HOW TO GENERATE SSH KEYS WITH TORTOISEGIT (WINDOWS)

Jordan Johnson
May 2024

INTRODUCTION

Using an SSH key pair is a secure way of authenticating and accessing source code stored on remote servers. TortoiseGit is an easy-to-use GUI for Git. This guide will describe how to generate an SSH pair so you can use TortoiseGit to connect with your version control system via SSH. This guide uses the Gitlab version control system but the steps are similar for most other popular tools such as GitHub.

PROCESS OVERVIEW

To properly connect to GitLab using SSH and TortoiseGit follow the below steps:

1. Install the required software
2. Generate SSH keys via OpenSSH and the **ssh-keygen** command
3. Generate SSH keys via TortiseGit's instance of PuTTY Key Generator
4. Register public SSH keys to GitLab

INSTALL THE REQUIRED SOFTWARE

To connect to GitLab using TortoiseGit and SSH ensure the following software is installed on your machine:

- Git: https://git-scm.com/download/win
- TortoiseGit: https://tortoisegit.org/download/
- OpenSSH (Version 6.5 or later): https://learn.microsoft.com/en-us/windows-server/administration/openssh/openssh_install_firstuse?tabs=gui
- PuTTy: https://www.putty.org/

Note: OpenSSH is automatically installed on versions of Windows 10 or later.

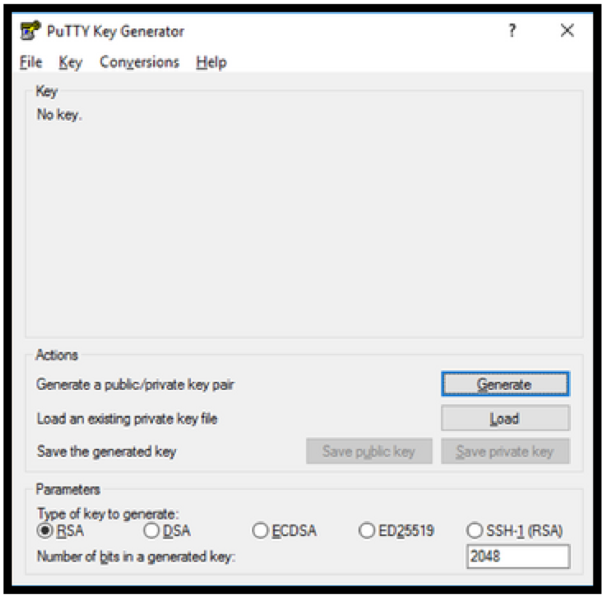
USING THE TORTISEGIT PUTTY KEY GENERATOR TO CREATE SSH KEYS

Important: In order to use TortoiseGit with GitLab via SSH, you must use the version of the PuTTY Key Generator that is bundled with your installed version of TortoiseGit to generate your key pair. If you install PuTTY separately, other versions may not be compatible with the version that comes with TortoiseGit. The default location for the PuTTY Key Generator instance that is installed with TortoiseGit is **C:\Program Files\TortoiseGit\bin.**

To generate an SSH key pair using the TortiseGit instance of the PuTTY Key Generator complete the following steps:

1. Navigate to **C:\Program Files\TortoiseGit\bin** and double-click **puttygen.exe** to open the PuTTY Key Generator.
2. At the bottom of the window select the type of key to generate.
3. Click **Generate** and start moving your mouse over the blank area to generate a key pair.
4. Enter a key comment.
5. Enter a passphrase and confirm it.
    a. You have the option to leave the passphrase blank.
6. Copy ***all*** of the text from the box containing the public key.
7. Open Notepad or Notepad++.
8. Paste ***all*** of the text into a text file.
9. In PuTTY Key Generator click **Save Private Key**.
    a. Ensure that you save the private key first to avoid any file extension confusion or overwrites.
10. Click **Save Public Key**.
    a. Choose the location to save your public key. The default location is **C:\User\[YOUR USER]\.ssh**

Important: Always keep your private key secure.



No key.

