

Logbook 2023

Virtualisation and Cloud computing

Computer Name & IP: _____, 148.197.28.109

Content page

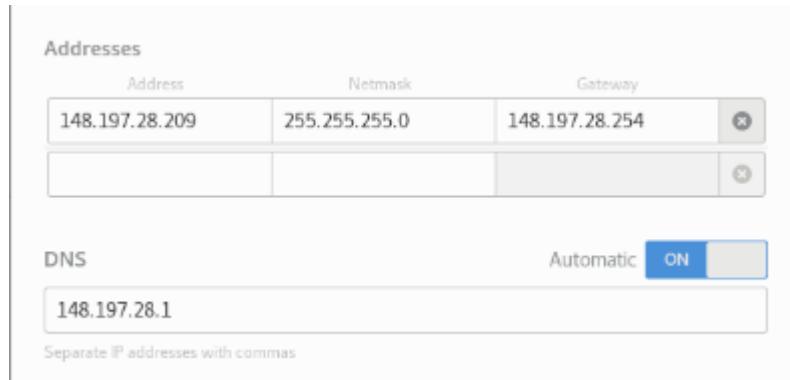
Introduction\Methodology	2
Lab Sheet 1 - VMware	4
Setting up a VM	7
Vmware and Virtual Machine settings	11
VMware default networking	16
VMware Custom networking	21
Final unassisted task for 70% (dynamic routing)	29
Lab Sheet 2 - Virtual box	36
Installation	36
Building a virtual machine	39
Virtual box and virtual machine settings	53
Virtualbox Networking	55
Lab Sheet 3 - Opennebula setup	71
Front-end installation/Add opennebula repositories	71
Installing the software	73
Starting OpenNebula	75
Verifying the installation	76
Sunstone	76
Dictionary structure	77
KVM Node Installation	78
Configure passwordless ssh	78
Lab Sheet 4 - VM creation	80
Adding a host	82
Adding a datastore	84
Adding networks	85
Adding images	87
Adding a template	90
Adding a virtual machine	96
Lab Sheet 5 - Sunstone	100
Basic usage - VMs	100
VM settings	107
Storage	107
Network card	108
Snapshot	110
Capacity	111
Permissions	112
Groups	114
Lab Sheet 6 - OVs	121
OVS Configuration	127
Networking in the cloud	132
Dynamic routing	136
Populating the cloud	141

Introduction\Methodology

This logbook is a demonstration of how a user can utilise virtualisation software like Vmware, Virtualbox, Opennebula and OVs to create multiple different virtual machines and use the host machines resources to make a virtual network that all the machines can run off. I will demonstrate commands that need to be run and files that may need some changes. These will need to be made according to your system's configuration.

Lab Sheet 1 - VMware

Creating a virtual network can be done in many different ways; this section of my logbook will explore how we can implement a virtual network while using VMware Workstation player 17.



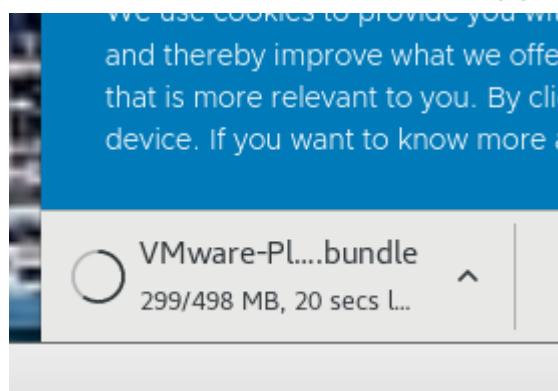
Firstly, I went and changed the IP, Netmask, Gateway, and DNS on my new host to allow me to connect to the internet and gain access to the network.

```

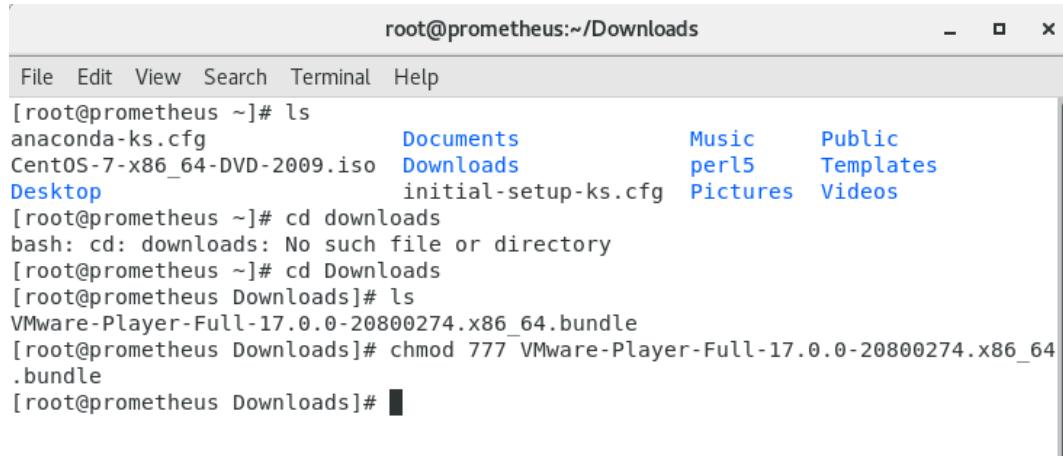
File Edit View Search Terminal Help
ABRT has detected 5 problem(s). For more info run: abrt-cli list --since 1675083
227
[root@prometheus ~]# ping google.com
PING google.com (142.250.200.14) 56(84) bytes of data.
64 bytes from lhr48s29-in-f14.1e100.net (142.250.200.14): icmp_seq=1 ttl=114 time=16.3 ms
64 bytes from lhr48s29-in-f14.1e100.net (142.250.200.14): icmp_seq=2 ttl=114 time=17.6 ms
64 bytes from lhr48s29-in-f14.1e100.net (142.250.200.14): icmp_seq=3 ttl=114 time=25.5 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 16.334/19.822/25.505/4.055 ms
[root@prometheus ~]#

```

Testing the connection on the machine to make sure I can ping google.



Went and downloaded VMware player from
[“https://www.vmware.com/uk/products/workstation-player/workstation-player-evaluation.html”](https://www.vmware.com/uk/products/workstation-player/workstation-player-evaluation.html) since we will be using this to run multiple virtual machines on our host system.



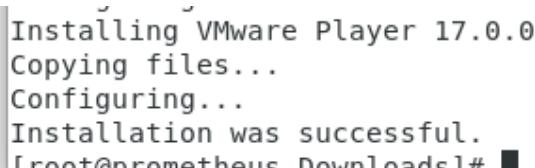
```
root@prometheus:~/Downloads
File Edit View Search Terminal Help
[root@prometheus ~]# ls
anaconda-ks.cfg      Documents      Music      Public
CentOS-7-x86_64-DVD-2009.iso  Downloads      perl5      Templates
Desktop              initial-setup-ks.cfg  Pictures   Videos
[root@prometheus ~]# cd downloads
bash: cd: downloads: No such file or directory
[root@prometheus ~]# cd Downloads
[root@prometheus Downloads]# ls
VMware-Player-Full-17.0.0-20800274.x86_64.bundle
[root@prometheus Downloads]# chmod 777 VMware-Player-Full-17.0.0-20800274.x86_64
.bundle
[root@prometheus Downloads]#
```

Going and giving vmplayer the ability to Read, Write, and execute, below shows now that VMware has max permissions.



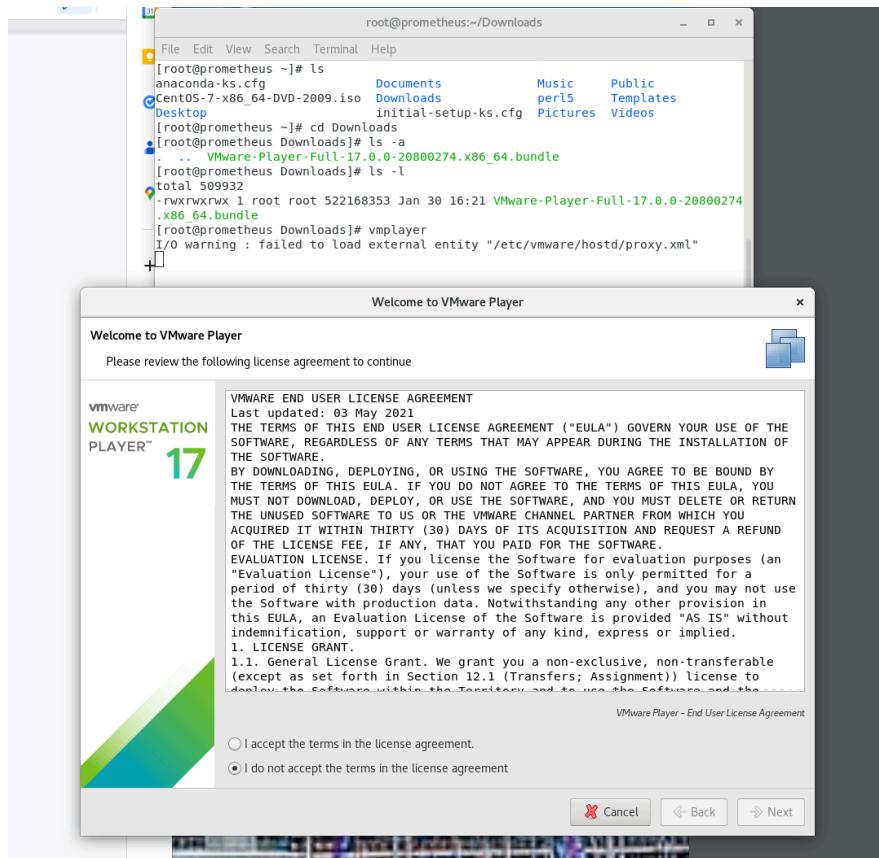
```
[root@prometheus Downloads]# ls -l
total 509932
-rwxrwxrwx 1 root root 522168353 Jan 30 16:21 VMware-Player-Full-17.0.0-20800274.x86_64.bundle
[root@prometheus Downloads]#
[root@prometheus Downloads]# ./VMware-Player-Full-17.0.0-20800274.x86_64.bundle
Extracting VMware Installer...done.
Installing VMware Installer 3.1.0
Copying files...
Configuring...
Installing VMware Player Setup 17.0.0
Copying files...
Configuring...
Installing VMware USB Arbitrator 20.8.0
Copying files...
Configuring...
Installing VMware Network Editor 17.0.0
Copying files...
Configuring...
Installing VMware VMX 17.0.0
Copying files...
Configuring...
```

Getting the VMware player application to install used the : “./”(filename*) to run the file and install the bundle package.

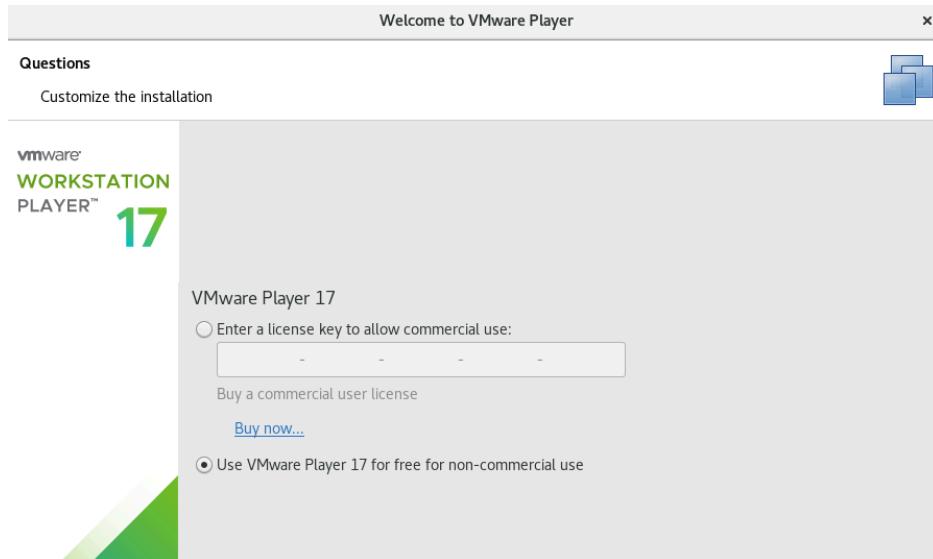


```
Installing VMware Player 17.0.0
Copying files...
Configuring...
Installation was successful.
[root@prometheus Downloads]#
```

This tells us that the package was successfully installed

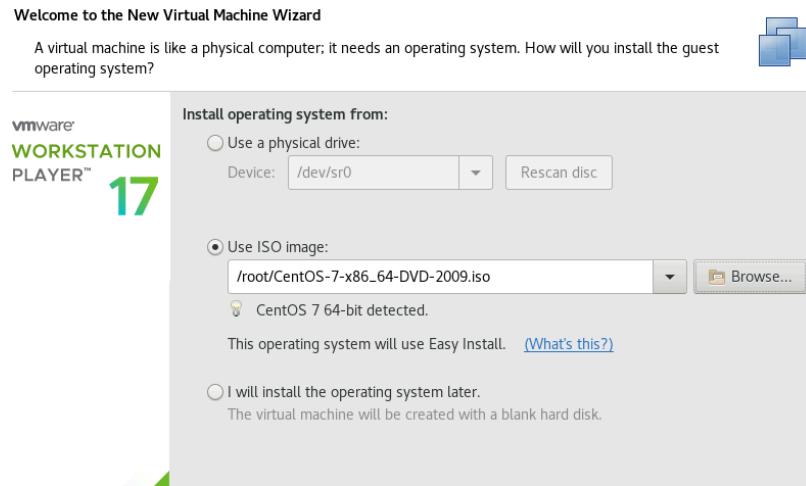


Going and invoking the vmware player application, make sure to accept the licensing agreement. Invoking it can be done by either clicking the icon that is present on the desktop or doing it through the command line

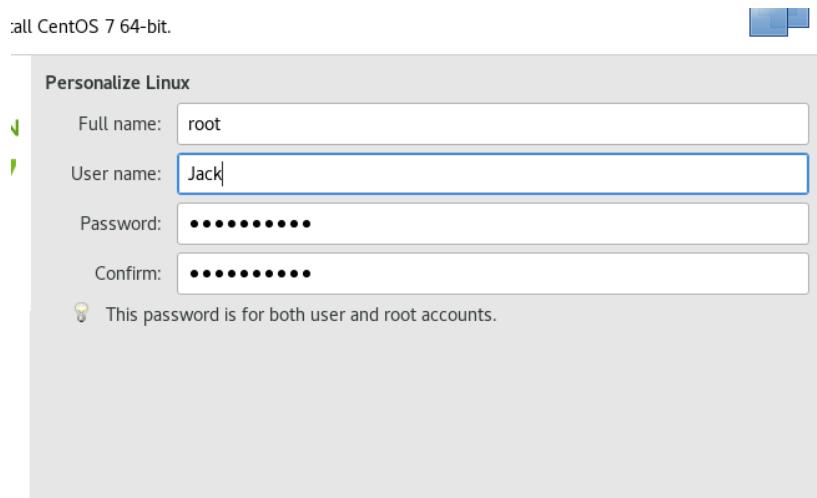


I do not have a licence key so we will be using the non-commercial version of VMware

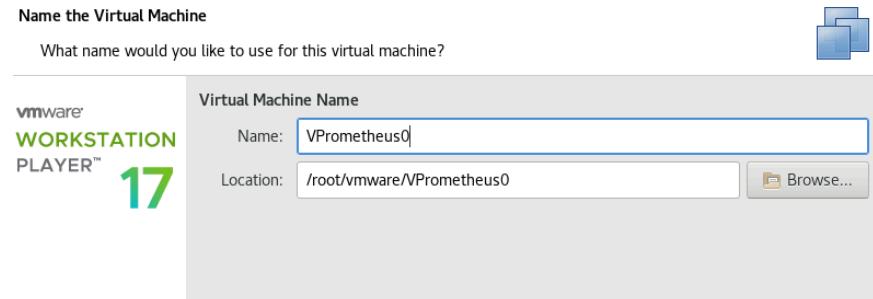
Setting up a VM



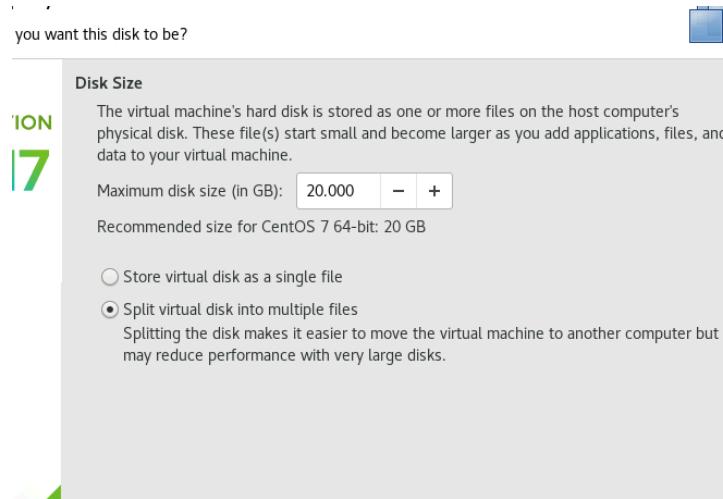
Picking the Centos-7 ISO file, this is so my machine install the centOS operating system



This is the user account and the root account. I gave it the full name root and set the password for both the accounts to “localhost1”. We will only be using root since it makes it a lot easier when using commands and changing files that may need “sudo” privileges. We can set the root password for the actual operating system in the installation process of the operating system.



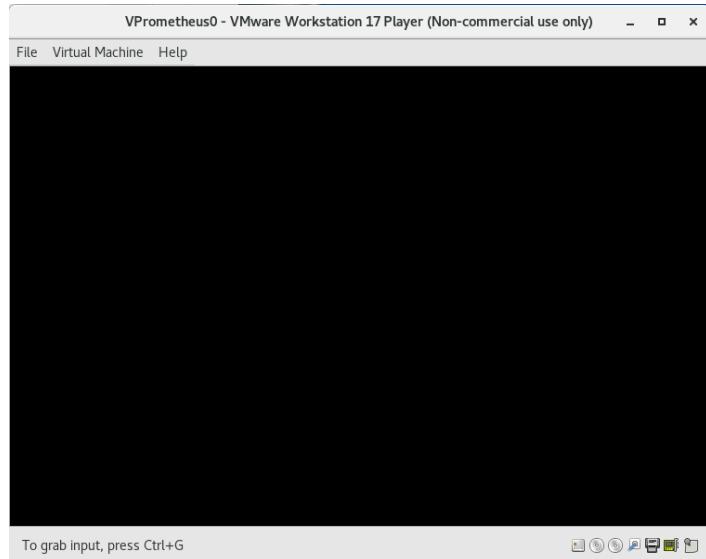
Setting the name for the vm to “VPrometheus0”, since this is the very first machine in the series



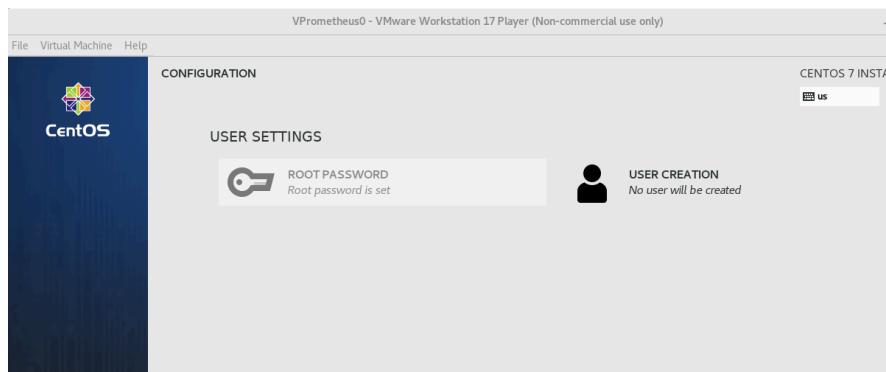
Keeping the storage on the VM as default “20GB” since this will be enough for the operating system and everything we will be installing

Device	Summary	Processors
Memory	4 GB	Number of pro
Processors	1	
New CD/DVD (IDE)	Using file /root/CentOS-7-	Virtualization Eng
Network Adapter	NAT	<input type="checkbox"/> Virtualize In
Sound Card	Auto detect	<input type="checkbox"/> Virtualize C
Printer	Present	<input type="checkbox"/> Virtualize IC
USB Controller	Present	
Display	Auto detect	

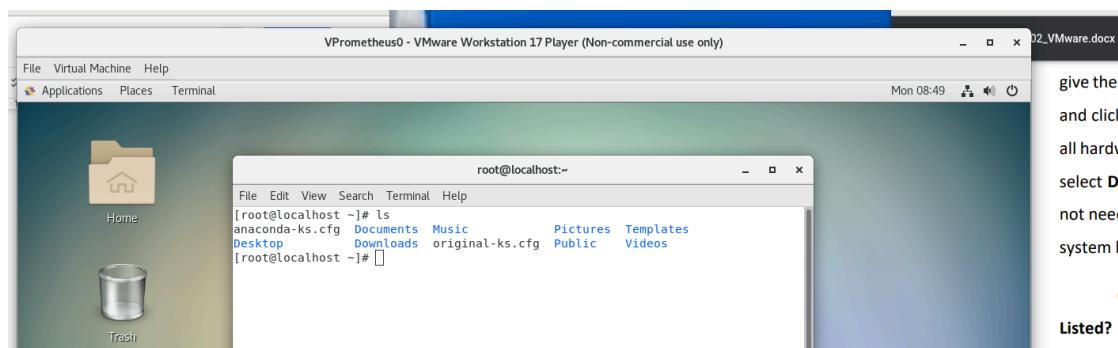
Going and setting the memory to “4GB” since that's the standard for running the operating system



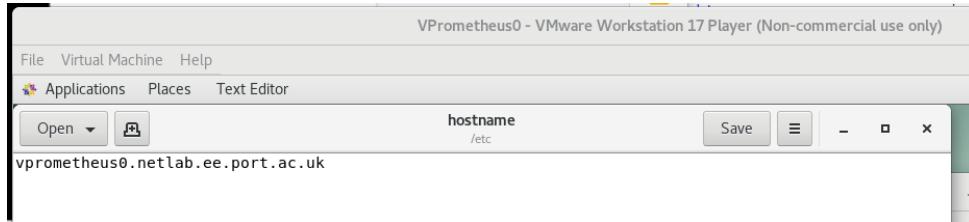
The new machine loading and beginning the installation process



Going and specifying the root password to “localhost” and maybe create a user account but we will not be using it, so you do not have to make one



Go and login to the root user account



Changing the hostname on the new machine to “vprometheus0.netlab.ee.port.ac.uk”

The screenshot shows a terminal window titled "root@vprometheus0:~". The terminal displays the command "ping google.com" followed by several lines of output showing ping results to "lhr48s27-in-f14.1e100.net". The output includes fields like icmp_seq, ttl, time (ms), and bytes sent.

```
[root@vprometheus0 ~]# hostname
vprometheus0.netlab.ee.port.ac.uk
[root@vprometheus0 ~]# ping google.com
PING google.com (142.250.178.14) 56(84) bytes of data.
64 bytes from lhr48s27-in-f14.1e100.net (142.250.178.14): icmp_seq=1 ttl=128 time=24.8 ms
64 bytes from lhr48s27-in-f14.1e100.net (142.250.178.14): icmp_seq=2 ttl=128 time=6.95 ms
64 bytes from lhr48s27-in-f14.1e100.net (142.250.178.14): icmp_seq=3 ttl=128 time=9.89 ms
64 bytes from lhr48s27-in-f14.1e100.net (142.250.178.14): icmp_seq=4 ttl=128 time=8.89 ms
64 bytes from lhr48s27-in-f14.1e100.net (142.250.178.14): icmp_seq=5 ttl=128 time=13.2 ms
^C
```

After changing the hostname I go and reboot the machine to make sure that fully goes into effect. Also, I ping google to make sure we have a full connection to the network and the internet

Vmware and Virtual Machine settings

Virtual machine questions :

Is it every bit as functional as a “real” computer? - Virtual machines are designed to emulate a “real” computer. Every functionality a regular computer system has will be present in a virtual machine, including operating systems and hardware, just in the form of software.

The only issue is that a virtual machine can be slower than a regular computer since you're running it through the software on a typical machine which means not all resources will be allocated to the VM.

Go to Virtual Machine Settings. What options are there? - In the setting of VMware, you can change a lot to the virtual machine even after creation: memory, processors, Hard Disk, CD/DVD, Network adapters, USB controller, Sound cards, Printers, and display. These are all related to the VM shows how you can do a lot of changing and editing with the machine at any point.

Also, there is an options tab where you can change some little things like shared folders, access control of the VM if you want VM tools and auto-login. These are all settings that allow you to modify how your machine works.

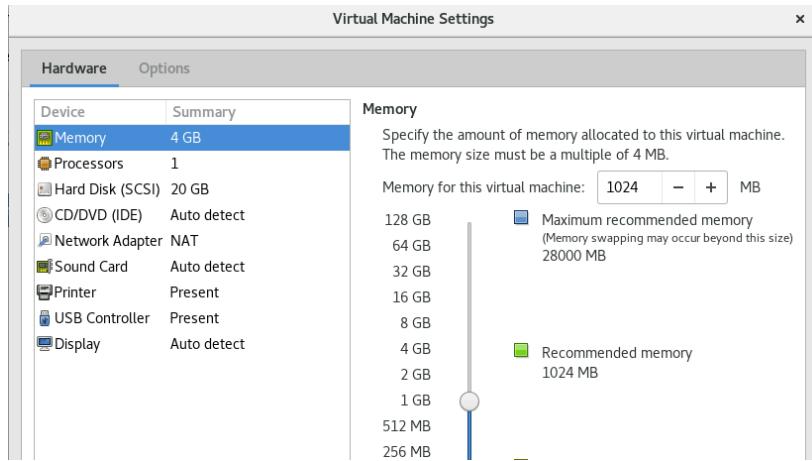
What is the difference between Suspend Guest and Power Off Guest? - When choosing “suspend guest”, the system's current state will be saved to the disk, meaning whenever you load the machine back up, even if you shut the host down, no matter what, it will be saved where you last suspended it, with all things you had open. It gets the saved state from the disk where it was saved.

When choosing “Power off guest” this is the normal way of shutting down any computer; it will wipe anything last open and anything not saved will be unsaved. This is usually done when pressing the power button on any computer or using the shutdown button

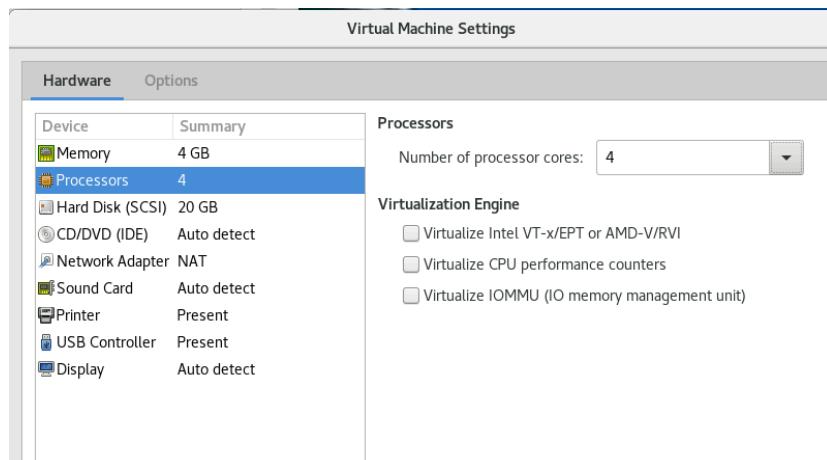
What can you do with the removable devices in a VM? - Virtual machines again do this very similarly to actual computers

You can add and remove devices such as USB and CD drives. This allows for the transfer of files or software. With this, though, if you want to connect a device to your VM, add it to

your host and then connect it to the virtual machine through Vmware. With this, you can add new ones, disconnect old ones, configure the removable devices to your VM and even allow the use of shared folders.



Changing memory from 4 - 1, what happens? - After going and changing my memory from 4GB - 1GB nothing really changed, the only difference is that the machine tended to run a little slower then it did with 4GB



Change the number of cores, what happens? - Again even after doing this nothing really changed, the system still ran and loaded the same but again it did start to run a little slower

Processors

Number of processor cores: 32

⚠️ Powering on the virtual machine will fail because it is configured to use more virtual processor cores than the host supports.

Virtualization Engine

Maximum number of cores on the system is 32

Research the 3 Virtualization engines and write a short paragraph about each of them explaining what they do:

With Vmware there are multiple different types of virtualisation engines three that I will be talking about are Full virtualisation, hardware virtualisation, paravirtualization.

Full virtualisation: This is one type of virtualisation engine that VMware implements where you leave the operating system stock so that you can run the binary of that operating system on top of the hypervisor. This is done using binary translation; this will not work if the operating system's code is modified. When this is all present, the VME Completely simulates the hardware.

- This engine reduces upfront hardware costs
- Eliminates downtime
- Increases productivity and latency
- And is faster at providing resources to applications

Hardware Virtualisation: This is a second type of virtualisation engine, quite similar to just Virtual machine software. But this engine separates resources\requests for specific services from the basic service. The use of physical hardware allows the full ability to use material resources which can power servers and desktops. This is done using a single host machine that can create multiple environments using a hypervisor. The visor will connect straight to the hardware on your system and then go and use some of the resources to power other virtual machines. This engine is made up of multiple components:

- The hardware layer, hypervisor, and virtual machines allow you to create virtual machines using your resources.

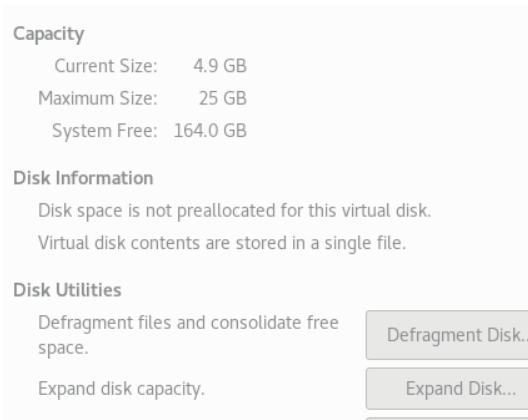
With hardware virtualisation, you get many different advantages:

- Increases security
- Increases performance

- Improves the reliability of the Virtual Machine

Paravirtualisation: Instead of attempting to simulate a natural hardware environment, this method modifies the virtual machine's operating system before the user installs a virtual machine, allowing all other VMs on the host system to share resources and work together.

- This engine is ideal for fixing issues that can be present in full virtualisation, allowing the user to make edits to the operating system. This engine also provides communication between the design and the hypervisor; this creates a performance increase in the system.
- The operating system can be modified by changing the "kernel" to be in use with the hypervisor; this is done by removing operations that run with "ring 0" of the processor when calls to the hypervisor are done.



I went and changed my maximum files size to see if it would affect any of the total size of any files on my machine

 <p> Names: vmware.log, vmware-0.log, ... Type: – Contents: 15 items, totalling 5.3 GB Parent Folder: /root/vmware/VPrometheus0 Accessed: – Modified: – </p>	<p>Properties</p> <table border="1"> <tr> <th>Basic</th> <th>Permissions</th> </tr> <tr> <td>  <p> Names: vmware.log, vmware-0.log, ... Type: – Contents: 15 items, totalling 5.3 GB Parent Folder: /root/vmware/VPrometheus0 Accessed: – Modified: – </p> </td> <td></td> </tr> </table>	Basic	Permissions	 <p> Names: vmware.log, vmware-0.log, ... Type: – Contents: 15 items, totalling 5.3 GB Parent Folder: /root/vmware/VPrometheus0 Accessed: – Modified: – </p>	
Basic	Permissions				
 <p> Names: vmware.log, vmware-0.log, ... Type: – Contents: 15 items, totalling 5.3 GB Parent Folder: /root/vmware/VPrometheus0 Accessed: – Modified: – </p>					

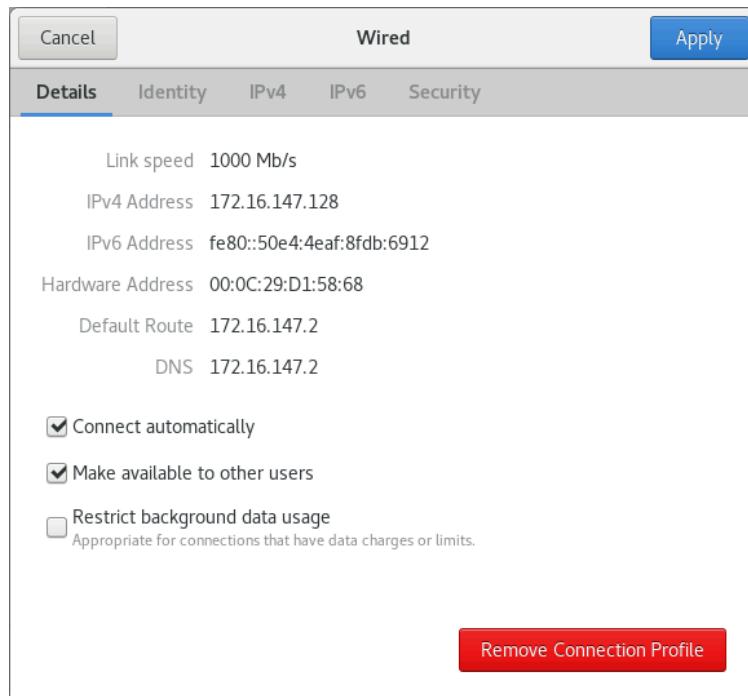
Before

After

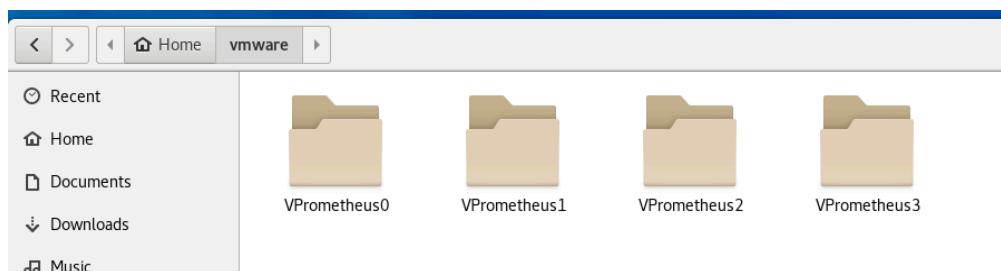
As you can see there was no change to the size of the files after changing the total size of the storage device

VMware default networking

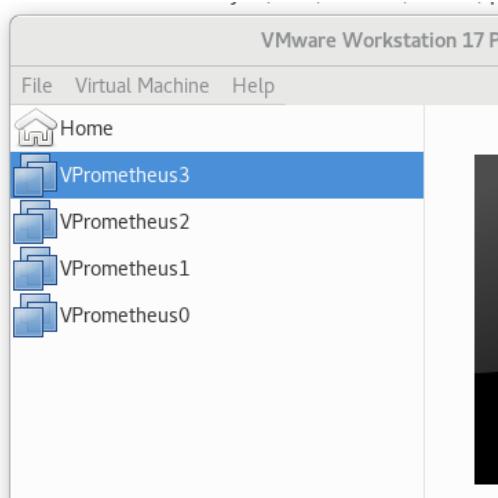
These are the machines: default IP address, gateway, DNS server. (**how did these get allocated?**)



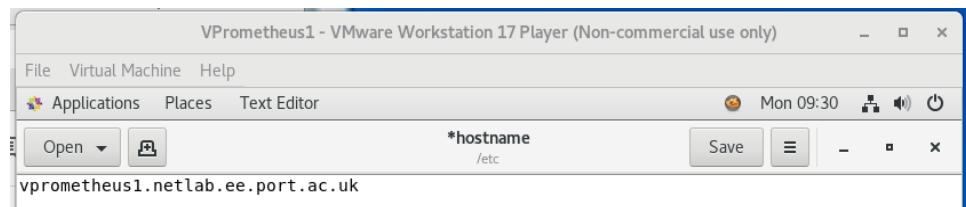
These were allocated through DHCP; they were automatically allocated after creating the machines since they are connected to a host and using NAT. NAT for the virtual machine uses the host's network access to get its network information and gain access to the network



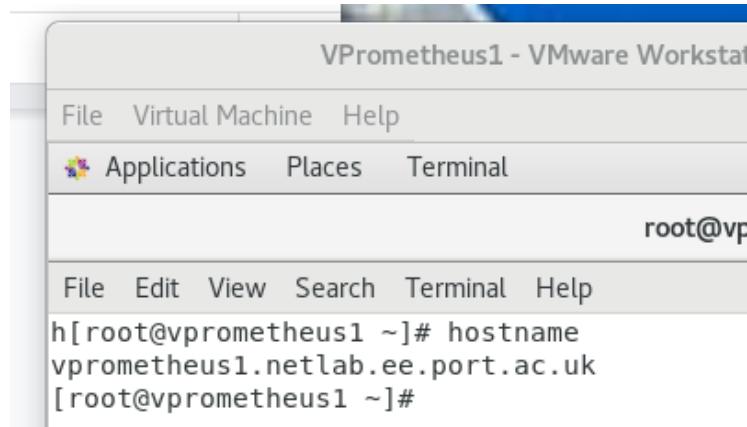
I went and copied the first virtual machine that I had created and then changed the name since we will be using multiple virtual machines. We need multiple so we can make the virtual network and run it through different machines



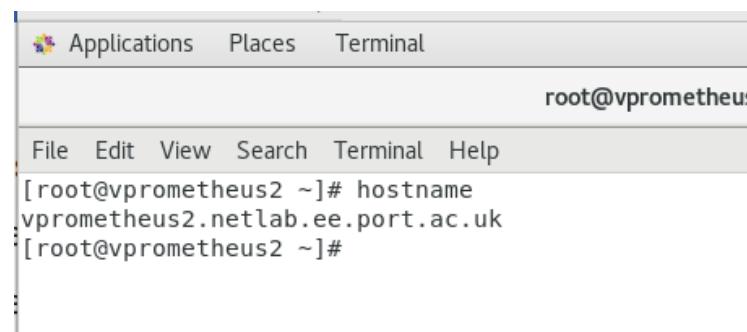
Added all of the VMs to VM-player and changed their names to there respective VM name



