

# Week 2: Linux Fundamentals

Name: V. Risvanth

## Introduction

Linux is a powerful and flexible operating system widely used in server environments, cybersecurity, and software development. Mastering basic Linux administration is essential for managing files, processes, users, and system resources efficiently. This report documents the execution of various **Linux administrative tasks**, including file management, user management, process control, system monitoring, and network troubleshooting.

## Scope of the Study

This task covers fundamental Linux operations, including:

1. **File and Directory Management** – Creating, copying, moving, and deleting files and directories.
2. **File Permissions and Execution** – Assigning and modifying file permissions.
3. **Package Management** – Installing, verifying, and removing software using the package manager.
4. **Process Management** – Identifying and terminating running processes.
5. **Network Configuration and Troubleshooting** – Checking the IP address, testing connectivity, and listing active network connections.
6. **User Management** – Creating users, switching accounts, and granting administrative privileges.
7. **Shell Scripting** – Writing and executing basic shell scripts.
8. **Archiving and Compression** – Creating tarballs, extracting files, and compressing archives.
9. **System Monitoring** – Checking disk usage, memory utilization, and CPU information.

## **Importance of the Study**

Understanding these core Linux commands and operations is essential for system administration, cybersecurity, and troubleshooting. These skills help in:

- Managing files and directories efficiently.
- Controlling access to files and scripts through permissions.
- Monitoring and terminating processes to optimize system performance.
- Troubleshooting network connectivity and diagnosing issues.
- Managing users and granting appropriate permissions for security.
- Automating tasks using shell scripts to improve productivity.
- Monitoring system resources to ensure optimal performance.

## **Tools and Commands Used**

This task utilizes common **Linux commands** such as:

- mkdir, cd, touch, cp, mv, rm (for file and directory management).
- chmod, ls -l (for modifying and viewing file permissions).
- apt-get (for package management).
- ps, kill (for process management).
- ifconfig, ping, netstat (for network troubleshooting).
- useradd, sudo, su (for user management).
- tar, gzip (for file archiving and compression).
- df, free, lscpu (for system monitoring).

## Tools Used and Their Purpose

### 1. File and Directory Management

- **mkdir** – Creates directories.
- **cd** – Navigates between directories.
- **touch** – Creates empty files.
- **cp** – Copies files and directories.
- **mv** – Moves or renames files and directories.
- **rm** – Deletes files and directories.

**Purpose:** These commands allow efficient management of files and directories, which is essential for organizing system data.

### 2. File Permissions and Execution

- **chmod** – Modifies file permissions.
- **ls -l** – Lists files with detailed permission attributes.

**Purpose:** Controlling file permissions ensures security by restricting unauthorized access or execution.

### 3. Package Management

- **apt-get update** – Updates the package repository list.
- **apt-get install htop** – Installs the **htop** package.
- **apt-get remove htop** – Uninstalls the **htop** package.

**Purpose:** Managing software packages allows users to install and remove applications efficiently, keeping the system updated and optimized.

#### **4. Process Management**

- **ps aux** – Lists all running processes.
- **kill <PID>** – Terminates a specific process by its process ID (PID).

**Purpose:** Monitoring and managing processes is essential for optimizing system performance and preventing resource overuse.

#### **5. Network Configuration and Troubleshooting**

- **ifconfig** – Displays the system's IP address.
- **ping google.com** – Tests network connectivity.
- **netstat -tulnp** – Lists active network connections.

**Purpose:** Understanding network configurations and diagnosing connectivity issues is critical for maintaining stable and secure communication.

#### **6. User and Permission Management**

- **useradd testuser** – Creates a new user.
- **su - testuser** – Switches to the new user.
- **usermod -aG sudo testuser** – Grants sudo (administrator) privileges.

**Purpose:** Managing users and their privileges ensures a secure and organized system, preventing unauthorized access.

#### **7. Shell Scripting**

- **echo 'echo "Hello, World!"' > hello.sh** – Creates a simple shell script.
- **chmod +x hello.sh** – Grants executable permissions.
- **./hello.sh** – Executes the script.

