STELLA MARY'S COLLEGE OF ENGINEERING

(Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai&Accredited by NAAC)
Aruthenganvilai, Kallukatti Junction, Azhikal Post Kanyakumari District – 629 202, Tamil Nadu

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LAB MANUAL

CCS372 – VIRTUALIZATION

REGULATION - 2021

SEMESTER-V

2023-2024(Odd)

INSTITUTE VISION

To be a beacon of academic excellence, empowering future innovators with technical mastery to harness technology for positive global change.

INSTITUTE MISSION

- To cultivate a vibrant learning environment where students delve into the frontiers of technical knowledge, hone their problem-solving skills, and embrace innovation to transform ideas into solutions that address global challenges.
- To bridge the gap between technical brilliance and real-world impact by forging strong industry partnerships, fostering cutting-edge research, and nurturing entrepreneurial drive in our students, empowering them to build a better future through technology.
- To ignite the spark of intellectual curiosity within every student, equip them with the tools and knowledge. To become pioneers in their chosen fields, and guide them towards ethical and responsible use of technology for the betterment of humanity.

DEPARTMENT VISION

To be a leading department in computer science education and innovation, equipping students with the expertise to create and solve real-world challenges through ethical, responsible, and transformative technological solutions for global progress.

DEPARTMENT MISSION

M1: To provide a strong foundation in computer science principles, fostering innovative problem-solving and a deep-rooted commitment to developing socially responsible and impactful engineering solutions

M2: To bridge academia and industry by fostering partnerships and research that equips students with practical skills and entrepreneurial drive.

M3: To inspire intellectual curiosity and ethical responsibility, guiding students to apply technology for the advancement of humanity.

PROGRAMME OUTCOMES (POs):

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural,societal, and environmental considerations.

4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources,and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one‘s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Educational Objectives (PEOs)

PEO1:

Graduates will be competent in creating innovative technologies through inter- disciplinary research and comprehensive skills sets that are suitable for the global computing industry

PEO2:

Graduates will be capable of managing leading positions with a broad understanding of the application of ethics in evolving computer-based solutions for the societal needs.

PEO3:

Graduates will imbibe entrepreneurial qualities and develop their career by upgrading their, communication, analytical and professional skills constantly.

Program Specific Outcomes (PSOs)

At the completion of the programme, the students will be able to:

PSO1:

Use data management techniques and algorithmic thinking for Software Design and Development practices.

PSO2:

Develop reliable IT solutions based on the expertise in Distributed Applications Development, Web Designing and Networking for various societal needs and entrepreneurial practices ethically.

PSO3:

Manage multidisciplinary environments effectively through their interpersonal and analytical skills and be responsible members and leaders of the society.

RUBRICS FOR ASSESSING LABORATORY

Sl. No.	Criteria	Marks	Excellent (3) 91% - 100%	Good (2) 71% - 90%	Average (1) 50% - 70%	Poor (0) <50%
1	Observation	3	Gives clear idea about the aim and having the capability of both recording & analyzing the data much easier. (3)	Capability of both recording & analyzing the data much easier but no proper clarification about the objective. (2)	Gives clear idea about the target and has less capability of both recording & analyzing the data. (1)	Gives indistinct idea about the target and has less capability of both recording & analyzing the data & who feel difficult to follow the objectives. (<1)
2	Assessment	3	Have executed the system in an efficient way & make credible and unbiased judgments regarding the conduct of the experiments. (3)	Executed the system with less difficulty & has partial judgements regarding the overall system. (2)	Executed the system with less efficiency and has no judgements regarding the system. (1)	Incomplete system execution & lack of judgments regarding the system. (<1)
3	Submission	4	Followed all the instructions given in the procedure and submitted the observation books in time. (4)	Followed all the instructions given in the procedure with some assisting (3)	Followed some of the instructions given in the procedure& late in submission of note books. (2)	Trying to follow the instructions given in the procedure & late in submission of note books. (<1)

COURSE OBJECTIVES

- Understand the architecture and types of virtualization (Type 1 vs Type 2).
- Learn installation, configuration, and management of virtual machines using various tools.
- Gain hands-on experience with disk management, snapshots, and volume management.
- Understand and apply desktop virtualization techniques using VNC and Chrome Remote Desktop.
- Explore network virtualization with VLANs and RAID configurations.
- Develop skills to implement and manage nested virtualization and open-source hypervisors.

COURSE OUTCOMES

CO1: Create and manage virtual machines using Type 2 hypervisors (e.g., VMware, VirtualBox).

CO2: Configure virtual disks, perform shrink/extend operations, and use snapshots effectively.

CO3: Design and implement spanned, mirrored, striped, and RAID volumes for data management.

CO4: Demonstrate desktop virtualization using tools like VNC and Chrome Remote Desktop.

CO5: Set up and manage nested virtualization and open-source solutions like KVM.

CO6: Configure VLANs using Cisco Packet Tracer and understand network virtualization basics.

CONTENTS

Ex. No.	Experiment Name
1	Create type 2 virtualization in VMWARE or any equivalent Open Source Tool.
2	shrink virtual hard disks in Hyper-V and How to compact virtual hard disks in Hyper-V
3	Create RAID 5 volume
4	Desktop Virtualization using VNC
5	Desktop Virtualization using Chrome remote Desktop
6	Create type 2 virtualization on ESXI 6.5 server
7	Create a VLAN in CISCO packet tracer
8	Install KVM in Linux
9	Create Nested Virtual Machine(VM under another VM)

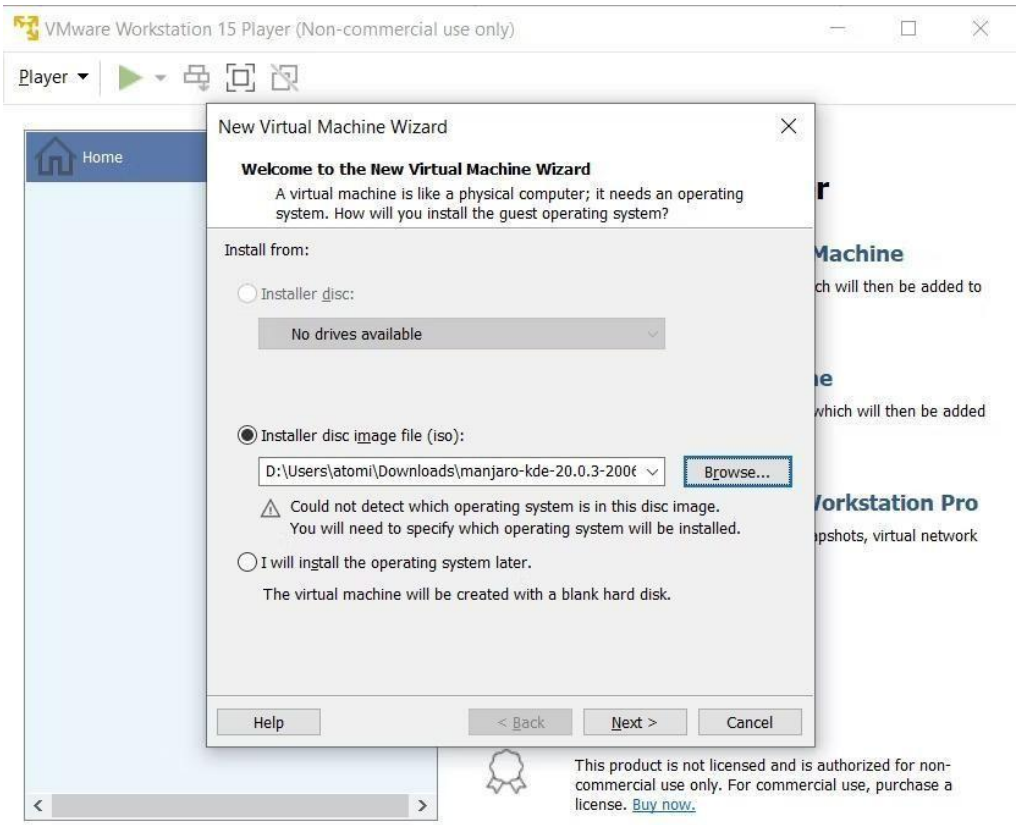
Signature of the Lab In-Charge

AIM:

