

GIT


# Use GitHub with SSH - Complete guide including VSCode setup



|            |       |            |            |  |
|------------|-------|------------|------------|--|
| PRICE DROP |       | PRICE DROP | PRICE DROP |  |
|            |       |            |            |  |
| \$285      | \$880 | \$285      | \$220      |  |

Game-Changing PXG® Clubs  
PXG

If you want to use Git without using password, then SSH key is the solution. Using SSH key is highly recommended and followed by professional developers often. In this post I will walk you through the setup of SSH keys for your GitHub account. And the process will be similar for any other services like Gitlab/Bitbucket also.

 **Note:** All of the following commands are to be run in Git Bash. And now that you started using SSH keys, **never** use your password unless you login in browser. I recommend to delete all your personal access tokens from your account, visit [Tokens](#) and delete all.

## Configure basic user info

First you need to tell git your user name and email address. This can be user name and email address with which you want to make commits.

```
BASH
git config --global user.name "YOUR_USERNAME"
git config --global user.email "YOUR_EMAIL_ADDRESS"
```

## Testing the config

You can verify the configured username and email using the following commands

```
BASH
git config --global user.name
```

```
git config --global user.email
```

✔ I recommended to config the line endings and case sensitivity for file names using `git config --global core.autocrlf true` and `git config --global core.ignorecase true`.

## Setting up SSH keys

### First check if there are any keys

To check if there are any existing SSH keys on your PC:

```
BASH
ls -al ~/.ssh
```

If you don't have an existing public and private key pair, or don't wish to use any that are already available to connect to GitHub, then see the below steps to generate a new SSH key.

💧 If you receive an error that `~/.ssh` doesn't exist, don't worry! We'll create that folder by generating a new SSH key

## Generate a new SSH key

To create a new SSH key, run the following command substituting in your GitHub email address.

```
BASH
ssh-keygen -t rsa -b 4096 -C "YOUR_GITHUB_EMAIL_ADDRESS"
```

💧 You can use a passphrase, for the SSH key being generated. Just enter the passphrase when the CLI prompts. This is **NOT** same as the password that you use to login GitHub.

## Connecting the generated SSH key to GitHub

To connect the generated SSH key to your GitHub profile, you have to add it in the account settings of your account. To do so:

- ✔ Navigate to **SSH and GPG keys** section.
- ✔ Click on New SSH key
- ✔ Give some title, (recommended: that you can identify your PC with).
- ✔ Paste the generated SSH key
- ✔ Click on Add SSH key



To copy the generated SSH key to clipboard, use the **clip** command

```
BASH
clip < ~/.ssh/id_rsa.pub
```

## Testing the whole setup

Run the following command, to test your entire SSH key setup.

```
ssh -T git@github.com
```

❌ **Note:** To use SSH keys effectively, you have use the SSH protocol while cloning a repo, or adding a remote to existing repo

```
BASH
1 # Avoid this, or it will ask for your password
2 git clone https://github.com/awsm-page/sample.git
3
4 # Do this
5 git clone git@github.com:awsm-page/sample.git
```

Similarly, you have to consider existing repos which are previously authenticated with *personal access tokens* also.

Check the remotes of any repository you work with, whether they are using HTTP or SSH.

```
BASH
1 # List all remotes
2 git remote -v
3
4 # Change to SSH remote, if needed
5 git remote set-url origin git@github.com:USERNAME/REPOSITORY.git
```

Complete guide for switching URLs can be found at [GitHub docs](#)

## Adding your SSH key to the ssh-agent(optional)

You can add your SSH key to the ssh-agent, if you don't want reenter your passphrase every time you use your SSH key. To do so:

Start the ssh-agent in the background, using the command



```
BASH
eval "$(ssh-agent -s)"
```

Delete everything that exists

```
BASH
ssh-add -D
```

Then add the generated key to the agent

```
BASH
ssh-add ~/.ssh/id_rsa
```

To list the added keys:

```
BASH
ssh-add -l
```

[more info](#)

## Setup VSCode

VS Code works most easily with SSH keys without a passphrase. To use SSH Git authentication with VS Code, you have launch VS Code from a Git Bash prompt to inherit its SSH environment.

```
BASH
code .
# or
code path/to/specific/folder
```

[more info](#)

*Published on Fri Jan 17 2020. Updated on Sun Jan 26 2020*

## Comments

2 Comments - powered by [utteranc.es](#)

synco1111 commented on Dec 27, 2021


Awesome Guide..!  
Tiny small typo:

```
eval \$(ssh-agent -s)
```

Should be (without the back slash



```
eval $(ssh-agent -s)
```

 1

Impulsleistung commented on Feb 6, 2022

Unable to connect to VS Code server: Error in request.  
Error: connect ENOENT /run/user/1000/vscode-ipc-50021f70-e7d5-4e63-aeaa-70a16c07ac54.sock  
at PipeConnectWrap.afterConnect [as oncomplete] (net.js:1146:16) {  
 errno: -2,  
 code: 'ENOENT',  
 syscall: 'connect',  
 address: '/run/user/1000/vscode-ipc-50021f70-e7d5-4e63-aeaa-70a16c07ac54.sock'  
}

WritePreview

Sign in to comment



Helping designers and developers through providing handpicked and high quality resources.



#### WEBSITE

All Posts

Topics

Search

#### TOPICS

JavaScript

Git

Electron

#### LEGAL

About

Terms & conditions

Privacy

#### TOOLS

Password Generator

Solid Wallpaper

ISO Date

Box Shadows

