

6.1 GitHub - Account Setup and Configuration

GitHub is the single largest host for Git repositories, and is the central point of collaboration for millions of developers and projects. A large percentage of all Git repositories are hosted on GitHub, and many open-source projects use it for Git hosting, issue tracking, code review, and other things. So while it's not a direct part of the Git open source project, there's a good chance that you'll want or need to interact with GitHub at some point while using Git professionally.

This chapter is about using GitHub effectively. We'll cover signing up for and managing an account, creating and using Git repositories, common workflows to contribute to projects and to accept contributions to yours, GitHub's programmatic interface and lots of little tips to make your life easier in general.

If you are not interested in using GitHub to host your own projects or to collaborate with other projects that are hosted on GitHub, you can safely skip to [Git Tools](#).

Account Setup and Configuration

The first thing you need to do is set up a free user account. Simply visit <https://github.com>, 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.

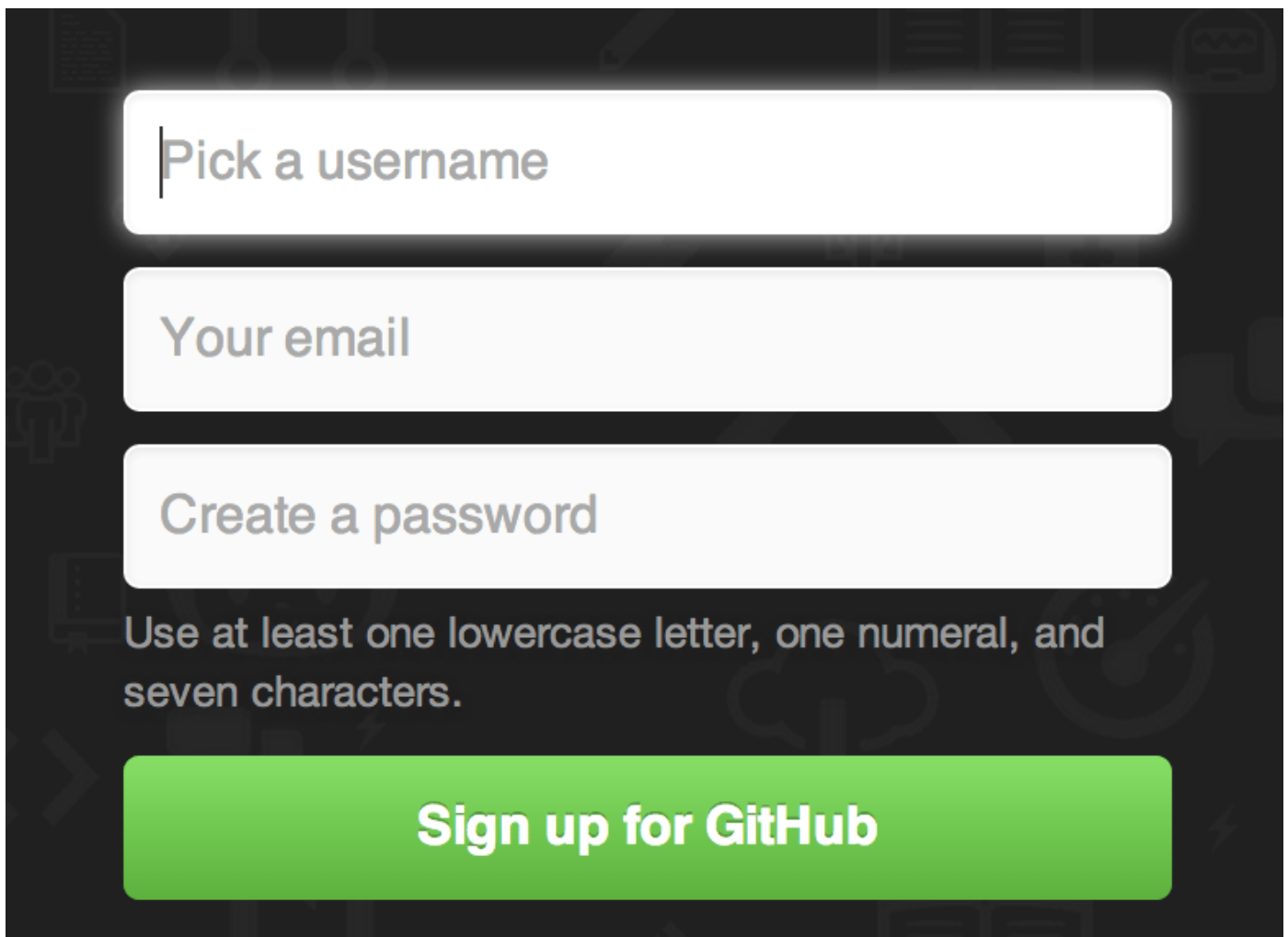
The image shows the GitHub sign-up form on a dark background. It consists of three white input fields stacked vertically. The first field is labeled 'Pick a username' with a vertical cursor line at the start. The second field is labeled 'Your email'. The third field is labeled 'Create a password'. Below the third field, there is a text requirement: 'Use at least one lowercase letter, one numeral, and seven characters.' At the bottom of the form is a large, rounded green button with the text 'Sign up for GitHub' in white.

