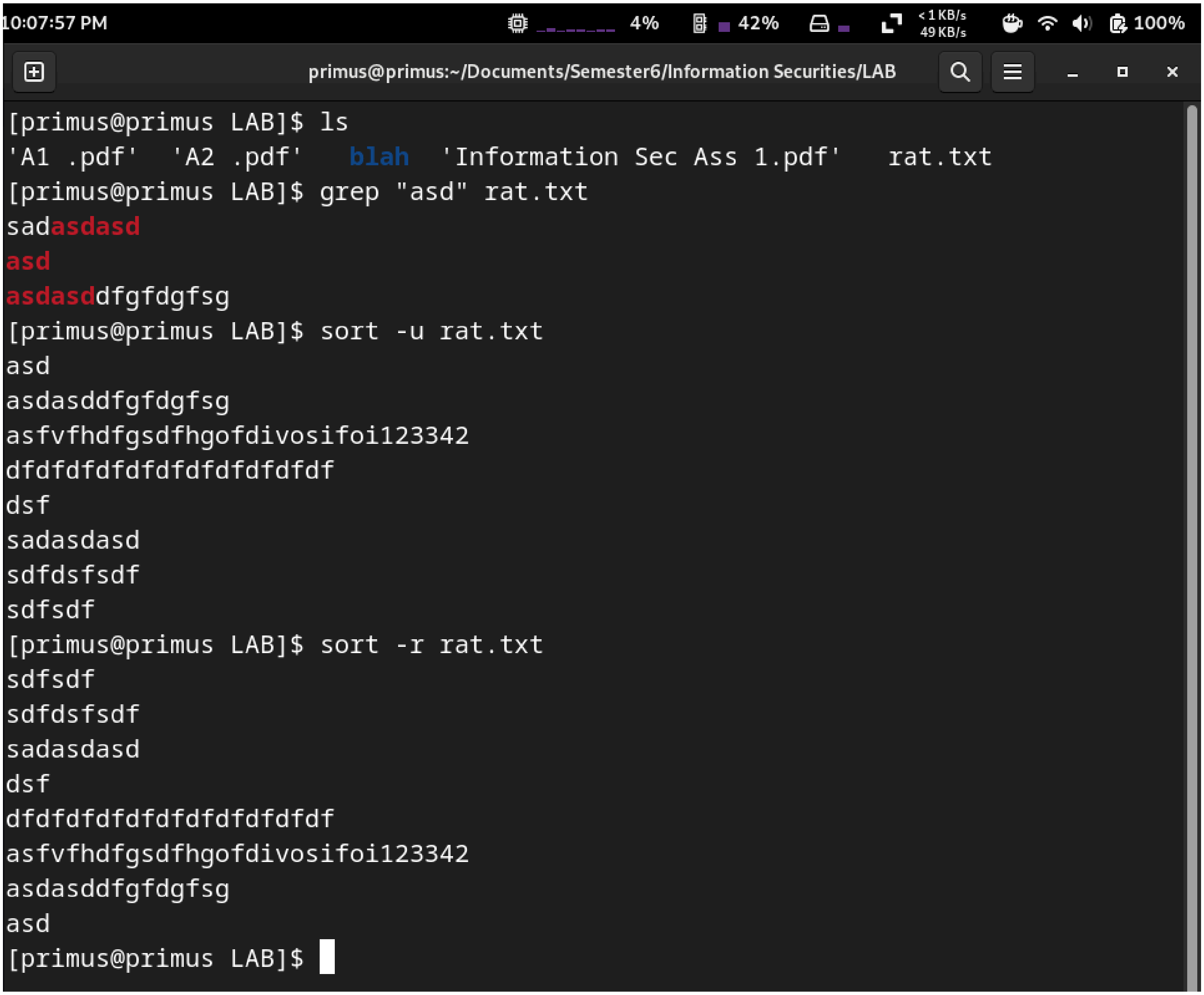
g. **Filter Commands:**
   - **i.** grep: A command-line utility for searching plain-text data using regular expressions. It filters lines that match a specified pattern.
   - **ii.** sort: This command sorts the lines of text files in ascending or descending order.


10:07:57 PM
4%
42%
< 1 KB/s
49 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

