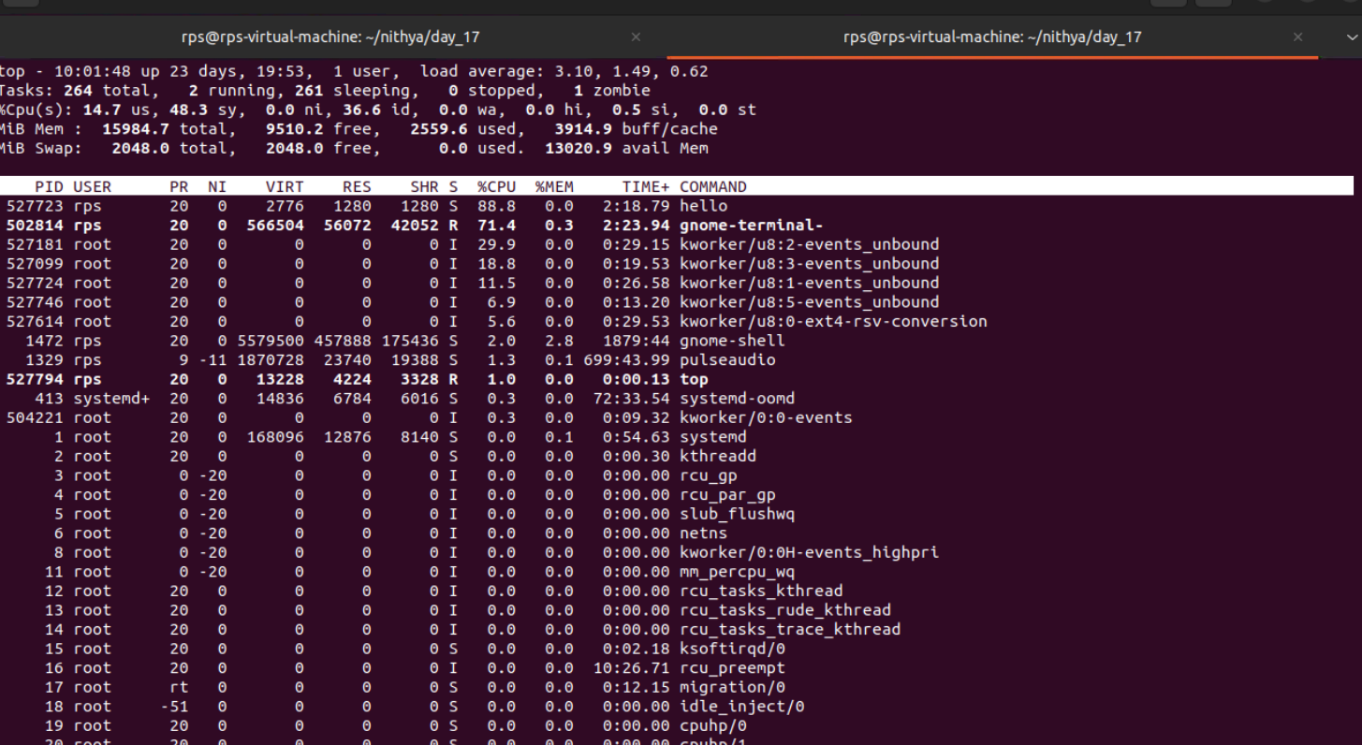
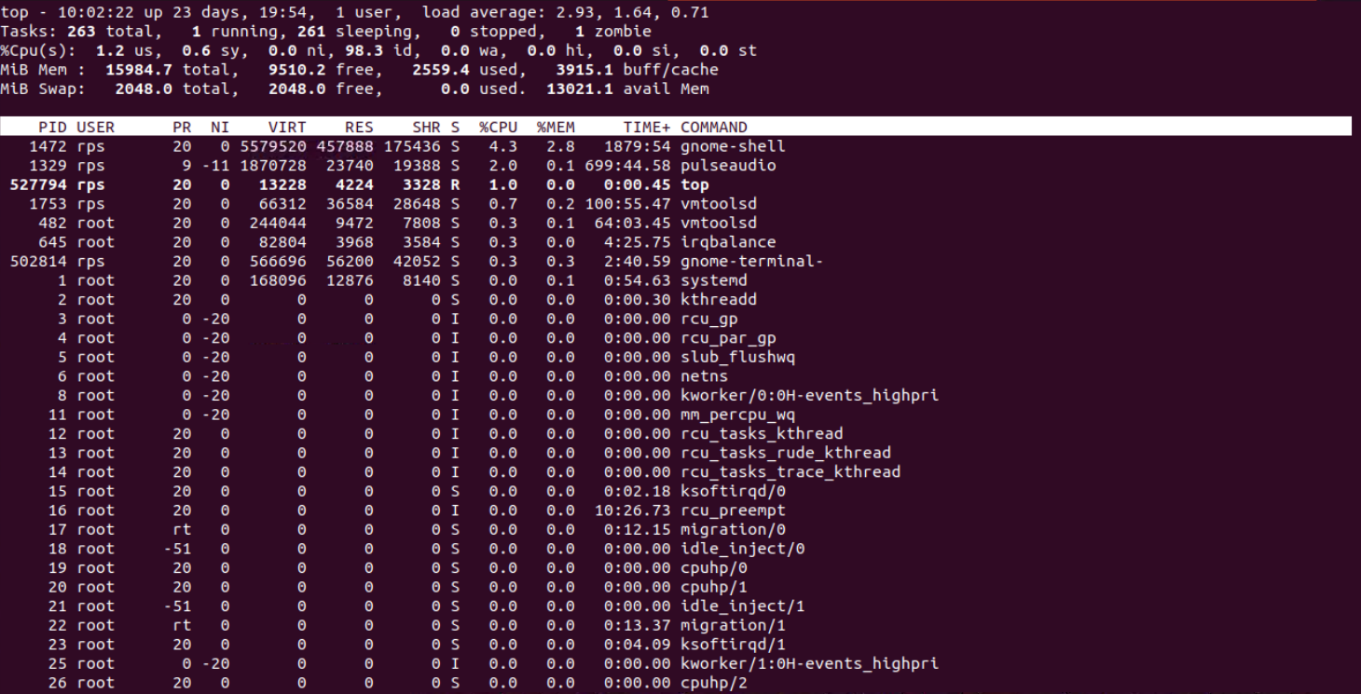
Task