Figure 81. The GitHub sign-up form

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.

Note GitHub's paid plans include advanced tools and features as well as increased limits for free services, but we won't be covering those in this book. To get more information about available plans and their comparison, visit <https://github.com/pricing>.

Clicking the Octocat logo at the top-left of the screen will take you to your dashboard page. You're now ready to use GitHub.

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. If you don't already have one, see [Generating Your SSH Public Key](#). Open up your account settings using the link at the top-right of the window:

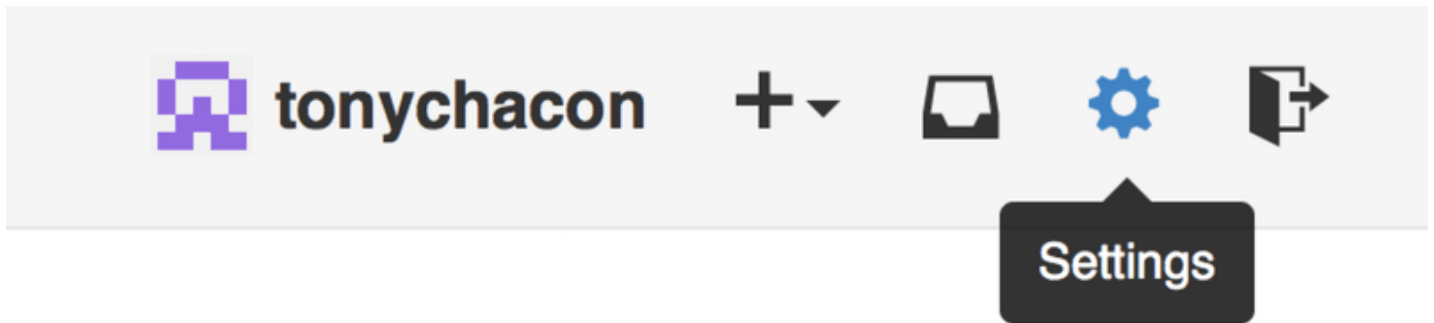


Figure 82. The “Account settings” link

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

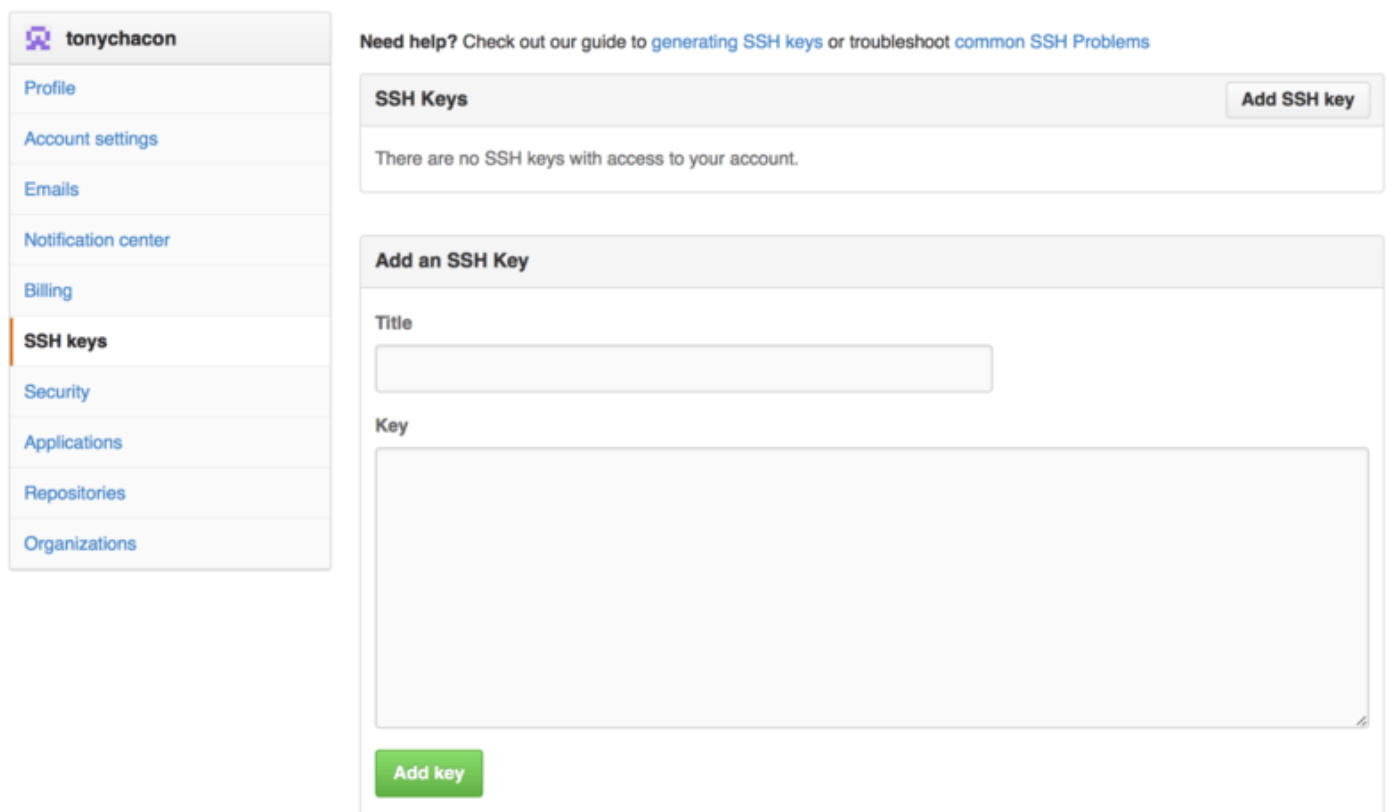


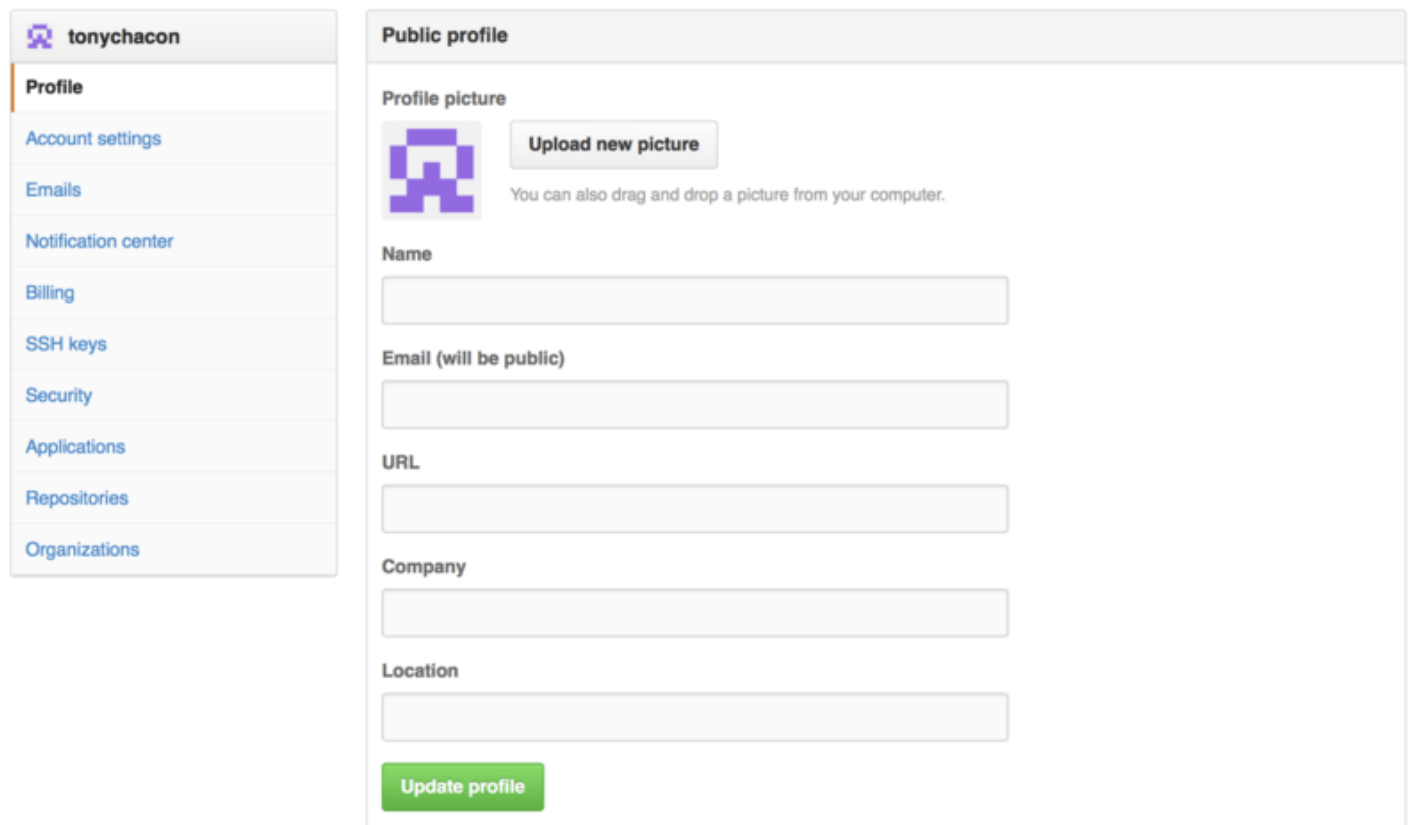
Figure 83. The “SSH keys” link.

