Checking for existing SSH keys

MAC | WINDOWS | LINUX

Before you generate an SSH key, you can check to see if you have any existing SSH keys.

Note: DSA keys were deprecated in OpenSSH 7.0. If your operating system uses OpenSSH, you'll need to use an alternate type of key when setting up SSH, such as an RSA key. For instance, if your operating system is MacOS Sierra, you can set up SSH using an RSA key.

1. Open Git Bash.
2. Enter `ls -al ~/.ssh` to see if existing SSH keys are present:

```
$ ls -al ~/.ssh
# Lists the files in your .ssh directory, if they exist
```

3. Check the directory listing to see if you already have a public SSH key.

By default, the filenames of the public keys are one of the following:

id_dsa.pub
id_ecdsa.pub
id_ed25519.pub

id_rsa.pub

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

If you see an existing public and private key pair listed (for example *id_rsa.pub* and *id_rsa*) that you would like to use to connect to GitHub, you can add your SSH key to the ssh-agent.

Tip: If you receive an error that *~/.ssh* doesn't exist, don't worry! We'll create it when we generate a new SSH key.

Contact a human

Article versions
GitHub.com
GitHub Enterprise 2.10
GitHub Enterprise 2.9
GitHub Enterprise 2.8