KVM VIRTUALIZATION IN CENTOS 7

Table of Contents

[KVM VIRTUALIZATION IN CENTOS 7 1](#_Toc513558113)

[1 Introduction 2](#_Toc513558114)

[2 KVM Architecture 2](#_Toc513558115)

[2.1 Quick emulator(QEMU) 2](#_Toc513558116)

[2.2 Libguestfs 3](#_Toc513558117)

[2.3 Libvirt 3](#_Toc513558118)

[3 Preparing the environment 3](#_Toc513558119)

[3.1 System requirement 3](#_Toc513558120)

[3.2 Virtualization 3](#_Toc513558121)

[3.3 KVM Installation 4](#_Toc513558122)

[3.4 Disk Space 5](#_Toc513558123)

[3.5 Networking 6](#_Toc513558124)

[4 Creating Virtual Machine 6](#_Toc513558125)

[5 Clone Virtual Machine 10](#_Toc513558126)

[6 Manage Virtual Machine 11](#_Toc513558127)

[6.1 Common tasks 11](#_Toc513558128)

[6.2 Accessing virtual Machine. 13](#_Toc513558129)

[6.3 Snapshots 13](#_Toc513558130)

[7 Changing Virtual Machine Configuration 14](#_Toc513558131)

[7.1 Changing memory 14](#_Toc513558132)

[7.2 Changing vcpus 16](#_Toc513558133)

[7.3 Changing the disk capacity. 16](#_Toc513558134)

[7.4 Network Interface Card(NIC) 17](#_Toc513558135)

[7.4.1 Adding a network interface card 17](#_Toc513558136)

[7.4.2 Detaching the network interface card 18](#_Toc513558137)

[8 Deleting the virtual machine 18](#_Toc513558138)

# Introduction

Kernel-based virtual machine(kvm) is a virtualization infrastructure which turns the kernel into a hypervisor. The hypervisor is a program which allows multiple operating system to share a single hardware, here linux kernel acts as a hypervisor. KVM reuses the linux kernel features like CPU scheduling, memory management, power management, drivers and many more. KVM needs a processor which has virtualization extension supports only on x86 processors.

# KVM Architecture

VM2

VM1

QEMU I/O

X86 hardware(VT-X/AMD-V)

Hypervisor kvm(/dev/kvm)

libguestfs

guestfish,virt-\*

Libvirt

Virt-install,virsh,boxes

## Quick emulator(QEMU)

Quick emulator will take care of input and output devices like mouse, keyboard, VGA cards, sound cards, Network Interface cards and all the pc like environment is handled by qemu. Each virtual machine runs as a qemu instance and each qemu instance is regular linux process where we perform kill and top to monitor virtual machine.

## Libguestfs

Libguestfs is a library API and set of tools that will let you modify, edit, rescue disk images. If the virtual machine is broken beyond the repair we can copy the content of the file and then recover the data which is stored on the virtual machine then, we can boot that image and copy all the content of the disk image.

It provides an interactive shell called guestfish to access the guest file system to inspect, edit, resize, rescue the virtual machine disk images. guestfish is scriptable using many languages line python, Perl, ruby. Libguestfs provides various tools to perform operations like virt-inspector,virt-cat,virt-edit,virt-rescue.

## Libvirt

Libvirt is a library,deamon tool to manage virtual machine, infrastructure to perform general virtual machine life cycle management.

* It stores the vm information in a xml format to define the virtual machine configuration.
* It provides virsh shell interface to manage the virtual machine operations.
* It will manage device, network, snapshots, storage pools.
* It provides SELinux based security to the user and groups.
* Libvirt interacts with qemu and takes the best practices from qemu and provides the best command line interface.

# Preparing the environment

## System requirement

You need to have a disk space of 100GB to run 4-5 virtual machine, you need to have 8-16gb of memory. In my environment, I have dell poweredge r610 with 2 intel Xeon E5540 processor,16GB of memory and about 500GB of disk space.

