# Creating SSH keys - Atlassian Documentation

This page describes how to create SSH keys.
SSH keys can be used to establish a secure connection with Bitbucket Server for:

- when you are performing Git operations from your local machine
- when another system or process needs access to repositories in Bitbucket Server (for example your build server)

The SSH key needs to be added to Bitbucket Server, and your Bitbucket Server administrator must have enabled SSH access to Git repositories, before you can make use of the key.

Supported key types are DSA and RSA2 – RSA1 is not supported.

*You can use an existing SSH key with Bitbucket Server if you want, in which case you can go straight to either SSH user keys for personal use or SSH access keys for system use.*

**On this page:**

## Creating an SSH key on Windows

### 1. Check for existing SSH keys

You should check for existing SSH keys on your local computer. *You can use an existing SSH key with Bitbucket Server if you want, in which case you can go straight to either SSH user keys for personal use or SSH access keys for system use.*

Open a command prompt, and run:

```
cd %userprofile%/.ssh
```

- If you see "No such file or directory", then there aren't any existing keys: ***go to step 3***.
- Check to see if you have a key already:

  ```
  dir id_*
  ```

  If there are existing keys, you may want to use those: go to either SSH user keys for personal use or SSH access keys for system use.

## 2. Back up old SSH keys

If you have existing SSH keys, but you don't want to use them when connecting to Bitbucket Server, you should back those up.

In a command prompt on your local computer, run:

```
mkdir key_backup
```

## 3. Generate a new SSH key

If you don't have an existing SSH key that you wish to use, generate one as follows:

1. Log in to your local computer as an administrator.
2. In a command prompt, run:

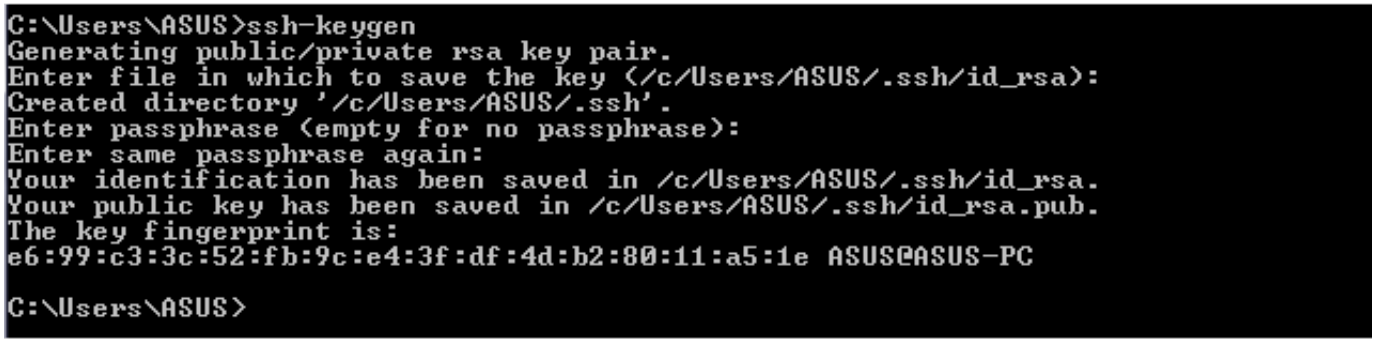
   ```
   ssh-keygen -t rsa -C "your_email@example.com"
   ```

   Associating the key with your email address helps you to identify the key later on.

   Note that the `ssh-keygen` command is only available if you have already installed Git (with Git Bash). You'll see a response similar to this:

3. Just press <Enter> to accept the default location and file name. If the `.ssh` directory doesn't exist, the system creates one for you.
4. Enter, and re-enter, a passphrase when prompted. The whole interaction will look similar to this:



# Creating an SSH key on Linux & Mac OS X

## 1. Check for existing SSH keys

You should check for existing SSH keys on your local computer. *You can use an existing SSH key with Bitbucket Server if you want, in which case you can go straight to either SSH user keys for personal use or SSH access keys for system use.*

Open a terminal and run the following:

```
1
cd ~/.ssh
```

- If you see "No such file or directory, then there aren't any existing keys: ***go to step 3***.
- Check to see if you have a key already:

  ```
  1
  ls id_*
  ```

- If there are existing keys, you may want to use them; go to either SSH user keys for personal use or SSH access keys for system use.

2. Back up old SSH keys

If you have existing SSH keys, but you don't want to use them when connecting to Bitbucket Server, you should back those up.

Do this in a terminal on your local computer, by running:

```
1
mkdir key_backup
```