SWitCH – Sistemas de Computação e Redes (SCOMRED)
2024/2025
Luís Nogueira (lmn@isep.ipp.pt)
Laboratory 1

Install Linux on UTM

1 – Introduction

This document describes the needed steps for installing Ubuntu on UTM, **if you have a MacBook with Apple Silicon.**

Most of the needed information is available on the main document for Lab 1. Please refer to that document.

2 – Download Ubuntu

If you have a MacBook with Apple Silicon, you need to download the ARM version, available at https://ubuntu.com/download/server/arm. Download the latest version of Ubuntu Server LTS, which is a 64-bit operating system. You will get a .iso file which is an image of the installation optical disk (i.e., a CD ROM). Remember where you stored the file.

3 – Install and configure UTM

UTM is the easiest way to run Ubuntu on your Apple Silicon Mac. It is a freely available open-source program that basically functions as a graphical user interface for QEMU, an extremely capable cross platform emulator and virtualization application.

UTM is available at https://mac.getutm.app/. Click on the **Download** button. Once you have downloaded the .dmg file, open it and move UTM into your applications directory. You can now launch UTM, which will take you directly to the main application window.

In order to add a new virtual machine in UTM, click the '+' button at the top of the main window (Figure 1).

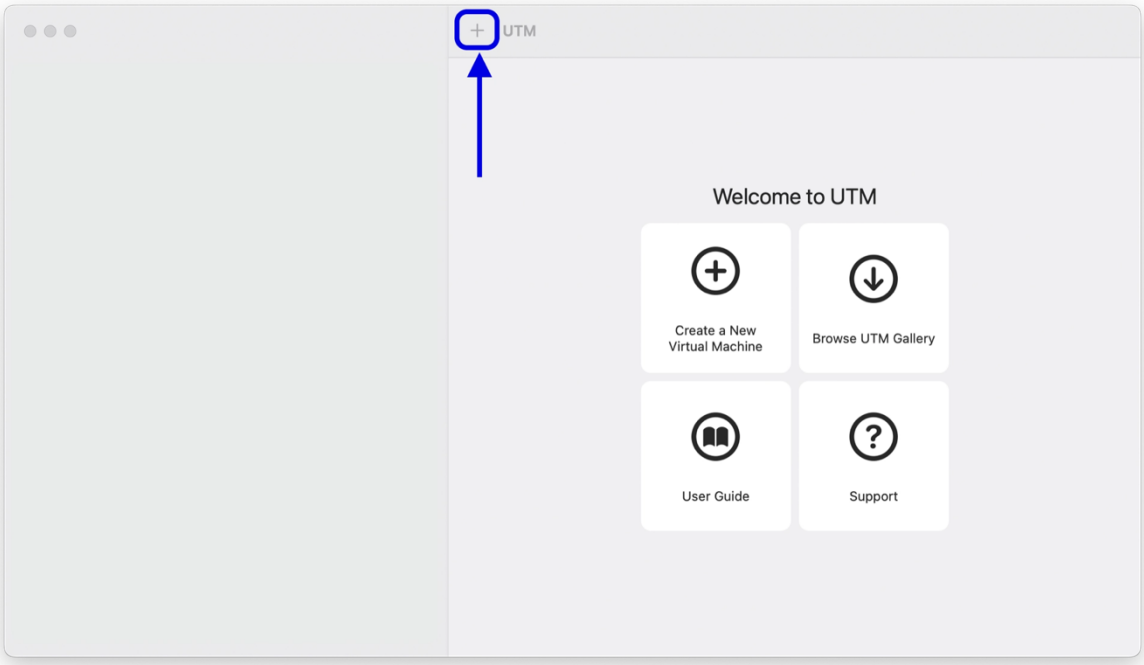

Figure 1. UTM main screen.

On the first screen you will be given a choice of whether to use virtualization or emulation. **Virtualization is always the best option**, as the performance is vastly improved over emulation. It is only necessary to use emulation when the guest OS or guest OS programs that you are trying to emulate are not compatible with your computer's architecture.

As Ubuntu is available for ARM architecture, it is possible to virtualize it on Apple Silicon. The only exception would be if you need to run software in Ubuntu that is only compatible with x86 (Intel compatible) architecture.