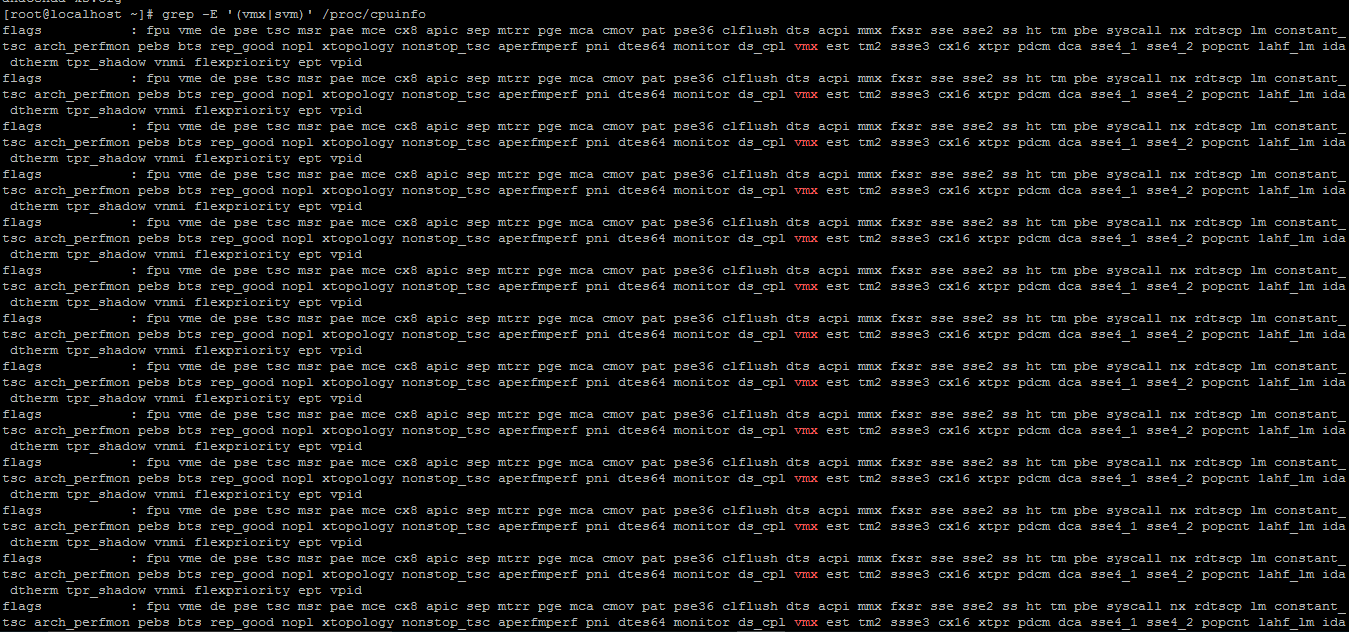
## Virtualization

To use KVM your processor should support virtualization technology to know whether the system supports Hardware virtualization run the below command.

Grep -E ‘(vmx|svm)’ /proc/cpuinfo

If you see vmx or svm in the output, then the cpu will support virtualization. Otherwise virtualization is not supported.

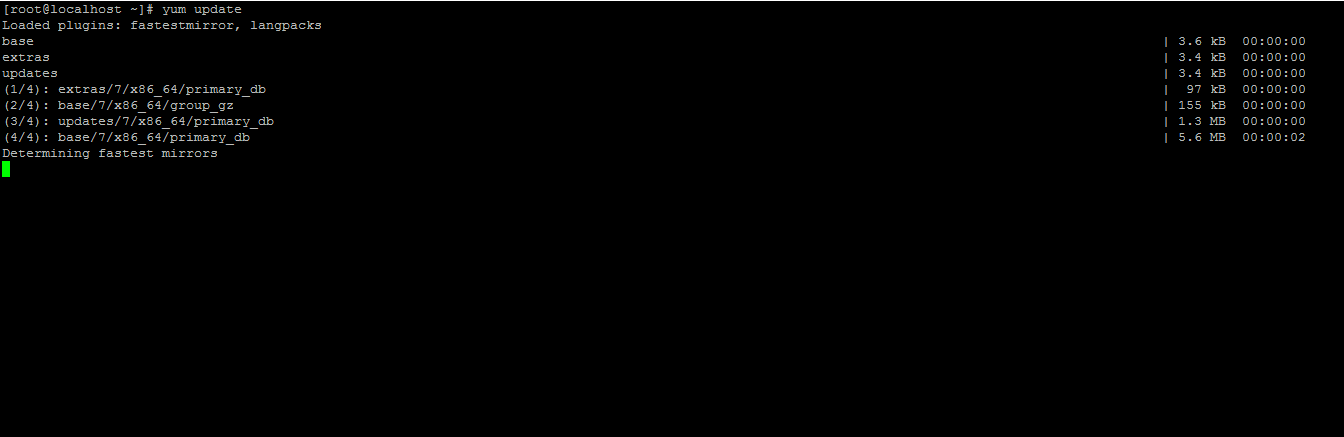
Vmx is for intel processors and svm for the AMD processors.



