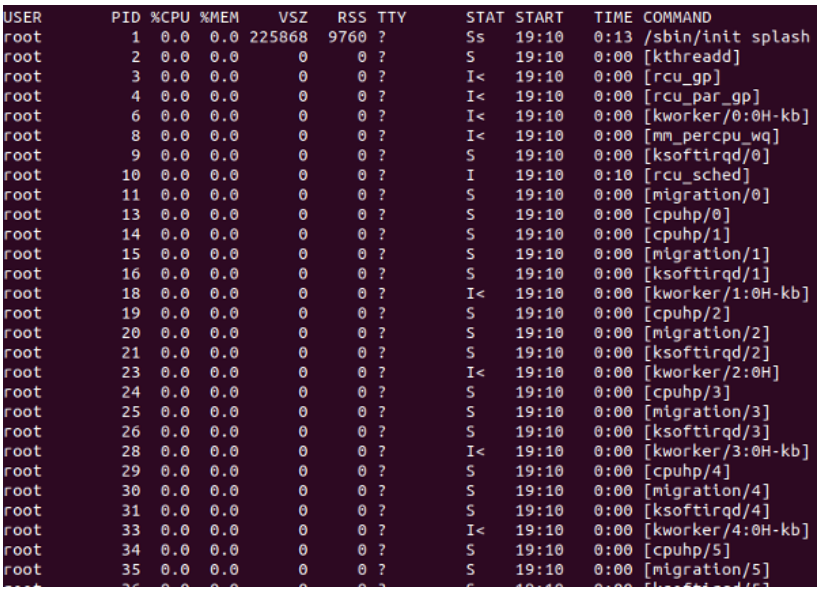
NOTE: Please note that **vivek@nixcraft:~$** is my shell prompt. You need to type commands after the $ prompt.

How to manage processes from the Linux terminal

The **ps** command is a traditional Linux command to lists running processes. The following command shows all processes running on your Linux based server or system:

```
vivek@nixcraft:~$ ps -aux
vivek@nixcraft:~$ sudo ps -a
```



```
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.0 225868  9760 ?        Ss   19:10   0:13 /sbin/init splash
root           2  0.0  0.0      0     0 ?        S    19:10   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   19:10   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   19:10   0:00 [rcu_par_gp]
root           6  0.0  0.0      0     0 ?        I<   19:10   0:00 [kworker/0:0H-kb]
root           8  0.0  0.0      0     0 ?        I<   19:10   0:00 [mm_percpu_wq]
root           9  0.0  0.0      0     0 ?        S    19:10   0:00 [ksoftirqd/0]
root          10  0.0  0.0      0     0 ?        I    19:10   0:10 [rcu_sched]
root          11  0.0  0.0      0     0 ?        S    19:10   0:00 [migration/0]
root          13  0.0  0.0      0     0 ?        S    19:10   0:00 [cpuhp/0]
root          14  0.0  0.0      0     0 ?        S    19:10   0:00 [cpuhp/1]
root          15  0.0  0.0      0     0 ?        S    19:10   0:00 [migration/1]
root          16  0.0  0.0      0     0 ?        S    19:10   0:00 [ksoftirqd/1]
root          18  0.0  0.0      0     0 ?        I<   19:10   0:00 [kworker/1:0H-kb]
root          19  0.0  0.0      0     0 ?        S    19:10   0:00 [cpuhp/2]
root          20  0.0  0.0      0     0 ?        S    19:10   0:00 [migration/2]
root          21  0.0  0.0      0     0 ?        S    19:10   0:00 [ksoftirqd/2]
root          23  0.0  0.0      0     0 ?        I<   19:10   0:00 [kworker/2:0H]
root          24  0.0  0.0      0     0 ?        S    19:10   0:00 [cpuhp/3]
root          25  0.0  0.0      0     0 ?        S    19:10   0:00 [migration/3]
root          26  0.0  0.0      0     0 ?        S    19:10   0:00 [ksoftirqd/3]
root          28  0.0  0.0      0     0 ?        I<   19:10   0:00 [kworker/3:0H-kb]
root          29  0.0  0.0      0     0 ?        S    19:10   0:00 [cpuhp/4]
root          30  0.0  0.0      0     0 ?        S    19:10   0:00 [migration/4]
root          31  0.0  0.0      0     0 ?        S    19:10   0:00 [ksoftirqd/4]
root          33  0.0  0.0      0     0 ?        I<   19:10   0:00 [kworker/4:0H-kb]
root          34  0.0  0.0      0     0 ?        S    19:10   0:00 [cpuhp/5]
root          35  0.0  0.0      0     0 ?        S    19:10   0:00 [migration/5]
```

The process ID (PID) is essential to kill or control process on Linux. For example consider the following outputs:

```
root         1  0.0  0.0 225868  9760 ?        Ss   19:10   0:13 /sbin/init splash
```

Where,

1. `root` – User name
2. `1` – PID (Linux process ID)
3. `19:10` – Process start time
4. `/sbin/init splash` – Actual process or command

There may be too many processes. Hence, it uses the following less command/more command as pipe to display process one screen at a time:

```
vivek@nixcraft:~$ ps -aux | more
vivek@nixcraft:~$ sudo ps -aux | less
```

Press `q` to exit from above Linux pagers. You can search for a particular Linux process using grep command/egrep command:

```
vivek@nixcraft:~$ ps aux | grep firefox
vivek@nixcraft:~$ sudo ps aux | grep vim
```

```
vivek@nixcraft:~$ sudo ps -aux | egrep 'sshd|openvpn|nginx'
```

Linux pgrep command

Many variants of Linux comes with the **pgrep** command to search/find process. The syntax is:

```
vivek@nixcraft:~$ pgrep {process}
vivek@nixcraft:~$ sudo pgrep sshd
vivek@nixcraft:~$ pgrep vim
vivek@nixcraft:~$ pgrep firefox
vivek@nixcraft:~$ pgrep -l nginx
```

The `-l` option passed to the pgrep command to display long format and process name too.