**Purpose:** Shell scripting automates repetitive tasks, increasing efficiency and reducing manual errors.

## 8. Archiving and Compression

- **tar -cvf my\_project.tar my\_project/** – Creates a tarball archive.
- **tar -xvf my\_project.tar** – Extracts the contents of the tarball.
- **gzip my\_project.tar** – Compresses the tarball using gzip.

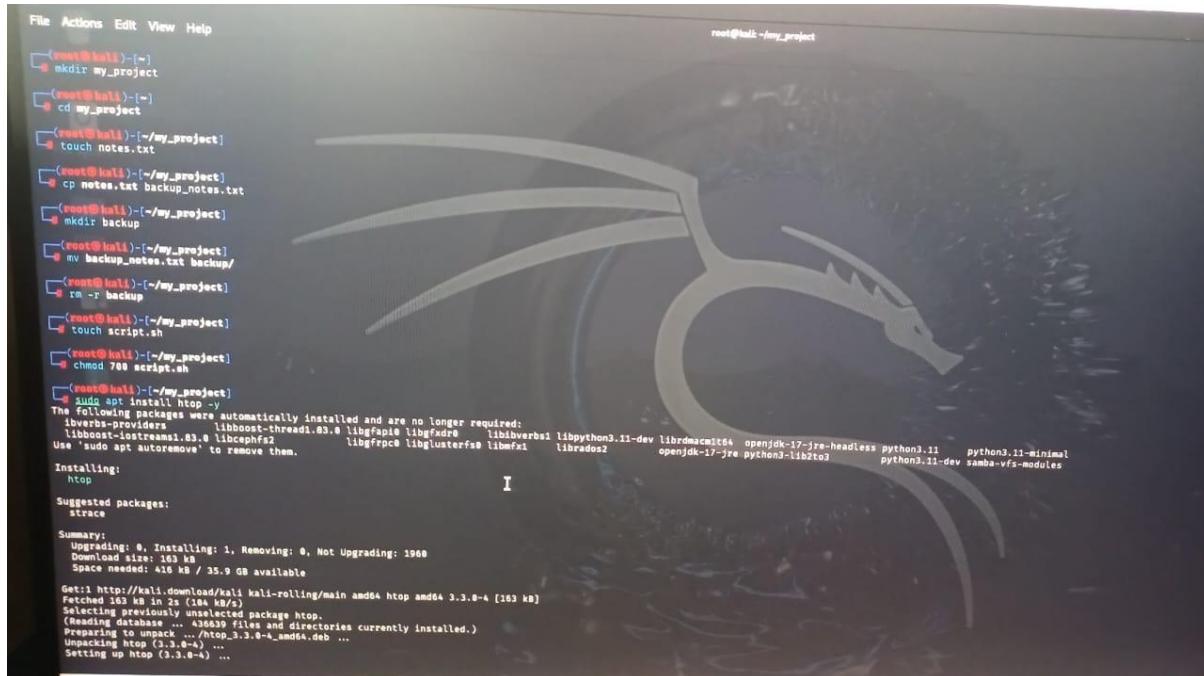
**Purpose:** File compression reduces storage space usage and enables easy data backup and transfer.

## 9. System Monitoring

- **df -h** – Displays disk usage.
- **free -m** – Checks memory (RAM) usage.
- **lscpu** – Shows CPU details.

**Purpose:** Monitoring system resources helps in performance tuning and early detection of issues like disk space shortages or memory leaks.