## KVM Installation

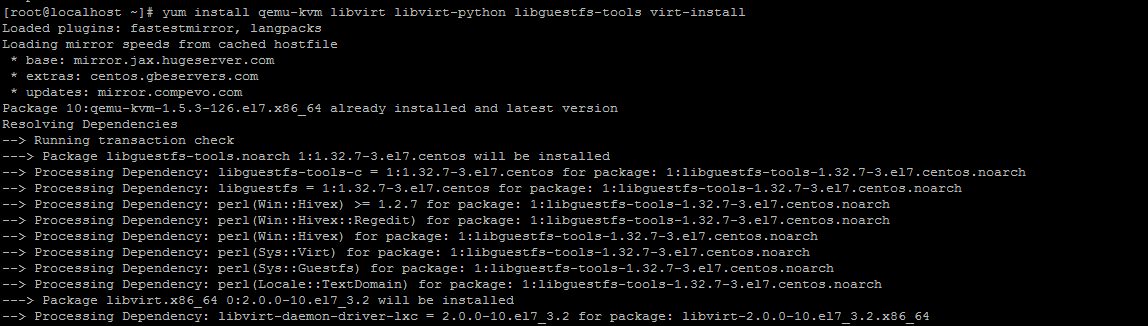
To install kvm we need install several packages. Before installing you need to update yum, to update run below command.

Yum update



After you update, to install kvm you need install several packages to install all the packages run the below command.

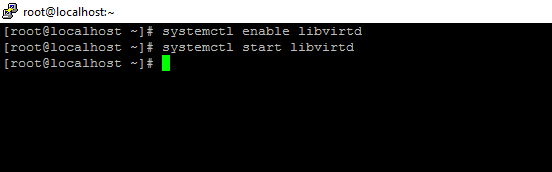
Yum install qemu-kvm libvirt libvirt-python libguestfs-tools virt-install



Now you need to enable and start the libvirtd by using below command.

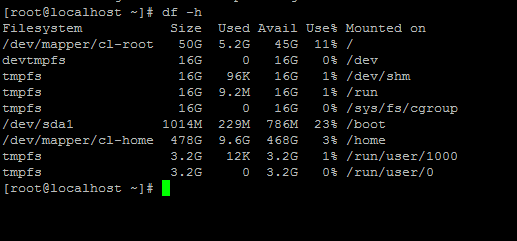
systemctl enable libvirtd

systemctl start libvirtd



## Disk Space

You need to have more disk space to run 3-4 virtual machines at once. So, select a drive which has more disk space or create s symlink to the drive which has more space, By default all your virtual machine image file will be stored in /var/lib/libvirt/images/ . In my case I have 478G to my home folder so I will create symlink to my home folder.



Create a directory in /home folder

mkdir -p /home/qemu/images

Change the owner permissions to qemu.qemu to the folder we created above.

chown -R qemu.qemu /home/qemu/images

To change the context of selinux run the below. This doesn’t allow selinux to take the context of /var/lib/libvirt/images folder.

chcon --reference /var/lib/libvirt/images /home/qemu

chcon --reference /var/lib/libvirt/images /home/qemu/images

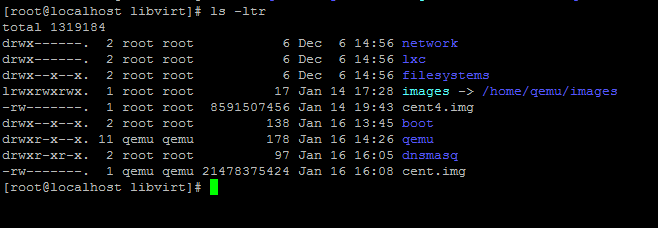
Now delete the /var/lib/libvirt/images using below command.

rmdir /var/lib/libvirt/images

create a symlink to the folder using below command.

ln -s /home/qemu/images /var/lib/libvirt/images

After you run the above command. It will create link to the /home/qemu/images and /var/lib/libvirt/images folder. Then /var/lib/libvirt/images folder will point to the /home/qemu/images as shown in the figure below.