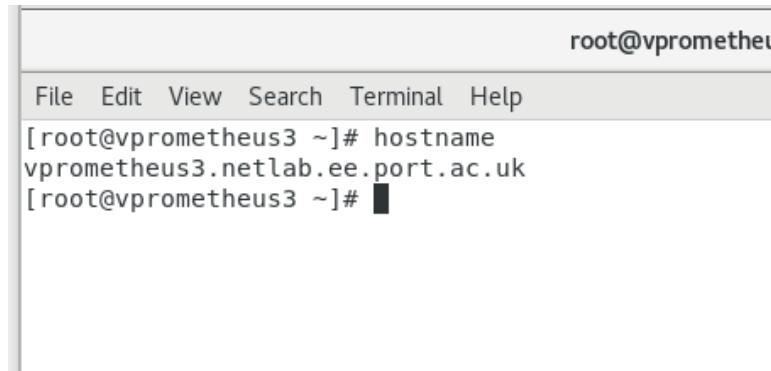
Changing the hostname to match the VM, after doing this go and reset the machines to fully allocated the new host name make sure you do this for all of them



Changing the hostname for VP1

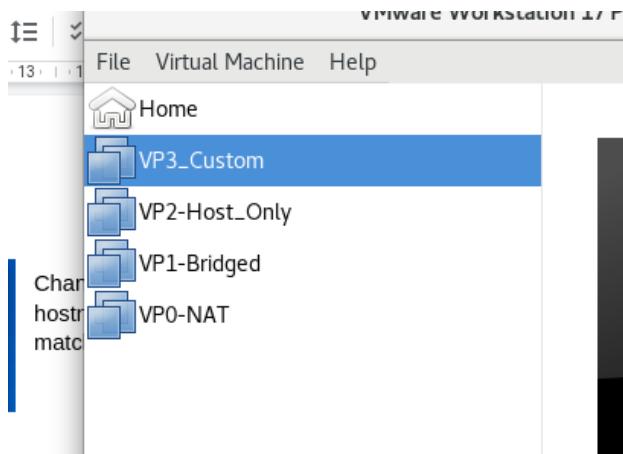


Changing the hostname for VP2

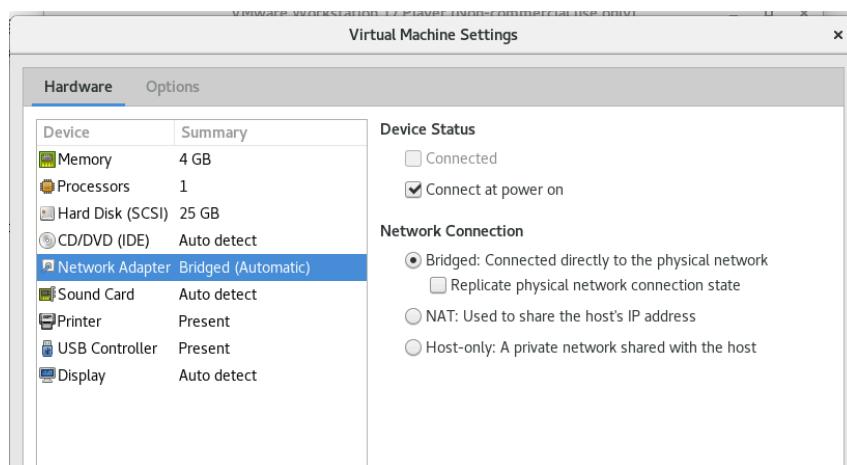


```
root@vprometheus3 ~]# hostname
vprometheus3.netlab.ee.port.ac.uk
[root@vprometheus3 ~]#
```

Changing the hostname for VP3

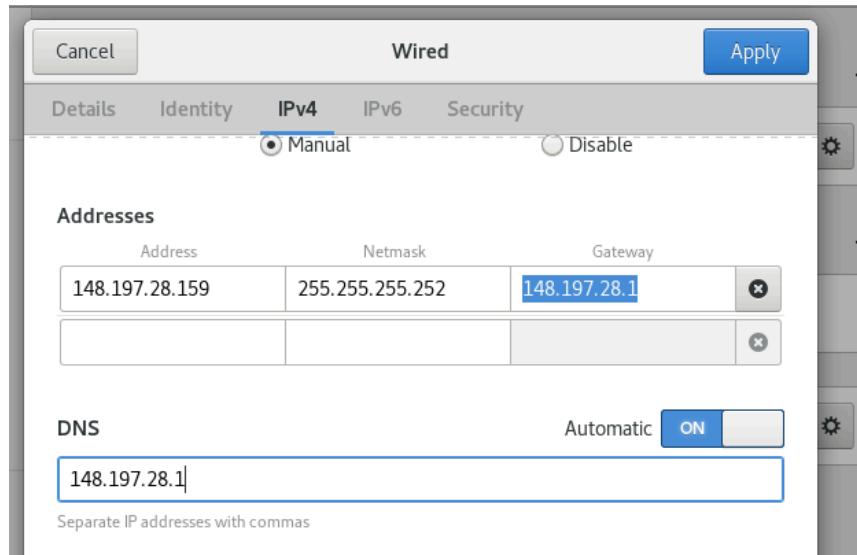


Changing the vm names to a more respective name

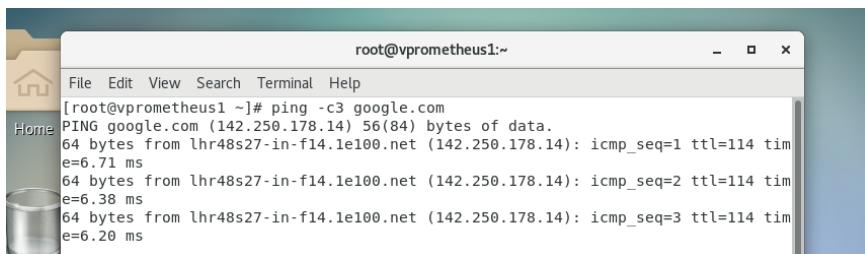


Setting the network adapter for VP1 to bridged (**what does bridged mean**) Bridged is a virtual machine connecting to a network using the physical network adapter on your host machine

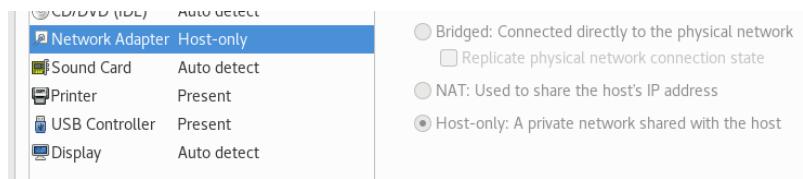
(How should you configure the IP address inside the VM?) I will need to go and specify the ip address, subnet mask, gateway, and DNS to make sure it actually works, usually, we would add plus 50, but since my ip host machine is 209 going to 259 would not work so I have to use the “.159”.



Giving the correct specified network addresses to the VP1 machine



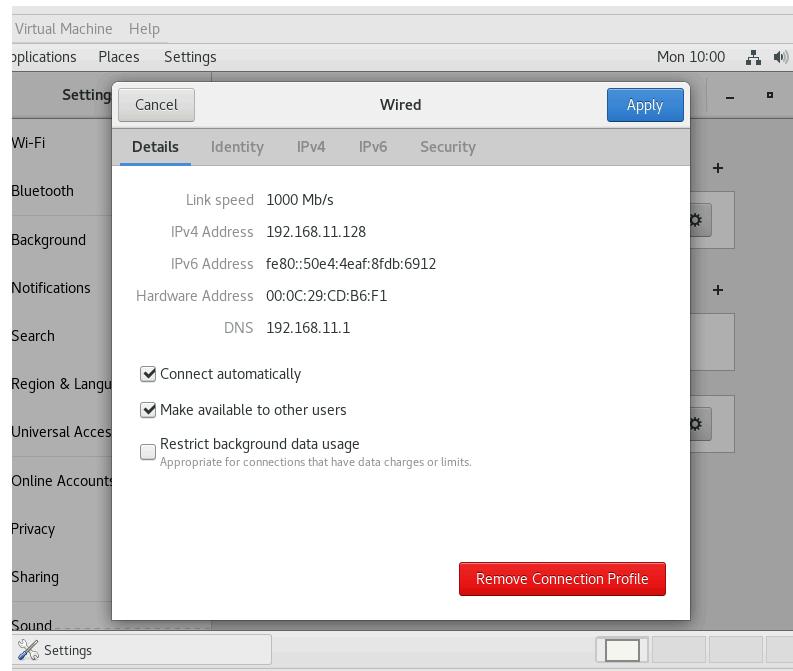
Testing the network for these machines by pinging Google



Setting the VM2 to host only **What does Host Only mode mean?** This type of network means that the guest os can only talk to the host, it is quite similar to bridge where it uses the host to gain a network connection



This was VM2 when it was on NAT



This is VM2 when it is on host only mode

VMware Custom networking

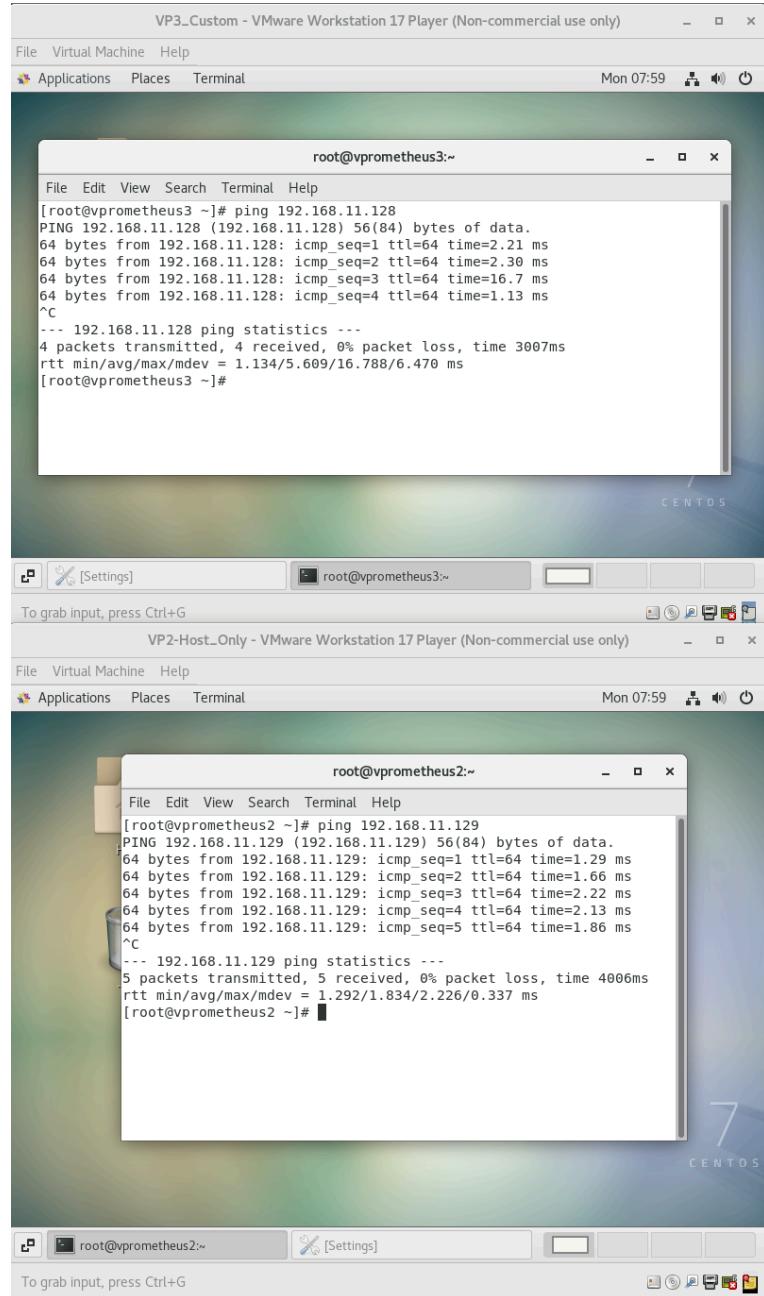
Hard Disk (SCSI)	25 GB
CD/DVD (IDE)	Auto detect
Network Adapter	Host-only
Sound Card	Auto detect
Printer	Present

I am setting the network adapter on VP3 from nat to host-only

Link speed 1000 Mb/s
IPv4 Address 192.168.11.129
IPv6 Address fe80::50e4:4eaf:8fdb:6912
ware Address 00:0C:29:01:F5:36
DNS 192.168.11.1

This is the IP address of the host only system (marked in red)

The below image demonstrates how I can ping between the two systems VP3 and VP2, I use the IP from one of the systems to make sure I can ping the other one, done by the command :ping *ip*



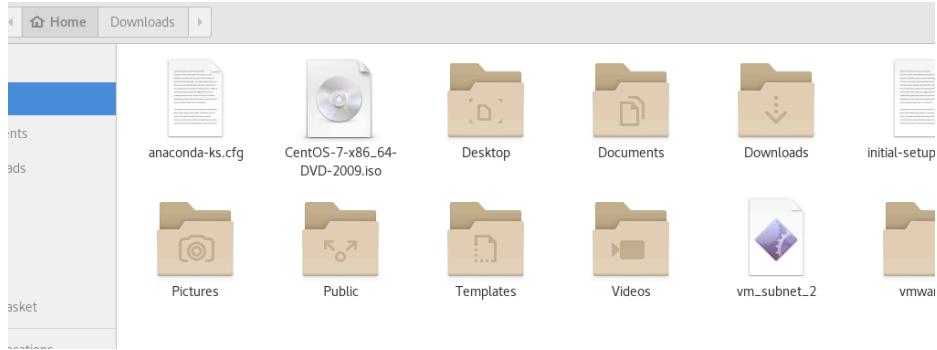
Although I can ping between the two systems I do have a connection to the outside world

```
* 4 packets transmitted, 4 received, 0% packet loss, time 2
rtt min/avg/max/mdev = 1.134/5.609/16.788/6.470 ms
[root@vprometheus3 ~]# ping google.com
^C
[root@vprometheus3 ~]# 
```

What are the limitations of HostOnly Networking?

Host-only networking is a network type that can be compared to bridged and internal; due to this, many different kinds of limitations can be caused by it:

- Host-only networking does not get internet access due to it being configured to where the guest machines only talk to the primary host.
 - This can also cause network connectivity issues since the main communication form is between the two machines.
 - This idea of the system being only connected to the host causes limited functionality since we are only given limited network ability. What we can do on the host-only networking system can only do specific things.
-



Downloading vm_subnet_2 and saving it to the home folder, since this is the easiest way to access it

```

      fi
done

#####
VMNET=2
VMNET_IP=148.197.30.253          #Device driver and interface number - /
NETMASK=255.255.255.252          #Ip address of the virtual interface on
NETWORK=148.197.30.252           #Subnet mask
BROADCAST=148.197.30.255         #Address of subnet - Must match NETMASK
                                #Broadcast address - last ip address in

#Start process that associates a device driver /dev/vmnet0 say with a n
(vmswitch)
if ps -efl | egrep vmnet-netifup.*vmnet${VMNET}$ > /dev/null; then echo
running;
else vmnet-netifup -d /var/run/vmnet-netifup-vmnet${VMNET}.pid /dev
${VMNET}; fi

#Bring up the interface vmnet2 say (VMNIC0)
#if ifconfig | egrep ^vmnet${VMNET} > /dev/null; then echo vmnet${VMNE
#else ifconfig vmnet${VMNET} ${VMNET_IP} netmask $NETMASK broadcast $BRO
#### vmnet+ netifup+ vmnet+ vnet+ netmask+ broadcast+ app

```

This is what the default file had saved, I went and changed it to match my system shown below, this is for VP3. This helps with overcoming the limitations of host-only networking

```

      chmod 777 /dev/vmnet$a
      fi
done

#####
VMNET=2
VMNET_IP=148.197.30.81          #Device driver and interface number - /
NETMASK=255.255.255.252          #Ip address of the virtual interface on
NETWORK=148.197.30.80           #Subnet mask
BROADCAST=148.197.30.83         #Address of subnet - Must match NETMASK
                                #Broadcast address - last ip address in

#Start process that associates a device driver /dev/vmnet0 say with a n
(vmswitch)
if ps -efl | egrep vmnet-netifup.*vmnet${VMNET}$ > /dev/null; then echo
running;
else vmnet-netifup -d /var/run/vmnet-netifup-vmnet${VMNET}.pid /dev
${VMNET}; fi

#### vmnet+ netifup+ vmnet+ vnet+ netmask+ broadcast+ app

```

```
[root@prometheus ~]# chmod +x vm_subnet_2
```

Giving the file the max permissions on the file, since we will need it to be able to read, write and execute

drwxr-xr-x. 2 root root	6 Jan 30 10:46 Templates
drwxr-xr-x. 2 root root	6 Jan 30 10:46 Videos
-rwxr-xr-x 1 root root	3654 Feb 6 16:16 vm_subnet_2
drwxr-xr-x 6 root root	86 Jan 30 17:22 vmware

The new permissions of the file

```
[root@prometheus ~]# ps -ef | grep vmnet
root      11632      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-bridge -s 6 -d
/var/run/vmnet-bridge-0.pid -n 0
root      11640      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-netifup -s 6 -d
/var/run/vmnet-netifup-vmnet1.pid /dev/vmnet1 vmnet1
root      11645      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-dhcpd -s 6 -cf
/etc/vmware/vmnet1/dhcpd/dhcpd.conf -lf /etc/vmware/vmnet1/dhcpd/dhcpd.leases -p
f /var/run/vmnet-dhcpd-vmnet1.pid vmnet1
root      11649      1  0 14:01 ?          00:00:08 /usr/bin/vmnet-natd -s 6 -m /e
tc/vmware/vmnet8/nat.mac -c /etc/vmware/vmnet8/nat/nat.conf
root      11651      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-netifup -s 6 -d
/var/run/vmnet-netifup-vmnet8.pid /dev/vmnet8 vmnet8
root      11657      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-dhcpd -s 6 -cf
/etc/vmware/vmnet8/dhcpd/dhcpd.conf -lf /etc/vmware/vmnet8/dhcpd/dhcpd.leases -p
f /var/run/vmnet-dhcpd-vmnet8.pid vmnet8
root      64608  63842  0 16:17 pts/6    00:00:00 grep --color=auto vmnet
[root@prometheus ~]#
```

The command “*ps -ef| grep vmnet*”, goes and collects everything single process with the name vmnet

```
[root@prometheus ~]# ./vm_subnet_2
[OK] Setting up and Configuring VMplayer devices..
vmmon already insmod
vmnet already insmod

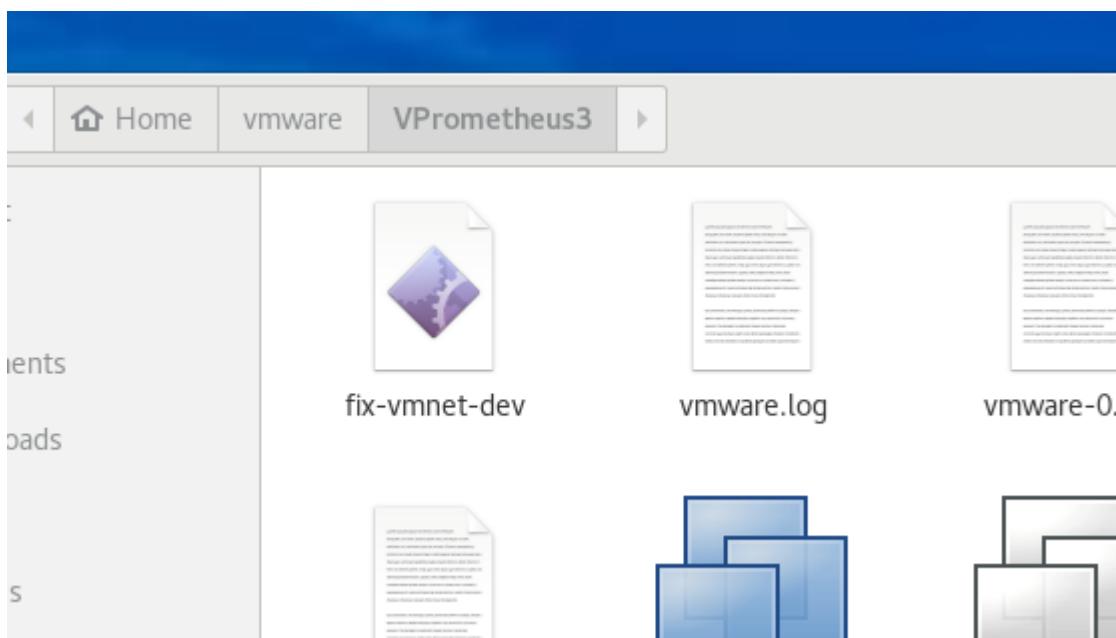
[OK] Virtual Switch running are..

root      11640      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-netifup -s 6 -d
/var/run/vmnet-netifup-vmnet1.pid /dev/vmnet1 vmnet1
root      11651      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-netifup -s 6 -d
/var/run/vmnet-netifup-vmnet8.pid /dev/vmnet8 vmnet8
root      64953      1  0 16:18 ?          00:00:00 vmnet-netifup -d /var/run/vmne
t-netifup-vmnet2.pid /dev/vmnet2 vmnet2
root      64980  64925  0 16:18 pts/6    00:00:00 grep /dev/vmnet
[root@prometheus ~]#
```

I then went and ran the “*vm_subnet_2*” script

```
[root@prometheus ~]# ps -ef | grep vmnet
root      11632      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-bridge -s 6 -d
/var/run/vmnet-bridge-0.pid -n 0
root      11640      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-netifup -s 6 -d
/var/run/vmnet-netifup-vmnet1.pid /dev/vmnet1 vmnet1
root      11645      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-dhcpd -s 6 -cf
/etc/vmware/vmnet1/dhcpd/dhcpd.conf -lf /etc/vmware/vmnet1/dhcpd/dhcpd.leases -p
f /var/run/vmnet-dhcpd-vmnet1.pid vmnet1
root      11649      1  0 14:01 ?          00:00:08 /usr/bin/vmnet-natd -s 6 -m /e
tc/vmware/vmnet8/nat.mac -c /etc/vmware/vmnet8/nat/nat.conf
root      11651      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-netifup -s 6 -d
/var/run/vmnet-netifup-vmnet8.pid /dev/vmnet8 vmnet8
root      11657      1  0 14:01 ?          00:00:00 /usr/bin/vmnet-dhcpd -s 6 -cf
/etc/vmware/vmnet8/dhcpd/dhcpd.conf -lf /etc/vmware/vmnet8/dhcpd/dhcpd.leases -p
f /var/run/vmnet-dhcpd-vmnet8.pid vmnet8
root      64953      1  0 16:18 ?          00:00:00 vmnet-netifup -d /var/run/vmne
t-netifup-vmnet2.pid /dev/vmnet2 vmnet2
root      65163  63842  0 16:19 pts/6    00:00:00 grep --color=auto vmnet
[root@prometheus ~]#
```

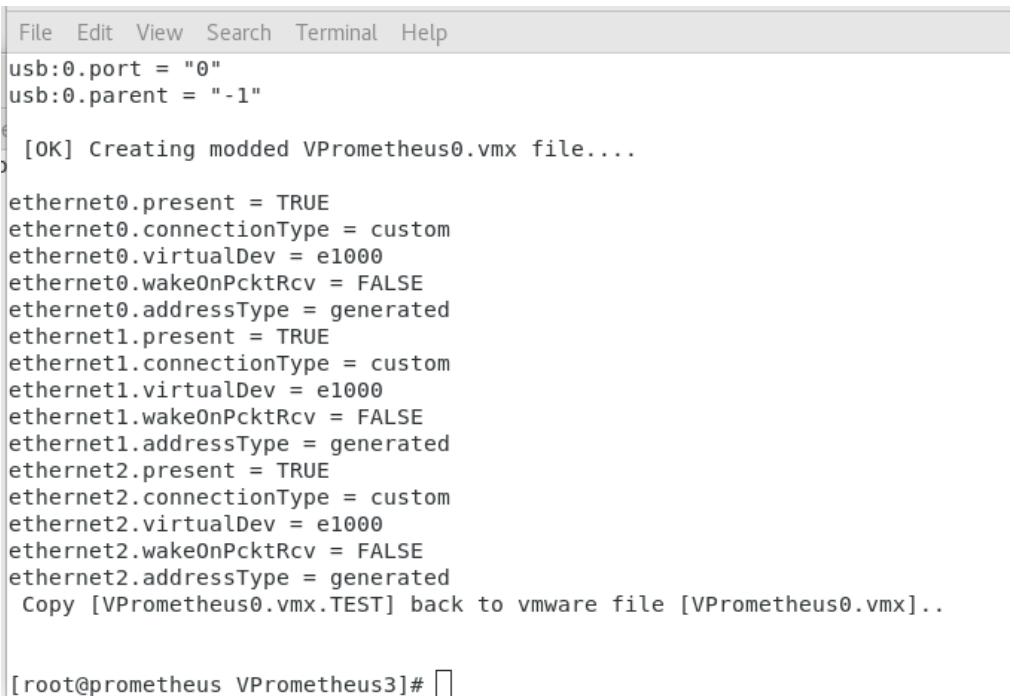
After running the command we can now see that there is a “vmnet2” process running, this will be used later on



I went and downloaded the “fix-vmnet-dev” script

```
[root@prometheus VPrometheus3]# chmod +x fix-vmnet-dev
[root@prometheus VPrometheus3]# ls
fix-vmnet-dev  VPrometheus0-6.scoreboard  VPrometheus0.vmdk
vmware-0.log    VPrometheus0-7.scoreboard  VPrometheus0.vmsd
vmware-1.log    VPrometheus0-8.scoreboard  VPrometheus0.vmx
vmware-2.log    VPrometheus0.nvram        VPrometheus0.vmxsf
vmware.log      VPrometheus0.scoreboard
[root@prometheus VPrometheus3]# 
```

I went and gave the “fix-vmnet-dev” file max permissions, as you can see



```
File Edit View Search Terminal Help
usb:0.port = "0"
usb:0.parent = "-1"

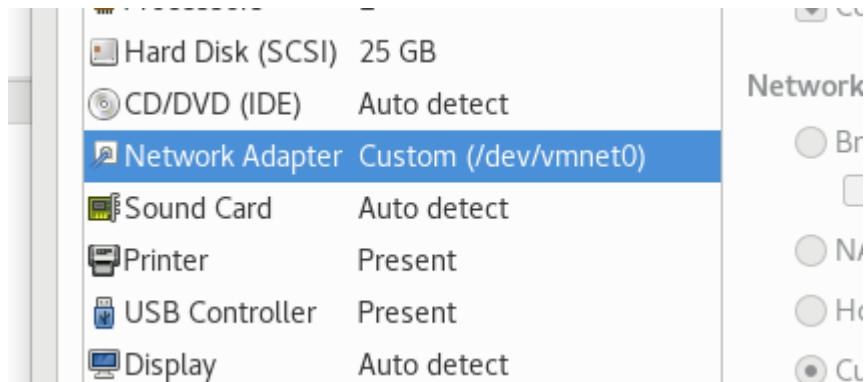
[OK] Creating modded VPrometheus0.vmx file.....

ethernet0.present = TRUE
ethernet0.connectionType = custom
ethernet0.virtualDev = e1000
ethernet0.wakeOnPcktRcv = FALSE
ethernet0.addressType = generated
ethernet1.present = TRUE
ethernet1.connectionType = custom
ethernet1.virtualDev = e1000
ethernet1.wakeOnPcktRcv = FALSE
ethernet1.addressType = generated
ethernet2.present = TRUE
ethernet2.connectionType = custom
ethernet2.virtualDev = e1000
ethernet2.wakeOnPcktRcv = FALSE
ethernet2.addressType = generated
Copy [VPrometheus0.vmx.TEST] back to vmware file [VPrometheus0.vmx]..

[root@prometheus VPrometheus3]# 
```

I went and ran the script using the “./(file name)” command

The result of me running this script is that it goes and rates two more network adapters for me to use, but for vmnet3 we only need one so we will be going and deleting the other Two.

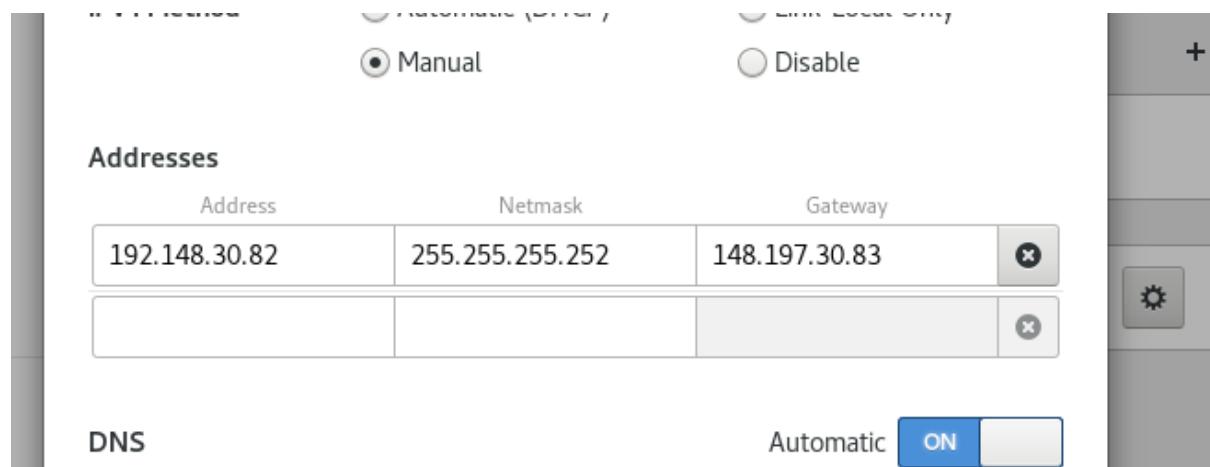


This is the one we did not delete we will be using this later

- Host-only: A private network shared with the host
- Custom: Specific virtual network

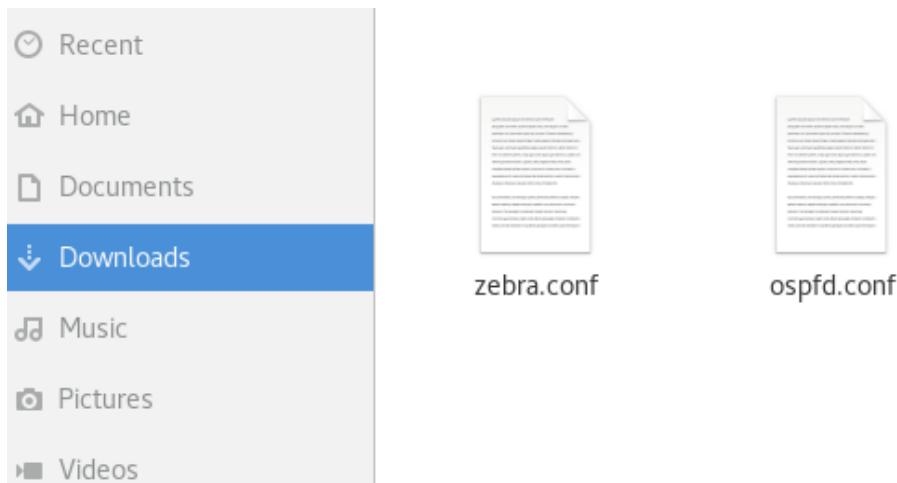
/dev/vmnet2

I have gone and set the custom adapter to “/dev/vmnet2”, after this we will be rebooting the machine



After rebooting the system I went and set the IP address, netmask and gateway.

Final unassisted task for 70% (dynamic routing)



I went and downloaded these two files since we will need them for quagga to allow for dynamic routing

```

File Edit View Search Terminal Help
[root@prometheus ~]# yum install quagga
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirrors.coreix.net
 * epel: ftp.uni-bayreuth.de
 * extras: mirror.as29550.net
 * updates: mirror.as29550.net
Resolving Dependencies
--> Running transaction check
--> Package quagga.x86_64 0:0.99.22.4-5.el7_4 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package           Arch         Version          Repository      Size
=====
Installing:
quagga           x86_64       0.99.22.4-5.el7_4   base           1.2 M

Transaction Summary
=====
Install 1 Package

```

Since we will be using Quagga, a network routing application that implements the Open shortest path first, routing information protocol, and border gateway protocols, I installed it.

```

interface ens33
multicast

interface vmnet0
multicast

interface vmnet1
multicast

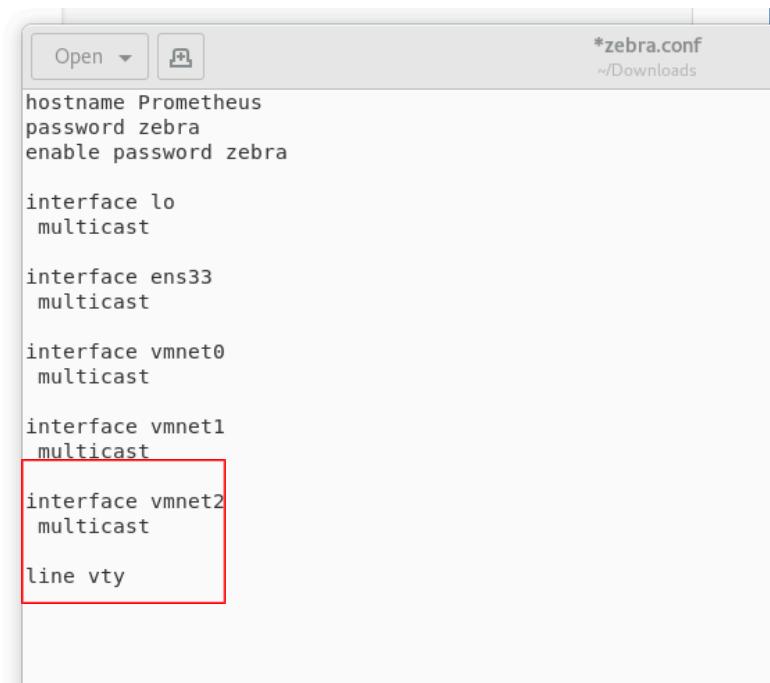
#interface vmnet2
# multicast

line vty

```

[VMware L](#)
[Referenc](#)

This is inside the default version of the zebra.conf file we will be going and changing one aspect of the file, shown below, we have removed the hashtags on “interface vmnet2” and “multicast”



```

Open ▾  *zebra.conf
~/Downloads

hostname Prometheus
password zebra
enable password zebra

interface lo
multicast

interface ens33
multicast

interface vmnet0
multicast

interface vmnet1
multicast
interface vmnet2
multicast

line vty

```

The new version of the “zebra.conf” file, We must make sure this meets the standards of our machine

```

hostname Prometheus
password zebra

router ospf
  ospf router-id 148.197.28.111
  network 148.197.28.0/24 area 0
  network 148.197.30.252/30 area 0
  network 148.197.31.252/30 area 0
#network 148.197.32.252/30 area 0

log stdout

line vty
  -f14.

