**NCL OpenStack VM Testing Intern Assignment**

**Image Creation Manual**

**Florentiana Yuwono and Thng Yu Xuan**

**Prerequisite**

This manual was created based on our experience in using **Windows 10**. If you are using a different operating system, there might be differences in setting up the necessary tools.

1. Make sure you have installed **Virt-Manager on Windows 10 using WSL**. KVM is a Linux kernel module and hence it can only be run on Linux systems. For Windows users, you can run KVM on a Linux virtual machine using VirtualBox/VMWare Workstation or use a Linux subsystem such as WSL. (Windows Subsystem for Linux) If you have not installed WSL, you can follow the installation procedure here.
   1. Enable **Windows Subsystem for Linux (WSL)**, **Virtual Machine Platform** and install **Ubuntu**. You might follow the guide here: <https://docs.microsoft.com/en-us/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package>
   2. After you are done with the installation, open your **Terminal** or **Command Prompt** and type in the following commands.

|  |  |
| --- | --- |
| Command | Description |
| ubuntu | Installing ubuntu. You will be prompted to create a default UNIX user account after this |
| sudo apt update | Update the packages |
| sudo apt install virt-manager | Install the virtual manager |
| virsh | Open the virtualization interactive terminal |
| virsh net-list  sudo virsh net-list --all | To see the libvirt network list. You must verify whether the default network is active. See here: <https://docs.openstack.org/image-guide/net-running.html> |
| virsh net-start default | To activate the default network, if it has not been activated |

* 1. If you do not see any network when running **virsh net-list**, it is likely that you must create the default network yourself. Type in the following commands to resolve this problem.

|  |  |
| --- | --- |
| Command | Description |
| sudo virsh net-list --all | If you indeed do not see any network listed, you could proceed. |
| touch default.xml | Create the default xml file |
| vim default.xml | Open the default xml file. If you have never installed vim before, follow the guide here <https://www.freecodecamp.org/news/vim-windows-install-powershell/> |
| <network>  <name>default</name>  <uuid>9a05da11-e96b-47f3-8253-a3a482e445f5</uuid>  <forward mode='nat'/>  <bridge name='virbr0' stp='on' delay='0'/>  <mac address='52:54:00:0a:cd:21'/>  <ip address='192.168.122.1' netmask='255.255.255.0'>  <dhcp>  <range start='192.168.122.2' end='192.168.122.254'/>  </dhcp>  </ip> </network> | Copy and paste the code. |
| :wq! | To save and exit from the default xml file |
| sudo virsh net-define --file default.xml | Add the network to the KVM host |

* 1. If you encounter an **error: failed to connect to the hypervisor** or **error: failed to connect to socket**, type in the following commands to resolve this problem.

|  |  |
| --- | --- |
| Command | Description |
| sudo apt install qemu qemu-kvm libvirt-clients libvirt-daemon-system virtinst bridge-utils | To make sure you have the proper libvirt and kvm |
| sudo apt-get install -y ebtables | To make sure you have installed ebtables |
| sudo systemctl enable libvirtd | To enable the libvirtd |
| sudo systemctl start libvirtd | To start the libvirtd |

* 1. If you encounter an error saying **System has not been booted with systemd as init system (PID 1)** or **failed to connect to bus: Host is down**, you need to type the following commands to resolve this problem.

|  |  |
| --- | --- |
| Command | Description |
| sudo -b unshare --pid --fork --mount-proc /lib/systemd/systemd --system-unit=basic.target | To start Systemd in a new namespace with its own PID mapping |
| sudo -E nsenter --all -t $(pgrep -xo systemd) runuser -P -l $USER -c "exec $SHELL" | To enter the namespace. You will be prompted to enter the password of your UNIX account |
| sudo systemctl enable libvirtd | To enable the libvirtd. You will be prompted to enter the password of your UNIX account |
| sudo systemctl start libvirtd | To start the libvirtd. If error persists, visit this guide for other ways of reboot <https://askubuntu.com/questions/1379425/system-has-not-been-booted-with-systemd-as-init-system-pid-1-cant-operate> |
| Sudo virsh net-list | You must now be able to see the activated default network. If you cannot see it, repeat step (C). Alternatively, visit this guide to see other possibilities for the error <https://blog.programster.org/kvm-missing-default-network> |