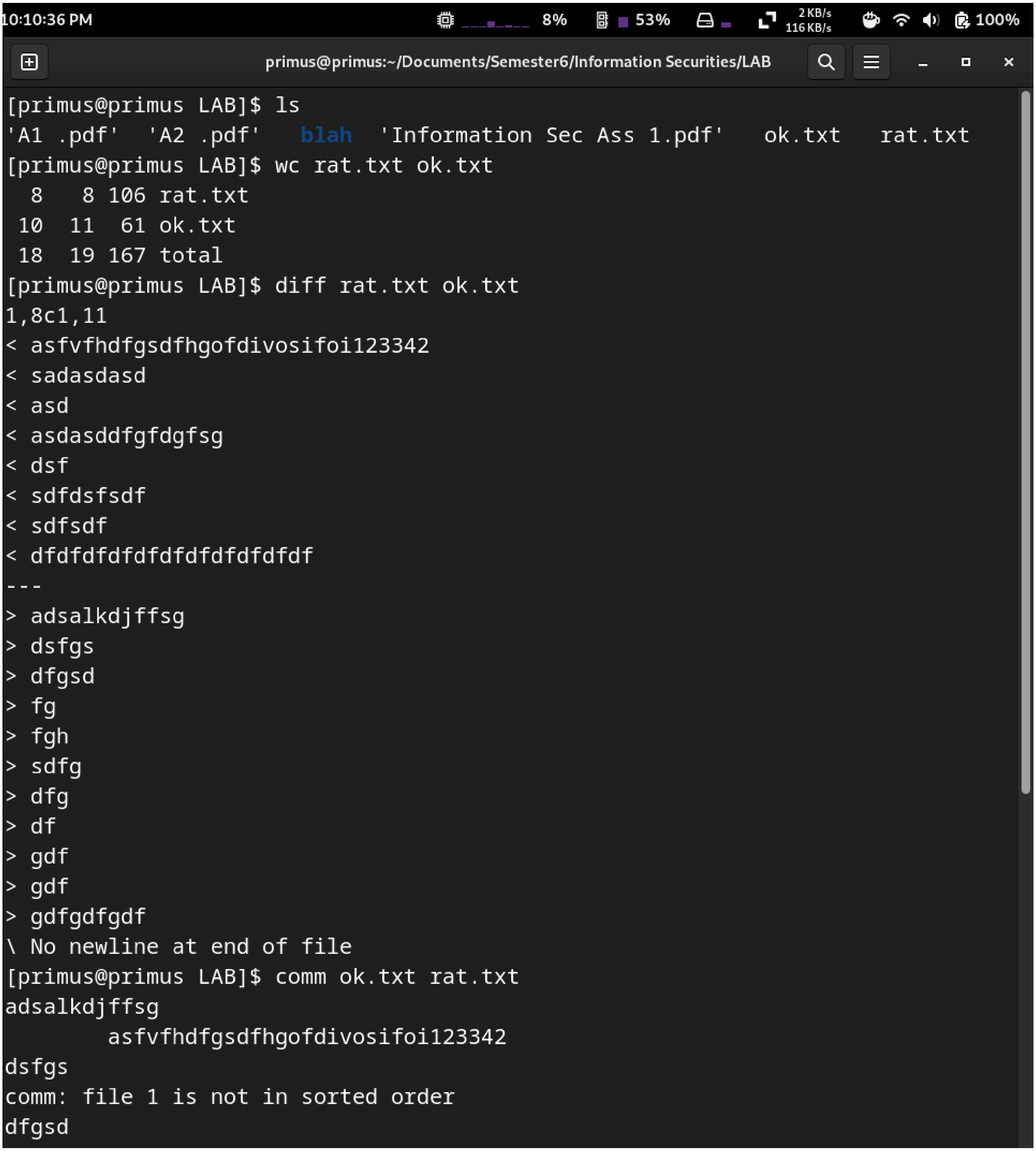
```
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   blah  'Information Sec Ass 1.pdf'   rat.txt
[primus@primus LAB]$ grep "asd" rat.txt
sadasdasd
asd
asdasddfgfdgfsg
[primus@primus LAB]$ sort -u rat.txt
asd
asdasddfgfdgfsg
asfvfhdfgsdfhgofdivosifoi123342
dfdfdfdfdfdfdfdfdfdfdf
dsf
sadasdasd
sdfdsfsdf
sdfsdf
[primus@primus LAB]$ sort -r rat.txt
sdfsdf
sdfdsfsdf
sadasdasd
dsf
dfdfdfdfdfdfdfdfdfdfdf
asfvfhdfgsdfhgofdivosifoi123342
asdasddfgfdgfsg
asd
[primus@primus LAB]$ 
```

