

## Learning Outcomes

After completing this assignment, a student should be able to

- **Copy** and **start** an Ubuntu virtual machine, `UbuntuServer`, using `VBoxManage`.
- **Stop** the virtual machine two different ways.
- **Use** `ssh` to connect to the running virtual machine.

## Procedure

1. Read the assignment. This is **always** where you should start and **always** something you should do as soon as the assignment is available.

The work **must** be done inside of a *virtual machine* that is a copy of `UbuntuServer` (full path and copy instructions for the lab in `VBoxManage.org`).

2. **Copy UbuntuServer and SSH in.**

This first activity is done on a *host* computer, a computer that will run the `VirtualBox` program which runs other computers, *guest* machines, virtually. You may install the `UbuntuServer` virtual machine and run it on any computer with `VirtualBox` installed. The commands here will assume the work is being done on a CS Department Lab machine.

- (a) **Open a terminal emulator for a command-line interface (CLI).**

Go to the **Activities** in the desktop graphical environment. Start the *Console* program. This will run the `bash` shell program in your home directory on the lab machine.

- (b) **Copy UbuntuServer VM.**

The following commands, taken from `VBoxManage.org`, will copy the VM image from the class folder to one below your home directory, register the machine, and start and stop the machine. All of these commands are run on the *host* machine:<sup>1</sup>

```
$ mkdir -p ~/VirtualBox\ VMs/  
$ cp -r /home/student/Classes/371/UbuntuServer ~/VirtualBox\ VMs/  
$ VBoxManage registervm <your-home>/VirtualBox\ VMs/UbuntuServer/UbuntuServer.vbox  
$ VBoxManage modifyvm UbuntuServer --bridgeadapter1 enp0s31f6
```

When all of that works, you will have copied, registered, and updated the network on your virtual machine for the semester. It has a brand new Ubuntu Server 24.04 installation. The next two commands start and then stop the virtual machine.

```
$ VBoxManage startvm UbuntuServer --type headless  
Waiting for VM UbuntuServer to power on...  
VM UbuntuServer has been successfully started.  
$ VBoxManage controlvm UbuntuServer shutdown
```

- (c) **Log into the guest machine.**

Power the machine back up, find out the IP address it is using. After the power up you may need to wait thirty seconds or so.

```
$ VBoxManage startvm UbuntuServer --type headless  
Waiting for VM UbuntuServer to power on...  
VM UbuntuServer has been successfully started.  
$ VBoxManage guestproperty get UbuntuServer /VirtualBox/GuestInfo/Net/0/V4/IP  
Value: 192.168.100.XX
```

The actual IP address will be a quad of *decimal* numbers. Use the `Value` returned above for the following.

---

<sup>1</sup>The `$` is the machine's prompt, not something you should type.

```
$ ssh csuser@192.168.100.XX
csuser@192.168.1.36's password:
```

The password is Ki10Ech0 (those are zeros). You will reset it in the next assignment.

Notice that the next I/O box is a different color. That is to indicate that you are talking to the *guest* machine. That will also be shown in the prompt.

```
csuser@UbuntuServer:~$ ls -lA
total 44
-rw----- 1 csuser csuser  52 Jan 10 18:18 .bash_history
-rw-r--r-- 1 csuser csuser 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 csuser csuser 3771 Mar 31 2024 .bashrc
drwx----- 2 csuser csuser 4096 Jan 10 17:10 .cache
-rw----- 1 csuser csuser  32 Jan 10 18:19 .lesshst
-rw-r--r-- 1 csuser csuser  807 Mar 31 2024 .profile
drwxrwxr-x 2 csuser csuser 4096 Jan 10 18:36 src
drwx----- 2 csuser csuser 4096 Jan 10 17:09 .ssh
-rw-r--r-- 1 csuser csuser   0 Jan 10 17:21 .sudo_as_admin_successful
-rw----- 1 csuser csuser  59 Jan 10 18:19 .Xauthority
```

That was a long directory (`-l`) listing of almost all (`-A`) of the files in the home directory.

Now you will check the *version* of `g++` (the GNU C++ compiler) installed in the machine. Take note of the version as emailing the answer to Dr. Ladd is how you complete this assignment.

```
csuser@UbuntuServer:~$ g++ --version
... some output ...
```

Finally, you will shut down the machine from **inside** using the shutdown command. But only the superuser can run that command. `csuser` can do superuser things inside the VM!

```
csuser@UbuntuServer:~$ sudo shutdown now
[sudo] password for csuser:

Broadcast message from root@NubuntuServer on pts/1 (Fri 2025-01-10 19:22:18 UTC):

The system will power off now!
```

And you are kicked back to the prompt on the *host* machine.

- When you are done, e-mail Dr. Ladd, [laddbc@potdam.edu](mailto:laddbc@potdam.edu) with the name and number of the course in the *subject* and the **version** number of `g++` inside the virtual machine.