

How to perform multitasking on Linux Terminal with Screen

What happens if you need to run some commands at the same time? GNU Screen can help terminal perform multitasking easily. Let's find out specifically through the following article!

If you're a Linux user, you'll have to get used to Linux terminals at some point. Some terminal commands are very common, but some are a bit confusing, but in many cases it is easier to run a command through the terminal window than through the GUI.

But what if you need to run some commands at the same time? GNU Screen can help terminal perform multitasking easily. Let's find out specifically through the following article!

How to run multitasking on Linux terminal

1. What is GNU Screen?
2. How to install Screen
3. Start a Screen session
4. View and disconnect from Screen sessions
5. The terminal commands in another Screen need to be remembered
 1. List of shortcuts in Screen
 2. Create and switch between windows in one session
 3. Create a session with a name that is easier to remember
 4. Share a Screen session
 5. Write Screen output to a file
 6. Lock session in Screen

What is GNU Screen?

GNU Screen is a tool for Linux terminals that acts to divide a terminal into multiple terminals. That means being able to run a command, suppose to download an image file with `wget`, while switching to the second command, such as `systemctl`, to check the SMART status of the drive.

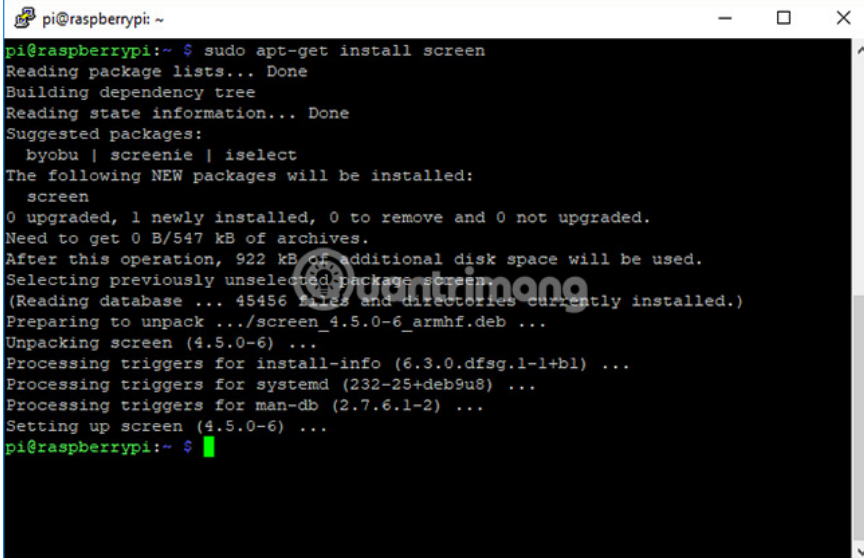
GNU Screen allows to run multiple independent commands in separate sessions that you connect or disconnect at will.

GNU Screen is not only useful for PC. It is very necessary if you are managing a remote server. If you are running commands on the server via SSH, what happens if you disconnect? The command may still be running,

but you cannot easily follow or interact with it.

Screen will solve that problem for you. Users can reconnect to the previous Screen session if they lose the connection or disconnect from the session and let it run in the background until they need to access again.

How to install Screen

A terminal window titled 'pi@raspberrypi: ~' showing the command 'sudo apt-get install screen' and its output. The output includes package lists, dependency tree building, state information, suggested packages (byobu, screenie, iselect, screen), and the installation progress for screen 4.5.0-6. The terminal ends with the prompt 'pi@raspberrypi:~ \$' and a green cursor.

```
pi@raspberrypi:~ $ sudo apt-get install screen
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  byobu | screenie | iselect
The following NEW packages will be installed:
  screen
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/547 kB of archives.
After this operation, 922 kB of additional disk space will be used.
Selecting previously unselected package screen.
(Reading database ... 45456 files and directories currently installed.)
Preparing to unpack .../screen_4.5.0-6_armhf.deb ...
Unpacking screen (4.5.0-6) ...
Processing triggers for install-info (6.3.0.dfsg.1-1+b1) ...
Processing triggers for systemd (232-25+deb9u8) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up screen (4.5.0-6) ...
pi@raspberrypi:~ $
```

Screen does not come with most Linux distributions, but since it is pre-Linux, Screen is very well supported.

