

Git Tutorial



Backlog lets you manage projects with Git integration

Learn more →

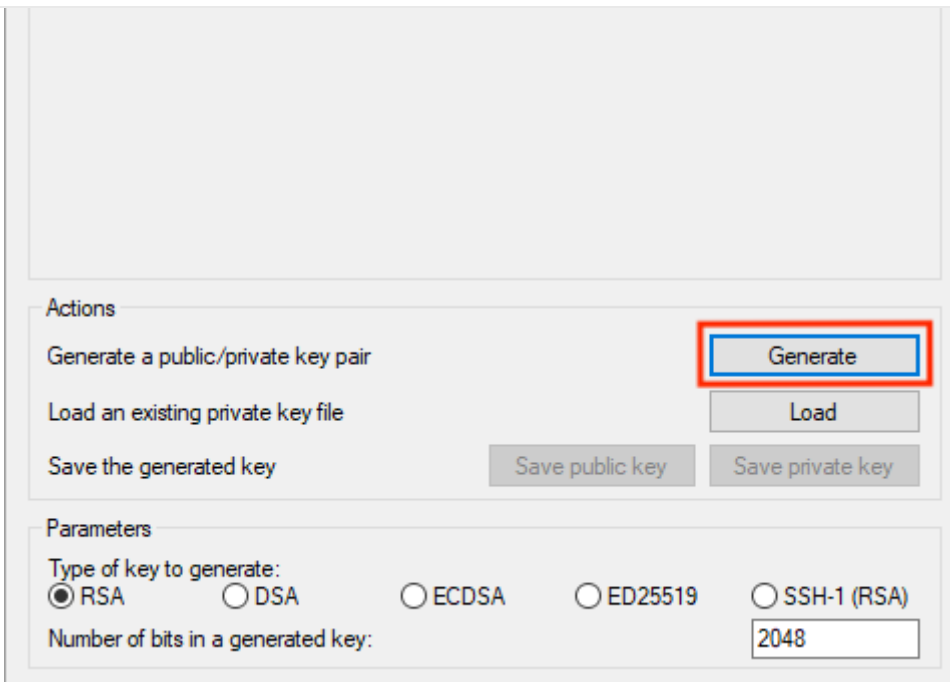
SSH connection settings

- [Set up an SSH connection \(Windows\)](#)
- [Set up an SSH connection \(Mac\)](#)
- [Set up an SSH connection \(Console\)](#)
- [Register my SSH public key to Backlog](#)

Set up an SSH connection (Windows)

Go to Start menu > All Programs > Open TortoiseGit and start Putty Key Generator.

Git Tutorial



The screenshot shows the PuTTY Key Generator window. The 'Generate' button is highlighted with a red rectangular box. The window is divided into three sections: 'Actions', 'Parameters', and a 'Key' section at the top.

Actions:

- Generate a public/private key pair (with the **Generate** button highlighted)
- Load an existing private key file (with a **Load** button)
- Save the generated key (with **Save public key** and **Save private key** buttons)