Select the **virtualization** option by clicking the large button (Figure 2).

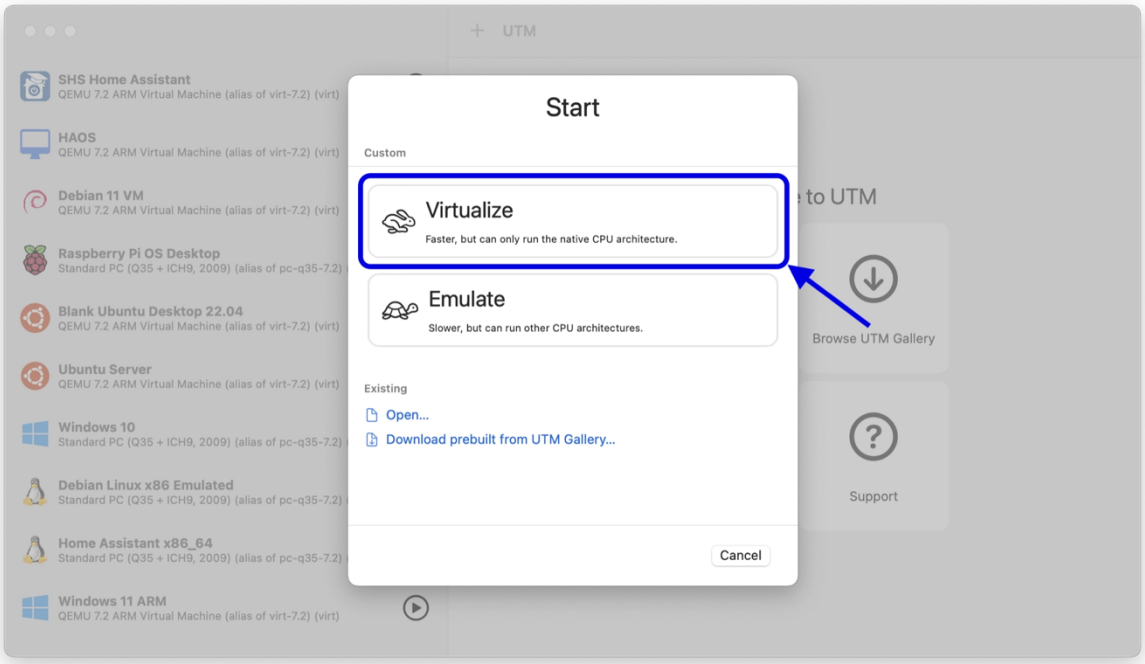

Figure 2. Create a virtualized VM.

On the next screen we are given the option to choose the type of operating system, which can simplify the configuration. Therefore go ahead and choose **Linux** for the operating system (Figure 3).

