

# OPS245 Lab 2

## Contents

- 1 LAB PREPARATION
  - 1.1 Purpose / Objectives of Lab 2
- 2 INVESTIGATION 1: SETUP FOR NESTED VIRTUAL MACHINES
  - 2.1 Part 1: Install KVM Virtualization Application
  - 2.2 Part 2: Configure VMware Workstation for Nested VMs
- 3 INVESTIGATION 2: INSTALL NESTED VIRTUAL MACHINES (KVM)
  - 3.1 Part 1: Installing VM from a Network (Graphical)
  - 3.2 Part 2: Installing from a Network (Minimal install - CLI)
  - 3.3 Part 3: Installing from a Network using a Kickstart File
- 4 INVESTIGATION 3: MANAGING VIRTUAL MACHINES (KVM)
  - 4.1 Part 1: Backing Up Virtual Machines
  - 4.2 Part 2: Restoring Virtual Machines
  - 4.3 Part 3: Using the Command Line for VM State Management
- 5 INVESTIGATION 4: USING PYTHON TO AUTOMATE MANAGING VIRTUAL MACHINES
- 6 LAB 2 SIGN-OFF (SHOW INSTRUCTOR)
- 7 Practice For Quizzes, Tests, Midterm & Final Exam

## LAB PREPARATION

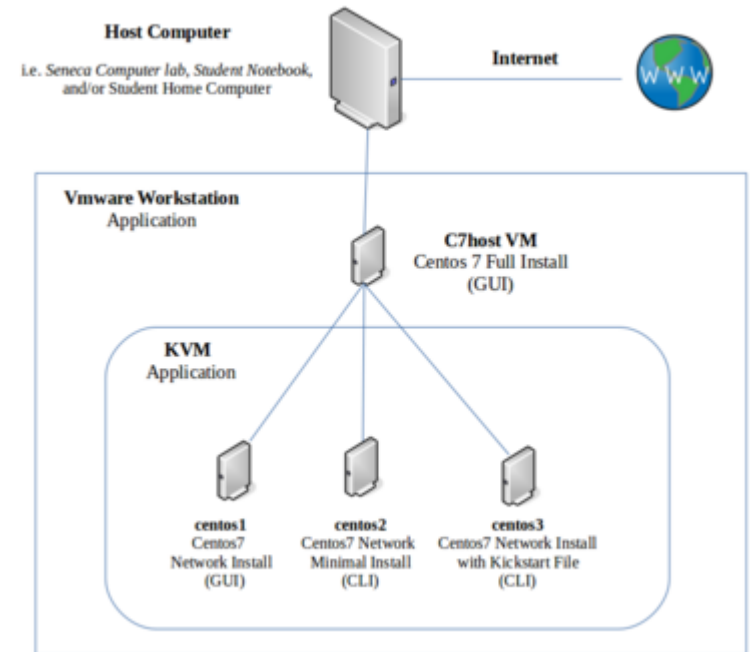
### Purpose / Objectives of Lab 2

In this lab, you will create 3 remaining virtual machines using another virtualization program called **KVM** that will run in your c7host VM. These VMs will be used throughout the remainder of this course to learn how to administer them (installing software, managing services, networking, etc).

While you are performing this lab, it is recommended to generally note the major differences in the different installation methods, and which method you prefer to use if you were a Linux system administrator in charge of installing many Linux distributions for an organization.

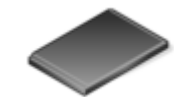
### Main Objectives

- Installing additional Virtualization Software on your **c7host** machine (**KVM**)
- Create 3 separate VMs (virtual machines) using different installation methods:
  - **centos1**: Network CentOS Installation (**Graphical**)
  - **centos2**: Network CentOS Installation (minimal install - **CLI only**)
  - **centos3**: Network CentOS Installation with Kickstart configuration file (**CLI only**)
- Manipulate virtual machines by CLI (**virsh**)
- Properly **backup VM images** and backup **VM configuration files**
- Create and run **Bash Shell scripts** to automatically create a post-install report for an installed VM.



At the end of lab2, your VMware Workstation application will contain 4 virtual machines (c7host in your VMware Workstation application, and centos1, centos2, centos3 VMs in your KVM application). You will now have the option to run one virtual machine at a time, or run all machines simultaneously to learn about networking (covered in later labs)

## Minimum Required Materials



Solid State Drive



USB key  
(for backups)



Lab2 Log Book

## Linux Command Reference

Virtualization  
 virt-manager (<http://linux.die.net/man/1/virt-manager>)  
 virsh (<http://linux.die.net/man/1/virsh>)  
 Installation Guides  
 Installing & Using KVM on CentOS (<http://wiki.centos.org/HowTos/KV>)

Miscellaneous  
 gzip , gunzip ([http://linuxcommand.org/lc3\\_man\\_pages/gzip1.html](http://linuxcommand.org/lc3_man_pages/gzip1.html))  
 ip (<http://man7.org/linux/man-pages/man8/ip.8.html>)  
 grep (<http://man7.org/linux/man-pages/man>)

## Matrix Online Tutorials:

- Shell Scripting - Part 2 (Logic & Math Expressions):  
</home/ops235/scripting-2>
- Shell Scripting - Part 3 (Loops)  
</home/ops235/scripting-3>

M)	l/grep.1.html)
Using KVM (tutorial) (http://www.dedoimedo.com/computers/kvm-intro.html)	wc (http://man7.org/linux/man-pages/man1/wc.1.html)
virsh command reference (https://libvirt.org/sources/virshcmdref/html-single/)	pwd (http://man7.org/linux/man-pages/man1/pwd.1.html)
	ls (http://man7.org/linux/man-pages/man1/ls.1.html)
	more (http://man7.org/linux/man-pages/man1/more.1.html)
	file (http://man7.org/linux/man-pages/man1/file.1.html)
	wget (http://man7.org/linux/man-pages/man1/wget.1.html)
	chmod (http://man7.org/linux/man-pages/man1/chmod.1.html)
	vi (http://ss64.com/vi.html)

# INVESTIGATION 1: SETUP FOR NESTED VIRTUAL MACHINES

## Part 1: Install KVM Virtualization Application

We will now install the KVM package in order to create our remaining "nested" VMs. We will also be starting several services (including iptables) and disabling the firewalld service. We will learn more about managing firewalls using iptables in lab6.