1. Install the [**MobaXterm Home Edition**](https://mobaxterm.mobatek.net/download-home-edition.html), which is an enhanced terminal for Windows with an X11 server, tabbed SSH client, network tools, etc. After that, you can open the application and run the **Virt-Manager GUI** on Windows 10. Follow this guide here. <https://www.how2shout.com/linux/how-to-install-and-use-virt-manager-on-windows-10/>
   1. Inside the MobaXterm WSL Session, type in the following commands.

|  |  |
| --- | --- |
| Commands | Description |
| sudo apt update | To run the ubuntu system update command |
| sudo apt install virt-manager | Make sure you have installed the virt-manager |
| virt-manager | You will be able to see a popup Virtual Machine Manager |
| sudo virt-manager | In case you cannot see the popup from command before |

* 1. If you encounter an error saying that **MoTTY X11 proxy: Unsupported authorization protocol**, **Connection refused**, or **cannot open display: localhost:10.0**, you can follow the guide here to resolve the problem <https://superuser.com/questions/1111900/how-to-fix-mobaxterm-x11-proxy-unsupported-authorisation-protocol>

**Procedure**

If you are using the **Virtual Machine Manager X11 GUI**, follow the steps below to create a new Virtual Machine.

1. Inside the **Virtual Machine Manager** popup, go to **File -> New Virtual Machine**. Alternatively, you could also click on the leftmost **desktop with star** icon.
2. Choose **Local install media (ISO image or CDROM)**. Alternatively, you could also choose another method, if you know what you are doing. Click **Forward**.
3. Choose the available ISO or CDROM install media. If you do not have one, follow the steps below on how to create a new volume.
   1. Click **Browse**. In **Volumes**, click the leftmost **photo** icon.
   2. Type in the name you want. You can also choose to adjust the **max capacity** or leave it to the default version. Click **Finish**.
   3. Click on the newly created volume and click **Choose Volume**.
4. Choose the operating system you are installing. If you tick the **Automatically detect from the installation media / source** and it displays **None detected**, you could untick and type manually. For example, you could type and choose **FreeBSD 12.0** and click **Forward**.
5. You will be prompted to configure other details about the **Memory and CPU Settings**. You could leave all these details to the default configuration and click on **Forward**.
6. Select **Enable storage for this virtual machine**. You could also select the default configuration of the **Create a disk image for the virtual machine** or **Select or create custom storage** for yourself.
7. Enter the name you desired, and tick **Customize configuration before install**. Click **Finish**.
8. You can view the details. Go to **VirtIO Disk 1** and on **Advanced options**, make sure that the **Storage format** is **qcow2**.
9. You could leave the other settings to its default and click **Begin Installation**.
10. You should now see the OS installation proceed within the Virtual Machine Manager window. Complete the installation and reboot.

If you are using **command line**, follow the steps below to create a Virtual Machine image:

1. Make a virtual drive using the command: qemu-img create -f qcow2 freebsd.qcow2 1G

This creates a virtual drive named ‘freebsd.qcow2’, format of qcow2, and 1GB of space.

1. Download the iso file of the operating system that you are using, using curl or wget command. For example, to download the FreeBSD 10.3 boot-only ISO for amd64 systems, go to the FreeBSD archive (<http://ftp-archive.freebsd.org/pub/FreeBSD-Archive/old-releases/ISO-IMAGES/10.3/>) and copy the link address of the os version ([FreeBSD-10.3-RELEASE-amd64-bootonly.iso).](http://ftp-archive.freebsd.org/pub/FreeBSD-Archive/old-releases/ISO-IMAGES/10.3/FreeBSD-10.3-RELEASE-amd64-bootonly.iso) Then the wget command will look like this: wget http://ftp-archive.freebsd.org/pub/FreeBSD-Archive/old-releases/ISO-IMAGES/10.3/FreeBSD-10.3-RELEASE-amd64-bootonly.iso
2. Run the command below to start a virtual machine using the virtual drive you created in Step 1, with the ISO file you downloaded in step 2 loaded in the virtual drive:

kvm -smp 1 -m 256 -cdrom FreeBSD-10.1-RELEASE-amd64-bootonly.iso -drive if=virtio,file=freebsd.qcow2 -net nic,model=virtio -net user

1. Select the first option in the first setup menu (Multi-user). Then follow the instructions on this page <https://docs.openstack.org/image-guide/freebsd-image.html>. Use TAB and ARROW KEYS to navigate in the FreeBSD installer.

There are also tools available for image creation. One such tool is disk-image-create, which provides an easy way to create images with just a single command. Simply run the command:

disk-image-create <distribution (ubuntu, fedora, centos etc.)> vm

This creates an image with qcow2 format which can be booted.