## Screenshots



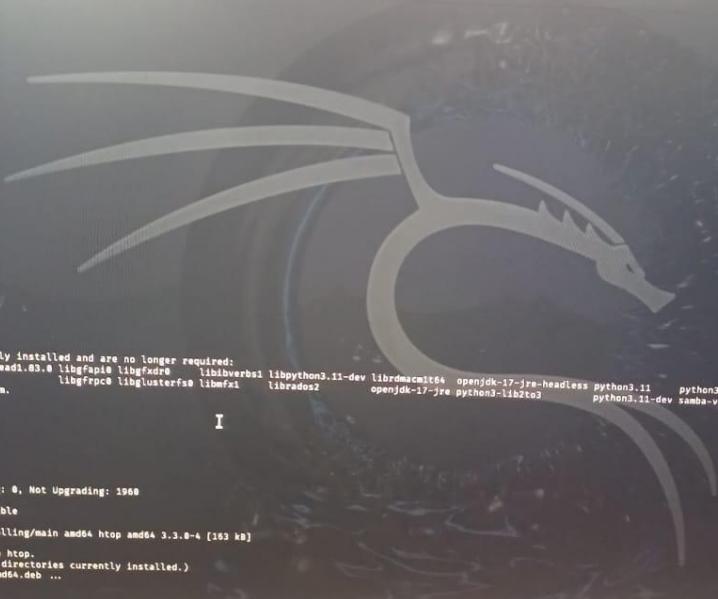
The screenshot shows a terminal window on a Kali Linux desktop. The terminal displays a series of commands being run by a user with root privileges. The commands include creating a directory, changing to it, creating a file, copying it to a backup, creating a backup directory, moving the file into it, renaming the file, creating a script, chmodding it to 700, and finally installing the 'http' package via apt. The terminal also shows the output of the 'apt' command, which lists packages being upgraded and the download progress of the 'http' package.

```
File Actions Edit View Help
root@kali:~# mkdir my_project
root@kali:~# cd my_project
root@kali:~/my_project# touch notes.txt
root@kali:~/my_project# cp notes.txt backup_notes.txt
root@kali:~/my_project# mkdir backup
root@kali:~/my_project# mv backup_notes.txt backup/
root@kali:~/my_project# rm -rf backup
root@kali:~/my_project# touch script.sh
root@kali:~/my_project# chmod 700 script.sh
root@kali:~/my_project# sudo apt install http -
The following packages were automatically installed and are no longer required:
  libverbs-providers  libboost-thread1.83.0  libgfaio  libgfuds  libibverbs1  libpython3.11-dev  librbdmac1t64  openjdk-17-jre-headless  python3.11-minimal
  libboost-iostreams1.83.0  libcephfs2  libgfrpc  libglusterfs  librados2  openjdk-17-jre  python3.11-lib2to3  python3.11-dev  samba-vfs-modules
Use 'sudo apt autoremove' to remove them.
Reading package lists...
Building dependency tree...
The following packages will be upgraded:
  http
0 upgraded, 1 newly installed, 0 to remove and 1968 not upgraded.
Need to get 163 kB of archives.
After this operation, 35.9 GB of additional disk space will be used.
Get:1 https://kali.download/kali kali-rolling/main amd64 http amd64 3.3.0-4 [163 kB]
Fetched 163 kB in 2s (104 kB/s)
Selecting previously unselected package http.
(Reading database ... 436639 files and directories currently installed.)
Preparing to unpack .../http_3.3.0-4_amd64.deb ...
Unpacking http (3.3.0-4) ...
Setting up http (3.3.0-4) ...
```

## 1. File and Directory Management

- **mkdir** – Creates directories.
- **cd** – Navigates between directories.
- **touch** – Creates empty files.
- **cp** – Copies files and directories.
- **mv** – Moves or renames files and directories.
- **rm** – Deletes files and directories.

