Software Development Practice 1

Instructor: RSP <rawat.s@eng.kmutnb.ac.th>

Exploring Linux with QEMU

Objective

- Learn how to use QEMU on Windows machine to emulate Linux images such as Tiny Core, Alpine Linux and Raspberry Pi OS.
- Learn how to install Alpine Linux to a virtual disk.
- Learn how to configure the SSH server and access it via an SSH client.

Expected Learning Outcomes

- Students will be able to install and configure QEMU.
- Students will be able to test different Linux distribution using QEMU.

Task 1: Boot Tiny Core Linux with QEMU on Windows (64-bit)

1. Download the **QEMU installation file** from:https://qemu.weilnetz.de/w64/
   - Use the latest version (e.g. `qemu-w64-setup-20250422.exe`).
   - Run the installer and follow the on-screen instructions.
   - After installation, add the QEMU installation path to the Windows System Environment Variable `PATH`.

     Example: If QEMU is installed in `C:\Program Files\qemu`, add that directory to `PATH`.

2. Download **Tiny Core iso** image file.Open Windows PowerShell and using the Invoke-WebRequest command to download the ISO file.

   ```
   > Invoke-WebRequest <URL> -OutFile <File>
   ```

   File: http://tinycorelinux.net/16.x/x86_64/release/TinyCorePure64-current.iso

3. Create a **virtual hard disk image**. Open Windows PowerShell and run the following command:

   ```
   > qemu-img.exe create tinycore_x64.img 1G
   ```

4. Boot into Tiny Core from CDROM image:

```
> qemu-system-x86_64.exe -m 256 -nic user `
-boot d -cdrom TinyCorePure64-current.iso `
-drive format=raw,file=tinycore_x64.img `
-netdev user,id=mynet0,hostfwd=tcp::2222-:22 `
-device virtio-net-pci,netdev=mynet0
```

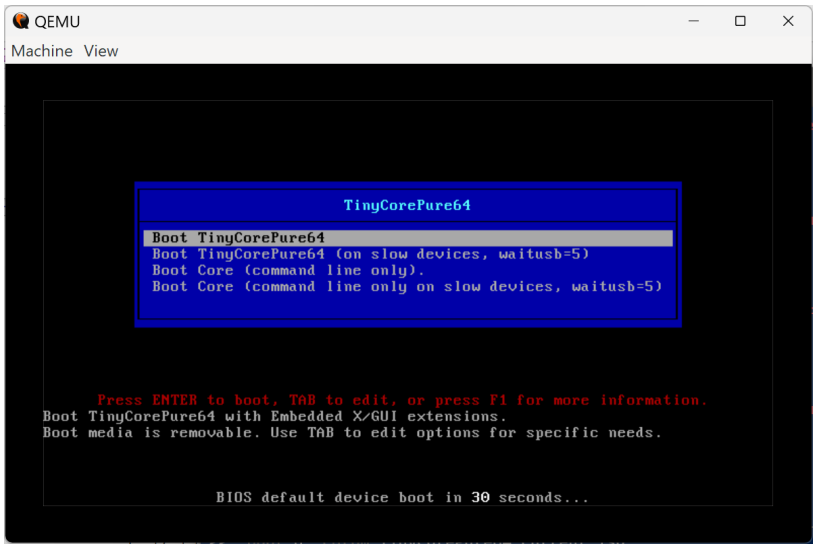
Note:

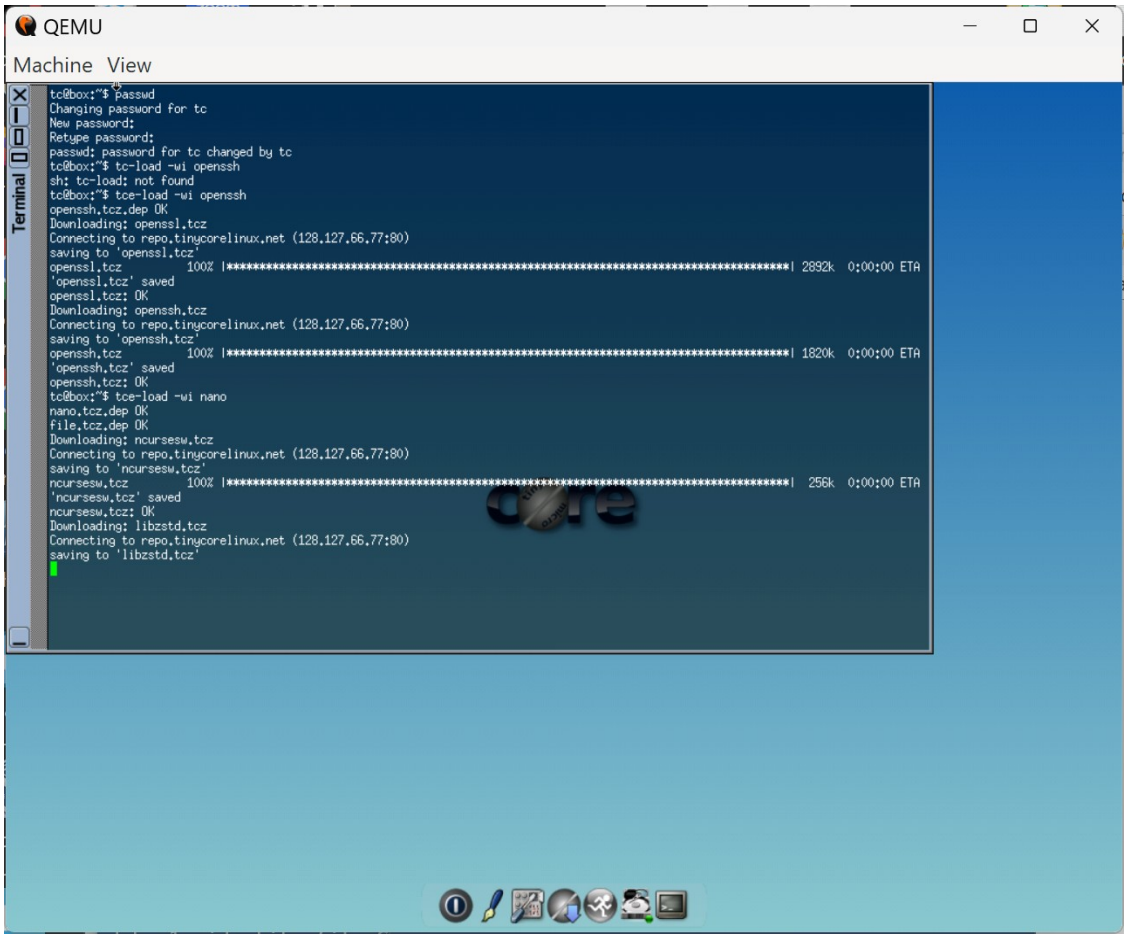
-m 256	Assigns 256MB of RAM to the VM.
-boot d	Boots from the ISO (CD-ROM).
-cdrom	Specifies the ISO image.
-drive	Attaches the virtual disk.
hostfwd=tcp::2222-:22	Forwards host port 2222 to guest SSH port 22.

