Setting up SSH access

Setting up SSH

In this guide, we will help you get familiar with connecting to the server over SSH. While we expect you to probably be already familiar with it, you'll still find some helpful pointers in here in case you run into trouble.

In order to connect with the server, you will require an SSH client. While in general any client will do, the below guide will use PuTTY (for Windows users) and the default ssh program (for MacOS and Linux users). Windows users will have to download and install the SSH manually. The PuTTY client can be found here: https://www.chiark.greenend.org.uk/~sgtatham/putty/

Setting up access on Windows

When opening PuTTY, you'll be met with the session options. Here you'll need to enter the **host name** and **port** that PuTTY should connect to. You may also save and load sessions here, for easier connecting next time.

Next, you'll need to set up your connection settings. Proceed to Connection > Data and fill out the **auto-login username** field with your given username. This will make PuTTY automatically login with your username. Leaving this blank means you'll have to type in your username at the login prompt once you connect.

Proceed to Connection > SSH > Auth to set up your private key for the connection. If you do not yet have a keypair, you can generate one with the PuTTYgen tool. Dile Up will only accept connections authenticating with a key, so it's imperative that you have one. Once you have a public/private keypair, click **Browse...**and select your *private key file*. They usually end in the .ppk extension. **Keep in mind that new keypairs must be set up first on the server before you can log in with them!**

And that's all! Going back to the Session options, be sure to save your session for easy use next time. Once done, click **Open** to connect to Dile Up!

Setting up access on MacOS or Linux

MacOS and most Linux distributions usually come with the standard openSSH client installed. To check, you can open your terminal of choice and enter:

```
ssh -V
```

If openSSH is installed on your system, this will print out the version you have installed.

To log in using the openSSH client, you will need your username and your private key file for the public key you have supplied. If you do not yet have one, you may generate them with this command:

```
ssh-keygen
```

Keep in mind that new keypairs must first be set up on the server first before you can log in with them. Please ask fristi for help if

Then, you may log in with the following command:

```
ssh your_username@subcon.town -p 420 -i /path/to/your/privatekey
```

To make logging in easier in the future, you may also set up a config file with these configurations in:

```
~/.ssh/config
```

An in-depth explanation of this file can be found here: https://linuxhandbook.com/ssh-config-file/. This way, you won't need to specify the extra options when running the ssh client. This config file will also work exactly the same way on MacOS.

When logging in, if you private key requires a password, you'll be asked to enter it now. After that, you'll be logged in and ready to go.

Revision #1
Created 13 September 2025 18:13:28 by Fristi
Updated 13 September 2025 19:05:11 by Fristi