Eclipse + GitHub + HTTPS

The following is an easy way to connect Eclipse to GitHub using SSH with encryption and signing. These instructions assume you have attempted using HTTPS as outlined in "Eclipse + GitHub + HTTPS."

Important: Remember to commit and push your projects. Your assignment will be graded based on what was committed and pushed to the repository. I will not track down your projects or changes after the assignment is due.

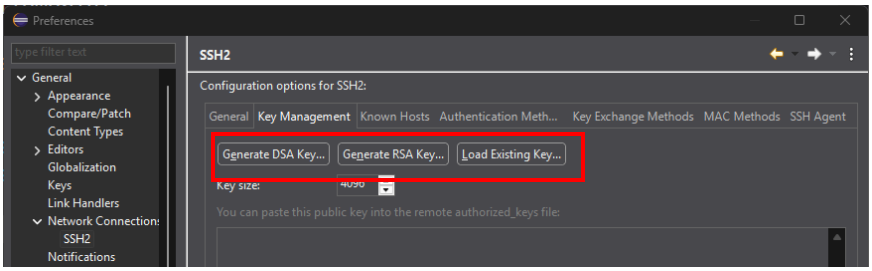
1. Generate or Load an SSH Key in Eclipse

To add an SSH key to GitHub from Eclipse, first generate an SSH key in Eclipse (or use an existing one). Then add the public key to your GitHub account under "SSH and GPG keys" settings. This allows Eclipse to authenticate with GitHub using an SSH key during operations such as cloning or pushing.

Menu -> Window -> Preferences -> General -> Network Connections -> SSH2 -> Key Management.

To generate a new key, click on Generate RSA Key or Generate DSA Key, then click Apply to create your key. Click Save Private Key to create the .ssh folder (if it does not already exist) and save the key. I typically recommend RSA.

To load an Existing Key: Click on Load Existing Key and browse to your private key file (e.g., ~/.ssh/id_rsa or ~/.ssh/id_dsa).

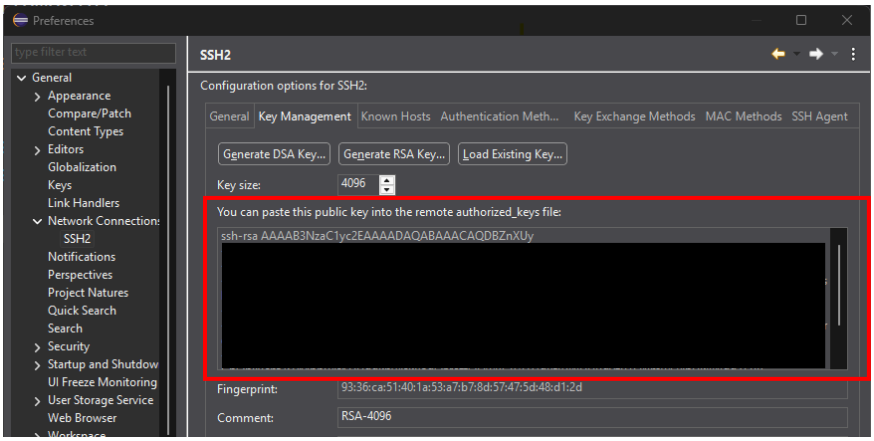


2. Copy the Public Key

In Eclipse, locate your generated or loaded key in the Key Management section. The public key content will be displayed, or you can access it from your ~/.ssh/ directory (e.g., ~/.ssh/id_rsa.pub). Copy the entire public key text.

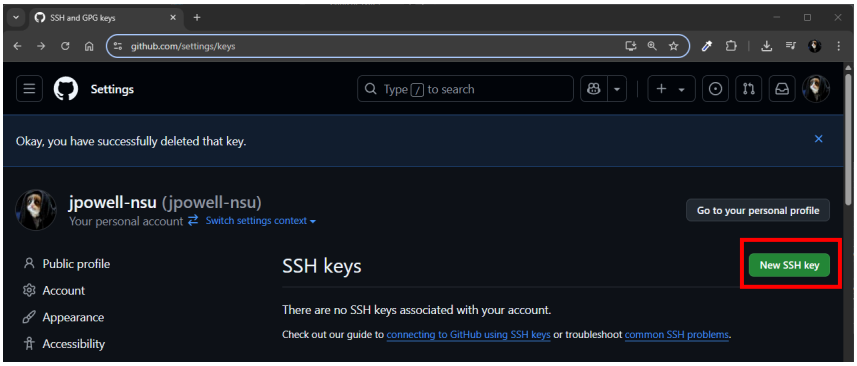
If you create a new key, type in a passphrase, and save the private key. It will also save your public key file.