### Perform the following steps:

1. Log into your **c7host** machine.
  2. perform a software update on your **c7host** VM by issuing the following command: **sudo yum update**
-



### Yum Update Hangs

If you experience yum update "hanging" around item 689 of over 1200 packages, issue the following commands **(in a new terminal!)**:

```
sudo pkill systemctl  
sudo yum clean all  
sudo yum update
```

**NOTE:** Do NOT press <ctrl>c since it may cause your machine to cause a kernel panic when you restart your machine.

3. Using elevated privileges, install the virtualization software by issuing the command:

```
sudo yum install qemu-kvm qemu-img virt-manager libvirt libvirt-python \  
libvirt-client virt-install virt-viewer bridge-utils
```

4. **Restart your c7host virtual machine.** If you fail to do this, you may experience virtualization network problems.

5. Start the virtualization service: `sudo systemctl start libvirtd`

**NOTE:** The most recent variants of CentOS and Fedora are using a service called **firewalld** that replaces **iptables**, however the *iptables* service is still in relatively common usage and knowing how to use it also works with firewalld. In this course we will concentrate on *iptables*.

6. To disable and remove firewalld, issue the following commands:

```
sudo systemctl disable firewalld  
sudo systemctl stop firewalld  
sudo yum remove firewalld
```

7. To install and enable the IPTables services, issue the following commands:

```
sudo yum install iptables-services  
sudo systemctl enable iptables  
sudo systemctl start iptables
```



### About KVM

There are actually several key programs installed for virtualization using KVM:

- **kvm/qemu** - the hypervisor and other hardware emulation systems.
- A system service named **libvirtd** that manages the VMs.
- A graphical tool for managing virtual machines (**virt-manager**) and the **virsh** command-line tool.



### Run virt-manager as a regular user, not as root

Otherwise all your virtual machines will be owned by root and you won't be able to use them as a regular user.

8. Start the graphical tool by selecting the menu options **Applications>System Tools>Virtual Machine Manager** or by typing the command **virt-manager** (without sudo!)

9. You will be learning in the next investigation to perform several different types of CentOS Linux installs.

## Part 2: Configure VMware Workstation for Nested VMs

Although we are using VMware Workstation to run our c7host VM, we will now install and configure another virtualization package called KVM in order to install the remaining VMs for this course. Since we are "nesting" VMs (i.e. running a VM inside another VM) we need to configure our c7host's Linux boot-up parameters in order to allow these VMs to run efficiently.

**Perform the following steps:**

1. In a terminal use elavated privileges to edit the file called: `/boot/efi/EFI/centos/grub.cfg`
  - **If this file doesn't exist, double-check your UEFI settings in VMWare Workstation for c7host. If BIOS is selected, you MUST redo Lab 1.**
2. Search for the **first occurrence** of the Linux Kernel boot command. Do not make the following changes on more than one entry!
3. Insert the boot option: `kvm-intel.nested=1` (for AMD processors `kvm-amd.nested=1`) at the end of the Linux kernel boot options.



**About the reference settings shown below**

- Only **ONE** of these settings might be applicable.
- Enter **JUST** the text highlighted in **BLUE** as your kernel version, root, and LVM settings might vary slightly.

```
linuxefi /vmlinuz-3.10.0-1062.1.2.el7.x86_64 root=/dev/mapper/centos_c7host-  
root ro crashkernel=auto rd.lvm.lv=centos_c7host/root  
rd.lvm.lv=centos_c7host/swap rhgb quiet LANG=en_CA.UTF-8 kvm-intel.nested=1  
initrdefi /initramfs-3.10.0-1062.1.2.el7.x86_64.img
```

**OR**

```
linuxefi /boot/vmlinuz-3.10.0-1062.el7.x86_64 root=/dev/mapper/centos_c7host-  
root ro crashkernel=auto rd.lvm.lv=centos_c7host/root  
rd.lvm.lv=centos_c7host/swap rhgb quiet LANG=en_CA.UTF-8 kvm-intel.nested=1  
initrdefi /boot/initramfs-3.10.0-1062.el7.x86_64.img
```

4. Save your editing changes, close the virtual machine application, and **reboot** your c7host VM.
5. If you configured your c7host VM for nested VMs, then you should get the output **Y** when you issue the following command:  
**cat /sys/module/kvm\_intel/parameters/nested**
  - For **AMD** processors, check the **/sys/module/kvm\_amd/parameters/nested** file. You should get the output **1**
  - And if **kvm\_intel** directory doesn't exist, double-check your **Processors => Virtualization Engine (Intel VT-x/EPT...)** settings in VMWare Workstation.

Answer the INVESTIGATION 1 observations / questions in your lab log book.

## INVESTIGATION 2: INSTALL NESTED VIRTUAL MACHINES (KVM)



### Keep the root password the same for Host and VMs

In order to simplify running the lab checking scripts in future labs, using the same root password for ALL machines (c7host and virtual machines). Also use the same username and passwords for all of your machines (c7host and virtual machines).