If you want to install it on a distribution based on Ubuntu or Debian, run the following command:

```
sudo apt-get update sudo apt-get install screen
```

Similarly, if you're running Arch, open the shell window and run:

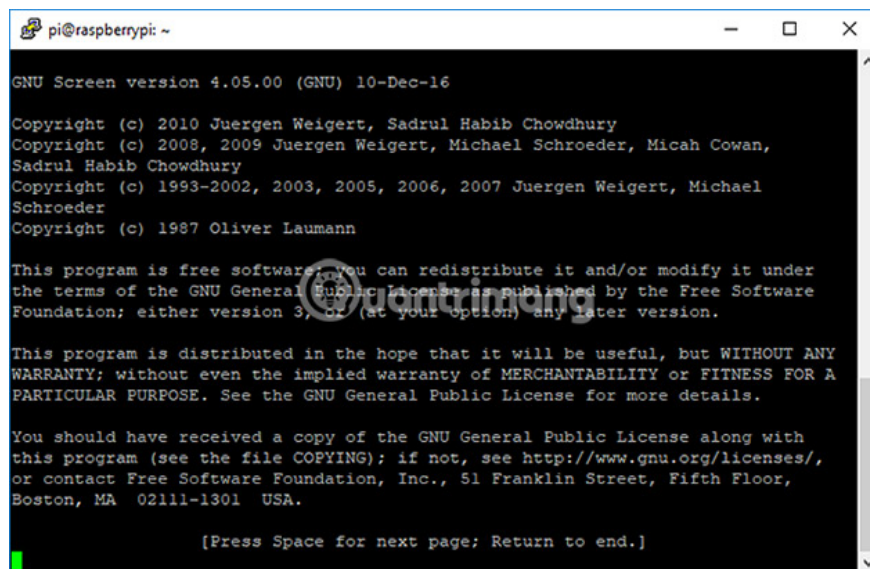
```
sudo pacman -Syu sudo pacman -S screen
```

If you are using Fedora, run the following command instead:

```
sudo yum update sudo yum install screen
```

Start a Screen session

Starting a Screen session is very simple. Open the terminal or set up an SSH connection and enter the **screen**. You will see a window introducing Screen. Press the SPACEBAR or the **Enter** key to close it.

A terminal window titled 'pi@raspberrypi: ~' with standard window controls. The terminal displays the GNU Screen version 4.05.00 (GNU) 10-Dec-16 license text. The text includes copyright information for Juergen Weigert, Michael Schroeder, and Oliver Laumann, and details about the GNU General Public License. At the bottom, it says '[Press Space for next page; Return to end.]'.

```
pi@raspberrypi: ~
GNU Screen version 4.05.00 (GNU) 10-Dec-16

Copyright (c) 2010 Juergen Weigert, Sadrul Habib Chowdhury
Copyright (c) 2008, 2009 Juergen Weigert, Michael Schroeder, Micah Cowan,
Sadrul Habib Chowdhury
Copyright (c) 1993-2002, 2003, 2005, 2006, 2007 Juergen Weigert, Michael
Schroeder
Copyright (c) 1987 Oliver Laumann

This program is free software; you can redistribute it and/or modify it under
the terms of the GNU General Public License as published by the Free Software
Foundation; either version 3, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with
this program (see the file COPYING); if not, see http://www.gnu.org/licenses/,
or contact Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor,
Boston, MA 02111-1301 USA.

[Press Space for next page; Return to end.]
```

After performing the above step, the terminal screen will return to normal. There is no clear indication that you are running a Screen session, but every Linux command that runs from this point will run in a session where you can disconnect and reconnect at will.

View and disconnect from Screen sessions

Readers will want to know how to connect and disconnect from the Screen session if they plan to use this tool again. If already in a Screen session, press **Ctrl + A** followed by the letter **d** (lowercase).