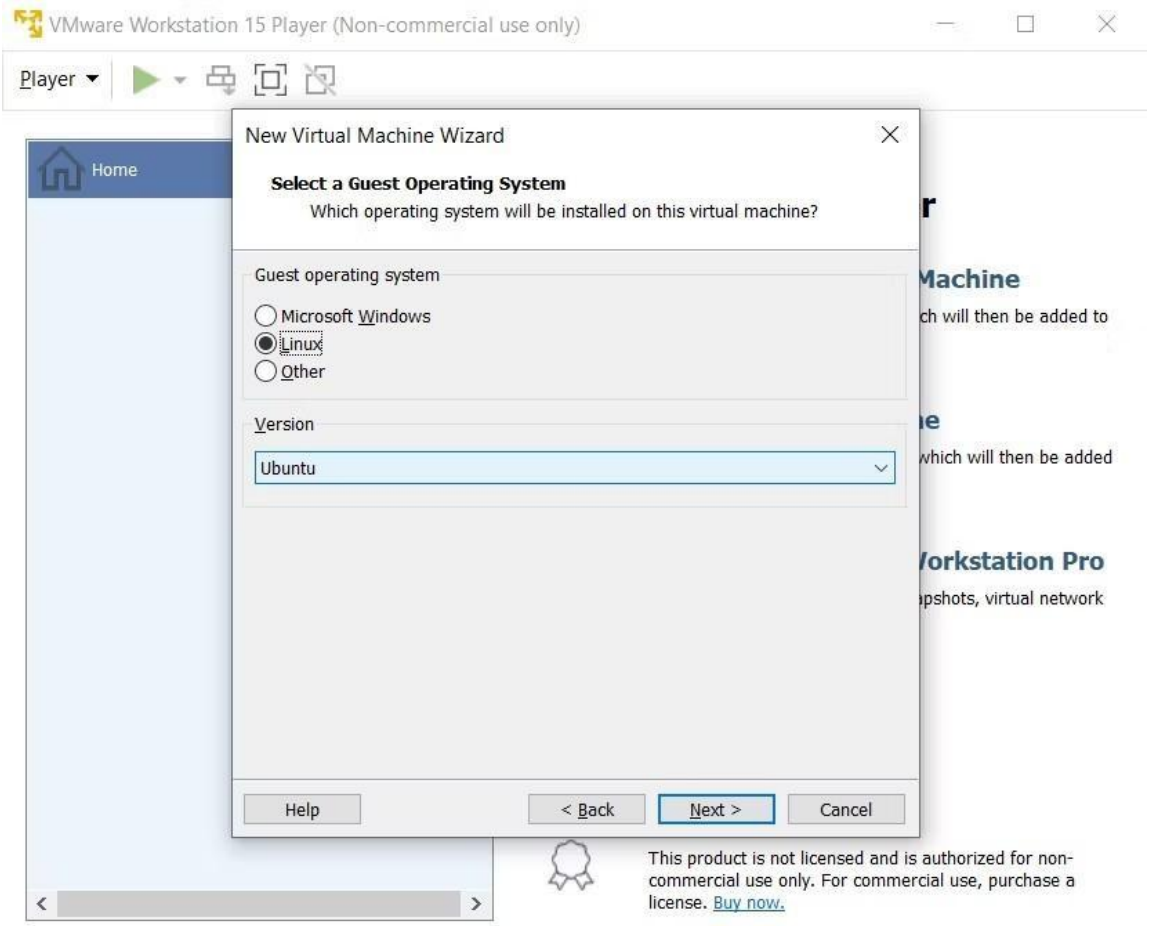
How to install operating system like linux using vmware.

Steps:

1. Click Create a New Virtual Machine
2. Select the default option, Installer disc image file (iso)
3. Click Browse to find the ISO file



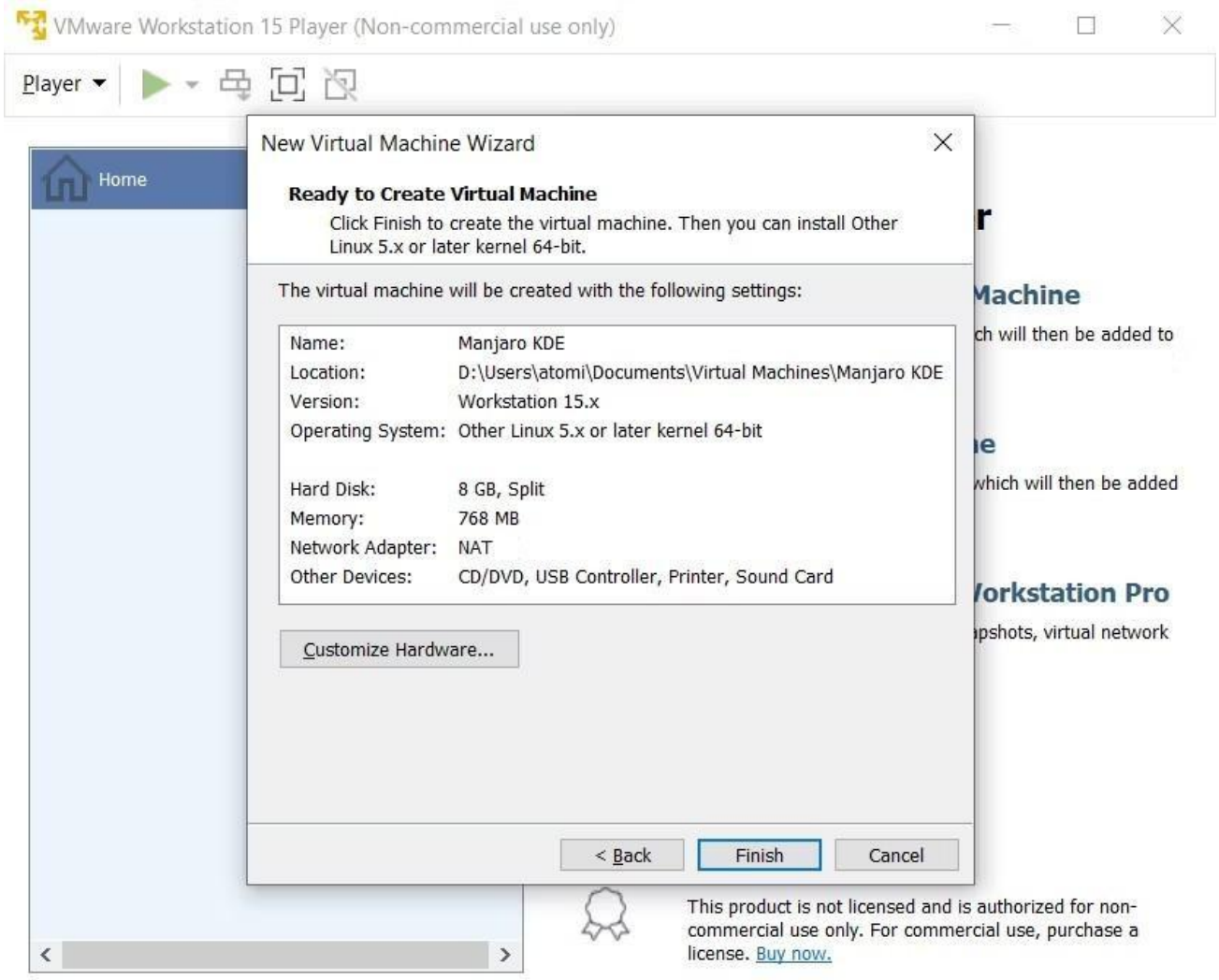
4. With "guest" OS selected, click Next
5. Select Linux as the Guest operating system type



6. Under Version, scroll through the list and select the OS
7. Click Next to proceed and if necessary, input a Virtual machine name
8. Confirm the storage Location and change if needed

With the operating system selected and configured, it's time to build the virtual machine.