Top command : check process is running or not

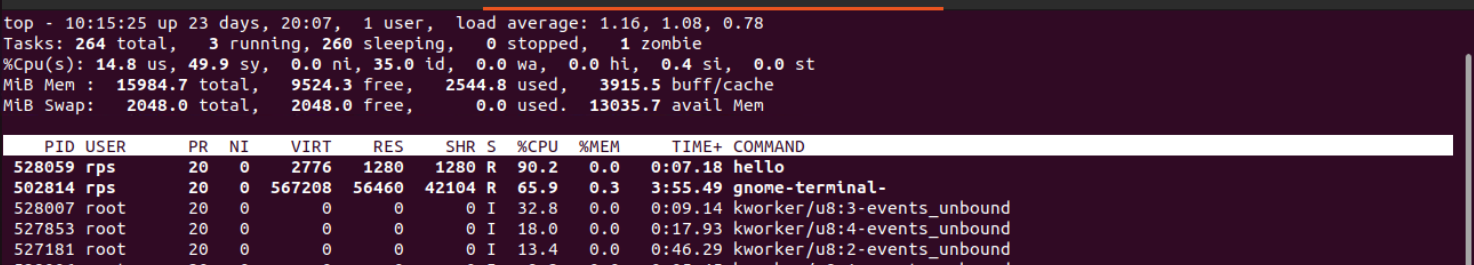


After process stop



Kill command to stop the process

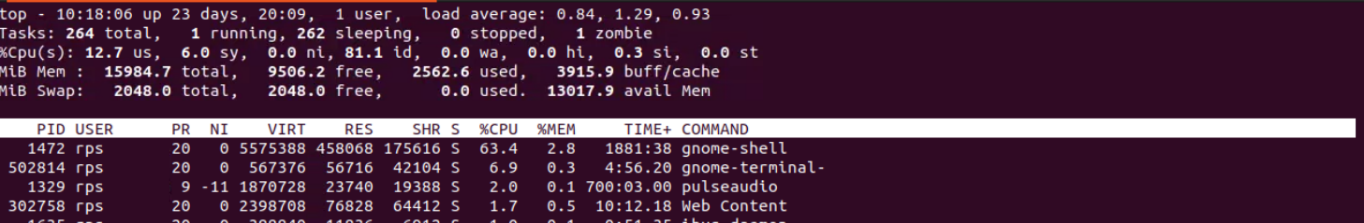
Before kill the process



Killing process

Capture2.PNG

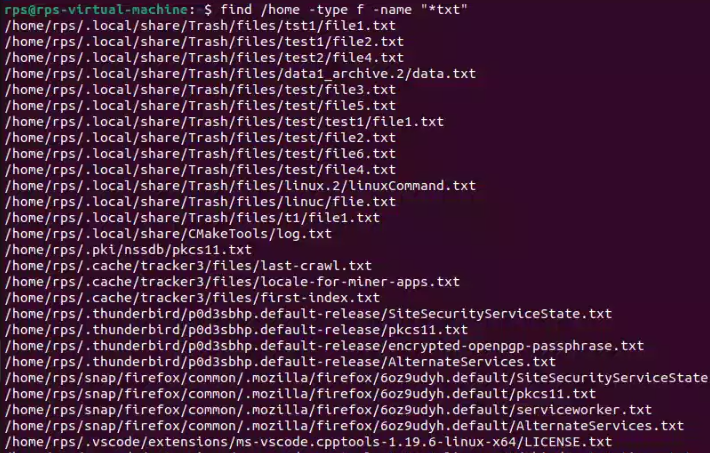
After kill the process



### Commands

**File Search**:  
Write a command to find all files with the extension .txt in the /home directory and its subdirectories.