**Purpose:** These commands allow efficient management of files and directories, which is essential for organizing system data



```
File Actions Edit View Help
root@kali:~/my_project
└─# mkdir my_project
└─# cd my_project
└─# touch notes.txt
└─# cp notes.txt backup_notes.txt
└─# mkdir backup
└─# mv backup_notes.txt backup/
└─# rm -r backup
└─# touch script.sh
└─# chmod 700 script.sh
└─# sudo apt install htop
The following packages were automatically installed and are no longer required:
libverbs-providers libboost-thread1.63.0 libgfaile libgfadre libibverbs1 libpython3.11-dev librdrmacm1t64 openjdk-17-jye-headless python3.11 python3.11-minimal
libboost-iostreams1.63.0 libcphf2 libgfpc libgusterfs libmfx1 librados2 openjdk-17-jre python3.11-libtzo3 python3.11-dev samba-vfs-modules
Use 'sudo apt autoremove' to remove them.
I
Installing:
htop
Suggested packages:
strace
Summary:
Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 1968
Download size: 163 kB
Space needed: 416 kB / 35.9 GB available
Get:1 http:// kali.download/kali kali-rolling/main amd64 htop amd64 3.3.0-4 [163 kB]
Fetched 163 kB in 0s (103 kB/s)
Selecting previously unselected package htop.
(Reading database ... 436639 files and directories currently installed.)
Preparing to unpack .../htop_3.3.0-4_amd64.deb ...
Unpacking htop (3.3.0-4) ...
Setting up htop (3.3.0-4) ...
```

```

File Actions Edit View Help
Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 1960
Download size: 163 kB
Space needed: 416 kB / 35.9 GB available
Get:1 http://kali.download/kali kali-rolling/main amd64 htop amd64 3.3.0-4 [163 kB]
Fetched 163 kB in 2s (104 kB/s)
Selecting previously unselected package htop.
(Reading database ... 436839 files and directories currently installed.)
Unpacking htop (3.3.0-4) ...
Setting up htop (3.3.0-4) ...
Processing triggers for mailcap (3.78+nmu1) ...
Processing triggers for kali-menu (2023.4.7) ...
Processing triggers for desktop-file-utils (0.27-2) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for man-db (2.12.1-1) ...
(root@kali:)[~/my_project]
└─# htop --version
htop 3.3.0
[root@kali:][~/my_project]
└─# sudo apt remove htop -y
The following packages were automatically installed and are no longer required:
libibus-1.6.0-1.83.0 libcephfs2 libgfrpc libglusterfs libmfx1 librados2
Use 'sudo apt autoremove' to remove them.
REMOVING:
htop
Summary:
Upgrading: 0, Installing: 0, Removing: 1, Not Upgrading: 1960
Free disk space: 416 kB
Removing htop (3.3.0-4) ...
Processing triggers for desktop-file-utils (0.27-2) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for man-db (2.12.1-1) ...
Processing triggers for mailcap (3.78+nmu1) ...
Processing triggers for kali-menu (2023.4.7) ...
[root@kali:][~/my_project]
└─# ps aux | grep sleep
root 9817 0.0 0.0 6356 2176 pts/0 S+ 11:41 0:00 grep --color=auto sleep
[root@kali:][~/my_project]
└─# kill 9817
kill: kill 9817 failed: no such process
[root@kali:][~/my_project]
└─# kill <9817>

```

## 2. Package Management

- apt-get update** – Updates the package repository list.
- apt-get install htop** – Installs the **htop** package.
- apt-get remove htop** – Uninstalls the **htop** package.

**Purpose:** Managing software packages allows users to install and remove applications efficiently, keeping the system updated and optimized.

```

File Actions Edit View Help
ps -o 9817
PID TTY TIME CMD
[root@kali:][~/my_project]
└─# ping -c google.com
ping: invalid argument: google.com
[root@kali:][~/my_project]
└─# ping -c 4 google.com
PING google.com (142.250.196.46) 56(84) bytes of data.
64 bytes from maab3a55-in-f1a.1e100.net (142.250.196.46): icmp_seq=1 ttl=128 time=18.2 ms
bytes from maab3a55-in-f1a.1e100.net (142.250.196.46): icmp_seq=2 ttl=128 time=18.4 ms
bytes from maab3a55-in-f1a.1e100.net (142.250.196.46): icmp_seq=3 ttl=128 time=18.6 ms
bytes from maab3a55-in-f1a.1e100.net (142.250.196.46): icmp_seq=4 ttl=128 time=18.4 ms
google.com ping statistics:
4 packets transmitted, 4 received, 0% packet loss, time 305ms
rtt min/avg/max/mdev = 10.238/10.377/10.559/0.117 ms
[root@kali:][~/my_project]
└─# ip addr show
1: eth0:  NOARP LOWER_UP mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
    link/ether 00:0c:29:1d:2f:70 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 0.0.0.0 scope host noprefixroute
        valid_lft forever preferred_lft forever
        link-layer scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth1:  NOARP MULTICAST LOWER_UP mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:1d:2f:70 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.10/24 brd 192.168.1.255 scope global noprefixroute eth0
        valid_lft forever preferred_lft forever
        link-layer scope host noprefixroute
        valid_lft forever preferred_lft forever
[root@kali:][~/my_project]
└─# netstat -an
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
Proto Recv-Q Send-Q Local Address:Port          Peer Address:Port
[root@kali:][~/my_project]
└─# ss -tln
Netid      State            Recv-Q           Send-Q
[root@kali:][~/my_project]
└─# sudo adduser testuser
info: Adding user 'testuser' (1002) ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group 'testuser' (1002) ...
info: Adding new user 'testuser' (1002) with group 'testuser' (1002) ...

