

[LOG IN](#)[SIGN UP](#)

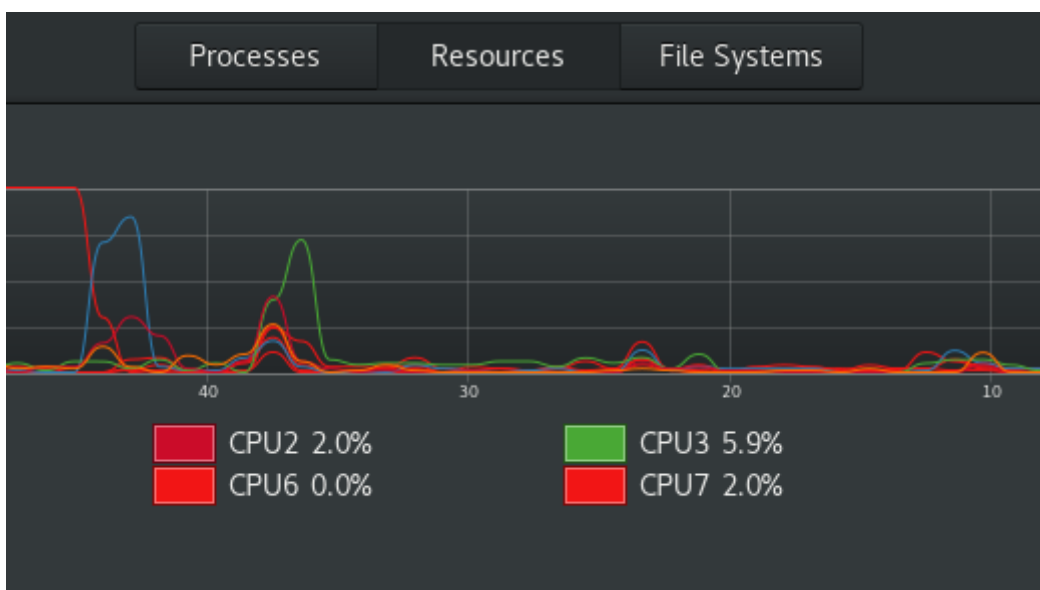
## Main menu

[Articles](#)[Resources](#)[Downloads](#)[About](#)[Open Organization](#)

# Monitor your Linux system in your terminal with procps-ng

How to find the process ID (PID) of a program. The most common Linux tools for this are provided by the procps-ng package, including the ps and pstree, pidof, and pgrep commands.

11 Aug 2021 | [Seth Kenlon \(Red Hat\) \(/users/seth\)](#) | 48 | [1 comment](#)



We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement](#). By using this website you agree to our use of cookies.



A process, in [POSIX \(https://opensource.com/article/19/7/what-posix-richard-stallman-explains\)](https://opensource.com/article/19/7/what-posix-richard-stallman-explains) terminology, is an ongoing event being managed by an operating system's kernel. A process is spawned when you launch an application, although there are many other processes running in the background of your computer, including programs to keep your system time accurate, to monitor for new filesystems, to index files, and so on.

## More Linux resources

- [Linux commands cheat sheet \(https://developers.redhat.com/cheat-sheets/linux-commands-cheat-sheet/?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://developers.redhat.com/cheat-sheets/linux-commands-cheat-sheet/?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [Advanced Linux commands cheat sheet \(https://developers.redhat.com/cheat-sheets/advanced-linux-commands/?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://developers.redhat.com/cheat-sheets/advanced-linux-commands/?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [Free online course: RHEL Technical Overview \(https://www.redhat.com/en/services/training/rh024-red-hat-linux-technical-overview?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://www.redhat.com/en/services/training/rh024-red-hat-linux-technical-overview?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [Linux networking cheat sheet \(https://opensource.com/downloads/cheat-sheet-networking?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://opensource.com/downloads/cheat-sheet-networking?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [SELinux cheat sheet \(https://opensource.com/downloads/cheat-sheet-selinux?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://opensource.com/downloads/cheat-sheet-selinux?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [Linux common commands cheat sheet \(https://opensource.com/downloads/linux-common-commands-cheat-sheet?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://opensource.com/downloads/linux-common-commands-cheat-sheet?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [What are Linux containers? \(https://opensource.com/resources/what-are-linux-containers?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://opensource.com/resources/what-are-linux-containers?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)
- [Our latest Linux articles \(https://opensource.com/tags/linux?intcmp=70160000000h1jYAAQ&utm\\_source=intcallout&utm\\_campaign=linuxconter\)](https://opensource.com/tags/linux?intcmp=70160000000h1jYAAQ&utm_source=intcallout&utm_campaign=linuxconter)

Most operating systems have a system activity monitor of some kind so you can learn what processes are running at any given moment. Linux has a few favorites



your system in your terminal. Regardless of which you choose, it's a common task for those who take an active role in managing their computer is to examine a specific process.