## Networking

After you install the KVM packages, bridge network will be created automatically in virbr0 file. Virtual machine will communicate with the Ethernet port via bridge network(virbr0).

# Creating Virtual Machine

To install the virtual machine from the location file run below command

virt-install --name centos7 --ram 2048 --disk path=/var/lib/libvirt/cent.img,size=20 --vcpus 1 --os-type linux --os-variant generic --network bridge=virbr0 --graphics none --console pty,target\_type=serial --location 'http://mirror.i3d.net/pub/centos/7/os/x86\_64/' --extra-args 'console=ttyS0,115200n8 serial'

--name: name of the virtual machine

--ram: memory for the virtual machine

--diskpath: the image file for the virtual machine and size of the virtual machine.

--vcpus: number of virtual cpus

--os-type: type of operating system ex:- linux, windows

--os-variant: flavor of the linux or windows operating system. If you don’t know it just type specific

--Network bridge: network bridge which in used for inbound and outbound connectivity for the virtual machine.

--graphics: if you don’t want GUI installation just type none, if you want GUI installation type spice it will give you the GUI installation.

--console: console is used to load the console window once the install begins

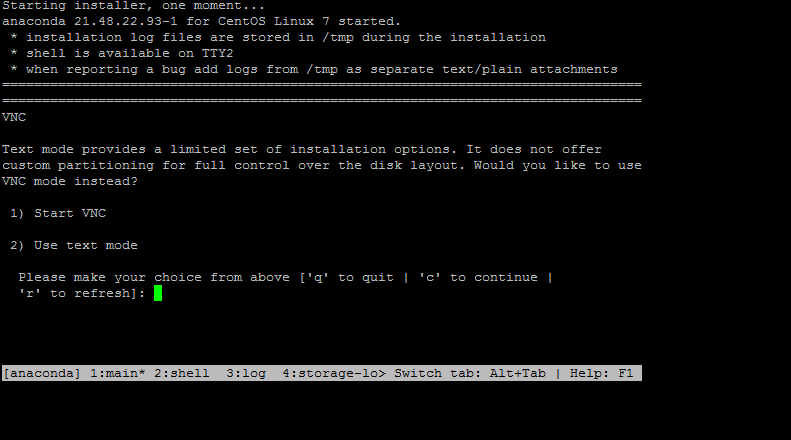
--location: 'http://mirror.i3d.net/pub/centos/7/os/x86\_64/’ this location consists of a centos iso file in the web. If you want to install from the location drive you can always do it by using

--cdrom /home/images/centos.iso

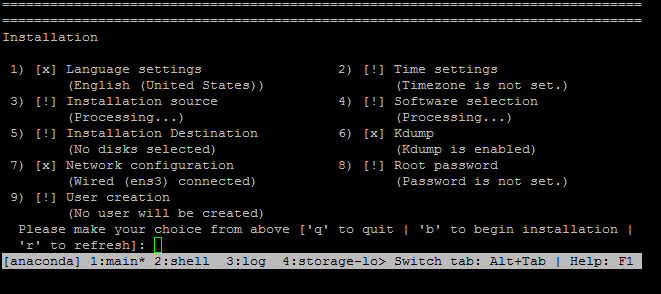
--extra args: the extra args is used to pass the kernel boot parameters to the OS installer.



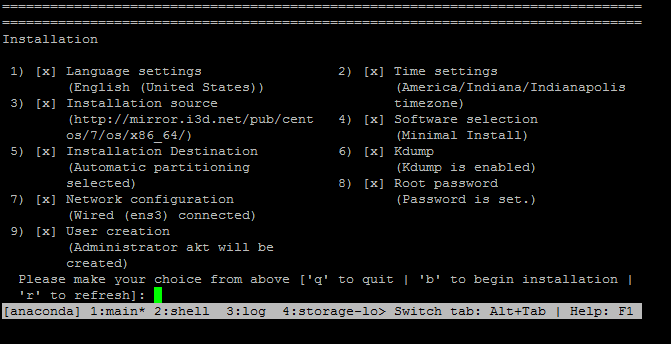
After the file is downloaded from the location the boot process will start either you start VNC Viewer or you can use text mode. I am using text mode for installation so type 2 and press enter.