```

This is the default version of the “ospfd.conf” file and we will be going and making a few changes to this file, they will be shown below marked in red.

```

hostname Prometheus
password zebra

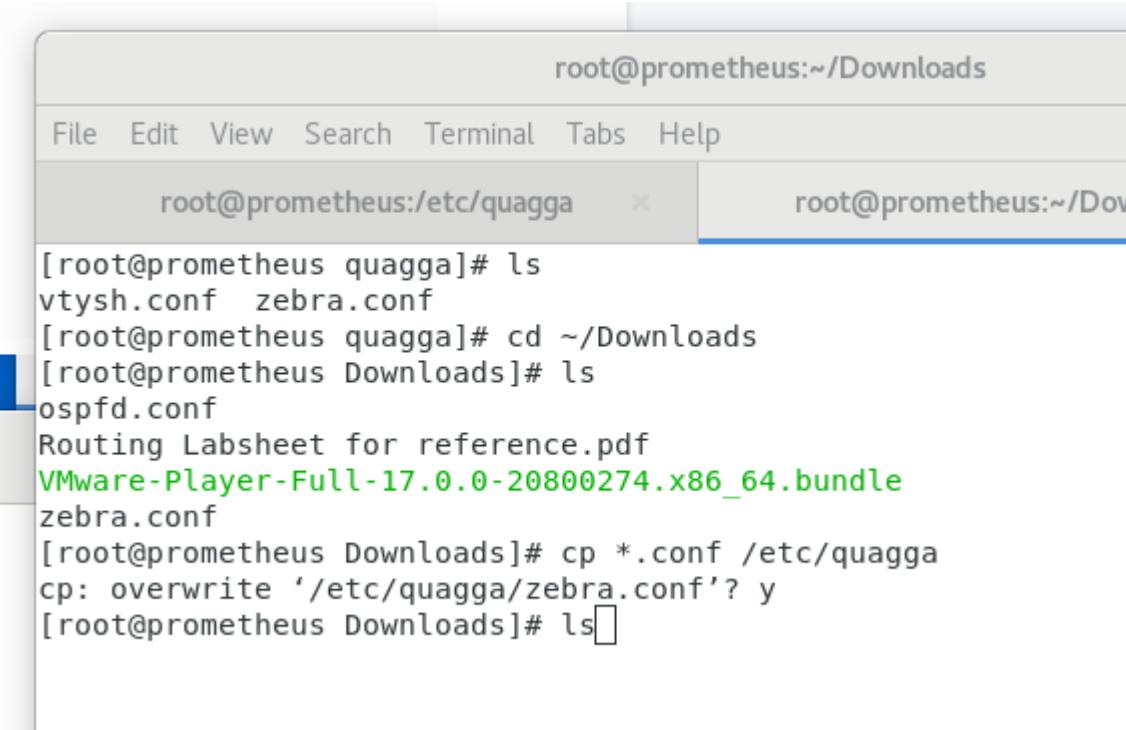
router ospf
  ospf router-id 148.197.28.209
  network 148.197.28.0/24 area 0
  network 148.197.30.80/30 area 0
  network 148.197.31.252/30 area 0
#network 148.197.32.252/30 area 0

log stdout

line vty

```

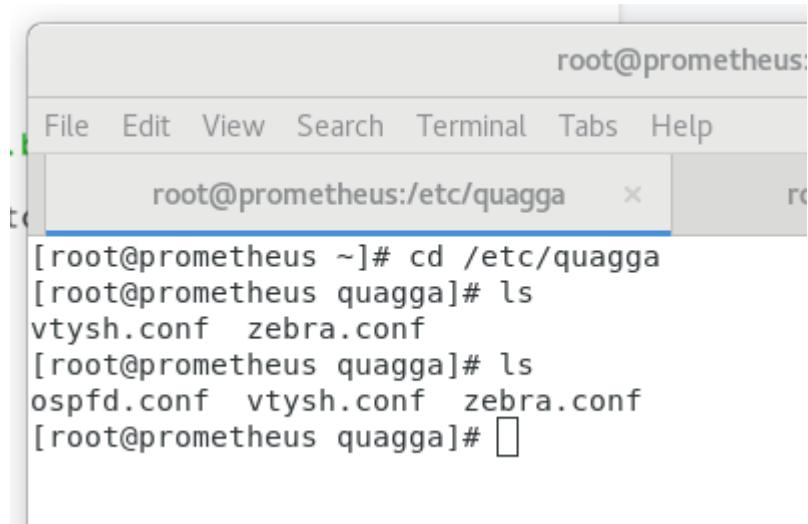
These are the changes made, these meet the standards of my machine



The screenshot shows a terminal window with two tabs. The active tab is titled 'root@prometheus:~/Downloads' and contains the following command-line session:

```
[root@prometheus quagga]# ls
vtysh.conf zebra.conf
[root@prometheus quagga]# cd ~/Downloads
[root@prometheus Downloads]# ls
ospfd.conf
Routing Labsheet for reference.pdf
VMware-Player-Full-17.0.0-20800274.x86_64.bundle
zebra.conf
[root@prometheus Downloads]# cp *.conf /etc/quagga
cp: overwrite '/etc/quagga/zebra.conf'? y
[root@prometheus Downloads]# ls
```

I went and moved the files to the directory “/etc/quagga”, shown above and below



The screenshot shows a terminal window with two tabs. The active tab is titled 'root@prometheus:' and contains the following command-line session:

```
[root@prometheus ~]# cd /etc/quagga
[root@prometheus quagga]# ls
vtysh.conf zebra.conf
[root@prometheus quagga]# ls
ospfd.conf vtysh.conf zebra.conf
[root@prometheus quagga]#
```

These are the files now in that directory

```
Routing Labsheet tor reference.pdf
VMware-Player-Full-17.0.0-20800274.x86_64.bundle
zebra.conf
[root@prometheus Downloads]# gedit ospfd.conf
[root@prometheus Downloads]# iptables -F
[root@prometheus Downloads]# /sbin/sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
[root@prometheus Downloads]#
```

After transferring all the files I went and deactivated the firewall and activated activated port forwarding, using the commands marked in red

```
root@prometheus:~/Downloads
File Edit View Search Terminal Tabs Help
root@prometheus:~/... × root@prometheus:~/... × root@prometheus:~/...
[root@prometheus Downloads]# zebra -f /etc/quagga/zebra.conf
```

This command goes and activates zebra

```
root@prometheus:~/Downloads
File Edit View Search Terminal Tabs Help
root@prometheus:~/... × root@prometheus:~/... × root@prometheus:~/...
[root@prometheus Downloads]# ospfd -f /etc/quagga/ospfd.conf
2023/02/06 17:04:00 OSPF: OSPFd 0.99.22.4 starting: vty@2604
2023/02/06 17:04:00 OSPF: interface 148.197.28.209 [2] join AllSPFRoute
ast group.
2023/02/06 17:04:00 OSPF: interface 148.197.30.81 [9] join AllSPFRouter
st group.
```

This command goes and activates ospfd

```

root@prometheus:~/Downloads
File Edit View Search Terminal Tabs Help
root@prometheus:~/... root@prometheus:~/... root@prometheus:~/...
Every 2.0s: route -n Mon Feb 6 17:06:16 2023

Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
0.0.0.0         148.197.28.254  0.0.0.0        UG    100    0        0 ens33
148.197.27.0   148.197.28.115  255.255.255.0  UG    20     0        0 ens33
148.197.28.0   0.0.0.0        255.255.255.0  U     100    0        0 ens33
148.197.30.80  0.0.0.0        255.255.255.252 U     0      0        0 vmnet2
148.197.30.92  148.197.28.178  255.255.255.252 UG    20     0        0 ens33
148.197.30.128 148.197.28.217  255.255.255.252 UG    20     0        0 ens33
172.16.147.0   0.0.0.0        255.255.255.0  U     0      0        0 vmnet8
192.168.11.0   0.0.0.0        255.255.255.0  U     0      0        0 vmnet1

```

This command goes and watches the kernels routing table

```

[root@vprometheus3 ~]# ping google.com
ping: google.com: Name or service not known
[root@vprometheus3 ~]# ping google.com
ping: google.com: Name or service not known
[root@vprometheus3 ~]# ping google.com
PING google.com (142.250.180.14) 56(84) bytes of data.
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=1 ttl=115 time=5.39 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=2 ttl=115 time=6.78 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=3 ttl=115 time=6.33 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=4 ttl=115 time=5.19 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=5 ttl=115 time=5.61 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=6 ttl=115 time=5.99 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=7 ttl=115 time=5.92 ms

```

This is me being able to ping google on VP3, which was only able to ping VP2 before, I now have access to the internet since I am routing my connection from my host

Cancel **Wired** Apply

Details Identity **IPv4** IPv6 Security

IPv4 Method

Automatic (DHCP) Link-Local Only
 Manual Disable

Addresses

Address	Netmask	Gateway	X
148.197.30.82	255.255.255.252	148.197.30.81	X
			X

DNS Automatic **ON**

148.197.28.1, 1.1.1.1

Separate IP addresses with commas

Routes Automatic **ON**

Address	Netmask	Gateway	Metric	X
				X

These were the final wired network settings I had after doing the full setup for dynamic routing

Lab Sheet 2 - Virtual box

Virtual box is a form of open-source virtualisation software that is used by the x86 computing architecture. It is one of the biggest hypervisors that can be used when making virtual machines. In this part of my log book I will be demonstrating how to create a virtual network using this software.

Installation

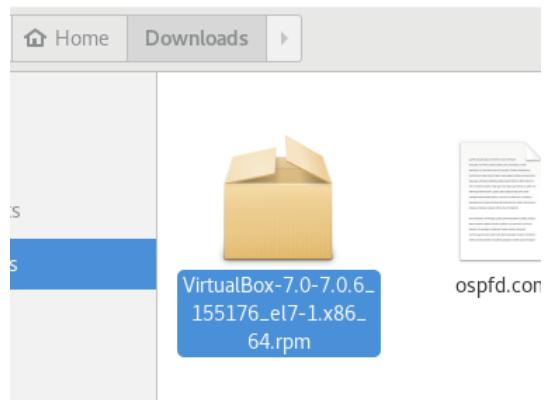
The website front page where we can install the software from, make sure to download the correct application corresponding to your operating system

To start off this lab sheet I will begin with installing a virtual box to my host system. I will be going to the Virtual box website (<https://www.virtualbox.org/wiki/Downloads>) and installing the latest version of it, retrospective to your operating system.



The screenshot shows the official Oracle VirtualBox download page for Linux hosts. The page features a large blue "VirtualBox" logo at the top right. To the left of the logo is a small icon of a blue cube with a white "M" on it, labeled "ORACLE" and "VirtualBox". On the far right are links for "search...", "Login", and "Preferences". Below the logo, the text "Download VirtualBox for Linux Hosts" is displayed. A note states: "Note: The package architecture has to match the Linux kernel architecture, that is, if you are running a 64-bit kernel, install the appropriate AMD64 package (it does not matter if you have an Intel or an AMD CPU). Mixed installations (e.g. Debian/Lenny) ships an AMD64 kernel with 32-bit packages) are not supported. To install VirtualBox anyway you need to setup a 64-bit chroot environment." Below this note, it says: "The VirtualBox base package binaries are released under the terms of the [GPL version 2](#)". A section titled "Please choose the appropriate package for your Linux distribution." follows. Under "VirtualBox 7.0.6 for Linux", there is a list of supported distributions with links to their respective package pages. The list includes: Oracle Linux 9 / Red Hat Enterprise Linux 9, Oracle Linux 8 / Red Hat Enterprise Linux 8, Oracle Linux 7 / Red Hat Enterprise Linux 7 / CentOS 7, Ubuntu 22.04, Ubuntu 20.04, Ubuntu 18.04 / 18.10 / 19.04, Debian 11, Debian 10, openSUSE 15.3 / 15.4, Fedora 36, Fedora 35, and All distributions (built on EL6 and therefore not requiring recent system libraries). A note below the distribution list says: "You might want to compare the checksums to verify the integrity of downloaded packages. The SHA256 checksums should be favored as the MD5 algorithm must be treated as insecure!" A bullet point lists "• [SHA256 checksums](#), [MD5 checksums](#)". A section titled "Oracle Linux" is present, with a note for users of Oracle Linux 6, 7, and 8 regarding yum repository configuration. The page has a sidebar on the left with links to "About", "Screenshots", "Downloads", "Documentation", "End-user docs", "Technical docs", "Contribute", and "Community".

Due to this being a linux distribution “centOS” we will be needing the RHEL7 (Red Hat Enterprise Linux 7) version to allow it to work on this Operating system



This is the executable file needed to install virtual box on my system. Since this is a .rpm we will be using the yum command in the terminal to install it

```
VMWARE-PLAYER-FULL-1.0.0-20800274.X86_64.BUNDLE
zebra.conf
[root@prometheus Downloads]# yum localinstall VirtualBox-7.0-7.0.6_155176_el7-1.
x86_64.rpm
Loaded plugins: fastestmirror, langpacks
Examining VirtualBox-7.0-7.0.6_155176_el7-1.x86_64.rpm: VirtualBox-7.0-7.0.6_155
176_el7-1.x86_64
```

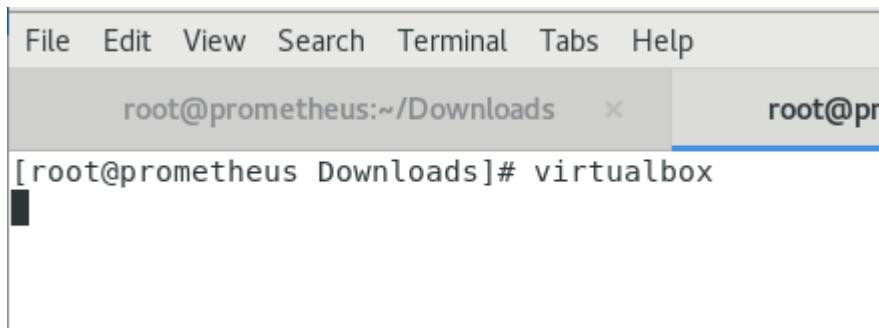
In this image you can see we used the “yum localinstall” to install the RPM file. The reason we used localinstall is because this searches the package in the working directory of the terminal. /home by default and this solves any issues if there is any with the folder it is located and then install them. ((why not use rpm -i install))

```
=====
Install 1 Package

Total size: 208 M
Installed size: 208 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : VirtualBox-7.0-7.0.6_155176_el7-1.x86_64          1/1
Creating group 'vboxusers'. VM users must be member of that group!
  Verifying  : VirtualBox-7.0-7.0.6_155176_el7-1.x86_64          1/1
Installed:
  VirtualBox-7.0.x86_64 0:7.0.6_155176_el7-1
Complete!
[root@prometheus Downloads]# 
```

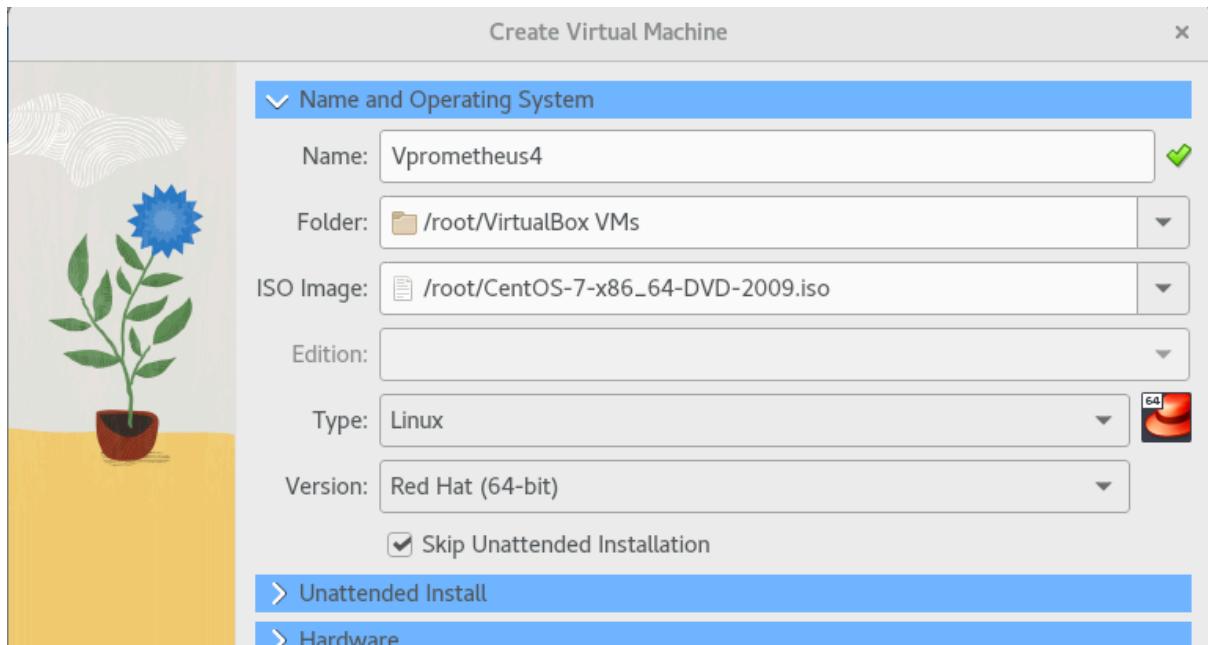
This is the finished installation of the virtual box.

Building a virtual machine

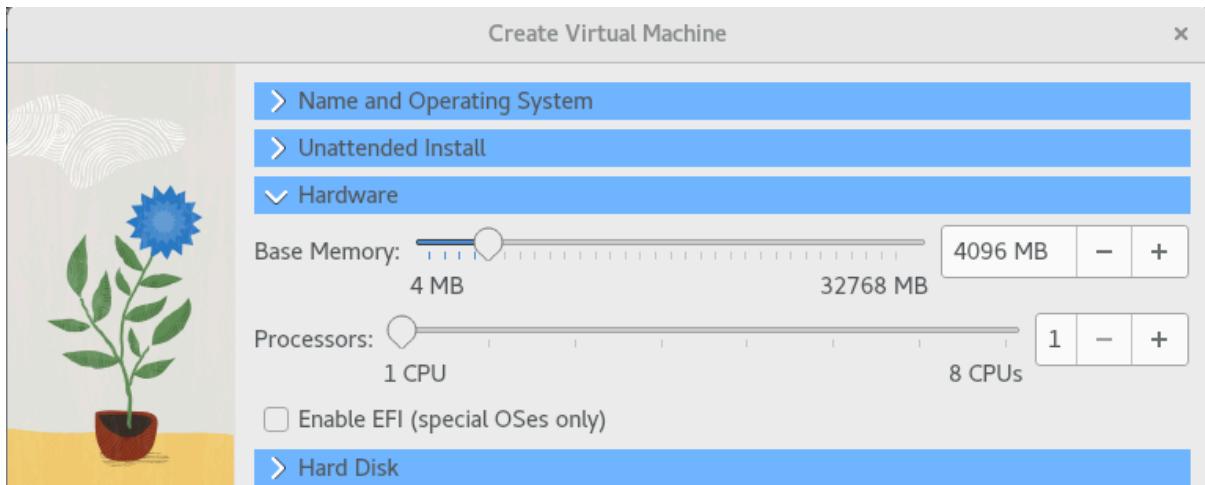


A screenshot of a terminal window. The window title bar says "File Edit View Search Terminal Tabs Help". The top status bar shows "root@prometheus:~/Downloads" and "root@pr". The main terminal area has a grey background and contains the command "[root@prometheus Downloads]# virtualbox". A small black progress bar is visible at the bottom left of the terminal window.

We go and type “virtualbox” in the terminal to invoke the application



In this image I have named the VM “Vprometheus4” and given it the ISO image for “centOS” since this is the operating system we will be using for this project. I have also ticked “Skip unattached Installation” <- search why



I have gone and given the system 4gb of ram from its starting 2gb.

The screenshot shows the 'Hard Disk' configuration dialog. It has a radio button for 'Create a Virtual Hard Disk Now' selected. Under 'Hard Disk File Location and Size', the file path is set to '/root/VirtualBox VMs/Vprometheus4/Vprometheus4.vdi'. The size slider is set to 20.00 GB, with 4.00 MB at the left end and 2.00 TB at the right end. Under 'Hard Disk File Type and Variant', the type is set to 'VDI (VirtualBox Disk Image)'. There are two unchecked checkboxes: 'Pre-allocate Full Size' and 'Split into 2GB parts'.

I have gone and allocated 20gb of storage to the system.

(Research and write a few lines about each of the different types of hard disks available here)

VirtualBox disk image: This is the accepted practice for the hard disc that VirtualBox can use. This file format can hold all the different forms of data that can be connected to a hard disc. You can alter this file type's layout, format, and size.

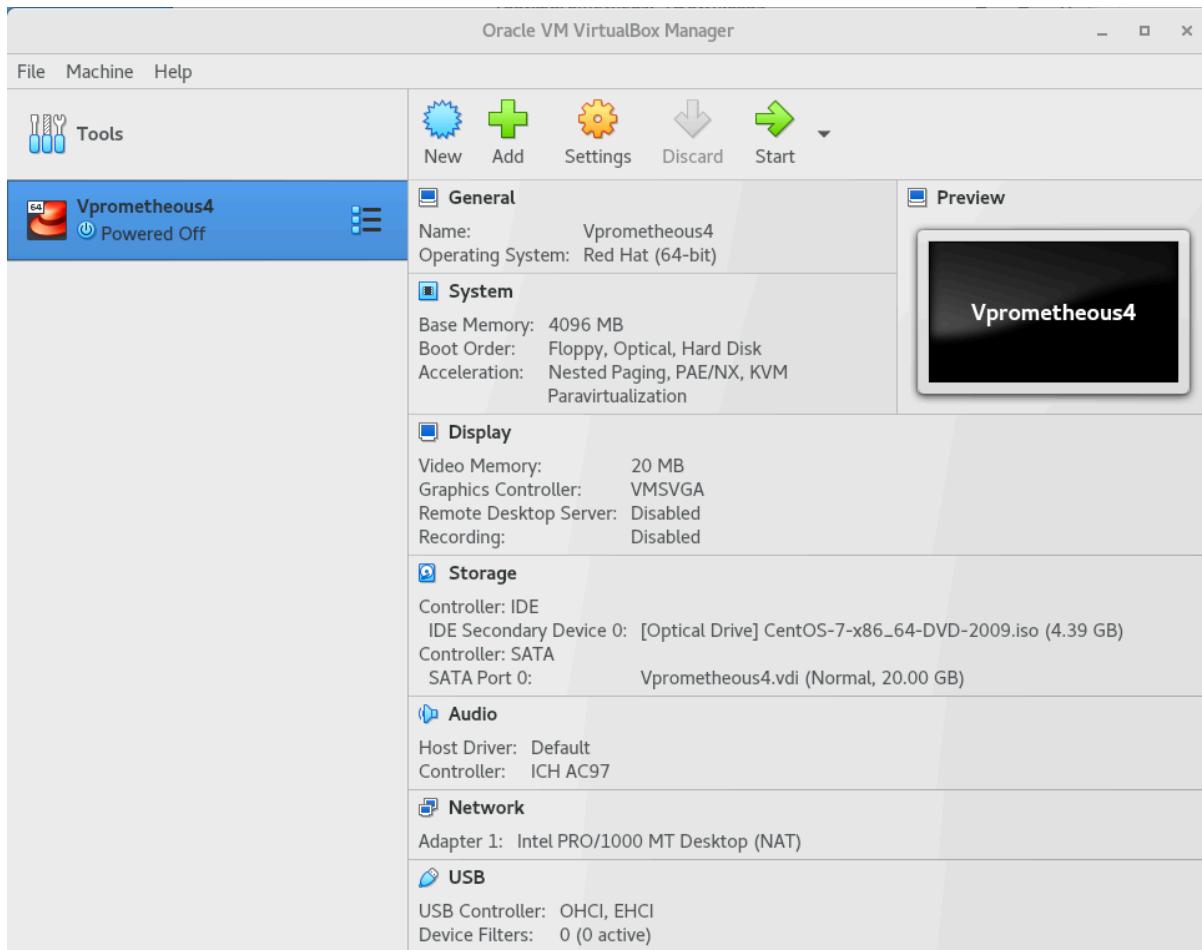
Virtual machine disk: Cloning of physical hard drives and the production of virtual machine off-host backups are made possible by the disc image format VMDK. Both dynamic and fixed files are permitted in this type of disc image. Additionally, it allows for the construction of tiny backups of minute adjustments, which substantially accelerates the backup process compared to other disc kinds.

VirtualBox hard disk: The most famous picture format for modern virtualized products is VHD. It looks like an actual disc. The size capacity needs to be specified when it is formed. They enable quick creation and come with pre-set setups. These image types provide straightforward backup and restoration, allowing you to instantly undo any activity that might have injured your system.

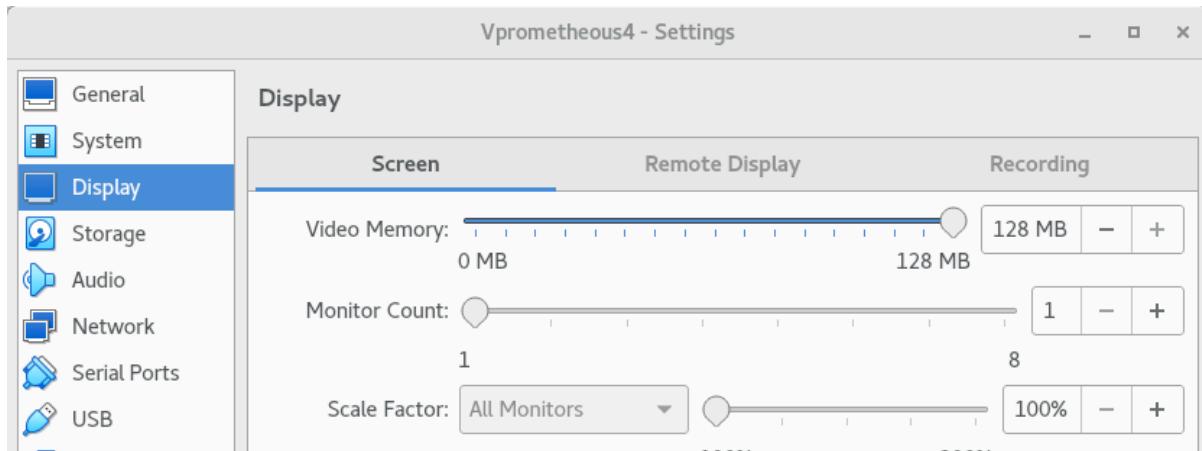
Parallels hard Disk: The data on the system is stored in a file type that mimics a hard drive. The main emphasis is on parallel desktops. This mainly pertains to MacOS, allowing them to run Windows as a virtual computer.

QEMU Copy-On-Write: This file type is typically stored in clusters with copy-on-write enabled. Using this file type makes it possible to keep several unique components of a file at various locations on the disc and delay their appearance until the system requires them.

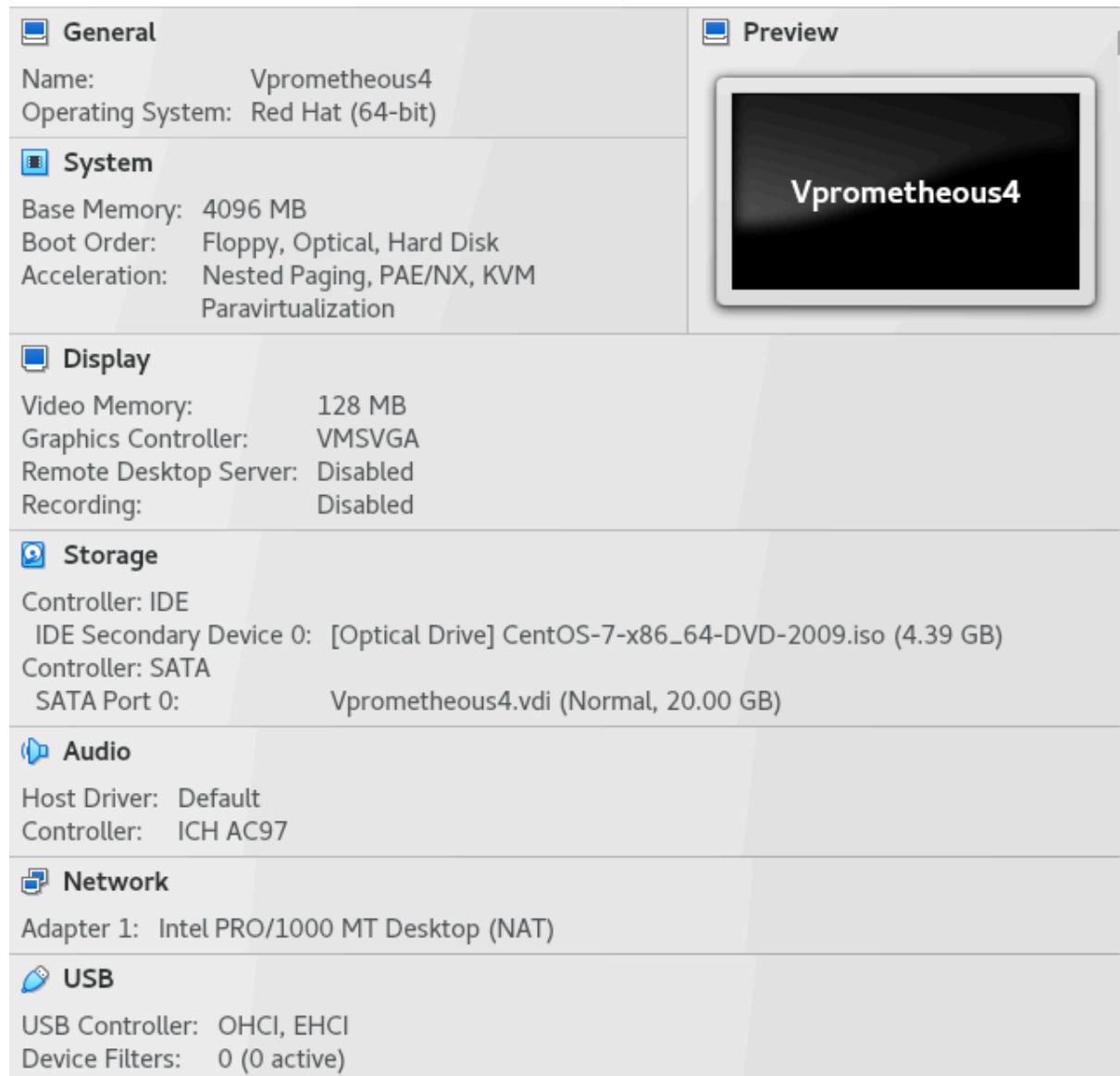
QEMU enhanced disk: This file type—also known as "qcow2"—is similar to QEMU copy-on-write but better for improved performance and data integrity.



This is the new VM that I have created, and you can see all the applied settings on the right of the figure.



In this image I have gone and pushed the video memory from its starting of 20MB to the maxim that is allowed of 128MB



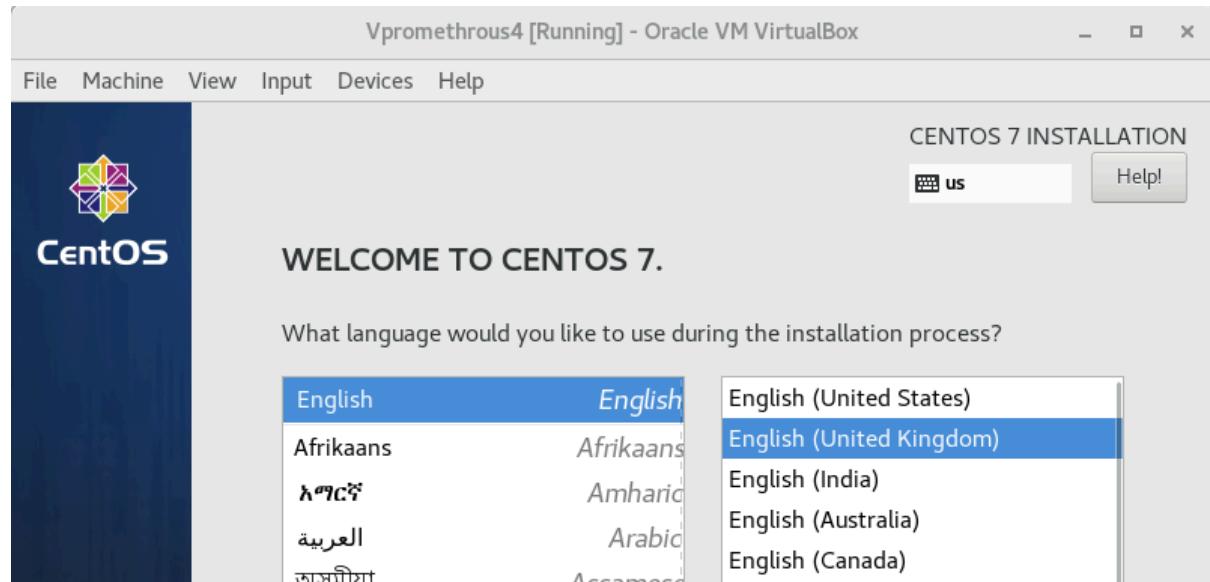
The settings of the virtual machine

```

Vpromethrous4 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
[ 0.000000] Warning: Intel Processor - this hardware has not undergone upstream testing. Please consult http://wiki.centos.org/FAQ for more information
[ 7.260614] dracut-pre-udev[328]: modprobe: ERROR: could not insert 'floppy':
No such device
Mounting Configuration File System...
[ OK ] Started Show Plymouth Boot Screen.
[ OK ] Reached target Paths.
[ OK ] Started Forward Password Requests to Plymouth Directory Watch.
[ OK ] Reached target Basic System.
[ OK ] Mounted Configuration File System.
[ OK ] Started Device-Mapper Multipath Device Controller.
Starting Open-iSCSI...
[ OK ] Started Open-iSCSI.
Starting dracut initqueue hook...
[ 11.132575] [drm:vmw_host_log [vmwgfx]] *ERROR* Failed to send host log message.
[ 11.136449] [drm:vmw_host_log [vmwgfx]] *ERROR* Failed to send host log message.
[ 12.088106] dracut-initqueue[707]: mount: /dev/sr0 is write-protected, mounting read-only
Mounting Configuration File System...
[ OK ] Started Show Plymouth Boot Screen.
[ OK ] Reached target Paths.
[ OK ] Started Forward Password Requests to Plymouth Directory Watch.
[ OK ] Reached target Basic System.
[ OK ] Mounted Configuration File System.
[ OK ] Started Device-Mapper Multipath Device Controller.
Starting Open-iSCSI...
[ OK ] Started Open-iSCSI.
Starting dracut initqueue hook...
[ 12.088106] dracut-initqueue[707]: mount: /dev/sr0 is write-protected, mounting read-only
[ OK ] Created slice system-checkisomd5.slice.
Starting Media check on /dev/sr0...
/dev/sr0: 10dc6ce3d0098cb5c67356cca02b785d
Fragment sums: 81cf91b42222bd851b8b4bf68cda818796ae4e9c332b739175158d555f85
Fragment count: 20
Press [Esc] to abort check.
Checking: 032.3%_

```

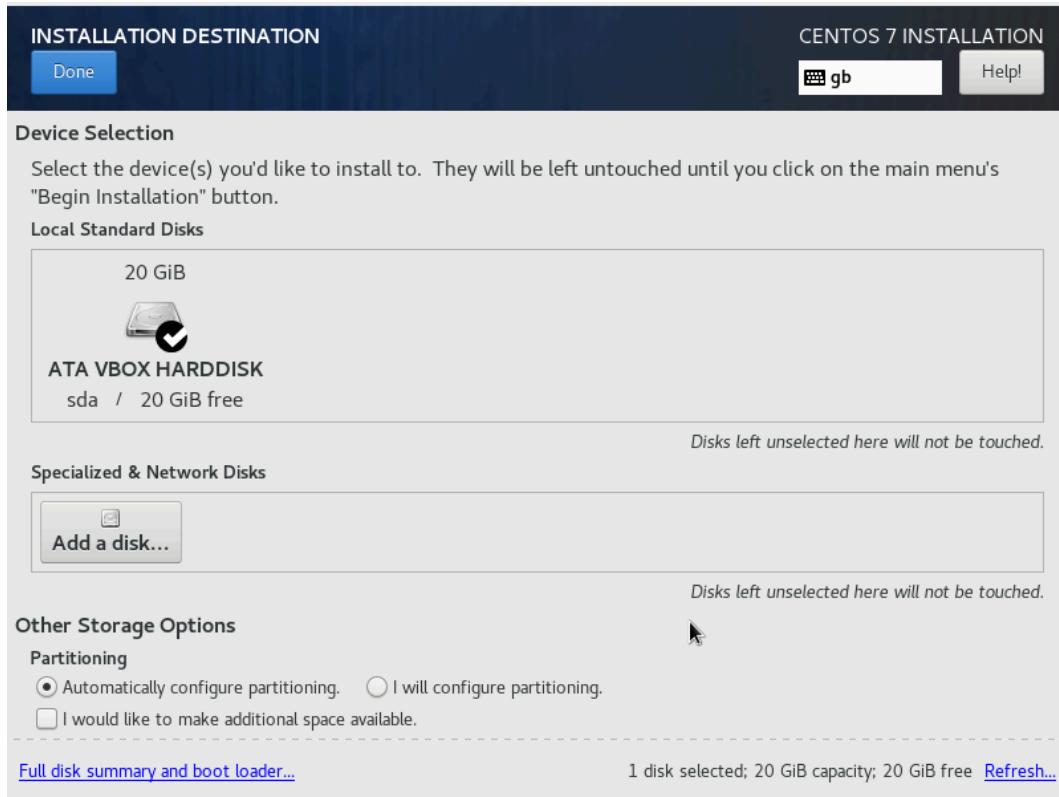
The beginning of the installation of CentOS on virtual box, this has launched the installation in Gnome mode.



