

PROCESS MANAGEMENT

A process refers to a program in execution; it's a running instance of a program. It is made up of the program instruction, data read from files, other programs or input from a system user.

Types of processes :-

There are fundamentally two types of processes in linux :-

- Foreground processes :- It is also referred to as interactive processes. These are initialized and controlled through a terminal session. In other words, there has to be a user connected to the system to start such processes; they haven't started automatically as part of the system functions / services.
- Background processes :- It is also referred to as non-interactive / automatic processes. These are processes not connected to a terminal. They don't expect any user input.

Running a foreground process :- Once you run a command (for ex :- ~~cat cloudcmd~~ ^{sleep 5}), it will start a process in the system. It will be connected to the terminal and a user can send input to it.

Running a background process :- To start a process in the background (non-interactive) use the & symbol here, the process doesn't read input from a user until it's moved to the foreground.

System

\$ sleep 5 &

States of a process :- During execution, a process changes from one state to another depending on its environment circumstances. In Linux, a process has the following possible states.

- Running :- it's either running or it's ready to run
- Waiting :- a process is waiting for an event to occur or for a system resource.
- Stopped :- a process has been stopped, usually by receiving a signal.
- Zombie :- a process is dead, it has been halted but it's still has an entry in the process table.

Example of Foreground Processes

1. \$ sleep 5

This command will be executed in the terminal and we would be able to execute another command after execution of above command.

The name of the process is sleep 5 but ^{we may} ~~you~~ change it as per our needs.

To stop a foreground process in between of its execution

2. To get the list of jobs that are either running or stopped
\$ jobs
3. To run all the pending and force stopped jobs in the background
\$ bg
4. To get details of a process running in background
\$ ps -ef | grep sleep
5. to run all the pending and force stopped jobs in the foreground.
\$ fg

Important process management commands

Commands

Description

bg	→	to send a process to the background
fg	→	to run a stopped process in the ^{fore} background
top	→	Details on all active processes
ps	→	give the status of processes running for a user
ps PID	→	gives the status of a particular process
pidof	→	gives the process ID of a process
kill PID	→	Kills a process
nice	→	starts a process with a given priority
renice	→	changes priority of an already running process.
df	→	gives free hard disk on your system
free	→	gives free RAM on your system.

User Management :- User management includes everything from creating a user to deleting a user on your system. User management can be done in three ways on a Linux system. Command line tools like `useradd`, `userdel`, `passwd` etc, are widely used by the server administrators.

1. Adduser

useradd :- with `useradd` command you can add a user

```
$ sudo adduser <username>
```

2. Disable account :- for disabling an account, remove the password set on the account.

```
$ sudo passwd -l <username>
```

3. Delete an account

```
$ sudo userdel -r <username>
```

4. Add user to a usergroup

```
$ sudo usermod -a -G <groupname> <username>
```

5. Check user is in group

```
$ cat /etc/group
```

6. Remove user from a group

\$ sudo deluser <username> <groupname>

7. finger

\$ finger

Gives information on all logged in user

8. \$ finger username

Gives information of a particular user.