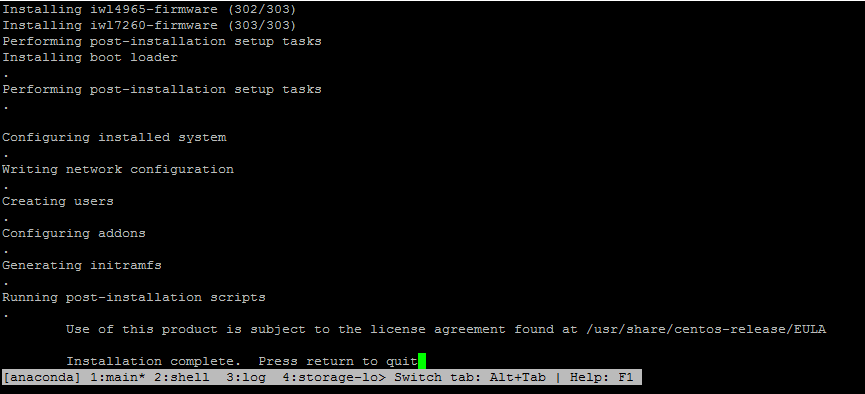
The installation process will begin in text only mode. Here you must configure which is displayed in the “!” symbol. So, follow the on-screen instruction and configure as per your requirement.



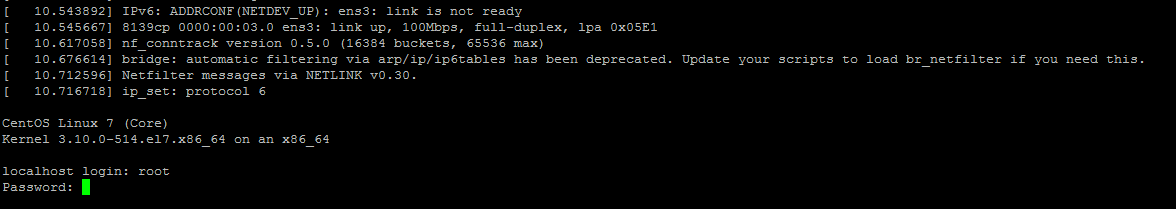
Once you configure everything, then all the option will show “[x]”, then type b and press enter to start installation process.



Once the installation is completed it will prompt you to press enter for restart.



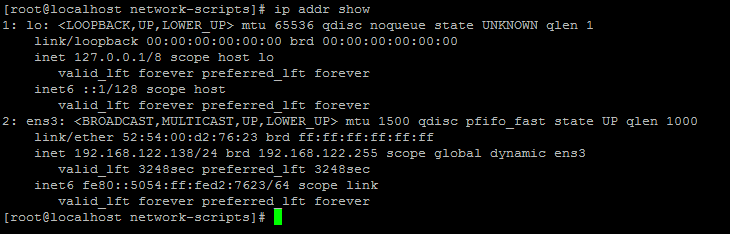
Once you restart you will be asked to login. Use the login details which you created while your installation.



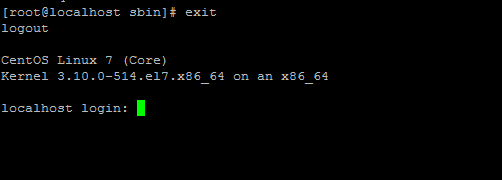
Once your login as the root user make sure you note down the ip address of the virtual machine. Later if we want to login to the virtual machine we can ssh into the ip address of the virtual machine.

Ip addr show

In my case ip address is 192.168.122.138



To come out of the virtual machine type quit and press enter.



# Clone Virtual Machine

If you want a virtual machine with same OS and configuration. We can clone the existing virtual machine rather than installing the virtual machine again.

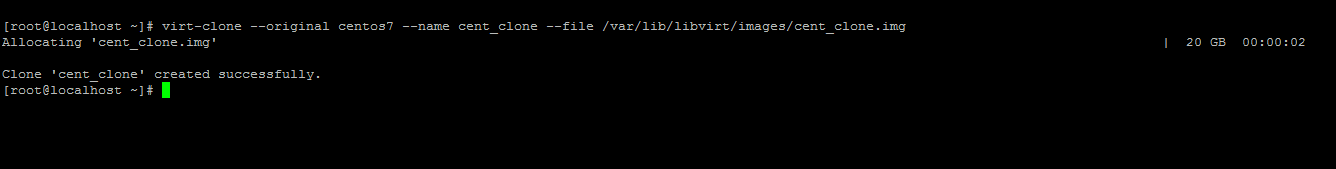
To clone the virtual machine, you must run the below command. Make sure the virtual machine is shut down before you perform a clone.