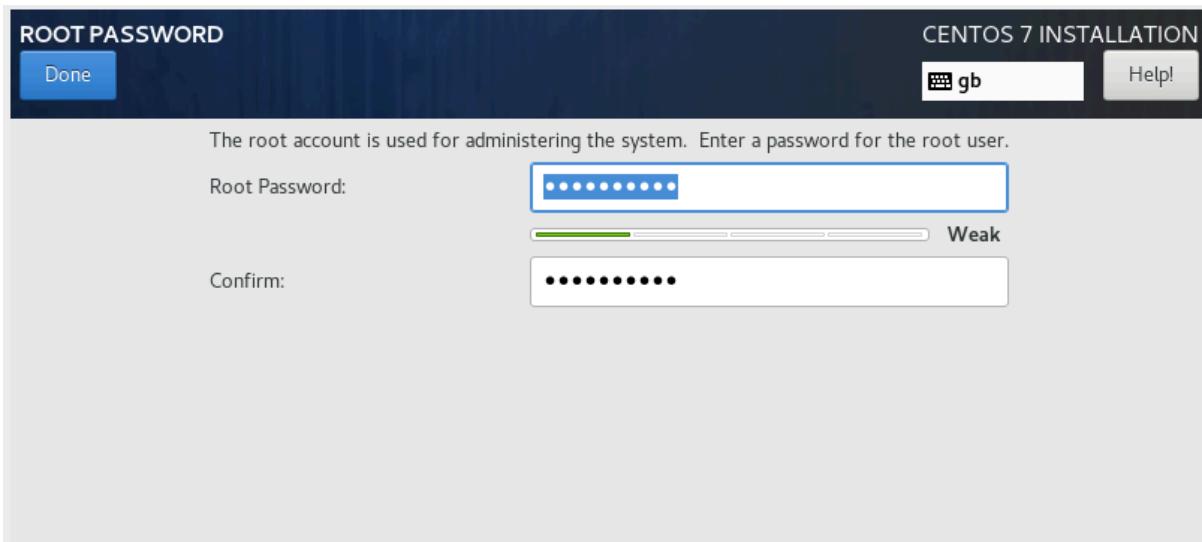
I have gone and changed the keyboard language to allow me to use all the special Characters on my keyboard when typing in the VM



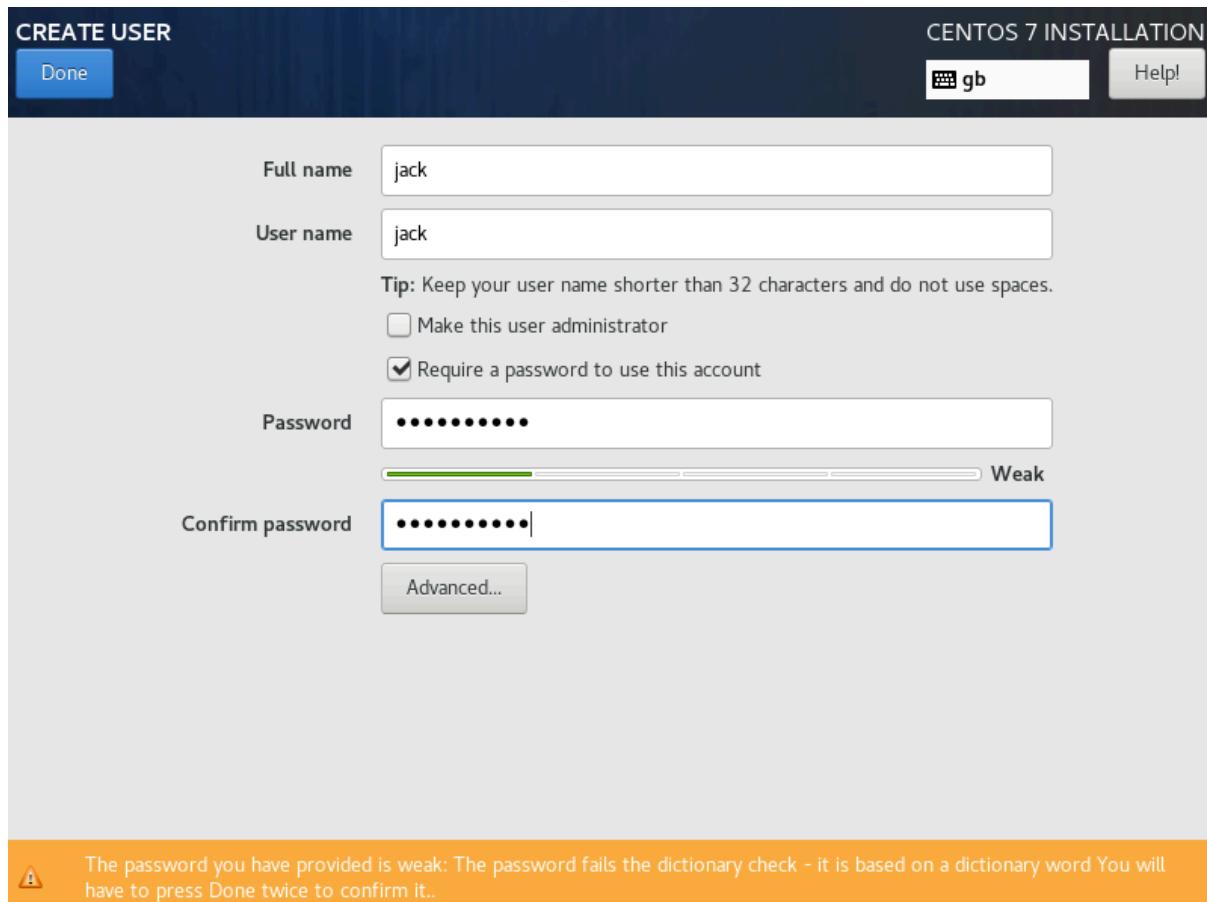
I have gone and given the vm a gnome desktop. This is a graphical user interface and a set of provided applications for the user of the operating system



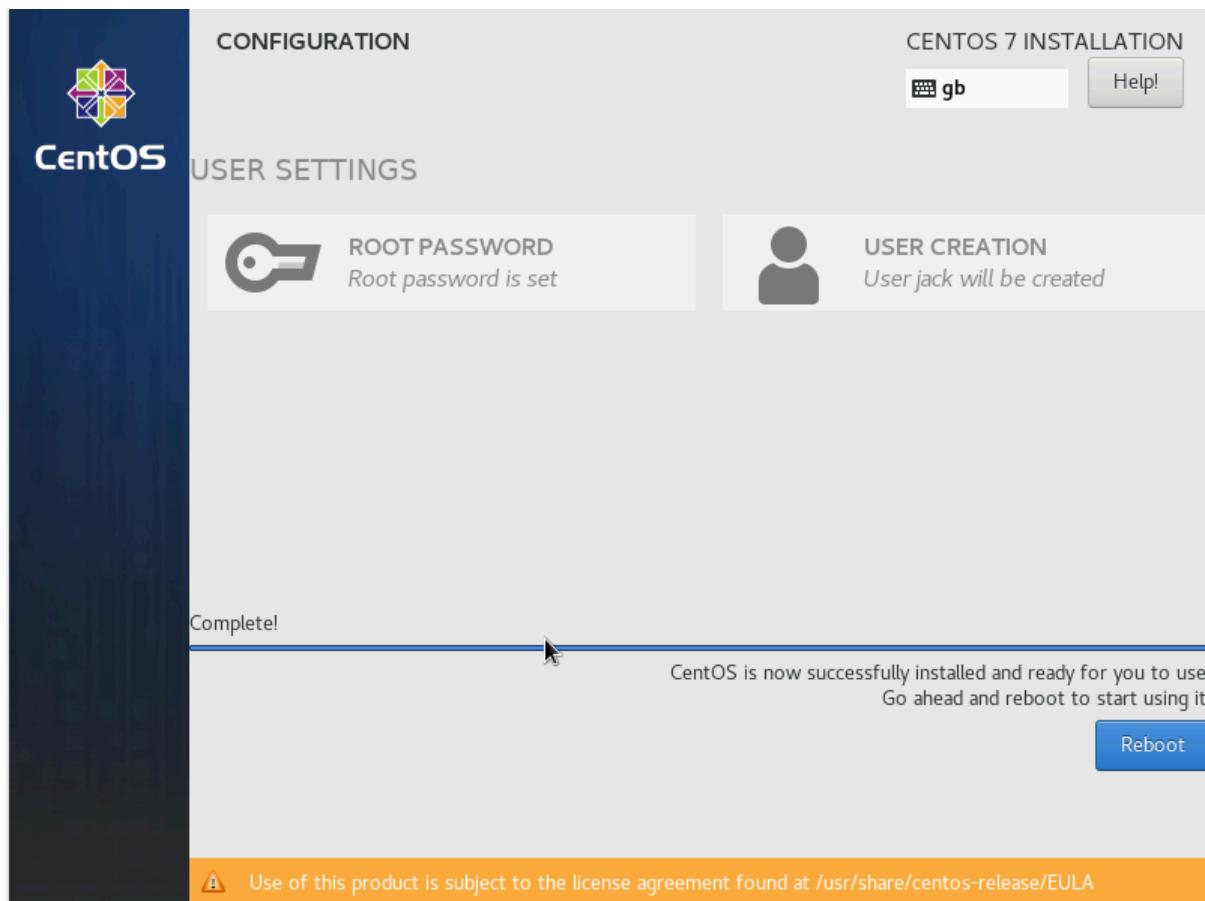
I had to go and allocate where the installation will take place so I can actually begin the installation of the virtual machines operating system, this is the default partition.



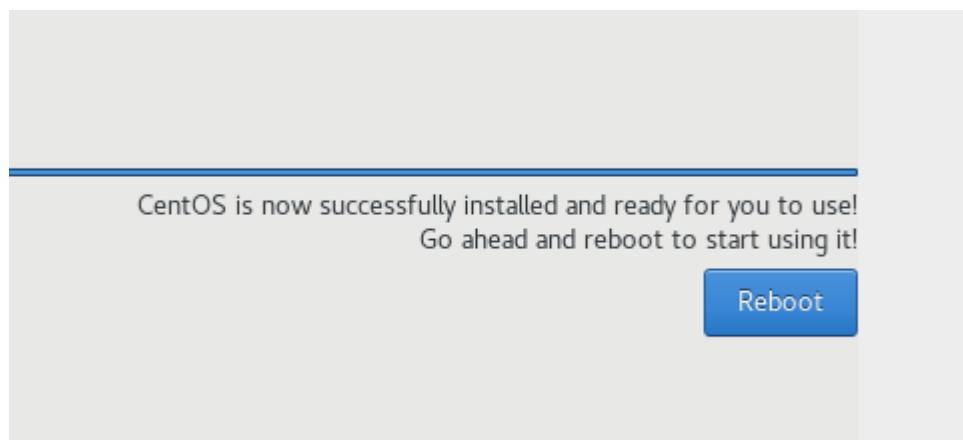
I have gone and allocated the root passwords to “localhost1” since the virtual machine has asked me to give it one since there is no automatic password.



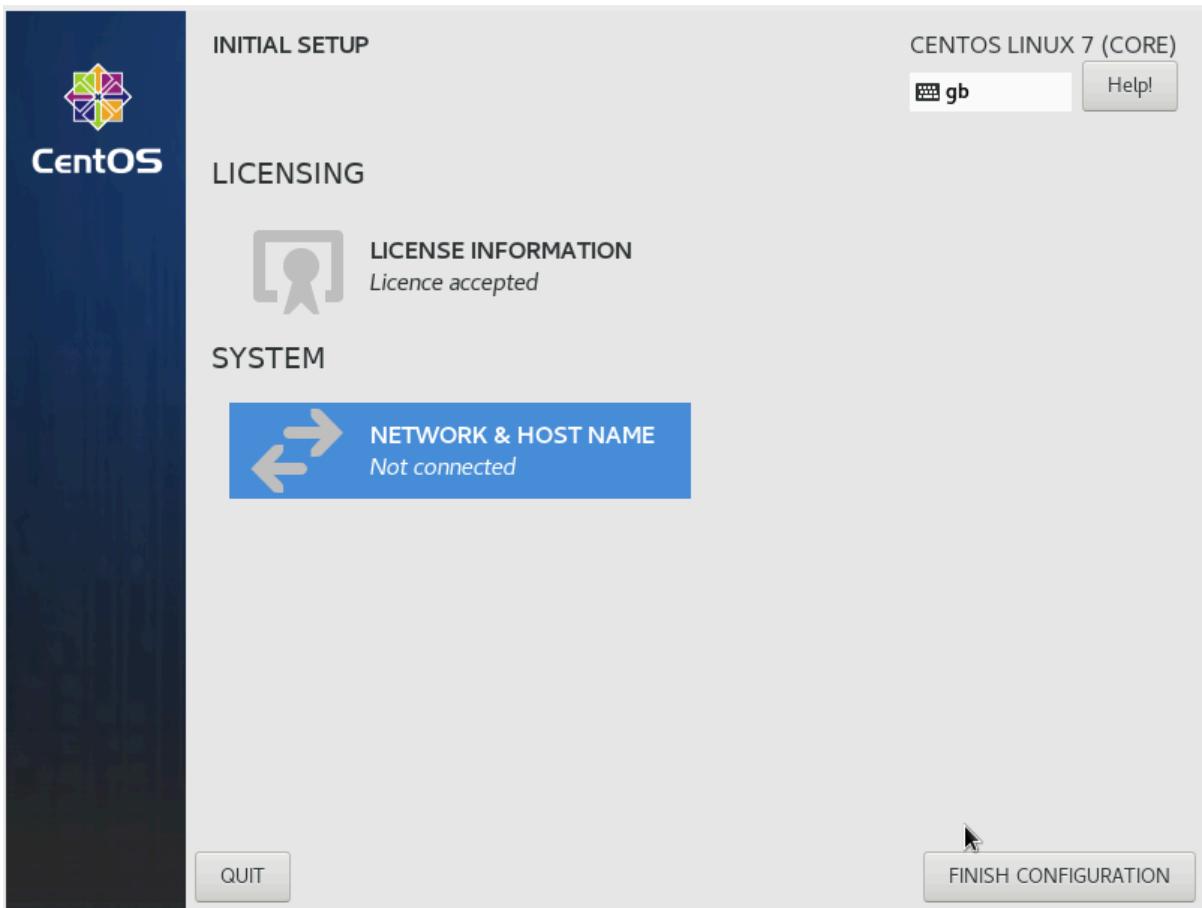
I have also gone and made an account on the system. But this account will never be used/opened; it is there just as a placeholder and will only be visual. We will be mainly using the root this is because it will allow me to always have access to root commands.



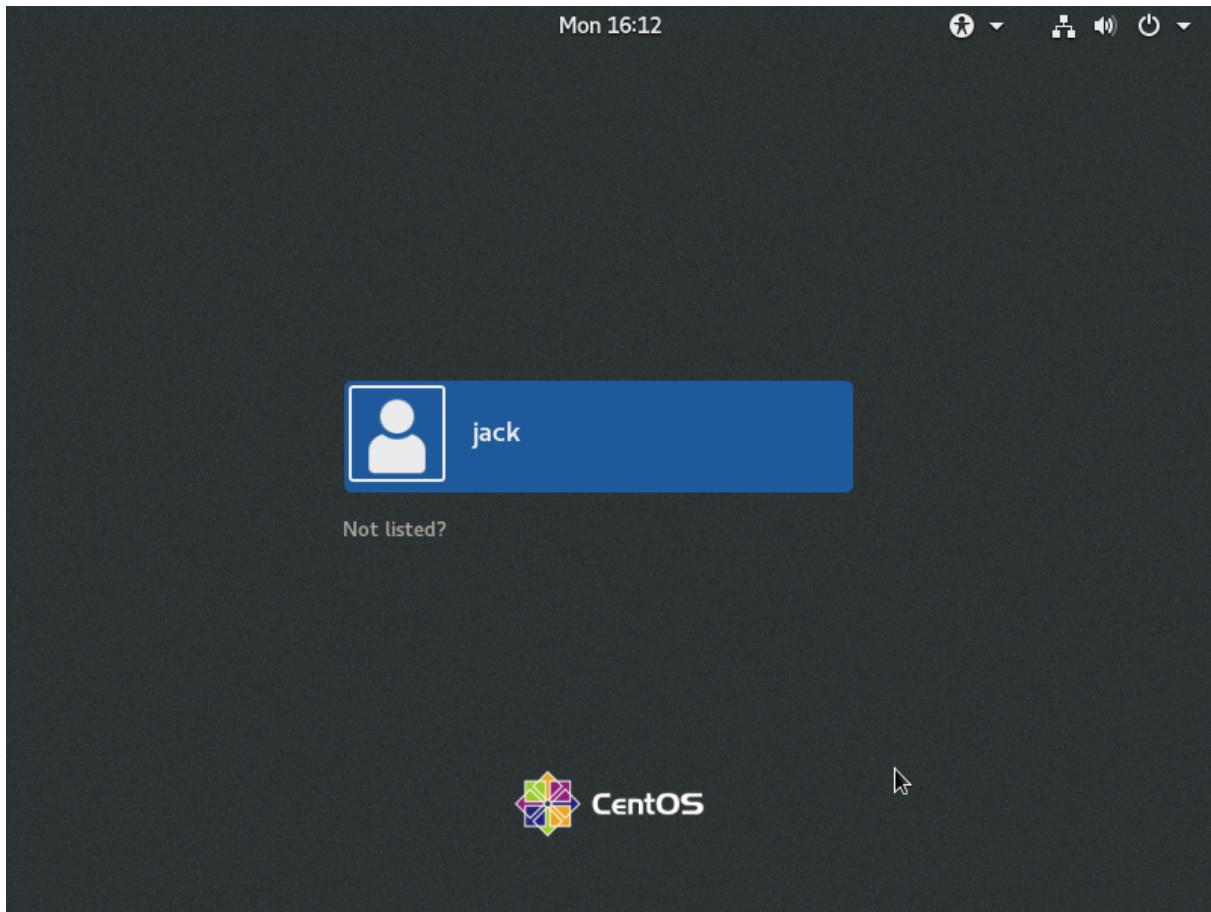
This shows both of my criteria for my created account and root.



Rebooting the VM since it has finished the installation of the VM. This will now allow me to access the actual system and we can begin the process of creating the network.



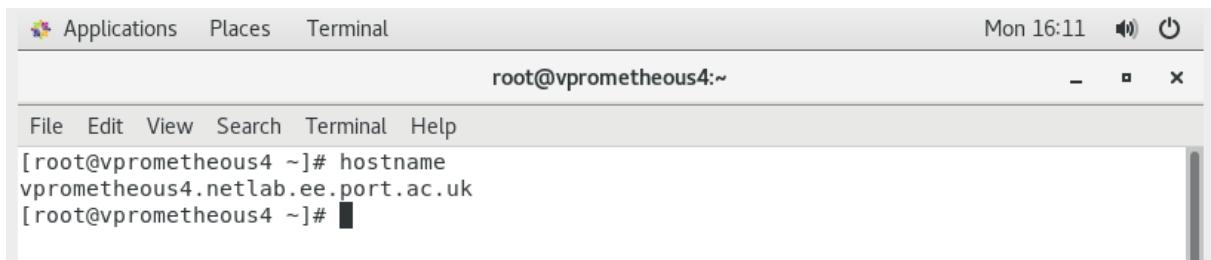
I have gone and accepted the listening and turned on the ethernet connection, letting me gain access to the network. In this section make sure to change the hostname, i forgot and will be doing it using the terminal hostname command



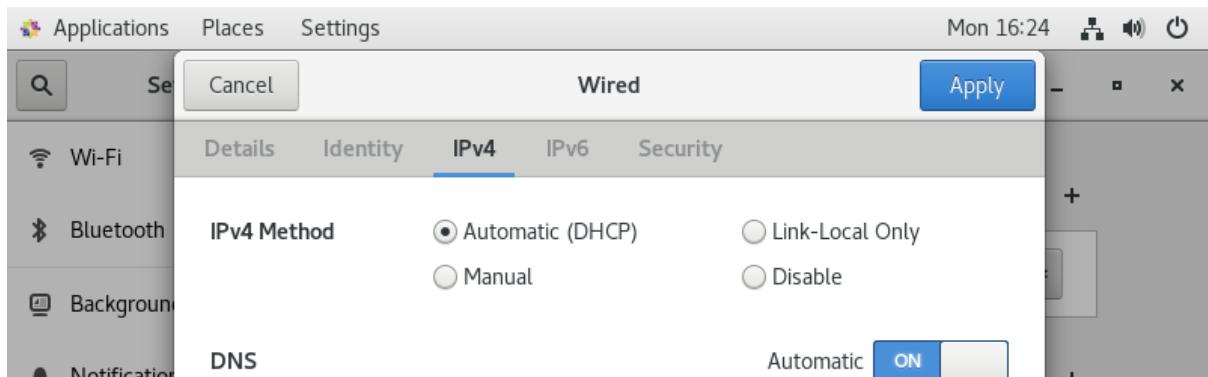
This is the login page of my VM. We will be going and logging into the root account

```
[root@w0v76ef3d80f931 ~]# hostnamectl set-hostname vprometheus4
[root@w0v76ef3d80f931 ~]# hostname
vprometheus4
[root@w0v76ef3d80f931 ~]#
```

Due to me forgetting to change the hostname before I had gone and used this command to allow me to change the hostname, this was not the final hostname that is demonstrated in the figure below.



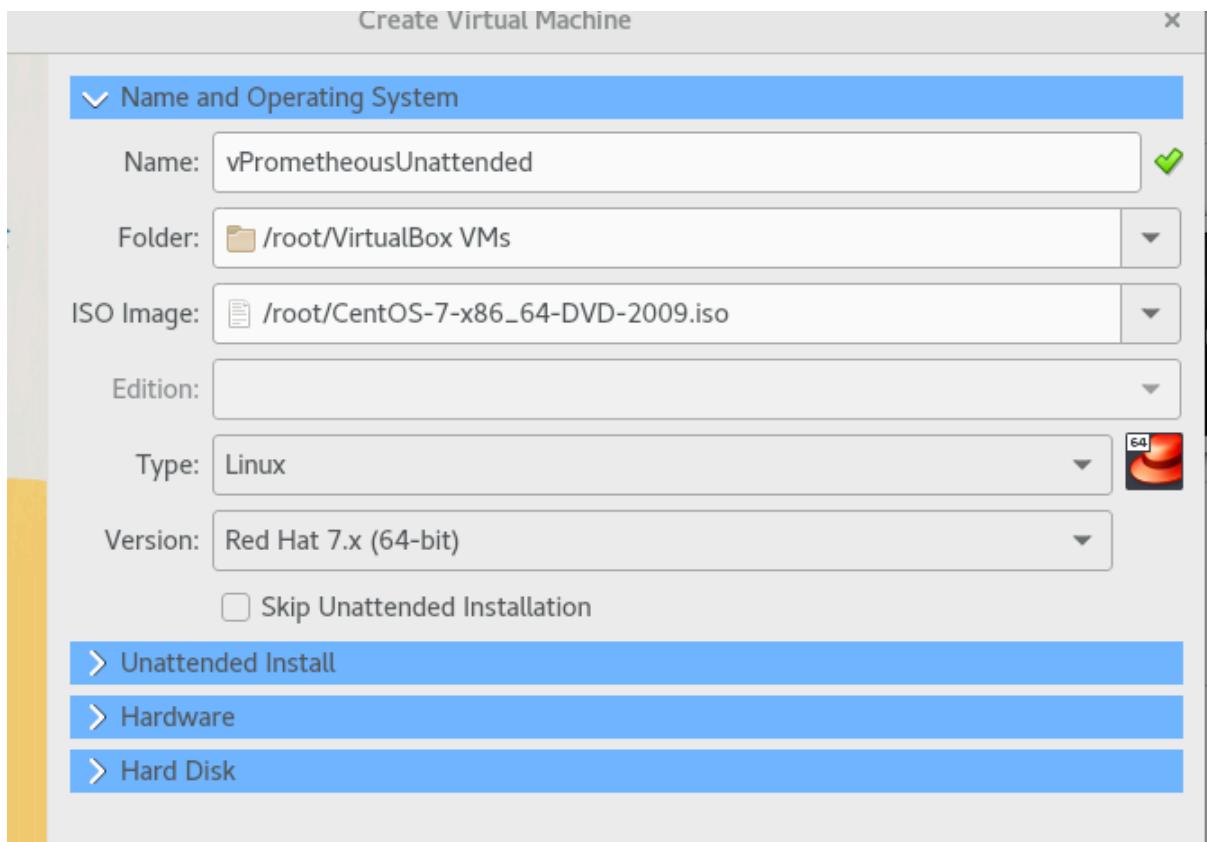
This is the actual hostname needed.



I went into the wired connection settings and set it to automatic DHCP then clicked apply, since the machine is a NAT system it will connect to the host machine and gain network access

```
[root@vprometheous4 ~]# ping google.com
ping: google.com: Name or service not known
[root@vprometheous4 ~]# ping google.com
PING google.com (142.250.187.206) 56(84) bytes of data.
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=1 ttl=63 time=5.65
ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=2 ttl=63 time=5.61
ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=3 ttl=63 time=5.96
ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=4 ttl=63 time=5.78
ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=5 ttl=63 time=5.72
ms
^C
```

This allowed me to gain access to the internet. Proof in figure below demonstrating a ping to google on my VM i had to go and connect the wired connection in the network settings.



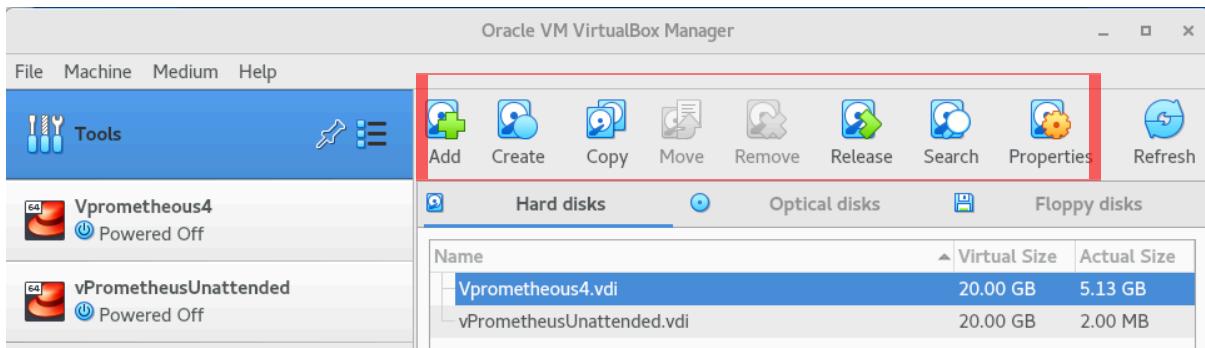
I am doing the same process giving it all the same specs as the other VM but i am not “checking” the “skip unattended installation” this is to see if everything is still possible for this machine even if i do not click that, all of this with the same installation process



```
No bootable medium found!
Please insert a bootable medium and reboot.
```

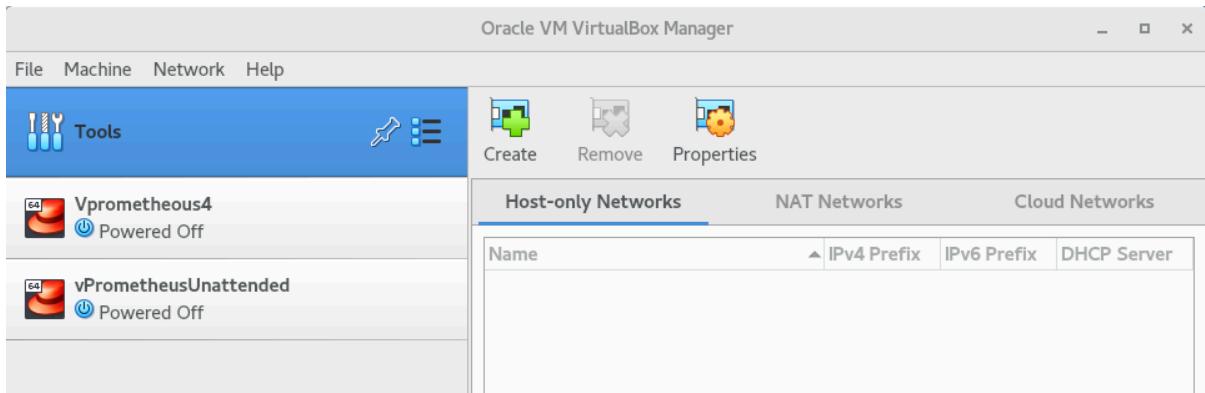
After trying a few different times with the same process this is the outcome that happens. It always gave me that there is no medium found. I assume this is something wrong with my end, but this happened multiple times when I was creating the system and after a few tries there was no way of fixing it

Virtual box and virtual machine settings



I have gone to “file-Virtual media manager” to see what function I can perform there.

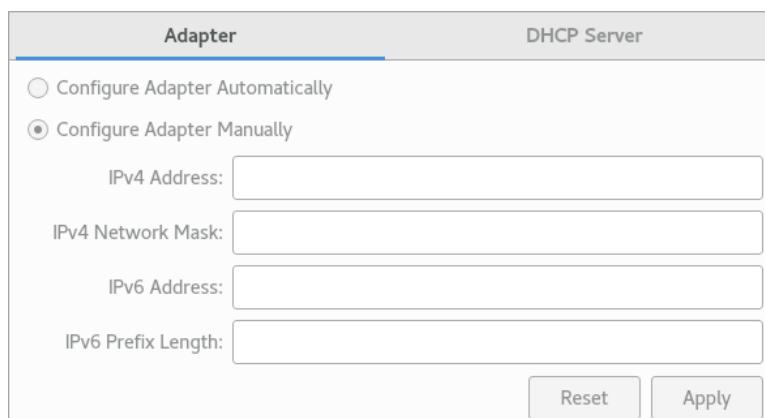
- From the part highlighted in red on the figure you can see that I am able to copy a virtual machine, also I am able to release a machine and search for a specific machine assuming this is for when people have loads of different machines.

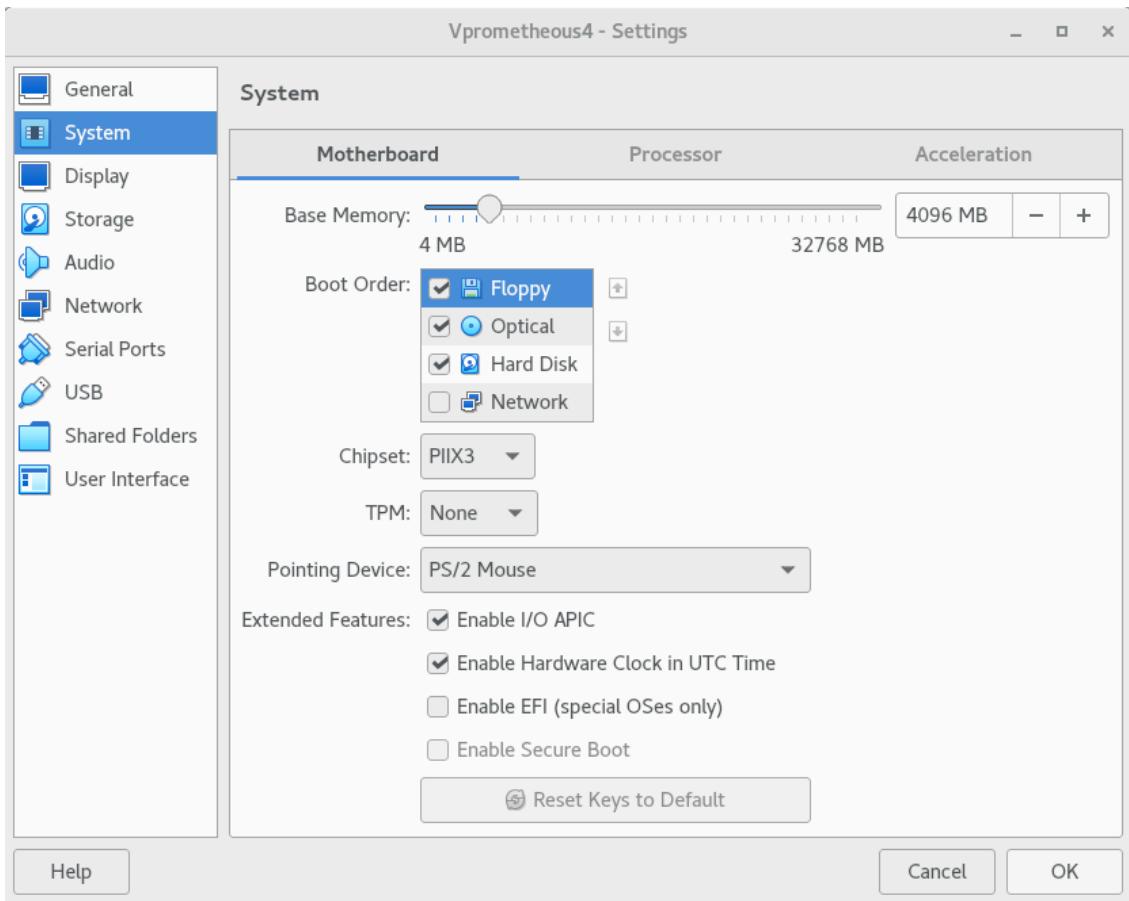


- I can also do the basic add, delete.

I have gone to “File-Network manager” to see what function I can perform there.

- From what I can see I can create the potential, Host-only networks, NAT network, and Cloud networks that will be available for my Virtual machines
- I can also look at the properties of a network type, shown in the figure below





In the settings of the VM you can basically change everything associated with it. Such as: Ram, Storage, Video memory, audio drivers and much more.

Some other things that you can change are:

Serial ports: Let the user create and configure serial ports on the guest system

USB: lets you use the usb port from the host machine and get them on the guest machine, allowing stuff like file transfer and more

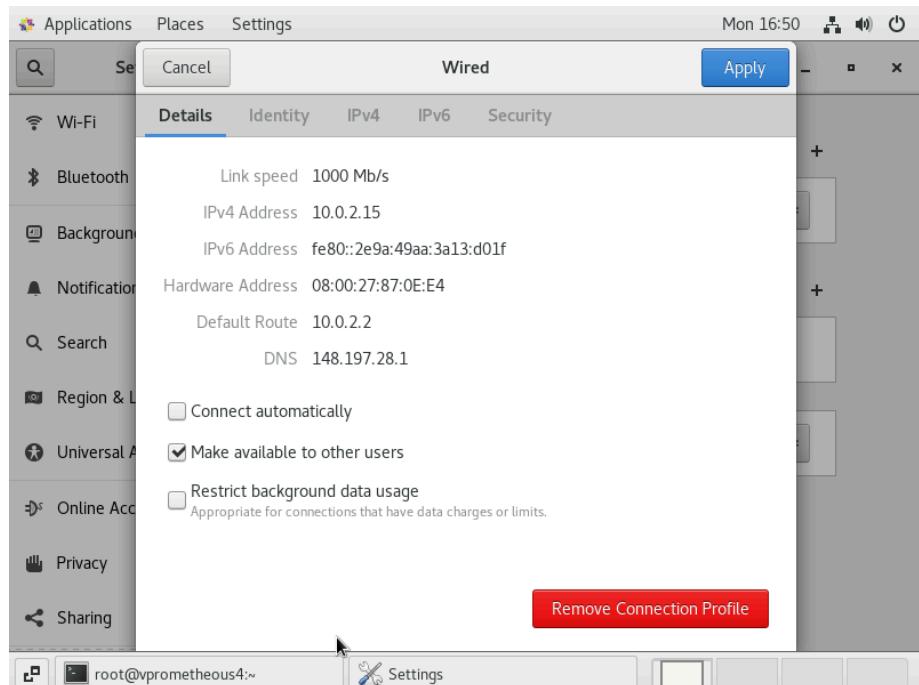
Shared folders: You can create file permission for different user accounts you can access host system files with the guest system

Display: allows the user to specific video memory, specific monitor count if they are using more than one, the scale factor and what graphics controller they are using, it's like using normal monitor settings on windows

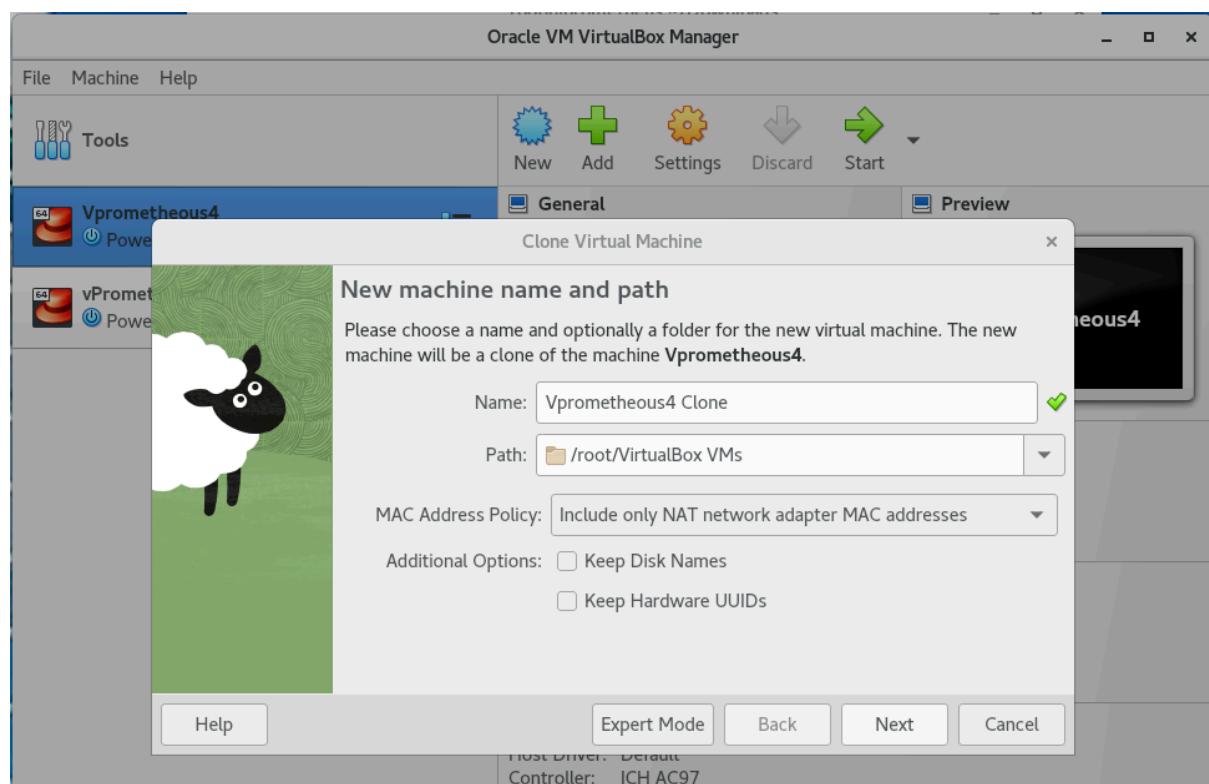
Audio: allows the user to change specific audio drivers, controllers, output and input

Overall, these settings are very similar to a regular computer and that is very useful when using the virtual machine provides an ease of use factor when changing stuff

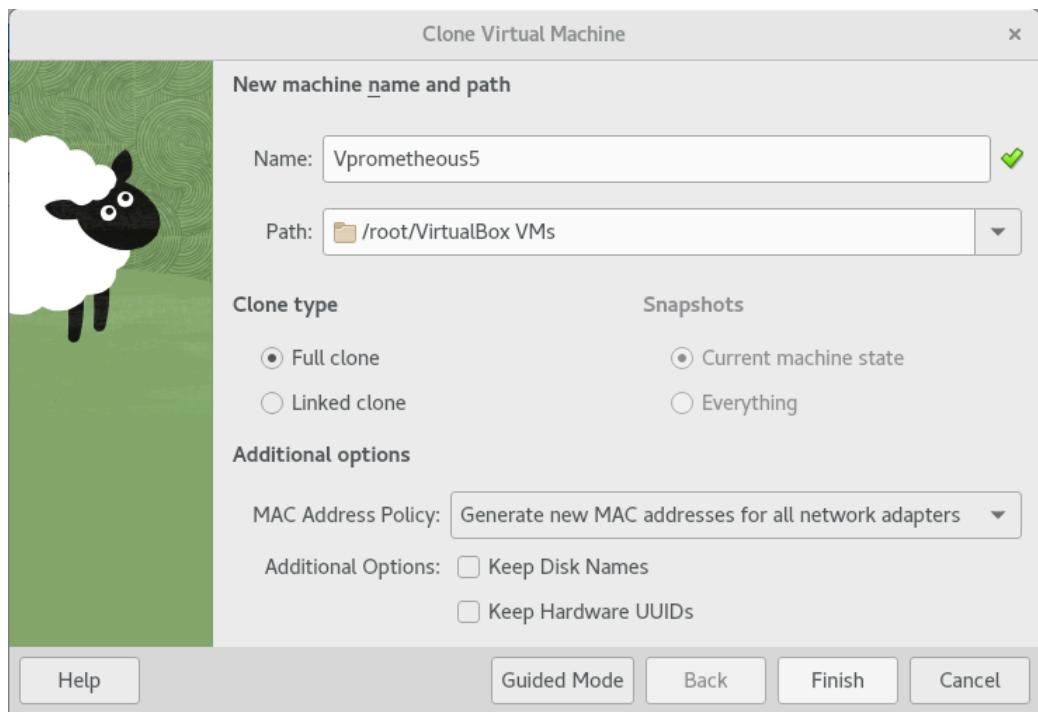
Virtualbox Networking



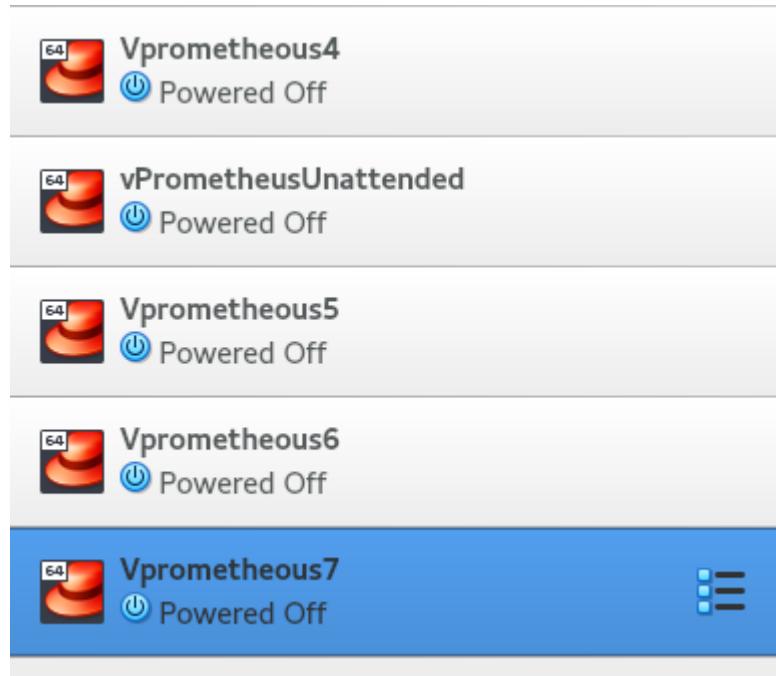
From this image we can see my IP, DNS Server, and Default gateway. They are allocated through NAT since it connected to the host machine so it gains a form of IP front that host



I am going to be making more VM by using the cloning feature that virtual box provides. I will be using expert mode for this.



