

## i) Process Management in Linux

Linux is a multi-user and multi-tasking operating system. This means that many users and many programs can run simultaneously. Each running program is treated as a process and is assigned a unique number called process ID (PID).

Process consumes system resources such as:

- 1] CPU time
- 2]. Main memory (RAM)
- 3] Disk and I/O resource.

When unnecessary or faulty processes continue running they increase system load and reduce performance. To manage this Linux provides command like:

- ① ps                    ③ kill                    ⑤ pkill
- ② top                    ④ killall

## 2) Identifying running processes.

Before killing any process, Adam must first identify which processes are running and which ones are consuming more system resources. This can be done using the following command.

### a) ps command.

The ps (process status) command displays information about active processes.

Ex - Ps -ef

This command shows:

PID, user who started the process, process name, CPU usage.

### 5) top Command

The top command displays process in real-time and shows CPU usage, memory usage running processes. By using Ps or top, admin can decide which process should be terminated to reduce system load.

### 3) Killing process by name

#### a) killall command

The killall command is used to terminate all processes having the same name.

Syntax - killall process-name.

The only - killall sends a terminates signal to processes with the specified name. It is useful when multiple copies of the same program are running. It reduces system load quickly by stopping all related processes at once.

Ex :- killall chrome

#### b) killing a process based on process name.

Linux also provides the pkill command to kill process using their name or pattern.

Syntax - pkill process-name.

Theory - pkill terminates process by matching their names. It is more flexible than killall because it can match partial names or patterns. It allows user to stop process without knowing their PID.

Ex - pkill firefox.

5) Killing a single process using process PID

- Sometimes Adam wants to terminate only one specific process without affecting others. For this purpose, Linux uses the kill command along with the process ID.

Step 1 - Find the PID

Adam first finds the PID of the process using:

ps -ep | grep process-name.

Step 2 - kill the process using PID.

Kill PID.

Theory - The kill command sends a termination signal to the process.

The process is given a chance to close properly.

Ex - kill 2456.

6) Force killing a process

Sometimes a process not respond to normal kill commands.

It may be stuck or frozen. In such case, Adam uses a force kill command.

Syntax - kill -9 PID

Name of Practical

Theory. - It immediately terminates the process. The process does not get a chance to save some data or clean resource. It should be used only when normal kill fails.

- 7] Signals in Linux process termination Linux uses signals to control processes

Important signals are:

① SIGTERM (15) : Normal termination.

② SIGKILL (9) : Force Termination.

③ SIGSTOP : Pause a process.

④ SIGCONT : continue a stopped process.

The kill command sends these signals to process.

- 8] Importance of killing processes.

Killing unwanted process is important for:

① Reducing CPU load.

② Freeing memory

③ Improving system performance.

④ Preventing system crashes.

⑤ Managing server stability.

System administrator uses these commands daily to keep healthy and responsive.

Teacher's Signature \_\_\_\_\_