Note that QEMU boots from the **ISO image**. When you boot from an ISO, Tiny Core typically runs entirely in RAM. It is **NOT running in a persistent mode** where changes are saved by default.

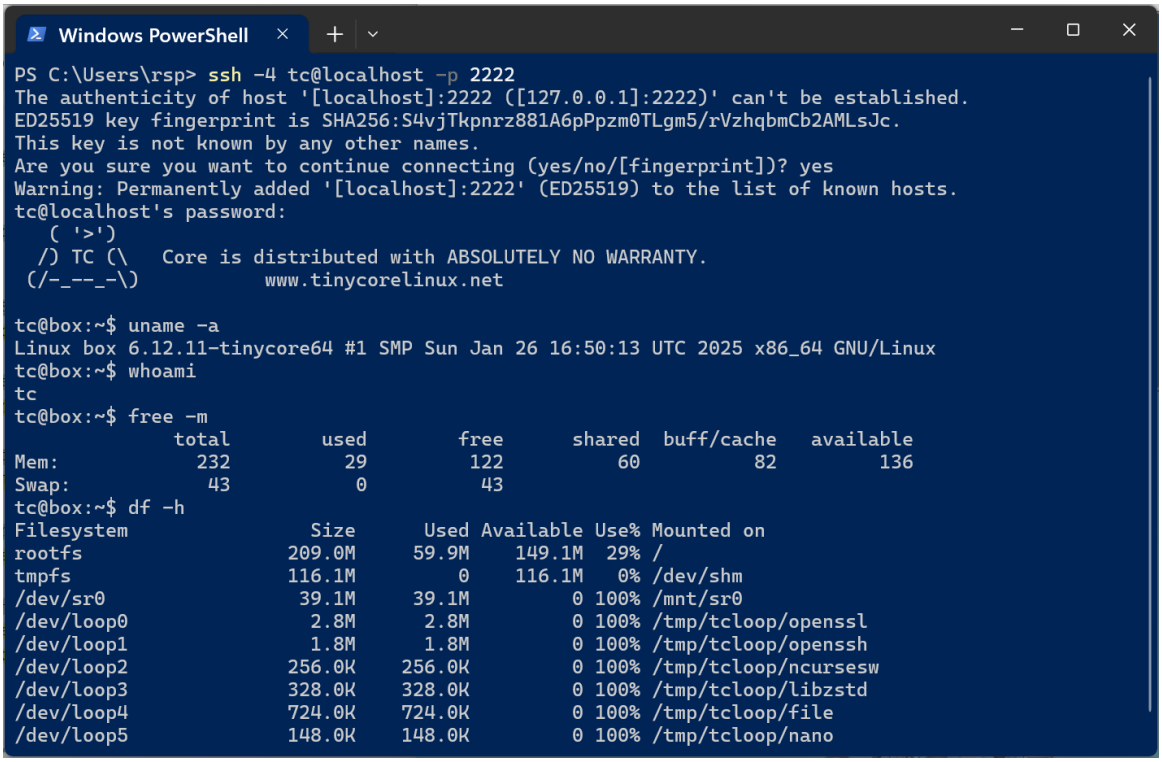
5. Run the following Linux command. Make sure that your Windows machine can connect to the Internet.

```
# Show kernel and system info.
$ uname -a

# Show the current user name.
$ whoami

# Set the password for the current user (tc).
$ passwd

# Install OpenSSH server.
$ tce-load -wi openssh

# Install nano editor.
$ tce-load -wi nano

# Create and edit SSH server configuration file.
$ sudo cp /usr/local/etc/ssh/sshd_config.orig /usr/local/etc/ssh/sshd_config

# Start the OpenSSH serer.
$ sudo /usr/local/etc/init.d/openssh start

# Verify the SSH server is running
$ sudo /usr/local/etc/init.d/openssh status
$ ps | grep sshd
```





TinyCore GUI Main Window



Using SSH client in Windows PowerShell to access the TinyCore Linux on QEMU.

Questions

- How to use **SSH client** (e.g. the scp command) to copy a text file from Windows drive to the home directory of the default user?
- Which command can be used to show the IP address of the network interface eth0 on Tiny Core machine?