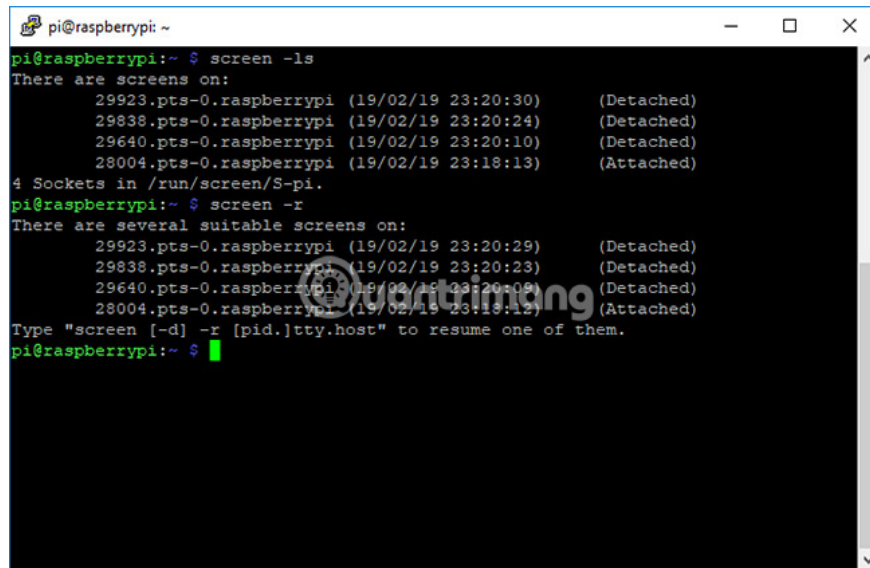
The session and any commands currently running within it will split to run in the background, ready to reconnect later. Assuming you have only one Screen session running, enter:

```
screen -r
```

This command will re-link the session and allow you to continue. If you need to leave a remote session, then reconnect to that session yourself, enter:

```
screen -rd
```

You can run more than one Screen session. If you want to reconnect to a specific session, you need to find the session ID number. Enter **screen -ls** or **screen -r** to list them.

A terminal window on a Raspberry Pi showing the execution of 'screen -ls' and 'screen -r' commands. The first command lists four screens: three detached (IDs 29923, 29838, 29640) and one attached (ID 28004). The second command lists the same four screens and prompts the user to type 'screen [-d] -r [pid.]tty.host' to resume one of them. A watermark 'www.dreamincode.net' is visible in the center of the terminal output.

```
pi@raspberrypi:~  
pi@raspberrypi:~$ screen -ls  
There are screens on:  
29923.pts-0.raspberrypi (19/02/19 23:20:30) (Detached)  
29838.pts-0.raspberrypi (19/02/19 23:20:24) (Detached)  
29640.pts-0.raspberrypi (19/02/19 23:20:10) (Detached)  
28004.pts-0.raspberrypi (19/02/19 23:18:13) (Attached)  
4 Sockets in /run/screen/S-pi.  
pi@raspberrypi:~$ screen -r  
There are several suitable screens on:  
29923.pts-0.raspberrypi (19/02/19 23:20:29) (Detached)  
29838.pts-0.raspberrypi (19/02/19 23:20:23) (Detached)  
29640.pts-0.raspberrypi (19/02/19 23:20:09) (Detached)  
28004.pts-0.raspberrypi (19/02/19 23:18:13) (Attached)  
Type "screen [-d] -r [pid.]tty.host" to resume one of them.  
pi@raspberrypi:~$
```

As the image above shows, enter **screen -r** , followed by the original ID number at the beginning of each session. For example:

```
screen -r 25407
```

If you want to close a session and cancel all the running commands in there, reconnect to it and type **exit**.

The terminal commands in another Screen need to be remembered

Screen has a few tricks to make the most of it. Here are some of the most common terminal commands to remember.