1. Under Specify Disk Capacity adjust Maximum disk size if required (the default should be enough)
2. Select Split virtual disk into multiple files as this makes moving the VM to a new PC easy
3. Click Next then confirm the details on the next screen
4. If anything seems wrong click Back, otherwise click Finish

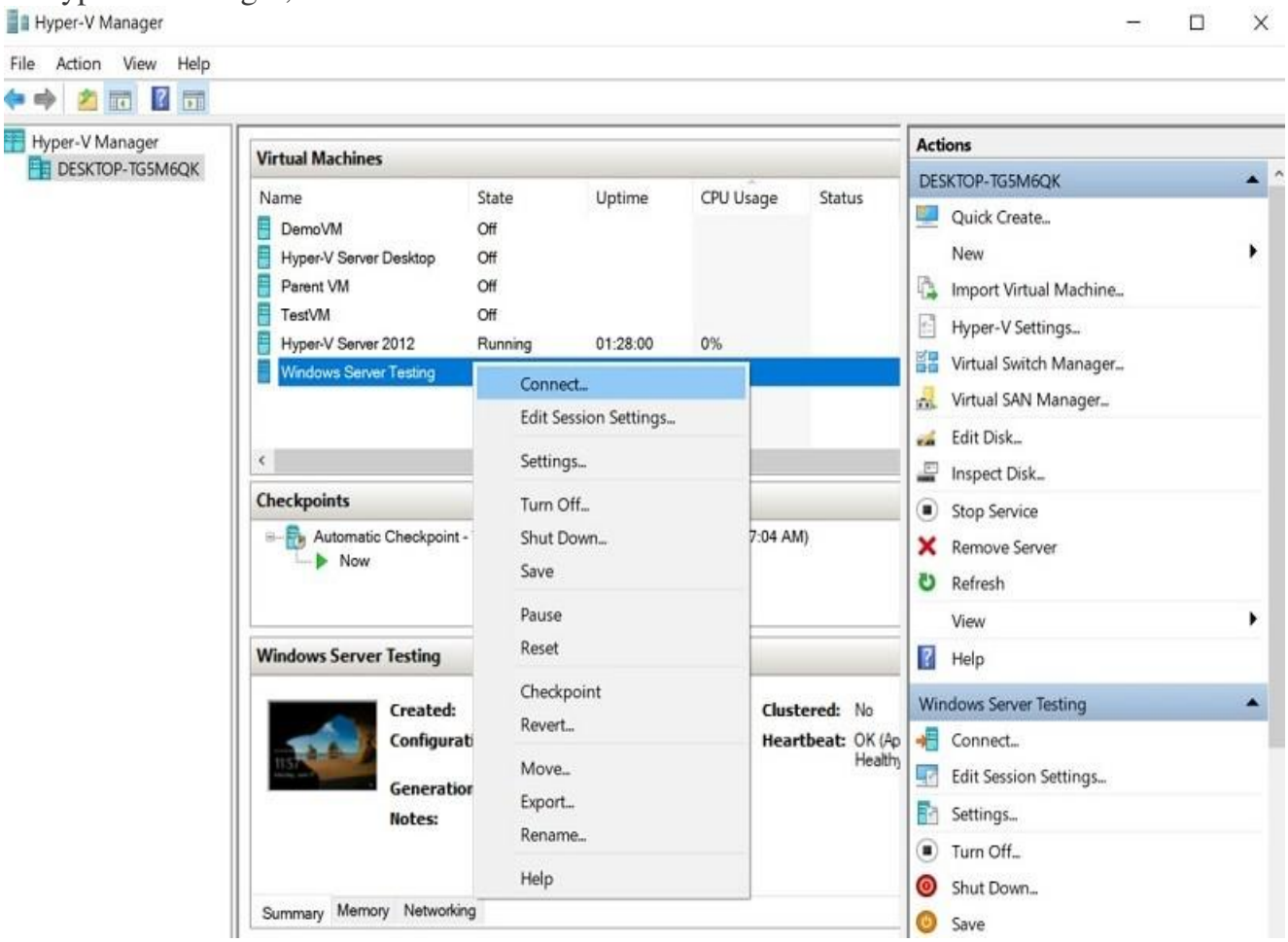

VMware Workstation 15 Player (Non-commercial use only)
Player
Home
New Virtual Machine Wizard
Ready to Create Virtual Machine
Click Finish to create the virtual machine. Then you can install Other Linux 5.x or later kernel 64-bit.
The virtual machine will be created with the following settings:
Name: Manjaro KDE
Location: D:\Users\atomi\Documents\Virtual Machines\Manjaro KDE
Version: Workstation 15.x
Operating System: Other Linux 5.x or later kernel 64-bit
Hard Disk: 8 GB, Split
Memory: 768 MB
Network Adapter: NAT
Other Devices: CD/DVD, USB Controller, Printer, Sound Card
Customize Hardware...
< Back
Finish
Cancel
Machine
which will then be added
Workstation Pro
This product is not licensed and is authorized for non-commercial use only. For commercial use, purchase a license. Buy now.

Ex No-2 shrink virtual hard disks in Hyper-V

Aim:

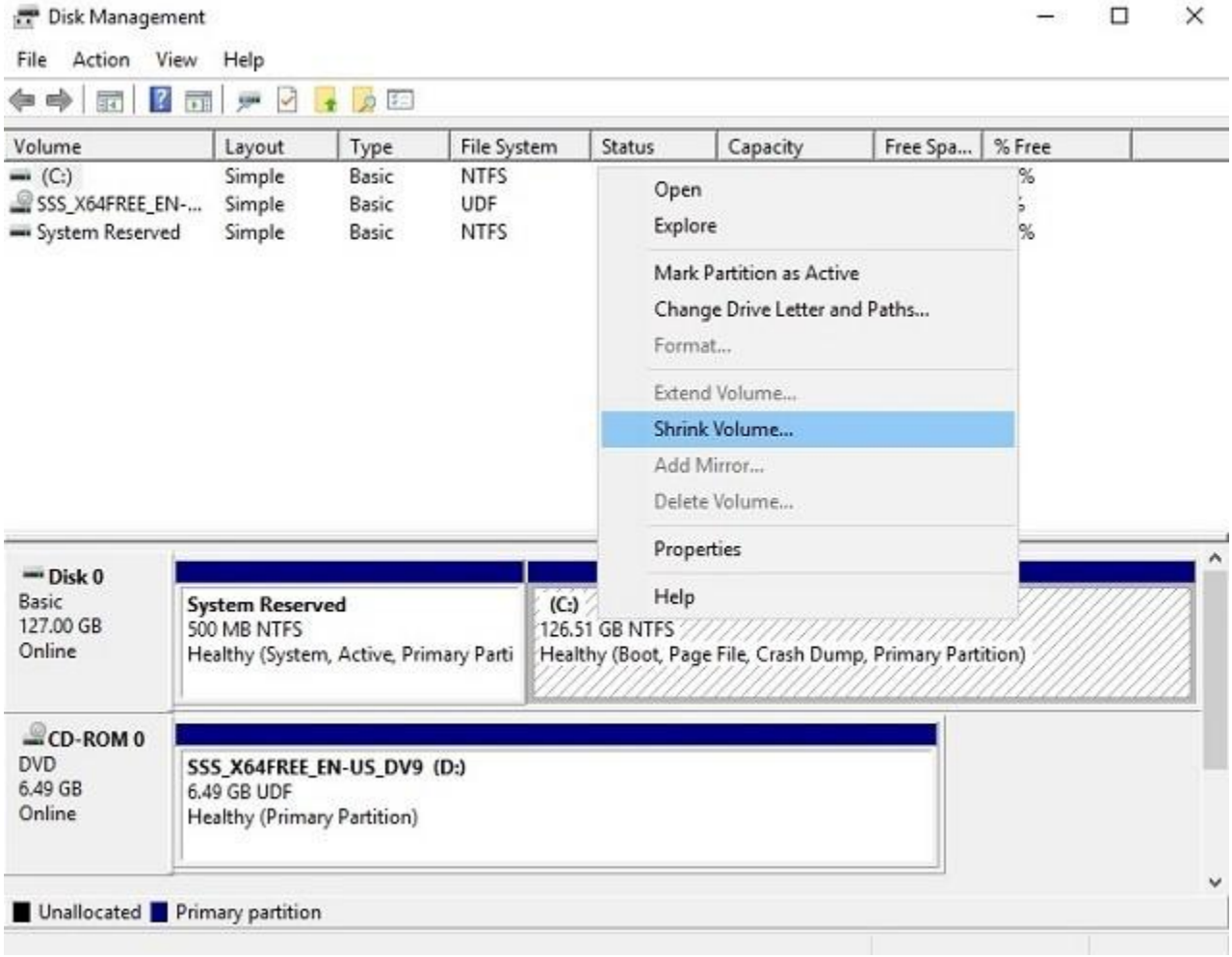
In Hyper-V environment, you can shrink, convert, expand, merge, reconnect or compact a virtual hard disk by editing the corresponding file using either the GUI or CLI tools.