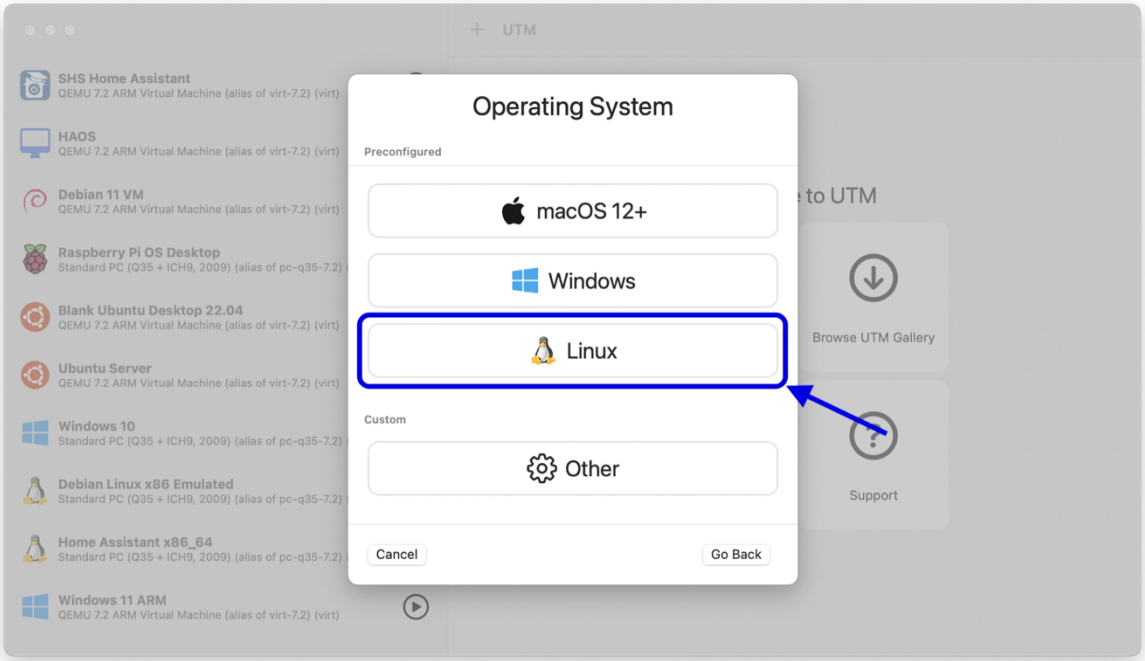

Figure 3. Create a Linux VM.

Next we need to supply UTM with a disk image that contains the Ubuntu installer.

We need to point our VM to this disk image. Under the **boot ISO image** section, (1) click **browse** and select the .iso file that you just downloaded. Once selected, (2) you will see the path to the .iso file. (3) Click **continue** to move to the next step (Figure 4).

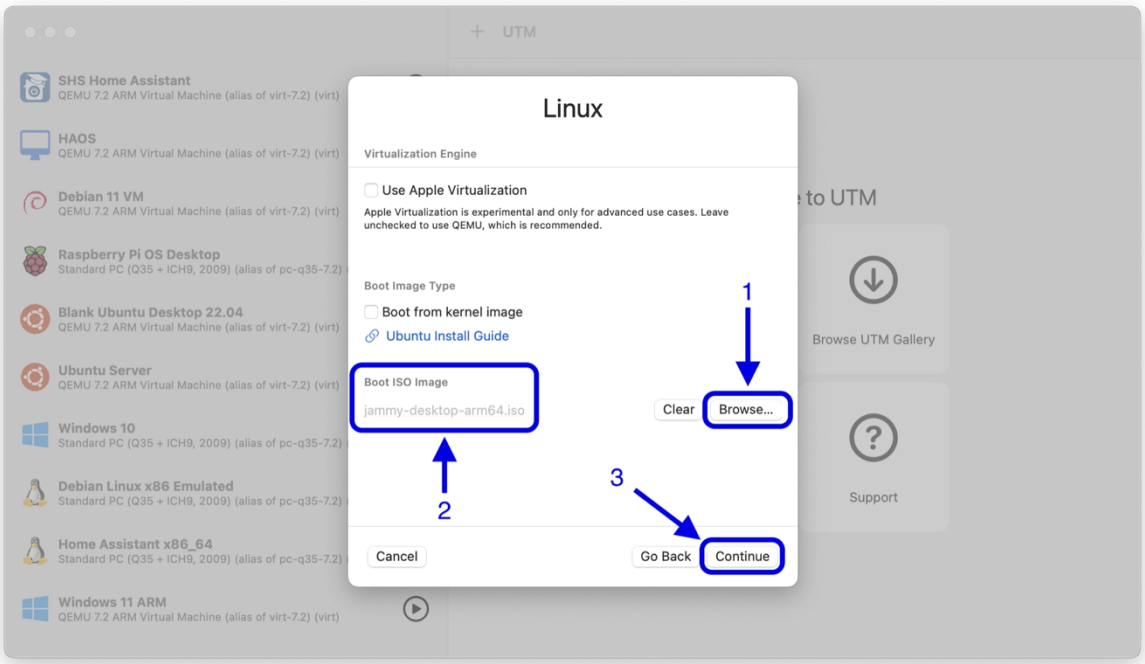

Figure 4. Select the .iso image.

Next, we can choose the hardware settings for our VM. For choosing the amount of RAM, a good rule of thumb is to choose somewhere between the guest OS minimum requirement and half of the host OS system's amount of RAM.