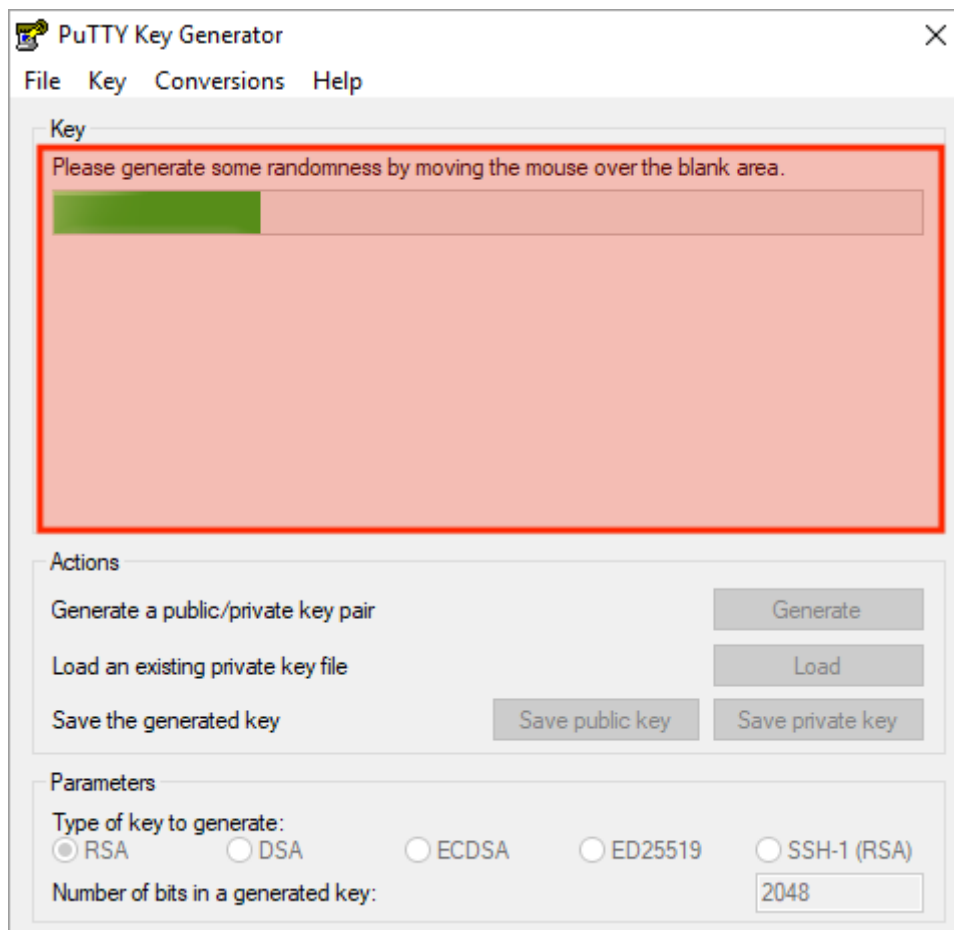
Parameters:

Type of key to generate:

- ☒ RSA
- ☐ DSA
- ☐ ECDSA
- ☐ ED25519
- ☐ SSH-1 (RSA)

Number of bits in a generated key:

Click on "Generate" and move the mouse cursor within the red frame until the progress bar completes. This will randomly generate a key.



The screenshot shows the PuTTY Key Generator window. The 'Key' section is highlighted with a red rectangular frame. It contains a progress bar and a text instruction: "Please generate some randomness by moving the mouse over the blank area." The progress bar is partially filled with green, indicating the key generation process is in progress.

Key

Please generate some randomness by moving the mouse over the blank area.

Actions:

- Generate a public/private key pair (with a **Generate** button)
- Load an existing private key file (with a **Load** button)
- Save the generated key (with **Save public key** and **Save private key** buttons)

Parameters:

Type of key to generate:

- ☒ RSA
- ☐ DSA
- ☐ ECDSA
- ☐ ED25519
- ☐ SSH-1 (RSA)

Number of bits in a generated key:

Git Tutorial

Key

Public key for pasting into OpenSSH authorized keys file:

ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAQEAwZHUKw22nI78VkHaRSq0phWcu9H1V7MG
sqaW8QU3jaAwA2kpV+kECil9q
+uBjtUJ1a3bK2qiopikuSGWUeTveVSe9Lj34xZxLxssnOU1V4tuTq
+U6zjuQDI32jVhPABasOET0XisPU7o

Key fingerprint: ssh-rsa 2048 5b:64:6a:a0:cd:ad:34:51:d8:b1:5a:a3:31:30:7a:e1

Key comment: rsa-key-20180315

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair

Generate

Load an existing private key file

Load

Save the generated key

Save public key

Save private key

Parameters

Type of key to generate:
☒ RSA ☐ DSA ☐ ECDSA ☐ ED25519 ☐ SSH-1 (RSA)

Number of bits in a generated key: 2048

You can view the public key again in future by loading the .ppk file into Putty Key Generator..

In order to configure SSH connection when we push with TortoiseGit, right-click on TortoiseGit, then choose "Manage".

Git Tutorial

Remote: ...