h. **File Comparison Commands:**

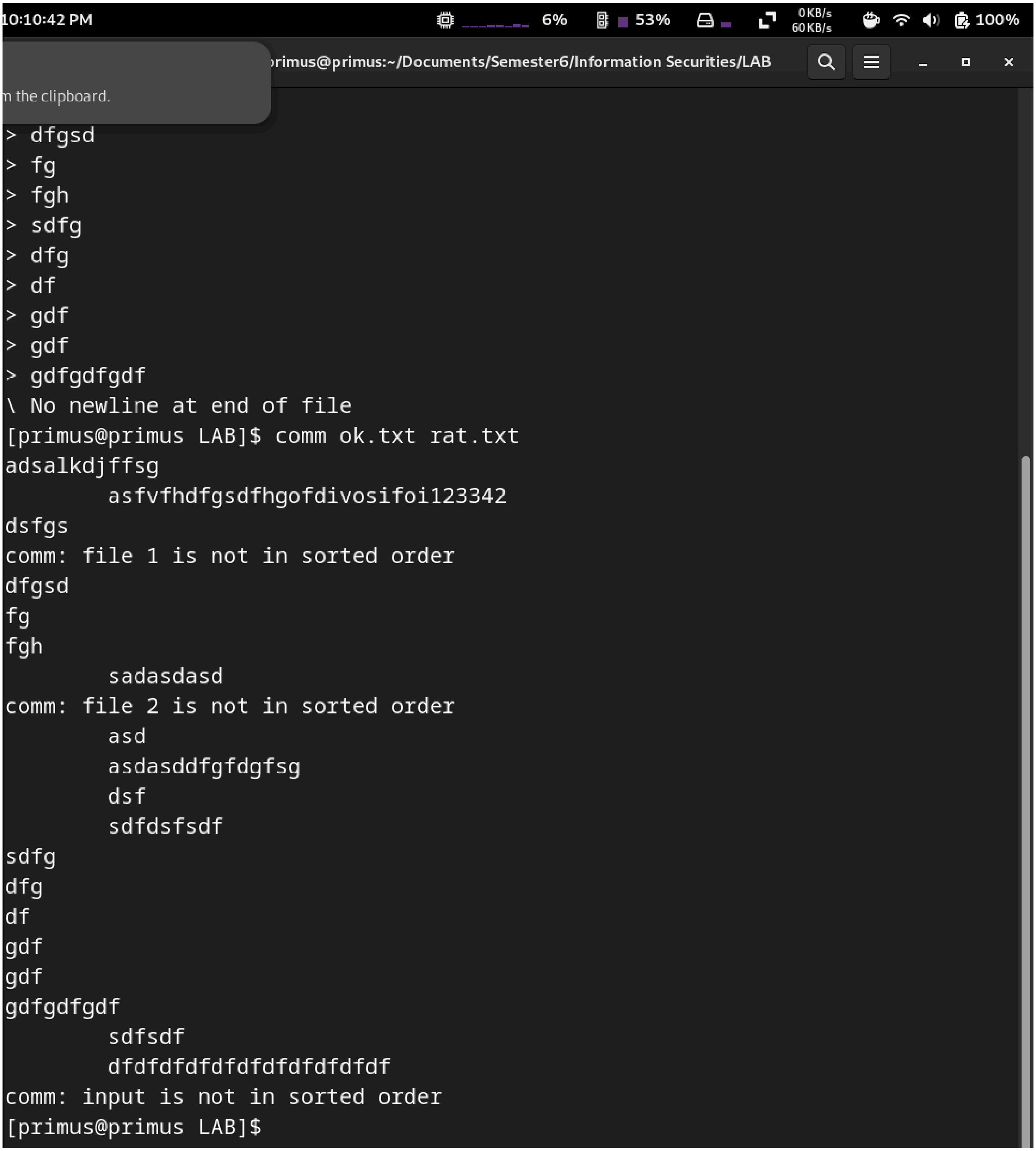
- **i.** cmp: Compares two files byte by byte and displays the first mismatch.
- **ii.** diff: Compares the contents of two files line by line and displays the differences between them.
- **iii.** comm: Compares two sorted files line by line and displays lines unique to each file and common to both.
- **iv.** wc: Counts the number of lines, words, and characters in a file or standard input.
- **v.** uniq: Filters adjacent matching lines from a sorted file and displays unique lines.