On the next screen you can specify the size of the virtual hard disk. This is the size of disk that will be 'seen' by Ubuntu. It is worth noting that although Ubuntu will see a disk of the size you choose here, the space will not be used on your host system's physical disk until you actually use the space on the guest OS.

If you wish to share files between your guest and host operating systems, you can choose a shared folder. So long as you install the guest additions (more on that later), you will be able to access this folder from both the guest and host operating systems, allowing you to easily transfer files. Click **browse** and navigate to a folder that you want to share, then click **open** to select it. Once selected, you will see the path appear under the shared directory path.

On the final screen you have the opportunity to give your new VM a name. You can review the settings and then click **save** to add your new virtual machine to UTM.

Note that the default display and network settings should be adequate to get things up and running. You can adjust these at a later point if necessary.

Select your new VM from the left-hand sidebar menu and then click the large play button to boot up the virtual machine (Figure 5).

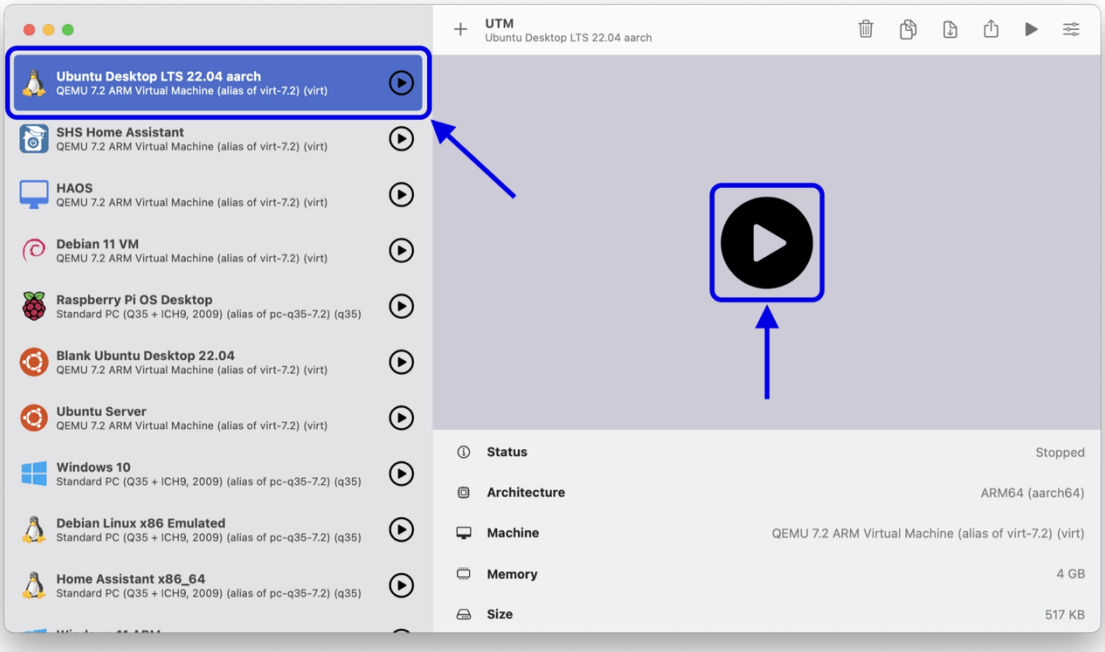

Figure 5. Start your VM.

Once you boot your new VM, you will be taken to GRUB, where you can choose to begin the installation. Select **Try or Install Ubuntu** and then press enter.

You must now refer to the main Lab 1 document available in Moodle and follow Section 5.

At the end of the installation process, **do not click "Reboot Now" yet**.

We need to remove the installation media. This can be done from the UTM main window (Figure 6). With the VM selected in the left-hand sidebar, click the **CD/DVD** dropdown menu and then click **clear**. This will remove the Ubuntu installation .iso so that the VM boots with the new installation, rather than booting the installer again.