In this article, I demonstrate how to find the process ID (PID) of a program. The most common tools for this are provided by the [procps-ng](https://gitlab.com/procps-ng) (<https://gitlab.com/procps-ng>) package, including the `ps` and `pstree`, `pidof`, and `pgrep` commands.

## Find the PID of a running program

Sometimes you want to get the process ID (PID) of a specific application you know you have running. The `pidof` and `pgrep` commands find processes by command name.

The `pidof` command returns the PIDs of a command, searching for the exact command by name:

```
$ pidof bash
1776 5736
```

The `pgrep` command allows for regular expressions (regex):

```
$ pgrep .sh
1605
1679
1688
1776
2333
5736
$ pgrep bash
5736
```

## Find a PID by file

You can find the PID of the process using a specific file with the `fuser` command.

## Get a process name by PID

If you have the PID *number* of a process but not the command that spawned it, you can do a "reverse lookup" with `ps`:

```
$ ps 3234
  PID TTY          STAT       TIME COMMAND
 5736 pts/1    Ss          0:00      emacs
```

## List all processes

The `ps` command lists processes. You can list every process on your system with the `-e` option:

```
$ ps -e | less
  PID TTY          TIME CMD
    1 ?            00:00:03 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 rcu_gp
    4 ?            00:00:00 rcu_par_gp
    6 ?            00:00:00 kworker/0:0H-events_highpri
[...]
```

```
5648 ?            00:00:00 gnome-control-c
5656 ?            00:00:00 gnome-terminal-
5736 pts/1        00:00:00 bash
5791 pts/1        00:00:00 ps
5792 pts/1        00:00:00 less
(END)
```

## List just your processes

The output of `ps -e` can be overwhelming, so use `-u` to see the processes of just one user:

```
$ ps -U tux | less
  PID TTY          TIME CMD
 3545 ?            00:00:00 systemd
 3548 ?            00:00:00 (sd-pam)
 3566 ?            00:00:00 pulseaudio
```

```

3592 tty2      00:00:00 gnome-session-b
3613 ?         00:00:00 gvfsd
3618 ?         00:00:00 gvfsd-fuse
3665 tty2      00:01:03 gnome-shell
[...]
```

That produces 200 fewer (give or take a hundred, depending on the system you're running it on) processes to sort through.

You can view the same output in a different format with the `pstree` command:

```

$ pstree -U tux -u --show-pids
[...]
```

```

├─gvfsd-metadata(3921)─┬─{gvfsd-metadata}(3923)
│                     └─{gvfsd-metadata}(3924)
├─ibus-portal(3836)─┬─{ibus-portal}(3840)
│                  └─{ibus-portal}(3842)
├─obexd(5214)
├─pulseaudio(3566)─┬─{pulseaudio}(3640)
│                  └─{pulseaudio}(3649)
│                  └─{pulseaudio}(5258)
├─tracker-store(4150)─┬─{tracker-store}(4153)
│                    └─{tracker-store}(4154)
│                    └─{tracker-store}(4157)
│                    └─{tracker-store}(4178)
└─xdg-permission-(3847)─┬─{xdg-permission-}(3848)
                       └─{xdg-permission-}(3850)
```

## List just your processes with context

You can see extra context for all of the processes you own with the `-u` option.

```

$ ps -U tux -u
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
tux	3545	0.0	0.0	89656	9708	?	Ss	13:59	0:00	/usr/lib/systemd/systemd
tux	3548	0.0	0.0	171416	5288	?	S	13:59	0:00	(sd-pam)
tux	3566	0.9	0.1	1722212	17352	?	S<sL	13:59	0:29	/usr/bin/pulseaudio [...]
tux	3570	0.0	0.0	664736	8036	?	SLl	13:59	0:00	/usr/bin/gnome-keyring-d
[...]										
tux	5736	0.0	0.0	235628	6036	pts/1	Ss	14:18	0:00	bash
tux	6227	0.0	0.4	2816872	74512	tty2	Sl+	14:30	0:00	/opt/firefox/firefox-bin

## Troubleshoot with PIDs

If you're having trouble with a specific application, or you're just curious about what else on your system an application uses, you can see a memory map of the running process with `pmap`:

```
$ pmap 1776
5736:  bash
000055f9060ec000    1056K r-x--  bash
000055f9063f3000     16K r----  bash
000055f906400000     40K rw---   [ anon ]
00007faf0fa67000   9040K r--s-  passwd
00007faf1033b000     40K r-x--  libnss_sss.so.2
00007faf10345000   2044K -----  libnss_sss.so.2
00007faf10545000      4K rw---  libnss_sss.so.2
00007faf10546000  212692K r----  locale-archive
00007faf1d4fb000   1776K r-x--  libc-2.28.so
00007faf1d6b7000   2044K -----  libc-2.28.so
00007faf1d8ba000      8K rw---  libc-2.28.so
[...]
```

## Process IDs

The **procs-ng** package has all the commands you need to investigate and monitor what your system is using at any moment. Whether you're just curious about how all the disparate parts of a Linux system fit together, or whether you're investigating an error, or you're looking to optimize how your computer is performing, learning these commands gives you a significant advantage for understanding your OS.



[\(/article/18/5/how-kill-process-stop-program-linux\)](https://opensource.com/article/18/5/how-kill-process-stop-program-linux)

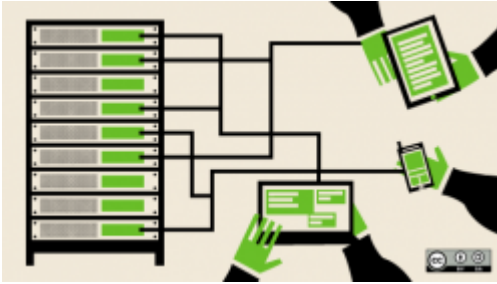
**How to kill a process or stop a program in Linux** [\(/article/18/5/how-](https://opensource.com/article/18/5/how-kill-process-stop-program-linux)

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement. By using this website you agree to our use of cookies.



Here are several options for terminating a program in Linux using the command line or a graphical interface.

[Sachin Patil \(Red Hat\) \(/users/psachin\)](#).



[\(/article/20/11/cockpit-server-management\)](#).

## How I use Cockpit for my home's Linux server management

[\(/article/20/11/cockpit-server-management\)](#).

Anyone—from home users to large-network admins—can access enterprise-grade server management with Cockpit.

[Alan Formy-Duval \(Correspondent\) \(/users/alanfdoss\)](#).



[\(/article/21/7/improve-linux-pc-performance\)](#).

## Open source tools and tips for improving your Linux PC's performance

Make changes to your software (and how you use it) to improve your Linux computer's performance.

[Howard Fosdick \(/users/howtech\)](#).

Topics : [Linux \(/tags/linux\)](#)



### About the author

**Seth Kenlon** - Seth Kenlon is a UNIX geek, free culture advocate, independent multimedia artist, and D&D nerd. He has worked in the [film](#) (<http://www.imdb.com/name/nm1244992>) and [computing](#)

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement. By using this website you agree to our use of cookies.



is one of the maintainers of the Slackware-based multimedia production project [Slackermidia](http://slackermidia.info) (<http://slackermidia.info>).

• [More about me \(/users/seth/\)](/users/seth/)

## Recommended reading



[Write a guessing game in ncurses on Linux \(/article/21/8/guess-number-game-ncurses-linux?utm\\_campaign=intrel\)](/article/21/8/guess-number-game-ncurses-linux?utm_campaign=intrel)



[Linux kernel modules we can't live without \(/article/21/8/linux-kernel-module?utm\\_campaign=intrel\)](/article/21/8/linux-kernel-module?utm_campaign=intrel)



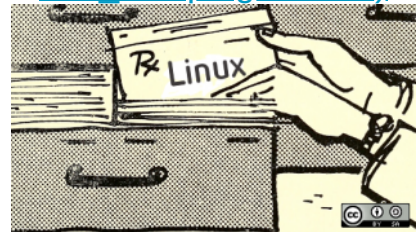
[30 things you didn't know about the Linux kernel \(/article/21/8/linux-kernel?utm\\_campaign=intrel\)](/article/21/8/linux-kernel?utm_campaign=intrel)



[Access your iPhone on Linux with this open source tool \(/article/21/8/libimobiledevice-iphone-linux?utm\\_campaign=intrel\)](/article/21/8/libimobiledevice-iphone-linux?utm_campaign=intrel)



[How to set up your printer on Linux \(/article/21/8/add-printer-linux?utm\\_campaign=intrel\)](/article/21/8/add-printer-linux?utm_campaign=intrel)



[Check file status on Linux with the stat command \(/article/21/8/linux-stat-file-status?utm\\_campaign=intrel\)](/article/21/8/linux-stat-file-status?utm_campaign=intrel)

1 Comment, [Register \(/user/register\)](/user/register) or [Log in \(/user/login?destination=node/67901%3Fsc\\_cid%3D7013a000002pe3mAAA\)](/user/login?destination=node/67901%3Fsc_cid%3D7013a000002pe3mAAA) to post a comment

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement. By using this website you agree to our use of cookies.





[Greg Pittman \(/users/greg-p\)](#) on 11 Aug 2021

Something I have used the PID for is to kill some process that's gone haywire.



[\(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/)

## Subscribe to our weekly newsletter

Subscribe

[Privacy Statement](#)

Get the highlights in your inbox every week.

Find us:

[Privacy Policy](#) | [Terms of Use](#) | [Contact](#) | [Meet the Team](#) | [Visit opensource.org](#)

