

---

## Introductory Lab

## Introduction

This lab introduces you to the tools you will use throughout the course. Therefore, it is important that you read carefully the instructions bellow. This introductory lab consists of two parts:

1. **Server.** This part will introduce you to the remote server, how to connect to your server, and how to run JupyterLab on the server. You will also learn to use SSH/SCP and clone the course repository.
2. **Introduction to Linux.** This part will introduce you to the basic concepts of Linux and the command line interface. You will learn to navigate the file system and use the essential commands.

## Milestone 1 — Server

In this course, you will use a server, which is a Virtual Machine (VM) running a Linux distribution called Ubuntu. You will use a server hosted on NTNU's server infrastructure. These servers are available 24/7 and can be accessed via NTNU's internal network. If you are not on campus, you can connect to the server using the [NTNU VPN](#). To connect to the server, you will be using PCs in [Sahara lab](#).

### 1.1 Connecting to the Server

- Before accessing the server, you need to open the Linux terminal in the **Sahara PC**. There are various ways to open the terminal. For example, you can use the keyboard and press the `Ctrl+Alt+T`, or navigate to the menu option at the bottom of the screen, click the application meny, find the terminal icon and launch it.

*Note: The keyboard shortcut to copy text within a Linux terminal is `Ctrl+Shift+C`. You can use `Ctrl+Shift+X` for cutting, and `Ctrl+Shift+V` for pasting.*

- To access the remote server securely, we will use SSH (Secure Shell) protocol.
- You can access the server through SSH by running the following command in the terminal:

```
ssh <your_NTNU_username>@<your_server_IP_address> .
```

*Replace `<your_NTNU_username>` with your NTNU username, and `<your_server_IP_address>` with your server IP address that can be found in "Group Description on the Blackboard". In all the following commands, you need to replace what is inside `<...>` with your own values. The password is the same as your NTNU password. Note that every team member can access the team's server with their own NTNU credentials. Note that when entering password in the Linux terminal, you cannot see what you are typing.*

- You can exit the server by typing the following command in the terminal: `exit` .

*The connection to the server is closed.*

### SSH Key

Instead of using the password every time you connect to the server, you can use an SSH key:

- Generate an SSH key **on Sahara PC**:

```
ssh-keygen -t ed25519 -f ~/.ssh/<key_name> -C "<your_email>"
```

*Replace `<key_name>` with a name for the key, and `<your_email>` with your email address.*

*You will be asked to enter a passphrase to protect the key. Leave it empty for now. This will generate two files: `~/.ssh/<key_name>` and `~/.ssh/<key_name>.pub`. The first file is the private key, and the second file is the public key.*

- Copy the public key from Sahara PC to the server. Then, connect to the server and add the key to the authorized keys file:

```
# On the Sahara PC
scp ~/.ssh/<key_name>.pub <your_NTNU_username>@<your_server_IP_address>:~/.ssh/

# Now, connect to the server, and do the following:
cat ~/.ssh/<key_name>.pub | sudo tee -a ~/.ssh/authorized_keys
```

*Alternatively, you can use the following single command on Sahara PC:*

```
cat ~/.ssh/<key_name>.pub | ssh <your_NTNU_username>@<your_server_IP_address> \
'sudo tee -a ~/.ssh/authorized_keys'
```

- Exit the server. Then, from **Sahara PC**, connect to the server using the private key:

```
ssh -i ~/.ssh/<key_name> <your_NTNU_username>@<your_server_IP_address>
```

### SSH Config

Instead of typing the username and the IP address every time you connect to the server, you can add the following lines to `~/.ssh/config` file on **Sahara PC**:

```
Host ntnu_server
  HostName <your_server_IP_address>
  User <your_NTNU_username>
  IdentityFile ~/.ssh/<key_name>
```

*If you have not used SSH before, you may not have a `~/.ssh/config` file. You can create it by running `touch ~/.ssh/config`. To edit the file, you can use `nano ~/.ssh/config` command. To exit and save changes, press 'Ctrl + X' and type 'yes'.*

Then, you can connect to the server by running `ssh ntnu_server`.

*Note: You can follow these instructions and use your computer to connect to the server. If you have Windows OS, use Sahara PC. Nevertheless, we recommend getting familiar with the instructions within this document, as the labs in this course are designed for Sahara PCs.*

### Copying Files to/from the Server

You can use `scp` to copy files and directories to/from the server. For example, you can run the following commands on **Sahara PC**:

- To copy a file from Sahara PC to the server:

```
scp <local_file> ntnu_server:~/<destination_directory> .
```

- To copy file from the server:

```
scp ntnu_server:~/<source_file> <local_directory> .
```

- To copy directories add the `-r` flag:

```
scp -r <local_directory> ntnu_server:~/<destination_directory> .
```

**Task:** show your TA that you can connect to the server using `ssh ntnu_server`

**Task:** show your TA that you can copy a file to/from the server using `scp`

**Task:** show your TA that you can copy a directory to/from the server using `scp`

## 1.2 Running JupyterLab on the Server

[JupyterLab](#) is a web-based interactive development environment for Jupyter notebooks, code, and data. You can run JupyterLab on the server and access it from Sahara PC. To do so, you need to establish an [SSH tunnel](#) from Sahara PC to the server.

- On the **Sahara PC**, run the following command:

```
ssh ntnu_server -L 8888:localhost:8888
```

*This will establish an SSH tunnel from port 8888 in Sahara PC to port 8888 in the server. The first port number is the port in Sahara PC, and the second port is the port in the server. You can use any port number above 1024 and below 65535. You can also use different port*

*numbers for the two ports, but using the same port number is easier.*

- On the **server**, start JupyterLab:

```
jupyter lab --no-browser --port=8888
```

*The port number should be the same as the second port number in the SSH tunnel.*

- On Sahara PC's browser, type `localhost:8888` to show JupyterLab. You can now use JupyterLab as you would do on Sahara PC, but it is actually running on the server. If needed, copy the token value from the terminal to the browser in the field `Password or token:`

*The port number should be the same as the first port number in the SSH tunnel. If JupyterLab asks you for a token, you can find it in the terminal where you started JupyterLab in the server.*

## 1.4 Course Repository

We will use a [Git](#) repository to distribute the course material throughout the semester.

- Clone the course repository to your **server** using the following command:

```
git clone https://github.com/ntnuttm4200/2025_labs.git /home/ttm4200/labs
```

This will create a folder named “labs” in your home directory. You can find the course material in this folder.

- We will update the repository with new assignments. **You need to run these commands at the beginning of each lab.** To get the latest version of the course material, run the following commands:

```
cd ~/labs
git add .
git commit -m "Commit message"
git pull origin main --no-edit
```

- You might need to set your username and email for git, when committing your changes for the first time. To do this, use the following commands:

```
git config --global user.name "Firstname Lastname"
git config --global user.email "Your email"
```

## Milestone 2 — Introduction to Linux

This is the last step that you have to follow on this PDF.

- Run JupyterLab on the server and open the notebook `~/labs/lab0/tasks_lab0.ipynb`. Follow the instructions in the notebook to complete the tasks. We will use JupyterLab for the rest of the course.