To shut down the existing virtual machine.

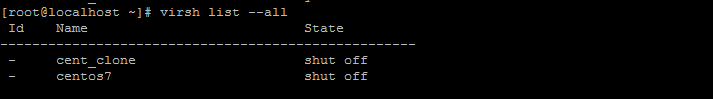
Virsh shutdown centos7

Run the below command to clone the virtual machine

Virt-clone --original centos7 --name cent\_clone --file /var/lib/libvirt/images/cent\_clone.img



Now if you run the “virsh list --all" you will see the 2virtual machines. Centos7 which in created by you and cent\_clone is which we cloned just now.



When you are done, you can start both virtual machine using below command.

Virsh start centos7

Virsh start cent\_clone

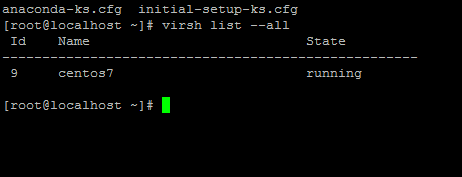
# Manage Virtual Machine

There are several commands to manage and administer the virtual machine below are some basic commands to manage virtual machines.

## Common tasks

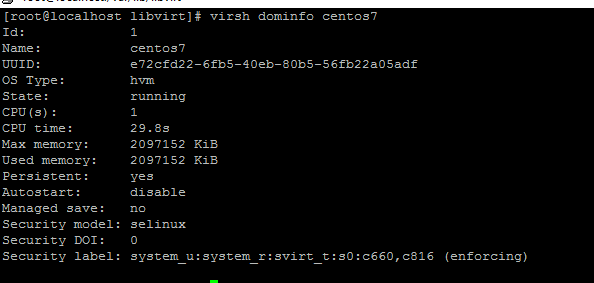
To list the virtual machine, you created above, run the below command.

virsh list –all.



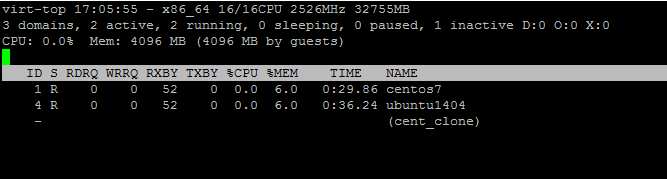
To see the vm information

virsh dominfo centos7



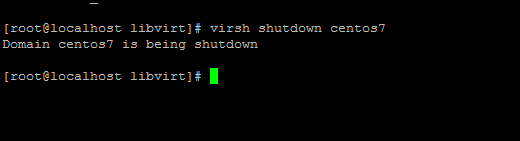
To see the VCPUS/Memory usage of the virtual machines.

virt-top



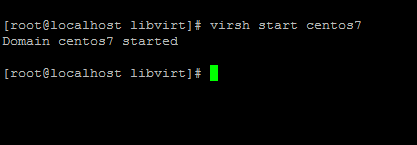
To shut down the virtual Machine.

virsh shutdown centos7



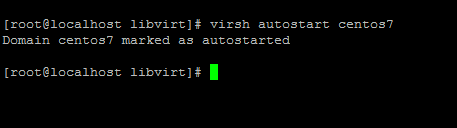
To Start the virtual machine

virsh start centos7



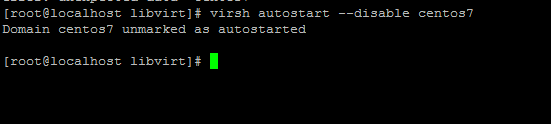
To mark the vm to auto start when the host start

virsh autostart centos7



To mark as manual start and disable autostart

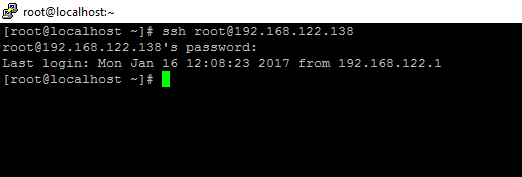
virsh autostart --disable centos7



## Accessing virtual Machine.

To access the virtual machine, you can ssh to the vm ip address by running the below command or you can also use virsh console centos7.

