

# Account setup!

## Table of contents

1. Account Setup and Configuration .....	1
1.1. SSH Access .....	1
1.2. Your Email Addresses .....	2
2. Generating a new SSH key and adding it to the ssh-agent .....	2
2.1. <a href="#">Add SSH key</a> .....	3

## 1. Account Setup and Configuration

- The first thing you need to do is set up a free user account. Simply visit [Github](#), choose a user name that isn't already taken, provide an email address and a password, and click the big green “**Sign up for GitHub**” button.
- The next thing you'll see is the pricing page for upgraded plans, but it's safe to ignore this for now.
  - GitHub will send you an email to verify the address you provided.
  - Go ahead and do this; it's pretty important (as we'll see later).

**GitHub provides almost all of its functionality with free accounts**, except some advanced features.



GitHub's **paid plans include advanced tools and features as well as increased limits for free services**.

To get more information about available plans and their comparison, visit [paid plans](#).

You're now ready to use GitHub.

### 1.1. SSH Access

As of right now, you're fully able to connect with Git repositories using the [https://](#) protocol, authenticating with the username and password you just set up. However, to simply clone public projects, you don't even need to sign up - the account we just created comes into play when we fork projects and push to our forks a bit later.

If you'd like to use SSH remotes, you'll need to configure a public key.

Open up your account settings using the link at the top-right of the window

Then select the “SSH keys” section along the left-hand side.

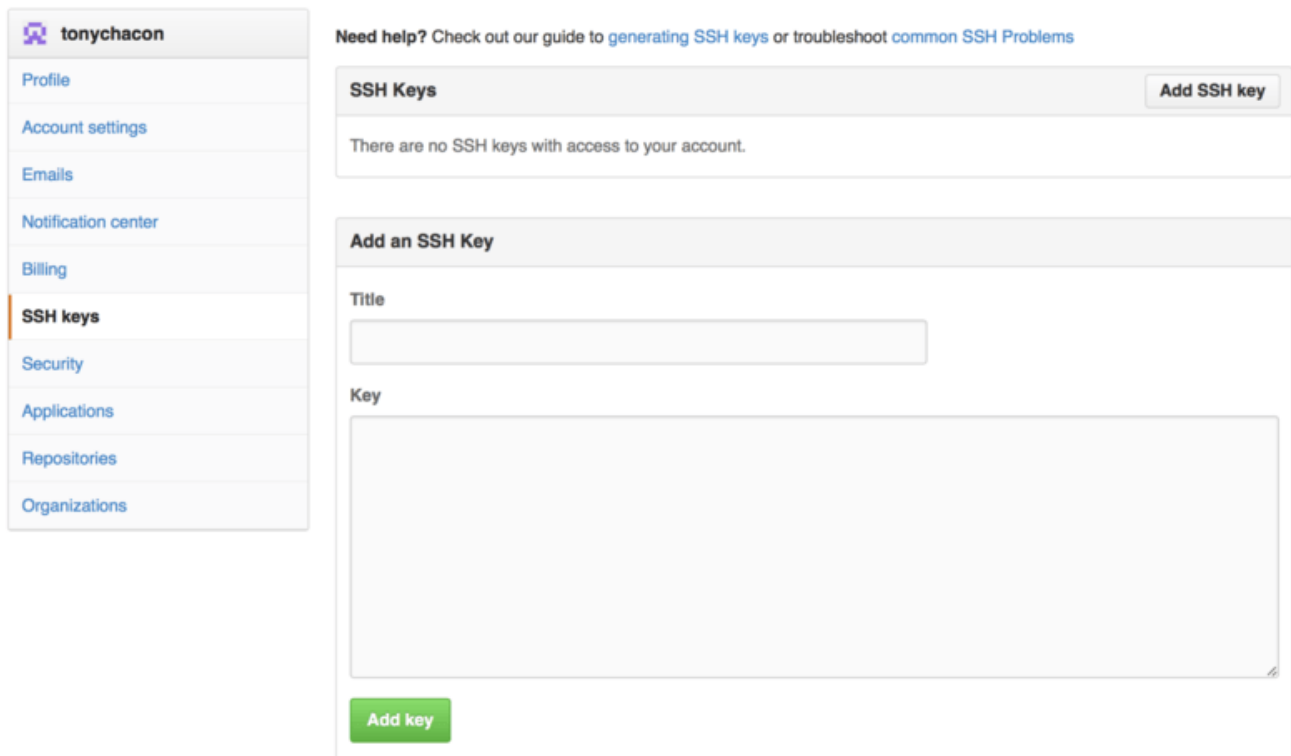


Figure 1. The “SSH keys” link.

From there, click the “Add an SSH key” button, give your key a name, paste the contents of your `~/.ssh/id_rsa.pub` (or whatever you named it) public-key file into the text area, and click “Add key”.



Be sure to name your SSH key something you can remember.

You can name each of your keys (e.g. "My Laptop" or "Work Account") so that if you need to revoke a key later, you can easily tell which one you're looking for.

## 1.2. Your Email Addresses

The way that GitHub maps your Git commits to your user is by email address.

If you use multiple email addresses in your commits and you want GitHub to link them up properly, you need to add all the email addresses you have used to the Emails section of the admin section.

# 2. Generating a new SSH key and adding it to the ssh-agent

## 2.1. Add SSH key

"Pro Git book" by Scott Chacon and Ben Straub , used under CC BY-NC-SA 3.0 /  
Desaturated from original