How to Access GitLab via SourceTree (UPDATED)

in Tutorials

1 JANUARY, 2019   .   *BY* CALRANCE KHO

If you want to access GitLab via SourceTree, you can connect it with an account profile now. Another method is to connect GitLab by using the clone function with SSH key.

What Will We Learn (Method 1)

- Access GitLab with personal access token

What Will We Learn (Method 2)

- Create SSH key
- Access GitLab via SourceTree

Method 1: Connect with Remote (Account Profile)

Since SourceTree 3.0, it adds support for additional remote hosting services. They are GitHub Enterprise,

Microsoft Visual Studio Team Services, and GitLab (Cloud, CE, EE).

Create Personal Access Token

To connect GitLab with remote, personal access token is required for authentication. To create personal access token, login GitLab and go to **Settings**.
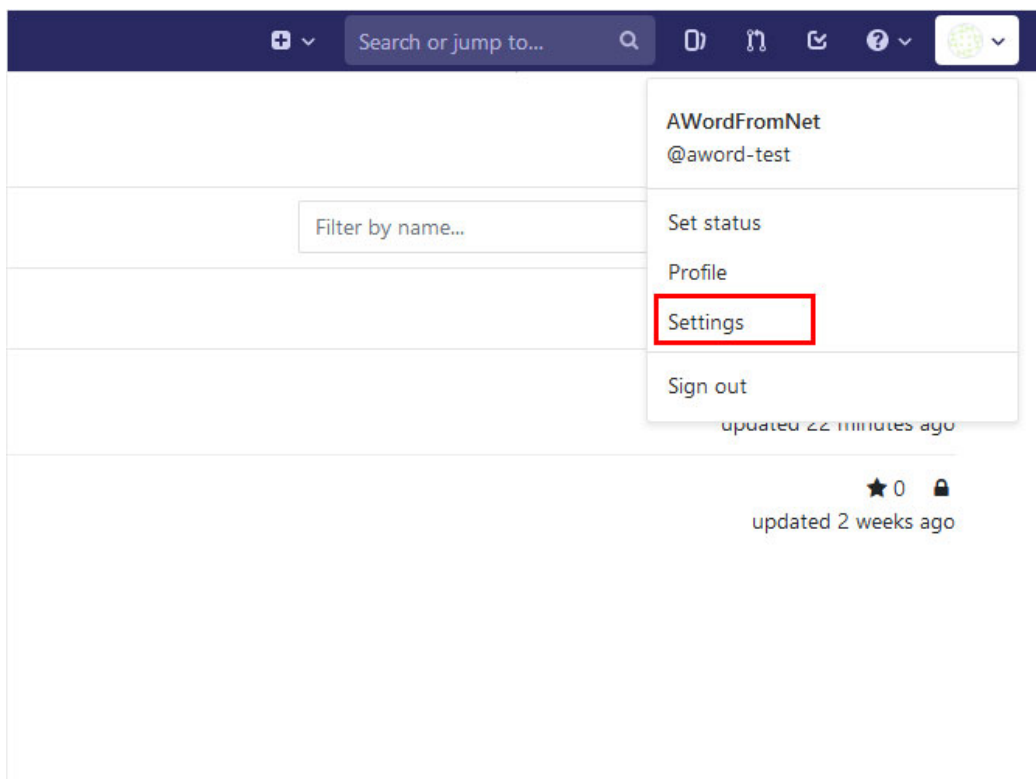
After that, select **Access Tokens** from the sidebar. You will create your personal access token here. Give access token a name. You are able to set an expiry date for your access token. By leaving it blank, access token will never expire. Next check the **api** and **read_user** checkbox, it is required for SourceTree to connect to GitLab remotely. Click on **Create personal access token** to complete the setup.

Add a personal access token

Pick a name for the application, and we'll give you a unique personal access token.

Name

gitlab-token

Expires at

2019-02-28 🗓

Scopes

☑ **api**
Grants complete read/write access to the API, including all groups and projects.