ssh [root@192.168.122.138](mailto:root@192.168.122.138)

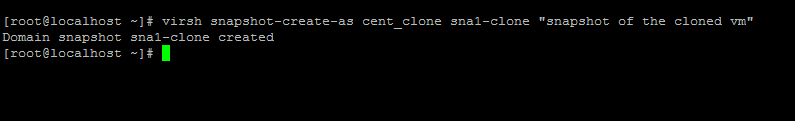


## Snapshots

Snapshots save the disk, memory and device state of the virtual machine at that point of time and use it in future. If something goes wrong with the virtual machine and if you want load the previous state of the virtual machine, you can always do it by using saved snapshots.

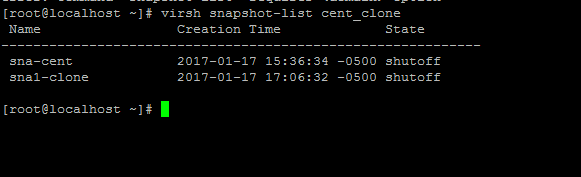
To create a snapshot of the virtual machine.

virsh snapshot-create-as cent\_clone sna1-clone “snapshot of the cloned centos”



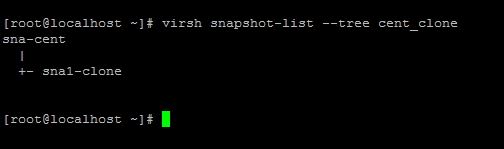
To list the snapshot of the virtual machine.

virsh snapshot-list cent\_clone



To see the relationship between the snapshot in a tree structure. It will list the hierarchy of the snapshots created to that virtual machine.

virsh snapshot-list --tree cent\_clone



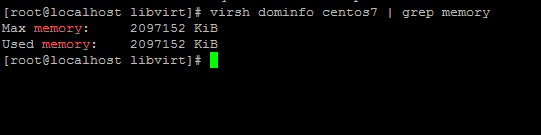
# Changing Virtual Machine Configuration

We can easily change virtual machine configuration after creating them like memory,vcpus and disk space.

## Changing memory

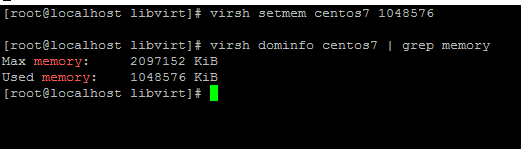
To view the current memory usage of the virtual machine.

virsh dominfo centos7 | grep memory



To dynamically set the memory for the virtual machine. Memory will be calculated in kb.

virsh setmem centos7 1048576



To increase the memory above the memory which we specify while creating the virtual machine. The best way does it is to shut down the virtual machine

To shut down virtual machine

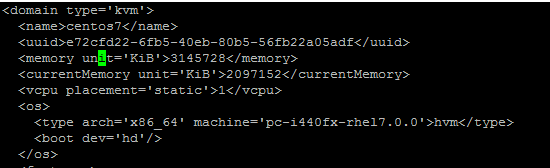
Virsh shutdown centos7

To edit virtual machine configuration.

virsh edit centos7

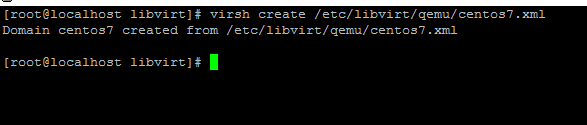
Once you’re into the file press “ i “ to go insert and edit the memory setting as shown in the figure below and then save and exit by pressing escape and the :wq then enter.

<memory unit=’KiB’> 3145728</memory>



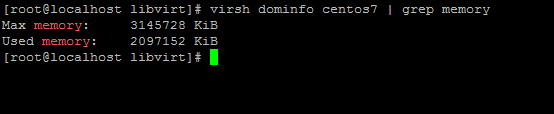
Restart the virtual machine from the configuration file.

virsh create /etc/libvirt/qemu/centos7.xml



Now view memory settings.

virsh dominfo centos7 | grep memory



## Changing vcpus

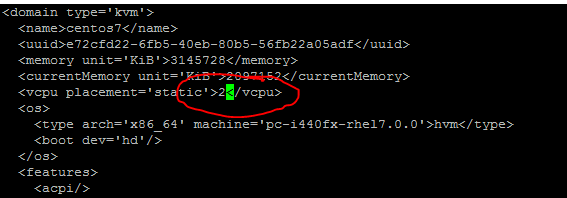
To change the vcpus we need to edit the virtual machine settings. To do that we need to shut down the virtual machine and then edit the virtual machine.

