Generating SSH Keys

HDC

June 25, 2012

Contents

1   Step1: Check for SSH Keys

Checks to see if there is a directory named ".ssh" in your user directory

```
$ cd ~/.ssh
```

If it says "No such file or directory" skip to **step 3**. Otherwise continue to **step 2**.

2   Step 2: Backup and remove existing SSH Keys

Since there is already an SSH directory you'll want to back the old one up and remove it:

```
$ ls
# Lists all the subdirectories in the current directory
config id_rsa id_rsa.pub known_hosts

$ mkdir key_backup
# Makes a subdirectory called "key_backup" in the current directory

$ cp id_rsa* key_backup
# Copies the id_rsa keypair into key_backup

$ rm id_rsa*
# Deletes the id_rsa keypair
```

3   Step 3: Generate a new SSH key

To generate a new SSH key, enter the code below. We want the default settings so when asked to enter a file in which to save the key, just press enter.

```
$ ssh-keygen # or ssh-keygen -t rsa -C "your_email@youremail.com"
# Creates a new ssh key using the provided email
Generating public/private rsa key pair.
Enter file in which to save the key (/home/you/.ssh/id_rsa):
```

Now you needd to enter a passphrase.

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

4   Step 4: Add your SSH key to GitHub

Run the following code to copy the key to your clipboard.

```
$ sudo apt-get install xclip
# Downloads and installs xclip

$ xclip -sel clip < ~/.ssh/id_rsa.pub
# Copies the contents of the id_rsa.pub file to your clipboard

   ...
```

5 Step 5: put the id_rsa.pub to the correct directory

Check *etc/ssh/sshd_ config, you will find a configuration line like "#Authorizedkeysfile %h*.ssh/authorized_keys", which tells us where the id_rsa.pub puts.

```
$ mv id_rsa.pub authorized_keys
```

Note that, **DO NOT** use cp command instead of mv command. If you do this, it won't work when you