Task 2: Boot Alpine Linux with QEMU on Windows (64-bit)

1. Download **Alpine Linux image file** (.iso) from:

   https://alpinelinux.org/downloads/.

   - Open Windows PowerShell and using the Invoke-WebRequest command to download the ISO file.

     ```
     > Invoke-WebRequest <URL> -OutFile <File>
     ```

   - `https://dl-cdn.alpinelinux.org/alpine/v3.22/releases/aarch64/alpine-standard-3.22.0-aarch64.iso -O alpine-standard-3.22.0-aarch64.iso`

2. Download the **UEF firmware file** compiled for QEMU 64-bit. It is used to emulate a UEFI BIOS environment for QEMU

   - `https://releases.linaro.org/components/kernel/uefi-linaro/16.02/release/qemu64/QEMU_EFI.fd -O QEMU_EFI.fd`

3. Create an **empty virtual hard drive image** using the **qcow2 format** (QEMU Copy-On-Write version 2).

   ```
   > C:\Tools\qemu\qemu-img.exe create -f qcow2 alpine.qcow2 1G
   ```

4. Boot Alpine in QEMU and install to virtual disk using the following command:

   ```
   > C:\Tools\qemu\qemu-system-aarch64.exe `
     -machine virt `
     -cpu cortex-a72 `
     -m 512 `
     -smp 2 `
     -bios QEMU_EFI.fd `
     -cdrom alpine-standard-3.22.0-aarch64.iso `
     -drive file=alpine.qcow2,format=qcow2 `
     -boot d `
     -netdev user,id=net0,hostfwd=tcp::2222-:22 `
     -device virtio-net-device,netdev=net0 `
     -nographic
   ```

5. After boot, login as root and then install and setup **Alpine-Linux** on the virtual disk:

   ```
   $ setup-alpine
   ```

   ```
   # Which disk(s) would you like to use? (or '?' for help) [none]
   # Enter: vdb
   # How would you like to use it?
   # Choose: sys (not data or lvm)
   ```

6. Run some basic Linux commands as follows.

```
$ uname -a
$ cat /etc/os-release | head -5
```

7. Run the following commmands to updata APK package list, install **openssh** and start the **SSH server**.

```
$ apk update
$ apk add openssh
$ rc-update add sshd
$ rc-service sshd start
```

8. Install the nano text editor:

```
$ apk add nano
```

9. Configure the SSH server and add the following lines to enable SSH root login and password-based authentication.

```
$ nano /etc/ssh/sshd_config

# Add the following to lines.
PermitRootLogin yes
PasswordAuthentication yes
```

