

How to run a shell script at startup

[Ask Question](#)

272 On an amazon linux instance, I have two scripts called `start_my_app` and `stop_my_app` which start and stop forever (which in turn run my node.js app). I use these scripts to manually start and stop my node app. So far so good.

110 My problem: I also want to set it up such that `start_my_app` is run whenever the system boots up. I know that I need to add a file inside `init.d` and I know how to symlink it to the proper directory within `rc.d`, but can't figure out what actually needs to go inside the file that I place in `init.d`. I'm thinking it should be just one line, like, `start_my_app`, but that hasn't been working for me.

[linux](#) [node.js](#) [init.d](#) [forever](#)

[share](#) [improve this question](#)

edited Dec 19 '15 at 22:09



[Benjamin](#)

14.7k 26 112 214

asked Oct 19 '12 at 11:56



[meetamit](#)

18.6k 8 44 56

[add a comment](#)

15 Answers

[active](#)

[oldest](#)

[votes](#)

In the file you put in `/etc/init.d/` you have to set it executable with:

227

```
chmod +x /etc/init.d/start_my_app
```

Thanks to [@meetamit](#), if this does not run you have to create a symlink to `/etc/rc.d/`

```
ln -s /etc/init.d/start_my_app /etc/rc.d/
```

Please note that on latest Debian, this will not work as your script have to be LSB compliant (provide, at least, the following actions: start, stop, restart, force-reload, and status):<https://wiki.debian.org/LSBInitScripts>

As a note, you should put the absolute path of your script instead of a relative one, it may solves unexpected issues:

```
/var/myscripts/start_my_app
```

And don't forget to add on top of that file:

```
#!/bin/sh
```

[share](#) [improve this answer](#)

edited Mar 19 '15 at 8:49

answered Oct 19 '12 at 11:59



[Jonathan Muller](#)

4,632 1 15 26

[show 8 more comments](#)

Set a crontab for this

170 `#crontab -e`
`@reboot /home/user/test.sh`

after every startup it will run the test script.

share improve this answer

answered Mar 25 '15 at 5:12



[Hemant kumar](#)

1,701 1 5 2

[add a comment](#)

A simple approach is to add a line in `/etc/rc.local` :

113 `/PATH/TO/MY_APP &`

or if you want to run the command as root :

```
su - USER_FOOBAR -c /PATH/TO/MY_APP &
```

(the trailing ampersand backgrounds the process and allows the rc.local to continue executing)

If you want a full init script, debian distro have a template file, so :

```
cp /etc/init.d/skeleton /etc/init.d/your_app
```

and adapt it a bit.

share improve this answer

edited Aug 30 '17 at 13:57



[peter.bartos](#)

8,409 2 35 59

answered Oct 19 '12 at 11:58



[Gilles Quenot](#)

93.1k 18 148 153

[show 2 more comments](#)

This is the way I do it on red-hat systems

26 Put your script in `/etc/init.d` , owned by root and executable. At the top of the script, you can give a directive for `chkconfig` . Example, the following script is used to start a java application as user oracle.

The name of the script is `/etc/init.d/apex`

```
#!/bin/bash
# chkconfig: 345 99 10
# description: auto start apex listener
#
case "$1" in
'start')
    su - oracle -c "cd /opt/apex ; java -jar apex.war > logs/apex.log 2>logs/apex_error.log &"
```

```
'stop')
  echo "put something to shutdown or kill the process here";;
esac
```

this says that the script must run at levels 3, 4 and 5 and the priority for start/stop is 99 and 10.

then, as user root you can use `chkconfig` to enable or disable the script at startup,

```
chkconfig --list apex
chkconfig --add apex
```

and you can use `service start/stop apex`

share improve this answer

edited Oct 20 '16 at 20:59

[user123456](#)

192 2 13

answered Jun 20 '14 at 7:27



[Saule](#)

389 3 3

[add a comment](#)

Another option is to have an `@reboot` command in your crontab.

10 Not every version of cron supports this, but if your instance is based on the Amazon Linux AMI then it will work.

share improve this answer

answered Nov 10 '13 at 19:07



[chris](#)

20k 41 130 212

[add a comment](#)

Just have a line added to your crontab..

7 Make sure the file is executable:

```
chmod +x /path_to_you_file/your_file
```

To edit crontab file:

```
crontab -e
```

Line you have to add:

```
@reboot /path_to_you_file/your_file
```

That simple!

share improve this answer

edited Nov 10 '16 at 14:07



[Patrick Q](#)

4,672 2 18 31

answered Nov 10 '16 at 13:55



[Luciano Ghilarducci](#)

71 1 2

[add a comment](#)

Enter cron using `sudo` :

5 `sudo crontab -e`

Add a command to run upon start up, in this case a script:

```
@reboot sh /home/user/test.sh
```

Save:

Press ESC then :x to save and exit, or hit ESC then ZZ (that's shift+zz)

Test Test Test:

1. Run your test script without cron to make sure it actually works.
2. Make sure you saved your command in cron, use `sudo crontab -e`
3. Reboot the server to confirm it all works `sudo @reboot`

share improve this answer

edited Mar 6 at 16:49



Omar Ali

5,478 3 26 56

answered Oct 25 '17 at 5:14



user3140639

61 1 1