1. In Hyper-V Manager, connect to the VM

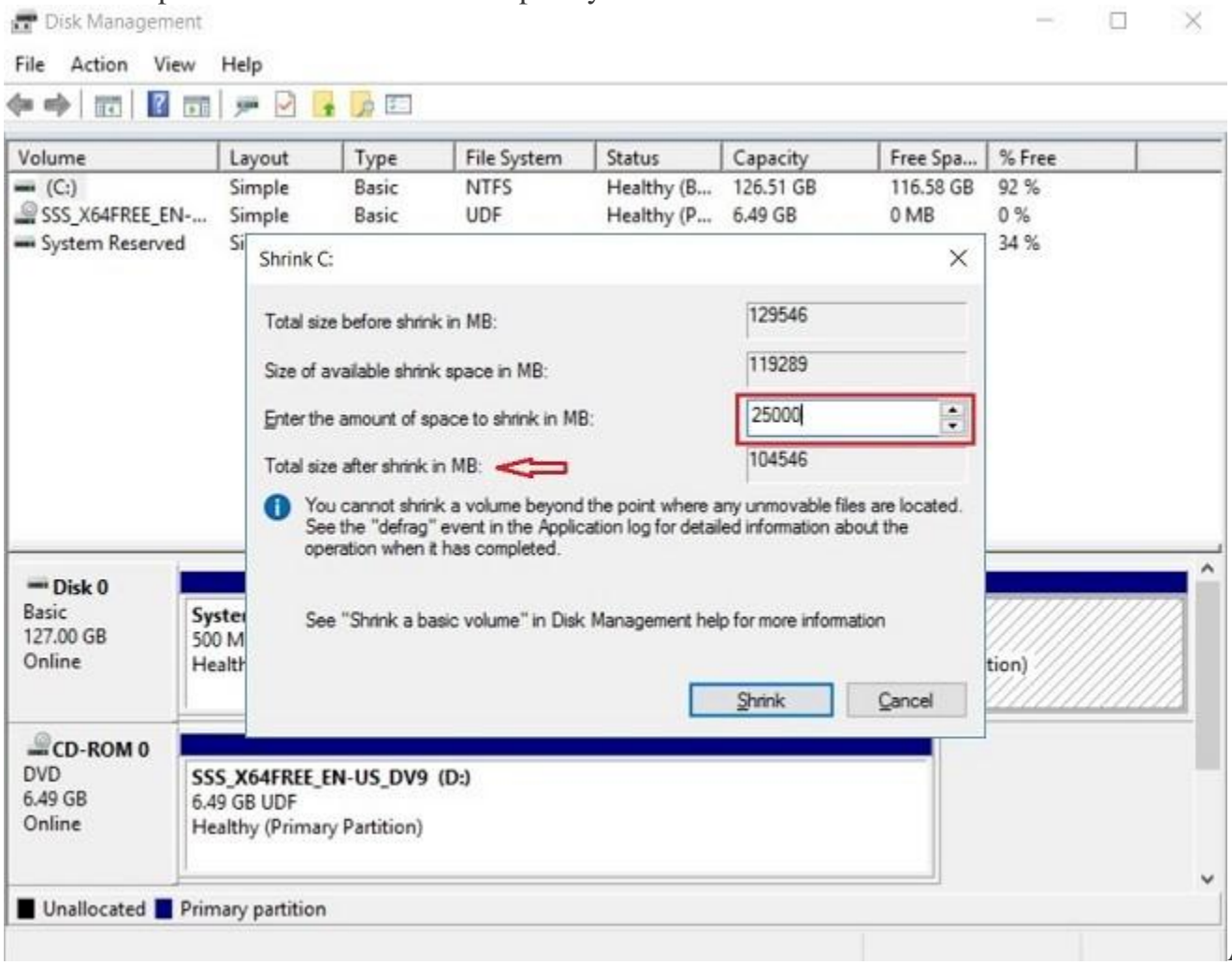


2. After logging into the VM, launch the Disk Management utility by typing **msc** in the search bar.

3. Right-click the disk volume you would like to shrink, and select the **Shrink Volume** option.

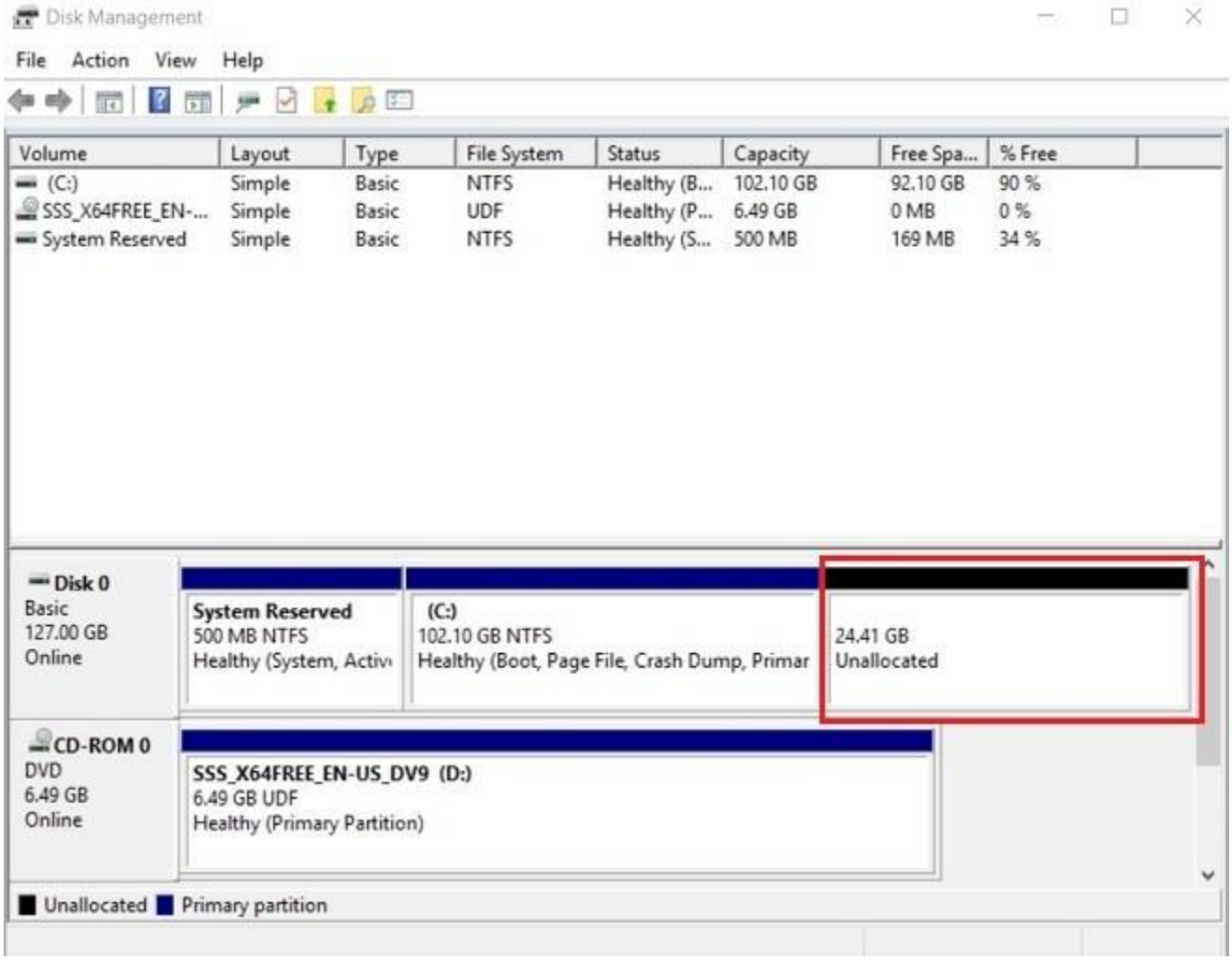


4. The next step is to enter the amount of space you wish to shrink in MB.



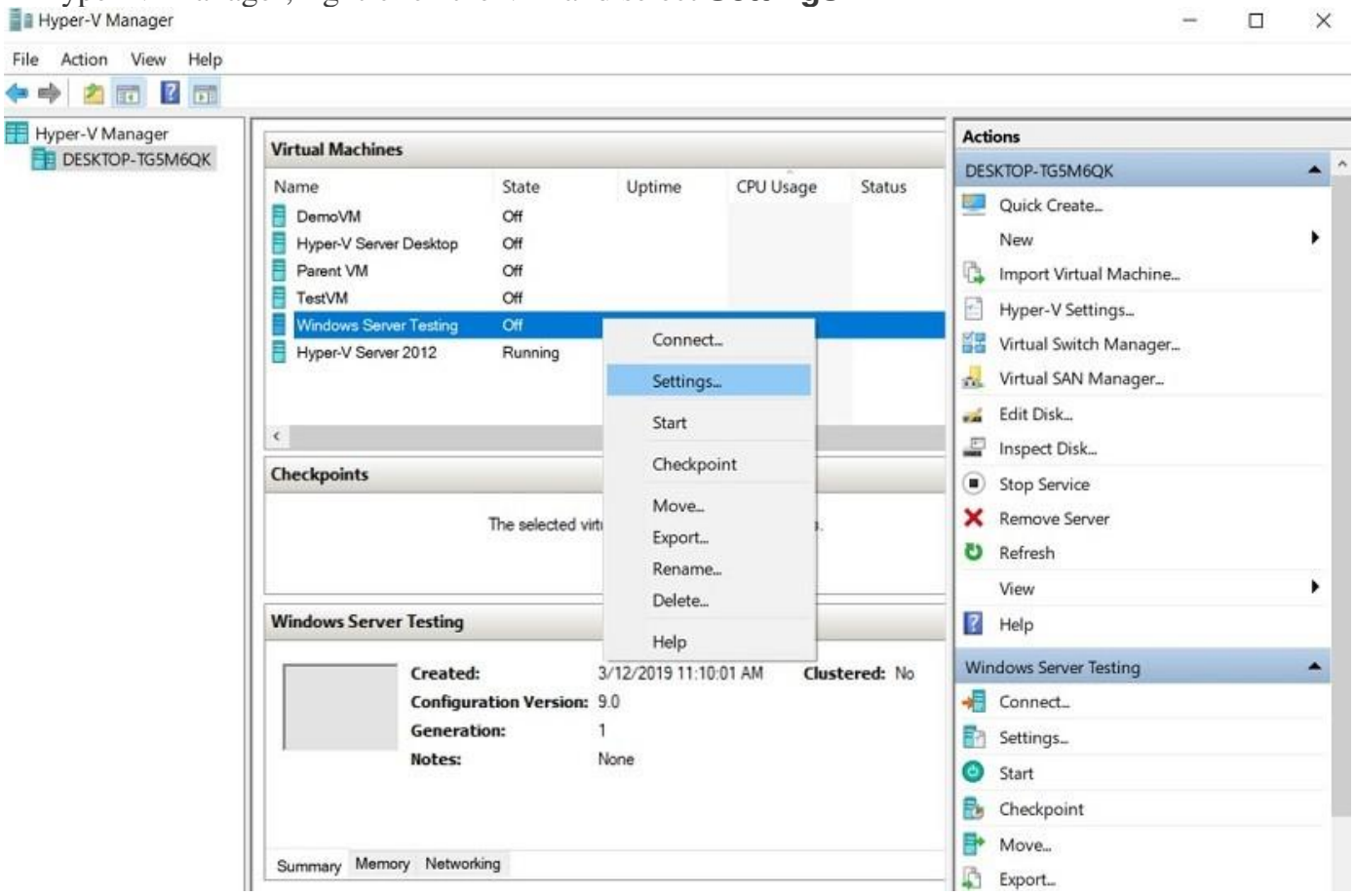
As you fill out this property, the **Total size after shrink in MB** value will change automatically, thus showing you what the disk storage capacity will be after the shrink operation is complete.

5. Click **Shrink** to start shrinking the disk volume. As a result, you will have roughly 25 GB of unallocated disk space.