10:07:57 PM
4%
42%
< 1 KB/s
49 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   blah  'Information Sec Ass 1.pdf'   rat.txt
[primus@primus LAB]$ grep "asd" rat.txt
sadasdasd
asd
asdasddfgfdgfsg
[primus@primus LAB]$ sort -u rat.txt
asd
asdasddfgfdgfsg
asfvfhdfgsdfhgofdivosifoi123342
dfdfdfdfdfdfdfdfdfdfdf
dsf
sadasdasd
sdfdsfsdf
sdfsdf
[primus@primus LAB]$ sort -r rat.txt
sdfsdf
sdfdsfsdf
sadasdasd
dsf
dfdfdfdfdfdfdfdfdfdfdf
asfvfhdfgsdfhgofdivosifoi123342
asdasddfgfdgfsg
asd
[primus@primus LAB]$
```


10:10:36 PM
8%
53%
2 KB/s
116 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

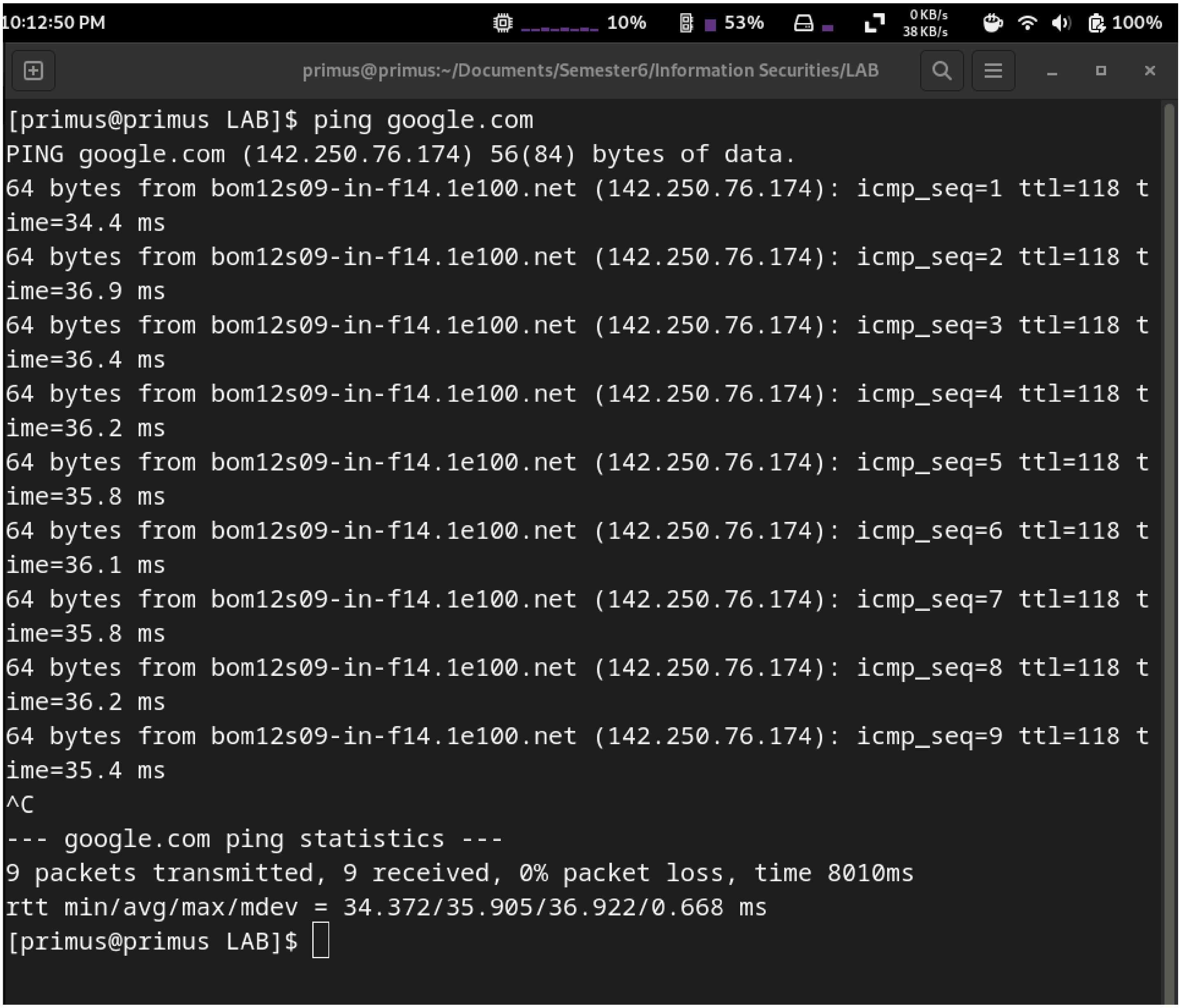
```
[primus@primus LAB]$ ls
'A1 .pdf'  'A2 .pdf'   blah  'Information Sec Ass 1.pdf'   ok.txt   rat.txt
[primus@primus LAB]$ wc rat.txt ok.txt
  8   8 106 rat.txt
 10  11  61 ok.txt
 18  19 167 total
[primus@primus LAB]$ diff rat.txt ok.txt
1,8c1,11
< asfvfhdfgsdfhgofdivosifoi123342
< sadasdasd
< asd
< asdasddfgfdgfsg
< dsf
< sdfdsfsdf
< sdfsdf
< dfdfdfdfdfdfdfdfdfdfdf
---
> adsalkdjffsg
> dsfgs
> dfgsd
> fg
> fgh
> sdfg
> dfg
> df
> gdf
> gdf
> gdfgdfgdf
\ No newline at end of file
[primus@primus LAB]$ comm ok.txt rat.txt
adsalkdjffsg
        asfvfhdfgsdfhgofdivosifoi123342
dsfgs
comm: file 1 is not in sorted order
dfgsd
```


10:10:42 PM
6%
53%
0 KB/s
60 KB/s
100%
rimus@primus:~/Documents/Semester6/Information Securities/LAB
n the clipboard.