To save changes to file in nano editor: Ctrl+O, Enter and Ctrl+X to exit

10. Restart the SSH server.

```
$ rc-service sshd restart
```

11. Turn off the Linux.

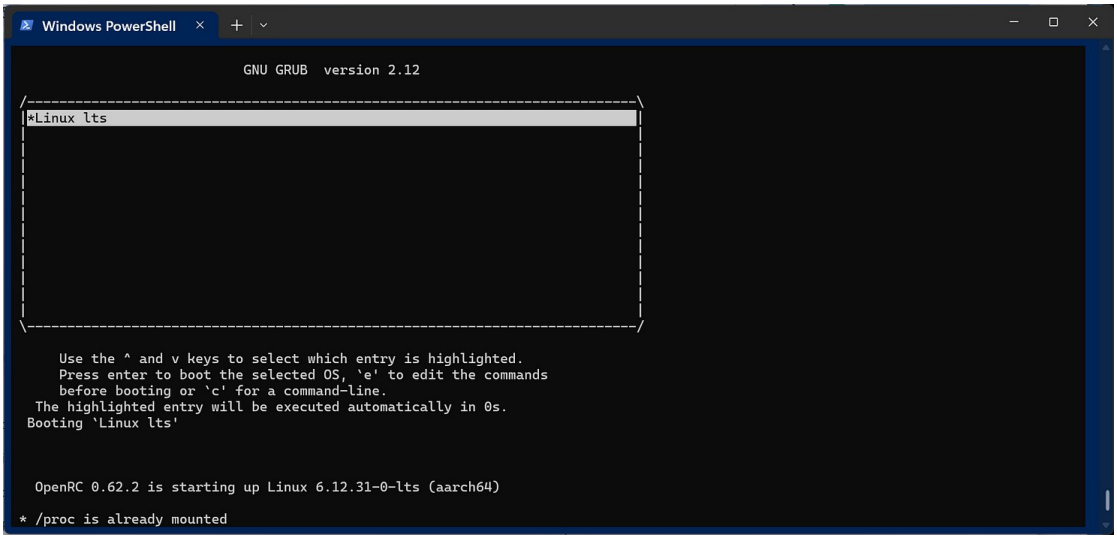
```
$ poweroff
```

12. Use QEMU to boot the Alpine Linux from disk again.

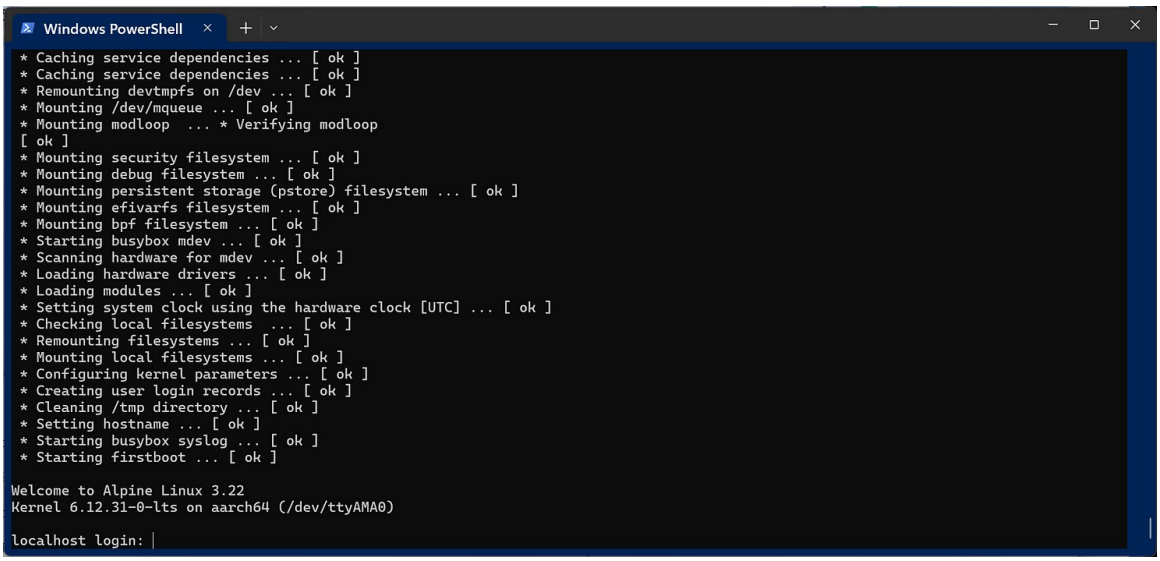
```
> C:\Tools\qemu\qemu-system-aarch64.exe `
  -machine virt `
  -cpu cortex-a72 `
  -m 512 `
  -smp 2 `
  -bios QEMU_EFI.fd `
  -drive file=alpine.qcow2,format=qcow2 `
  -netdev user,id=net0,hostfwd=tcp::2222-:22 `
  -device virtio-net-device,netdev=net0 `
  -nographic
```

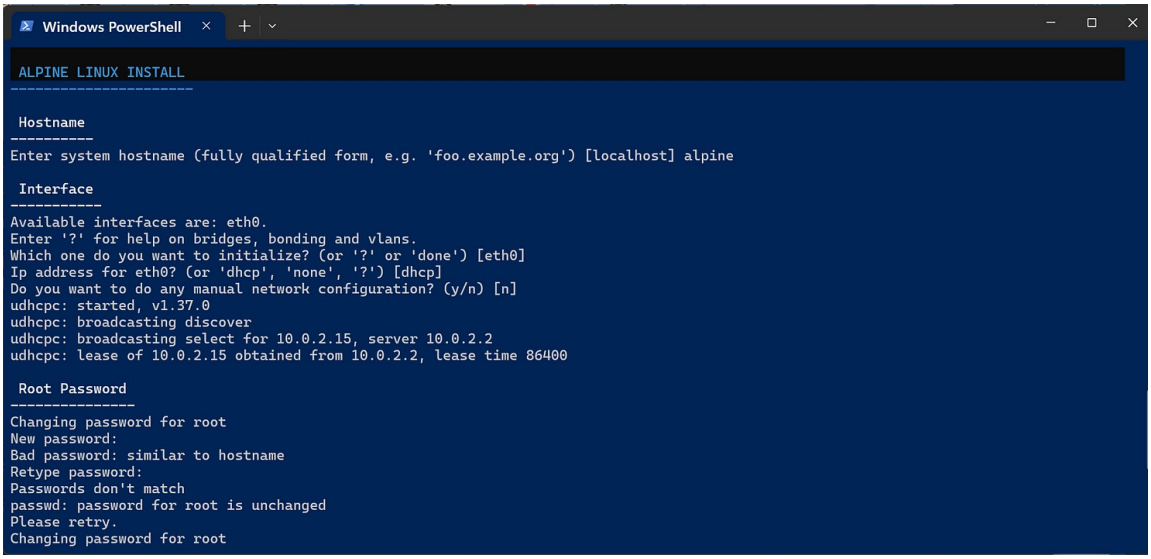
13. After alpine boot, use SSH client to connect. Connect to Alpine via SSH client (from Windows environment).