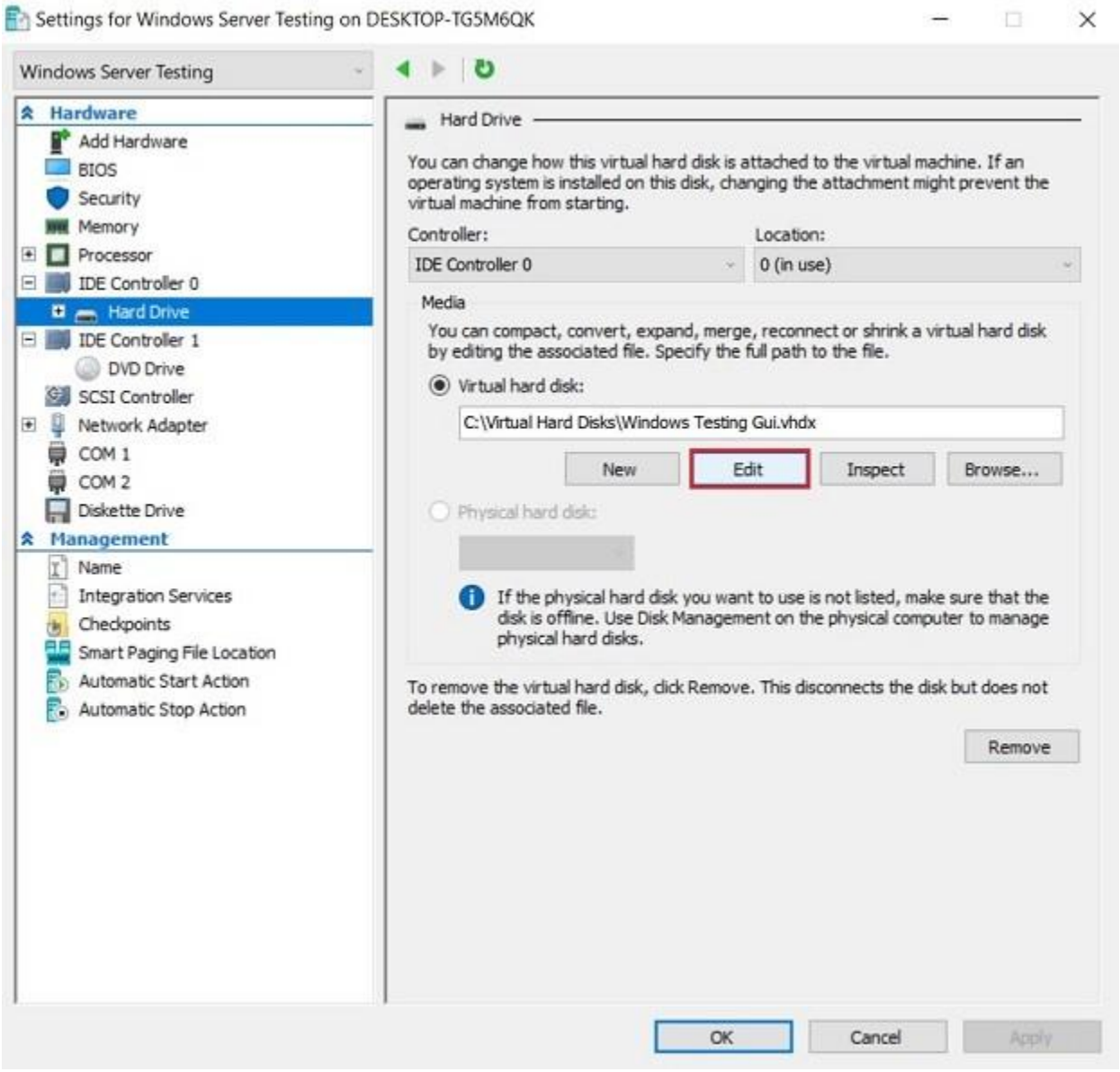


6. After that, shut down the VM.

7. In Hyper-V Manager, right-click the VM and select **Settings.**

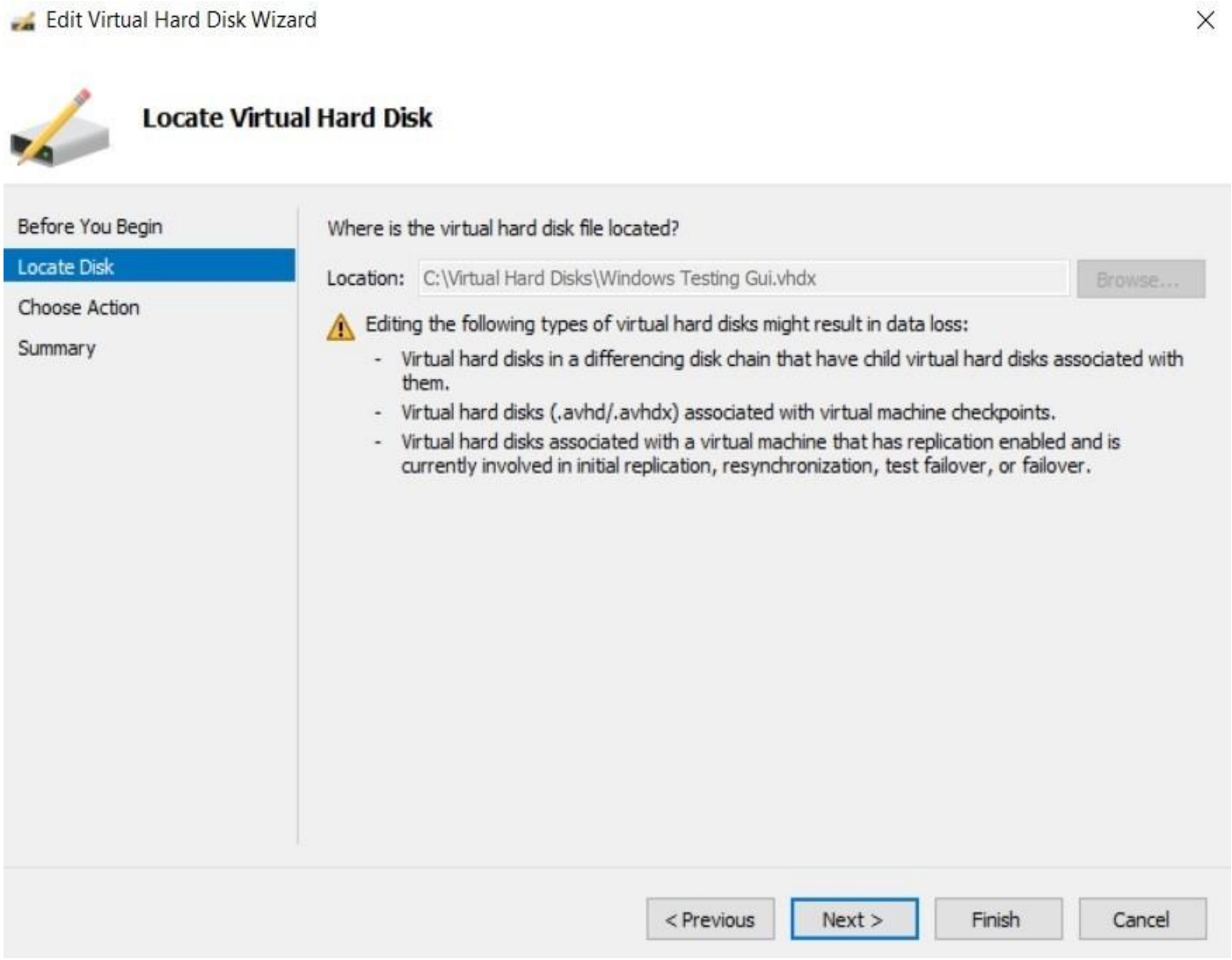


8. In the Hardware section, click **Hard Drive** to get access to the virtual hard disk settings.

