README

Nada Dandachi

June 17, 2024

Contents

1 Installation

1.1 Prerequisites

Install **VirtualBox**, an open-source virtual machine emulator available online: `https://www.virtualbox.org/wiki/Downloads`

1.2 Create the virtual machines (VMs)

In the terminal, launch the `manual_install.sh` file in the root of the `package_wehe` directory.

```
sh wehe_demonstrator/script/manual_install.sh
```

If you encounter the following error:

```
vboxdrv kernel module is not loaded
```

Execute the following commands in the terminal:

```
uname -r
yum install kernel-devel-<uname -r>
```

1.3 Install Ubuntu on the VMs

Start VirtualBox, then launch each of the virtual machines by double clicking on it. Follow the instructions to install Ubuntu (Figure 1), this operation takes some time (~30 minutes per machine).

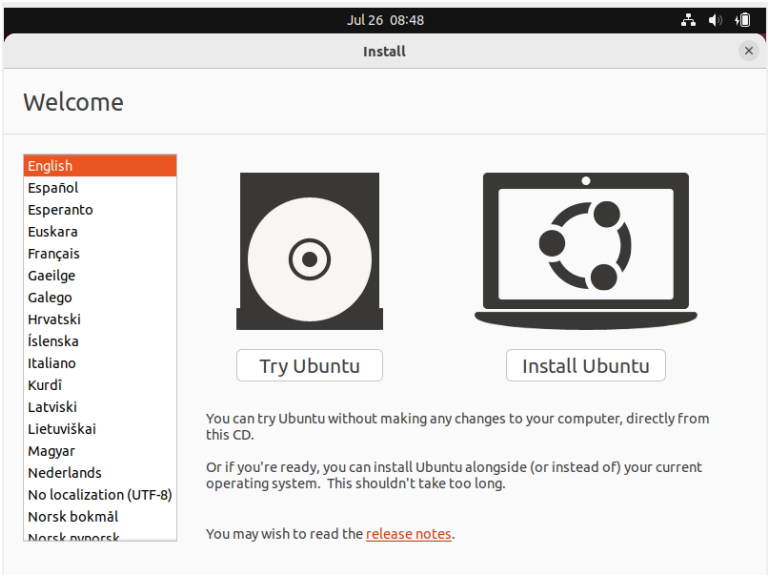


Figure 1: Install page

Name the virtual machines server, client and middlebox.

- The username for the server VM will be "server" and the password will be "Server" (Figure 2).
- The username for the client VM will be "client" and the password will be "Client".
- The username for the middlebox VM will be "middlebox" and the password will be "Middlebox".

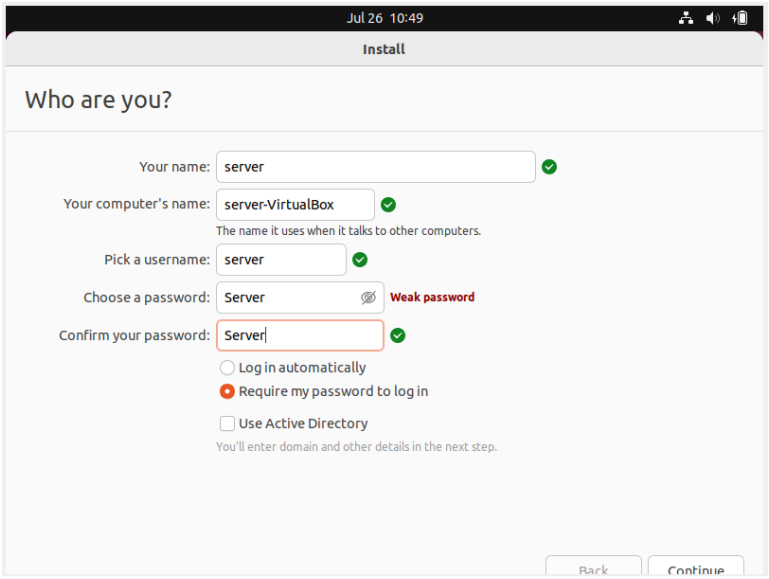


Figure 2: Filled-in login page

1.4 Provide secure remote access to the VMs

In the terminal of each VM, install `openssh-server` by executing the following command:

```
sudo apt-get install openssh-server
```

Then, go to:

```
cd /etc/ssh
sudo chmod 777 sshd_config
nano sshd_config
```

In `sshd_config`, unmask the line:

```
#PasswordAuthentication yes
```

by changing it to:

```
PasswordAuthentication yes
```

Type C-o to save, then C-x to leave.

In the home directory of your host (your computer), generate a public key:

```
ssh-keygen -t rsa -b 4096
```

And send it to the VMs:

```
ssh-copy-id -p 3333 server@localhost
ssh-copy-id -p 2222 middlebox@localhost
ssh-copy-id -p 1111 client@localhost
```

In the terminal of each VM, disable the password:

```
sudo passwd -d <username>
```

