Connecting to GitHub with SSH

You can connect to GitHub using SSH.

#### **[About SSH→](https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/about-ssh)**

Using the SSH protocol, you can connect and authenticate to remote servers and services. With SSH keys, you can connect to GitHub without supplying your username and personal access token at each visit.

#### **[Checking for existing SSH keys](https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/checking-for-existing-ssh-keys)**

**Note:** DSA keys (SSH-DSS) are no longer supported. Existing keys will continue to function, but you cannot add new DSA keys to your GitHub account.

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

1. 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\_rsa.pub
* Id\_ecdsa.pub
* Id\_ed25519.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](https://docs.github.com/en/free-pro-team@latest/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent).

### [**Generating a new SSH key**](https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#generating-a-new-ssh-key)

1. Open Git Bash.
2. Paste the text below, substituting in your GitHub email address.

$ ssh-keygen -t ed25519 -C "*your\_email@example.com*"

**Note:** If you are using a legacy system that doesn't support the Ed25519 algorithm, use:

$ ssh-keygen -t rsa -b 4096 -C [your\_email@example.com](mailto:your_email@example.com)

This creates a new ssh key, using the provided email as a label.

Generating public/private ed25519 key pair.

1. When you're prompted to "Enter a file in which to save the key," press Enter. This accepts the default file location.

Enter a file in which to save the key (/c/Users/*you*/.ssh/id\_ed25519):

*[Press enter]*

1. At the prompt, type a secure passphrase. For more information, see ["Working with SSH key passphrases"](https://docs.github.com/en/free-pro-team@latest/articles/working-with-ssh-key-passphrases).

Enter passphrase (empty for no passphrase): [Type a passphrase]

Enter same passphrase again: [Type passphrase again]

### [**Adding your SSH key to the ssh-agent**](https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#adding-your-ssh-key-to-the-ssh-agent)

Before adding a new SSH key to the ssh-agent to manage your keys, you should have [checked for existing SSH keys](https://docs.github.com/en/free-pro-team@latest/articles/checking-for-existing-ssh-keys) and [generated a new SSH key](https://docs.github.com/en/free-pro-team@latest/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent#generating-a-new-ssh-key).

If you have [GitHub Desktop](https://desktop.github.com/) installed, you can use it to clone repositories and not deal with SSH keys.

1. Ensure the ssh-agent is running. You can use the "Auto-launching the ssh-agent" instructions in "[Working with SSH key passphrases](https://docs.github.com/en/free-pro-team@latest/articles/working-with-ssh-key-passphrases)", or start it manually:

# start the ssh-agent in the background

$ eval `ssh-agent -s`

> Agent pid 59566

1. Add your SSH private key to the ssh-agent. If you created your key with a different name, or if you are adding an existing key that has a different name, replace id\_ed25519 in the command with the name of your private key file.

$ ssh-add ~/.ssh/id\_ed25519

#### **[Testing your SSH connection](https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/testing-your-ssh-connection)**

When you test your connection, you'll need to authenticate this action using your password, which is the SSH key passphrase you created earlier. For more information on working with SSH key passphrases, see ["Working with SSH key passphrases"](https://docs.github.com/en/free-pro-team@latest/articles/working-with-ssh-key-passphrases).

1. Open Git Bash.
2. Enter the following:

$ ssh -T git@github.com

# Attempts to ssh to GitHub

You may see a warning like this:

> The authenticity of host 'github.com (IP ADDRESS)' can't be established.

> RSA key fingerprint is SHA256:nThbg6kXUpJWGl7E1IGOCspRomTxdCARLviKw6E5SY8.

> Are you sure you want to continue connecting (yes/no)?

1. Verify that the fingerprint in the message you see matches one of the messages in step 2, then type yes:

> Hi username! You've successfully authenticated, but GitHub does not

> provide shell access.

1. Verify that the resulting message contains your username. If you receive a "permission denied" message, see ["Error: Permission denied (publickey)"](https://docs.github.com/en/free-pro-team@latest/articles/error-permission-denied-publickey).