```
> ssh -4 root@localhost -p 2222 -v
```



Booting Alpine Linux on QEMU





Installing Alpine Linux on Virtual Disk

Task 3: Boot Raspberry Pi OS with QEMU on Windows (64-bit)

1. Download **Raspberry Pi OS image file for Raspberry Pi 3B (aarch64)** from:

   `https://downloads.raspberrypi.com/raspios_lite_arm64/images/`

   - File: `2025-05-13-raspios-bookworm-arm64-lite.img.xz`

2. Unzip the image .xz file.

3. Use **7-Zip File Manager** to extract the following files from the boot partition:

   - `bcm2710-rpi-3-b-plus.dtb`
   - `kernel8.img`

4. Resize the image file.

```
> qemu-img.exe resize 2025-05-13-raspios-bookworm-arm64-lite.img 4G
```

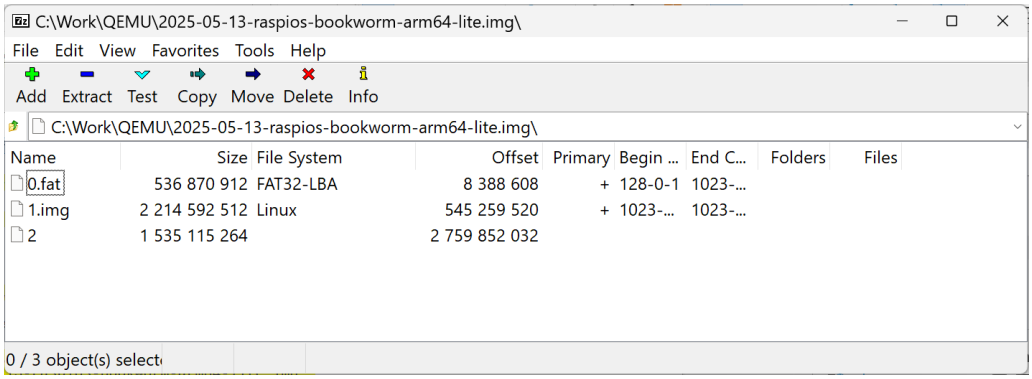
4. Convert the image file from .img to .qcow2