1.5 Allow shared folders

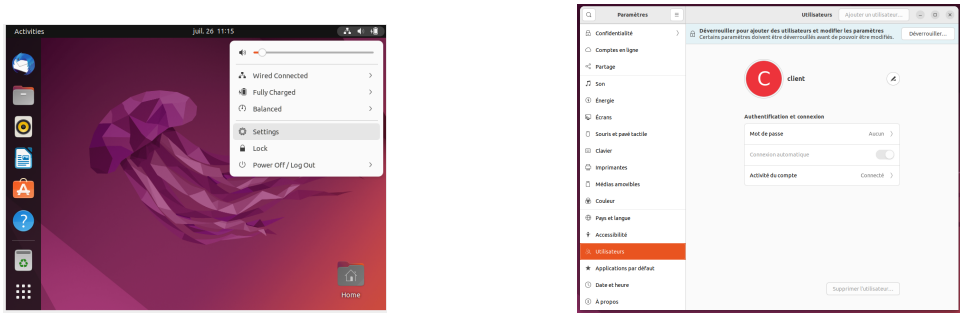
On the VMs, run:

```
sudo apt-get install virtualbox-guest-x11
sudo usermod -aG vboxsf <username>
```

1.6 Enhance graphical performance and integration of the VMs

For each VM:

- In Users, enable automatic login (Figure 3).
- Adjust the screen size according to your display, particularly for the client VM. (Figure 4).



(a) Settings tab

(b) Users tab

Figure 3: Enabling automatic login

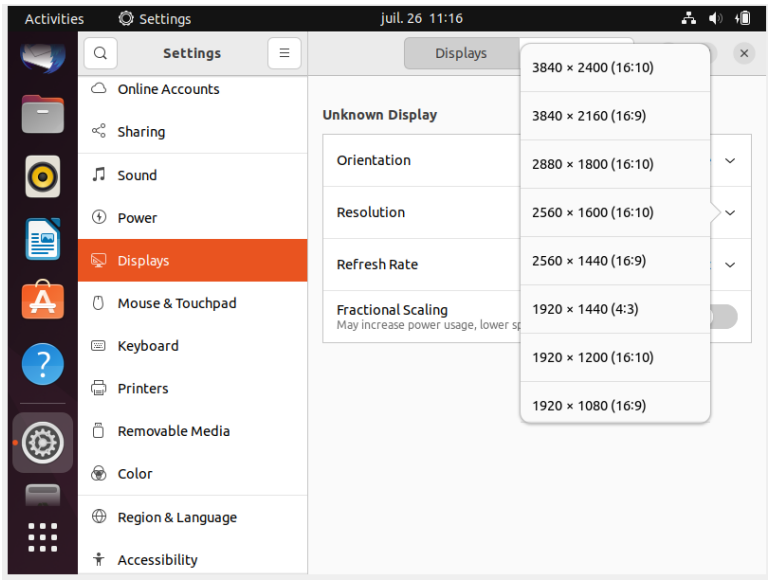


Figure 4: Resolution tab

1.7 Upload the video files in the server VM

On the host, in the working directory of your wehe package, run:

```
scp -P 3333 -r dash server@localhost:~
```

On the server VM, move the file to the location `/var/www`:

```
cd /home/server
sudo mkdir /var/www
mv dash /var/www
```

1.8 Finish the installation

On the host, run:

```
sh wehe_demonstrator/script/manual_install2.sh
```

On each VM, run:

```
cd /home/<username>
sh manual_install_<username>.sh
```

Say yes to all and allow wireshark for non superusers. If for the server VM, you are asked to fill DNs, fill with blank values.

2 Use

2.1 Launch the demonstrator

On host, run:

```
python3 wehe_demonstrator/src/hmi.py
```

It takes approximately 2 minutes to start.

2.2 Standard use

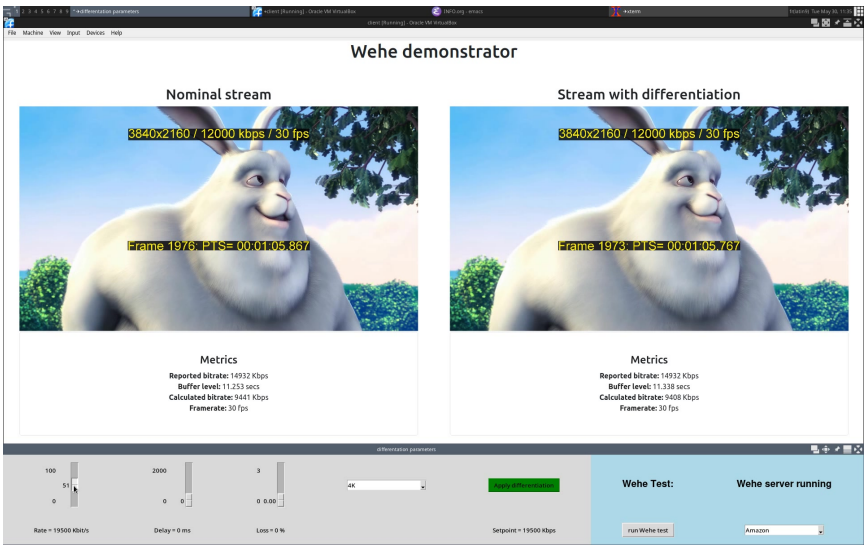


Figure 5: Screenshot of the application

In the central combobox (Figure 5):

- Select the nominal resolution.
- Select the desired differentiation with the scales for `Rate`, `Delay` and `Loss`.
- Click on the `Apply differentiation` button.
- Click on the `run Wehe test` button.

3 Check the maximum bandwidth between the server and the client

3.1 Server side

Set up a listener:

```
nc -l -p <port> < /dev/zero
```

Run an IP traffic monitor:

```
sudo apt-get install iptraf
sudo iptraf
```

3.2 Client side

Communicate with the server:

```
telnet <server ip_adress> <port>
```