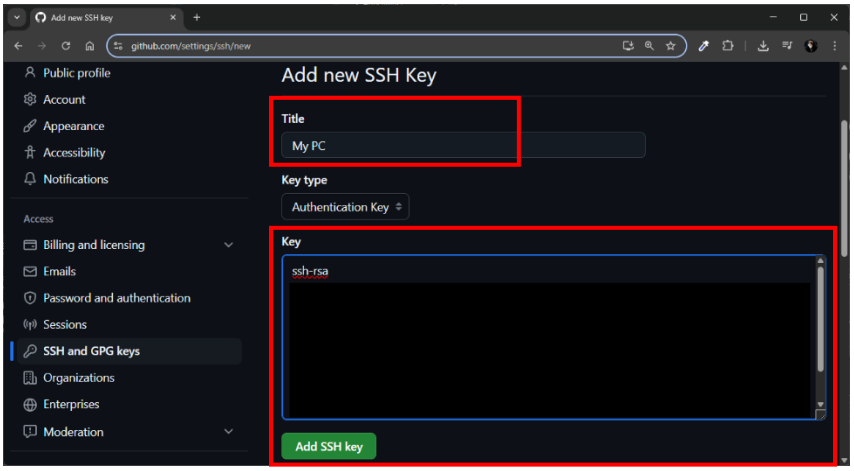
Click Apply and Close



3. Register Public Key on GitHub.

Sign in to your GitHub account and go to Settings -> SSH and GPG keys. Click New SSH key or Add SSH key. Paste the copied public key into the Key field. Give your key a descriptive title and click Add SSH key.



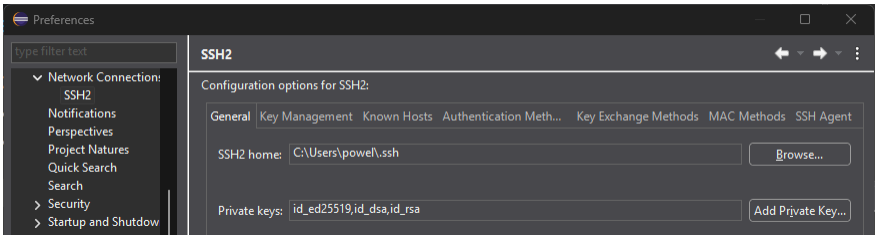


4. Configure Eclipse to Use the Key

It should already be set up, but double-check that Eclipse uses your SSH key.

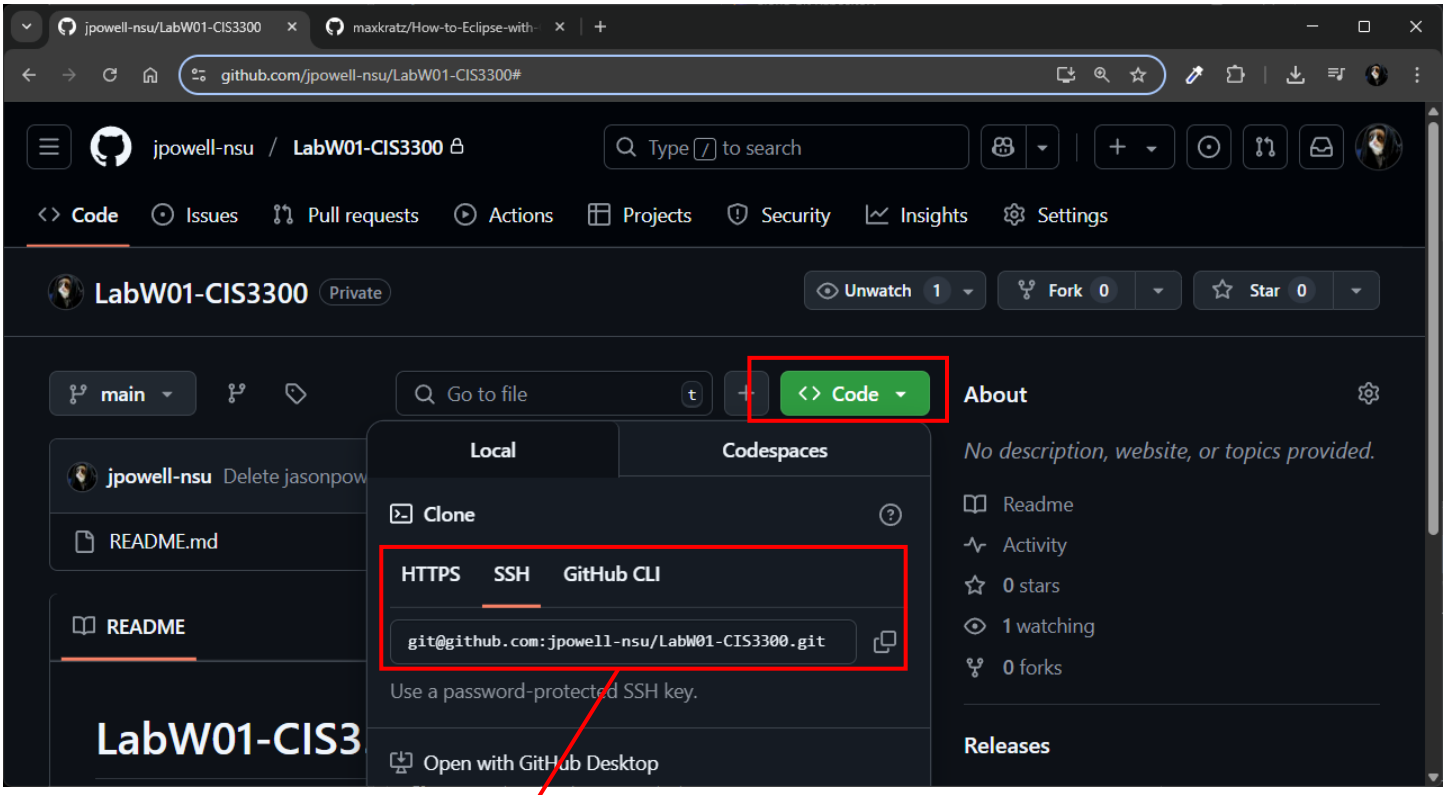
Window -> Preferences -> General -> Network Connections -> SSH2 -> General

