

you remember what you're done)

edited Mar 26 '16 at 4:10



waltinator

14.5k 5 27 58

answered Aug 1 '14 at 16:41



Erick

161 1 3

Brilliant. Thank you for this clear and concise guide. Helped me out! – James Lalor Mar 23 at 20:21

Adding another guide to the long list after I realized that the most intrusive steps mentioned elsewhere are not necessary.

My guide needs **no reboot** and I have successfully finished it **over SSH**. It should also work if there are **other eCryptfs mounts** on the machine.

The only non-obvious requirement is to be able to login as another user with `sudo` access.

1. Login as the user with the encrypted home directory (called `user` in this guide).
2. Just to keep the paths in the following steps simple and symmetric:

```
cd /home
```

3. Duplicate the decrypted home folder contents to another directory.

```
sudo cp -rp user user.new
```

4. Logout (`exit` or `logout`). Login as another sudoer.
5. Check that the eCryptfs mount is unmounted. **Do not unmount it manually, it may lead to data loss!** (Experienced myself. I was having another SSH session with `user` logged in.)

```
mount | grep ecryptfs
```

The output must be empty. Or, if there are any other eCryptfs mounts, it must *not* contain `/home/user`.

6. Just like step 2.

```
cd /home
```

7. Mark the duplicate as not being managed with eCryptfs. (There is no need to uninstall eCryptfs packages, especially when there are other eCryptfs mounts.)

```
sudo rm user.new/.ecryptfs user.new/.Private
```

8. Move the not encrypted home directory into place.

```
sudo mv user user.old && sudo mv user.new user
```

9. Login as `user` and check that everything works and eCryptfs does not kick in.

```
mount | grep ecryptfs
```

10. Remove the original contents of `user`'s home folder and the encrypted data.

```
rm -r /home/user.old  
sudo rm -r /home/.ecryptfs/user
```

edited May 25 at 17:28

answered Dec 20 '16 at 9:00



Melebius

2,264 9 30

- if you can backup to an external storage, do it. (Not directly needed, but always good before file manipulation.)
- get familiar with `tty`, at least a little. (Press `Ctrl Alt F7`, `Ctrl Alt F1` etc.) You will need to work with two `tty`-s. One `root` and one of your main user.

- log in as `root` on a separate `tty`, create a new home directory

```
mkdir /home/user.newdir  
chown user:user /home/user.newdir
```

- log out of the graphical session. You can reboot to make sure no programs are working from under your normal user.

- log in as your normal user on a separate `tty`, move all your files to the new destination:

```
mv ~/* /home/user.newdir/
```

Can confirm
these steps
for 16.04 LTS