find /home -type f -name "\*.txt"



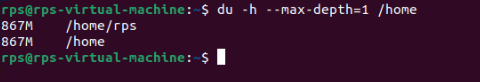
**File Permissions**:  
Write a command to change the permissions of all files in the /var/log directory to 644.

find /var/log -type f -execchmod 644 {} \;

Capture2.PNG

**Disk Usage**:  
Write a command to display the disk usage of all directories in the /home directory in a human-readable format.

du -h --max-depth=1 /home



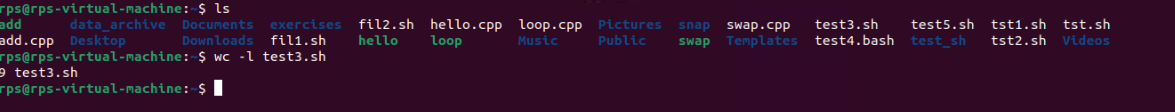
**Process Management**:  
Write a command to list all running processes that contain the name "apache" in their command line.

ps aux | grep apache

Capture4.PNG

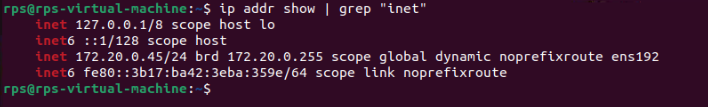
**Text Processing**:  
Write a command to count the number of lines in a file named error.log.

wc -l error.log



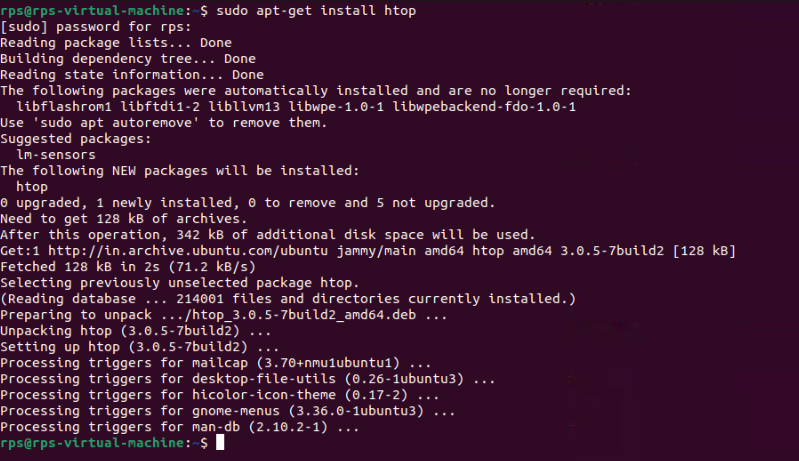
**Network Configuration**:  
Write a command to display the IP address of all network interfaces on the system.

ipaddr show | grep "inet "