Destination

☒ Remote: Manage

☐ Arbitrary URL:

Options

Force: May discard ☐ known changes ☐ unknown changes

☐ Use Thin Pack (For slow network connections)

☐ Include Tags

☒ Autoload Putty Key

☐ Set upstream/track remote branch

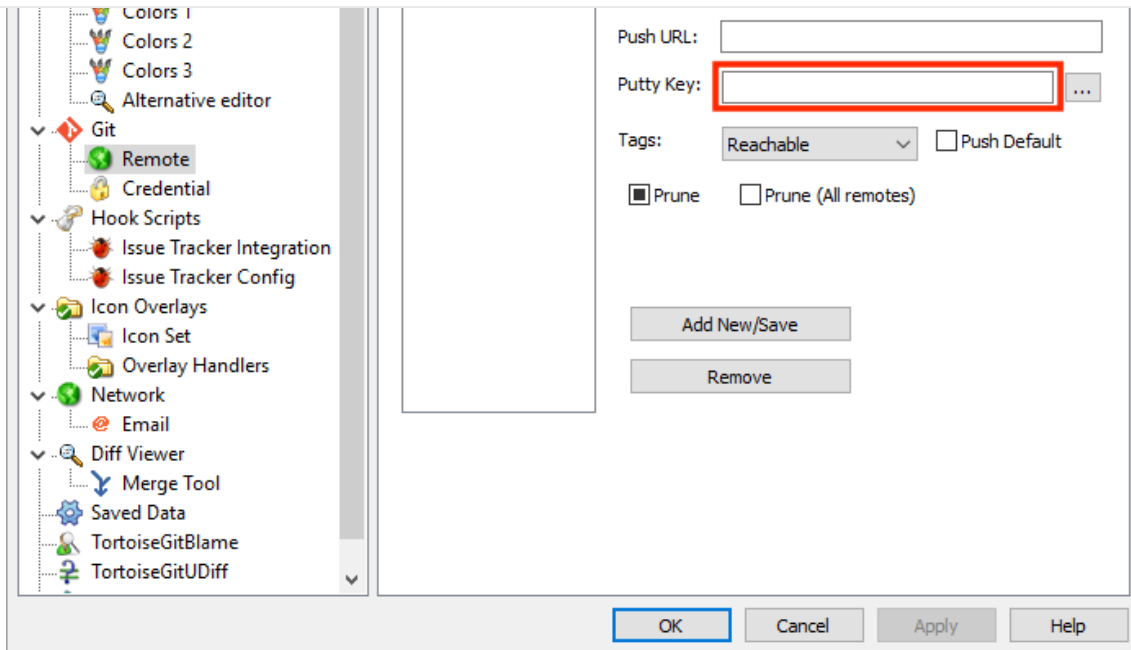
☐ Always push to the selected remote archive for this local branch

☐ Always push to the selected remote branch for this local branch

Recurse submodule

With "origin" selected in the remote column, assign the SSH path to the URL field, and assign the ppk file path that we have just saved to the Putty Key field. Click "Add New/Save". Now, the origin remote that is associated with the SSH URL path will be added to the list of remote in this repository. Click "OK" to complete.

Git Tutorial



Set up an SSH connection (Mac)

Open terminal under application/utility and execute the following command.

```
$ ssh-keygen
```

You will see the following output. You may optionally enter the passphrase for this generated rsa key pair.

Continue by hitting enter if you do not wish to set a passphrase.

```
Generating public/private rsa key pair.
Enter file in which to save the key
(/Users/eguchi/.ssh/id_rsa): <Press Enter key>
Created directory '/Users/eguchi/.ssh'.
Enter passphrase (empty for no passphrase):
<Input passphrase>
Enter same passphrase again: <Please enter the
```

Git Tutorial

```
/Users/eguchi/.ssh/id_rsa.pub.  
The key fingerprint is:  
57:15:3c:ca:f2:dc:27:6d:c2:9a:88:d0:70:cf:8d:31  
eguchi@eguchi-no-MacBook-Air.local  
The key's randomart image is:  
+--[ RSA 2048]-----+  
|           .o.  |  
|           .o   |  
|          ...   |  
|       . . E.o   |  
|       +So.O o .  |  
|       . ..+ + = +|  
|           . . . o = |  
|           . . o   |  
|                   |  
+-----+  

```

You can view the public SSH key with the following command.

```
$ cat ~/.ssh/id_rsa.pub
```

