

Guide to SSH Keys and GitLab

What are SSH Keys?

SSH (Secure Shell) keys are a pair of cryptographic keys that can be used to secure communication between two parties. One key is private, and it should be kept secure on your local machine, while the other key is public and can be shared with others. SSH keys are commonly used for authentication, allowing secure access to remote servers and services without the need for a password.

Generating SSH Keys

To generate an SSH key pair, follow these steps:

1. Open a terminal on your local machine.
2. Use the following command to generate a new SSH key:

```
ssh-keygen -t rsa -b 2048 -C "example key"
```

3. Press Enter to accept the default file location and provide a secure passphrase when prompted.
4. Two files, `id_rsa` (private key) and `id_rsa.pub` (public key), will be generated in the `~/.ssh/` directory.

Adding SSH Key to the SSH Agent

1. Start the SSH agent:

```
eval "$(ssh-agent -s)"
```

2. Add your private key to the SSH agent:

```
ssh-add ~/.ssh/id_rsa
```

Adding SSH Key to GitLab

1. Copy the contents of your public key:

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

2. Log in to your GitLab account.
3. Navigate to **Settings > SSH Keys**.
4. Paste the copied public key into the "Key" field and give it a descriptive title.
5. Click **Add Key**.

Testing SSH Connection

To test if your SSH key is set up correctly:

```
ssh -T git@git.cs.bham.ac.uk
```

You should see a message indicating a successful connection.

Using SSH with Git

1. Change the remote URL of your Git repository to use SSH:

```
git clone git@git.cs.bham.ac.uk:zxq190/repo
```

Replace `repo` with the repository you want to clone.

2. Now you can push and pull from GitLab without entering a username and password.