☑ **read_user**
Grants read-only access to the authenticated user's profile through the /user API endpoint, which includes username, public email, and full name. Also grants access to read-only API endpoints under /users.

☐ **read_repository**
Grants read-only access to repositories on private projects using Git-over-HTTP (not using the API).

☐ **read_registry**
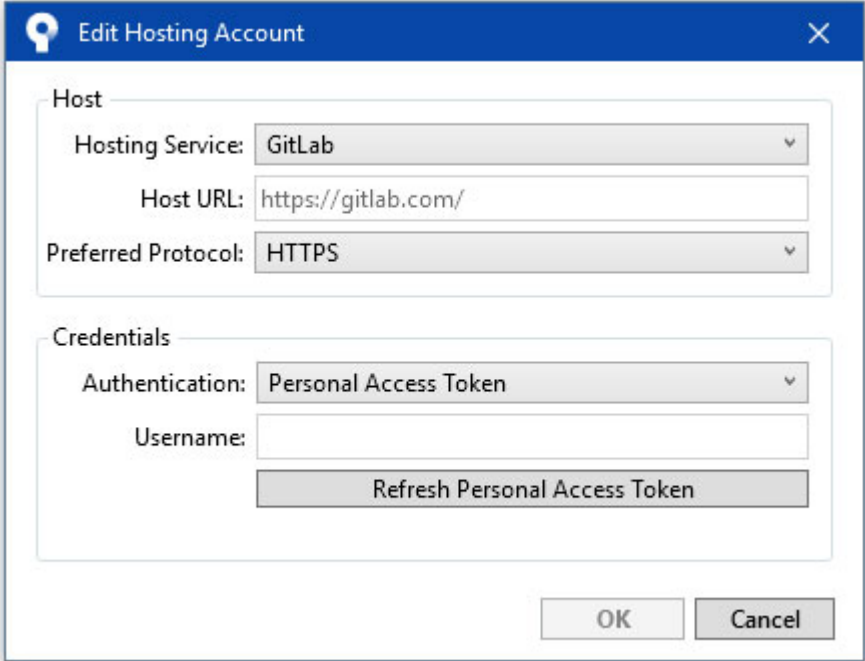Grants read-only access to container registry images on private projects.

Create personal access token

Next open up your SourceTree, click on **Remote** and click on **Add an account**.
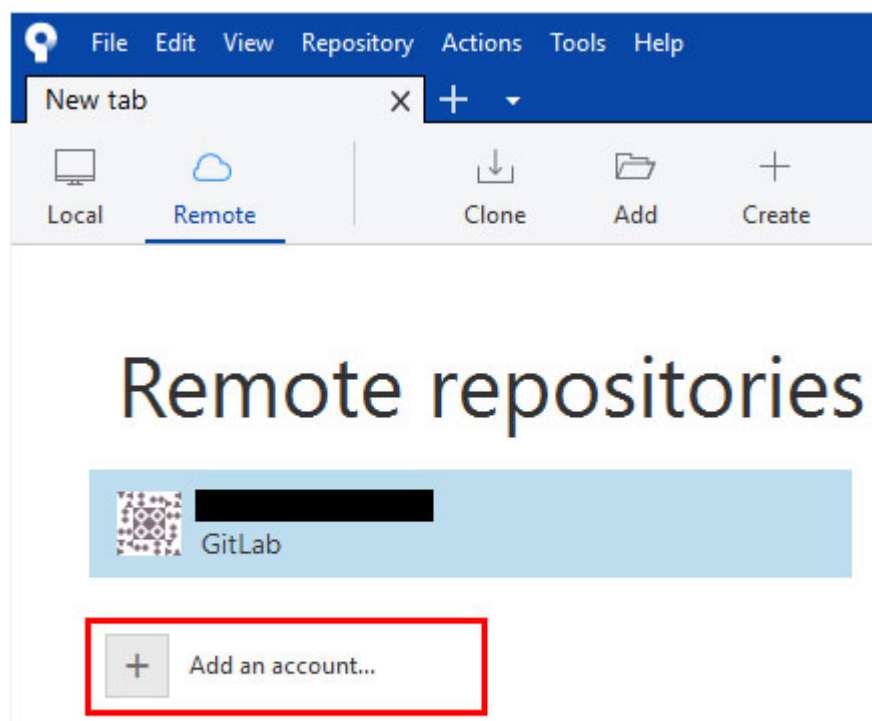
Select GitLab as the hosting services, and choose **HTTPS** as the Preferred Protocol. Click on **Refresh Personal Access Token**.

