

The official Slack client for Linux is pretty good. It works well, has a clean user interface, and is easy to set up. That said, if you primarily spend time on your Linux PC in the terminal, you may be wishing they had an official command-line app.

Unfortunately, there isn't any talk of an official Slack app for the terminal on Linux. However, there's an excellent, unofficial terminal client for Slack known as Slack-term. Here's how to set it up on your system.

Install Slack-term

Using Slack from the command-line requires the Slack-term application. To install this program, open up a terminal window on your Linux PC by pressing **Ctrl + Alt + T** or **Ctrl + Shift + T**. Then, once the terminal window is ready to use, follow the command-line instructions that correspond with the Linux distribution you use.

Arch Linux AUR

If you use Arch Linux, you'll be happy to know that there's a Slack-term AUR package available. To start the installation process of the Slack-term AUR package, install "Base-devel" and "Git" with the **Pacman** command below.

```
sudo pacman -S base-devel git
```

After installing both "Base-devel" and "Git," it's time to download the latest Slack-term AUR package from the internet using the **git clone** command.

```
git clone https://aur.archlinux.org/slack-term.git
```

When the files are done downloading to your Arch Linux PC, use the **CD** command to move into the "slack-term" directory.

```
cd slack-term
```

Build and install Slack-term on your Arch Linux PC by executing the **makepkg** command. Be aware that this build may fail. If you need help troubleshooting, refer to the comment section of the package.

```
makepkg -sri
```

Snap package

Perhaps the most user-friendly way to install the Slack-term application on Linux is through the Snap package store. Why? It's easy to update, and it only takes one command to install the app!

To install Slack-term from the Snap store, make sure that you have Snap support set up on your Linux PC. After that, run the **snap install** command below to set up Slack-term.

```
sudo snap install slack-term
```

Generic Linux Binary

The developer of Slack-term has a generic Linux binary release of the program that works on nearly every Linux operating system. This version of the app is perfect if you're not using Arch Linux and can't use Snap packages.

To start the installation process for Slack-term, use the **wget** command to download the latest binary release.

```
wget  
https://github.com/erroneousboat/slack-term/releases/download/v0.4.1/slack-term-linux-amd64
```

After the Slack-term file is done downloading to your Linux PC, use the **chmod** command to update its permissions.

```
sudo chmod +x slack-term-linux-amd64
```

With the permissions to Slack-term up to date, follow the steps in the next section to learn how to configure the app.

Configure Slack-term

Slack-term requires some configuration before it will run correctly. The first thing you must do is create a new Slack-term configuration file. Using the **touch** command, create the file “.slack-term”.

```
touch ~/.slack-term
```

After creating the new config file, open it up in the Nano text editor.

```
nano -w ~/.slack-term
```

Paste the following code into the configuration file. You can paste items into Nano by pressing the **Ctrl + Shift + V** on the keyboard.

```
{  
"slack_token": "yourslacktokenhere"  
}
```

Head over to the Slack API website and log into your account. Once logged in, create a new legacy token. Paste the token code into Nano where “yourslacktokenhere” is. Do not remove the “ ” marks!

When the token is in the configuration file in Nano, save the edits by pressing **Ctrl + O** on the

keyboard. Then, exit the Nano text editor by pressing **Ctrl + X**.

Use Slack from the command-line

Using Slack-term from the command-line is surprisingly easy, even though it's done in the Linux terminal. To access your Slack messages, type in the **slack-term** command.

```
slack-term
```

Did you download the generic binary? If so, you must execute Slack-term with the command below instead, as the one above will not work correctly.

```
./slack-term
```

Slack-term, when launched, will bring you to your Slack workspace, in the default channel, complete with message history, etc. To start chatting with people on the default channel, start typing. Send messages by pressing **Enter**.