In this have gone and changed the name to Vprometheous5 and made it so the clone will “Generate new MAC addresses for all the network adapters” this will make new MAC address for all the adapter in this new VM, I will be going and repeating this two more times for a total of 4 VMs.



In this image you can see all of the other virtual machines that I have created. Please ignore the “Prometheus Unattended” this was for a test back before in the labsheet.

For every figure below you will see the Hostname being changed for each of their virtual Machines

The image consists of three vertically stacked screenshots of a Linux desktop environment, likely Kali Linux, showing terminal windows for three different virtual machines. Each terminal window has a title bar with a menu (File, Machine, View, Input, Devices, Help) and a toolbar with icons for Applications, Places, and Terminal. The desktop background shows a grid of icons.

Screenshot 1 (Top):

```
File Machine View Input Devices Help
Applications Places Terminal
root@vprometeous5 ~]# hostname
vprometeous5.netlab.ee.port.ac.uk
[root@vprometeous5 ~]#
```

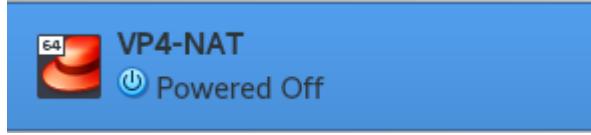
Screenshot 2 (Middle):

```
File Edit View Search Terminal Help
hos[root@vprometeous6 ~]# hostname
vprometeous6.netlab.ee.port.ac.uk
[root@vprometeous6 ~]#
```

Screenshot 3 (Bottom):

```
File Edit View Search Terminal Help
[root@vprometeous7 ~]# hostname
vprometeous7.netlab.ee.port.ac.uk
[root@vprometeous7 ~]#
```

These are all the hostnames after a restart of the machines to fully confirm the change.



I have gone and changed the name of “Vprometheous4” to “VP4-NAT” since it will be a NAT networking device

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 08:00:27:87:0e:e4 txqueuelen 1000 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 208 bytes 18968 (18.5 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 208 bytes 18968 (18.5 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
        ether 52:54:00:04:f5:36 txqueuelen 1000 (Ethernet)
          RX packets 0 bytes 0 (0.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 0 bytes 0 (0.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
[root@vprometheous4 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref  Use Iface
192.168.122.0   0.0.0.0       255.255.255.0   U     0      0      0 virbr0
```

These are the results of me typing “ifconfig” on my VP4-NAT system.

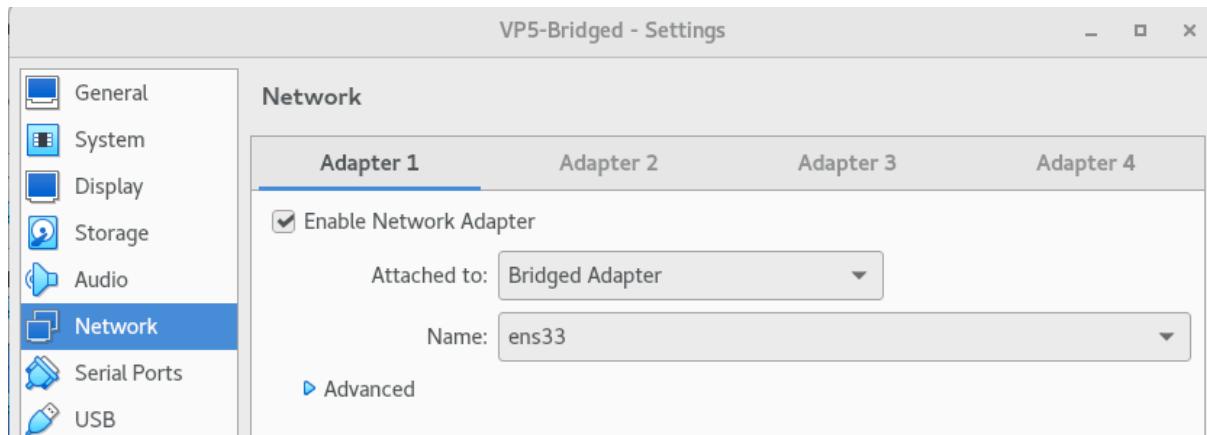
These are the results of me typing “route -n” on my VP4-NAT system.

```
[root@vprometheous4 ~]# ping google.com
PING google.com (142.250.187.206) 56(84) bytes of data.
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=1 ttl=63 time=5.64
ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=2 ttl=63 time=6.25
ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=3 ttl=63 time=5.68
ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 5.643/5.863/6.257/0.279 ms
```

These are the results of me pinging google on my VP4-NAT system.



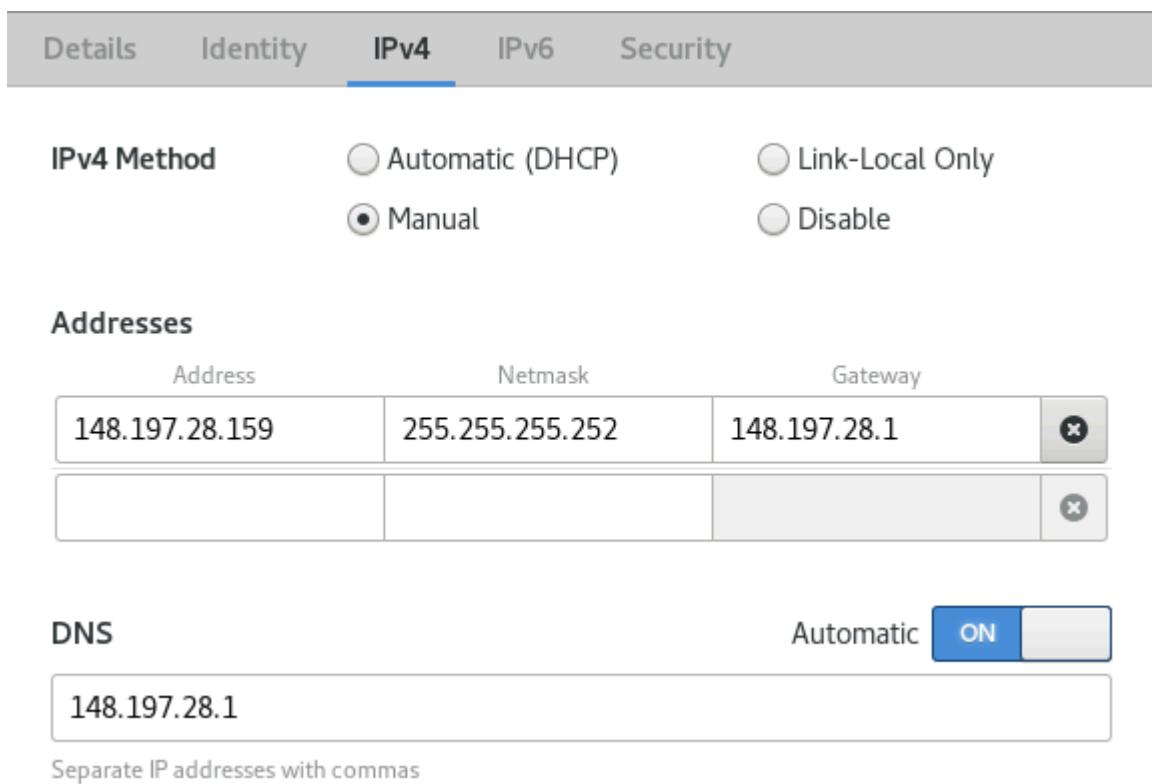
I have gone and changed the name of “Vprometheus5” to “VP5-Bridged” since it will be a



bridged networking device.

I have gone and changed the attachment of the first adapter for this VM to bridged since it will be a bridged networking device.

When loading the new VM I can no longer just connect it to the internet after it has opened. I have to go and give it a specific IP relating to my host VM. My host VM's IP end is 209 but we need to add 50 which is not possible for my IP so I have gone and given it this one, and given it all the old information needed for its other parts. (all shown in the figure below)



```

File Edit View Search Terminal Help
[[root@vprometheus5 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 148.197.28.159 netmask 255.255.255.252 broadcast 148.197.28.159
    inet6 fe80::2e9a:49aa:3a13:d01f prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:6b:eb:f6 txqueuelen 1000 (Ethernet)
        RX packets 1831 bytes 279731 (273.1 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 50 bytes 8009 (7.8 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
        RX packets 432 bytes 39504 (38.5 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 432 bytes 39504 (38.5 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
        ether 52:54:00:04:f5:36 txqueuelen 1000 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

These are the results of me typing “ifconfig” on my VP5-Bridged system.

```

[[root@vprometheus5 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
0.0.0.0         148.197.28.1   0.0.0.0        UG    100    0    0 enp0s3
148.197.28.1    0.0.0.0        255.255.255.255 UH    100    0    0 enp0s3
148.197.28.156  0.0.0.0        255.255.255.252 U      100    0    0 enp0s3
192.168.122.0   0.0.0.0        255.255.255.0   U      0      0    0 virbr0
[[root@vprometheus5 ~]# ]

```

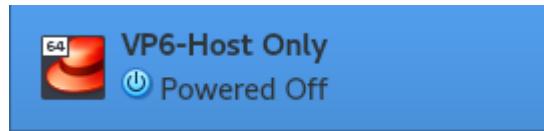
These are the results of typing “route -n” on my VP5-Bridged system.

This is the result of me pinging google.com

```

[root@vprometheus5 ~]# ping google.com
PING google.com (142.250.180.14) 56(84) bytes of data.
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=1 ttl=57 time=5.48 ms
64 bytes from lhr25s32-in-f14.1e100.net (142.250.180.14): icmp_seq=2 ttl=57 time=5.35 ms

```



I have gone and changed the name of “Vprometheous6” to “VP6-Host Only” since it will be a bridged networking device.

The screenshot shows the Oracle VM VirtualBox Manager interface. In the top navigation bar, the title "Oracle VM VirtualBox Manager" is visible. Below the title, there are three buttons: "Create" (green plus), "Remove" (red minus), and "Properties" (gear icon). The main area has three tabs: "Host-only Networks" (selected), "NAT Networks", and "Cloud Networks". Under the "Host-only Networks" tab, a table lists a single entry:

Name	IPv4 Prefix	IPv6 Prefix	DHCP Server
vboxnet0	192.168.56.1/24		Enabled

Below the table, there is a section titled "Adapter" with the following configuration options:

- Configure Adapter Automatically
- Configure Adapter Manually

Fields for manual configuration include:

- IPv4 Address: 192.168.56.1
- IPv4 Network Mask: 255.255.255.0
- IPv6 Address: (empty)
- IPv6 Prefix Length: 0

At the bottom of the adapter configuration section are "Reset" and "Apply" buttons.

After changing the name of the VM I went to the network setting on virtualbox and created a new host-only network called vboxnet0, with this I have also allowed it to be configured automatically. (all marked in red)

Adapter	DHCP Server
<input checked="" type="checkbox"/> Enable Server	
Server Address:	192.168.56.100
Server Mask:	255.255.255.0
Lower Address Bound:	192.168.56.101
Upper Address Bound:	192.168.56.254
<input type="button" value="Reset"/> <input type="button" value="Apply"/>	

I have enabled the DHCP server. For some reason the application did not want this to work. So we had to figure out how to fix it.

Adapter	DHCP Server
<input checked="" type="checkbox"/> Enable Server	
Server Address:	148.197.30.81
Server Mask:	255.255.255.252
Lower Address Bound:	148.197.30.82
Upper Address Bound:	148.197.30.82
<input type="button" value="Reset"/> <input type="button" value="Apply"/>	

I had changed the DCHP server to match my settings to my VM. This still did not let me apply so we had to change some more settings.



Adding this allows the network to gain access from ips in the range of 0 - 255 since before it was being stopped at 192.

Adapter	DHCP Server
<input type="radio"/> Configure Adapter Automatically	
<input checked="" type="radio"/> Configure Adapter Manually	
IPv4 Address:	148.197.30.81
IPv4 Network Mask:	255.255.255.252
IPv6 Address:	
IPv6 Prefix Length:	0
	<input type="button" value="Reset"/> <input type="button" value="Apply"/>

After this I went and changed the adapter settings to these , After all of this I was able to click apply and it no longer broke

```
[root@vprometheus6 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 148.197.30.82 netmask 255.255.255.252 broadcast 148.197.30.83
        inet6 fe80::2e9a:49aa:3a13:d01f prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:54:71:39 txqueuelen 1000 (Ethernet)
            RX packets 3 bytes 1770 (1.7 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 26 bytes 4890 (4.7 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 272 bytes 24824 (24.2 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 272 bytes 24824 (24.2 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
          ether 52:54:00:04:f5:36 txqueuelen 1000 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
[root@vprometheus6 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
148.197.30.80   0.0.0.0       255.255.255.252 U        100    0        0 enp0s3
192.168.122.0   0.0.0.0       255.255.255.0   U        0        0        0 virbr0
```

These are the results of running “ifconfig” on VP6-Host Only.

These are the results of running “route -n” on VP6-Host Only.

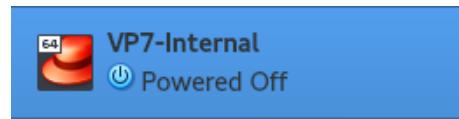
```
PING 148.197.30.81 (148.197.30.81) 56(84) bytes of data.
64 bytes from 148.197.30.81: icmp_seq=1 ttl=255 time=1.42 ms
64 bytes from 148.197.30.81: icmp_seq=2 ttl=255 time=0.321 ms
^C
--- 148.197.30.81 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.321/0.872/1.424/0.552 ms
[root@vprometheus6 ~]# ping google.com
ping: google.com: Name or service not known
[root@vprometheus6 ~]# ping google.com
```

The machine can not ping google but it can ping the host

```
[root@vprometheous6 ~]# ping 148.197.30.81
PING 148.197.30.81 (148.197.30.81) 56(84) bytes of data.
64 bytes from 148.197.30.81: icmp_seq=1 ttl=255 time=1.42 ms
64 bytes from 148.197.30.81: icmp_seq=2 ttl=255 time=0.321 ms
^C
--- 148.197.30.81 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1ms
rtt min/avg/max = 0.321/1.42/1.42 ms
```

The next part of the labsheet asks me to do many things involving changing stuff within the “vboxnet0”. But, when doing the tasks for VP6 i had to go and do all of the stuff mentioned early since it would not allow me to actually carry on with the host only task. If you look through the previous section you can see that I completed most of the tasks asked for this side of the task. I will provide the screenshots again below that relate to this part of the section. However, their explanation will be above.

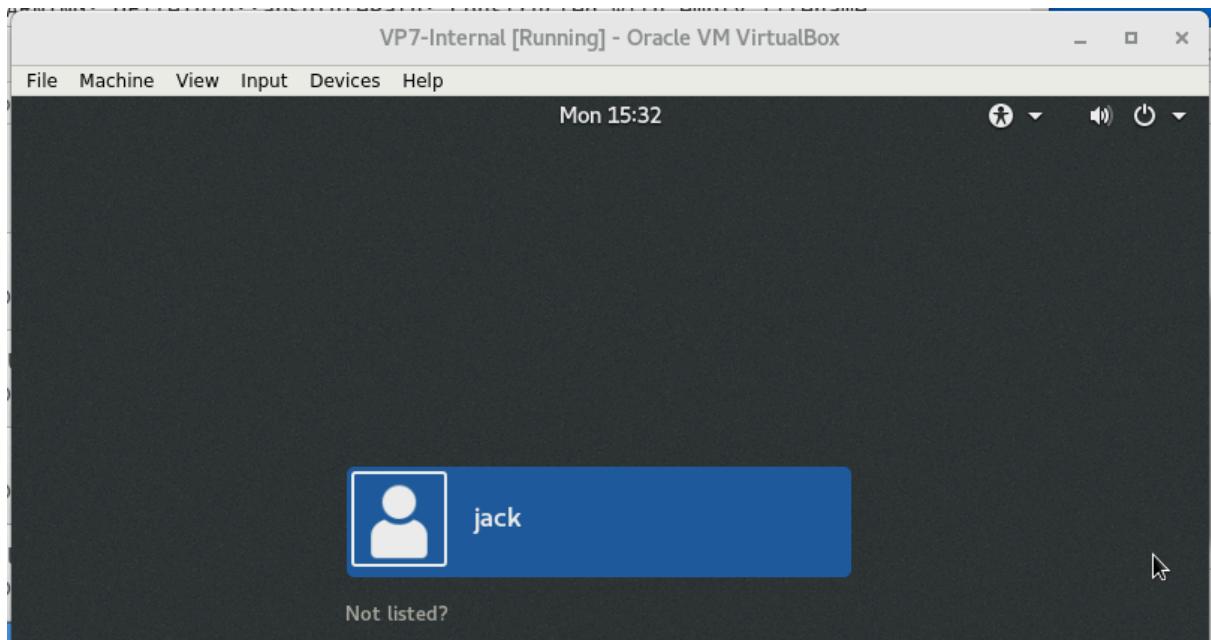
The image shows two network configuration interfaces side-by-side. The left interface is titled "Adapter" and "DHCP Server". It has a checked checkbox for "Enable Server". Under "DHCP Server", the "Server Address" is set to 192.168.56.100, "Server Mask" to 255.255.255.0, "Lower Address Bound" to 192.168.56.101, and "Upper Address Bound" to 192.168.56.254. The right interface also has "Adapter" and "DHCP Server" tabs. It has an unchecked checkbox for "Enable Server". Under "Adapter", the "Configure Adapter Manually" radio button is selected. The "IPv4 Address" is set to 148.197.30.81, "IPv4 Network Mask" to 255.255.255.252, and "IPv6 Prefix Length" to 0. Both interfaces have "Reset" and "Apply" buttons at the bottom right.



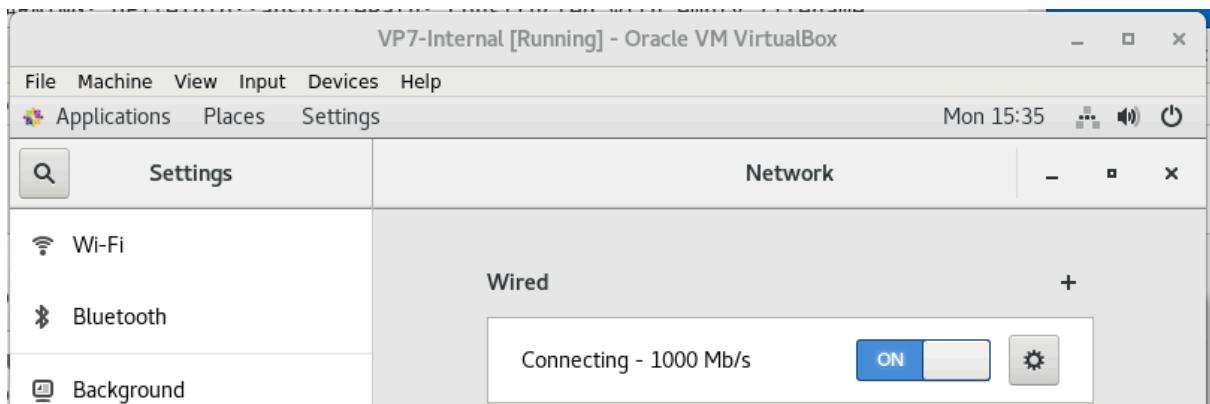
I have gone and changed the name of “Vprometheus7” to “VP7-Internal” since it will be a bridged networking device.

Adapter 1	Adapter 2	Adapter 3	Adapter 4
<input checked="" type="checkbox"/> Enable Network Adapter			
Attached to: Internal Network			
Name: intnet			
Advanced			

I have gone and changed it from NAT to internal network.



This is me loading into the VM after changing the network settings and seeing that I most likely need to configure the internet access for this VM. As soon as i got passed the login screen i can see there is no connection so i need to go and check what is wrong



After just turning it on you can see there is an issue since it has no ability to connect and eventually fails.

This detailed screenshot shows the 'Wired' network configuration in the 'IPv4' tab. The 'IPv4 Method' is set to 'Manual'. The 'Addresses' section lists one entry: Address 148.197.30.81, Netmask 255.255.255.252, and Gateway 148.197.28.1. The 'DNS' section shows 'Automatic' is turned on, and the primary DNS is 148.197.28.1. A note at the bottom says 'Separate IP addresses with commas'.

Address	Netmask	Gateway
148.197.30.81	255.255.255.252	148.197.28.1

I went and changed the network setting to these and now there is access to the internet

```
[root@vprometheus7 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 148.197.30.81 netmask 255.255.255.252 broadcast 148.197.30.83
        inet6 fe80::2e9a:49aa:3a13:d01f prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:d8:a0:0f txqueuelen 1000 (Ethernet)
                RX packets 0 bytes 0 (0.0 B)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 61 bytes 6301 (6.1 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
                RX packets 362 bytes 33466 (32.6 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 362 bytes 33466 (32.6 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
        ether 52:54:00:04:f5:36 txqueuelen 1000 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
[root@vprometheus7 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
0.0.0.0         148.197.28.1   0.0.0.0       UG    100    0      0 enp0s3
148.197.28.1    0.0.0.0        255.255.255.255 UH    100    0      0 enp0s3
148.197.30.80   0.0.0.0        255.255.255.252 U      100    0      0 enp0s3
192.168.122.0   0.0.0.0        255.255.255.0    U      0      0      0 virbr0
```

These are the results of “ifconfig” on “VP7-Internal”

This is the result of running the “route -n” command on my VP7

This VM is not able to ping google.com or prometheus, this is because –

```
[root@vprometheus7 ~]# ping google.com
^C
[root@vprometheus7 ~]# ping 148.197.28.209
PING 148.197.28.209 (148.197.28.209) 56(84) bytes of data.
From 148.197.30.81 icmp_seq=1 Destination Host Unreachable
From 148.197.30.81 icmp_seq=2 Destination Host Unreachable
From 148.197.30.81 icmp_seq=3 Destination Host Unreachable
^C
--- 148.197.28.209 ping statistics ---
4 packets transmitted, 0 received, +3 errors, 100% packet loss, time 3006ms
pipe 3
```

Write a comparison between the two. Keep it around 500 to 1000 words and compare features or tasks that were easy on the one but difficult on the other. Which do you prefer and why?

In the beginning sections of these lab sheets, we discussed how to create virtual networks through programs/software like VMware and Virtual Box, with two differences in how "easy" one was to use when making this virtual network.

I will be discussing those differences and giving my opinion on them.

After using both Virtual Box and VMware, I can say that VMware provides more "ease of use" when creating virtual machines, let alone a virtual network. With the creation of the virtual machine, VMware offers more information throughout the slide. This is great because it allows me to understand what I am affecting regarding the system. The virtual Box has it all on one page with drop-down menus that do not provide me with the most information, just what I want to do in that section.

One advantage for VMware regarding the two lab sheets is the set-up of the host-only machine. With VMware, it was a very clean-cut process. I set the device to host only and changed whatever needed change to allow it to work fully; this was great when making the virtual network.

However, with Virtual Box, we had to go into the network settings of the virtual machine manager and add a new host-only network with a DHCP server. After following it correctly and putting the correct information down, it just would not work. After looking around the network files, I found a file under "network.conf" with a digit stopping my IP range from being the max 0 - 255; this is an issue because it would not let me create the host-only network. For a new user of this software, messing with configuration files could be more appealing when another software works immediately. This could have been because of something I could not control, but it still affects how I use it.

Virtual Box provides more settings that can help a new user. Virtualbox provides a lot more options when it comes to settings of the overall virtual machines. At a point in the lab sheet, there is a moment where we need to clone the virtual machine multiple times; when it comes to VMware's ability, it is a lot "harder" to do. You have to go into the file settings and copy the files. But the Virtual Box software gives you a whole section dedicated to cloning a machine you have created. It creates a more accessible and obvious idea that you can clone your virtual machine and present it to the user more clearly.

Virtual Box is a lot more open source than VMware; this is great because it allows users to be able to do a lot with the virtual machines; this is where I think the issue with me creating the host-only network adapter came into play since it allows its user to do a lot more with the source of the system.

With the work that I did with both virtual machine software, I have to say I prefer VMware as a software since it was a lot clearer and more straightforward when I used the tool; it provides a more ease of use factor when using the software which helped a lot when doing the task. This is quite ironic since VirtualBox is a software meant for beginners but for me seems a lot more advanced than something like VMware.

Lab Sheet 3 - Opennebula setup

This segment of my logbook is formed around open nebula and how we can use cloud computing to create virtual machines that can be used to manage a virtual network. I will be demonstrating it all from the beginning how to install and how to set up, leading to the ability to use the virtual network

Front-end installation/Add opennebula repositories

Community Edition

The community edition of OpenNebula offers the full functionality of the Cloud Management Platform. You can configure the community repositories as follows:

AlmaLinux/RHEL

To add OpenNebula repository, execute the following as user `root`:

RHEL 8, 9

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo
[opennebula]
name=OpenNebula Community Edition
baseurl=https://downloads.opennebula.io/repo/6.6.0/RedHat/$releasever/$basearch
enabled=1
gpgkey=https://downloads.opennebula.io/repo/repo2.key
gpgcheck=1
repo_gpgcheck=1
EOT
# yum makecache
```

AlmaLinux 8, 9

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo
[opennebula]
name=OpenNebula Enterprise Edition
baseurl=https://downloads.opennebula.io/repo/6.6.0/AlmaLinux/$releasever/$basearch
enabled=1
gpgkey=https://downloads.opennebula.io/repo/repo2.key
gpgcheck=1
repo_gpgcheck=1
EOT
# yum makecache
```

Debian/Ubuntu

Note

If the commands below fail, ensure you have `gnupg`, `wget` and `apt-transport-https` packages installed and retry. E.g.,

```
# apt-get update
# apt-get -y install gnupg wget apt-transport-https
```

First, add the repository signing GPG key on the Front-end by executing as user

For the start of this installation we will need to go to the open nebula website and download a repository for our linux distribution. Since we are using centOS for this we will be getting the RHEL 8, 9 version ((Red Hat Enterprise Linux 8,9)

We will be adding this repository to root.



The screenshot shows a text editor window with the title bar "opennebula.repo /etc/yum.repos.d". The content of the file is as follows:

```
[opennebula]
name=OpenNebula Community Edition
baseurl=https://downloads.opennebula.io/repo/6.2/CentOS/7/$basearch
enabled=1
gpgkey=https://downloads.opennebula.io/repo/repo.key
gpgcheck=1
repo_gpgcheck=1
```

After checking the root file, this was its default, I went and added the extra stuff the website recommends shown below.



The screenshot shows a text editor window with the title bar "*opennebula.repo /etc/yum.repos.d". The content of the file is as follows:

```
# cat << "EOT" > /etc/yum.repos.d/opennebula.repo
[opennebula]
name=OpenNebula Community Edition
baseurl=https://downloads.opennebula.io/repo/6.2/CentOS/7/$basearch
enabled=1
gpgkey=https://downloads.opennebula.io/repo/repo.key
gpgcheck=1
repo_gpgcheck=1
EOT
# yum makecache
```

This is the new version of the repo

Installing the software

I will be going and activating the EPEL repo shown below:

```
Loading mirror speeds from cached hostfile
epel/x86_64/metalink | 27 kB 00:00
  * base: mirrors.vinters.com
  * epel: mirror.netcologne.de
  * extras: mirror.freethought-internet.co.uk
  * updates: mirrors.vinters.com
base | 3.6 kB 00:00
extras | 2.9 kB 00:00
google-chrome | 1.3 kB 00:00
opennebula/x86_64/signature | 488 B 00:00
opennebula/x86_64/signature | 3.0 kB 00:00 !!!
updates | 2.9 kB 00:00
(1/3): epel/x86_64/filelists_db | 12 MB 00:01
(2/3): epel/x86_64/prestodelta | 772 B 00:00
(3/3): google-chrome/filelists | 1.8 kB 00:00
google-chrome
Metadata Cache Created
3/3
[root@prometheus ~]#
```

First command used: yum makecache

The result of this is that it created a Metadata cache this is a local datastore that keeps metadata information about virtual machines

```
Total size: 15 k
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating : epel-release-7-14.noarch 1/2
  Cleanup  : epel-release-7-11.noarch 2/2
  Verifying : epel-release-7-14.noarch 1/2
  Verifying : epel-release-7-11.noarch 2/2

Updated:
  epel-release.noarch 0:7-14

Complete!
```

Second command: yum install espel-release

For a front end installation there are packages distributed in many different components that conform to OpenNebula, and packages for the virtualization. For us to install a centOS/REHL OpenNebula front-end with packages from our repository we need to use these two lines of code.

```
[root@prometheus ~]# yum install opennebula-server opennebula-sunstone
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirrors.vinters.com
 * epel: mirror.netcologne.de
 * extras: mirror.freethought-internet.co.uk
 * updates: mirrors.vinters.com
Package opennebula-6.2.0.1-1.el7_9.x86_64 already installed and latest version
Package opennebula-sunstone-6.2.0.1-1.el7_9.noarch already installed and latest
version
Nothing to do
```

First command: yum install opennebula-server opennebula-sunstone
I ran the code but there was nothing to install since it was already installed

```
[root@prometheus ~]# yum install opennebula-gate opennebula-flow
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirrors.vinters.com
 * epel: mirror.netcologne.de
 * extras: mirror.freethought-internet.co.uk
 * updates: mirrors.vinters.com
Package opennebula-gate-6.2.0.1-1.el7_9.noarch already installed and latest vers
ion
Package opennebula-flow-6.2.0.1-1.el7_9.noarch already installed and latest vers
ion
Nothing to do
```

Second command: yum install opennebula-gate opennebula-flow
As same as the old code i did not need to install anything

```
[root@prometheus ~]# su oneadmin
[oneadmin@prometheus root]$ 
```

Starting OpenNebula

I am logging in as one admin this is for the ability to start open nebula

```
[root@prometheus ~]# su oneadmin
[oneadmin@prometheus root]$ echo "oneadmin:localhost1"> ~/.one/one_auth
[oneadmin@prometheus root]$ 
```

I then run this command and as you can see there is no result since it is making a file with a bunch of auth commands.

I went and typed “gedit /etc/one/sunstone-server.conf” and added comments to the following lines.

```
#####
# FireEdge
#####

#:private_fireedge_endpoint: http://localhost:2616
#:public_fireedge_endpoint: http://localhost:2616
```

Plain Text ▾ Tab Width: 8 ▾ Ln 26


```
[root@prometheus ~]# gedit /etc/one/sunstone-server.conf
[root@prometheus ~]# systemctl start opennebula
[root@prometheus ~]# systemctl start opennebula-sunstone
[root@prometheus ~]# 
```

The two commands highlighted in red are commands that are starting the two applications that are needed to be running

Verifying the installation

Linux Command line interface

At the start of verifying the installation I went and typed the command “oneuser show”

```
[root@prometheus ~]# oneuser show
USER 0 INFORMATION
ID          : 0
NAME        : oneadmin
GROUP       : oneadmin
PASSWORD    : 6f66acfca520218d60fee60f2a230c71122b413574a042ef224cc9a29cd742e
AUTH_DRIVER : core
ENABLED     : Yes

TOKENS

USER TEMPLATE
TOKEN_PASSWORD="df7051a3153753dba9393731c1894f7f13338ef26aed57b14643e62c1c1efd18"

VMS USAGE & QUOTAS

VMS USAGE & QUOTAS - RUNNING

DATASTORE USAGE & QUOTAS

NETWORK USAGE & QUOTAS

IMAGE USAGE & QUOTAS

[root@prometheus ~]#
```

into my terminal and this is the result. We can see we have oneadmin and the password hash.

