

Setting up SSH authentication is the **best and most secure way** to manage your Git operations without needing to use your username or a Personal Access Token (PAT) every time.

Here are the steps to set up an SSH key and use it with GitHub:

---

## Step 1: Check for Existing SSH Keys

First, check if you already have an SSH key pair on your machine.

Open your Terminal (you're already using it) and type:

Bash

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

- 1.
2. Look for files named `id_rsa.pub`, `id_ecdsa.pub`, or `id_ed25519.pub`.
  - **If you see one:** Skip to **Step 3** (Copy the Public Key).
  - **If you don't see one:** Proceed to **Step 2** (Generate a New Key).

---

## Step 2: Generate a New SSH Key Pair

If you don't have a key, create a new one. The `ed25519` algorithm is the modern and recommended choice.

Run the following command, replacing your GitHub email:

Bash

```
ssh-keygen -t ed25519 -C "your_email@example.com"
```

- 1.
2. When prompted for "Enter a file in which to save the key," just press **Enter** to accept the default location (`~/.ssh/id_ed25519`).
3. When prompted for a "passphrase," **it's highly recommended to set one**. This passphrase adds an extra layer of security, protecting your key even if someone gains access to your computer.

---

## Step 3: Copy the Public Key

You need the contents of the public key file (`.pub` file) to upload to GitHub.

Use the `cat` command to display the key's contents:

Bash

```
cat ~/.ssh/id_ed25519.pub
```

- 1.
2. **Copy the entire output**, starting with `ssh-ed25519` and ending with your email address.

---

## Step 4: Add the Key to Your GitHub Account

This links your computer's identity to your GitHub account.

1. Go to **GitHub**  $\rightarrow$  **Settings** (Click your profile picture)  $\rightarrow$  **SSH and GPG keys**.
2. Click the **New SSH key** button.
3. **Title:** Give it a descriptive name (e.g., "MacBook Pro").
4. **Key type:** Select **Authentication Key**.
5. **Key:** **Paste the entire text** you copied in Step 3 into the key field.

6. Click **Add SSH key**.



## Step 5: Change the Remote URL and Push

Finally, update your local repository to use the SSH URL instead of the HTTPS URL.

### Remove the old remote:

Bash

```
git remote remove origin
```

1.

### Add the new SSH remote: (Use the `git@github.com:...` format)

Bash

```
git remote add origin git@github.com:maverick447/Fabio.git
```

2.

### Test the connection: You may be asked to confirm the host.

Bash

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

3. (You should see a message like: "Hi maverick447! You've successfully authenticated...")

### Push your code (this time it will use SSH):

Bash

```
git push -u origin main
```

4.

You will now only be prompted for the passphrase you set in Step 2 (if you set one), and you will not have to enter your username or PAT again for future operations on this machine.