### Part 1: Installing VM from a Network (Graphical)

#### VM Details:

**VM Name (and hostname):** centos1

**Boot media:** Network installation

**CentOS Full Network Install URL:**

- Seneca Lab: [http://mirror.senecacollege.ca/centos/7/os/x86\\_64/](http://mirror.senecacollege.ca/centos/7/os/x86_64/) (**NOTE: requires VPN**)
- Home: [http://mirror.netflash.net/centos/7/os/x86\\_64/](http://mirror.netflash.net/centos/7/os/x86_64/)

**VM Image Pathname:** /var/lib/libvirt/images/centos1.qcow2

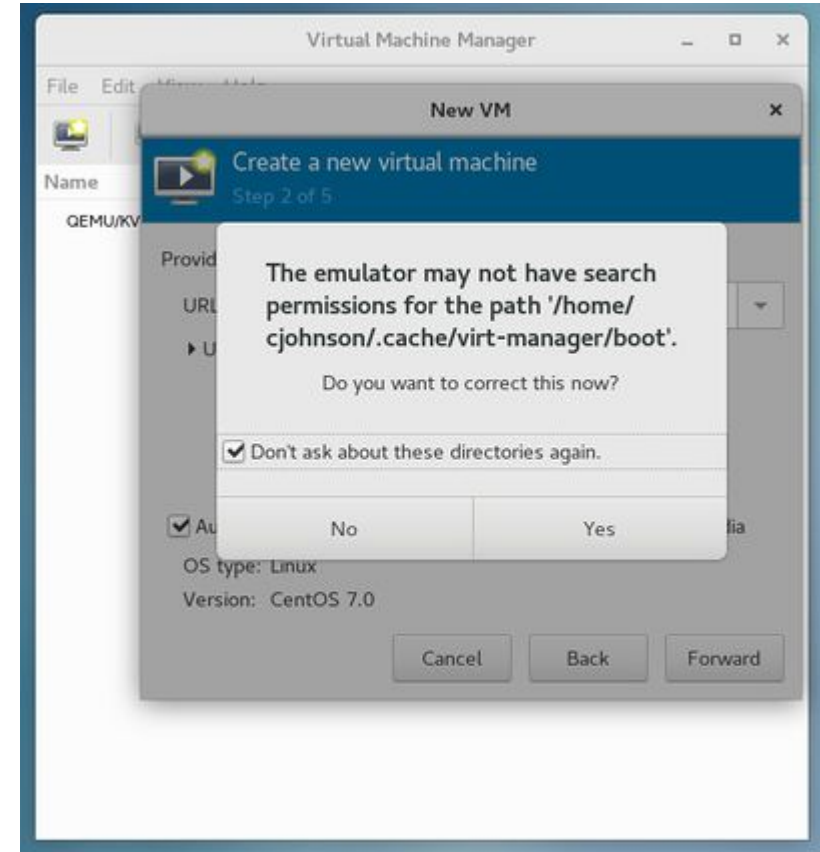
**Memory:** 2048MB

**Disk space: 15GB**

**CPUs: 2**

**Perform the following steps:**

1. Launch the *KVM virtual machine manager* by clicking **Applications -> System Tools -> Virtual Machine Manager**.
2. When prompted, enter your password.
3. Click the **create a new VM icon** located near the top left-corner of the application window.
4. Select the **Network Install** option and click **Forward**.
5. Enter (copy and paste) the URL located at the top of this section (depending whether you are at Seneca College or not) and click **Forward**.
6. Set RAM size to 2048 MB and CPUs to 2, then click **Forward**.
7. Set Hard Disk size to 15GB and click **Forward**.
8. Enter the name: **centos1**, AND then select the option: **Customize configuration before install**, and click **Finish**.
9. Another dialog will appear. Click **CPUs** (or "processors") and on right-side under Configuration select **Copy Host CPU Configuration**, click **Apply**, and then click **Begin Installation** at the top left-hand side.
10. During the install, select **Gnome Desktop** software selection). For partitioning, select **I will configure partition settings**, click done, then select **Click here to create them automatically**. Set the / partition for **ext4** file-system type, and click **Done**.
11. Set the correct **Date and Time Zone**, and then click on **Network and Hostname**. The network should be turned **ON**. For hostname, enter: **centos1** and then click **Done**.
12. Make sure that when you create your regular user account you check the box to make them an administrator.
13. Complete the installation. Login to your regular user account, and perform a **sudo yum update** for the centos1 VM (reboot if required). Make certain to adjust your screen-saver settings if desired.



You may see this warning when creating your first VM inside CentOS. Select ***Don't ask me about these directories again*** and click **Yes**.



**Use same root password / regular username / regular user passwords for c7host and ALL VMs**

To simplify the lab checking process make certain that you use the identical root password, regular username, and regular username password for VMs that you create in this labs as you did for c7host machine in lab1.

14. Repeat the steps as you did in the previous investigation (Investigation1 Part 1 ([https://wiki.cdot.senecacollege.ca/wiki/OPS245\\_Lab\\_2#Part\\_1:\\_I\\_nstall\\_KVM\\_Virtualization\\_Application](https://wiki.cdot.senecacollege.ca/wiki/OPS245_Lab_2#Part_1:_I_nstall_KVM_Virtualization_Application))) to **stop, disable and remove firewalld, install iptables-services, start and enable iptables** for this newly-created VM.
15. Repeat the steps as you did with c7host post-install to **turn off (permissive) SELinux** and perform a **yum update**.
16. Issue the following command to obtain the IPADDR for your centos1 VM to record in your lab2 logbook: **ip address show**
17. Record the time taken to install, and compare this to the time taken by the previous installations in your lab2 logbook.