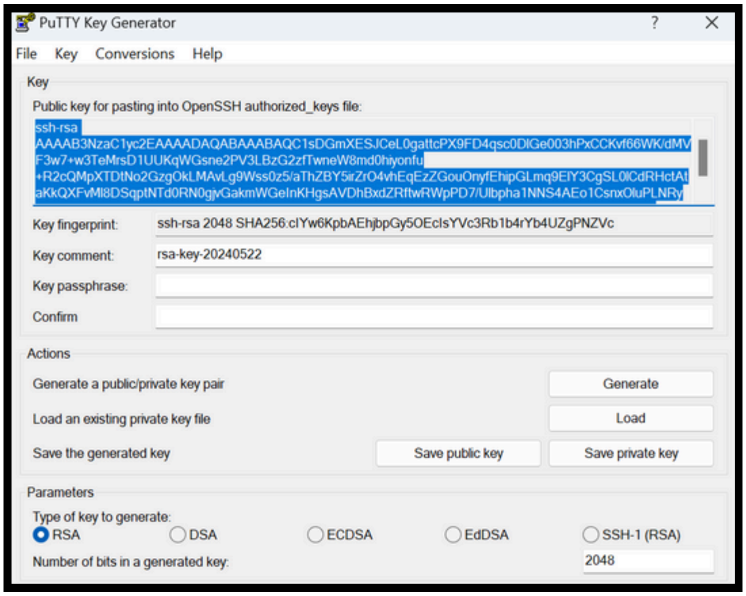
Generate a public/private key pair

Load an existing private key file

Save the generated key

Type of key to generate:
● RSA ○ DSA ○ ECDSA ○ ED25519 ○ SSH-1 (RSA)

Number of bits in a generated key: 2048



Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC1sDGmXESJCeL0gattcPX9FD4qsc0DlGe003hPxCCKvf66WK/dMV
F3w7+w3TeMrsD1UUKqWGsne2PV3LBzG2zfTwneW8md0hiyonfu
+R2cQMpXTDtNo2GzgOkLMAvLg9Wss0z5/aThZBY5irZrO4vhEqEzZGouOnyfEhipGLmq9ElY3CgSL0lCdRHctAt
aKkQXFvMl8DSqptNTd0RN0gjvGakmWGelnKHgsAVDhBxdZRftwRWpPD7/Ulbpha1NNS4AEo1CsnxOluPLNRy
```

Key fingerprint: ssh-rsa 2048 SHA256:clYw6KpbAEhjbpGy5OEclsYVc3Rb1b4rYb4UZgPNZVc

Key comment: rsa-key-20240522

Key passphrase:

Confirm

Generate a public/private key pair

Load an existing private key file

Save the generated key

Type of key to generate:
● RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

ADDING PUBLIC SSH KEYS TO GITLAB

After generating your SSH keys, you will now need to add the generated public key to your GitLab account by completing the following steps:

1. In File Explorer, navigate to the folder where you stored your SSH keys after generating them. The default location is **C:\User\[YOUR USER]\.ssh**.
   a. **.ssh** is a hidden folder. To view hidden folders in Windows 11, in File Explorer, navigate to **View > Show** and select **Hidden Items**. In previous Windows versions, navigate to **Folder Options** and select **Show Hidden Files, Folders, and Drives**.
2. Open the **.pub** file associated with the keys you generated in Notepad and copy all the text from the file (ctrl + A, ctrl + C).
   a. In File Explorer, navigate to **View > Show** and select the **file extensions** checkbox to display file extensions.
3. In Gitlab, navigate to **User Settings > SSH Keys**.
4. Click **Add New Key**.
5. In the key field, paste in ***all*** the text from the **.pub** file.
6. Enter a title for the key.
7. Set the usage type to **Authentication & Signing**.
8. Set the expiration date (optional).
9. Click **Add Key**.

TEST YOUR SSH KEY CONNECTION

To test that you can use the SSH keys you generated to connect to GitLab:

1. After registering your SSH key in GitLab, open a terminal and run **ssh -T git@gitlab.[YOUR GITLAB URL].com**.
2. If this your first time connecting to the GitLab host, verify the authenticity of the host.
3. Run **ssh -T git@gitlab.[YOUR GITLAB URL].com** again and you should get a **Welcome to GitLab, @username!** message.
4. If the welcome message doesn't appear, you can troubleshoot by running SSH in verbose mode using **ssh -Tvvv git@gitlab.[YOUR GITLAB URL].com**.

SET THE DEFAULT SSH KEY IN PUTTY

1. Start PuTTY and navigate to **Connection > SSH > Auth > Credentials.**
   a. The default location of the TortiseGit instance of PuTTY is **C:\Program Files\TortoiseGit\bin.**
2. Select the private key file that you generated using the TortoiseGit PuTTY Key Generator.
3. Navigate to **Session**, select default settings, and click **Save**.
   a. Now PuTTY will use your private key for all new connections.

Important: When cloning a repository using TortoiseGit ensure the **load putty key** option is selected and is referencing the correct key file.

RESOURCES

SSH.com
https://www.ssh.com/academy/ssh/keygen

Gitlab SSH Documentation
https://docs.gitlab.com/ee/user/ssh.html

TortiseGit SSH Documentation
https://tortoisegit.org/docs/tortoisegit/tgit-ssh-faq.html#tgit-ssh-faq-defaultkey