```

```
[root@kali:~/my_project]
# ps aux | grep sleep
root      9817  0.0  0.0  6356  2176 pts/0    S+   11:41   0:00 grep --color=auto sleep
[root@kali:~/my_project]
# kill 9817
kill: kill 9817 failed: no such process
[root@kali:~/my_project]
# kill <9817>
```

### 3. Process Management

- **ps aux** – Lists all running processes.
- **kill <PID>** – Terminates a specific process by its process ID (PID).

**Purpose:** Monitoring and managing processes is essential for optimizing system performance and preventing resource overuse.

### 4. Network Configuration and Troubleshooting

- **ifconfig** – Displays the system's IP address.
- **ping google.com** – Tests network connectivity.
- **netstat -tulnp** – Lists active network connections.

**Purpose:** Understanding network configurations and diagnosing connectivity issues is critical for maintaining stable and secure communication.

```

Netid          State           Recv-Q           Send-Q           Local Address:Port
[root@kali]-(~/my_project)
└─$ sudo adduser testuser
info: Adding user 'testuser' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group 'testuser' (1002) ...
info: Adding new user 'testuser' (1002) with group 'testuser' (1002) ...
info: Creating home directory '/home/testuser' ...
info: Copying files from '/etc/skel' ...
New password:
Retype new password:
password: password updated successfully
Changing the user information for testuser
Enter the new value, or press ENTER for the default
  Full Name []: trainuser
  Room Number []: 201
  Work Phone []:
  Home Phone []: 98000
  Other []:
Is the information correct? [Y/n] y
info: Adding new user 'testuser' to supplemental / extra groups 'users' ...
info: Adding user 'testuser' to group 'users' ...
[root@kali]-(~/my_project)
└─$ sudo usermod -aG sudo testuser
[root@kali]-(~/my_project)
└─$ su -testuser
su: invalid option -- '-'
Try 'su --help' for more information.
[root@kali]-(~/my_project)
└─$ su testuser
(testuser@kali)-[~]
└─$ whoami
testuser
(testuser@kali)-[~]
└─$ sudo apt update
[sudo] password for testuser:

```

```

* su - testuser
(testuser@kali)-[~]
└─$ whoami
testuser
(testuser@kali)-[~]
└─$ sudo apt update
[sudo] password for testuser:
Get:1 https://dl.google.com/linux/chrome/deb stable InRelease [1,825 B]
Get:2 https://apt.corretto.aws stable InRelease [10.7 kB]
Err:2 https://apt.corretto.aws stable InRelease
  The following signatures were invalid: EXPKEYSIG A122542AB04F24E3 Amazon Services LLC (Amazon Corretto release) <corretto-team@amazon.com>
Get:3 https://dl.google.com/linux/chrome/deb stable/main amd64 Packages [2,212 B]
Ign:5 https://download.docker.com/linux/debian kali-rolling InRelease
Err:6 https://download.docker.com/linux/debian kali-rolling Release
  404 Not Found [IP: 55.8.76.23:443]
Get:4 http://mirr0r.johnnybegood.fr/kali kali-rolling InRelease [44.5 kB]
Get:7 http://mirr0r.johnnybegood.fr/kali kali-rolling/main i386 Packages [19.8 MB]
Get:8 http://mirr0r.johnnybegood.fr/kali kali-rolling/main amd64 Packages [19.8 MB]
Get:9 http://mirr0r.johnnybegood.fr/kali kali-rolling/main all Packages [19.8 MB]
Get:10 http://mirr0r.johnnybegood.fr/kali kali-rolling/main amd64 Contents (deb) [45.4 MB]
Get:11 http://mirr0r.johnnybegood.fr/kali kali-rolling/contrib i386 Packages [93.6 kB]
Get:12 http://mirr0r.johnnybegood.fr/kali kali-rolling/contrib amd64 Packages [111 kB]
Get:13 http://mirr0r.johnnybegood.fr/kali kali-rolling/contrib i386 Contents (deb) [174 kB]
Get:14 http://mirr0r.johnnybegood.fr/kali kali-rolling/contrib amd64 Contents (deb) [258 kB]
Get:15 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free i386 Packages [144 kB]
Get:16 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free amd64 Packages [139 kB]
Get:17 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free i386 Contents (deb) [255 kB]
Get:18 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free amd64 Contents (deb) [657 kB]
Get:19 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free-firmware i386 Packages [10.1 kB]
Get:20 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free-firmware amd64 Packages [10.6 kB]
Get:21 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free-firmware amd64 Contents (deb) [23.4 kB]
Get:22 http://mirr0r.johnnybegood.fr/kali kali-rolling/non-free-firmware i386 Contents (deb) [23.4 kB]
Warning: An error occurred during the signature verification. The repository will be used anyway and the previous index files will be used. GPG error: https://apt.corretto.aws stable InRelease:2: Error: The repository 'https://download.docker.com/linux/debian kali-rolling' does not have a Release file.
Notice: Updating from such a repository can't be done securely, and is therefore disabled by default.
Notice: See apt-secure(8) manpage for repository creation and user configuration details.

(testuser@kali)-[~]
└─$ nano hello.sh
(testuser@kali)-[~]
└─$ chmod +x hello.sh
(testuser@kali)-[~]
└─$ ./hello.sh
hello world
(testuser@kali)-[~]