You will be prompted to enter the **username** and **password**. After entering the username and password then press **OK** to continue.
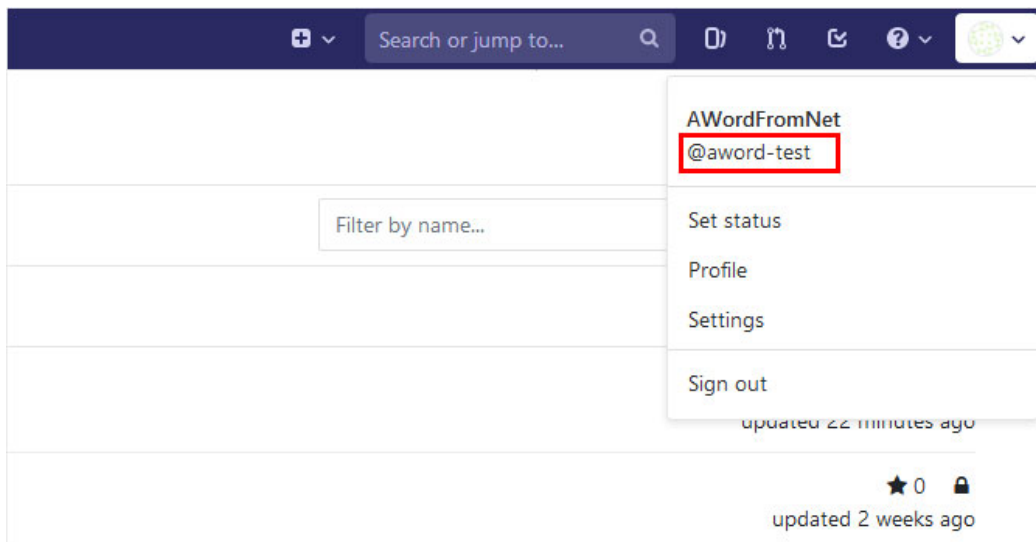
Do take note that email address you used to login GitLab is not your username. If you do not know your username. Go to GitLab and click on the top right profile icon. The text after @ is your username. In this example, it would be aword-test.

