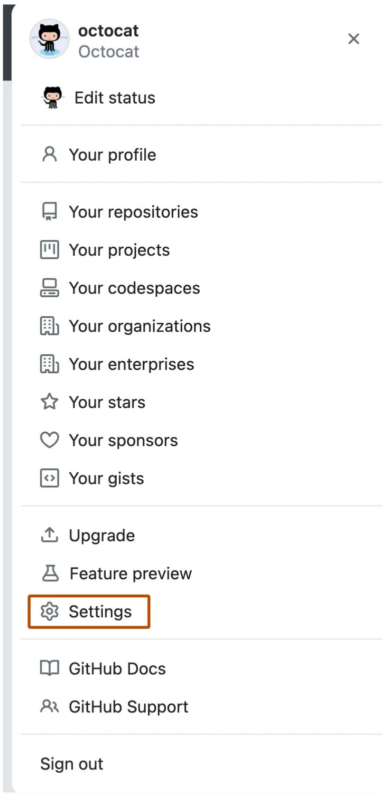
Reviewing your SSH keys

To keep your credentials secure, you should regularly audit your SSH keys, deploy keys, and review authorized applications that access your account on GitHub.com.

Mac Windows Linux

You can delete unauthorized (or possibly compromised) SSH keys to ensure that an attacker no longer has access to your repositories. You can also approve existing SSH keys that are valid.

1. In the upper-right corner of any page, click your profile photo, then click **Settings**.



2. In the "Access" section of the sidebar, click **SSH and GPG keys**.
3. Under "SSH keys", take note of the SSH keys associated with your account. For those that you don't recognize, or that are out-of-date, click **Delete**. If there are valid SSH keys you'd like to keep, click **Approve**.

   > **Note:** If you're auditing your SSH keys due to an unsuccessful Git operation, the unverified key that caused the SSH key audit error will be highlighted in the list of SSH keys.

4. Open Terminal.
5. Start the ssh-agent in the background.

```
$ eval "$(ssh-agent -s)"
> Agent pid 59566
```

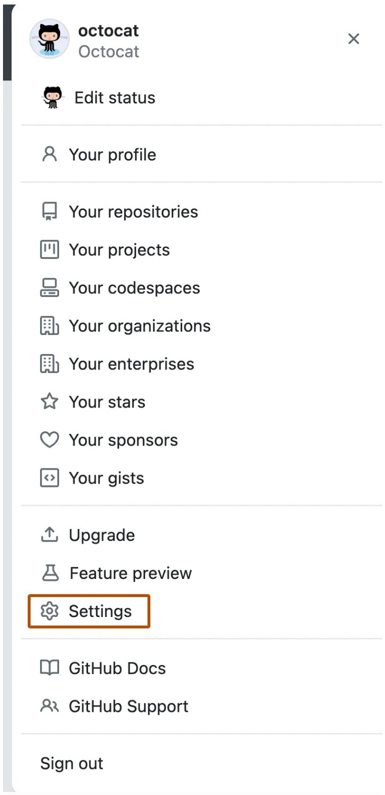
Depending on your environment, you may need to use a different command. For example, you may need to use root access by running `sudo -s -H` before starting the ssh-agent, or you may need to use `exec ssh-agent bash` or `exec ssh-agent zsh` to run the ssh-agent.

6. Find and take a note of your public key fingerprint.

```
$ ssh-add -l -E sha256
> 2048 SHA256:274ffWxgaxq/tSINAykStUL7XWyRNcRTlcST1Ei7gBQ
/Users/USERNAME/.ssh/id_rsa (RSA)
```

7. The SSH keys on GitHub Enterprise Cloud *should* match the same keys on your computer.

1. In the upper-right corner of any page, click your profile photo, then click **Settings**.



2. In the "Access" section of the sidebar, click **SSH and GPG keys**.

3. Under "SSH keys", take note of the SSH keys associated with your account. For those that you don't recognize, or that are out-of-date, click **Delete**. If there are valid SSH keys you'd like to keep, click **Approve**.

> **Note:** If you're auditing your SSH keys due to an unsuccessful Git operation, the unverified key that caused the SSH key audit error will be highlighted in the list of SSH keys.

4. Open Git Bash.

5. **If you are using Git Bash**, turn on ssh-agent:

```
# start the ssh-agent in the background
```

```
$ eval "$(ssh-agent -s)"
> Agent pid 59566
```

If you are using another terminal prompt, such as [Git for Windows](), turn on ssh-agent:

~~~
```shell
# start the ssh-agent in the background
$ eval $(ssh-agent -s)
> Agent pid 59566
```
~~~

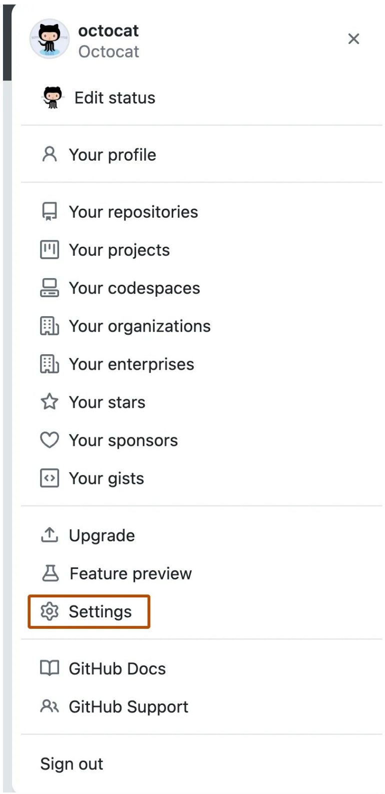
> **Note:** The eval commands above start ssh-agent manually in your environment. These commands may fail if ssh-agent already runs as a background system service. If that happens, we recommend you check the relevant documentation for your environment.

6. Find and take a note of your public key fingerprint.

```
$ ssh-add -l -E sha256
> 2048 SHA256:274ffWxgaxq/tSINAykStUL7XWyRNcRTlcST1Ei7gBQ
/Users/USERNAME/.ssh/id_rsa (RSA)
```

7. The SSH keys on GitHub Enterprise Cloud *should* match the same keys on your computer.

1. In the upper-right corner of any page, click your profile photo, then click **Settings**.



2. In the "Access" section of the sidebar, click **SSH and GPG keys**.

3. Under "SSH keys", take note of the SSH keys associated with your account. For those that you don't recognize, or that are out-of-date, click **Delete**. If there are valid SSH keys you'd like to keep, click **Approve**.

> **Note:** If you're auditing your SSH keys due to an unsuccessful Git operation, the unverified key that caused the SSH key audit error will be highlighted in the list of SSH keys.

4. Open Terminal.
5. Start the ssh-agent in the background.

   ```
   $ eval "$(ssh-agent -s)"
   > Agent pid 59566
   ```

   Depending on your environment, you may need to use a different command. For example, you may need to use root access by running `sudo -s -H` before starting the ssh-agent, or you may need to use `exec ssh-agent bash` or `exec ssh-agent zsh` to run the ssh-agent.
6. Find and take a note of your public key fingerprint.

   ```
   $ ssh-add -l -E sha256
   > 2048 SHA256:274ffWxgaxq/tSINAykStUL7XWyRNcRTlcST1Ei7gBQ
   /Users/USERNAME/.ssh/id_rsa (RSA)
   ```

7. The SSH keys on GitHub Enterprise Cloud *should* match the same keys on your computer.

> **Warning**: If you see an SSH key you're not familiar with on GitHub Enterprise Cloud, delete it immediately and contact us through the GitHub Support portal for further help. An unidentified public key may indicate a possible security concern.