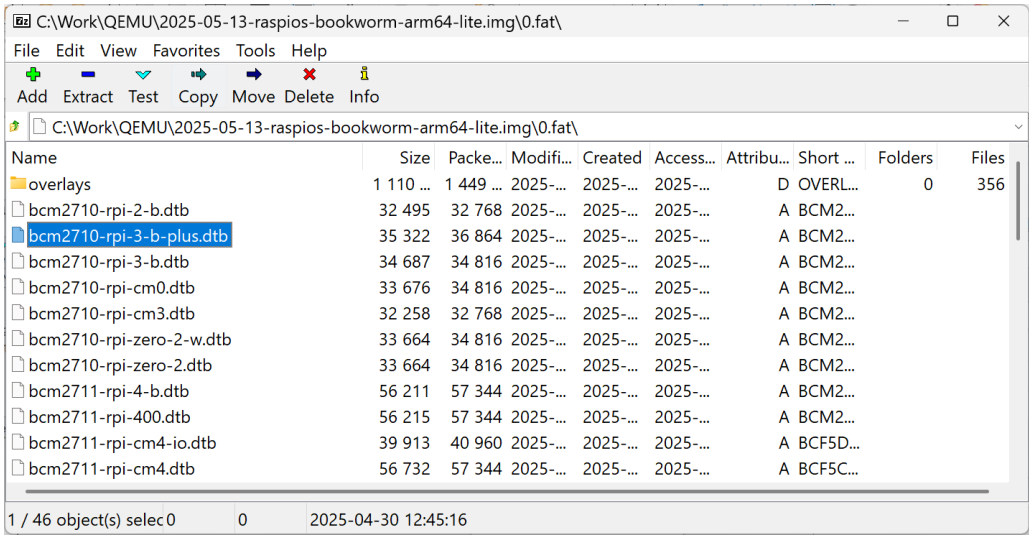
```
> qemu-img.exe convert -f raw -O qcow2 `
2025-05-13-raspios-bookworm-arm64-lite.img `
2025-05-13-raspios-bookworm-arm64-lite.qcow2
```

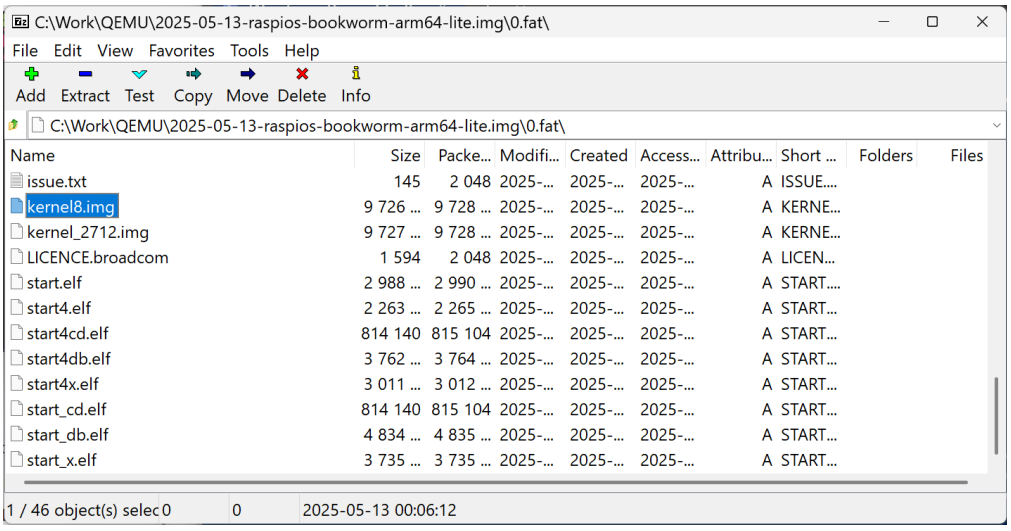
6. Boot the Raspberry Pi OS for Raspberry Pi 3B (64-bit).

```
> qemu-system-aarch64.exe `
  -machine raspi3b `
  -cpu cortex-a72 `
  -m 1024 `
  -smp 4 `
  -dtb bcm2710-rpi-3-b-plus.dtb `
  -kernel kernel8.img `
  -sd 2025-05-13-raspios-bookworm-arm64-lite.qcow2 `
  -append "console=ttyAMA0 root=/dev/mmcblk0p2 rw rootwait modules-
load=dwc2,g_ether rootfstype=ext4" `
  -usb `
  -device usb-net,netdev=net0 `
  -device usb-mouse `
  -device usb-kbd `
  -serial mon:stdio `
  -netdev user,id=net0,hostfwd=tcp::2222-:22
```