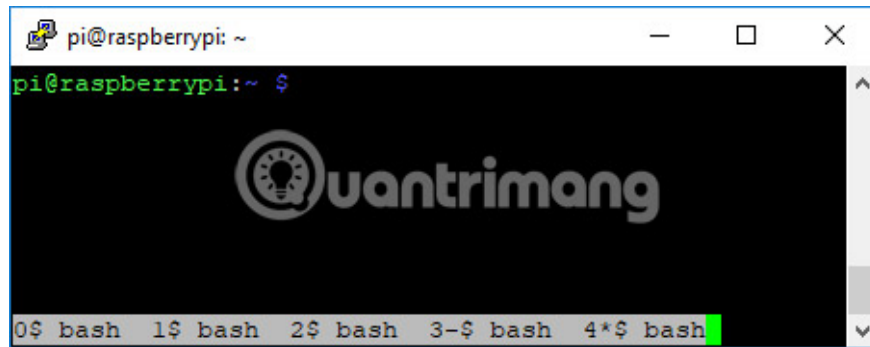
List of shortcuts in Screen

Like all terminal programs, Screen has shortcuts to use. For example, **Ctrl + A** and **d** , to separate existing sessions as mentioned above.

If you want to see other useful shortcuts, just type **Ctrl + A** followed by a **?** to get a list of shortcuts you can use.

Create and switch between windows in one session

There is no need to switch between sessions to run the command. You can switch between windows in one session.



To create a new window in the session, press **Ctrl + A** followed by the letter **c** (lowercase) to create a new window. The first window starts with 0, the next window is 1, etc.

Press **Ctrl + A** and then filter through numbers from **0** to **9** . To list each window one by one, use **Ctrl + A** followed by the letter **w** (lowercase) to see a list of sessions with a single-digit ID.

Create a session with a name that is easier to remember

Randomly created IDs are difficult to remember. Naming a session can make things easier. If you want to start a session with a memorable name, enter, replace `examplename` with the name you want to set:

```
screen -S examplename
```

If you want to reconnect to this session by name, enter:

```
screen -X examplename
```

Share a Screen session

You can share a terminal session with colleagues or friends thanks to Screen. Please enter:

```
screen -rx
```

Instead of disconnecting anyone who is currently connected to this session, you just need to join that session. Other users will see what you type and the commands you run. You can also follow other people if they do so.

Write Screen output to a file

If necessary, write Screen output to a file for maintenance or testing reasons. To do so, enter:

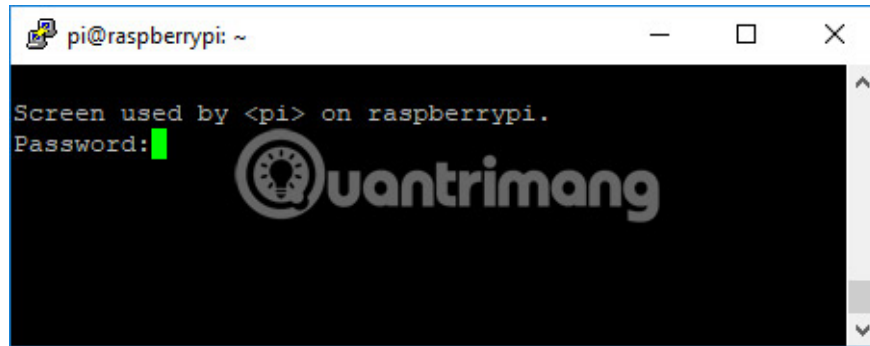
```
screen -L
```

A session will start with the ability to write to a file named **screenlog.x** (where **X** is a number, starting from 0) in the home directory. To start logging a session, enter **Ctrl + A** followed by the letter **H** (**Shift + h**).

Lock session in Screen

If you want to protect a session in Screen, you can lock the session with your existing Linux password.

Type **Ctrl + A** followed by the letter **x** (lowercase) to lock the session while connecting to it.



This command locks it in the current terminal window. Please enter your account password to unlock.

Thanks to GNU Screen, users don't have to worry about waiting for a terminal command to end. This tool is useful if you are looking for ways to control remote servers. It is also a great tool for home PCs if you need to run a few commands from a window. GNU Screen is the perfect tool for system administrators.

Wish you success in installing and using this useful tool!

You finished reading the article "**How to perform multitasking on Linux Terminal with Screen**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.