## Part 2: Installing from a Network (Minimal install - CLI)

### VM Details:

**VM Name (and hostname):** centos2

**Boot media:** Network installation

**CentOS Full Network Install URL:**

- Seneca Lab: [http://mirror.senecacollege.ca/centos/7/os/x86\\_64/](http://mirror.senecacollege.ca/centos/7/os/x86_64/) (**NOTE: requires VPN**)
- Home: [http://mirror.netflash.net/centos/7/os/x86\\_64/](http://mirror.netflash.net/centos/7/os/x86_64/)

**VM Image Pathname:** /var/lib/libvirt/images/centos2.qcow2

**Memory:** 2048MB

**Disk space:** 20GB

**CPUs:** 1

### Perform the following steps:

1. Create the VM (called **centos2**) as you did with the *centos1* machine.
2. Make certain to enter the name: **centos2**, AND then select the option: **Customize configuration before install**, and select **Copy Host CPU Configuration**, click **Apply**, and then click **Begin Installation**.
3. When selecting the install options for centos2, do the same operation that you did in centos1 (but with **Minimal Install** software selection instead), but after **automatically creating the partitions**, reduce the size of the root logical volume to **8 GiB** and add a logical volume with a size of **2 GiB** (mount point: **/home**, name: **home**, and make certain root and /home logical volumes have **ext4** file system).
4. Complete the installation. Login to your regular user account.
5. Repeat the steps as you did in the previous investigation (Investigation1 Part 1 ([https://wiki.cdot.senecacollege.ca/wiki/OPS245\\_Lab\\_2#Part\\_1:\\_I\\_nstall\\_KVM\\_Virtualization\\_Application](https://wiki.cdot.senecacollege.ca/wiki/OPS245_Lab_2#Part_1:_I_nstall_KVM_Virtualization_Application))) to **stop and disable firewalld, install iptables-services, start and enable iptables** for this newly-created VM.
6. Repeat the steps as you did with c7host post-install to **turn off (permissive) SELinux** (using the command 'vi' instead of 'vim') and perform a **yum update**.
7. Issue the following command to obtain and record your centos2 IPADDR in your lab2 logbook: **ip address show**



8. Record the time taken to install, and compare this to the time taken by the previous installations in your lab2 logbook.

### Part 3: Installing from a Network using a Kickstart File

#### VM Details:

**VM Name (and hostname):** centos3

**Boot media:** Network installation

**CentOS 7 Full Install Network URL:**

- Seneca Lab:

[https://mirror.senecacollege.ca/centos/7/os/x86\\_64/](https://mirror.senecacollege.ca/centos/7/os/x86_64/)

- Home: [http://mirror.netflash.net/centos/7/os/x86\\_64/](http://mirror.netflash.net/centos/7/os/x86_64/)

#### Kickstart File URL (Kernel options):

- Home: **ks=**<https://raw.githubusercontent.com/OPS245/labs/main/centos7-kickstart.cfg>

**VM Image Pathname:** /var/lib/libvirt/images/centos3.qcow2

**Memory:** 2048MB (**IMPORTANT** Do not use less than 2048MB during installation.)

**Disk space:** 15GB

**CPUs:** 2



#### Using a Kickstart file with a local install

Under most circumstances, a network install would be drawing the installation data from a server in the local network, allowing it to transmit data at higher speeds than we can achieve over the internet. If your download speed is slow enough that network installs are failing, it is acceptable to install these VMs from local media. However, you will still need to provide a kickstart file to Centos3.

To do so, start the install process as normal for an installation from an iso file, but when presented with the menu giving you the option to 'Install CentOS 7' or 'Test this media & install CentOS 7', highlight 'Install CentOS7' and press <tab>. This will open a prompt at the bottom of the window for you to enter grub configuration options. Type **inst.ks=**<https://raw.githubusercontent.com/OPS245/labs/main/centos7-kickstart.cfg> and hit <enter>.



#### Include **ks=** in the URL options field!

When using a kickstart file, make sure you include the **ks=** portion of the link. If done correctly, you should not be able to select partitions or any other settings.

### Perform the following steps:

1. Create the VM (called **centos3**)
2. During the install, copy the network URL, then click the **URL options** to expand the **kernel options** input textbox. Type the following in the kernel options textbox:
  - **ks**=<https://raw.githubusercontent.com/OPS245/labs/main/centos7-kickstart.cfg>
3. Then click the **forward** button to proceed. Make certain to select the correct Memory Size and Disk Space size shown in the VM Details above
4. Make certain to enter the name: **centos3**, AND then select the option: **Customize configuration before install**, and select **Copy Host CPU Configuration**, click **Apply**, and then click **Begin Installation**.
5. Observe the installation. How is it different from booting from a downloaded image?
6. Record the time taken to install, and compare this to the time taken by the previous installations.

If during the installation, you see the message at the bottom **Pane is Dead**, click the **Virtual Machine** menu at the top, select **Shut Down -> Force Off**, right-click on **centos3** in the *virtual manager* window and select **Delete**. Redo the VM setup for a new instance of the *centos3* VM.