```

## 1. User and Permission Management

- useradd testuser** – Creates a new user.
- su - testuser** – Switches to the new user.
- usermod -aG sudo testuser** – Grants sudo (administrator) privileges.

**Purpose:** Managing users and their privileges ensures a secure and organized system, preventing unauthorized access.

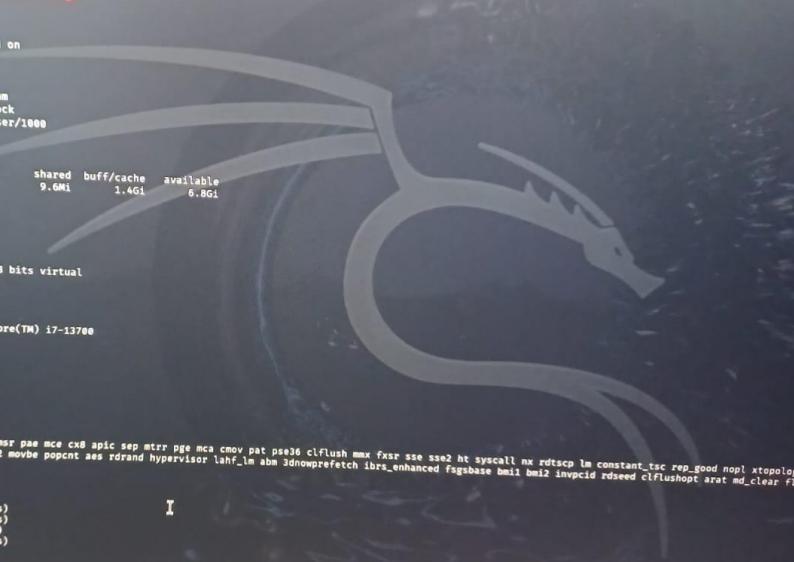


```

(testuser㉿kali)-[~]
$ nano hello.sh
(testuser㉿kali)-[~]
$ chmod +x hello.sh
(testuser㉿kali)-[~]
$ ./hello.sh
hello world
(testuser㉿kali)-[~]
$ exit
logout

(root㉿kali)-[~/my_project]
# tar -cvf my_project.tar my_project/
tar: my_project: Cannot stat: No such file or directory
tar: Exiting with failure status due to previous errors
# mkdir my_project
(root㉿kali)-[~/my_project]
# tar -cvf my_project.tar my_project/
(my_project@kali)-[~/my_project]
# tar -cvf my_project.tar my_project/
(my_project@kali)-[~/my_project]
# gzip my_project.tar
(root㉿kali)-[~/my_project]
# ls -l my_project.tar.gz
-rw-rw-r-- 1 root root 132 Jan 22 12:02 my_project.tar.gz
[root@kali]-[~/my_project]
# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            3.9G   0  3.9G  0% /dev
tmpfs           795M  100K  794M  1% /run
/dev/sda1        79G  41G  34G  55% /
tmpfs           3.9G  252K  3.9G  1% /dev/shm
tmpfs           5.0M   0  5.0M  0% /run/lock
tmpfs           795M  130K  795M  1% /run/user/1000
[root@kali]-[~/my_project]
# free -h
total        used        free      shared  buff/cache   available
Mem:       7.8Gi     1.0Gi    5.6Gi    9.6Mi    1.4Gi     6.8Gi
Swap:      1.0Gi      0B    1.0Gi
[root@kali]-[~/my_project]
# free -h