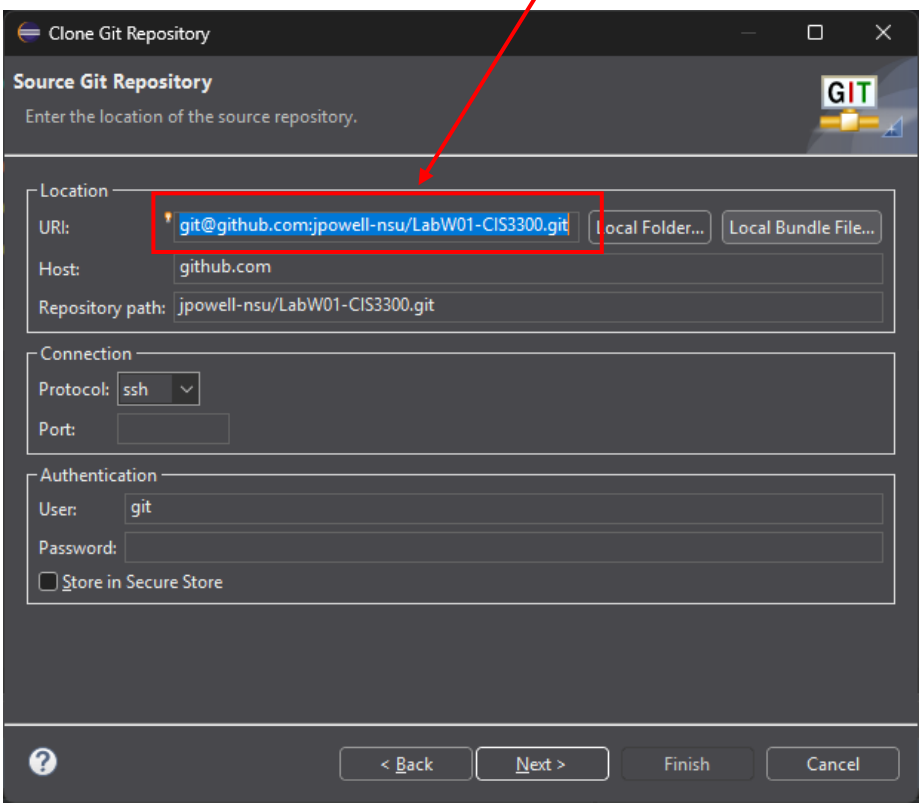
It should point to your .ssh folder and look for default file names. If you name your private key file differently, you may need to add its name to the list.



5. When you clone a repository or push changes to your GitHub repository from Eclipse, you should be able to use the SSH key for authentication.

Get the SSH URI for your repository in GitHub. Select Clone a Git Repository in Eclipse. Paste the URI, and the rest of the field should autofill. You do not need a password. When you click Next, it may open a series of windows to establish the connection. In my case, I saved the information whenever possible. Once complete, you should have access to your repository, as shown in the previous documents.


jpowell-nsu / LabW01-CIS3300
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
LabW01-CIS3300
Private
Unwatch 1
Fork 0
Star 0
main
Go to file
Code
Local
Codespaces
Clone
HTTPS
SSH
GitHub CLI
git@github.com:jpowell-nsu/LabW01-CIS3300.git
Use a password-protected SSH key.
Open with GitHub Desktop
README.md
README
About
No description, website, or topics provided.
Readme
Activity
0 stars
1 watching
0 forks
Releases


Clone Git Repository
Source Git Repository
Enter the location of the source repository.
GIT
Location
URI:
git@github.com:jpowell-nsu/LabW01-CIS3300.git
Local Folder...
Local Bundle File...
Host:
github.com
Repository path:
jpowell-nsu/LabW01-CIS3300.git
Connection
Protocol:
ssh
Port:
Authentication
User:
git
Password:
Store in Secure Store
< Back
Next >
Finish
Cancel