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”.

Note 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.

Your Avatar

Next, if you wish, you can replace the avatar that is generated for you with an image of your choosing. First go to the “Profile” tab (above the SSH Keys tab) and click “Upload new picture”.



The screenshot shows the GitHub profile settings page for user 'tonychacon'. On the left is a sidebar with navigation links: Profile, Account settings, Emails, Notification center, Billing, SSH keys, Security, Applications, Repositories, and Organizations. The 'Profile' link is highlighted. The main content area is titled 'Public profile' and contains the following sections:

- Profile picture:** Displays a default purple pixelated avatar. To its right is a button labeled 'Upload new picture' and a note: 'You can also drag and drop a picture from your computer.'
- Name:** An empty text input field.
- Email (will be public):** An empty text input field.
- URL:** An empty text input field.
- Company:** An empty text input field.
- Location:** An empty text input field.
- Update profile:** A green button at the bottom of the form.

Figure 84. The “Profile” link

We’ll choose a copy of the Git logo that is on our hard drive and then we get a chance to crop it.

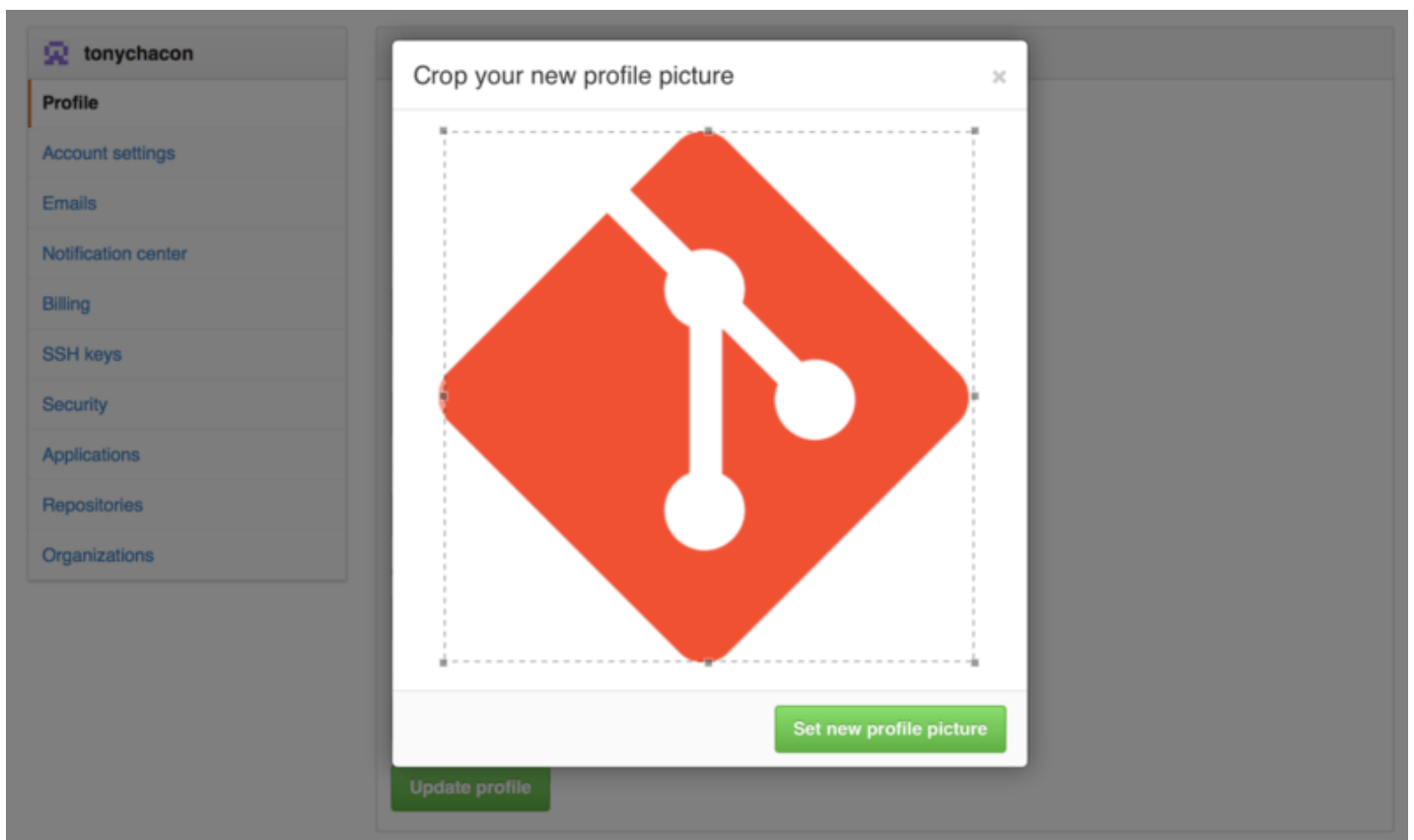


Figure 85. Crop your avatar

Now anywhere you interact on the site, people will see your avatar next to your username.

If you happen to have uploaded an avatar to the popular Gravatar service (often used for Wordpress accounts), that avatar will be used by default and you don't need to do this step.

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.

The screenshot shows the GitHub 'Email' settings page for a user named 'tonychacon'. On the left is a sidebar with navigation links: Profile, Account settings, Emails (highlighted), Notification center, Billing, SSH keys, Security, Applications, Repositories, and Organizations. The main content area is titled 'Email' and contains the following information:

- A message: "Your **primary GitHub email address** will be used for account-related notifications (e.g. account changes and billing receipts) as well as any web-based GitHub operations (e.g. edits and merges)."
- A list of email addresses:
 - `tonychacon@example.com` with a green 'Primary' badge and a 'Public' badge. A trash icon is on the right.
 - `tchacon@example.com` with a 'Set as primary' button and a trash icon.
 - `tony.chacon@example.com` with an 'Unverified' badge and a 'Send verification email' button. A trash icon is on the right.