Channels

- # announcements
- # general
- slackbot
- greetbot
- alan

announcements - Launching a new MUD, client, or website? Hiring? Let us know her...

```

[22:39] <jsweber> @jsweber has joined the channel
[03:04] <michael.josh.weiss> @michael.josh.weiss has joined the channel
[19:49] <ethdem> @ethdem has joined the channel
[15:51] <nocjallen> @nocjallen has joined the channel
[20:25] <kvanderlaag> @kvanderlaag has joined the channel
[11:06] <tom-kouran> @tom-kouran has joined the channel
[22:00] <yekta> @yekta has joined the channel
[02:13] <dalton.nickerson> @dalton.nickerson has joined the channel
[15:55] <swiftausterity> <https://www.mudlet.org/2018/05/mudlet-3-9-0-custom-font
s-in-miniconsoles/>
Mudlet 3.9.0 - custom fonts in miniconsoles | Mudlet
May is here, and a new Mudlet is here! Latency spikes If you Mudlet on Windows wi
th wifi, and you've had occasional spikes of lag - it could have been caused by M
udlet. This release fixes it! If you're still getting lag after upgrading, then i
t's something else :slightly_frowning_face: Thanks to bbailey for finding the iss
ue and figuring it
[15:55] <swiftausterity> if y'all arent going to post it here i will :wink:
[16:27] <keneanung> Heh, I was going to (sooner or later) :stuck_out_tongue:
[18:37] <tristan> @tristan has joined the channel
[03:29] <scott> @scott has joined the channel
[02:22] <seanlyons> @seanlyons has joined the channel
[10:01] <juha.lindfors> @juha.lindfors has joined the channel
[01:22] <oliver.l.carrillo> @oliver.l.carrillo has joined the channel
[02:41] <mattwood11486> @mattwood11486 has joined the channel
[07:07] <gilhulysean> @gilhulysean has joined the channel
[13:41] <swiftausterity>
<!channel> I try to use mass mentions sparingly, I thought it important to let ev
eryone know that I pushed the game jam submission deadline to 11PM MST tonight!

```

NORMAL

Hotkeys



```
branch-master
branch-testing
development
general
random
support
upgrades

james.madison ----- actually, it is a count
james.madison ----- {"type":"pref_change","name":"emoji_use","value":{"\confused\":7,\disappointed\":2,\stuck_out_tongue\":1,\cms\":1,\coffee
hm
james.madison ----- test
james.madison ----- test2
james.madison ----- hey, I have live updates working :simple_smile:
benjamin.franklin ----- Nice!
james.madison ----- I feel like I need to draw out a graph of manager goroutines and who has their channels. It's getting too complex to hold in m
james.madison ----- test send
benjamin.franklin ----- Ack
james.madison ----- Heh. Sending from the client works
james.madison ----- But sent messages are never returned as normal messages on the RTM socket. They're acked with a {reply_to:id, ok:true}
james.madison ----- So...I need to store the id and sent message in a pending queue, and add it to the message map when the reply_to/id is receive
james.madison ----- Ack
benjamin.franklin ----- Yeah.
james.madison ----- It occurs to me I could be lazy and maintain two websockets :stuck_out_tongue:
benjamin.franklin ----- lol
james.madison ----- I really want to colour each user name a different color. But every Go terminal GUI library I've found except ncurses requires
..which means It'd have to have a window for every single message :-/
james.madison ----- Also, apparently I need to strip newlines. heh.
benjamin.franklin ----- Hrm, I see the newlines in the wrong spots. But it's useful to be able to put in actual newlines. I assume those are showing u
james.madison ----- They're showing up because when the user hits 'enter' in the GUI, it inserts a newline at the cursor position before I get the
james.madison ----- It looks like there might not be a way to remove them, with this GUI library :confused:
james.madison ----- The cursor position is reset before I get the text, so I don't know where it was
james.madison ----- I can think of several ways to get it, but they're all ugly, so I'll probably just strip all newlines for now
john.adams ----- <@0CQRDR2CU|richard.castillo> has joined the channel
james.madison ----- test
james.madison ----- hm
james.madison ----- Well, I got replyto working, for messages sent by the client.
james.madison ----- It appears join messages bork it tho :-/
james.madison ----- Ah, the join message didn't break it. Messages with newlines did.
james.madison ----- Well. It still needs a lot of work, but I think it's generally usable at this point.
james.madison ----- <https://github.com/robert-butts/slackterm>
benjamin.franklin ----- Awesome!
james.madison ----- 1077 lines fwIW
```

Slack-term has several different keyboard shortcuts. So many that it might be challenging to figure them out on your own. Here are all of the hotkeys and what they do.

Note: all hotkeys are case-sensitive.

- i** - insert mode.
- /** - search mode.
- k** - move channel cursor up.
- j** - move channel cursor down.
- g** - move channel cursor top.
- G** - move channel cursor bottom.
- pg-up** - scroll chat pane up.
- ctrl-b** - scroll chat pane up.
- ctrl-u** - scroll chat pane up.
- pg-down** - scroll chat pane down.
- ctrl-f** - scroll chat pane down.
- ctrl-d** - scroll chat pane down.
- n** - next search match.

N - previous search match.

q - quit.

f1 - help.

insert left - move input cursor left.

insert right - move input cursor right.

insert enter - send a message.

insert esc - command mode.

search esc - command mode.

search enter - command mode.

SUBSCRIBE TO OUR NEWSLETTER

Source =

["https://www.addictivetips.com/ubuntu-linux-tips/use-slack-from-the-command-line-on-linux/"](https://www.addictivetips.com/ubuntu-linux-tips/use-slack-from-the-command-line-on-linux/)

Loading...

(Visited 5 times, 1 visits today)