Then you have completed the steps to connect GitLab with SourceTree. If you connected successfully, you will able to see your repositories in SourceTree.
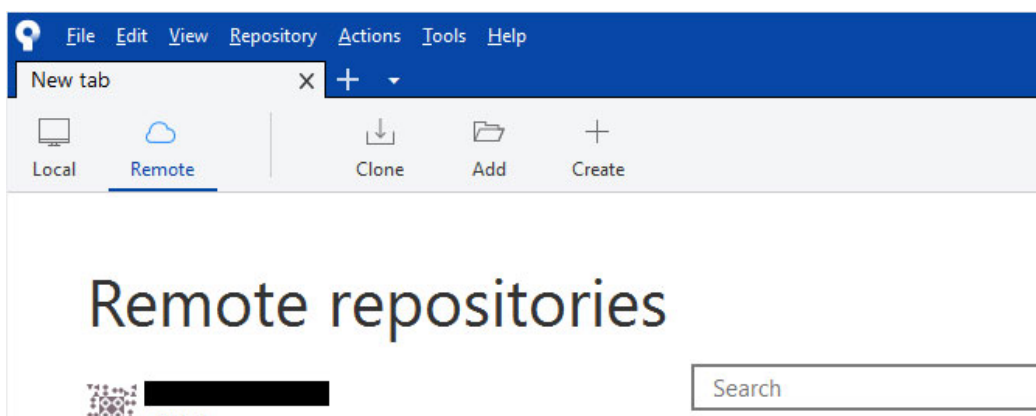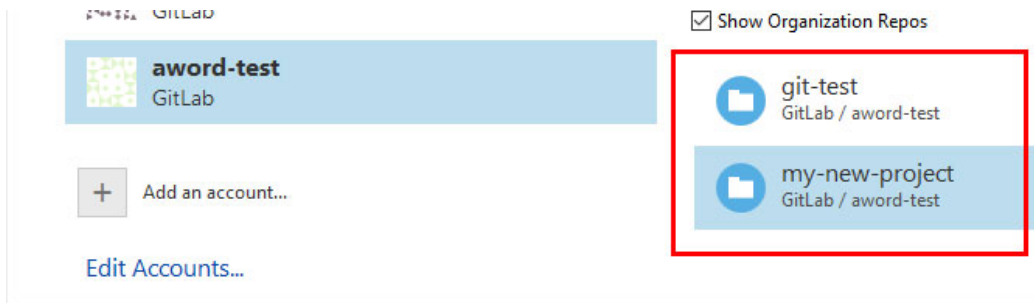
Method 2: Connect with SSH Key

This method is useful for users who are not using SourceTree 3.0 and above.

What is SSH Key

The Secure Shell (SSH) key is based on the principle of Public-key cryptography. It is used as a way of authentication. By generating an SSH key pair, you will get a private key and a public key. You can image public key as a padlock. You can distribute it to any systems on the Internet. Private key is like a key of the padlock. You should keep the private key securely and avoid it from being compromised. To access any system with public key, you need to use the private key to authenticate yourself.
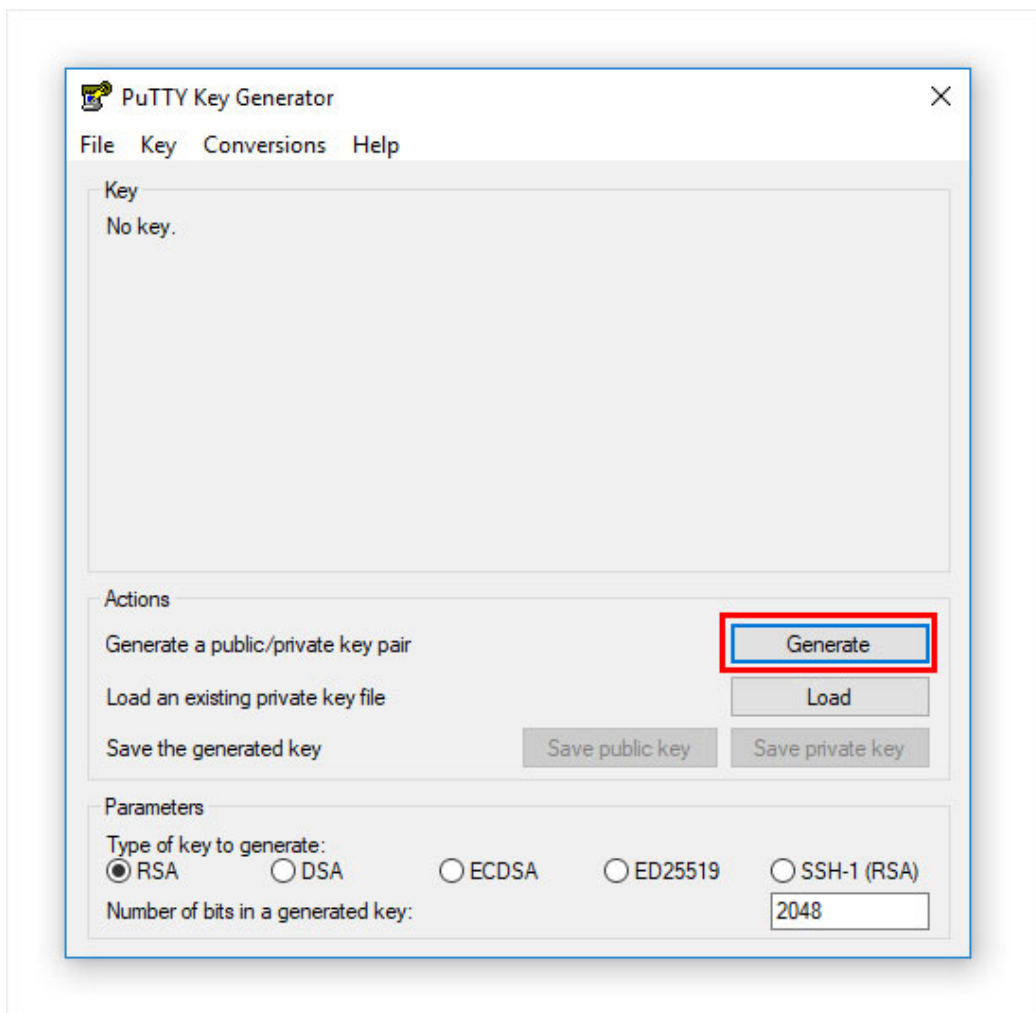
Create SSH Key