7. What happens when the installation is finished?
8. **In a web browser**, click the kickstart (KS) link above. This link is a text file. Read through it to find the following information (pay attention to lines starting with #) and record it in your Lab Logbook:
  - **Regular-user account name**
  - **Regular-user account password**
  - **Root Password**
9. Boot the virtual machine and log in (use the user ID and password information from the previous step to gain access to this VM).
10. Compare the experience to the first time you booted the other virtual machines.
11. Repeat the steps as you did in the previous investigation (Investigation1 Part 1 ([https://wiki.cdott.senecacollege.ca/wiki/OPS245\\_Lab\\_2#Part\\_1:\\_Install\\_KVM\\_Virtualization\\_Application](https://wiki.cdott.senecacollege.ca/wiki/OPS245_Lab_2#Part_1:_Install_KVM_Virtualization_Application))) to **stop and disable firewalld, install iptables-services, start and enable iptables** for this newly-created VM.
12. Repeat the steps as you did with c7host post-install to **turn off SELinux** and perform a **yum update**.

You'll notice something when you go to set **SELinux** to **permissive**. The kickstart file already did that for you. It could even have performed the switch from **firewalld** to **iptables** for you (but it didn't).

1. Issue the following command to obtain and record your centos2 IPADDR in your lab2 logbook: **ip address**
2. Remember that centos3 is text-based interface only (no graphics). To recover from a blank screen, press a key (like the SPACE key) to return to the screen display.
3. Record the time taken to install, and compare this to the time taken by the previous installations in your lab2 logbook.

**Answer the INVESTIGATION 2 observations / questions in your lab log book.**

# INVESTIGATION 3: MANAGING VIRTUAL MACHINES (KVM)



## Root Privileges

As part of this investigation you will learn how to switch over to the root account in order to run several privileged commands in sequence. It can be tempting to just use this technique all the time, and never have to worry about `sudo`, but do **not** do so. It undermines the security of your system. Use it only when you need it.

## Part 1: Backing Up Virtual Machines

### Perform the following steps:

1. Shut down your **centos1**, **centos2**, and **centos3** VMs. For *centos2* and *centos3*, which are CLI-only, you can issue the following command to shutdown: **poweroff**. Please be patient, the VMs will shut down!
2. In your **c7host** VM, open a new Terminal window.
3. Use elevated privileges to list the size and names of files in **/var/lib/libvirt/images/**
  - What do these files contain?
4. Use the command **sudo -i** and enter your password if prompted. You are now root until you use the command **exit** to return to your normal user account.
5. Change to the images directory by issuing the following command: **cd /var/lib/libvirt/images/**. Note that you did not need to use `sudo`, as you are already using elevated permissions.
6. Make a compressed backup of your **centos1.qcow2**, **centos2.qcow2**, and **centos3.qcow2** files to your regular user's home directory by issuing each command - one at a time (create **backups** directory **within your regular user's home directory** before running these commands):  

```
gzip < centos1.qcow2 >
~YourRegularUsername/backups/centos1.qcow2.gz
gzip < centos2.qcow2 >
~YourRegularUsername/backups/centos2.qcow2.gz
gzip < centos3.qcow2 >
~YourRegularUsername/backups/centos3.qcow2.gz
```

**NOTE:** Make certain to use the redirection signs "<" and ">" properly in the command!



## Backing up VMs

There are two general processes in order to back up your images:

1. **Compressing your images** (also recommended to backup up to external storage USB Key) using the **gzip** command.
2. **Backup the VM xml configuration file** (preferably to USB key) using **virsh** shell command to add VM to virtual machine manager list (in the event that the HOST machine is "wiped" and re-installed, but VM images and xml configuration files have been backed up external storage).

Taking the time to backup the image of the Operating System's file system allows the user to return to a **"restoration point"** using the **gunzip** command in case something bad occurs to the OS during a lab.

Failure to take the time to make and confirm backups can result in loss of lab work for the student!



## Please be patient

It may look like the command prompt is stuck but it could take a while for `gzip` to compress an entire operating system. **NOTE:** Do NOT press

<ctrl>c to cancel this process. If you do, your archive will become incomplete and your recovery will be corrupt.

7. Compare the size of the compressed and original files (hint: use **ls -lh**). If file is very large (like 15GB), you didn't compress it and you need to remove that file and perform the previous step until you get it right!
8. Once you are **sure you have all three VMs backed up**, use the **exit** command to revert back to your normal user.
9. Start the *centos3* VM.
10. **Make certain that you are in your VM and not in your main system!**
11. Wreck only your centos3 system! Try this command inside the centos3 virtual machine: **sudo rm -rf /\*** (ignore error messages).
12. Shut down and restart the centos3 VM (you may need to use the **Force Reset** option to do so).
13. When the machine restarts it will not boot since all system files have been removed!
14. Use the **Force Off** option to turn centos3 back off.
15. Restore the original image from the backup from your home directory to your **images** directory by typing **sudo -i** command first [do not forget to **exit** when you are done], then this command:  
**gunzip < ~/YourRegularUsername/backups/centos3.qcow2.gz > /var/lib/libvirt/images/centos3.qcow2**
16. Restart the VM. Is it working normally?
17. You should also make a copy of the XML configuration file for each VM in case you "wipe" and re-install the host machine, and want to add a restored VM backups to the virtual machine manager list. We will demonstrate using the centos3 XML configuration file, and prove that a "clone" can be added to your list. Please perform the following step:
18. Execute the following command: **sudo virsh dumpxml centos3 > centos3.xml**
19. Examine the file **centos3.xml**. What does it contain? What format is it in?

