Setup Git Client with TortoiseGit on Windows 10

TortoiseGit is a kind of family project with TortoiseSVN and TortoiseCVN, so if you are familiar with that, you will be easy to use it.


TortoiseGit is easy to use, because it is integrated with Windows shell as well as lots of development utilities, but I won't cover about that at the moment.

To setup an environment, we need following three components:

- Git for Windows
- TortoiseGit Tool
- PuTTYgen Tool

STEP 1. Install git on Windows 10

For git, you need to download a core component at https://git-scm.com/downloads
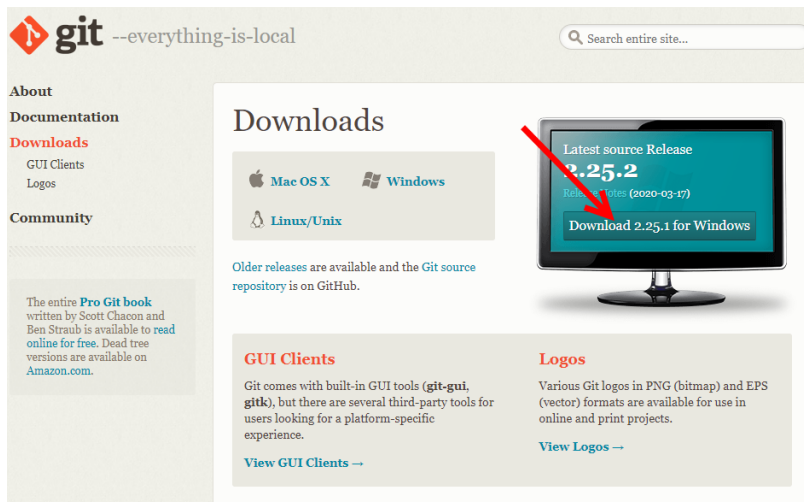
Once download is done, you need to install it on your PC.

STEP 2. Install PuTTYgen and Generate public/private key pairs

- Download PuTTYgen at https://www.puttygen.com/download-putty
- Install. I will skip installation progress at the moment.

STEP 3. Install TortoiseGit client

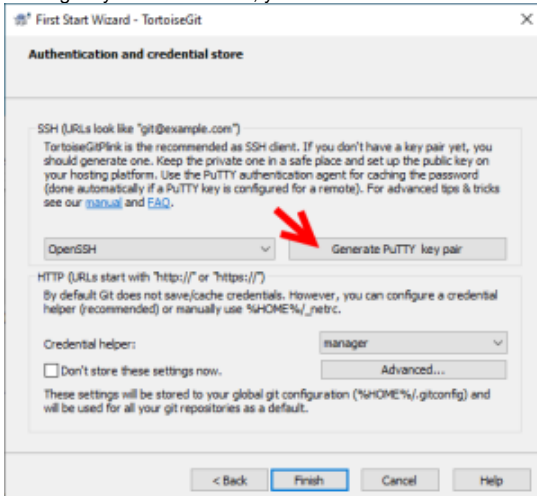- Download TortoiseGit at https://tortoisegit.org/download/