To create SSH key, you need to generate your own SSH key with SourceTree. Go to **Tools** and select **Create or Import SSH Keys**.
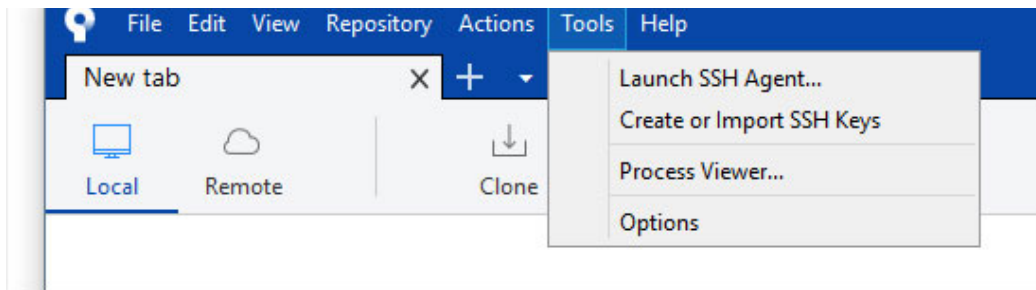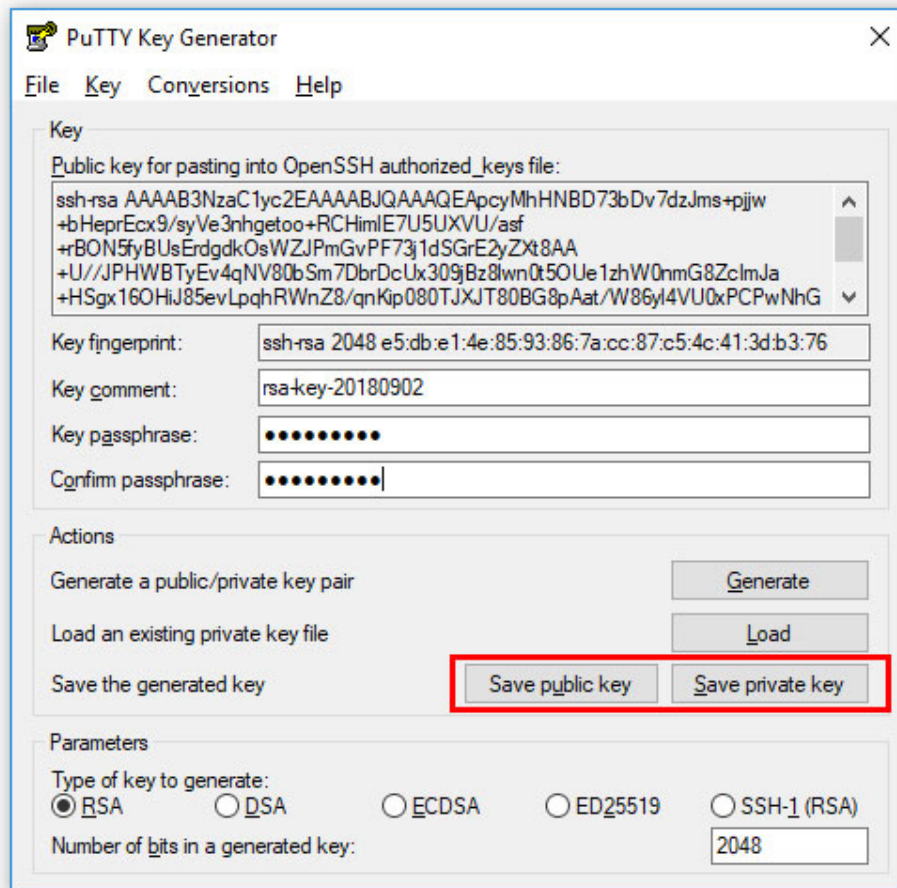
Click on **Generate** to create SSH Key. Move your mouse cursor within the PuTTY Key Generator continuing to generate the SSH Key.


