# Practical-4

# PART A

| AIM | List down all processes with their states sorted by their CPU Usage. Identify current  running process. |
| --- | --- |
| Command | Ps |
| Output |  |
| AIM | List down all processes associated with current user. |
| Command | Ps -u |
| Output |  |
| AIM | List down all processes associated with their terminal and their states. Identify current  running process. |
| Command | Ps aux |
| Output |  |
| AIM | Compare the output of “ps lx” and “ps l” commands. |
| Command | Ps lx  Ps l |
| Output |  |
| AIM | List down all the names and numbers of all available signals. |
| Command | Kill -l |
| Output |  |
| AIM | Run the “sleep 10000” in background. (i.e. sleep 10000 &) |
| Command | Sleep 10000 & |
| Output |  |
| AIM | Check the PID of sleep process and kill it using PID. |
| Command | Kill 4411 |
| Output |  |
| AIM | Apply w command and observer the output |
| Command | w |
| Output |  |
| AIM | Open the firefox browser. Check the processes associated with firefox. |
| Command | Pgrep firefox |
| Output |  |
| AIM | Kill all processes associated with firefox by its name. |
| Command | Kill 3012 |
| Output |  |
| AIM | Give the difference between kill and pkill. |
| Command | Kill  Pkill |
| Output | pkill is similar to kill, but it allows you to send signals to processes based on their name or other attributes. |
| AIM | Run “lscpu” command and observer the output. |
| Command | lscpu |
| Output |  |

# 

# PART B

### Aim : Control Services and daemons

| 1. a | List all services on your system.(systemctl list-units --type=service) |
| --- | --- |
| Output |  |
| 1. b | Check whether the ssh service is active or not. (sudosystemctl status service\_name) |
| Output |  |
| 2. | If the package is not available, i nstall ssh package (sudo apt-get install ssh) |
| Output |  |
| 3. | If the service is available and active, check the process state usng ps –p PID |
| Output |  |
| 4. | Add the firewall rule to allow remote service using ssh(sudo ufw allow ssh) |
| Output |  |
| 5. | Check your IP address |
| Output |  |
| 6. | Access another user terminal using ssh |
| Output |  |
| 7. | Stop the service and check the status |
| Output |  |
| 8. | Disable the service and check the status |
| Output |  |
| 9. | Enable it again and check the status |
| Output |  |
| 10. | Restart the service and check the status |
| Output |  |
| 11. | Observe the analyze the output of below mentioned command  1. systemctl is-active ssh  2. systemctl is-enabled ssh  3. systemctl is-failed ssh |
| Output |  |

# Part C

### Aim: Improve Command Line Productivity

| 1. | Create a file named “newfile.txt” and insert a text into created file as follow:  The operating system is a system program that serves as an interface between the  computing system and the end-user. |
| --- | --- |
| Output |  |
| 2. | Redirect the output of “newfile.txt” file to file “new.txt” using command. |
| Output |  |
| 3. | Type command cat, then enter key and enter some text. Observe the output. |
| Output |  |
| 4. | Type command i) cat <newfile.txt ii) cat newfile.txt. Interpret the output in both  cases. |
| Output | i)  ii) |
| 5. | Type command cat – and enter any text. |
| Output |  |
| 6. | Use both redirection operator < and > at once to redirect the output of one file to  another. |
| Output |  |
| 7. | Summarize the use of cat command with redirection operator based on your done  exercise. |
| Output | Cat command has various uses such as coping file from another, displaying text and seeing content of the file. |
| 8. | Try following command and interpret the output:  a) ls >filelist  b) cat newfile.txt new.txt >> report  c) cat newfile.txt > newfile.txt  d) date; who  e) date; who>logfile  f) (date; who) > logfile |
| Output |  |