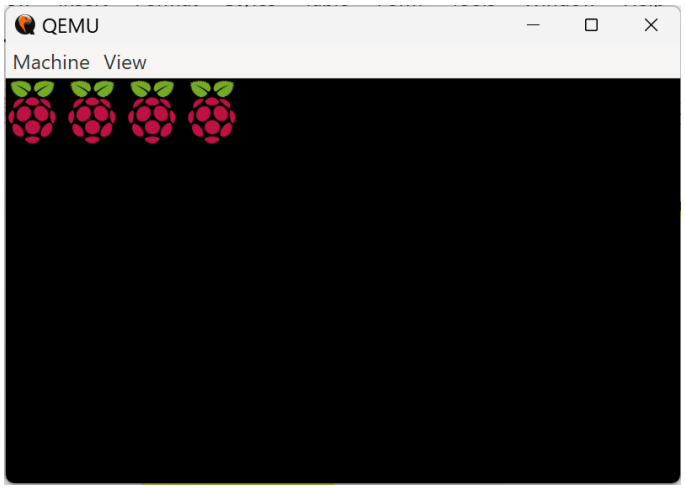
File Extraction from the Boot Partition of the Image File



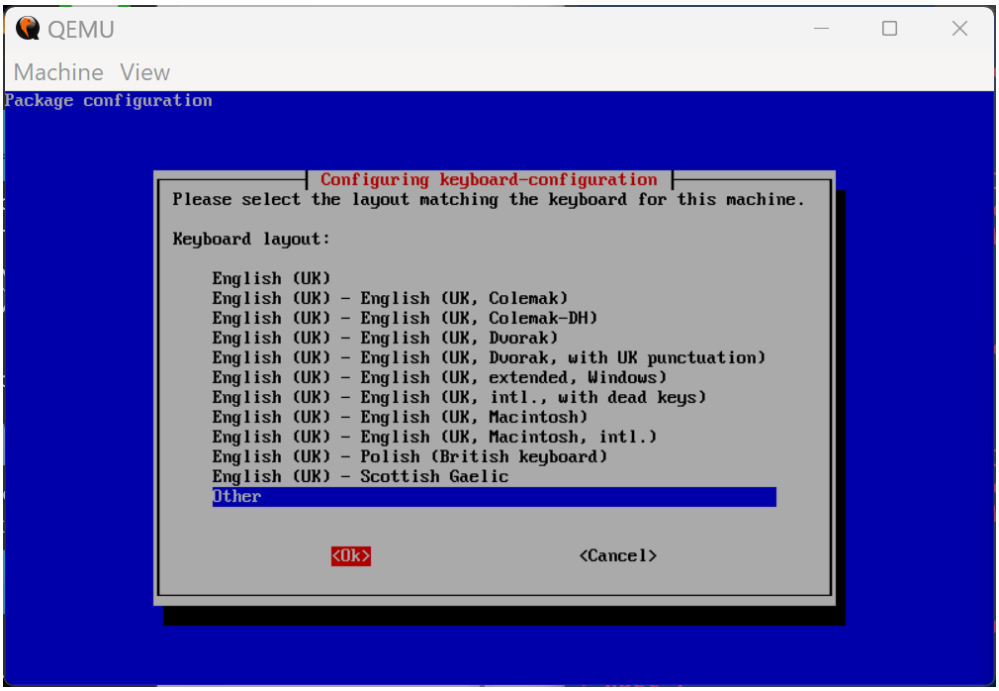


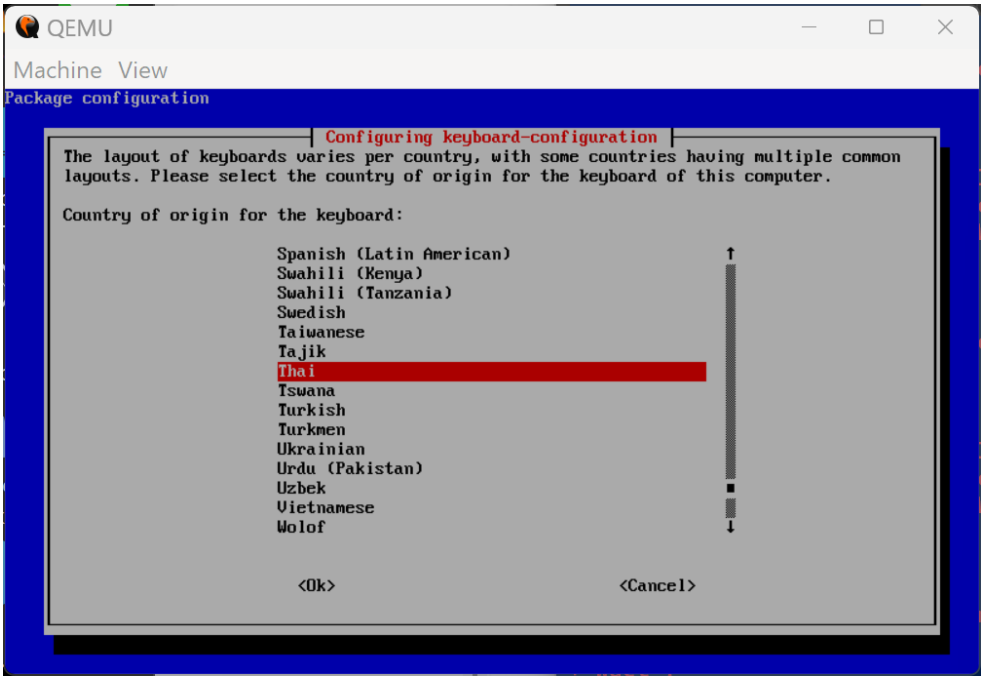


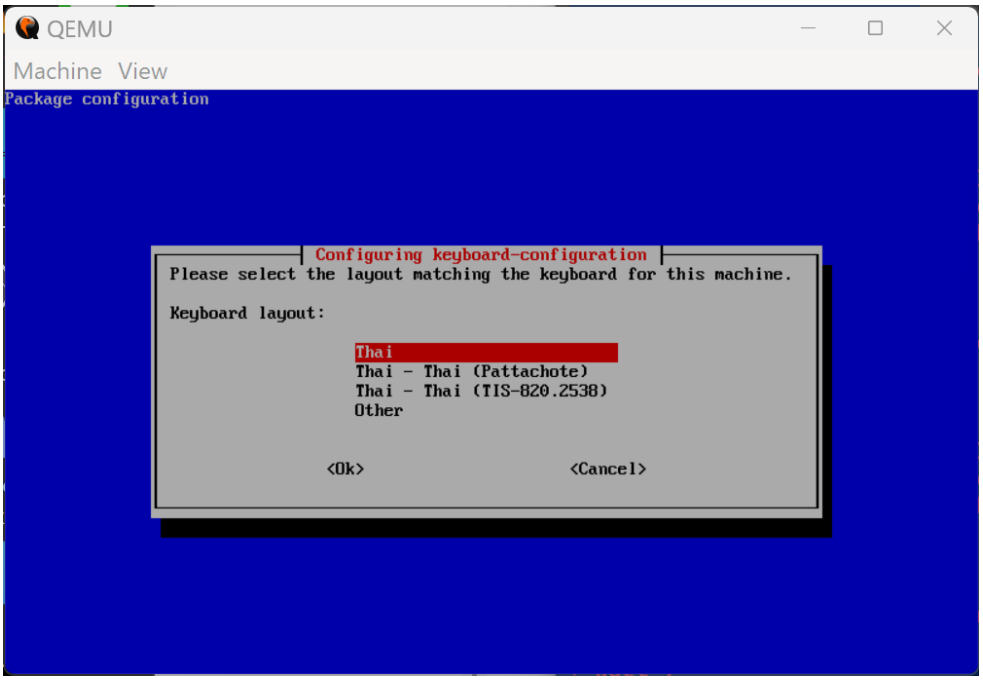
First-Run Installation Process for Raspberry Pi OS



Keyboard Layout Setup (Thai language)