After SSH Key is generated, type your own **Key passphrase** for your SSH Key. Key passphrase is like a
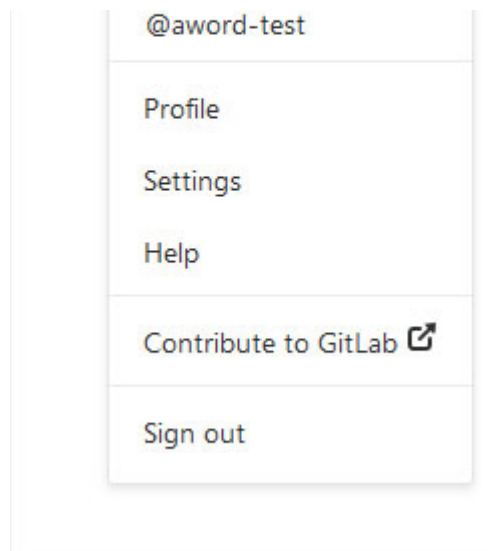
password to protect your SSH private key. Save your public key and private key for future use.

Add SSH Public Key to GitLab

To add your SSH Key to GitLab, you need to log into you GitLab account. Click on your profile icon on the top right corner, then select **Settings**. Click on **SSH Keys** on the side menu.

@aword-test

Profile

Settings

Help

Contribute to GitLab ⤢

Sign out
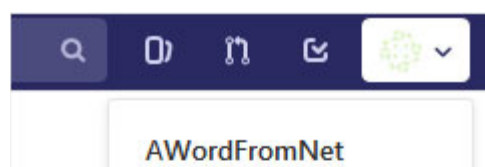
Copy your public key inside your PuTTY Key Generator, then paste it into GitLab. Click on **Add Key** to add your SSH Key into the GitLab.

Add an SSH key

To add an SSH key you need to generate one or use an existing key.

Key

Paste your public SSH key, which is usually contained in the file '~/.ssh/id_rsa.pub' and begins with 'ssh-rsa'. Don't use your private SSH key.