**Package Management**:  
Write a command to install a package named htop using the package manager.

sudo apt-get install htop



**User Management**:  
Write a command to add a new user named developer to the system.

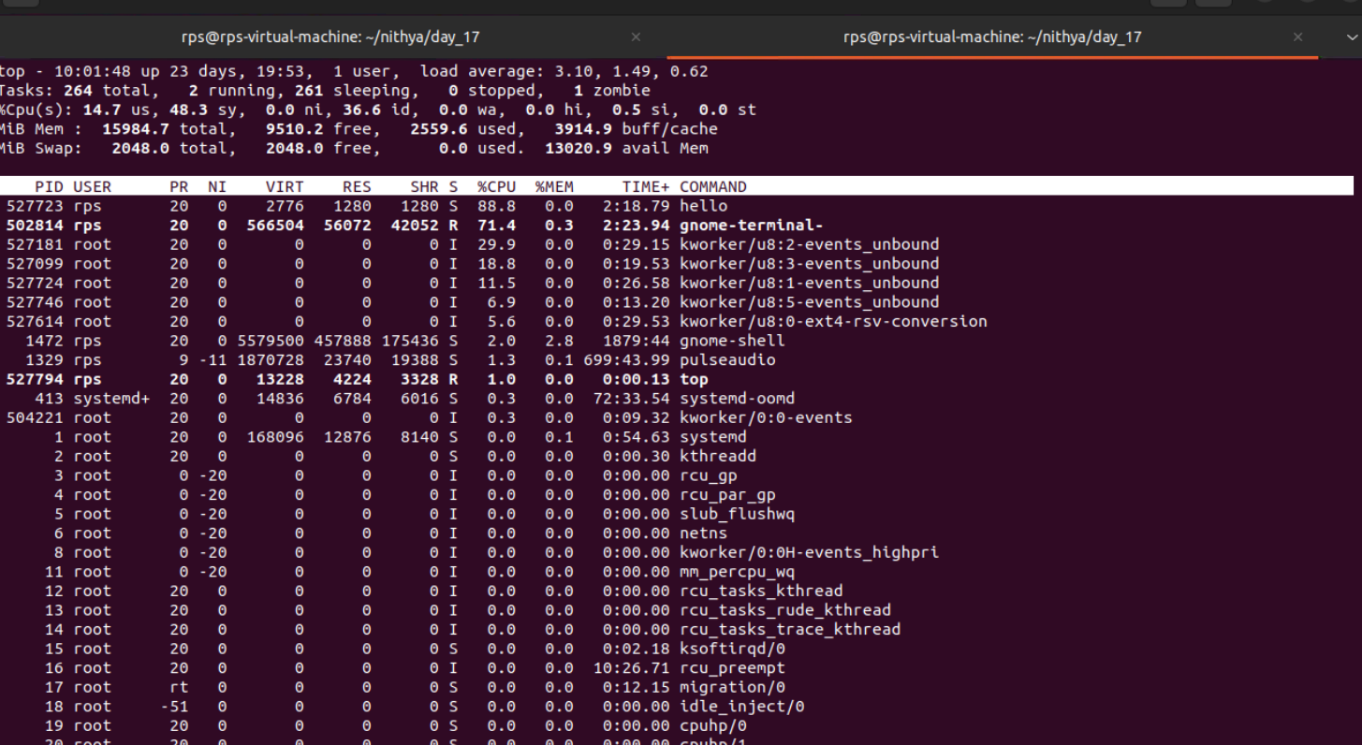
Sudo useradd developer

**File Compression**:  
Write a command to compress a directory named backup into a .tar.gz file.

tar -czvf backup.tar.gz backup

**System Monitoring**:  
Write a command to display real-time system resource usage, including CPU, memory, and disk I/O.

Top



### Shell Scripts

**Backup Script**:  
Write a shell script to back up a directory named /data to /backup with the current date appended to the backup file name.

#!/bin/bash

src="/data"

dest="/backup"

date=$(date +%Y-%m-%d)

backup\_file="backup\_$date.tar.gz"

tar -czvf"$dest/$backup\_file""$src"

**Log Rotation**:  
Write a shell script to rotate log files in the /var/log directory, keeping only the last 7 days of logs.

#!/bin/bash

find /var/log -type f -mtime +7 -execrm -f {} \;

**User Report**:  
Write a shell script to generate a report of all users currently logged into the system and save it to a file named user\_report.txt.

#!/bin/bash

who> user\_report.txt

**Disk Space Alert**:  
Write a shell script to check the disk usage of the /home directory and send an email alert if the usage exceeds 80%.

#!/bin/bash

usage=$(df /home | awk 'NR==2 {print $5}' | sed's/%//')

if [ "$usage" -gt80 ]; then

echo"Disk usage of /home exceeds 80%" | mail -s "Disk Space Alert" your\_email@example.com

fi

**Service Monitor**:  
Write a shell script to check if the nginx service is running and restart it if it is not.

#!/bin/bash

if !systemctl is-active --quiet nginx; then

systemctl restart nginx

fi

**File Cleanup**:  
Write a shell script to delete all files older than 30 days in the /tmp directory.

#!/bin/bash

find /tmp -type f -mtime +30 -execrm -f {} \;

**Automated Updates**:  
Write a shell script to automatically update all installed packages on the system.

#!/bin/bash

sudo apt-get update &&sudo apt-get upgrade -y

**Database Backup**:  
Write a shell script to back up a MySQL database named sales to a file named sales\_backup.sql.

#!/bin/bash

mysqldump -u your\_username -p your\_password sales >sales\_backup.sql

**System Information**:  
Write a shell script to display system information, including hostname, OS version, and kernel version.

#!/bin/bash

echo"Hostname: $(hostname)"

echo"OS Version: $(lsb\_release -d | cut -f2)"

echo"Kernel Version: $(uname -r)"

**Cron Job**:  
Write a shell script to schedule a cron job that runs a specific command every day at midnight.

#!/bin/bash

# Command to schedule a cron job that runs a specific command every day at midnight

(crontab -l 2>/dev/null; echo"0 0 \* \* \* /path/to/your/command") | crontab -