## Part 2: Restoring Virtual Machines

1. We will now learn how to download a compressed image file and XML configuration file and add it as a VM to the Virtual Machine Manager menu.
2. Issue the following commands:
  - **wget https://senecafts.senecacollege.ca/link/mpxZ1JBICQUDrKDYPX26z/download/centos4.qcow2.backup.gz**
  - **wget https://raw.githubusercontent.com/OPS245/labs/main/centos4.xml**
3. Use gunzip with elevated privileges to decompress the qcow2 image file into the **/var/lib/libvirt/images** directory.
4. Issue the command: **sudo virsh define centos4.xml**
5. What happened in the virtual manager window? In order to remove a VM entry in the Virtual Manager window, simply issue the command **virsh undefine VM\_name** without the **.xml** file extension
6. Start up your new centos4 VM.
7. Click on the user *OPS245*, and click the cog icon.
8. Notice *Cinnamon (Software Rendering)* is selected. The Cinnamon desktop environment has been installed on this VM. From this menu, you can select other installed desktop environments. This is how you switch between them. Write it down.
9. Login with the password **ops245**. Feel free to explore the new environment.
10. Prior to your practical test, you will be required to perform a similar operation to download, unzip and run a VM image for your practical test.



### Shutting Down the Host while Virtual Machines are Running

If you shut down your host system while virtual machines are running, they will be suspended, and will resume the next time you boot your host system. Note that it is better to shut down the VMs prior to shutting down the host

12. For the remainder of these labs, it is assumed that you will backup **both** the images and XML configuration files for **all** Virtual machines, when asked to backup your virtual machines. It is also highly recommended to backup these files to an external storage device (eg. USB key) in case the host machine gets "wiped" and you need to rebuild your HOST machine and then restore your Virtual Machines...

13. Answer this question in your log book:

- In order to fully back up a virtual machine, what information should be saved in addition to the virtual machine image?

## Part 3: Using the Command Line for VM State Management

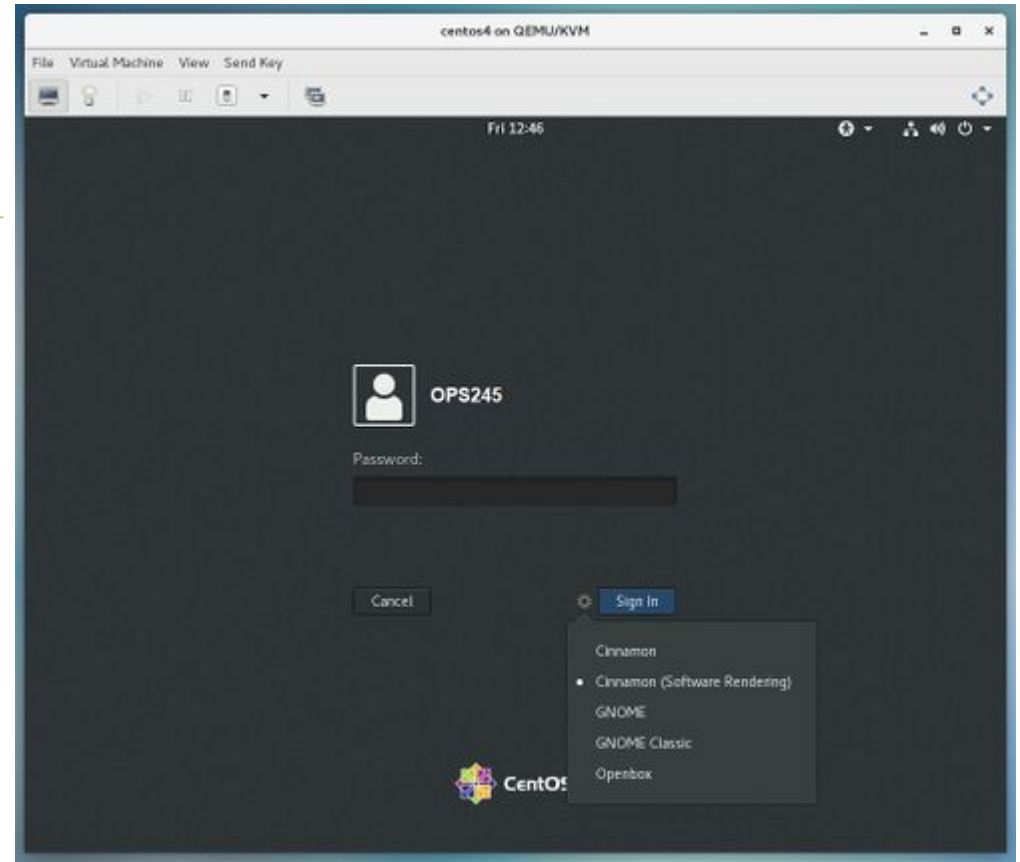
You will



### Bash Shell Tips:

#### Data Input

- A shell can obtain data from a number of methods: **reading input files**, using **arguments when issuing command** (positional parameters), or **prompting for data to store in a variable**. The later method can be accomplished by using the **read** command.
- Example:



Click on the cog icon to select different installed desktop environments.

```
read -p "Enter your name: " userName.
```

### **Mathematical Expressions**

- In the bash shell, data is stored in variable as text, not other data types (ints, floats, chars, etc) like in compiled programs like C or Java. In order to have a shell perform **mathematical operations**, number or variable need to be surrounded by two sets of parenthesis **((..))** in order to convert a number stored as text to a binary number.

- Examples

```
var1=5;var2=10  
echo "$var1 + $var2 = $((var1+var2))"
```

**Note:** shell does not perform floating point calculations (like **5/10**). Instead, other commands like **awk** or **bc** would be required for floating point calculations (decimals)

continue our use of the Bash Shell by examining commands will allow the Linux sysadmin to gather information about and manage their Virtual Machines.