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAQEApcyMhHNBD73bDv7dzJms+pjjw+bHeprEcx9/syVe3nhgetoo
+RCHimIE7U5UXVU/asf+rBON5fyBUsErdgdkOsWZJPmGvPF73j1dSGrE2yZXt8AA+U
//JPHWBTyEv4qNV80bSm7DbrDcUx309jBz8lwn0t5OUe1zhW0nmG8ZcImJa+HSgx16OHiJ85evLpq
hRWnZ8/qnKip080TJXJT80BG8pAat
/W86yl4VU0xPCPwNhGYU8zE1UiZKoKPp8O1qiQusuAmzEODgiFd48amk3UPNyrzw7fvA9jSR4Q0q
49qDheTBHX+OHhLV7s+21xM6hBMoFqarIPp4kRLcLQGfFw== rsa-key-20180902
```
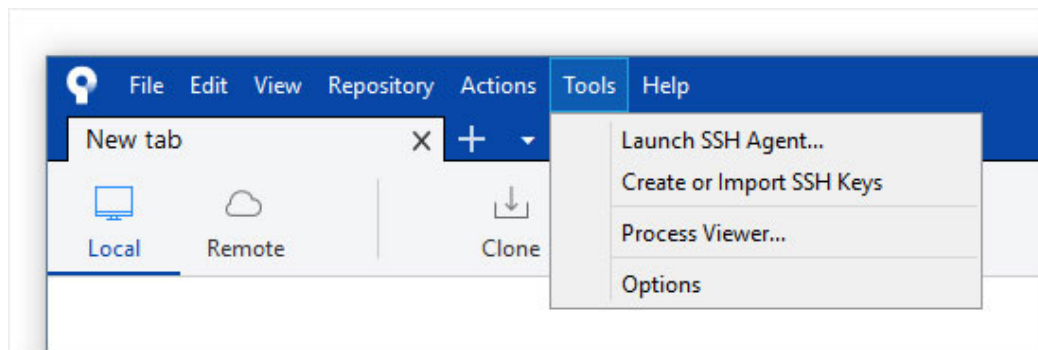
Title

rsa-key-20180902

Name your individual key via a title

Add key

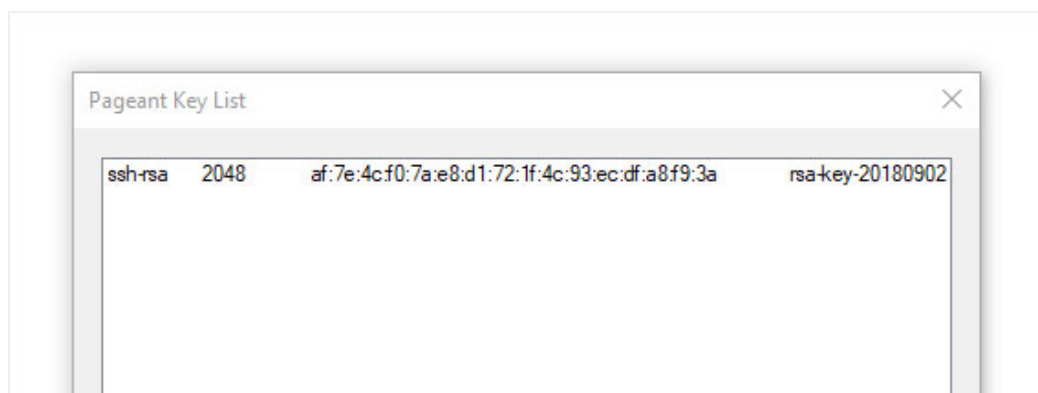# Add SSH Private Key to SourceTree

To add private key to SourceTree, Click on **Tools** then click **Launch SSH Agent...**
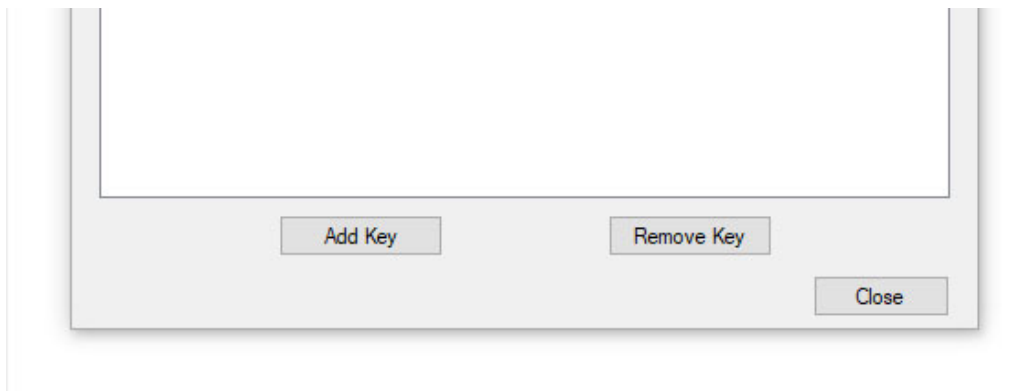