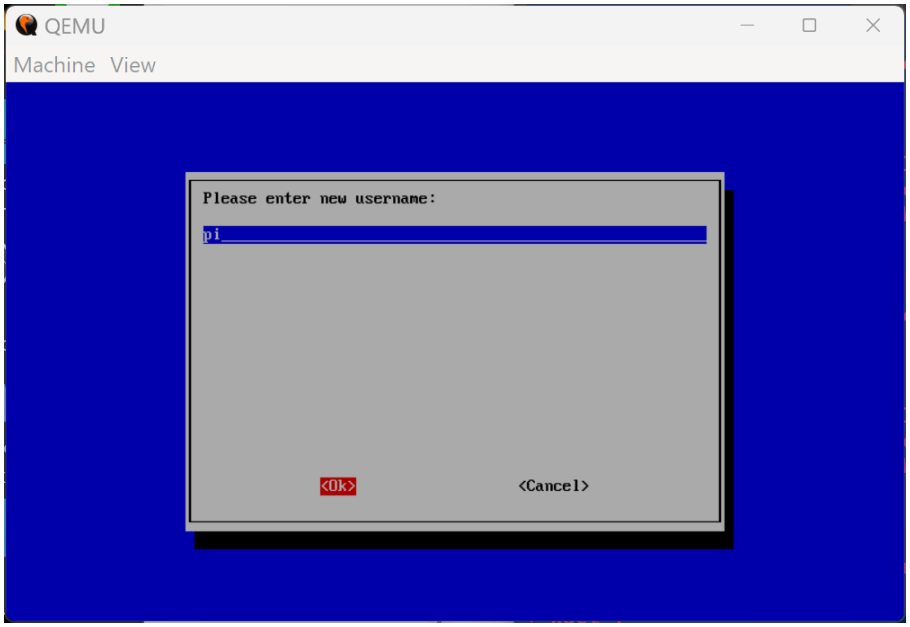


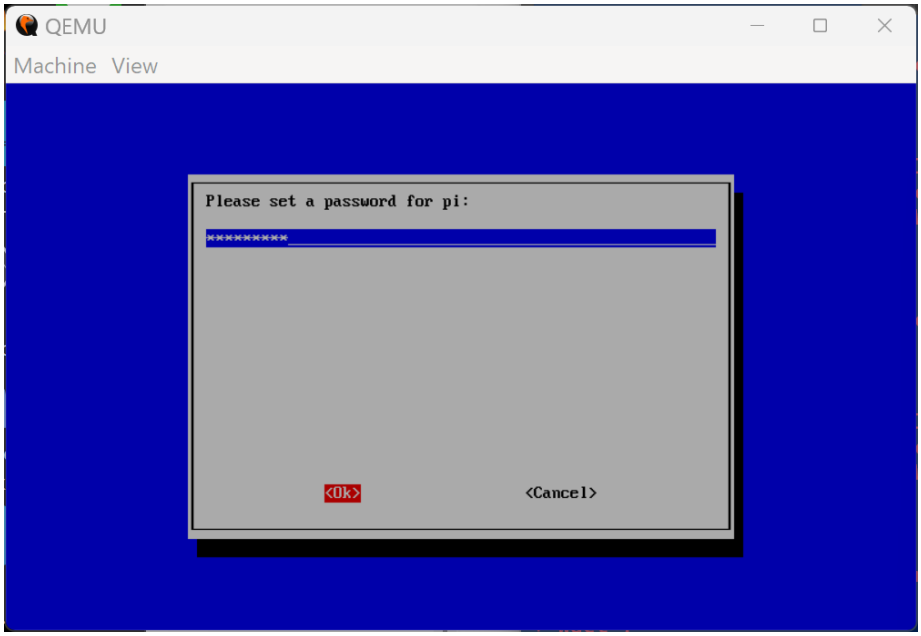




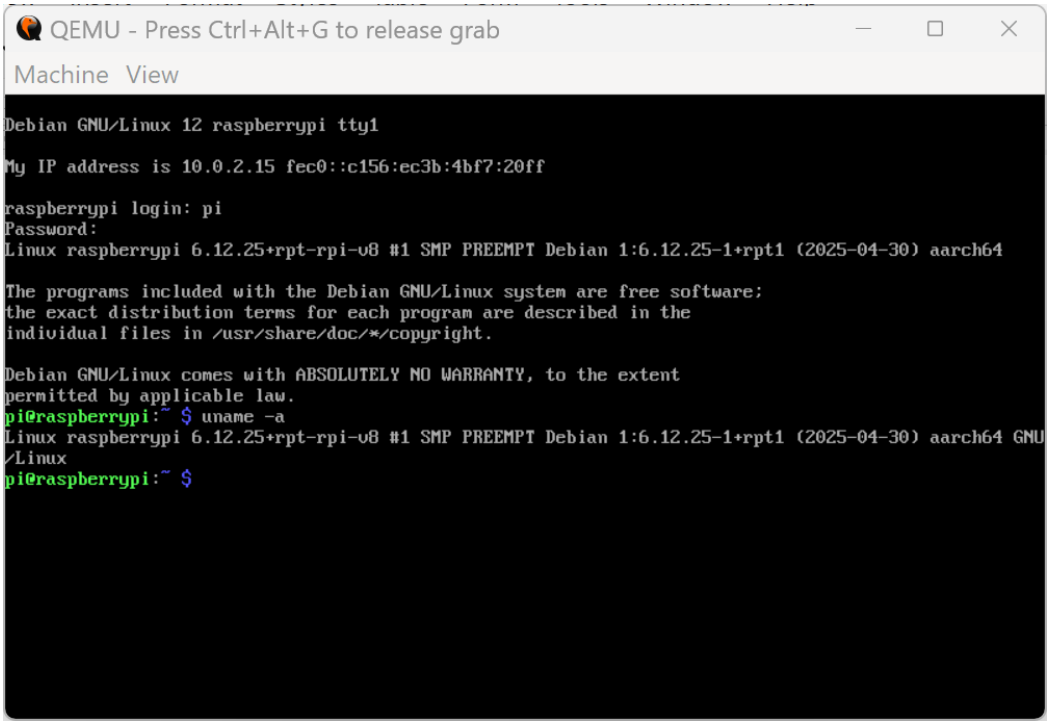
Keyboard Layout Setup (Thai)

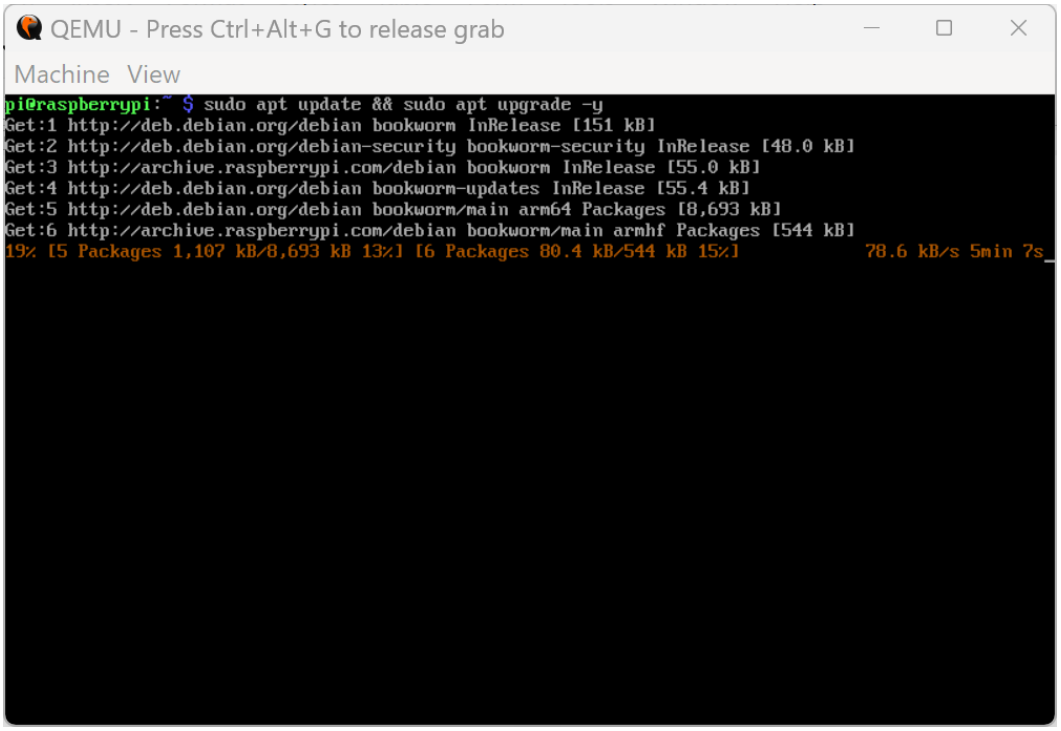
Add a new user and set the password
(pi : raspberry as default)





Login on the Linux console





Last update: 2025-06-28