Sunstone



After loading the website “localhost:9869” we are greeted with this –

This is the website created for the Graphical User Interface for the OpenNebula application

Dictionary structure

This image below is the dictionary structure on file paths that are available in the “Front-end” after installing it

Path	Description
/etc/one/	Configuration Files
/var/log/one/	Log files, notably: oned.log, sched.log, sunstone.log and <vmid>.log
/var/lib/one/	oneadmin home directory
/var/lib/one/datastores/<dsid>/	Storage for the datastores
/var/lib/one/vms/<vmid>/	Action files for VMs (deployment file, transfer manager scripts, etc...)
/var/lib/one/.one/one_auth	oneadmin credentials
Path	Description
/var/lib/one/remotes/	Probes and scripts that will be synced to the Hosts
/var/lib/one/remotes/hooks/	Hook scripts
/var/lib/one/remotes/vmm/	Virtual Machine Manager Driver scripts
/var/lib/one/remotes/auth/	Authentication Driver scripts
/var/lib/one/remotes/im/	Information Manager (monitoring) Driver scripts
/var/lib/one/remotes/market/	MarketPlace Driver scripts
/var/lib/one/remotes/datastore/	Datastore Driver scripts
/var/lib/one/remotes/vnm/	Networking Driver scripts
/var/lib/one/remotes/tm/	Transfer Manager Driver scripts

KVM Node Installation

I will be running two commands:

- yum install opennebula-node-kvm.noarch: this command goes and install a KVM node package, this allows us to utilise the kernel, to create kernel based virtual machines
- service libvirtd restart: This command goes and restarts the libvirtd service

```
[root@prometheus ~]# yum install opennebula-node-kvm.noarch
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirrors.vinters.com
 * epel: mirror.netcologne.de
 * extras: mirror.freethought-internet.co.uk
 * updates: mirrors.vinters.com
Package opennebula-node-kvm-6.2.0.1-1.el7_9.noarch already installed and latest version
Nothing to do

[root@prometheus ~]# service libvirtd restart
Redirecting to /bin/systemctl restart libvirtd.service
[root@prometheus ~]#
```

Configure passwordless ssh

OpenNebula will need to SSH passwordlessly from any node (including the frontend) to any other node. Add the following snippet to `~/.ssh/config` as `oneadmin` so it doesn't

prompt to add the keys to the `known_hosts` file:

```
Host *
  HashKnownHosts no
  StrictHostKeyChecking no
  UserKnownHostsFile /dev/null
  ServerAliveInterval 10
  # IMPORTANT: set the following 'Control*' options the same way as above
  ControlMaster no
  ControlPersist 70s
  ControlPath /run/one/ssh-socks/ctl-%C.sock

  chmod: missing operand after '/var/lib/one/.ssh/config'
  Try 'chmod --help' for more information.
  [oneadmin@prometheus root]$ chmod 600 ~/.ssh/config
  [oneadmin@prometheus root]$
```

Giving it the “600” permissions for the .ssh config file

```
Try 'chmod --help' for more information.  
oneadmin@prometheus root]$ chmod 600 ~/.ssh/config  
oneadmin@prometheus root]$ ssh localhost  
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.  
Last login: Mon Feb 20 16:52:12 2023  
oneadmin@prometheus ~]$ exit  
logout  
Connection to localhost closed.  
oneadmin@prometheus root]$  
oneadmin@prometheus root]$  
oneadmin@prometheus root]$ exit  
exit  
[root@prometheus ~]#  
[root@prometheus ~]#  
[root@prometheus ~]#  
[root@prometheus ~]#  
[root@prometheus ~]# ssh localhost  
The authenticity of host 'localhost (::1)' can't be established.  
ECDSA key fingerprint is SHA256:EGu35w9plmvuy0b8j3+kH9e0v7HwGwKEPvskH1QADsQ.  
ECDSA key fingerprint is MD5:4f:f2:8f:af:89:83:11:41:4f:99:85:62:6b:82:d6:73.  
Are you sure you want to continue connecting (yes/no)? no  
Host key verification failed.  
[root@prometheus ~]#  
[root@prometheus ~]#  
[root@prometheus ~]#  
[root@prometheus ~]# █
```

Running the ssh localhost results marked in red

Lab Sheet 4 - VM creation

This labsheet is where we will be going and changing some files for opennebula to be able to use them to create network interface cards for the system. With this we will be going and making the actual Vm to company them

```
[root@prometheus ~]# mkdir backupfiles
[root@prometheus ~]# ls
anaconda-ks.cfg          Documents      perl5       Videos
backupfiles               Downloads      Pictures    VirtualBox VMs
CentOS-7-x86_64-DVD-2009.iso initial-setup-ks.cfg Public     vm_subnet_2
Desktop                   Music        Templates   vmware
[root@prometheus ~]#
```

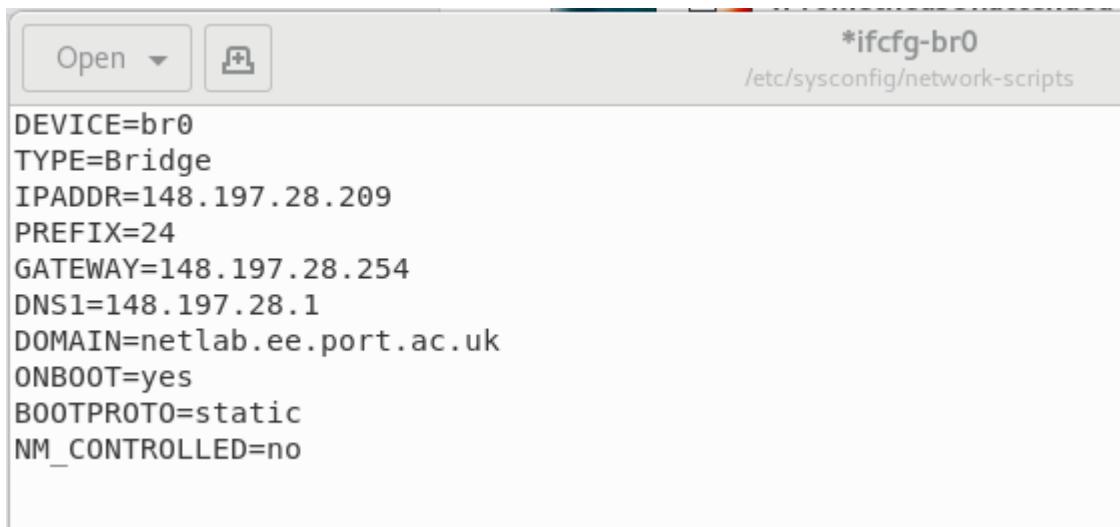
I will be making a new directory that is used to hold backup files just in case there is an issue with the one we will be changing



```
Open ▾  ifcfg-ens33
/etc/sysconfig/network-scripts

DEVICE=ens33
BOOTPROTO=none
NM_CONTROLLED=no
ONBOOT=yes
TYPE=Ethernet
BRIDGE=br0
```

I have gone and added all this data into the “ifcfg-ens33” file



```
Open ▾  *ifcfg-br0
/etc/sysconfig/network-scripts

DEVICE=br0
TYPE=Bridge
IPADDR=148.197.28.209
PREFIX=24
GATEWAY=148.197.28.254
DNS1=148.197.28.1
DOMAIN=netlab.ee.port.ac.uk
ONBOOT=yes
BOOTPROTO=static
NM_CONTROLLED=no
```

I then went and created a “br0” file, and added this data to it and named the file “ifcfg-br0”

```
[root@prometheus ~]# systemctl restart network.service
[root@prometheus ~]# ping google.com
PING google.com (142.250.187.206) 56(84) bytes of data.
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=1 ttl=57 time=6.47 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=2 ttl=57 time=5.67 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=3 ttl=57 time=5.63 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=4 ttl=57 time=5.71 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=5 ttl=57 time=10.4 ms
^C
--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 5.634/6.791/10.463/1.863 ms
```

I have gone and restarted the “network.service” this is so all the changes we have just made are fully added to the system services

Adding a host

```
[root@prometheus ~]# service opennebula restart; chkconfig opennebula on
Redirecting to /bin/systemctl restart opennebula.service
Note: Forwarding request to 'systemctl enable opennebula.service'.
[root@prometheus ~]# service opennebula-sunstone restart; chkconfig opennebula-sunstone on
Redirecting to /bin/systemctl restart opennebula-sunstone.service
Note: Forwarding request to 'systemctl enable opennebula-sunstone.service'.
[root@prometheus ~]#
```

I went and ran the two commands:

```
service opennebula restart; chkconfig opennebula on
service opennebula-sunstone restart; chkconfig opennebula-sunstone on
```

These commands go and restart some services and then turns them on, we will be using opennebula to run the cloud service, and we will be using sunstone as a “GUI” for a more ease of use experience

The screenshot shows the 'Create Host' page. At the top, there are three buttons: a back arrow, 'Reset', and a green 'Create' button. Below these are two dropdown menus. The first dropdown, labeled 'Type', has 'KVM' selected and is highlighted with a red border. The second dropdown, labeled 'Cluster', shows '0: default'. There is also a 'Hostname' input field which is currently empty.

When creating the host we need to make sure we add the right information, I have gone and specified that the type of host adding is going to be a kvm

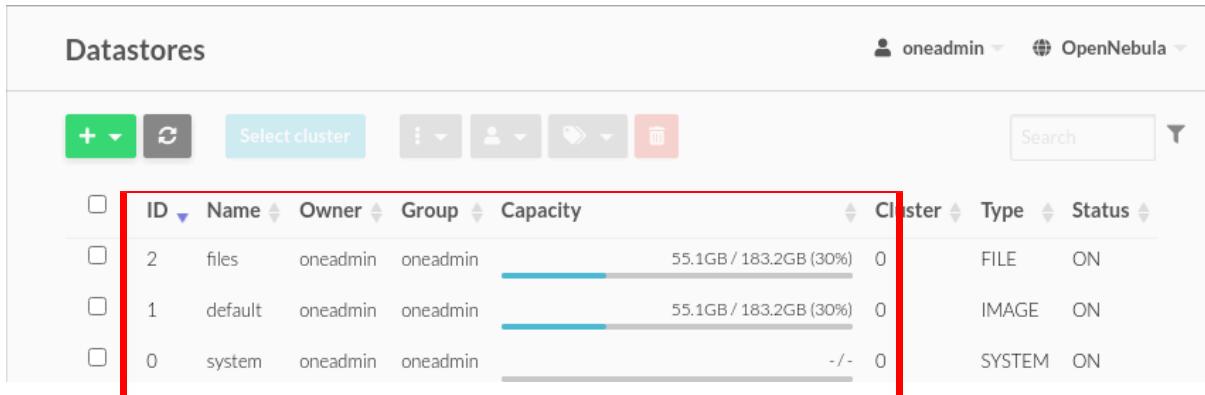
This screenshot shows the same 'Create Host' interface as before, but with a different focus. The 'Hostname' input field now contains the value 'prometheus.netlab.ee.port.ac.uk' and is highlighted with a red border, indicating it is the active or selected field. The 'Type' dropdown is still set to 'KVM' and the 'Cluster' dropdown is still set to '0: default'.

I have gone and added a host name to the system to help when connecting to the network

Hosts							oneadmin	OpenNebula		
	<input type="button" value="+"/>	<input type="button" value="refresh"/>	Select cluster	Enable	Disable	Offline	<input type="button" value="Import"/>	<input type="button" value="Delete"/>	Search	<input type="button" value="Filter"/>
	ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status			
	0	prometheus.netlab.ee.port.ac.uk	0	0	0 / 400 (0%)	0KB / 31.2GB (0%)	ON			
10	Showing 1 to 1 of 1 entries									Previous 1 Next

This is the new host after going through the creation process, we can now allocate this to any machine that we create

Adding a datastore

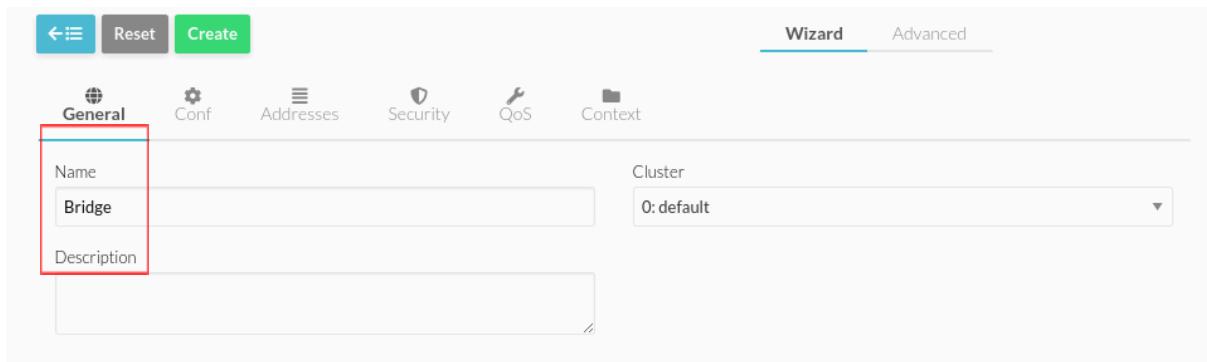


The screenshot shows the 'Datastores' page in the OpenNebula interface. At the top, there are buttons for creating a new datastore (+), cloning an existing one (copy), selecting a cluster (Select cluster), and other management functions. A search bar and a filter icon are also present. The main area is a table listing three datastores:

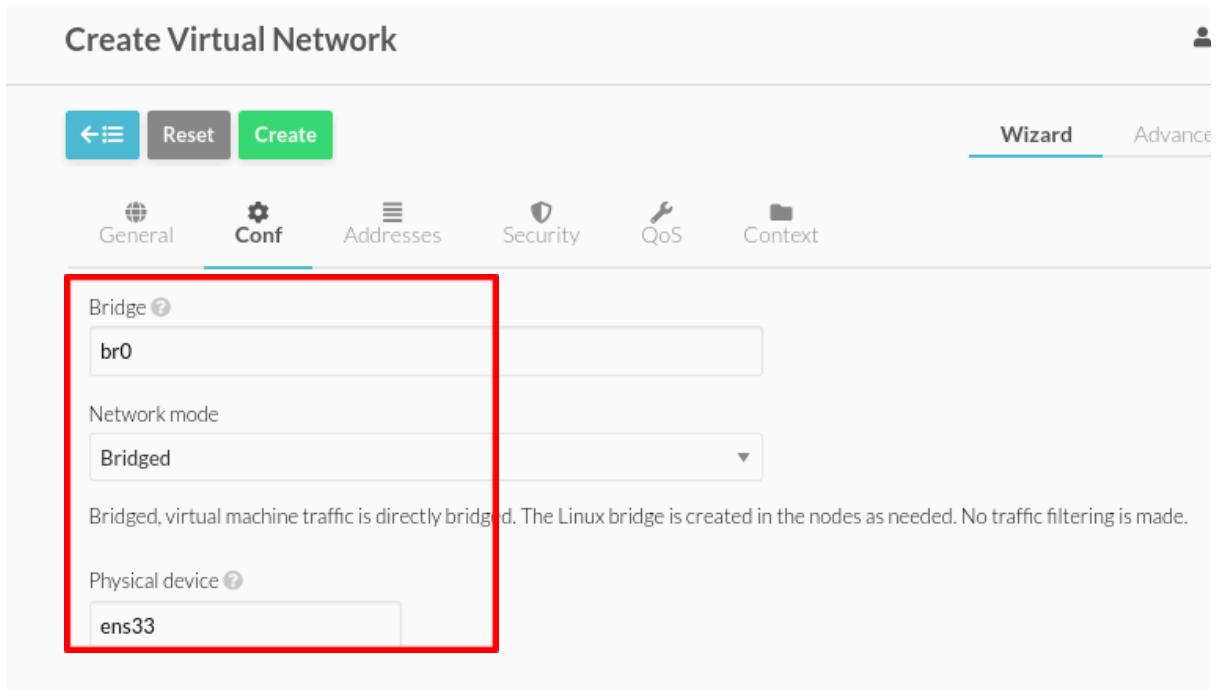
ID	Name	Owner	Group	Capacity	Cluster	Type	Status
2	files	oneadmin	oneadmin	55.1GB / 183.2GB (30%)	0	FILE	ON
1	default	oneadmin	oneadmin	55.1GB / 183.2GB (30%)	0	IMAGE	ON
0	system	oneadmin	oneadmin	-/-	0	SYSTEM	ON

Needed to check the three different data stores to the opennebula, but there was nothing wrong with them so I left them as normal. In some situations you may need to create these data stores to make sure you know what to do before making them.

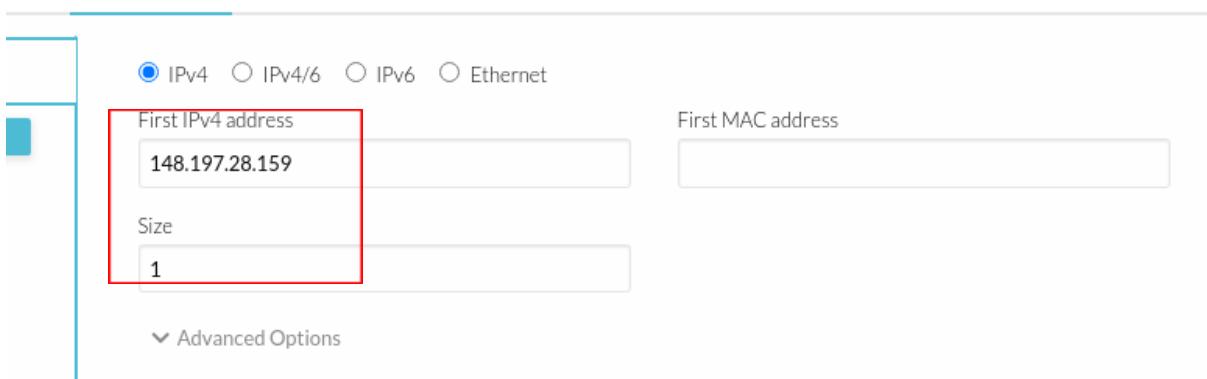
Adding networks



I am adding a virtual network and giving it the name bridge



I have gone and specified “br0” for the bridge and given it the physical device of “ens33”



I have set the IPV4 to “148.197.28.159” and changed the size to “1”

General	Conf	Addresses	Security	QoS	Context
Network address					Network mask
148.197.28.0					255.255.255.0
Gateway					IPv6 Gateway
148.197.28.254					
DNS					MTU of the Guest interfaces
148.197.28.256					
Custom attributes					

In the context file I have added a “network address”, “gateway”, “network mask” and “DNS”, this is so on the system I can apply the network addresses and gain access to the network

Adding images

The screenshot shows the OpenNebula web interface. On the left, there's a sidebar with categories like Dashboard, Instances, Templates, Storage (with sub-options Datastores, Images, Files), MarketPlaces, Apps, and Network. Under Storage, 'Images' is selected and highlighted with a red box. At the top right, there's a toolbar with various icons like '+', 'Clone', 'Lock', etc., and a 'Create' button which is also highlighted with a red box. Below the toolbar is a table header with columns: Name, Owner, Group, Datastore, Type, and Stat. The main area displays a large cloud icon with an 'i' and the message 'There is no data available'. At the bottom, it says 'Showing 0 to 0 of 0 entries' and '0 TOTAL 0 MB TOTAL SIZE'.

I went and clicked on storage → images, At this point i will be going and creating an image this is a form of storage device which will be holding data

The screenshot shows the 'Create Image' wizard. At the top, there are buttons for 'Reset' and 'Create'. Below that, there are two tabs: 'Image' (which is selected and highlighted with a red box) and 'Dockerfile'. The main form has several fields: 'Name' (containing 'vPrometheus0'), 'Description' (empty), 'Type' (set to 'Generic storage datablock'), 'Datastore' (set to '1: default'), and 'This image is persistent' (set to 'YES'). Below the form, there's a section titled 'Image location' with three radio buttons: 'Path/URL' (selected), 'Upload', and 'Empty disk image'. There's also a text input field for 'Path in OpenNebula server or URL'. At the bottom, there's a 'Advanced Options' dropdown.

At the point of creation, i went and added the name of the machine “vPromethrous0” changed the type to a “generic storage data block” and set the “image is persistent” to yes, this means we can go and save whatever we want on it like an operating system

Image location

Path/URL Upload Empty disk image

Size

10

GB ▾

After giving it all the other information i went and change it to a empty disc image and have given it 10GB of size for storage

▲ Advanced Options

BUS

Target device

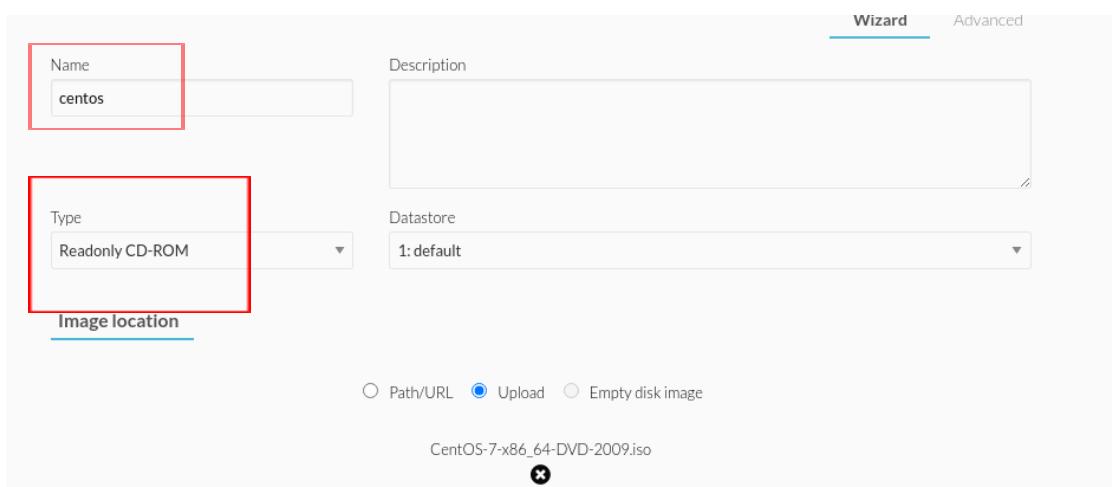
Format

Filesystem

qcow2

--

I went and changed the format in the advanced options to “qcow2” this was under the image location. This format is a form of storage mainly used for virtual machines.



On the creation of the second image, I gave it the name “centos” changed the type to “readonly CD_ROM” and also set the image to the centOS iso image.

Adding a template

A template is a pre-saved version of the virtual machine we will be making this means we can come here and it will already have every setting we need for when we make virtual machines and when we want to make more

The screenshot shows the OpenNebula management interface. On the left, a sidebar lists various categories: Dashboard, Instances, Templates (selected), Storage, Network, Infrastructure, System, and Settings. A red banner at the bottom left of the sidebar area says "Not officially supported". The main content area has a header with buttons for "+", "Update", "Instantiate", "Clone", and others. Below the header is a table with columns: ID, Name, Owner, Group, and Registration time. A large cloud icon with an info symbol is centered, and the message "There is no data available" is displayed. At the bottom right, it says "0 TOTAL". The bottom of the interface has a footer with a "Showing 0 to 0 of 0 entries" message and a page number "10".

I am going to the “template” section then “VMs” of OpenNebula to make the template for the vm

The screenshot shows a configuration interface for a virtual machine. The fields are as follows:

- Name:** vPO
- Hypervisor:** KVM (selected)
- Description:** (empty text area)
- Logo:** (empty dropdown menu)
- Memory:** 2 GB, Enable hot resize? yes, Max memory (empty), Memory modification (any value)
- Physical CPU:** 1, CPU modification (any value)
- Virtual CPU:** 1, Enable hot resize? yes, Max VCPU (empty), VCPU modification (any value)
- Cost:** Total: 0.00 COST/MONTH
- Disk:** (empty dropdown menu)

Below is what I have added to the template for the virtual machine.

Name = vPO

Hypervisor = KVM

Memory = 2gb and selected yes to “Enable host resize?”

Physical CPU = 1

Virtual CPU = 1 and selected yes to “Enable host resize?”

DISK 0

(Image) (Volatile disk)

You selected the following image: **vPrometheus0**

ID	Name	Owner	Group	Datastore	Type	Status	#VMS
1	centos	oneadmin	oneadmin	default	CDROM	READY	0
0	vPrometh...	oneadmin	oneadmin	default	DATABLO...	READY	0

Showing 1 to 2 of 2 entries

Previous 1 Next

DISK 1

(Image) (Volatile disk)

You selected the following image: **centos**

ID	Name	Owner	Group	Datastore	Type	Status	#VMS
1	centos	oneadmin	oneadmin	default	CDROM	READY	0
0	vPrometh...	oneadmin	oneadmin	default	DATABLO...	READY	0

Showing 1 to 2 of 2 entries

Previous 1 Next

Advanced options

I have gone and set the “disk 0” vPrometheus0, this is where the storage device where everything will be installed

I have gone and make a new disk “disk 1” and set it to centOS, this one holds the operating system and this is how the main storage gets the installation for the os

The screenshot shows the configuration interface for a network interface card (NIC). On the left, a sidebar displays 'NIC 0' with a '+' button. The main area has several sections: 'Interface type' (with 'Alias' and 'Automatic selection' options), 'RDP connection' (with 'Activate' option), and 'SSH connection' (with 'Activate' option). Below these is a table titled 'Network selection' showing a single entry: ID 0, Name Bridge, Owner oneadmin, Group oneadmin, Reservation No, Cluster 0, Leases 0/1. A note says 'Showing 1 to 1 of 1 entries'. At the bottom, there's a 'Search' bar and navigation buttons for 'Previous' (disabled), '1', and 'Next'.

I have gone and set “NIC” network interface card to bridge, this is the one we created a little earlier in the lab sheet, this will allow our VM to gain network access

The screenshot shows the 'Boot' tab selected in the VM settings. The configuration includes:

- CPU Architecture: x86_64
- Bus for SD disks: (dropdown menu)
- Machine type: (dropdown menu)
- Root device: sda1
- Boot order:
 - disk0: vPrometheus0 (selected)
 - disk1: centos (selected)
 - nic0: Bridge (unchecked)
- Kernel boot parameters: ro console=tty1
- Path to the bootloader executable: (empty input field)
- Unique ID of the VM: (empty input field)
- Firmware: None

In the OS&CPU settings I have gone and changed the CPU Architecture to “x86_64” and made the boot order do disk 0 then disk 1 both selected.

The screenshot shows the 'Input/Output' tab selected in the VM settings. The configuration includes:

- Graphics:
 - None
 - VNC / GUAC
 - SDL
 - SPICE
- Inputs:
 - Type: (dropdown menu)
- Listen on IP: 0.0.0.0
- Server port: (empty input field)
- Keymap: en-us

I have gone to the input and output settings and selected VNC under the graphics settings

The screenshot shows the OpenNebula web interface. On the left, there is a sidebar with the following navigation items:

- Dashboard
- Instances
 - VMs
 - Services
 - Virtual Routers
- Templates
 - VMs
 - Services
 - Virtual Routers
 - VM Groups
- Storage
- Network

The main area is titled "VMs". It features a toolbar with several icons: a green plus sign for creating new VMs, a refresh icon, and other management icons. Below the toolbar is a table header with columns: ID, Name, Owner, and Group. A message "There is no" is visible on the right side of the table area. At the bottom of the table area, there is a search bar with the value "10" and a message "Showing 0 to 0 of 0 entries". At the very bottom of the page, there are summary statistics: "0 TOTAL", "0 ACTIVE", and "0 DELETED".

After you have gone through the process of creating an instance it will appear in this section of the opennebula website

Adding a virtual machine

For this part we will need to be in the Instances → VM section of opennebula then select the created template we made

ID	Name	Owner	Group	Registration time
0	vPO	oneadmin	oneadmin	27/02/2023 16:11:58

Showing 1 to 1 of 1 entries Previous **1** Next

ID	Name	Owner	Group	Registration time
0	vPO	oneadmin	oneadmin	27/02/2023 16:11:58

Showing 1 to 1 of 1 entries Previous **1** Next

Instantiate as persistent ?

VM name Number of instances Start on hold ?

vPO

Capacity

Memory ? GB ▼

Physical CPU ?

Virtual CPU ?

Disks

DISK 0: vPrometheus0 ? MB ▼

DISK 1: centos ? MB ▼

Network

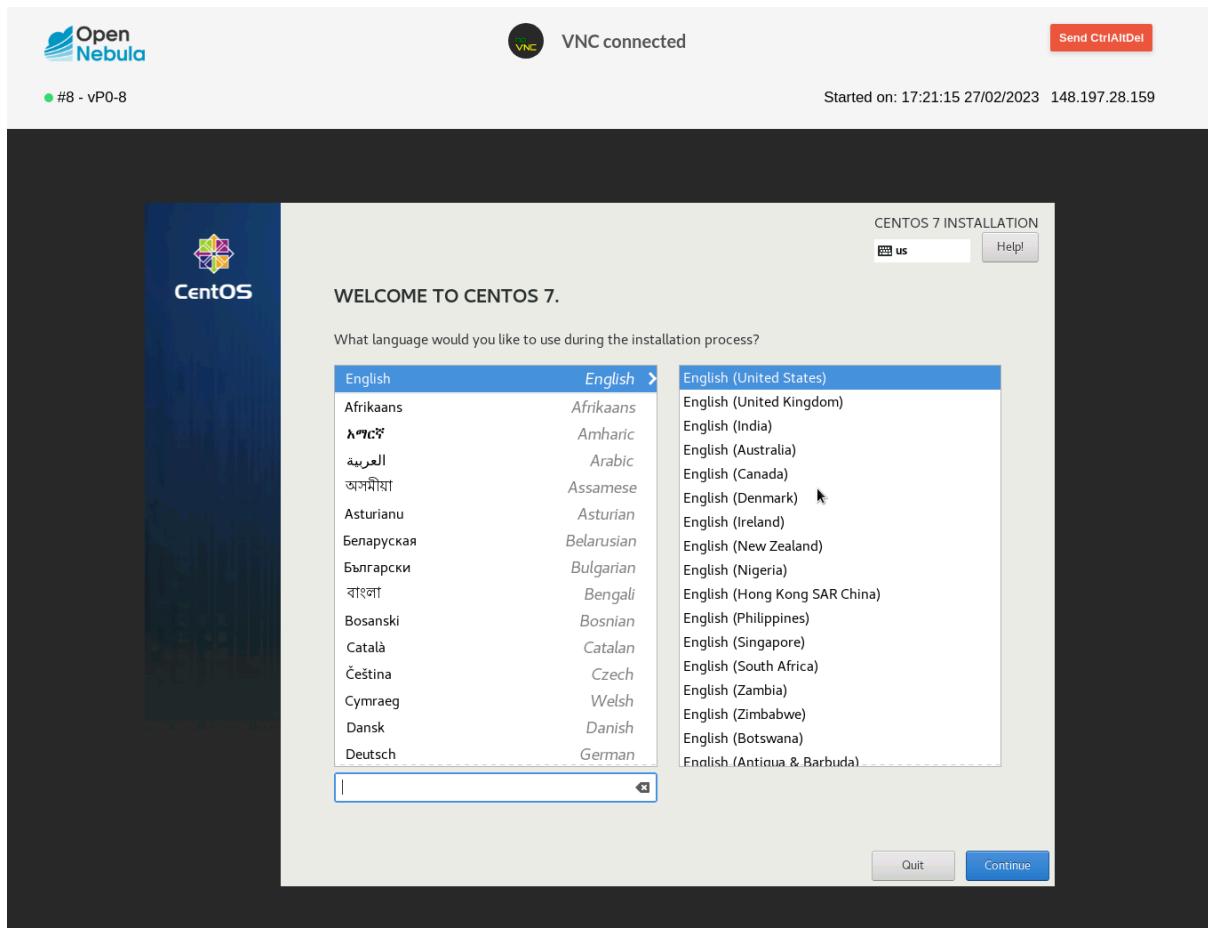
Instantiate as different User / Group

Deploy VM in a specific Host

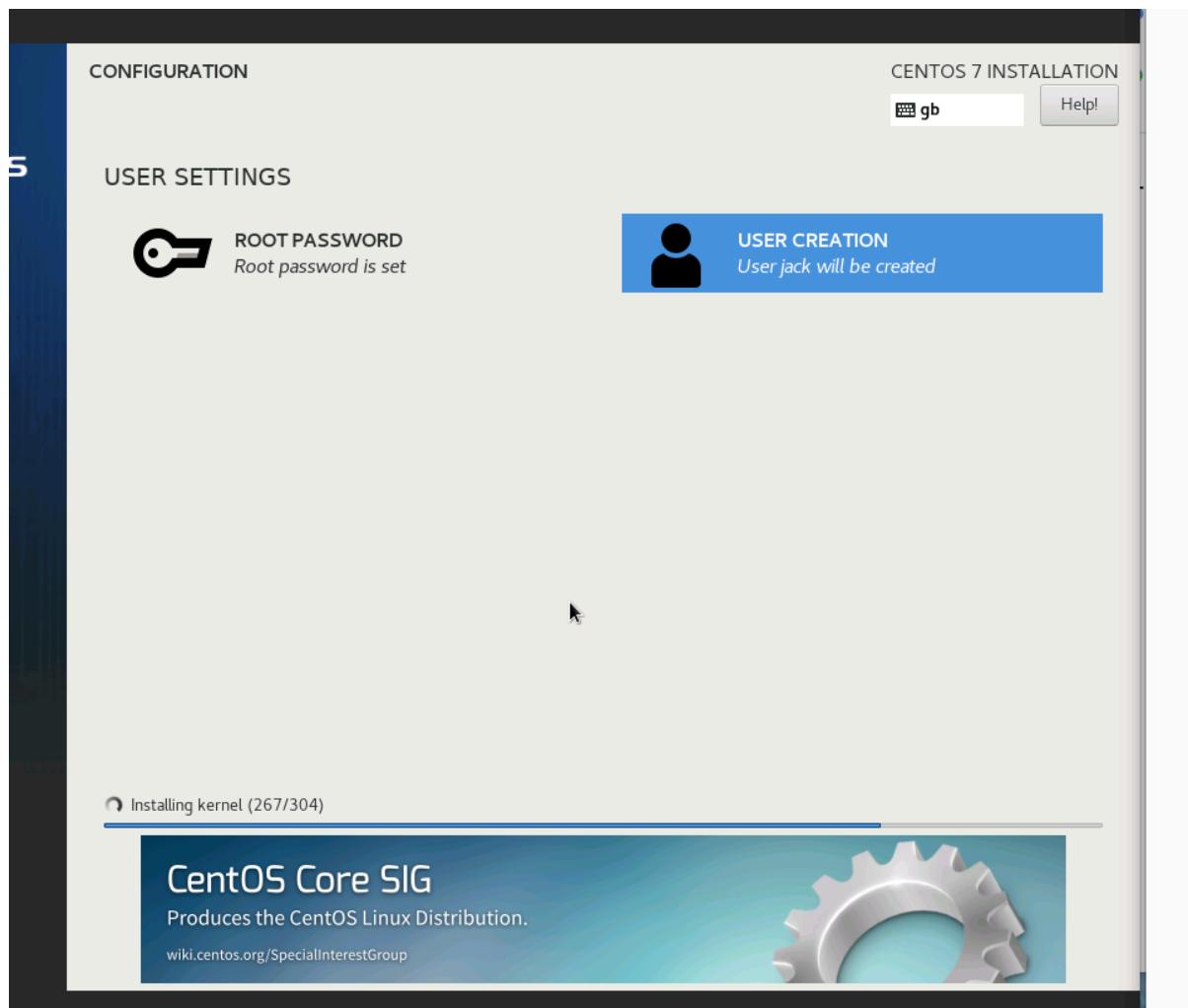
Deploy VM in a specific Datastore

Schedule Actions for VM

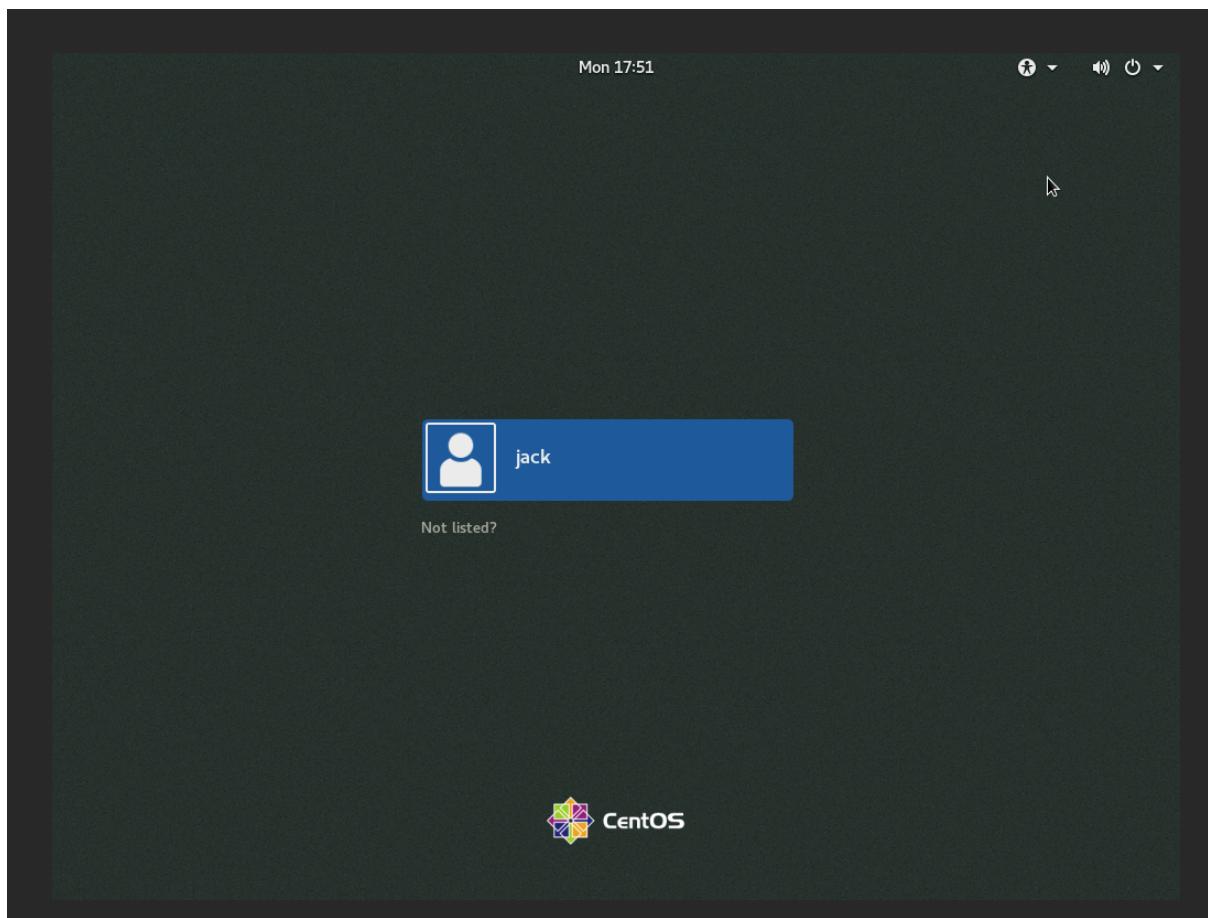
I check to make sure everything is proper before clicking "Create." If the storage devices are not green, we must modify them. This is by creating a new one and changing the template to have that one instead.



And now the virtual machine is up and running



I have began the installation of centos on opennebula, setting the root password to "localhost1"



I am able to get into the VM and get to the login screen.

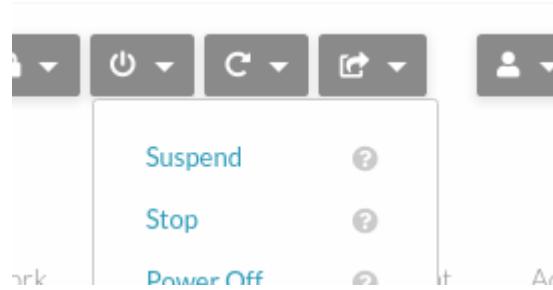
Lab Sheet 5 - Sunstone

For this part I will be using “sunstone” to demonstrate the things you can do with the “admin” account when it comes to messing with the VMs

Sunstone - This is the graphical user interface we have been using to create templates, virtual machines, network interface cards, storage images etc. This is also where we control the virtual machines, the modification of everything happens here it allows us to do what we can do other other softwares like vmware and virtual box

Admin account - The administrator account on sunstone can control everything, its like a root account on a computer. This account can also create other accounts for specific users like school students etc.

Basic usage - VMs

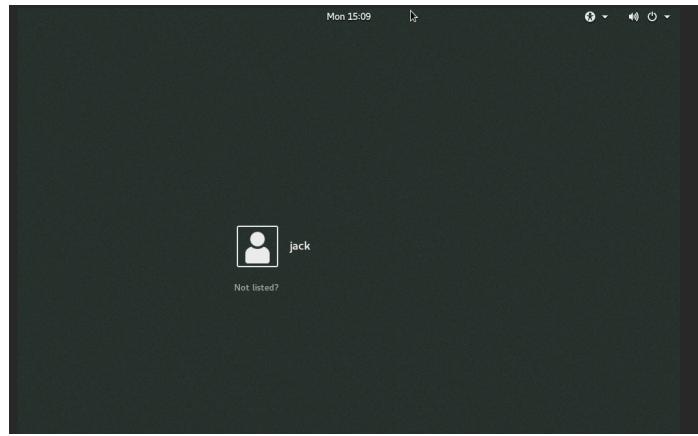


We will click the suspend option for this. Suspend is the ability to freeze the VM on what it is currently running. If we load the VM again it should put us back where we last left the machine.