You will find SSH Agent named as Pageant on the taskbar. Double click on Pageant.
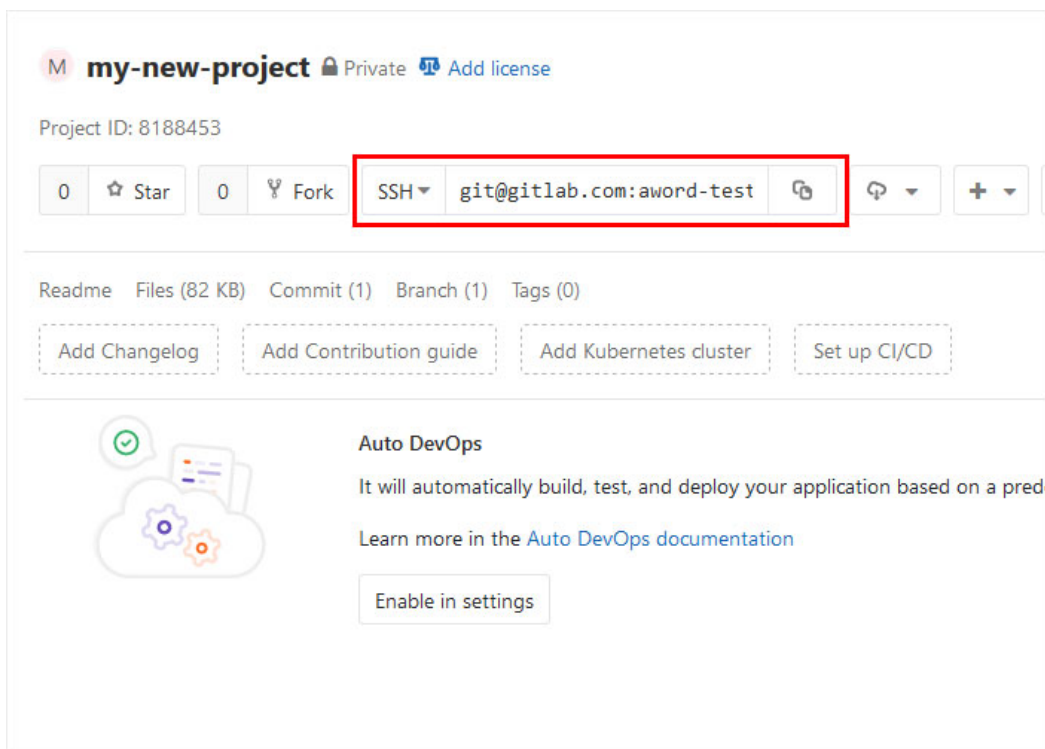

Now click on **Add Key**. Add your private key which have been saved earlier.

Clone Repository to Access GitLab via SourceTree

Go into your projects in your GitLab. Copy the SSH URL.

In SourceTree, click on **Clone**. Paste the repository URL into Source Path. Click on **Clone**, you have successfully connected GitLab with SourceTree. Now you are able to pull and push your code to GitLab with SourceTree.

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name

Email

Website

POST COMMENT

ABOUT

A Word From Net is a blog that publishes tutorials and articles about web design and development.

ME

Calrace Kho
Front-End Developer

MOTTO

Practice Makes Perfect