

# How to manage and restore Tmux sessions in Linux

Tmux is a multiplexer terminal (a tool that allows to use multiple terminals in a window) that integrates many useful features and is supported by a large number of community-created plugins.

**Tmux** is a multiplexer terminal (a tool that allows to use multiple terminals in a window) that integrates many useful features and is supported by a large number of community-created plugins.

Terminal multiplexers like **tmux** and **Screen** give the terminal window super-power, allowing you to open multiple sessions in tabs and split screens.

Combined with numerous terminal tools and session saving features, you can also restore Tmux sessions after a hard reboot.

## How to manage and restore Tmux sessions in Linux?

1. Install tmux
2. Create Windows and Pane
3. Install the plugin
  1. Plugin manager
  2. Resurrect

## Install tmux

Running tmux on a system is relatively simple if you are using a Linux distribution.

For Ubuntu, this means 'summoning' **apt** from the terminal and installing the package from the distribution's official repository. For other distributions, use the bundled package manager to handle installation (tmux is available on most distributions.)

```
sudo apt-get install tmux
```

```
jeff@everliving:~$ sudo apt-get install tmux
[sudo] password for jeff:
Reading package lists... Done
Building dependency tree... 50%
```



## Create Windows and Pane

In tmux, there are two terms used to describe the main types of layout configurations that a user can create. '**Window**' is the term in tmux for tabs. Creating a new window will create a tab that you can switch to with a simple command.

On the other hand, '**Pane**' is the separated part in the current tab or 'Window', which can also be switched using the command.

Before starting to create these things, it's important to note that tmux is a method in the function. This means that the interaction with the terminal session and tmux occurs in separate 'modes'. This is useful, since you can use each individual terminal session normally open, without accidentally triggering the tmux command.

To get access to tmux and start issuing commands, first start a new tmux session and name it:

```
tmux new -s babytmux
```



Feel free to change your name (like '**babytmux**' for example)!

Tmux will open immediately and a new shell session will start. But we want to open more sessions, remember?

To enter tmux's command mode, you need to use the prefix, usually press `Ctrl` and `b` simultaneously. You won't see anything change after pressing the prefix, but you can type the commands by typing `:` right after. To exit command mode, complete the command by pressing `Enter` or pressing the `ESC` key to exit without making changes.

We will use a shortcut to create horizontal division: Press `Ctrl` and `b`, then `"`.

```
jeff@everliving:~$  
  
                                Quantrimang  
  
jeff@everliving:~$  
  
[babytmux]0:bash* "everliving" 16:23 30-Aug-19
```

Now you can navigate between the two sessions with the following keys: `Ctrl` and `b` , then `o` .

```
jeff@everliving:~$ |  
  
                                Quantrimang  
  
jeff@everliving:~$  
  
[babytmux]0:bash* "everliving" 16:25 30-Aug-19
```

Each session is independent of each other, so you can open separate programs within each session. Divide the screen into 4 parts as follows: `Ctrl` and `b` , then press `%` .

Open whatever you want in each pane and it will keep running while you access other sessions.

```

-->EJDic
-->hello

（電話の応答で）『もしもし』；（あいさつ・呼びかけ・驚きの声などに用いて）『こんにちは』，やあ，おい，ちよっと，おや，まあ，こんにちは（やあ，おいなど）という呼びかけ（あいさつ）

jeff@everliving:~$

jeff@everliving:~$

```

The screenshot shows a tmux window with a terminal session. The terminal output includes Japanese text about phone greetings and a file manager interface. The file manager shows a table with columns for Left, File, and Command. The table contains entries for .name, S, Modify, and R, with a date of 2017. The status bar at the bottom of the window shows "[No Name] 0,0-1 ALL".

Left	File	Command	Op
<- ~	.[^]>	<- ~	.[^]>
.name	S Modify	.name	S Modify
/..	R 2017	/..	R 2017

```

UP--DIR
UP--DIR
070M/118G (7%) — 070M/118G (7%) —
Hint: Want your plain shell? Press
jeff@everliving:~$
1Help 2Menu 3View 4Edit
[abytmux]0:bash* "everliving" 16:30 30-Aug-19

```

To create a window, do the following: Press `Ctrl` and `b` , then `c` .