```



```

# ls -l my_project.tar.gz
-rw-rw-r-- 1 root root 132 Jan 22 12:02 my_project.tar.gz
[root@kali]-[~/my_project]
# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            3.9G   0  3.9G  0% /dev
tmpfs           795M  100K  794M  1% /run
/dev/sda1        79G  41G  34G  55% /
tmpfs           3.9G  252K  3.9G  1% /dev/shm
tmpfs           5.0M   0  5.0M  0% /run/lock
tmpfs           795M  130K  795M  1% /run/user/1000
[root@kali]-[~/my_project]
# free -h
Mem:       7.8Gi     1.0Gi    5.6Gi    9.6Mi    1.4Gi     6.8Gi
Swap:      1.0Gi      0B    1.0Gi
[root@kali]-[~/my_project]
# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Address sizes:         39 bits physical, 48 bits virtual
Byte Order:             Little Endian
CPU(s):                8
On-line CPU(s) list:  0-7
Vendor ID:             GenuineIntel
Model name:            Intel(R) Core(TM) i7-13700
BIOS Model name:       CRD 0 0.00Hz
CPU Family:             6
MHz:                  183
Thread(s) per core:    8
Core(s) per socket:    8
Sockets:               1
Stepping:              1
BogomIPS:              4223.99
Flags:                 fpu vme pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtsck lm constant_tsc rep_good nopl xtopology nonstop_tsc cpuid lahf_lm abm 3dnowprefetch ibrs_ibrs_enhanced fsgsbase bmi2 bmi2_enhanced invpcid rdseed clflushopt arat md_clear flush_lid
Virtualization features:
Hyperervisor vendor:   KVM
Virtualization type:   full
Cache (sum of all):
L1d:                  384 KiB (8 instances)
L1i:                  256 KiB (8 instances)
L2:                   16 MiB (8 instances)
L3:                   240 MiB (8 instances)
NUMA:
NUMA node(s):          1
NUMA node(s) CPU(s):   0-7
Vulnerabilities:
```

## 1.. File Permissions and Execution

- chmod** – Modifies file permissions.
- ls -l** – Lists files with detailed permission attributes.

**Purpose:** Controlling file permissions ensures security by restricting unauthorized access or execution.

## 2.. Shell Scripting

- `echo "Hello, World!" > hello.sh` – Creates a simple shell script.
- `chmod +x hello.sh` – Grants executable permissions.
- `./hello.sh` – Executes the script.