```
> dfgsd
> fg
> fgh
> sdfg
> dfg
> df
> gdf
> gdf
> gdfgdfgdf
\ No newline at end of file
[primus@primus LAB]$ comm ok.txt rat.txt
adsalkdjffsg
        asfvfhdfgsdfhgofdivosifoi123342
dsfgs
comm: file 1 is not in sorted order
dfgsd
fg
fgh
        sadasdasd
comm: file 2 is not in sorted order
        asd
        asdasddfgfdgfsg
        dsf
        sdfdsfsdf
sdfg
dfg
df
gdf
gdf
gdfgdfgdf
        sdfsdf
        dfdfdfdfdfdfdfdfdfdfdf
comm: input is not in sorted order
[primus@primus LAB]$
```

- **i. Networking Commands:**
  - **i.** Ping:

1. **ping [hostname or IP address]:** Sends ICMP echo request packets to the specified host and waits for a response. Example: `ping google.com` or `ping 8.8.8.8`.

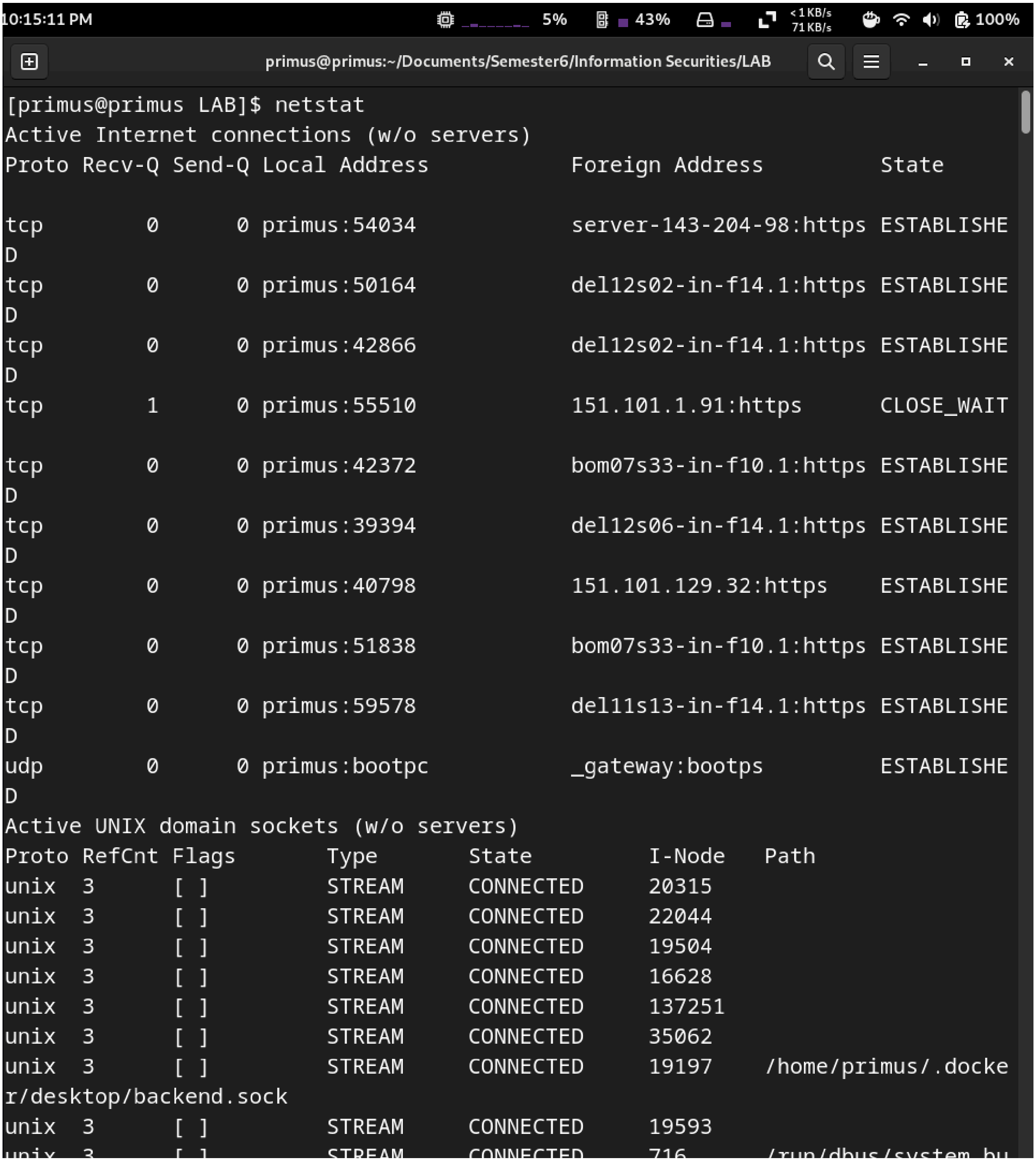

10:12:50 PM
10%
53%
0 KB/s
38 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ ping google.com
PING google.com (142.250.76.174) 56(84) bytes of data.
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=1 ttl=118 t
ime=34.4 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=2 ttl=118 t
ime=36.9 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=3 ttl=118 t
ime=36.4 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=4 ttl=118 t
ime=36.2 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=5 ttl=118 t
ime=35.8 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=6 ttl=118 t
ime=36.1 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=7 ttl=118 t
ime=35.8 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=8 ttl=118 t
ime=36.2 ms
64 bytes from bom12s09-in-f14.1e100.net (142.250.76.174): icmp_seq=9 ttl=118 t
ime=35.4 ms
^C
--- google.com ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8010ms
rtt min/avg/max/mdev = 34.372/35.905/36.922/0.668 ms
[primus@primus LAB]$ 
```

2. **ping -c [count]**: Specifies the number of ICMP echo request packets to send. Example: `ping -c 5 google.com` will send 5 packets
3. **ping -t [ttl]**: Sets the Time-To-Live (TTL) value for outgoing packets. Example: `ping -t 10 google.com` sets the TTL to 10.
4. **ping -i [interval]:** Specifies the interval between sending ICMP echo request packets. Example: `ping -i 2 google.com` sets the interval to 2 seconds.

ii. Netstat:

1. **netstat -a**: Displays all active connections and listening ports.

