# Prerequisites to the Tenant-in-a-day-training

## General information

This guide is ensure the training proceeds as desired, with all participants having fullfilled all basic requirements. Steps everybody needs to perform:

1. You will generate your own unique keypair.
2. You will send your public key to your trainer.
3. Your public key will be given to your VM, which will then be able to recognize and authenticate your SSH session.
4. You will download UMP, to be able to interact with the DSH.

Optionally, if you don't already have it, you may need to set up a way to use SSH.

If you are already familiar with SSH and creating a keypair, you can skip straight to your preferred version of 'connecting'.

## Options to connect over SSH

- Linux and Mac have an SSH client built in. Proceed to Generating a keypair in Linux/Mac.
- Windows sometimes comes with ssh. To check, you open Powershell (WIN+X, A, click `yes`) and type `ssh`. You will either get feedback stating:
  - you didn't supply required parameters (meaning you have it, and should proceed to generating a keypair in PowerShell.
  - ssh is unknown, meaning you don't have it, so you should continue to the next step.
- If Windows doesn't have ssh, you have several options:
  - Windows Subsystem for Linux (WSL) allows you to run Linux commands (including SSH) on your Windows system.
  - Git for windows allows you to run git commands, through a bash shell. There is even a portable version that does not require admin rights.
  - Many more options. You can use whatever you're comfortable with. *(note that the format in which PuttyGen stores the keys is not standard; you will need to copy the generated key in a new text file)*

WSL has our preference, with Git for Windows acting as a backup-option. Guide for both options have been included.

## Getting WSL

1. Open PowerShell (WIN+X, A), and run the following command:

```
Enable-WindowsOptionalFeature -Online -Featureame Microsoft-Windows-Subsystem-Linux
```

2. Reboot when prompted.

3. After rebooting, open the Windows Store, and search for `Ubuntu`.
4. Click `install` or `download`, and wait for it to complete.
5. When it's done, there should be an icon for Ubuntu. Click it, set (and remember!) your password.

## Generating a keypair in WSL

6. In your Ubuntu terminal, type

```
sudo apt install ssh
ssh-keygen -t rsa
```

7. Follow the instructions. Note that you will have to set a password.
8. Open the `run` dialog in Windows (WIN+R), and enter `%LocalAppData%\Packages\`
9. Find the folder that has `Ubuntu` in its name. For example `CanonicalGroupLimited.Ubuntu18.04onWindows_79rhkp1fndgsc`
10. Drill down to `\LocalState\rootfs\home\`
11. Open the folder corresponding to your Ubuntu username, and open the .ssh folder.
12. Mail the public key file to the trainer.

## Using WSL

After you've mailed your public key to the trainer, you will receive the IP of your VM. So:

- Your `ssh key` is in `.ssh`, and is called `id_iot`.
- The IP you got is 52.59.203.96. You will then run:

```
ssh -i .ssh/id_iot ubuntu@52.59.203.96
```

Where the -i flag stands for `identity`, and `ubuntu` is the default username.

## Getting Git Bash

1. Download Git for Windows.
   - If you have admin rights, use the installer.
   - If you do *not* have admin rights, use the portable version.
2. Install or unpack Git for Windows.
3. Start Git Bash
   - If Git for Windows was installed, it will create an entry in the context menu, allowing you to open a Git Bash in whatever folder you prefer.
   - If the portable version is used, you will need to start the bash from the executable found in the unpacked folder.
4. If needed, navigate to a preferred folder. (use `pwd` to figure out where you are, use `cd` to navigate)

## Generating a keypair in Git Bash

1. In Git Bash, type `ssh-keygen -o`, and follow the instructions.

    - Note that you will have to set a password.
    - If the .ssh folder does not exist, it will give an error. You can solve this by navigating to `%userprofile%` and running `mkdir .ssh`.
2. Locate the keypair, which by default ends up in `%userprofile%/.ssh`.
3. Mail the public key file to the trainer.

## Using Git Bash

After you've mailed your public key to the trainer, you will receive the IP of your VM. So:

- Your `ssh key` is in `.ssh`, and is called `id_iot`.
- The IP you got is 52.59.203.96. You will then navigate to the folder where your private key exists, and run:

```
ssh -i id_iot ubuntu@52.59.203.96
```

Where the -i flag stands for `identity`, and `ubuntu` is the default username.

## Generating a keypair in Linux/Mac

1. In your terminal, type

```
ssh-keygen -t rsa
```

2. Follow the instructions. Note that you will have to set a password.
3. It will state in which folder your keypair will have been saved. Use a file explorer to navigate to this folder.
4. Mail the public key file to the trainer.

The rest of the instructions are the same as those under Using WSL

## Generating a keypair in Windows PowerShell

1. In PowerShell, type the following commands (press ENTER after each)

```
mkdir %userprofile%/.ssh
cd %userprofile%/.ssh
ssh-keygen -t rsa -C "your_email@example.com"
```

2. Follow the instructions. Note that you will have to set a password.
3. It will save the key in the folder %userprofile%/.ssh. Use a file explorer to navigate to this folder.
4. Mail the public key file to the trainer.

The rest of the instructions are the same as those under Using WSL

## Getting UMP

Use one of the following links to download UMP, and install it.

Linux

Mac

Windows

You will be guided through the usage of UMP during the training.

## Glossary

- VM: Virtual Machine. The trainer made one for everyone. These will be identified based on their IP.
- SSH: secure shell. A way to interact with remote systems (such as our VM).
- Authentication: the VM only allows SSH connections from systems it knows. So, you'll need to authenticate.
- A keypair authenticates you as a user. These files should start with `id_` followed by your name (`id_jan`), and consists of two parts:
    - A private key. You do *NOT* share this, ever.
    - A public key, which you can share. This file tends to end in `.pub`