Linux top command

The **top** command is another highly recommended method to see your Linux servers resource usage. One can see a list of top process that using the most memory or CPU or disk.

```
vivek@nixcraft:~$ top
vivek@nixcraft:~$ sudo top
vivek@nixcraft:~$ sudo top [options]
```

Press `q` to exit from the top session and `h` to get help.

Linux htop command

The **htop** command is an interactive process viewer and recommended method for Linux users. One can see a list of top process that using the most memory or CPU or disk and more:

```
vivek@nixcraft:~$ htop
vivek@nixcraft:~$ sudo htop
vivek@nixcraft:~$ sudo htop [options]
```

See how to install htop on a CentOS/RHEL system for more info.

Linux kill command

Want to kill a process? Try **kill** command. The syntax is:

```
vivek@nixcraft:~$ kill pid
vivek@nixcraft:~$ kill -signal pid
```

Find PID using ps, pgrep or top commands. Say you want to kill a PID # 16750, run:

```
vivek@nixcraft:~$ kill 16750
```

For some reason if the process can not be killed, try forceful killing:

```
vivek@nixcraft:~$ kill -9 16750
```

OR

```
vivek@nixcraft:~$ kill -KILL 16750
```

Linux pkill command

If you wish to kill a process by name, try pkill command. The syntax is:

```
vivek@nixcraft:~$ pkill processName
vivek@nixcraft:~$ pkill vim
vivek@nixcraft:~$ pkill firefox
vivek@nixcraft:~$ pkill -9 emacs
vivek@nixcraft:~$ sudo pkill -KILL php7-fpm
```

Linux killall command

The **killall** command kills processes by name, as opposed to the selection by PID as done by kill command:

```
vivek@nixcraft:~$ killall vim
vivek@nixcraft:~$ killall -9 emacs
```

Linux nice and renice command

The primary purpose of the **nice** command is to run a process/command at a lower or higher priority. Use the **renice** command to alter the nice value of one or more running Linux processes. The nice value can range from -20 to 19, with 19 being the lowest priority. Say, you want to compile software on a busy Linux server. You can set a very low priority, enter:

```
vivek@nixcraft:~$ nice -n 13 cc -c *.c &
```

Set a very high priority for a kernel update. Before rebooting Linux server, run:

```
nice --10 wall <<end
System reboots in 5 minutes for Linux kernel update!
Save all your work!!!

-- Sysadmin
end
```

To change the priority of a running process, type the following:

```
vivek@nixcraft:~$ renice {Priority} -p {PID}
vivek@nixcraft:~$ renice {Priority} {PID}
vivek@nixcraft:~$ pgrep vim
renice 10 69947
vivek@nixcraft:~$ sudo renice -10 $(pgrep vim)
```

Conclusion

This page showed how to manage the process on the Linux terminal. For further information see man pages or our example pages:

- ps command
- pgrep command
- top command
- kill command
- kilall command
- nice command
- renice command

UP NEXT


How to check running process in Ubuntu Linux using command line


Debian / Ubuntu Linux Delete Old Kernel Images Command


Kill Process in Linux or Terminate a Process in UNIX / Linux Systems


How to Compile and Install Linux Kernel v4.9.11 Source On a Debian / Ubuntu Linux


How to check running process in Ubuntu using command line


Linux / UNIX: Kill Command Examples


How to check running process in Unix using command line

Category	List of Unix and Linux commands
File Management	cat
Firewall	Alpine Awall • CentOS 8 • OpenSUSE • RHEL 8 • Ubuntu 16.04 • Ubuntu 18.04 • Ubuntu 20.04

Category	List of Unix and Linux commands
Network Utilities	dig • host • ip • nmap
OpenVPN	CentOS 7 • CentOS 8 • Debian 10 • Debian 8/9 • Ubuntu 18.04 • Ubuntu 20.04
Package Manager	apk • apt
Processes Management	bg • chroot • cron • disown • fg • jobs • killall • kill • pidof • pstree • pwdx • time
Searching	grep • whereis • which
User Information	groups • id • lastcomm • last • lid/libuser-lid • logname • members • users • whoami • who • w
WireGuard VPN	CentOS 8 • Debian 10 • Firewall • Ubuntu 20.04

1 comment… add one ↓

Wendy
Jan 29, 2018 @ 18:54

htop is the best tool, but not install on my system by default.

reply link

Leave a Reply

Your email address will not be published.

Comment

Name

Email

Website

Post Comment

Use HTML <pre>...</pre>, <code>...</code> and <kbd>...</kbd> for code samples.

Next FAQ: How to check list of users in Unix

Previous FAQ: How to exclude file when using scp command recursively

How to check running process in Linux using command line

Author: Vivek Gite
Last updated: May 20, 2019
1 comment

I am a new system administrator for the Linux operating system. How do I check running process in Linux using the command line option?

One can use the Linux command line or terminal app to display a running process, change their priorities level, delete process and more. This page shows how to use various commands to list, kill and manage process on Linux.

Check running process in Linux

The procedure to monitor the running process in Linux using the command line is as follows:

1. Open the terminal window on Linux
2. For remote Linux server use the ssh command for log in purpose
3. Type the ps aux command to see all running process in Linux
4. Alternatively, you can issue the top command or htop command to view running process in Linux

Let us see some example and usage in details.