- An 'Add email address' section with a text input field and an 'Add' button.
- A checkbox labeled 'Keep my email address private'. Below it, a note states: "We will use `tonychacon@users.noreply.github.com` when performing Git operations and sending email on your behalf."

Figure 86. Add email addresses

In [Add email addresses](#) we can see some of the different states that are possible. The top address is verified and set as the primary address, meaning that is where you'll get any notifications and receipts. The second address is verified and so can be set as the primary if you wish to switch them. The final address is unverified, meaning that you can't make it your primary address. If GitHub sees any of these in commit messages in any repository on the site, it will be linked to your user now.

Two Factor Authentication

Finally, for extra security, you should definitely set up Two-factor Authentication or "2FA". Two-factor Authentication is an authentication mechanism that is becoming more and more popular recently to mitigate the risk of your account being compromised if your password is stolen somehow. Turning it on will make GitHub ask you for two different methods of authentication, so that if one of them is compromised, an attacker will not be able to access your account.

You can find the Two-factor Authentication setup under the Security tab of your Account settings.

tonychacon

Profile

Account settings

Emails

Notification center

Billing

SSH keys

Security

Applications

Repositories

Organizations

Two-factor authentication

Status: Off

Set up two-factor authentication

Two-factor authentication provides another layer of security to your account. [Learn more about two-factor auth at GitHub Help.](#)

Sessions

This is a list of devices that have logged into your account. Revoke any sessions that you do not recognize.

Paris 85.168.227.34

Your current session

Safari on OS X 10.9.4

Location:
Paris, Ile-de-France, France

Signed in:
September 30, 2014

Figure 87. 2FA in the Security Tab

If you click on the “Set up two-factor authentication” button, it will take you to a configuration page where you can choose to use a phone app to generate your secondary code (a “time based one-time password”), or you can have GitHub send you a code via SMS each time you need to log in.

After you choose which method you prefer and follow the instructions for setting up 2FA, your account will then be a little more secure and you will have to provide a code in addition to your password whenever you log into GitHub.

[prev](#) | [next](#)