Restore tmux session after reboot

175

103

Is there any way to save a tmux session? In other words, if I reboot the computer, will I always lose the sessions?

tmux

share improve this question

asked Jun 21 '12 at 4:17

KendallB

978   3   7   6

migrated from stackoverflow.com Jun 22 '12 at 3:01

This question came from our site for professional and enthusiast programmers.

---

1   You can hibernate to disk as an alternative – Mâtt Frëëman Jun 23 '12 at 5:33

25   Hibernating is an alternative to leaving your computer running constantly, not rebooting. – chepner Jun 23 '12 at 17:31

@chepner: There are those days when `*** System restart required ***` – karatedog Dec 7 '15 at 22:11

@karatedog Yes, so we both agree that hibernation is not a solution to losing your tmux session on reboot. – chepner Dec 7 '15 at 22:21

add a comment

5 Answers

active   oldest   votes

123

Yes, if you reboot you computer you will lose the sessions. Sessions cannot be saved. But, they can be scripted. What most do in fact is to *script* some sessions so that you can re-create them. For instance, here's a trivial shell script to create a session:

```zsh
#!/bin/zsh

SESSIONNAME="script"
tmux has-session -t $SESSIONNAME &> /dev/null

if [ $? != 0 ]
 then
    tmux new-session -s $SESSIONNAME -n script -d
    tmux send-keys -t $SESSIONNAME "~/bin/script" C-m
 fi

tmux attach -t $SESSIONNAME
```

Here's what it does. First, it checks if there's any session already with that name (in this case, the very original name is "script") with **tmux has-session**. It checks the return code. If there's a ongoing session with that name already, it skips the "if" cycle and go straight to the last line, where it attaches to the session. Otherwise, it creates a session and sends some keys to it (just running a random script for now). Then it exits the "if" block and attaches.

This is a **very** trivial sample. You can create multiple windows, panes, and the like before you attach.

This will **not** be the very same thing you asked for, though. If you do any changes to the session, for instance you rename a window and create a new pane in it, if you reboot those changes won't of course be saved.

There are some tools that ease the process of scripting sessions, although I prefer to do things manually (I think it is more versatile). Those tools are Tmuxinator and Teamocil.

My main source of informations was "The Pragmatic Bookshelf" Tmux book.

share improve this answer

edited Apr 11 '13 at 19:54

user72923
469　6　15　28

answered Jun 22 '12 at 8:58

Dakatine
1,515　1　8　11

1　doesn't sound like that will do anything if I want to restore a session with 5 files open. No way to do that? –
　chovy Jan 25 '13 at 1:38

3   Tmux doesn't know anything about the state of processes you may have had running. You could script having the *same* files open by having the 'send-keys' or 'split-window' command be 'vim file1 file2 file3' or look into your editor's session management (vim -S and the like) – bloy Feb 2 '13 at 14:44

     What is the purpose of the `tmux send-keys ...` line? – Dominykas Mostauskis Apr 15 '14 at 13:38

1   @DominykasMostauskis that command *sends* key presses to the specified session. It's like entering the session, and inputing those very keys from the keyboard. In this case, you send "~/bin/script" followed by Enter. – Dakatine Apr 16 '14 at 15:47

     can I have the script do ssh login with username and pass? (I know its not secured just want to know if its possible for systems where i don't care about seurity but still have to have user and pass). – Jas Aug 11 '16 at 5:12

add a comment

84

I wrote a simple bash script that persists open tmux sessions, windows and current working directories in each.

Call it like so manually or periodically from cron (because you might forget):

```
tmux-session save
```

It will write to `~/.tmux-session` . Restore them after reboot like so:

```
tmux-session restore
```

I find this much better than a several hundred line long Perl script.

share improve this answer   edited Jan 28 '14 at 13:41   answered Jul 4 '13 at 22:37

     mislav
     1,240   10   9

add a comment

50

I wrote a tmux plugin that enables you to save complete tmux environment and restore it later. It strives to be really detailed so that you have a feeling you never quit tmux.

https://github.com/tmux-plugins/tmux-resurrect

Update: now there's a tmux-continuum plugin that performs automatic background saves of tmux environment. Optionally it also *automatically* restores tmux env after computer reboot.

share improve this answer

edited Feb 24 '15 at 13:18

answered Aug 30 '14 at 9:16

Bruno Sutic
611 5 5

Just as I posted my "answer" with the same link I looked up - "do'h!" Found this while trying to remember what your plugin was called. – phatskat Sep 8 '14 at 13:20

This plugin is not bad, but it did not restore all my programs. Will read more of your docs and maybe submit an issue on github. – Arne Jul 31 '16 at 12:05

@Arne Depending on the program, this may require program checkpointing. Instead, I would recommend configuring your programs to restore - persistent .vimrc files and cursor positions for vim, etc. - and storing the tmux pane_current_command for programs like man that can be re-opened. Checkpointing is very complicated in my opinion, but worth looking into in any case. – John P Apr 3 at 1:52

add a comment

tmuxinator is a tool written in Ruby, that could be used to create and manage tmux sessions with ease. It could be used to create a project, which could later be instantiated as as tmux session.

8

share improve this answer

answered Jun 26 '14 at 17:40

Praveen Kumar
191 1 3

add a comment

Consider this partial solution found here

4

The author creates a function that saves the history of the tmux session in order to persist the state of the tmux session post a server reboot.

share improve this answer

edited Feb 2 '13 at 23:06

rubo77
1,255 4 20 49

answered Feb 2 '13 at 21:54

greg
41 1

12 Could you please post what they say? Links can rot. – cpast Feb 2 '13 at 22:30

@cpast: This is true, but comments can also rot. Best to give both :) – danielpops Apr 21 '16 at 16:44