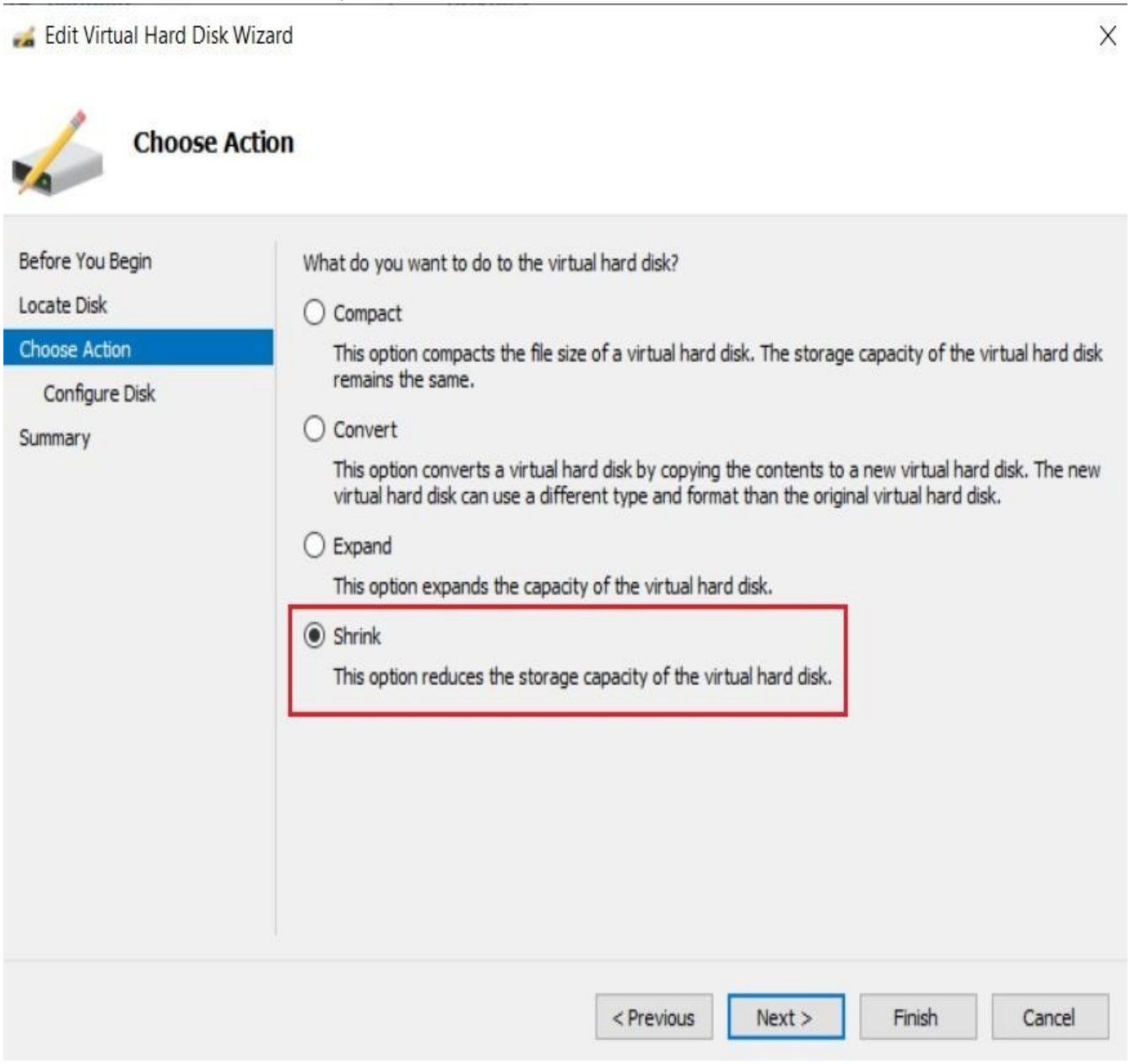


Select **Edit** to launch the Edit Virtual Hard Disk Wizard.

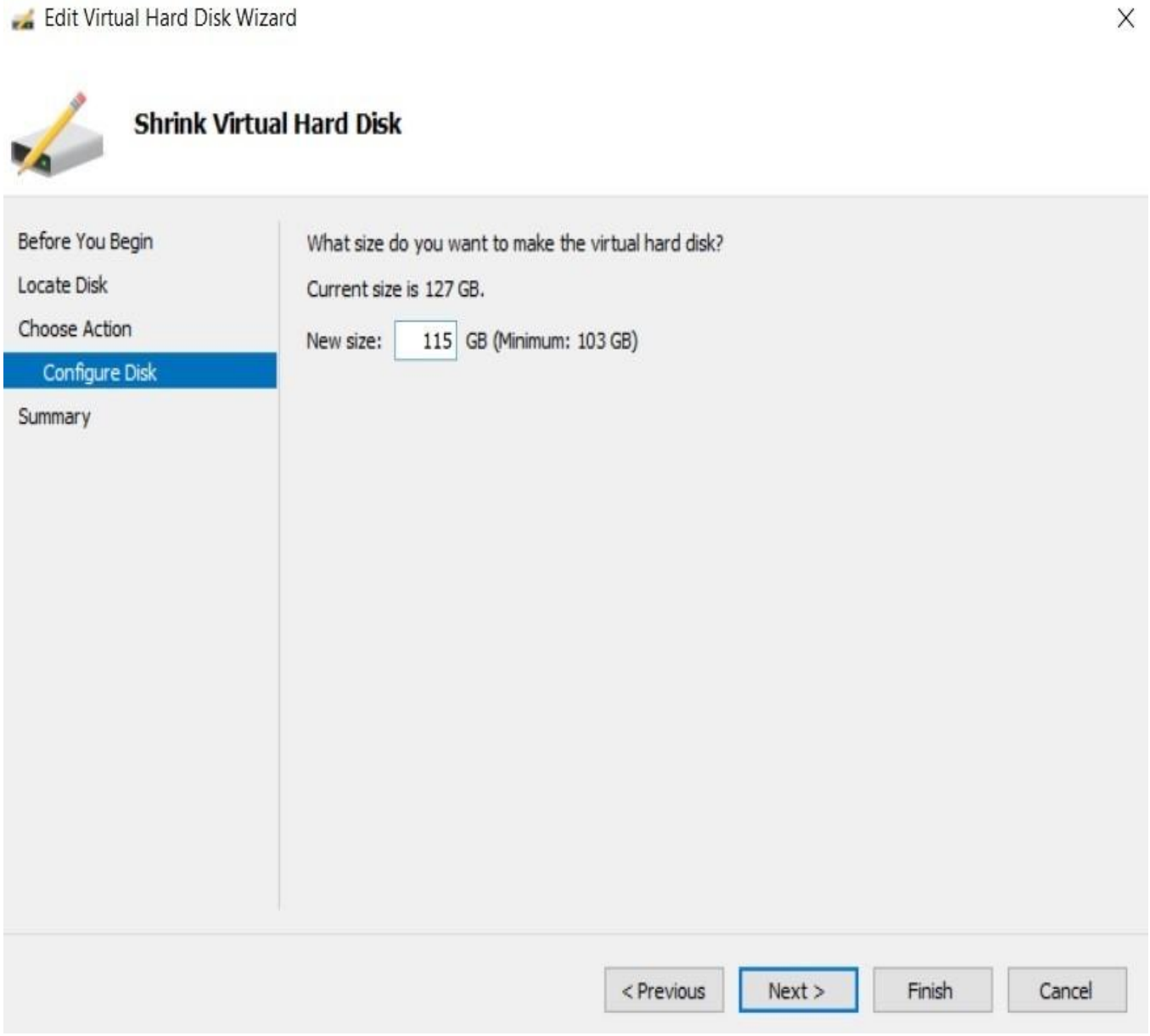
9. You can skip the Locate Disk step, as you have already selected which disk you would like to shrink. Click **Next.**