virsh shutdown centos7

virsh edit centos7

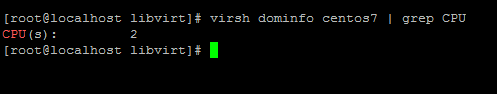
Replace 1 with 2 in the below command and then save and exit :wq

<vcpu placement = ‘static’>2</vcpus>



To check the virtual cpus

virsh dominfo centos7 | grep CPU

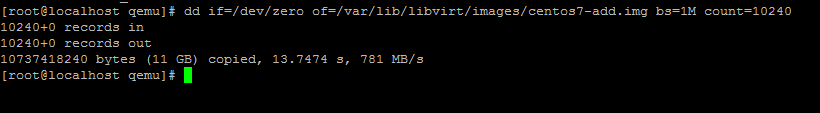


## Changing the disk capacity.

If you want add more disk space to the virtual machine, you can always do that by editing the virtual machine xml files.

Create a 10gb disk image in the host machine

dd if=/dev/zero of=/var/lib/libvirt/images/centos7-add.img bs=1M count=10240



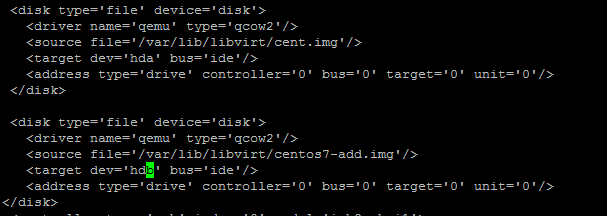
Now shutdown the virtual machine.

virsh shutdown centos7

virsh edit centos7

Copy the <disk> </disk> part of code and paste below and edit the source file address to which we created above and target device as hdb and then save an exit. Then restart the virtual machine from the update xml configuration file.

virsh create /etc/libvirt/qemu/centos7.xml

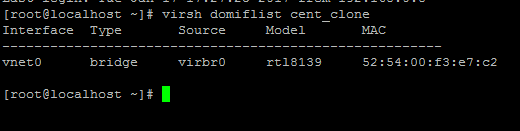


## Network Interface Card(NIC)

### Adding a network interface card

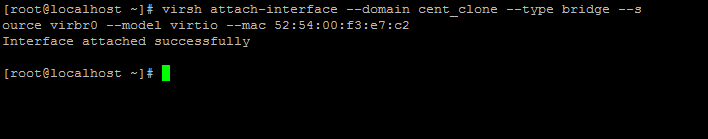
You can add the network interface card(nic) to the virtual machine even if the virtual machine is running. Now let’s add nic to the virtual machine. In my virtual machine, I have only one nic that you can see in the below picture.

virsh domiflist cent\_clone



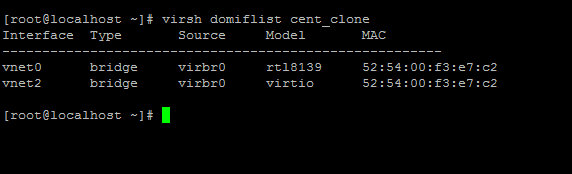
To add the nic to the virtual machine. Note down the mac address from the above picture and use it to add nic.

virsh --type bridge --domain cent\_clone --source virbr0 --model virtio --mac 52:54:00:f3:e7:c2



We have successfully configured another nic on the virtual machine. To list the network configuration of the virtual machine run below command. You will see 2nics

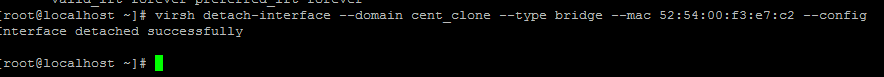
virsh dominlist cent\_clone



### Detaching the network interface card

You can also remove the nic by using below command.

virsh detach-interface --domain cent\_clone --type bridge --mac 52:54:00:f3:e7:c2 --config



# Deleting the virtual machine

If you no longer need the virtual machine, then we can delete completely.

Shutdown the virtual machine.

Virsh shutdown centos7

Virsh undefine centos7

Finally remove the centos7 image file from the local directory by deleting the image file,

rm /var/lib/libvirt/images/cent.img