#### **Perform the following steps:**

1. Start the **centos1** virtual machine, and stop the **centos2** and **centos3** virtual machines.
2. Switch to the **c7host** machine, and open a shell terminal.
3. Enter these admin commands into your **c7host** machine and note the result:

```
sudo virsh list  
sudo virsh list --all  
sudo virsh list --inactive
```

4. Now, shut-down your centos1 VM normally, and close the centos1 VM window.

5. Switch to your terminal and issue the command: `sudo virsh start centos1`
6. Using the appropriate command check to see if your centos1 VM is now running.
7. There are other commands that can be used (such as **suspend**, or **shutdown**). The "shutdown" command may not always work since it relies on the guest handling a particular ACPI event. Why do you think it is useful to have commands to manipulate VMs?



#### Virtual Machine Does not Shutdown from Command

If the Virtual machine fails to shutdown from the `virsh shutdown` command, then you can go to the **Virtual Machine manager** and **halt** or **shutdown** within the VM itself, then you can click the **PowerOff** button in the VM window. You'll want to avoid a forced shutdown since those are equivalent to yanking the power cord out of the wall on a physical machine!

Answer INVESTIGATION 3 observations / questions in your lab log book.

## INVESTIGATION 4: USING PYTHON TO AUTOMATE MANAGING VIRTUAL MACHINES

This week you have added some significant capabilities to your python scripting. The ability to run loops and make decisions makes your scripts much more powerful. In this investigation you will write a python script that backs up the centos1, centos2, and centos3 VMs, or lets the user specify which VMs they want backed up.

1. In your **bin** directory, create the file **backupVM.py**, and populate with our standard beginning

```
#!/usr/bin/env python3
# backupVM.py
# Purpose: Backs up virtual machines
#
# USAGE: ./backupVM.py
#
# Author: *** INSERT YOUR NAME ***
```



#### Python Scripting Tips:

##### String Methods

- The string data type has a number of methods that we can use on it to get a modified version of the data a string variable holds.
- Note that these don't change the existing string, just return a copy of it with some change. You can store this updated copy in a variable, or use it in a command.
- Some examples include:



```
# Date: *** CURRENT DATE ***
import os
currentuser = os.popen('whoami')
if currentuser.read() != 'root':
    print("You must be root")
    exit()
else:
    print('Backing up centos1')
    os.system('gzip < /var/lib/libvirt/images/centos1.qcow2 >
~YourRegularUsername/backups/centos1.qcow2.gz')
    print('Backing up centos2')
    os.system('gzip < /var/lib/libvirt/images/centos2.qcow2 >
~YourRegularUsername/backups/centos2.qcow2.gz')
    print('Backing up centos3')
    os.system('gzip < /var/lib/libvirt/images/centos3.qcow2 >
~YourRegularUsername/backups/centos3.qcow2.gz')
```

2. Try to run that script. You'll notice it does not work. No matter what you do, it always says you are not root.
3. Modify the print statement that tells the user they must be root to also include the current username, then run the program again.
4. It should print out root, but with an extra new-line. You may have noticed this in your other python scripts so far: the data we get from os.popen() has an extra new-line on the end. We will need to modify the string(s) it gives us a bit. See the side-bar for hints on how to do so.
5. Modify the if statement so it is just getting the current username, not the username and a newline. You can do this using several steps and several variables, but it can also be done in a single line.
6. Test your script to make sure it works. If it doesn't, go back and fix it. **Do not continue until it successfully makes backups of your VMs.**
7. There is a weakness to this script as written. Every time you run it, it will make a backup of all three VMs. But what if you only made a change to one of them? Do we really need to wait through a full backup cycle for two machines that didn't change? As the script is currently written, we do. But we can make it better. We've provided the scripts with some comments below.

```
8. #!/usr/bin/env python3
# backupVM.py
# Purpose: Backs up virtual machines
#
# USAGE: ./backupVM.py
#
# Author: *** INSERT YOUR NAME ***
```

- find() - locates a value (e.g. another string) if it is in this string.
- lower() - convert the entire string to lower-case
- replace() - finds a value in the string, and replaces it with a different value.
- split() - breaks the string up on a value, and returns a list.
- strip() - removes leading and trailing whitespace
- upper() - convert the entire string to upper-case
- There are many more options ([https://www.w3schools.com/python/python\\_ref\\_string.asp](https://www.w3schools.com/python/python_ref_string.asp)) than this, but you will get a lot of utility out of these.

### Methods in General

- Python is an object oriented programming language. In this course, that won't actually affect us much, but it does change how you run some commands.
- In bash, every command ran on its own (while you might provide the command some arguments, or pipe different data into it). In python, the modules we import, and the types of variables we use can (will) have commands built into them. For example: converting all letters in a string to UPPER CASE).
- While you could write a command in bash that would do that and pass it a variable holding a series of letters as an argument, in python that behaviour (and more) is built into the string type of variable.
- The actions that are built into types are called **methods**. To use a method on a variable just add .methodname() to the end



```
# Date: *** CURRENT DATE ***
```

```
import os
```

```
#Make sure script is being run with elevated permissions
```

```
currentuser = os.popen('whoami').read().strip()
```

```
if currentuser != 'root':
```

```
    print("You must be root")
```

```
    exit()
```

```
else
```

```
#The rest of this script identifies steps with comments 'Step
```

```
<something>'.
```

```
#This is not a normal standard for commenting, it has been done here to link the script
```

```
# to the instructions on the wiki.
```

```
#Step A: Find out if user wants to back up all VMs
```

```
#Step B-1:use the existing loop to back up all the VMs
```

```
    print('Backing up centos1')
```

```
    os.system('gzip < /var/lib/libvirt/images/centos1.qcow2 > ~YourRegularUsername/backups/centos1.qcow2.gz')
```

```
    print('Backing up centos2')
```

```
    os.system('gzip < /var/lib/libvirt/images/centos2.qcow2 > ~YourRegularUsername/backups/centos2.qcow2.gz')
```

```
    print('Backing up centos3')
```

```
    os.system('gzip < /var/lib/libvirt/images/centos3.qcow2 > ~YourRegularUsername/backups/centos3.qcow2.gz')
```

```
#Step B-2: They don't want to back up all VMs, prompt them for which VM they want to back up
```

```
#Step C: Prompt the user for the name of the VM they want to back up
```

```
#Step C-1: If the user chose Centos1, back up that machine.
```

```
#Step C-2: If the user chose Centos2, back up that machine.
```

```
#Step C-3: If the user chose Centos3, back up that machine.
```