You can switch to the previous window or transition to another window with: `Ctrl` and `b` , then press `p` or `Ctrl` and `b` , then press `n` .

```

jeff@everliving:~$ |

```

The screenshot shows a tmux window with a terminal session. The terminal output is empty, and the Qvantrimang logo is visible in the center. The status bar at the bottom of the window shows "[abytmux]0:bash- 1:bash\*" and "everliving" 16:38 30-Aug-19.

```

[abytmux]0:bash- 1:bash* "everliving" 16:38 30-Aug-19

```

The status bar at the bottom of the screen shows you which process is running in each window and which window you're currently in.

To leave this tmux session, use: `Ctrl` and `b` , then type `:detach-client` .

```
[detached (from session babytmux)]
jeff@everliving:~$

jeff@everliving:~$ |
```



The session will not stop running unless the machine is rebooted or you manually cancel it. To access it again, use the following command:

```
tmux a -t babytmux
```

To make tmux sessions truly 'invincible' that you can come back after rebooting, you need to install one or two plugins.

## Install the plugin

Using the plugin in tmux is relatively simple. However, there is a plugin manager you can use to make installing the next plugin easier.

### Plugin manager

To install Tmux Plugin Manager, we will copy its files from Github with the following code:

```
# List of plugins set -g @plugin 'tmux-plugins/tpm' set -g @plugin 'tmux-plugins'
```

```
# List of plugins
set -g @plugin 'tmux-plugins/tpm'
set -g @plugin 'tmux-plugins/tmux-sensible'

# Other examples:
# set -g @plugin 'github_username/plugin_name'
# set -g @plugin 'git@github.com/user/plugin'
# set -g @plugin 'git@bitbucket.com/user/plugin'

# Initialize TMUX plugin manager (keep this line at the very bottom of tmux.conf)
run -b '~/.tmux/plugins/tpm/tpm'
```



As you can see above, it is necessary to add the github username and plugin name (found in the plugin's github URL) for each plugin that you want to install from now on. The plugin manager will handle the rest. Use this

code to make the plugin manager work:

```
tmux source ~/.tmux.conf
```

Now, for plugins that need to recover a session after rebooting, continue reading the back of the article.

## Resurrect

Tmux Resurrect does exactly what its name implies and returns the saved session using a simple command.

This plugin can be installed by adding the following to the **.tmux.conf** file :

```
set -g @plugin 'tmux-plugins/tmux-resurrect'
```

```
# List of plugins
set -g @plugin 'tmux-plugins/tpm'
set -g @plugin 'tmux-plugins/tmux-sensible'

set -g @plugin 'tmux-plugins/tmux-resurrect'

# Other examples:
# set -g @plugin 'github_username/plugin_name'
# set -g @plugin 'git@github.com/user/plugin'
# set -g @plugin 'git@bitbucket.com/user/plugin'

# Initialize TMUX plugin manager (keep this line at the very bottom of tmux.conf)
run -b '~/.tmux/plugins/tpm/tpm'
```



Now let Tmux Plugin Manager install it using the following command in tmux: Ctrl and b , then press I (This is an uppercase 'i').

A terminal window with a black background. At the top, it says 'TMUX environment reloaded. Done, press ESCAPE to continue.' in white text. In the center, there is a large, semi-transparent watermark of the 'Quantrimang' logo. At the bottom, a green status bar shows '[babytmux]0:bash- 1:[tmux]\*' on the left and '"everliving" 17:10 30-Aug-19' on the right. A small yellow box in the top right corner contains '[0/0]'.

```
TMUX environment reloaded.
Done, press ESCAPE to continue.

Quantrimang

[babytmux]0:bash- 1:[tmux]* "everliving" 17:10 30-Aug-19
```

Once installed, you can start a session and save it with the following commands: Ctrl and b , then press Ctrl + s .

To restore a session, you can use Ctrl and b , then press Ctrl + r .

```
1Help 2Menu 3View 4Edit 5Copy 6Re-ov 7Mkdir 8De-te 9Pu-Dn
16*10000
160000
15*.4
6.0
40*33
1320
1320*4
5280
5280*12
63360
1600*12
19200
jeff@everliving:~$ bc
bc 1.07.1
Copyright 1991-1994, 1997, 1998, 2000, 2004, 2006, 2008, 2012-2017 Fr
ee Software Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
| Restoring...
```

Using the above method, you can maintain an 'immortal' tmux session with all your favorite tools and processes, even after rebooting. Try and explore more features of tmux to make the most of the terminal.

Hope you are successful.

You finished reading the article "**How to manage and restore Tmux sessions in Linux**" 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.