10. In the Choose Action section, select **Shrink** and click **Next.**

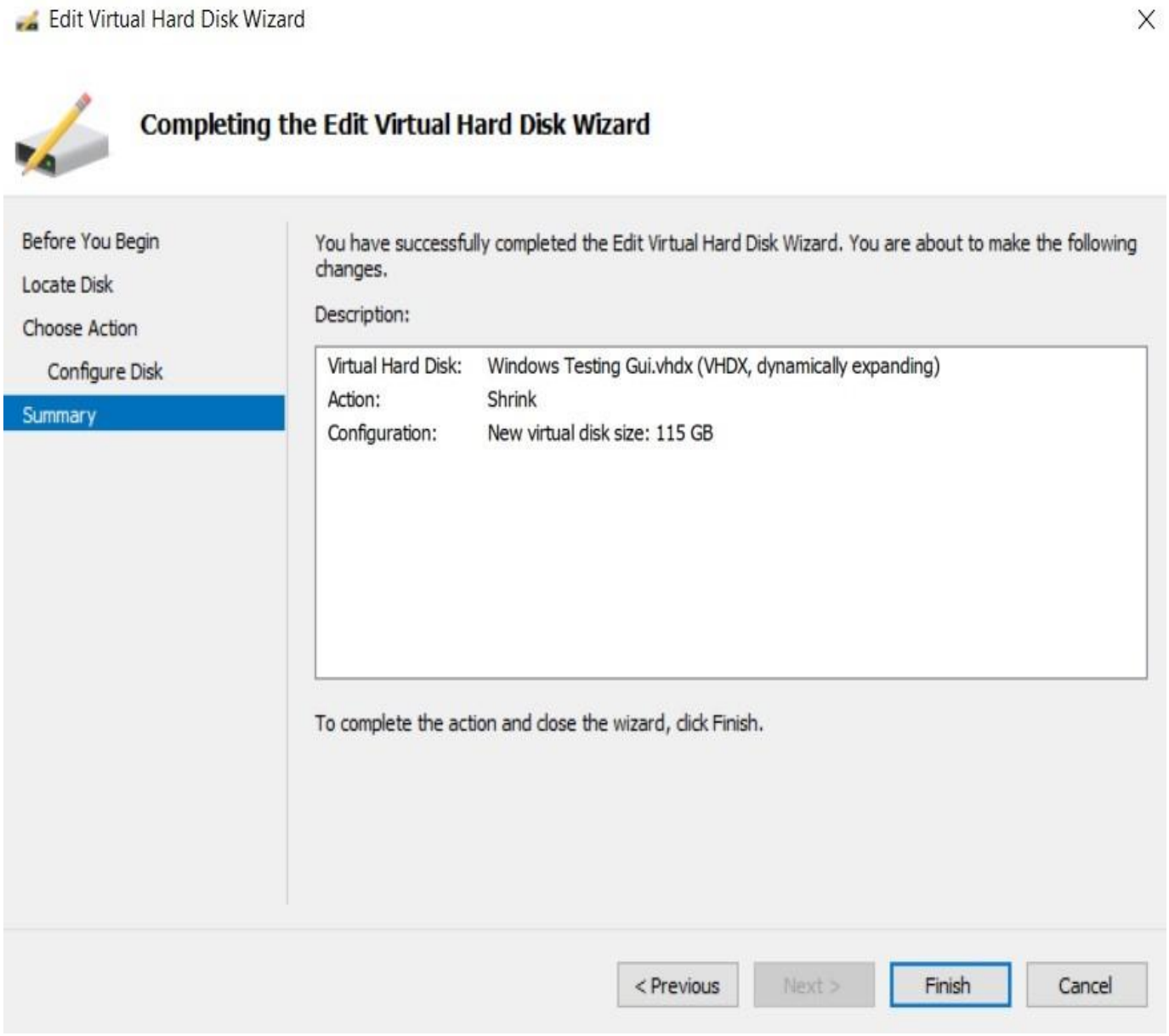


11. The next step is to configure the new size of the virtual hard disk.

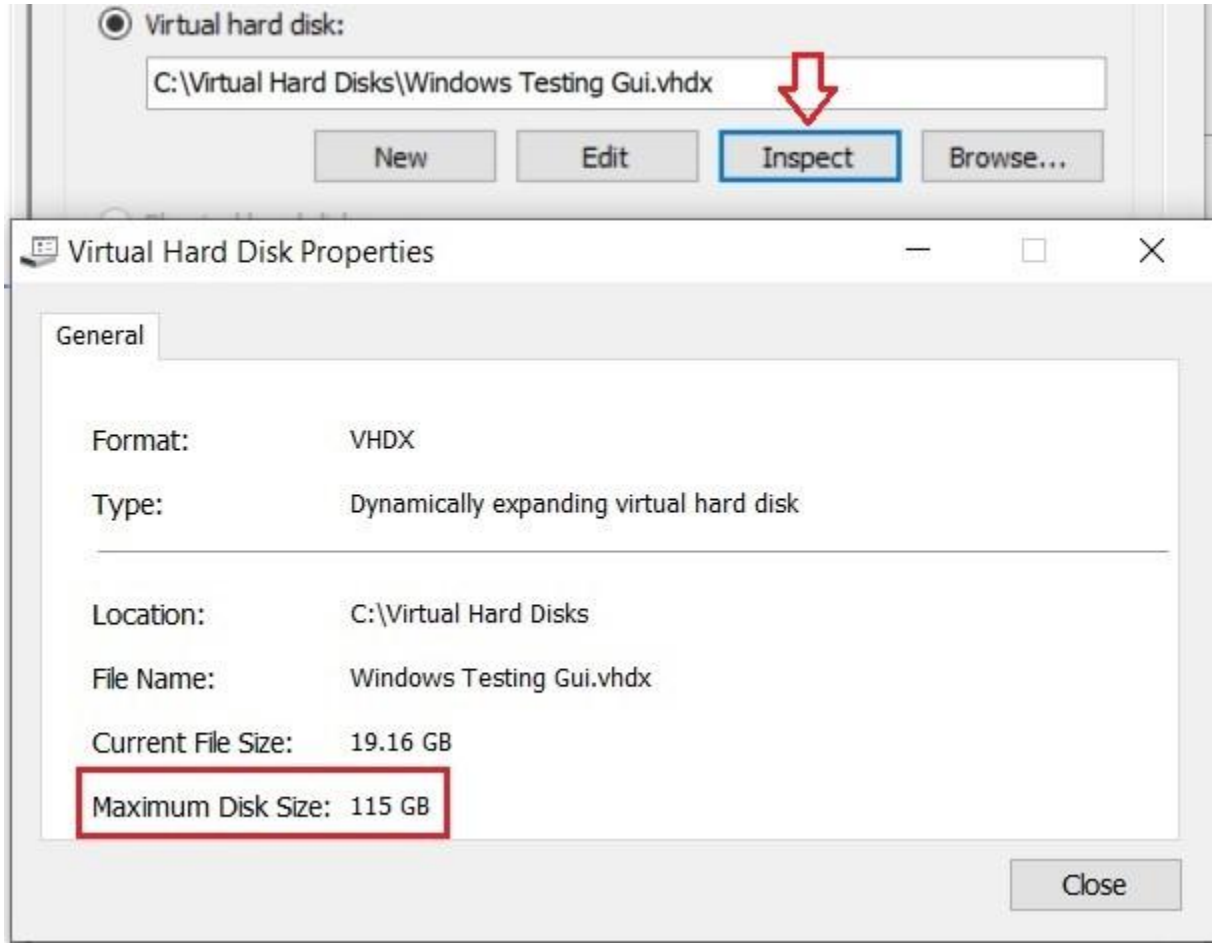


As you may notice, the difference between the current disk size and the minimum size is equal to the amount of extra disk space we have previously created inside the VM.