**Purpose:** Shell scripting automates repetitive tasks, increasing efficiency and reducing manual errors.

```
[root@kali:~/my_project]
# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            3.9G   0K  3.9G  0% /dev
tmpfs           795M  100K 794M  1% /run
/dev/sda1        79G  41G  34G  53% /
tmpfs           3.9G  252K  3.9G  1% /dev/shm
tmpfs           5.0M   0K  5.0M  0% /run/lock
tmpfs           795M  136K  795M  1% /run/user/1000

[root@kali:~/my_project]
# free -h
Mem:      total        used        free     shared    buff/cache   available
Swap:     7.8Gi       1.8Gi      5.6Gi      9.6Mi      1.4Gi      6.8Gi

[root@kali:~/my_project]
# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Address sizes:         39 bits physical, 48 bits virtual
Byte Order:             Little Endian
CPU(s):                8
On-line CPU(s) list:  0-7
Vendor ID:             GenuineIntel
Model name:            13th Gen Intel(R) Core(TM) i7-13700
BIOS Model name:       CPU @ 0.0GHz
BIOS CPU family:       0
CPU family:            6
Model:                 183
Threads(s) per core:  1
Core(s) per socket:   8
Socket(s):             1
Stepping:              1
BogomIPS:              4223.99
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_good nopl xtalogify nonstop_tsc cpuid lahf_lm abm 3dnowprefetch ibrs_enhanced fsgsbase bmi1 bmi2 invpcid rdseed clflushopt arat md_clear flush_l1d
Virtualization features:
Hyperervisor vendor: KVM
Virtualization type: full
Caches (sum of all):
L1d:                  384 KiB (8 instances)
L1i:                  256 KiB (8 instances)
L2:                   16 MiB (8 instances)
L3:                   240 MiB (8 instances)
NUMA:
NUMA node(s):          1
NUMA node0 CPU(s):    0-7
Vulnerabilities:
```

```
Model name:          13th Gen Intel(R) Core(TM) i7-13700
BIOS Model name:    CPU @ 0.0GHz
BIOS CPU family:    0
CPU family:          6
Model:               183
Thread(s) per core: 1
Core(s) per socket: 8
Socket(s):          1
Stepping:            1
BogomIPS:            4223.99
Flags:                fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_good nopl xtalogify nonstop_tsc cpuid lahf_lm abm 3dnowprefetch ibrs_enhanced fsgsbase bmi1 bmi2 invpcid rdseed clflushopt arat md_clear flush_l1d
Virtualization features:
Hypervisor vendor: KVM
Virtualization type: full
Caches (sum of all):
L1d:                  384 KiB (8 instances)
L1i:                  256 KiB (8 instances)
L2:                   16 MiB (8 instances)
L3:                   240 MiB (8 instances)
NUMA:
NUMA node(s):          1
NUMA node0 CPU(s):    0-7
Vulnerabilities:
Gather data sampling: Not affected
Itlb multihit:        Not affected
L1tf:                 Not affected
Mds:                  Not affected
Meltdown:             Not affected
Mpxio stale data:     Not affected
Retbleed:              Mitigation: Enhanced IBRS
Spec rstack overflow: Not affected
Spec store bypass:    Vulnerable
Spectre v1:            Mitigation: usercopy/swaps barriers and __user pointer sanitization
Spectre v2:            Mitigation: Enhanced / Automatic IBRS, RSB filling, PBRSB-eIBRS SW sequence
Srbds:                Not affected
Tsx async abort:       Not affected
[root@kali:~/my_project]
#
```

## 1. System Monitoring

- **df -h** – Displays disk usage.
- **free -m** – Checks memory (RAM) usage.
- **lscpu** – Shows CPU details.

**Purpose:** Monitoring system resources helps in performance tuning and early detection of issues like disk space shortages or memory leaks.

## Final Findings and Takeaways

The completion of these **Linux administrative tasks** provided a hands-on understanding of essential system management principles. Each task demonstrated the importance of **efficient system navigation, user and file management, process control, package handling, network troubleshooting, and system monitoring**. These fundamental skills are crucial for anyone working with Linux, whether in **system administration, cybersecurity, or software development**.

One of the key takeaways from this study is the significance of **file permissions and user management** in maintaining system security. By properly assigning permissions and restricting access, organizations can prevent unauthorized modifications and ensure data integrity. Similarly, **process management and system monitoring** are vital for optimizing performance and diagnosing issues before they escalate.

Another crucial aspect is **network configuration and troubleshooting**, which ensures smooth communication between devices. The ability to retrieve IP addresses, verify connectivity, and analyze network connections helps in identifying and resolving network-related problems efficiently.

Furthermore, **automation through shell scripting** simplifies repetitive tasks, reducing human errors and improving productivity. Shell scripts allow administrators to execute complex operations with minimal manual intervention, making them an invaluable tool for system maintenance.

The use of **archiving and compression** techniques enhances data management by enabling efficient storage and transfer of files. This is particularly useful for **backup strategies and disaster recovery planning**.

Additionally, **package management** plays a significant role in keeping the system up to date with the latest software and security patches. Regular updates ensure the system remains **stable, secure, and free from vulnerabilities**.