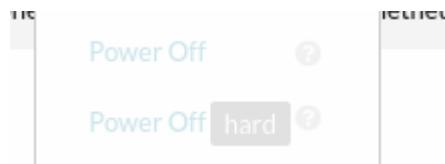
ID	Name	Owner	Group	Status	Host
10	vPO	oneadmin	oneadmin	SUSPENDED	prometh

Showing 1 to 1 of 1 entries

The orange is the colour to show that it has had a change in status, also we can see that in the current status tab it says that it has been suspended.



after we are running the computer from suspended we can see that it just loads its previous session

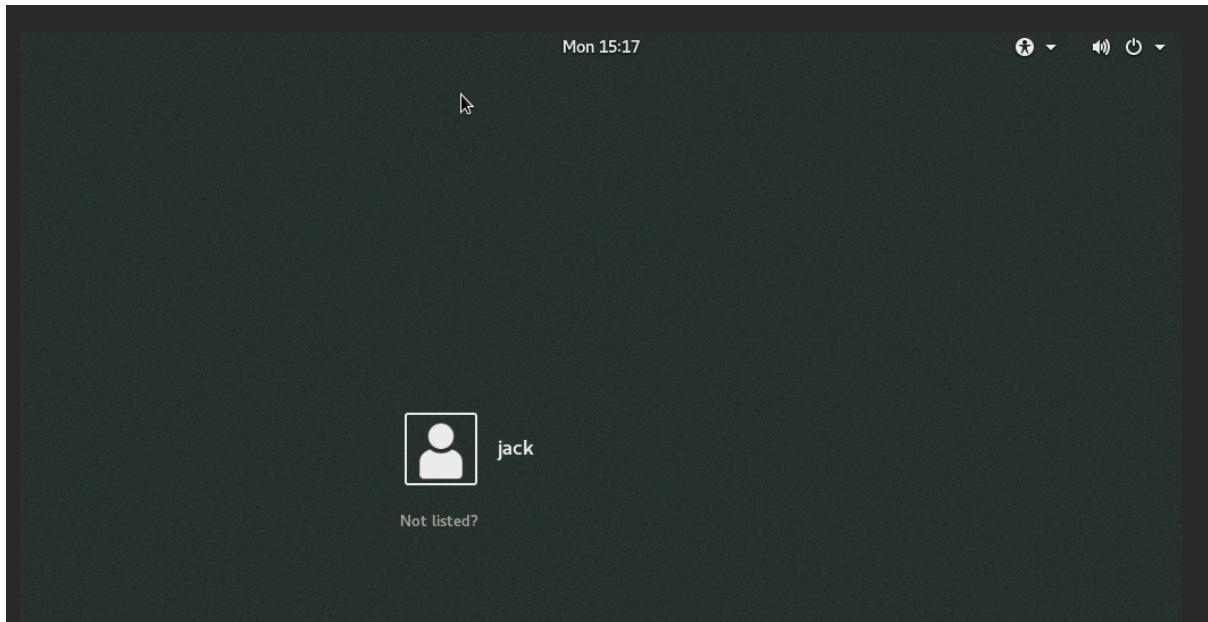


These two options basically do the same thing, power off slowly power off the system by sending it singles, like if we were to suspend the machine. Power off hard does the same thing but the VM is instantly turned off

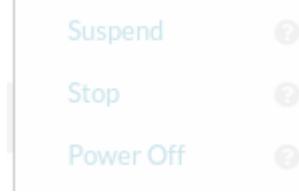
	Name	Owner	Group	Status
<input checked="" type="checkbox"/>	10	vP0	oneadmin	oneadmin POWEROFF pr

Showing 1 to 1 of 1 entries

Yet again we have the colour orange to demonstrate the machine is in the different state, but we also have the "POWEROFF" status to have a readable identifier on the state of the machine



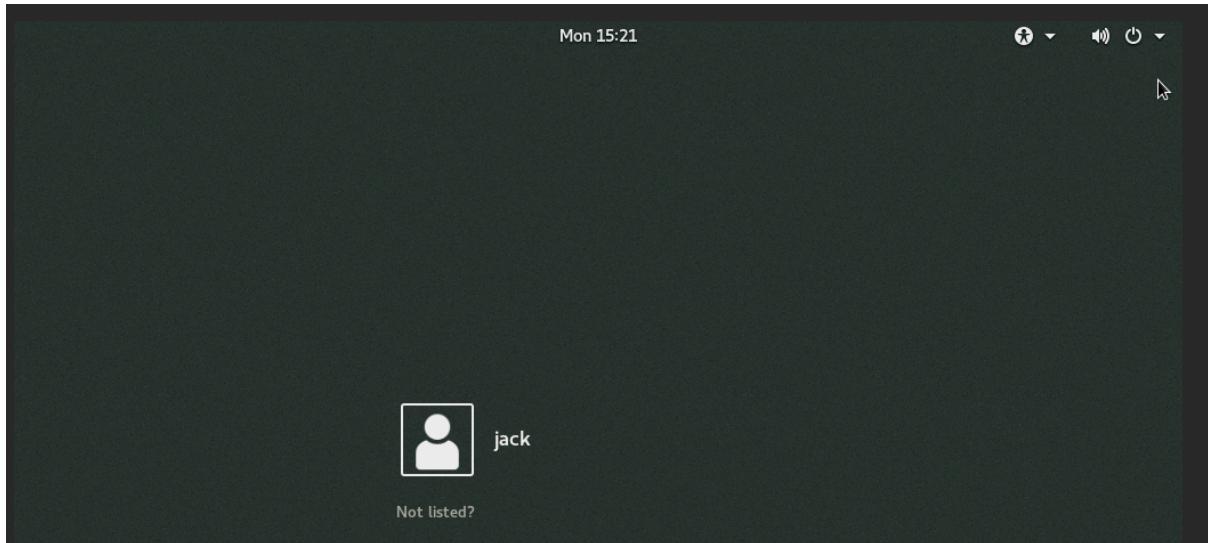
If we was to load the machine it would go through the regular boot sequence till it is back at the login screen waiting for us to enter the information, this is the normal process of just shutting down and turning a computer on. Shown above.



This time we are using the stop command this one is similar to the suspend but the main difference is although it saves the state of the system, it takes to the datastore and is not kept on the host

ID	Name	Owner	Group	Status	Host
10	vPO	oneadmin	oneadmin	STOPPED	--
10		Showing 1 to 1 of 1 entries			
		1 TOTAL		0 ACTIVE	1 OFF
				0 PENDIN	

Again we have the colour orange to show a different state, and we have the status stopped to show it has been stopped. If we resume the machine again it will load us back to where we were before like suspend. Shown below.

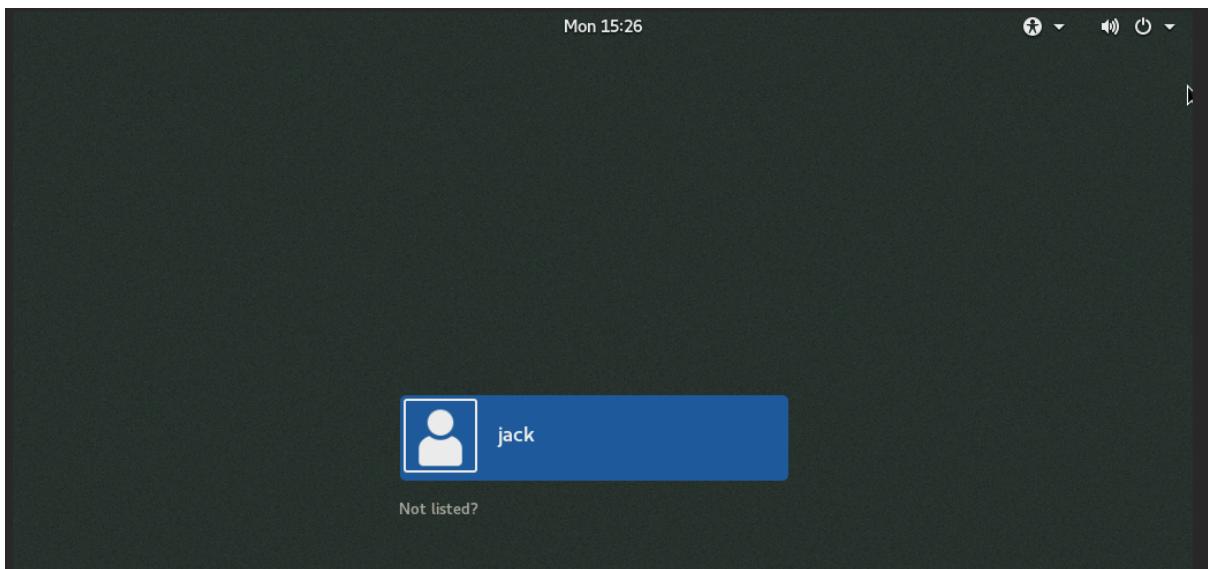




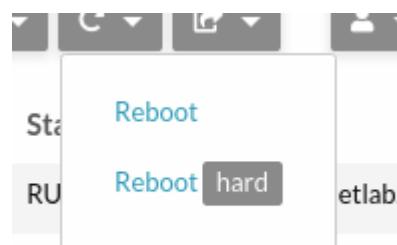
This time we will be using the undeploy hard. Now “undeploy” and “undeploy hard” are quite similar with the hard version it will immediately destroy the running VM, but the non hard version will save it to the data store.

<input type="checkbox"/>	ID	Name	Owner	Group	Status
<input checked="" type="checkbox"/>	10	vPO	oneadmin	oneadmin	UNDEPLOYED
Showing 1 to 1 of 1 entries					
	1	TOTAL	0	ACTIVE	0 OFF

Again we have the colour orange to show a different state, but we also have the undeployed status to show us what option has been picked.



Once running again we can see it has taken us back to the state we left it in.

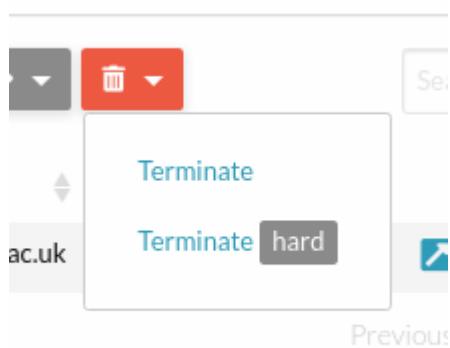
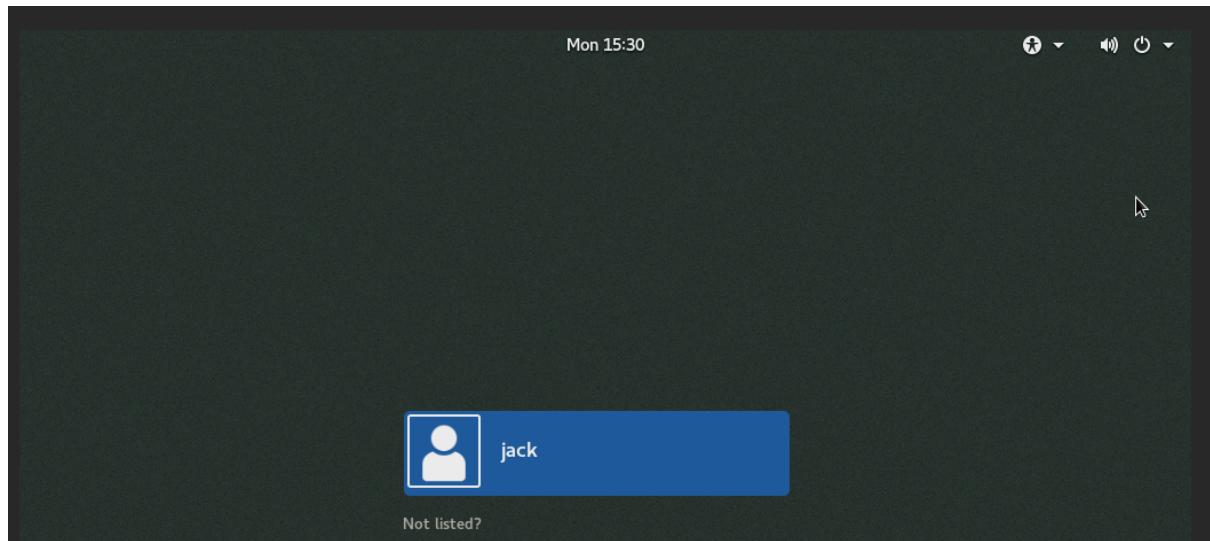


We will be selecting the “reboot hard” option for this one. This option will just reboot the RM like it has been powered off and on.

<input checked="" type="checkbox"/>	ID	Name	Owner	Group	Status	Host
<input checked="" type="checkbox"/>	10	vPO	oneadmin	oneadmin	RUNNING	prometheus.netlab.ee.p

Showing 1 to 1 of 1 entries

For this option there was no change since it does it quite fast. But we can reload the system and see that we will be back at the login screen. Shown below.



This option is equivalent to a kill\delete. It will remove the option to see the virtual machine in the vms tab, but it can still be saved onto the storage devices of the base machine. Sometimes you will need to wipe the drive to make sure that the os on the machine is removed.

VM settings



We will be in the VP0 setting for this segment

Storage

We will be adding a new storage device. I added a name, changed the type to “generic storage data block” gave it “100mb” and changed the format to “qcow2”

Name	Value

ID	Target	Image / Size-Format	Size	Persistent	Actions	Attach disk
0	sda	vPrometheus0	4.6GB/10GB	YES	attach/detach in progress	
1	hda	centos	4.4GB/4.4GB	NO	attach/detach in progress	
2	hdb	Context	1MB/-	NO	attach/detach in progress	
3	sdb	Datablock	-/100MB	NO	attach/detach in progress	

Showing 1 to 4 of 4 entries Previous 1 Next

I have gone and added the new storage device to the VM, i will then reboot it to fully allocate it.

Network card

We will be adding a network interface card.

General Conf Addresses Security QoS Context

Name: NewNic Cluster: 0: default

Description: New nic for test

I have gone and started to create a new network interface card, I have given it a basic name. This is just a placeholder name since we will not be using this in the future. It is just a test.

General Conf Addresses Security QoS Context

AR +

IPv4 IPv4/6 IPv6 Ethernet

First IPv4 address: 192.168.0.1 First MAC address:

Size: 1

Advanced Options

I have also added a generic ip and a size of one, this is because we will be only using one ip for this network interface card

ID	Name	Owner	Group	Reservation
4	NewNic	oneadmin	oneadmin	No
3	Bridge	oneadmin	oneadmin	No

This is the new network interface card, we will be using this on a virtual machine and seeing what happens

ID	Network	IP	MAC	PCI address	IPv6 ULA	IPv6 Global	Actions	Attach nic
0	Bridge	148.197.28.159	02:00:94:c5:1c:9f	--	--	--	attach/detach in progress	
1	NewNic	192.168.0.1	02:00:c0:a8:00:01	--	--	--	attach/detach in progress	

I have gone and attached the new network interface card to the VM. After this I hard reboot the VM to make sure it fully got allocated to the machine

```
[root@localhost ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 02:00:94:c5:1c:9f txqueuelen 1000 (Ethernet)
        RX packets 585 bytes 71136 (69.4 KiB)
        RX errors 0 dropped 7 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet6 fe80::2b36:ad19:5cb5:d0c5 prefixlen 64 scopeid 0x20<link>
        ether 02:00:c0:a8:00:01 txqueuelen 1000 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 28 bytes 4360 (4.2 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

This shows the new network card and the old one since before there was only one, but we can now see two since I added this to the virtual machine that already had a network interface card on it. Shown below

```
57 'abrt-cli status' timed out
[root@localhost ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 02:00:94:c5:1c:9f txqueuelen 1000 (Ethernet)
      RX packets 1074 bytes 131736 (128.6 KiB)
      RX errors 0 dropped 9 overruns 0 frame 0
      TX packets 0 bytes 0 (0.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

This is after I removed the new network interface card since I will not be using it at any point forward

Snapshot

We will be now taking a snapshot of the system as shown above. A snapshot captures the state and data at the specific time the machine was snapped, this is where the snapshot of the machine will be saved.

Info	Capacity	Storage	Network	Snapshots	Placement	Actions	Conf	Template	Log
ID	Name	Timestamp		Actions					
0	Snap1	16:21:48 13/03/2023		snapshot in progress					

But for some reason the snapshot does not actually work, the log below shows the error since they are not able to be made. I still do not know the reason as to why it can not be screen shotted in whatever state I leave it in

```
Mon Mar 13 16:28:45 2023 [Z0][VMM][E]: SNAPSHOTCREATE: error: unsupported configuration: internal
snapshot for disk sda unsupported for storage type raw Could not create snapshot for domain 67d22d99-
3e5c-49eb-a763-3e372c411bbb.
```

Capacity

ID	Target	Image / Size-Format	Size	Persistent	Actions	Attach disk
0	sda	vPrometheus0	4.6GB/10GB	YES		
1	hda	centos	4.4GB/4.4GB	NO		
2	hdb	Context	1MB/-	NO		

Showing 1 to 3 of 3 entries

Previous 1 Next

This was the original starting size of the disk and I have gone and resized it to 20GB marked in red on the image below

ID	Target	Image / Size-Format	Size	Persistent	Actions	Attach disk
0	sda	vPrometheus0	4.6GB/20GB	YES		
1	hda	centos	4.4GB/4.4GB	NO		
2	hdb	Context	1MB/-	NO		

Showing 1 to 3 of 3 entries

Previous 1 Next

```
Mon Mar 13 16:43:01 2023 [Z0][VMM][I]: VM disk successfully resized
Mon Mar 13 16:43:01 2023 [Z0][VM][I]: New LCM state is RUNNING
Mon Mar 13 16:43:01 2023 [Z0][LCM][I]: VM disk resize operation completed.
Mon Mar 13 16:43:08 2023 [Z0][VMM][I]: ExitCode: 0
```

This is the log of the resizing of the hard drive you can see how it says “VM disk resize operation completed”

Permissions

These permissions are being set since later on in this lab sheet we will be creating a student account and making a virtual machine for them. These permissions need to be set so we can actually make the machine

Information		Permissions	Use	Manage	Admin
ID	4	Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Name	NewNic	Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
VLAN ID	--	Other	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
AUTO VLAN ID	NO	Ownership			
OUTER VLAN ID	--	Owner	oneadmin		<input checked="" type="checkbox"/>
AUTO OUTER VLAN ID	NO	Group	oneadmin		<input checked="" type="checkbox"/>

I went and changed the permissions of the new network interface card called “NewNic” Changes marked in red.

Information		Permissions	Use	Manage	Admin
ID	11	Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Name	Datablock	Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Datastore	default	Other	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Register time	16:03:51 13/03/2023	Ownership			
Type	DATABLOCK	Owner	oneadmin		<input checked="" type="checkbox"/>
Persistent	no	Group	oneadmin		<input checked="" type="checkbox"/>
Filesystem type					

The new permissions made for the “datablock” storage image. Changes marked in red

Information		Permissions	Use	Manage	Admin
	CentOS	Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
		Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		Other	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ID	2	Ownership			
Name	vP0	Owner	oneadmin		<input checked="" type="checkbox"/>
Register time	17:00:46 27/02/2023	Group	oneadmin		<input checked="" type="checkbox"/>

The new permissions made for the “VP0” template, this lets the new account gain access to the template and let them make a machine using it. Changes marked in red.

The new permissions made for the “vPrometheus0” storage image, this allows the user to use the storage devices and save there new operating system for the machine on it.

Changes marked in red

Information		Permissions	Use	Manage	Admin
ID	10	Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Name	vPrometheus0	Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Datastore	default	Other	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Register time	17:36:37 27/02/2023	Ownership			

The new permissions for the “centos” storage image, this is the storage image tha has the ISO on it this allows it to be installed on the other storage device. Marked in red

Information		Permissions	Use	Manage	Admin
ID	9	Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Name	centos	Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Datastore	default	Other	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Register time	17:00:21 27/02/2023	Ownership			
Type	CDROM	Owner		oneadmin	<input type="checkbox"/>
Persistent	no	Group		oneadmin	<input type="checkbox"/>
Filesystem type					

Groups

<input type="checkbox"/>	ID	Name	Users	VMs
<input type="checkbox"/>	1	users	0	
<input type="checkbox"/>	0	oneadmin	2	

Showing 1 to 2 of 2 entries

2 TOTAL

I am going to go and create a new group. I have gone to the group section in the system tab. Groups are a set amount of accounts that have an admin user controlling them . Group of user accounts that can have their own hierarchy of who can do what

New Groups are automatically added to the default VDC

Create an administrator user ?

Username
students_admin

Password

Confirm Password

Authentication
core

I have gone and set the user name to “students-admin” and given it the password “localhost1”

The screenshot shows the 'Permissions' tab of a user group configuration. There are several checkboxes for different resources:

- Allow users to view the VMs and Services of other users in the same group ⓘ
- Allow users in this group to create the following resources ⓘ
- VMs:
- VNets:
- Security Groups:
- Virtual Routers:
- Images:
- Templates:
- Documents ⓘ:

Left the permissions as default, since this is all that will be needed for this user group

The screenshot shows a 'Create' dialog for a new group. The 'General' tab is selected, showing the following fields:

- Name: students

Other tabs shown are Views, Admin, and Permissions.

And set the name to students, this will be the main group holder for the specific users

<input type="checkbox"/>	13	User students-admin	Groups	ID 100	manage	All
<input type="checkbox"/>	12	User students-admin	Virtual Routers	All	create	All
<input type="checkbox"/>	11	User students-admin	Virtual Machines, Virtual Networks, Images, VM Templates, Documents, Security Groups, Virtual Routers, VM Groups	Group students	use, manage	All
<input type="checkbox"/>	10	User students-admin	Users	Group students	use, manage, admin, create	All
<input type="checkbox"/>	9	Group students	Virtual Machines, Images, VM Templates, Documents, Security Groups, Virtual Routers	All	create	All
<input type="checkbox"/>	8	Group students	Datastores	All	use	0
<input type="checkbox"/>	7	Group students	Virtual Networks	All	use	0
<input type="checkbox"/>	6	Group students	Hosts	All	manage	0

There were several new ACLs created after the group was founded. This enables the admin account to specify a set of guidelines that regulate what the users are permitted to do to their systems.

Students-admin/VDC

The screenshot shows the OpenNebula interface with a top navigation bar featuring user authentication and system status. Below the navigation is a search bar. The main area displays two tabs: "Memory" and "CPU".

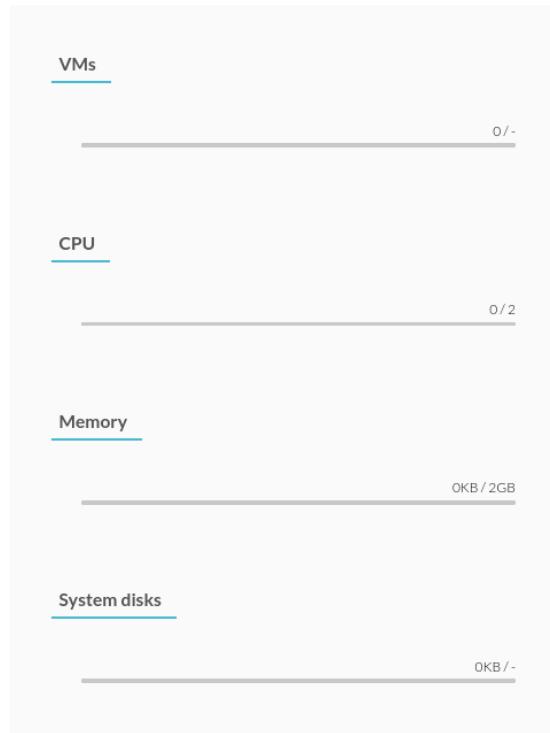
I have gone and logged into the “students-admin” account on opennebula. Here, we can create more user accounts that can be monitored, modified, created etc.

The screenshot shows a user creation form with the following fields:

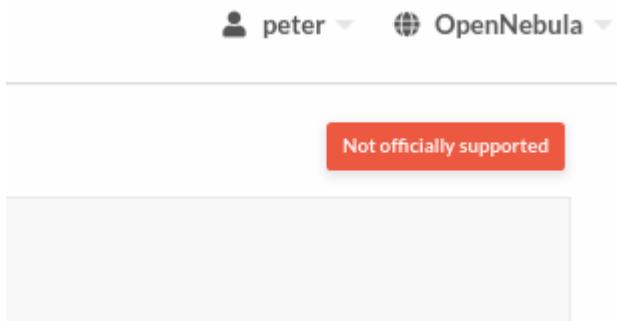
- Username:** peter
- Password:** (redacted)
- Confirm Password:** (redacted)
- Authentication:** core
- Main Group:** Default
- Secondary Groups:** A section with a note: "Please select one or more groups from the list". It includes a search bar and a refresh icon.
- Table:** A table showing existing groups:

ID	Name
100	students
- Pagination:** A footer showing "Showing 1 to 1 of 1 entries", "Previous", "1", and "Next".

I have gone and started to create a user called Peter, I will be demonstrating what you can do with these different accounts



I have gone and given it two CPU cores and 2GB of memory, this should be enough for peter to create a virtual machine according to our templates



Signing into the new student vm account to demonstrate how the user can create a virtual machine

Create Virtual Machine

Peters VM Persistent 1 Create

Template

vPO CentOS

Capacity

Memory 2GB / 25MB
2 GB ▾

Physical CPU 1.00 / 25
1

Virtual CPU 1

Disks

14.4GB / ∞

DISK 0: vPrometheus0
10240 MB ▾

DISK 1: centos
4494 MB ▾

Network

Interface Bridge x

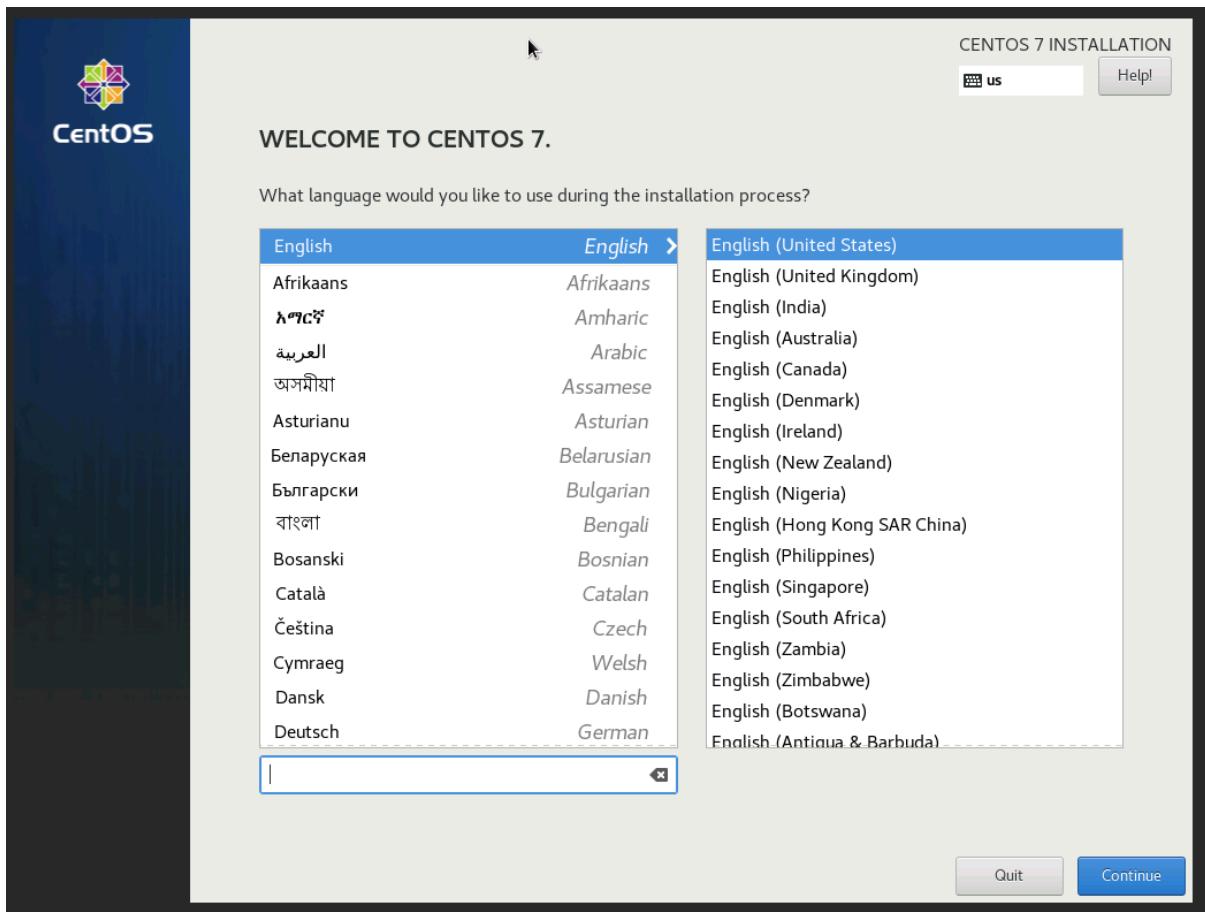
+ Network Interface

VM Group

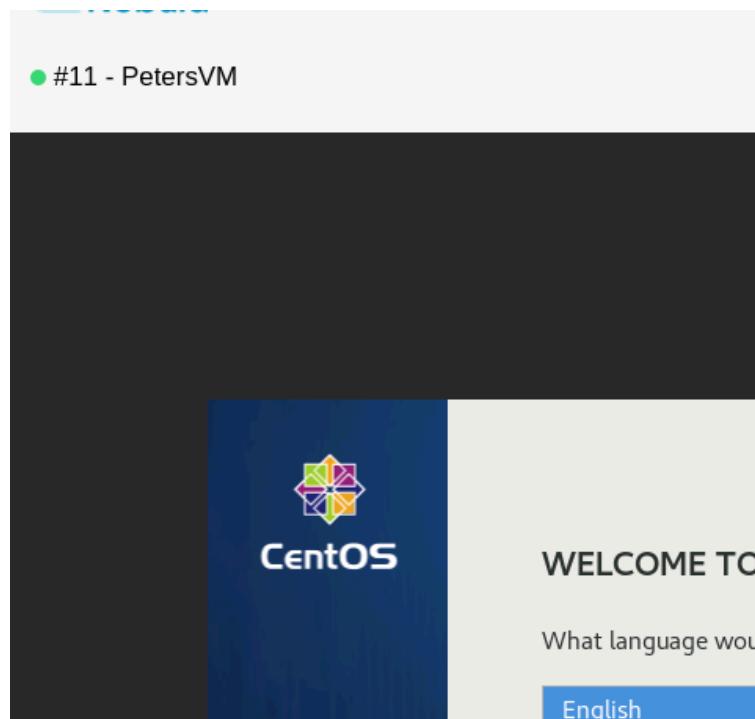
+ VM Group

Datastore

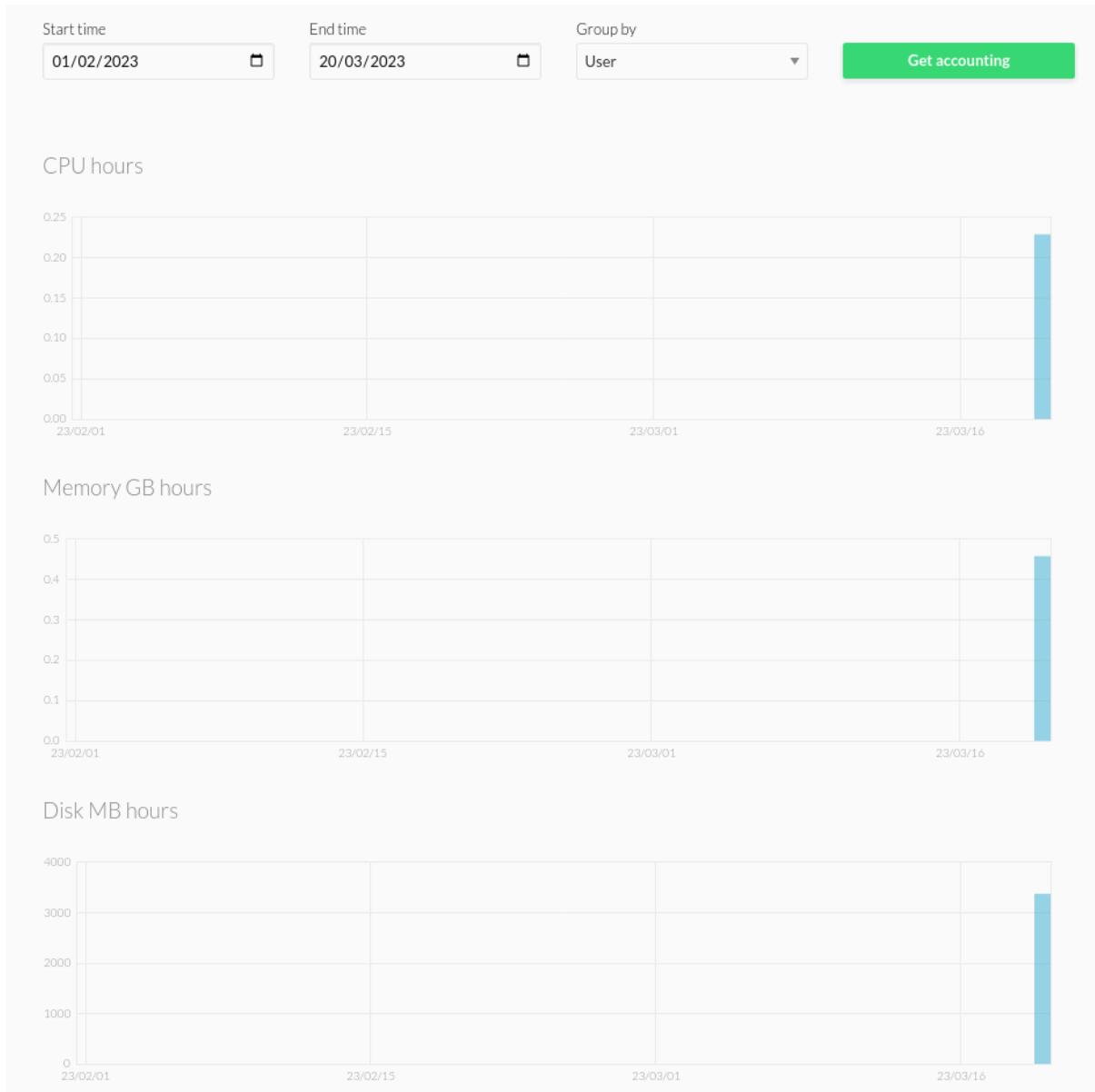
In this image, there are a few issues. I have gone and identified them in red. I went and fixed these by adding a new disc image which allowed me to give the storage to hold the VMs os on it and not have the issue with the "disc0" since it was clashing with the admin VM. I went and reassigned the memory since I accidentally gave too little to the account.



This is the students virtual machine in the installation segment



Proof of it being the virtual machine for peter



This is the accounting table that shows what is used and how long it is being used for. This provides a form of data to see how much of that specific virtual machine is being used and for how long, we can see that:

CPU hours - The CPU has been used for 30 mins on this day

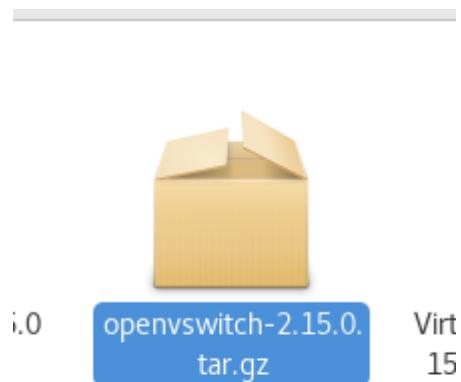
Memory GB hours - The memory has been used for around 30 mins on this day

disk MB hours - Just above 3000mb have been used on the disk on this day

Lab Sheet 6 - OV

OVs, commonly known as Open vSwitch, is an open source switch that offers network characteristics that we can utilise to adjust our system; I will use it to alter my systems and build a virtual network. This lab sheet still adheres to Open Nebula's guidelines.

First I went and installed a file using this link "[openvswitch-2.15.0.tar.gz](#)". This is an installation link that can be used to get the tar file, make sure it is in the "gz" installation format.