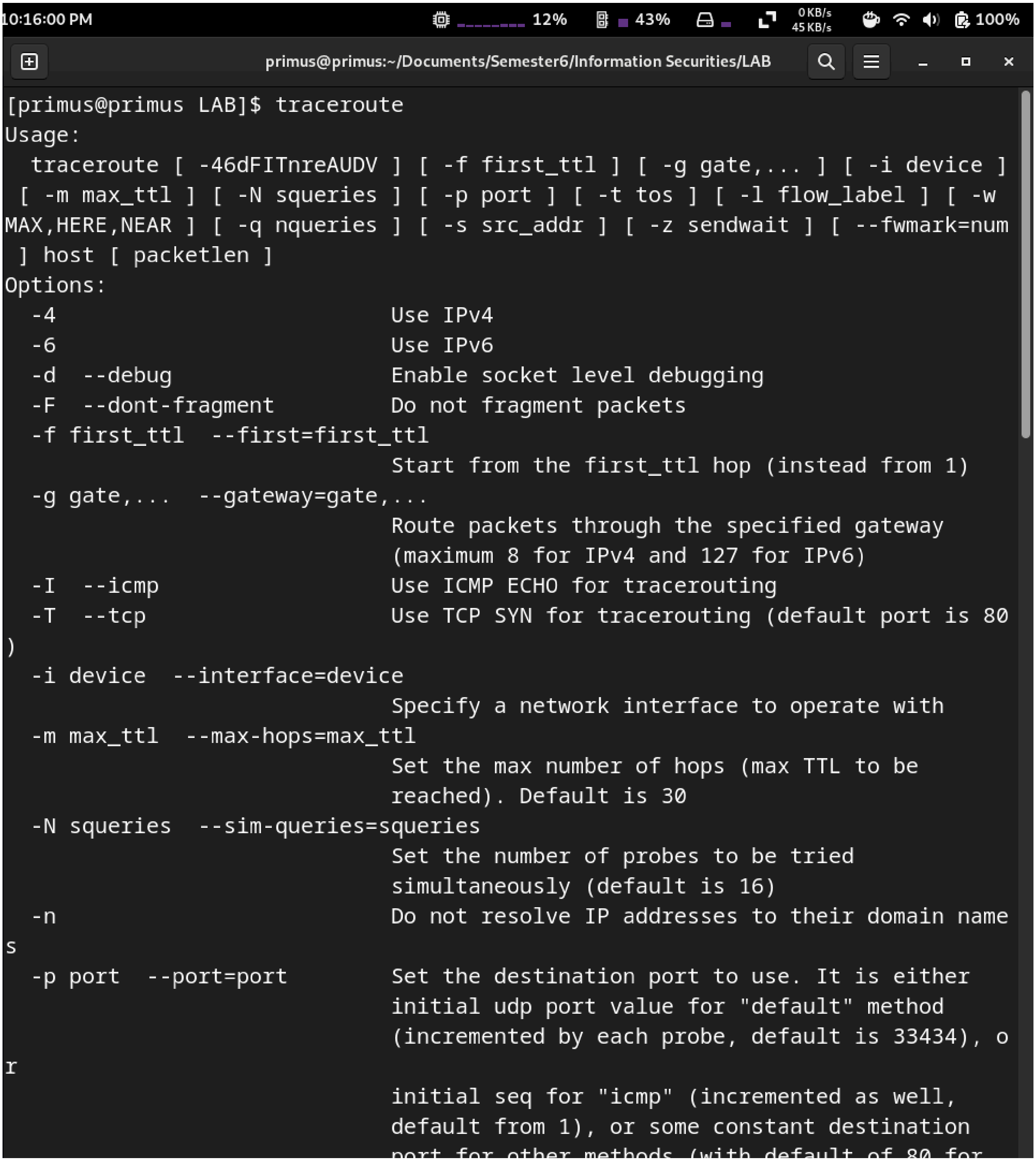

10:15:11 PM
5%
43%
< 1 KB/s
71 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State      

tcp        0      0 primus:54034            server-143-204-98:https ESTABLISHE
D
tcp        0      0 primus:50164            del12s02-in-f14.1:https ESTABLISHE
D
tcp        0      0 primus:42866            del12s02-in-f14.1:https ESTABLISHE
D
tcp        1      0 primus:55510            151.101.1.91:https      CLOSE_WAIT