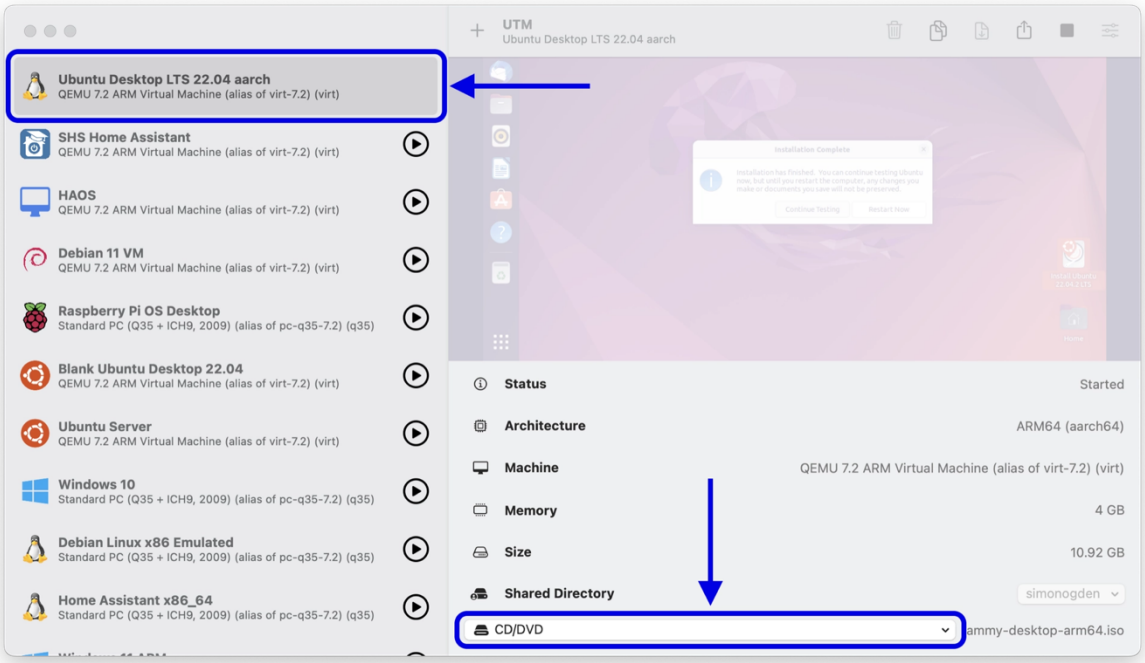

Figure 6. Remove the installation media.

4 – Install guest additions

In most cases the guest additions are already installed and you do not need to complete this step. However, in some cases, you may need to install or update them and can, therefore, follow the next step.

In the Linux command line, enter the following command and press enter:

```
$ sudo apt update
$ sudo apt install spice-vdagent
```

5 – Create and mount the shared folder

A shared folder is a directory on your host operating system (i.e., macOS) that is also accessible by the guest operating system (i.e., Ubuntu). A shared folder is the easiest way to transfer files between the host and the guest.

Create a directory in your host operating system that you wish to be shared, such as **~/shared** in macOS.

Ensure that your Linux VM is turned off. From the UTM main window (Figure 6), with the VM selected in the left-hand sidebar, select the shared directory with the “Browse…” button.

Start your Linux VM and login.

You can mount the share with the following sequence of commands:

```
$ cd /media/
$ sudo mkdir sf_shared
$ sudo mount -t 9p -o trans=virtio share /media/sf_shared -oversion=9p2000.L
```

To test the shared folder, put some files into the shared directory. Then use the **ls** command in the Ubuntu terminal to see the files listed:

```
$ ls /media/sf_shared
```

You may notice that accessing the mount point fails with “access denied” unless you’re the root user. This is because by default the directory inherits the UID/GID from macOS which has a different numbering scheme. You can fix the error with the following command:

```
$ sudo chown -R $USER /media/sf_shared
```

This will not change the permissions on your host system but will store the guest ownership in a file attribute.

To make this mount permanent so that it will be in effect the next time you start Ubuntu, execute the following commands:

```
$ sudo su
# cp /etc/fstab /etc/fstab-save
# echo “share /media/sf_shared 9p trans=virtio,version=9p2000.L,rw,_netdev,nofail 0 0” >> /etc/fstab
# exit
```

Be especially careful typing the **echo** command, particularly the double **>>** symbol. If you accidently trash the system file `/etc/fstab`, you can recover it from the copy `/etc/fstab-save`.