Output example

```
ssh-rsa  
AAAAB3NzaC1yc2EAAAADAQABAAQDkkJvxyDVh9a+zH1f7ZQc  
4RzttQwfK+sgWEr0aAgfnxdxQeDKxIqxI1SwyTY8oCcWzvpORuF  
5jvwKM82hincEWwI3wzcnVg2Mn8dH86b5m6REDzwRgozQ3lqrgv  
XGppOP/QXGEVhZ6Hy4m3b1wMjjrbYwmWIEYklgoGHyrldhAaDYc  
2oj4K+1q36iviCHxCori0FDmn2dzylRCI4S+A2/P7Y7rVfdT+8C  
Qmj9ikyGTEAW1zQCN7hVwYdjL  hoge@hoge.local
```

Git Tutorial

Execute the following command.

```
$ ssh-keygen
```

You will see the following output. You may optionally enter the passphrase for this generated rsa key pair.

Continue by hitting enter if you do not wish to set a passphrase.

```
Generating public/private rsa key pair.
Enter file in which to save the key
(/Users/eguchi/.ssh/id_rsa): <Press Enter key>
Created directory '/Users/eguchi/.ssh'.
Enter passphrase (empty for no passphrase):
<Input passphrase>
Enter same passphrase again: <Please enter the
same pass phrase again>
Your identification has been saved in
/Users/eguchi/.ssh/id_rsa.
Your public key has been saved in
/Users/eguchi/.ssh/id_rsa.pub.
The key fingerprint is:
57:15:3c:ca:f2:dc:27:6d:c2:9a:88:d0:70:cf:8d:31
eguchi@eguchi-no-MacBook-Air.local
The key's randomart image is:
+--[ RSA 2048 ]-----+
|           .o.  |
|           .o  |
|          ... . |
|      . . E.o   |
|      +So.O o . |
```

Git Tutorial

```
+-----+
```

You can view the public SSH key with the following command.

```
$ cat ~/.ssh/id_rsa.pub
```

Output example

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDkkJvxyDVh9a+zH1f7ZQc
4RzttQwfK+sgWEr0aAgfnxdxQeDKxIxqI1SwyTY8oCcWzvpORuF
5jvwKM82hincEWwI3wzcnVg2Mn8dH86b5m6REDzwRgozQ3lqrgv
XGppOP/QXGEVhZ6Hy4m3b1wMj jrbYwmWIEYklgoGHyrldhAaDYc
2oj4K+1q36iviCHxCORi0FDmn2dzylRCI4S+A2/P7Y7rVfdT+8C
Qmj9ikyGTEAW1zQCN7hVwYdjL  hoge@hoge.local
```

Copy and paste this key in the remote repository setting.

Register my SSH public key to Backlog

Login to Backlog with your account username that has access to the Git repository. Once logged in, click on "Personal Settings".

Git Tutorial

BLGGIT

My Pull Requests

Filters: Assigned to me (0) Created by me (2)

| Key | Summary | Related Issue | Updated |
|------------------------------|---------|---------------|---------|
| No pull requests to display. | | | |

My Issues

Batch Update Calendar Import

Filters: Assigned to me (0) Created by me (2)

Due date: All 4 Days (0) Due Today (0) Overdue (0)

| Key | Subject | Priority | Status | Due |
|-----|---------|----------|--------|-----|
|-----|---------|----------|--------|-----|

Katrina posted a comment on the issue BLGGIT-2 improve emails. Add BLGGIT/pull-request-test2#3 Improved copy of sign up email.

Katrina updated the pull request BLGGIT/pull-request-test2#3 Improved copy of sign up email [Related issue: BLGGIT-2]

Katrina added a new issue BLGGIT-2 improve emails

Click "Register SSH public key".

Dashboard Projects Recently Viewed Issues Recently Viewed Wikis Filters Upgrade

Git Tutorial

- Personal Settings
- User Information
- Emails
- Private addresses
- SSH public key**
- Applications
- API
- 2-factor authentication

Register SSH public key ?

Register your SSH public key to access Git repository through SSH.
It is not necessary if you are not going to use Git.

SSH Public Key

Memo

Submit

Registered SSH public keys

Paste the SSH public key into the text area and click "Register".

Git Tutorial

SSH public key

Applications

API

2-factor authentication

SSH Public Key

Memo

Submit

Registered SSH public keys

Prev

Next

Backlog has built-in Git repositories for every project

Try it free

Follow us on



English

Terms

Privacy

© 2021 Nulab, Inc. All rights reserved.