tcp        0      0 primus:42372            bom07s33-in-f10.1:https ESTABLISHE
D
tcp        0      0 primus:39394            del12s06-in-f14.1:https ESTABLISHE
D
tcp        0      0 primus:40798            151.101.129.32:https    ESTABLISHE
D
tcp        0      0 primus:51838            bom07s33-in-f10.1:https ESTABLISHE
D
tcp        0      0 primus:59578            del11s13-in-f14.1:https ESTABLISHE
D
udp        0      0 primus:bootpc           _gateway:bootps         ESTABLISHE
D
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  3      [ ]         STREAM     CONNECTED     20315    
unix  3      [ ]         STREAM     CONNECTED     22044    
unix  3      [ ]         STREAM     CONNECTED     19504    
unix  3      [ ]         STREAM     CONNECTED     16628    
unix  3      [ ]         STREAM     CONNECTED     137251   
unix  3      [ ]         STREAM     CONNECTED     35062    
unix  3      [ ]         STREAM     CONNECTED     19197    /home/primus/.docke
r/desktop/backend.sock
unix  3      [ ]         STREAM     CONNECTED     19593    
unix  3      [ ]         STREAM     CONNECTED     716      /run/dbus/system_bu
```

2. **netstat -r**: Shows the routing table, including the default gateway and interface-specific routes.

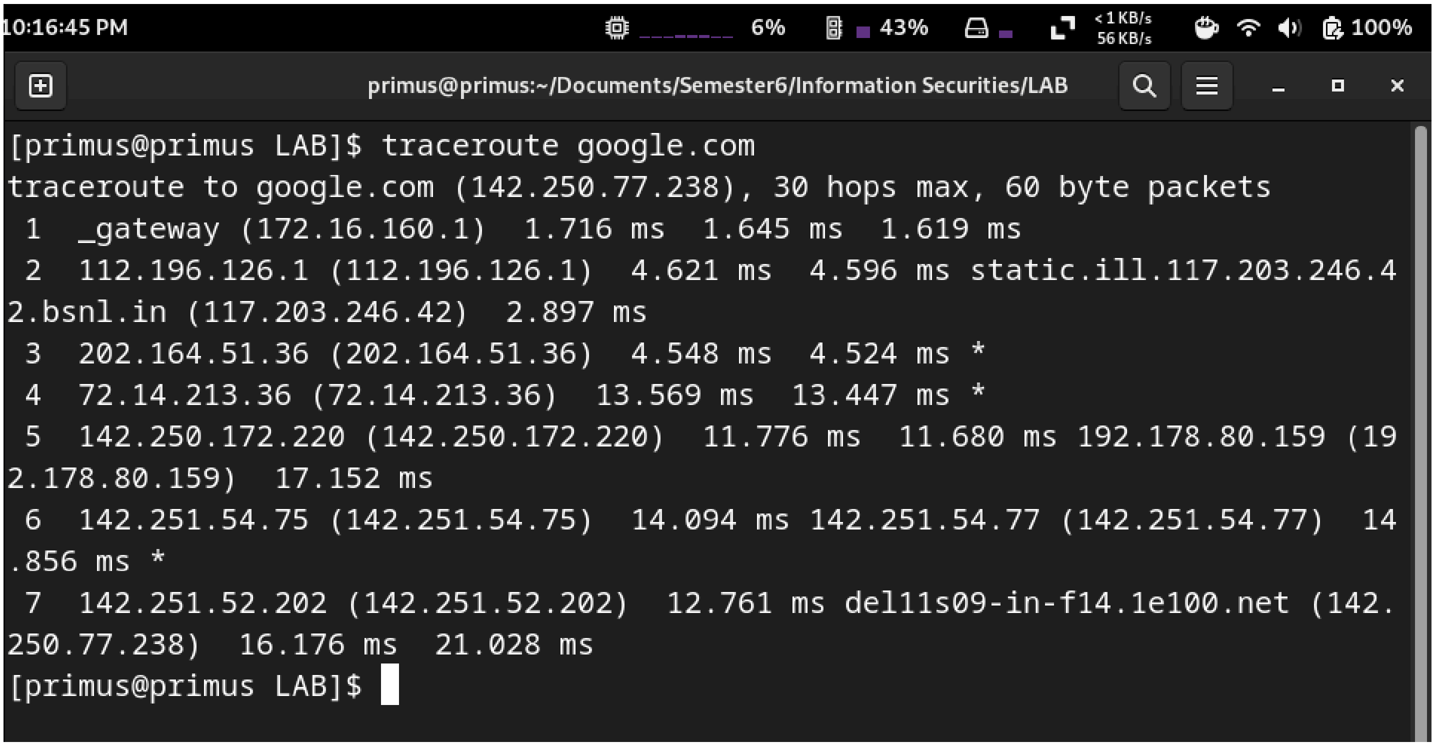
3. **netstat -n:** Displays numerical addresses and port numbers instead of resolving them to hostnames and service names.
4. **netstat -t:** Shows TCP connections.
5. **netstat -u:** Displays UDP connections.
6. **netstat -p:** Shows the process IDs (PIDs) of the processes associated with each connection.

iii. tracert (traceroute): traceroute is for linux


10:16:00 PM
12%
43%
0 KB/s
45 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ traceroute
Usage:
  traceroute [ -46dFITnreAUDV ] [ -f first_ttl ] [ -g gate,... ] [ -i device ]
 [ -m max_ttl ] [ -N squeries ] [ -p port ] [ -t tos ] [ -l flow_label ] [ -w
MAX,HERE,NEAR ] [ -q nqueries ] [ -s src_addr ] [ -z sendwait ] [ --fwmark=num
 ] host [ packetlen ]
Options:
  -4                          Use IPv4
  -6                          Use IPv6
  -d  --debug                 Enable socket level debugging
  -F  --dont-fragment         Do not fragment packets
  -f first_ttl  --first=first_ttl
                              Start from the first_ttl hop (instead from 1)
  -g gate,...  --gateway=gate,...
                              Route packets through the specified gateway
                              (maximum 8 for IPv4 and 127 for IPv6)
  -I  --icmp                  Use ICMP ECHO for tracerouting
  -T  --tcp                   Use TCP SYN for tracerouting (default port is 80
)
  -i device  --interface=device
                              Specify a network interface to operate with
  -m max_ttl  --max-hops=max_ttl
                              Set the max number of hops (max TTL to be
                              reached). Default is 30
  -N squeries  --sim-queries=squeries
                              Set the number of probes to be tried
                              simultaneously (default is 16)
  -n                          Do not resolve IP addresses to their domain name
s
  -p port  --port=port        Set the destination port to use. It is either
                              initial udp port value for "default" method
                              (incremented by each probe, default is 33434), o
r
                              initial seq for "icmp" (incremented as well,
                              default from 1), or some constant destination
                              port for other methods (with default of 80 for
```

1. **tracert [hostname or IP address]:** Traces the route taken by packets from the source to the destination, displaying the IP addresses of routers along the way. Example: `tracert google.com` or `traceroute 8.8.8.8`.