9. Before the block that backs up each machine add a prompt to ask the user if they want to back up all machines. Use an if statement to check if they said yes (See comment 'Step A').

- if they did say yes, back up all the VMs using your existing block (Comment step B-1).
- If they didn't say yes, do nothing for now (you could even use python's pass statement).

10. Test your script to make sure it works. Check what happens if you say 'yes' to the prompt, and check what happens if you say things other than 'yes'.

11. Now we have a script that asks the user if they want to back up all VMS, and if they say they do, it does. But if they don't want to back up every VM, it currently does nothing.

12. Add an else statement at comment Step B-2 to handle the user not wanting to back up every VM. Inside that else clause (Comment step C) ask the user which VM they would like to back up (you can even give them the names of available VMs (Centos1, Centos2, Centos3).

13. Now nest an if statement inside that else (Comments C-1, C-2, and C-3) so that your script can handle what your user just responded with. If they asked for Centos1, back up Centos1. If they want to back up Centos2, only back up Centos2, etc. Hint: You might want to use elif for this.


14. Test your script again. You should now have a script that:

of the variable. Instead of getting back the value, you'll get something else (based on what that method does). Suppose we had a variable called course that currently had the value 'ops245'. Course codes are usually written in ALL-CAPS, so instead of just saying print(course), we could say print(course.upper())

- Makes sure the user is running the script with elevated permissions.
- Asks the user if they want to back up every VM.
- If they want to back up every VM, it backs up every VM.
- If the user does not want to back up every VM, the script asks them which VM they do want to back up.
- If the user selected a single VM, the script will back up that one VM.
- Now you may notice another issue with the script: The gzip lines are almost identical. The only difference in them is the name of the VM file being backed up. In the portion of code where you back up each machine individually (comment steps C-1, C-2, and C-3) try replacing the machine name in the gzip command with a string variable that holds the machine's name instead. Note that you will have to make use of string concatenation for this to work correctly.

## LAB 2 SIGN-OFF (SHOW INSTRUCTOR)

Follow the submission instructions for lab 2 on Blackboard.

 **Backup ALL of your VMs!**  
If you have successfully completed this lab, make a new backup of all of your virtual machines onto your USB Key.

### Perform the Following Steps:

1. Use the **virsh start** command to launch all the VMs (**centos1**, **centos2**, and **centos3**).
2. Inside each virtual machine, run **ip a** on the command line. Open a Terminal window in centos1 to do so. You'll need the IP address of each machine for the next steps.
3. Switch to your **c7host** VM, open a terminal, login as root, and change directory to **/root/bin**.
4. Issue the Linux command: **wget https://raw.githubusercontent.com/OPS245/labs/main/lab2-check.bash**
5. Give the **lab2-check.bash** file execute permissions (for the file owner).
6. Run the shell script and if any warnings, make fixes and re-run shell script until you receive "congratulations" message.
7. Arrange proof of the following on the screen:

#### ✓ All VMs:

- All 4 VMs **created** and **running**
- Proof of **yum updates** on ALL VMs (i.e. results from **yum update** command)

#### ✓ **c7host VM:**

- Run the **lab2-check.bash** script in front of your instructor (must have all OK messages)

#### ✓ Lab2 logbook notes completed.

1. Upload a screenshot of the proof listed above, the output file generated by the lab2-check.bash script, your log book, and your backupVM.py to blackboard.

## Practice For Quizzes, Tests, Midterm & Final Exam

1. What is the name of the CentOS installation program?
2. What is the name of the file created by the CentOS installation program?
3. Which type of installation works best for confirming compatibility with hardware before installation? Why?
4. Which type of installation works best for installing large numbers of computers? Why?
5. How can you reduce the number of software updates required immediately after installation?
6. How do you start and stop virtual machines?
7. How do you SSH into your virtual machines?
8. List the steps to install a VM from:
  - Downloaded iso file
  - Network install (without kickstart file)
  - Network install (with kickstart file)
9. What is the purpose of the virsh command?
10. How to start and stop VMs using the virsh command?
11. List the steps to correctly backup your VMs to a USB disk
12. List the steps to correctly restore your VMs from a USB disk to your c7host VM.
13. How can you prompt the user for data and store into a variable?
14. Show a few examples how loops can be used to error-check when prompting the user for data.
15. What does the command **rpm -qi centos-release** do and why is it important?
16. What is the difference between **rpm -q centos-release** and **uname -a**?

Retrieved from "[https://wiki.cdote.senecacollege.ca/w/index.php?title=OPS245\\_Lab\\_2&oldid=158470](https://wiki.cdote.senecacollege.ca/w/index.php?title=OPS245_Lab_2&oldid=158470)"

- 
- This page was last edited on 8 May 2022, at 23:17.