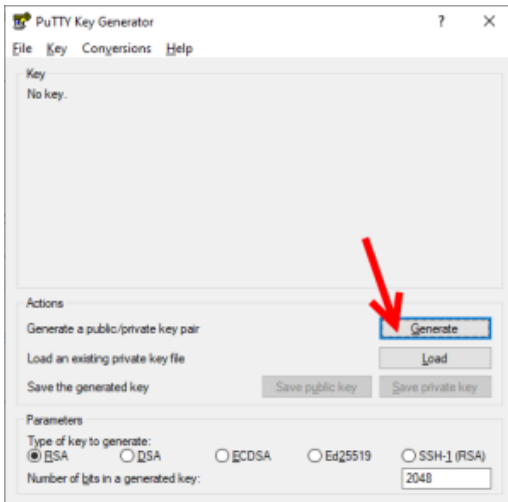- Once installation is done, you can run start wizard.
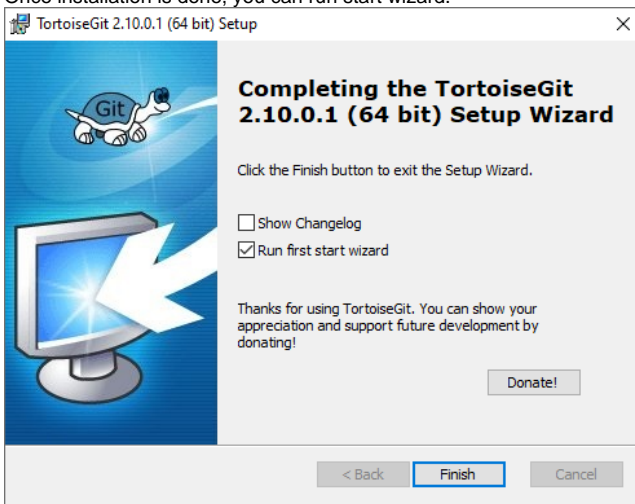
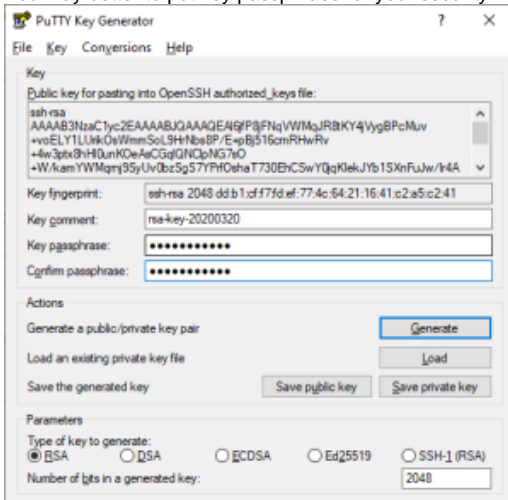It will check Git for Windows that we installed earlier.
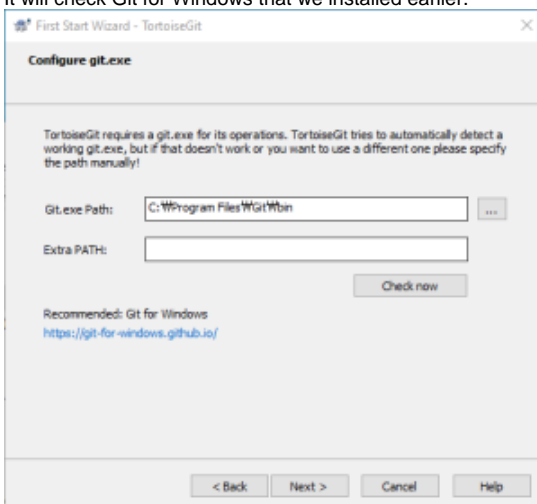
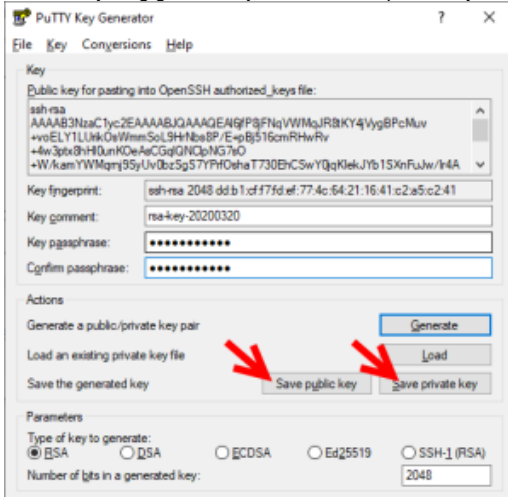As long as you will use SSH, you'd better click "Generate PuTTY key pair"

You may better to put key passphrase for your security

Once everything goes fine, just click "Save public key" and "Save private key"

ⓘ Once key is created, you need to ask admin to add your public key to git repository in server. If you use github, you need to check other references. If your environment is based on CentOS 8, please refer at Setup Git server on CentOS 8#STEP3. Add client's public key to /home/git/.ssh/authorized_keys

STEP 4. Create a repository on your PC

In order to manage the source code efficiently, you need to have your own location on your PC, so you can add new features, modify the legacy codes without any dependency in external sources.

- In some cases, someone may need to develop some parts at the same time - it will cause lots of issues such as making errors in the previously verified codes.
- If you have your own codes based on the master repository, you will be able to develop your code more efficiently.

The action you should do on windows is run following commands on DOS Shell. For your information, I will use project name as **chun** below.

```
mkdir c:\chun.git
c:\
cd c:\chun.git
```

Note. if you need to use the different directory, you can simply replace C:\Repository\chun by your preferred location.

If you have your own private server, having no dependency with others, which can run full features, it will be the best development environment - some people usually use VM like QEMU.

To be updated....