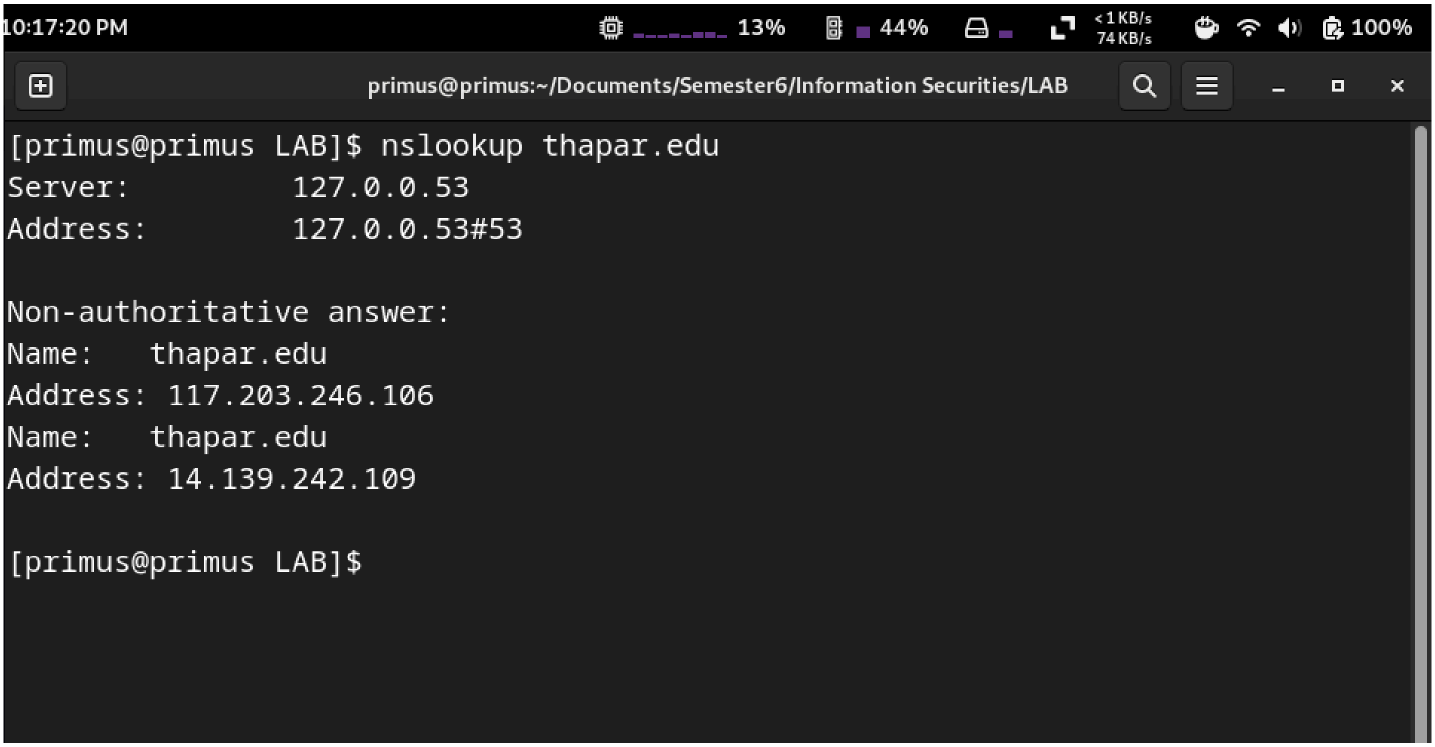


10:16:45 PM
6%
43%
< 1 KB/s
56 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ traceroute google.com
traceroute to google.com (142.250.77.238), 30 hops max, 60 byte packets
 1  _gateway (172.16.160.1)  1.716 ms  1.645 ms  1.619 ms
 2  112.196.126.1 (112.196.126.1)  4.621 ms  4.596 ms static.ill.117.203.246.4
2.bsnl.in (117.203.246.42)  2.897 ms
 3  202.164.51.36 (202.164.51.36)  4.548 ms  4.524 ms *
 4  72.14.213.36 (72.14.213.36)  13.569 ms  13.447 ms *
 5  142.250.172.220 (142.250.172.220)  11.776 ms  11.680 ms 192.178.80.159 (19
2.178.80.159)  17.152 ms
 6  142.251.54.75 (142.251.54.75)  14.094 ms 142.251.54.77 (142.251.54.77)  14
.856 ms *
 7  142.251.52.202 (142.251.52.202)  12.761 ms del11s09-in-f14.1e100.net (142.
250.77.238)  16.176 ms  21.028 ms
[primus@primus LAB]$ 
```

2. **tracert -d:** Performs a trace without attempting to resolve IP addresses to hostnames.
3. **tracert -h [max_hops]:** Specifies the maximum number of hops (routers) in the path. Example: `tracert -h 10 google.com` limits the trace to 10 hops.

iv. Nslookup:



10:17:20 PM
13%
44%
< 1 KB/s
74 KB/s
100%
primus@primus:~/Documents/Semester6/Information Securities/LAB

```
[primus@primus LAB]$ nslookup thapar.edu
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   thapar.edu
Address: 117.203.246.106
Name:   thapar.edu
Address: 14.139.242.109

[primus@primus LAB]$
```

1. **nslookup [hostname]:** Queries the DNS to obtain domain name or IP address information for the specified hostname. Example: `nslookup google.com`.
2. **nslookup -type=[query_type] [hostname]:** Specifies the type of DNS record to query. Example: `nslookup -type=mx google.com` queries the MX (Mail Exchange) records for google.com.
3. **nslookup -querytype=[query_type] [hostname]:** Similar to the above, but with a different syntax. Example: `nslookup -querytype=ns google.com` queries the NS (Name Server) records for google.com.