This is the file I downloaded; I will be now going and extracting the tar file, which will provide us with a new folder



This is the extracted file, we will be using this later

Now we will be going and installing a few things needed to use OV's, and the commands will be under the images. All these commands will install something needed for us to utilise OV's

```
Installed:  
graphviz.x86_64 0:2.30.1-22.el7  
kernel-devel.x86_64 0:3.10.0-1160.88.1.el7  
python-sphinx.noarch 0:1.1.3-11.el7  
  
Dependency Installed:  
python-babel.noarch 0:0.9.6-8.el7  
python-docutils.noarch 0:0.11-0.3.20130715svn7687.el7  
python-jinja2.noarch 0:2.7.2-4.el7  
python-pillow.x86_64 0:2.0.0-23.gitd1c6db8.el7_9  
python-pygments.noarch 0:1.4-10.el7  
  
Complete!
```

yum install gcc make python-devel kernel-devel graphviz python-sphinx

```
Installing : selinux-policy-devel-3.13.1-268.el7_9.2.noarch  
Installing : libevent-devel-2.0.21-4.el7.x86_64  
Installing : unbound-devel-1.6.6-5.el7_8.x86_64  
Verifying : selinux-policy-devel-3.13.1-268.el7_9.2.noarch  
Verifying : libevent-devel-2.0.21-4.el7.x86_64  
Verifying : unbound-devel-1.6.6-5.el7_8.x86_64  
Verifying : policycoreutils-devel-2.5-34.el7.x86_64  
  
Installed:  
selinux-policy-devel.noarch 0:3.13.1-268.el7_9.2  
unbound-devel.x86_64 0:1.6.6-5.el7_8  
  
Dependency Installed:  
libevent-devel.x86_64 0:2.0.21-4.el7  
policycoreutils-devel.x86_64 0:2.5-34.el7  
  
Complete!  
[root@prometheus ~]#
```

yum install kernel-debug-devel autoconf automake rpm-build openssl-devel

```
Install 1 Package

Total download size: 18 M
Installed size: 38 M
Is this ok [y/d/N]: y
Downloading packages:
kernel-debug-devel-3.10.0-1160.88.1.el7.x86_64.rpm | 18 MB 00:01
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : kernel-debug-devel-3.10.0-1160.88.1.el7.x86_64
  Verifying  : kernel-debug-devel-3.10.0-1160.88.1.el7.x86_64

Installed:
  kernel-debug-devel.x86_64 0:3.10.0-1160.88.1.el7
```

yum install redhat-rpm-config selinux-policy-devel libtool unbound-devel

```
Dependency Installed:
  libunwind.x86_64 2:1.2-2.el7
  python3-libs.x86_64 0:3.6.8-18.el7
  python3-pip.noarch 0:9.0.3-8.el7
  python3-setuptools.noarch 0:39.2.0-10.el7
  python3-sphinx-common.noarch 0:1.2.3-6.el7
  python36-docutils.noarch 0:0.14-1.el7
  python36-jinja2.noarch 0:2.11.1-1.el7
  python36-markupsafe.x86_64 0:0.23-3.el7
  python36-pgments.noarch 0:2.4.2-1.el7
  python36-sphinx.noarch 0:1.2.3-6.el7

Complete!
[root@prometheus ~]#
```

yum install python36 python3-sphinx libunwind-devel

We will now be navigating to the openvswitch rpm file and running some commands, which will be under the images. These commands go and start the process of allowing us to go and start the program and configure some of the files.

```
data@data:~/Downloads$ ./boot.sh
[root@prometheus openvswitch-2.15.0]# ./boot.sh
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, `build-aux'.
libtoolize: copying file `build-aux/ltmain.sh'
libtoolize: putting macros in AC_CONFIG_MACRO_DIR, `m4'.
libtoolize: copying file `m4/libtool.m4'
libtoolize: copying file `m4/ltoptions.m4'
libtoolize: copying file `m4/ltsugar.m4'
libtoolize: copying file `m4/ltversion.m4'
libtoolize: copying file `m4/lt~obsolete.m4'
[root@prometheus openvswitch-2.15.0]#
```

./boot.sh: This command in the Openvswitch section runs a “script” called boot.sh

```
[root@prometheus openvswitch-2.15.0]# ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking how to create a pax tar archive... gnutar
checking for style of include used by make... GNU
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
```

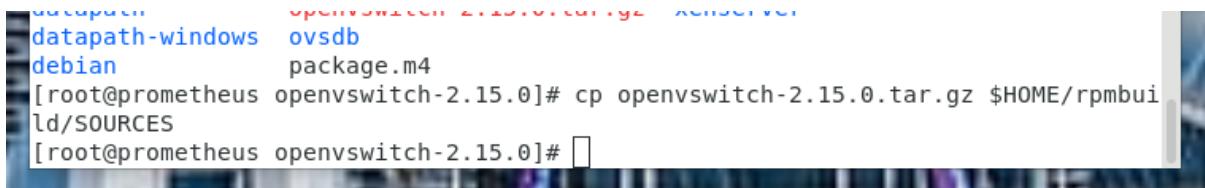
./configure: This command will go and configure options for the application

```
[root@prometheus openvswitch-2.15.0]# make dist
make dist-gzip am_post_remove_distdir=':@'
make[1]: Entering directory `/root/Downloads/openvswitch-2.15.0'
printf '\043 Generated automatically -- do not modify!    -*- buffer-read-only
:t -*-\n' && sed -e 's,[@]VERSION[@],2.15.0,g') < ./rhel/openvswitch-dkms.spec
.in > openvswitch-dkms.spec.tmp || exit 1; if cmp -s openvswitch-dkms.spec.tmp
rhel/openvswitch-dkms.spec; then touch rhel/openvswitch-dkms.spec; rm openvswitch-
dkms.spec.tmp; else mv openvswitch-dkms.spec.tmp rhel/openvswitch-dkms.spec;
fi
printf '\043 Generated automatically -- do not modify!    -*- buffer-read-only
:t -*-\n' && sed -e 's,[@]VERSION[@],2.15.0,g') < ./rhel/kmod-openvswitch-rhel
6.spec.in > kmod-openvswitch-rhel6.spec.tmp || exit 1; if cmp -s kmod-openvswitch-
rhel6.spec.tmp rhel/kmod-openvswitch-rhel6.spec; then touch rhel/kmod-openvswitch-
rhel6.spec; rm kmod-openvswitch-rhel6.spec.tmp; else mv kmod-openvswitch-
rhel6.spec.tmp rhel/kmod-openvswitch-rhel6.spec; fi
```

Make dist: this is used to create a distribution package

```
[root@prometheus openvswitch-2.15.0]# mkdir -p ~/rpmbuild/{SOURCES,BUILD,RPMS,SPECS,SRPMS}
[root@prometheus openvswitch-2.15.0]# ls
acinclude.m4      Documentation          poc
aclocal.m4       include                python
appveyor.yml     ipsec                 README.rst
AUTHORS.rst      lib                   rhel
autom4te.cache   libtool               selinux
boot.sh          LICENSE              stamp-h1
build-aux        m4                   tests
config.h         MAINTAINERS.rst      third-party
config.h.in      Makefile             tutorial
config.h.in~     Makefile.am          utilities
config.log       Makefile.in          Vagrantfile
config.status    manpages.mk          Vagrantfile-FreeBSD
configure        NEWS                 vswitchd
configure.ac     NOTICE              vtep
CONTRIBUTING.rst ofproto              windows
datapath         openvswitch-2.15.0.tar.gz xenserver
datapath-windows ovsdb
debian           package.m4          —
```

`mkdir -p ~/rpmbuild/{SOURCES,BUILD,RPMS,SPECS,SRPMS}`: this command is used to create a parent directory in the rpmbuild folder



```
[root@prometheus ~]# cd /tmp
[root@prometheus ~]# rm -rf openvswitch-2.15.0
[root@prometheus ~]# curl -O https://openvswitch.org/releases/openvswitch-2.15.0.tar.gz
[root@prometheus ~]# tar -xzf openvswitch-2.15.0.tar.gz
[root@prometheus ~]# cd openvswitch-2.15.0
[root@prometheus openvswitch-2.15.0]# ./configure --prefix=/usr --libdir=/usr/lib64
[root@prometheus openvswitch-2.15.0]# make
[root@prometheus openvswitch-2.15.0]# make install
[root@prometheus openvswitch-2.15.0]# cd ..
[root@prometheus ~]# cd /root
[root@prometheus ~]# mkdir rpmbuild
[root@prometheus ~]# cd rpmbuild
[root@prometheus rpmbuild]# mkdir {SOURCES,BUILD,RPMS,SPECS,SRPMS}
[root@prometheus rpmbuild]# ls
SOURCES  BUILD  RPMS  SPECS  SRPMS
```

`cp openvswitch-2.15.0.tar.gz $HOME/rpmbuild/SOURCES`: this command goes and copies the openvswitch gz file and moves it to the rpmbuild sources folder



```
[root@prometheus ~]# cd /root/rpmbuild
[root@prometheus rpmbuild]# ls
SOURCES  BUILD  RPMS  SPECS  SRPMS
[root@prometheus rpmbuild]# cp openvswitch-2.15.0.tar.gz SOURCES
[root@prometheus rpmbuild]# rpmbuild -bb openvswitch.spec
Wrote: /root/rpmbuild/RPMS/x86_64/openvswitch-devel-2.15.0-1.x86_64.rpm
Wrote: /root/rpmbuild/RPMS/noarch/openvswitch-selinux-policy-2.15.0-1.noarch.rpm
Wrote: /root/rpmbuild/RPMS/x86_64/openvswitch-debuginfo-2.15.0-1.x86_64.rpm
Executing(%clean): /bin/sh -e /var/tmp/rpm-tmp.jY2gYa
+ umask 022
+ cd /root/rpmbuild/BUILD
+ cd openvswitch-2.15.0
+ rm -rf /root/rpmbuild/BUILDROOT/openvswitch-2.15.0-1.x86_64
+ exit 0
You have mail in /var/spool/mail/root
[root@prometheus openvswitch-2.15.0]#
```

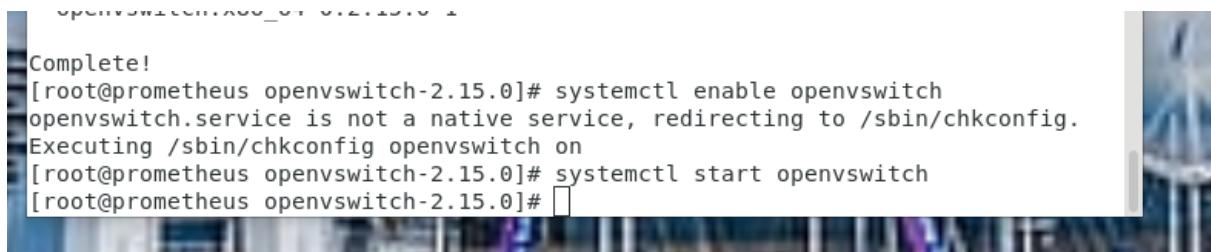
`rpmbuild -bb rhel/openvswitch.spec`

```
=====
Install 1 Package

Total size: 6.4 M
Installed size: 6.4 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : openvswitch-2.15.0-1.x86_64          1/1
  Verifying   : openvswitch-2.15.0-1.x86_64          1/1

Installed:
  openvswitch.x86_64 0:2.15.0-1
```

yum localinstall ~/rpmbuild/RPMS/x86_64/openvswitch-2.15.0-1.x86_64.rpm



I ran both the `systemctl enable` = enabling the new openvswitch application, and the `systemctl start openvswitch` to start the application

```
[root@prometheus openvswitch-2.15.0]# ovs-vsctl show
0c6ac8e0-2b82-4385-b914-fbcf08ea0cb2
    ovs_version: "2.15.0"
[root@prometheus openvswitch-2.15.0]#
```

This final screenshot shows that the application is running and shows the version of the application

OVS Configuration

For this section, we will be making multiple ports, I have only shown vmnet 28, but we will be repeating it for vmnet30, 31, 32. Since we will be making multiple VMs that will use the other vmnets.

```
[root@prometheus openvswitch-2.15.0]# ovs-vsctl add-br vmnet28
```

The command above goes and adds the bridge called vmnet28; we will use this instead of br0 from now on

```
[root@prometheus openvswitch-2.15.0]# ovs-vsctl add-port vmnet28 net28 -- set interface net28 type=internal
```

This command goes and add the two ports, “vmnet28” and “net28,” and sets the interface for net28 to the type internal

```
ovs-vsctl: the default log directory is /var/log/openvswitch.
[root@prometheus openvswitch-2.15.0]# ovs-vsctl add-port vmnet28 ens33
ovs-vsctl: cannot create a port named ens33 because a port named ens33 already exists on bridge vmnet28
```

We have to add the port for “vmnet28” ens33 but we can do this since it is already a thing on the system, this is shown by the error code at the bottom with how it can not create

```
[root@prometheus openvswitch-2.15.0]# ovs-vsctl show
0c6ac8e0-2b82-4385-b914-fbcf08ea0cb2
  Bridge vmnet28
    Port net28
      Interface net28
        type: internal
    Port vmnet28
      Interface vmnet28
        type: internal
    Port ens33
      Interface ens33
        error: "could not add network device ens33 to ofproto (Device or resource busy)"
  ovs_version: "2.15.0"
```

I went and checked what was causing this by running the command “ovs-vsctl show” it shows us all the ports and interfaces we have present in the ovs application

```
[root@prometheus openvswitch-2.15.0]# locate ifcfg-br0
/etc/sysconfig/network-scripts/ifcfg-br0
[root@prometheus openvswitch-2.15.0]# cd /etc/sysconfig/network-scripts
[root@prometheus network-scripts]# ls
ifcfg-br0  ifdown-bnep  ifdown-ipv6  ifdown-ppp   ifdown-TeamPort  ifup-bnep  ifup-ipv6  ifup-plusb  ifup-sit      ifup-wireless
ifcfg-ens33 ifdown-eth  ifdown-isdn  ifdown-routes ifdown-tunnel  ifup-eth   ifup-isdn  ifup-post   ifup-Team     init.ipv6-global
ifcfg-lo   ifdown-ib   ifdown-ovs   ifdown-sit   ifup          ifup-ib   ifup-ovs   ifup-ppp   ifup-TeamPort  network-functions
ifdown    ifdown-ipp  ifdown-post  ifdown-Team   ifup-aliases   ifup-ipp  ifup-plip  ifup-routes ifup-tunnel  network-functions-ipv6
[root@prometheus network-scripts]# mv ifcfg-br0 ifcfg-net28
[root@prometheus network-scripts]# ls
ifcfg-ens33 ifdown-bnep  ifdown-ipv6  ifdown-ppp   ifdown-TeamPort  ifup-bnep  ifup-ipv6  ifup-plusb  ifup-sit      ifup-wireless
ifcfg-lo   ifdown-eth  ifdown-isdn  ifdown-routes ifdown-tunnel  ifup-eth   ifup-isdn  ifup-post   ifup-Team     init.ipv6-global
ifcfg-net28 ifdown-ib   ifdown-ovs   ifdown-sit   ifup          ifup-ib   ifup-ovs   ifup-ppp   ifup-TeamPort  network-functions
ifdown    ifdown-ipp  ifdown-post  ifdown-Team   ifup-aliases   ifup-ipp  ifup-plip  ifup-routes ifup-tunnel  network-functions-ipv6
[root@prometheus network-scripts]#
```

I went and used the locate command too look for “ifcfg-br0” we need to find this file so we can change the content to match our new vmnet28, we will also create more files for the other vmnets “30, 31, 32”

```
[root@prometheus network-scripts]# ovs-vsctl add-br vmnet30
[root@prometheus network-scripts]# ovs-vsctl add-port vmnet30 net30 --set interface net30 type=internal
ovs-vsctl: Port does not contain a column whose name matches "--set"
[root@prometheus network-scripts]# ovs-vsctl add-port vmnet30 net30 -- set interface net30 type=internal
[root@prometheus network-scripts]# ovs-vsctl show
0c6ac8e0-2b82-4385-b914-fbcf08ea0cb2
    Bridge vmnet28
        Port net28
            Interface net28
                type: internal
        Port vmnet28
            Interface vmnet28
                type: internal
        Port ens33
            Interface ens33
    Bridge vmnet30
        Port net30
            Interface net30
                type: internal
        Port vmnet30
            Interface vmnet30
                type: internal
ovs_version: "2.15.0"
```

```

TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=net28
DEVICE=net28
ONBOOT=yes
IPADDR=148.197.28.209
PREFIX=24
GATEWAY=148.197.28.254
DNS1=148.197.28.1
DOMAIN=netlab.ee.port.ac.uk
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_PRIVACY=no
NM_CONTROLLED=no
~
```

```

TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=net30
DEVICE=net30
ONBOOT=yes
IPADDR=148.197.30.81
PREFIX=30
GATEWAY=148.197.28.254
DNS1=148.197.28.1
DOMAIN=netlab.ee.port.ac.uk
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_PRIVACY=no
NM_CONTROLLED=no
```

On the left is vmnet 28, this is a new file that is present in the network scripts folder, this is so the system will use. When you type ifconfig in the system you will see it present after we have finished this for all files

On the right is the new file and its content for Vmnet30

```

TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=net31
DEVICE=net31
ONBOOT=yes
IPADDR=148.197.31.81
PREFIX=30
GATEWAY=148.197.28.254
DNS1=148.197.28.1
DOMAIN=netlab.ee.port.ac.uk
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_PRIVACY=no
NM_CONTROLLED=no
~
```

```

OTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=net32
DEVICE=net32
ONBOOT=yes
IPADDR=148.197.32.81
PREFIX=30
GATEWAY=148.197.28.254
DNS1=148.197.28.1
DOMAIN=netlab.ee.port.ac.uk
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_PRIVACY=no
NM_CONTROLLED=no
```

On the left is the new content file for vmnet31 and on the right is the new content file for vmnet32

```
[root@prometheus network-scripts]# cat ifcfg-ens33
DEVICE=ens33
BOOTPROTO=none
NM_CONTROLLED=no
ONBOOT=yes
TYPE=Ethernet
BRIDGE=br0
[root@prometheus network-scripts]# ]
```

This is the original content of the file that we have duplicated and changed, we will be leaving this one the same since we will not be using it anyways

```
[root@prometheus network-scripts]# service network restart
Restarting network (via systemctl): ] [ OK ]
```

We will be going and restraining the network service this is so we can have these four new networks we made active and shown in the systems ifconfig

```
net28: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 148.197.28.209 netmask 255.255.255.0 broadcast 148.197.28.255
      inet6 fe80::6c49:37ff:fea5:973e prefixlen 64 scopeid 0x20<link>
        ether 6e:49:37:a5:97:3e txqueuelen 1000 (Ethernet)
          RX packets 662 bytes 110851 (108.2 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 368 bytes 184356 (180.0 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

This is the new net28 network that we configred in the previous images

```
net30: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 148.197.30.80 netmask 255.255.255.0 broadcast 148.197.30.255
      inet6 fe80::cceb:bfff:fe1:f321 prefixlen 64 scopeid 0x20<link>
        ether ce:eb:3b:e1:f3:21 txqueuelen 1000 (Ethernet)
          RX packets 0 bytes 0 (0.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 165 bytes 24444 (23.8 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

net31: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 148.197.31.81 netmask 255.255.255.0 broadcast 148.197.31.255
      inet6 fe80::7c84:41ff:fe17:a907 prefixlen 64 scopeid 0x20<link>
        ether 7e:84:41:17:a9:07 txqueuelen 1000 (Ethernet)
          RX packets 0 bytes 0 (0.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 86 bytes 13198 (12.8 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

net32: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 148.197.32.82 netmask 255.255.255.0 broadcast 148.197.32.255
      inet6 fe80::8441:43ff:fee8:270d prefixlen 64 scopeid 0x20<link>
        ether 86:41:43:e8:27:0d txqueuelen 1000 (Ethernet)
          RX packets 0 bytes 0 (0.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 25 bytes 3747 (3.6 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

These are the other networks that we configured showing up after typing ifconfig

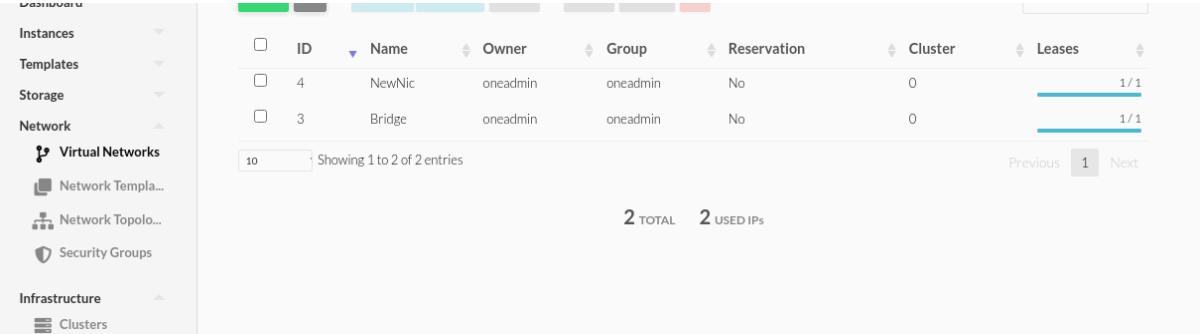
```
[root@prometheus network-scripts]# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 fe80::20c:29ff:feb7:86dd prefixlen 64 scopeid 0x20<link>
          ether 00:0c:29:b7:86:dd txqueuelen 1000 (Ethernet)
            RX packets 982278 bytes 827930445 (789.5 MiB)
            RX errors 0 dropped 723 overruns 0 frame 0
            TX packets 501169 bytes 207448435 (197.8 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

This is ens33 that we left since we will not be using it, it is still here since we have not changed anything about it

```
[root@prometheus network-scripts]# ovs-vsctl add-br vmnet30
[root@prometheus network-scripts]# ovs-vsctl add-port vmnet30 net30 --set interface net30 type=internal
ovs-vsctl: Port does not contain a column whose name matches "--set"
[root@prometheus network-scripts]# ovs-vsctl add-port vmnet30 net30 -- set interface net30 type=internal
[root@prometheus network-scripts]# ovs-vsctl show
0c6ac8e0-2b82-4385-b914-fbcf08ea0cb2
  Bridge vmnet28
    Port net28
      Interface net28
        type: internal
    Port vmnet28
      Interface vmnet28
        type: internal
    Port ens33
      Interface ens33
  Bridge vmnet30
    Port net30
      Interface net30
        type: internal
    Port vmnet30
      Interface vmnet30
        type: internal
  ovs_version: "2.15.0"
```

This is the evidence of all of my other added ports all shown in the image under the “ovs-vsctl show” section.

Networking in the cloud



	ID	Name	Owner	Group	Reservation	Cluster	Leases
<input type="checkbox"/>	4	NewNic	oneadmin	oneadmin	No	0	<div style="width: 100%;">1/1</div>
<input type="checkbox"/>	3	Bridge	oneadmin	oneadmin	No	0	<div style="width: 100%;">1/1</div>

Showing 1 to 2 of 2 entries

2 TOTAL 2 USED IPs

I have gone and deleted bridge after detaching it from my admin virtual machine since we will no longer be needing it this point forward.

Name

Description

Making a new network interface giving it the name net28, this will link to the new net28 in the network scripts.

Bridge [?](#)

vmnet28

Network mode

Open vSwitch ▼

Open vSwitch, restrict network access with Open vSwitch Virtual Switch. Security Groups are not applied.

MAC spoofing filter

IP spoofing filter

VLAN ID	Physical device ?
No VLAN network ▼	

Have gone and added a network bridge specification called vmnet28 since that's what's in the network scripts, changed the network mode to Open vSwitch, and set the vlan id to No Vlan Network.

IPv4 IPv4/6 IPv6 Ethernet

First IPv4 address <input style="background-color: #e0f2fd; border: 1px solid #ccc; padding: 2px; width: 100%;" type="text" value="148.197.28.159"/>	First MAC address <input style="border: 1px solid #ccc; padding: 2px; width: 100%;" type="text"/>
Size <input style="width: 20px; height: 20px;" type="text" value="1"/> ▼	
▼ Advanced Options	

I went and set the IPv4 address and the size of one for the network

I went and added the network address, network mask, and gateway for this new network interface, these correlate to the ones in the network scripts a little

Network address 148.197.28.159	Network mask 255.255.255.0
Gateway 148.197.28.254	IPv6 Gateway
DNS 148.197.28.1	MTU of the Guest interfaces
<u>Custom attributes</u>	

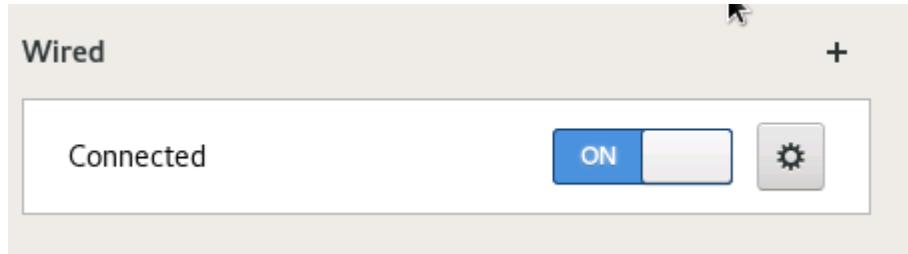
I did this 3 more times for the other three networks. I will be adding the other interfaces with changes retrospective to their ips.

<input type="checkbox"/>	8	net32	oneadmin	oneadmin	No	0	0/1
<input type="checkbox"/>	7	net31	oneadmin	oneadmin	No	0	0/1
<input type="checkbox"/>	6	net30	oneadmin	oneadmin	No	0	0/1
<input type="checkbox"/>	5	net28	oneadmin	oneadmin	No	0	0/1

Even if they are connected to the virtual machine we need to go and do one more thing. After we load the virtual machine with this network interface card we have to go and manually type it into the machine to make sure it fully connects since it can not use dhcp to automatically connect and apply the network settings

Wired		Cancel	Apply	
Details	Identity	IPv4	IPv6	Security
IPv4 Method	<input type="radio"/> Automatic (DHCP)	<input type="radio"/> Link-Local Only		
	<input checked="" type="radio"/> Manual	<input type="radio"/> Disable		
Addresses				
Address	Netmask	Gateway		
148.197.28.159	255.255.255.0	148.197.28.254	<input type="button" value="x"/>	
			<input type="button" value="x"/>	
DNS	Automatic	<input type="button" value="ON"/>		
148.197.28.1				
Separate IP addresses with commas				

Evidence of me tying it into the actual virtual machine to make sure it fully goes and connects to the system



It was now able to connect and be used, but it does not have an actual network connection. I will be going and fixing that in the next section of this labsheet

Dynamic routing

```
[root@prometheus ~]# yum install quagga
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: anorien.csc.warwick.ac.uk
 * epel: cdn.centos.no
 * extras: mirror.quickhost.uk
 * updates: mirror.cov.ukservers.com
Package quagga-0.99.22.4-5.el7_4.x86_64 already installed and latest version
Nothing to do
[root@prometheus ~]#
```

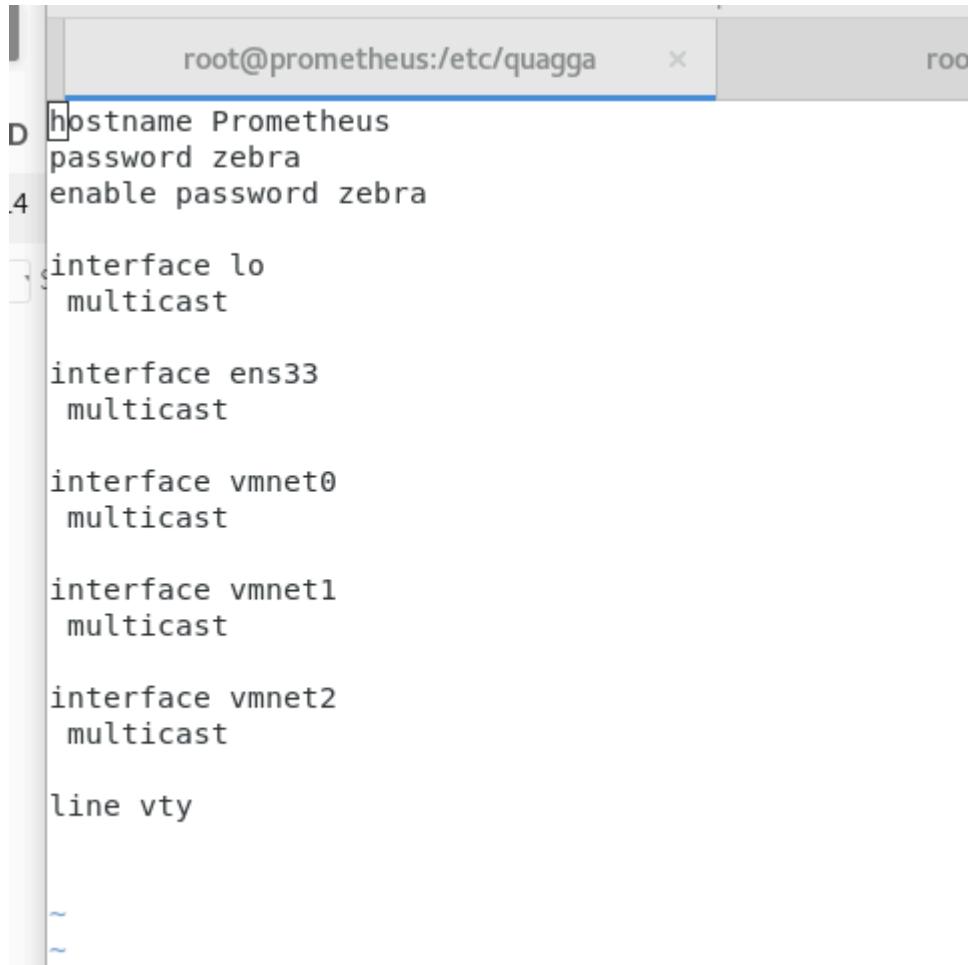
I have gone and installed quagga using “yum install quagga” we will be using this to create a routing method that can transfer to other network cards on our system and allow our virtual machines to gain proper network connection

```
[root@prometheus quagga]# ls
ospfd.conf  vtysh.conf  zebra.conf
[root@prometheus quagga]#
```

There are two files that we will be making some content changes to. This is so the files work for what we want it to do, to create this dynamic route.

```
root@prometheus:/etc/quagga
root@pro
ID hostname Prometheus
password zebra
14 router ospf
  ospf router-id 148.197.28.209
  network 148.197.28.0/24 area 0
  network 148.197.30.80/30 area 0
  network 148.197.31.80/30 area 0
  network 148.197.32.80/30 area 0
log stdout
line vty
```

This is the new version of the zebra file and we went and mainly changed the “/30” section to make sure it meets our systems requirements



The screenshot shows a terminal window with the title bar "root@prometheus:/etc/quagga". The terminal content displays the configuration of the OSPFD daemon. The configuration includes setting the hostname to "Prometheus", defining a password for the Zebra daemon, and enabling OSPF on various interfaces (lo, ens33, vmnet0, vmnet1, vmnet2) with multicast support. It also defines a VTY line for remote access.

```
hostname Prometheus
password zebra
enable password zebra

interface lo
    multicast

interface ens33
    multicast

interface vmnet0
    multicast

interface vmnet1
    multicast

interface vmnet2
    multicast

line vty
```

This is the new content of the OSPFD file, we have removed any comments that stop anything from running

systemctl enable zebra; systemctl enable ospfd; systemctl start zebra; systemctl start ospfd

I have gone and ran the commands above in red, this command goes and enables zebra and ospfd and starts both of them and now I can go and start dynamic routing for them and hopefully get one of my cloud machines to gain network access

```
root@prometheus:~/etc/quagga
File Edit View Search Terminal Tabs Help
root@prometheus:/etc/quagga x root@prometheus:~ x
[1] [root@prometheus quagga]# systemctl enable zebra; systemctl enable ospfd; systemctl start zebra; systemctl start ospfd
[2] Created symlink from /etc/systemd/system/multi-user.target.wants/zebra.service to /usr/lib/systemd/system/zebra.service.
[3] Created symlink from /etc/systemd/system/multi-user.target.wants/ospfd.service to /usr/lib/systemd/system/ospfd.service.
[4] [root@prometheus quagga]# 
```

The screenshot above is the command in action

```
bad argument '-r'
Try `iptables -h' or 'iptables --help' for more information.
[root@prometheus quagga]# iptables -F
[root@prometheus quagga]# 
```

This image show the command “`iptables -F`”, This is the command that goes and disables the firewall on the system

Next we will be running this command “`/sbin/sysctl -w net.ipv4.ip_forward=1`”, this command goes and enables a form of port forwarding that we will need so that the other virtual machine can gain network access and ping google

```
net.ipv4.ip_forward = 1
[root@prometheus quagga]# /sbin/sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
[root@prometheus quagga]# 
```

Evidence of the command being ran, it does not reply with anything

```
[root@prometheus quagga]# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
0.0.0.0         148.197.28.254  0.0.0.0       UG    0      0      0 net28
148.197.27.0   148.197.28.115  255.255.255.0  UG    20     0      0 net28
148.197.28.0   0.0.0.0        255.255.255.0  U     0      0      0 net28
148.197.29.0   148.197.28.115  255.255.255.0  UG    20     0      0 net28
148.197.30.4   148.197.28.165  255.255.255.252 UG   20     0      0 net28
148.197.30.72  148.197.28.208  255.255.255.252 UG   20     0      0 net28
148.197.30.80  0.0.0.0        255.255.255.252 U     0      0      0 net30
148.197.30.88  148.197.28.210  255.255.255.252 UG   20     0      0 net28
148.197.30.128 148.197.28.217  255.255.255.252 UG   20     0      0 net28
148.197.30.168 148.197.28.224  255.255.255.252 UG   20     0      0 net28
148.197.31.4   148.197.28.165  255.255.255.252 UG   20     0      0 net28
148.197.31.72  148.197.28.208  255.255.255.252 UG   20     0      0 net28
148.197.31.80  0.0.0.0        255.255.255.252 U     0      0      0 net31
148.197.31.88  148.197.28.210  255.255.255.252 UG   20     0      0 net28
148.197.31.128 148.197.28.217  255.255.255.252 UG   20     0      0 net28
148.197.31.168 148.197.28.224  255.255.255.252 UG   20     0      0 net28
148.197.32.4   148.197.28.165  255.255.255.252 UG   20     0      0 net28
148.197.32.72  148.197.28.208  255.255.255.252 UG   20     0      0 net28
148.197.32.80  0.0.0.0        255.255.255.252 U     0      0      0 net32
148.197.32.88  148.197.28.210  255.255.255.252 UG   20     0      0 net28
148.197.32.128 148.197.28.217  255.255.255.252 UG   20     0      0 net28
148.197.32.168 148.197.28.224  255.255.255.252 UG   20     0      0 net28
148.197.34.0   148.197.28.215  255.255.255.0   UG   20     0      0 net28
169.254.0.0    0.0.0.0        255.255.0.0     U     1002    0      0 ens33
169.254.0.0    0.0.0.0        255.255.0.0     U     1004    0      0 net31
169.254.0.0    0.0.0.0        255.255.0.0     U     1006    0      0 net32
169.254.0.0    0.0.0.0        255.255.0.0     U     1008    0      0 net28
169.254.0.0    0.0.0.0        255.255.0.0     U     1010    0      0 net30
172.16.147.0   0.0.0.0        255.255.255.0   U     0      0      0 vmnet8
192.168.11.0   0.0.0.0        255.255.255.0   U     0      0      0 vmnet1
192.168.122.0  0.0.0.0        255.255.255.0   U     0      0      0 virbr0
[root@prometheus quagga]#
```

The image above is what is shown after running the command “route -n” this lets us see all possible route on the network

```
[root@stewart ~]# ping google.com
PING google.com (142.250.187.206) 56(84) bytes of data.
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=1 ttl=116 time=5.96 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=2 ttl=116 time=10.4 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=3 ttl=116 time=6.26 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=4 ttl=116 time=6.32 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=5 ttl=116 time=6.18 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=6 ttl=116 time=6.19 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=7 ttl=116 time=5.61 ms
64 bytes from lhr25s33-in-f14.1e100.net (142.250.187.206): icmp_seq=8 ttl=116 time=7.07 ms
```

After all of this I go and go on one of my virtual machines and as we can see i can now ping google from the machine.

Populating the cloud

This section of the labsheet we will be going and populating our cloud network with different types of machines and seeing how they work on the network

Name

Hypervisor
 KVM vCenter LXC Firecracker

Description

Logo


Memory
 GB

Enable hot resize?

Max memory
 GB

Memory modification

Physical CPU

CPU modification

Virtual CPU

Max VCPU

VCPU modification

Cost COST / MONTH

0.00 COST / MONTH

Total: 0.00 COST / MONTH

Memory
 MB

CPU

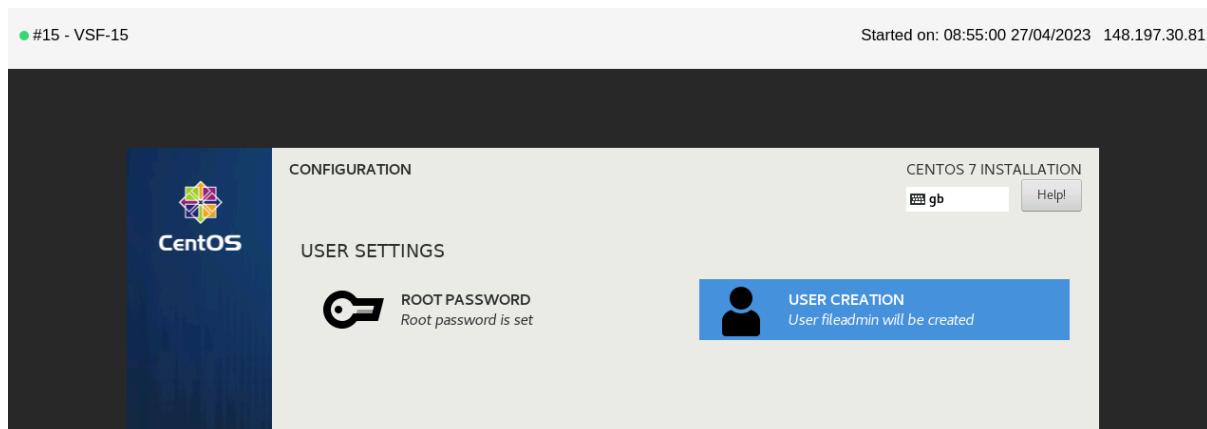
Disk

Backups

Backup Frequency in Seconds

Marketplace ID

Firstly started off with making a new template for the new system i was making, named it VSF so it is identifiable in the VMs tab. Created a root password and also a user account called “FileAdmin”, but this account will not be used for configuration purposes.



References

Contributor, T. (2021, October 27). What is paravirtualization? definition from searchservervirtualization.

IT Operations.

<https://www.techtarget.com/searchitoperations/definition/paravirtualization>

GeeksforGeeks. (2021, August 25). Virtualization: Vmware: Full virtualization.

GeeksforGeeks.

<https://www.geeksforgeeks.org/virtualization-vmware-full-virtualization/>

Hard disk. (n.d.).

https://download.parallels.com/desktop/v12/docs/en_US/Parallels%20Desktop%20User%27s%20Guide/33140.htm

Hunter, A. (2021, June 1). Vdi vs VHD VS VMDK: Learn the different virtual disk formats.

Parallels Remote Application Server Blog - Application virtualization, mobility and VDI. <https://www.parallels.com/blogs/ras/vdi-vs-vhd-vs-vmdk/>

Umair, K. (2022, June 10). Deep Dive in virtualization: All you need to know.

StoneFly.

<https://stonefly.com/blog/deep-dive-virtualization>

What is hardware virtualization? how hardware virtualization works - citrix.

Citrix.com. (n.d.).

<https://www.citrix.com/solutions/vdi-and-daas/what-is-hardware-virtualization.html#:~:text=Hardware%20virtualization%20is%20the%20method>