12. In the Summary section, you can look through the changes you are about to implement. Click **Finish** to complete the action and close the wizard.



13. Click **Inspect** to verify that the disk size has actually changed and you have successfully shrunk the virtual hard disk.



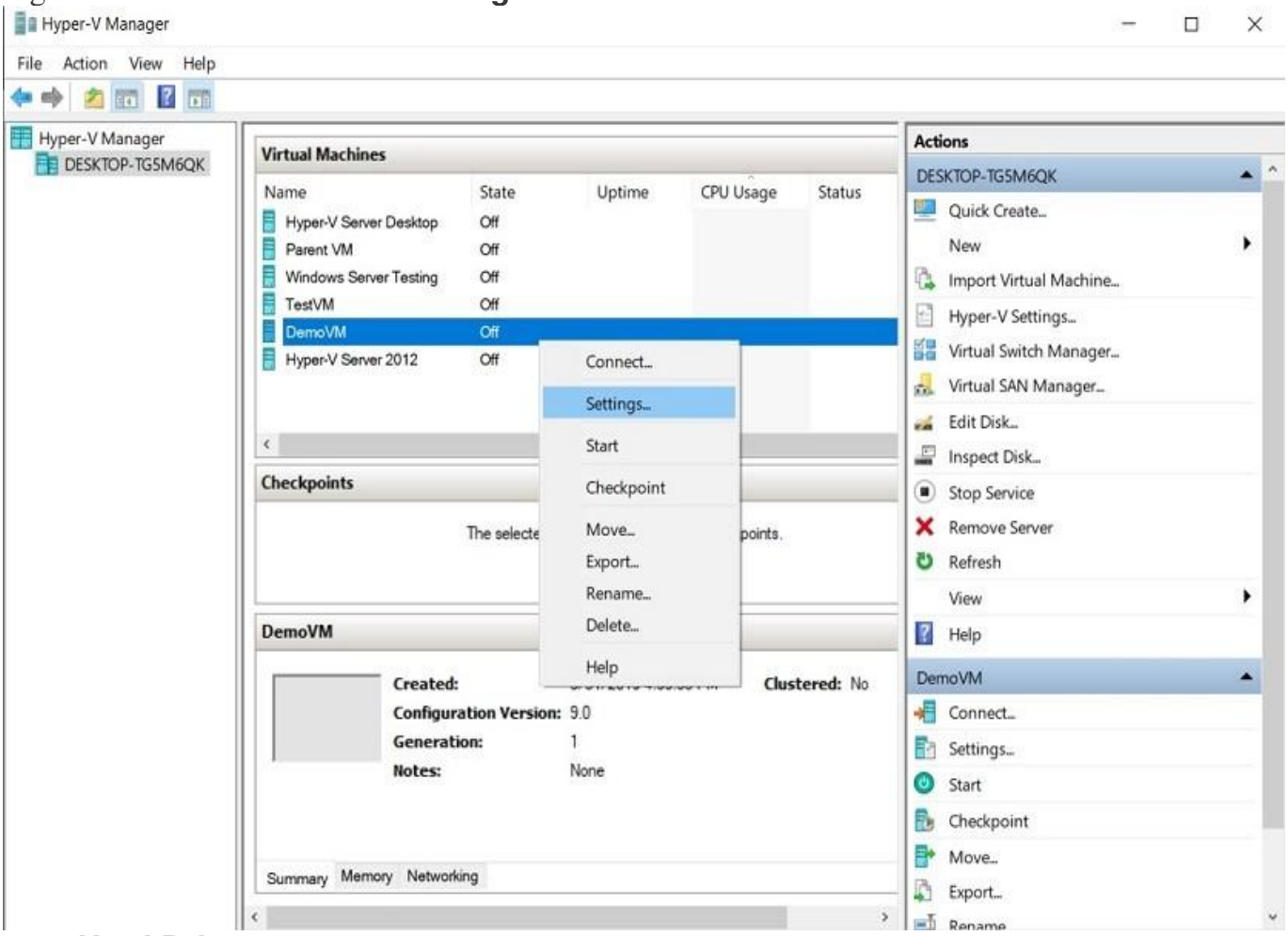
As you can see, the maximum disk size has been successfully reduced to 115 GB.

How to compact virtual hard disks in Hyper-V

Unlike shrinking, the compact operation doesn't reduce the storage capacity of the virtual hard disk. This action reduces the file size of a virtual hard disk by removing empty blocks from the file. Note that you cannot compact fixed virtual hard disks.

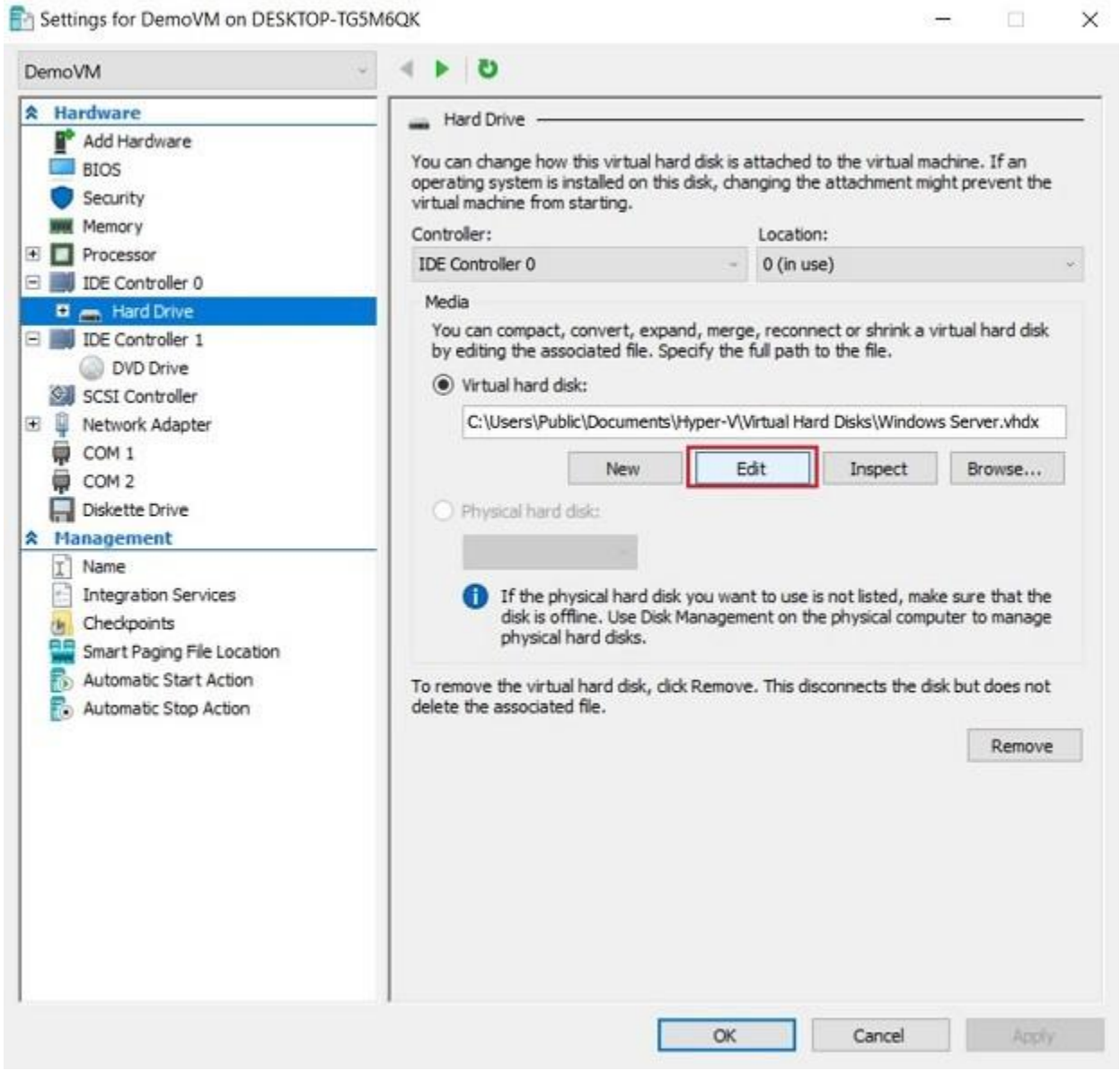
Before compacting or shrinking a virtual hard disk, it is recommended that you empty the Recycle Bin, defragment the disk, and then create its backup to protect critical data in case of disk failure. Remember that the VM using the disk needs to be turned off for the compact operation to work.

1. Open Hyper-V Manager.
2. Right-click the VM and select **Settings**.

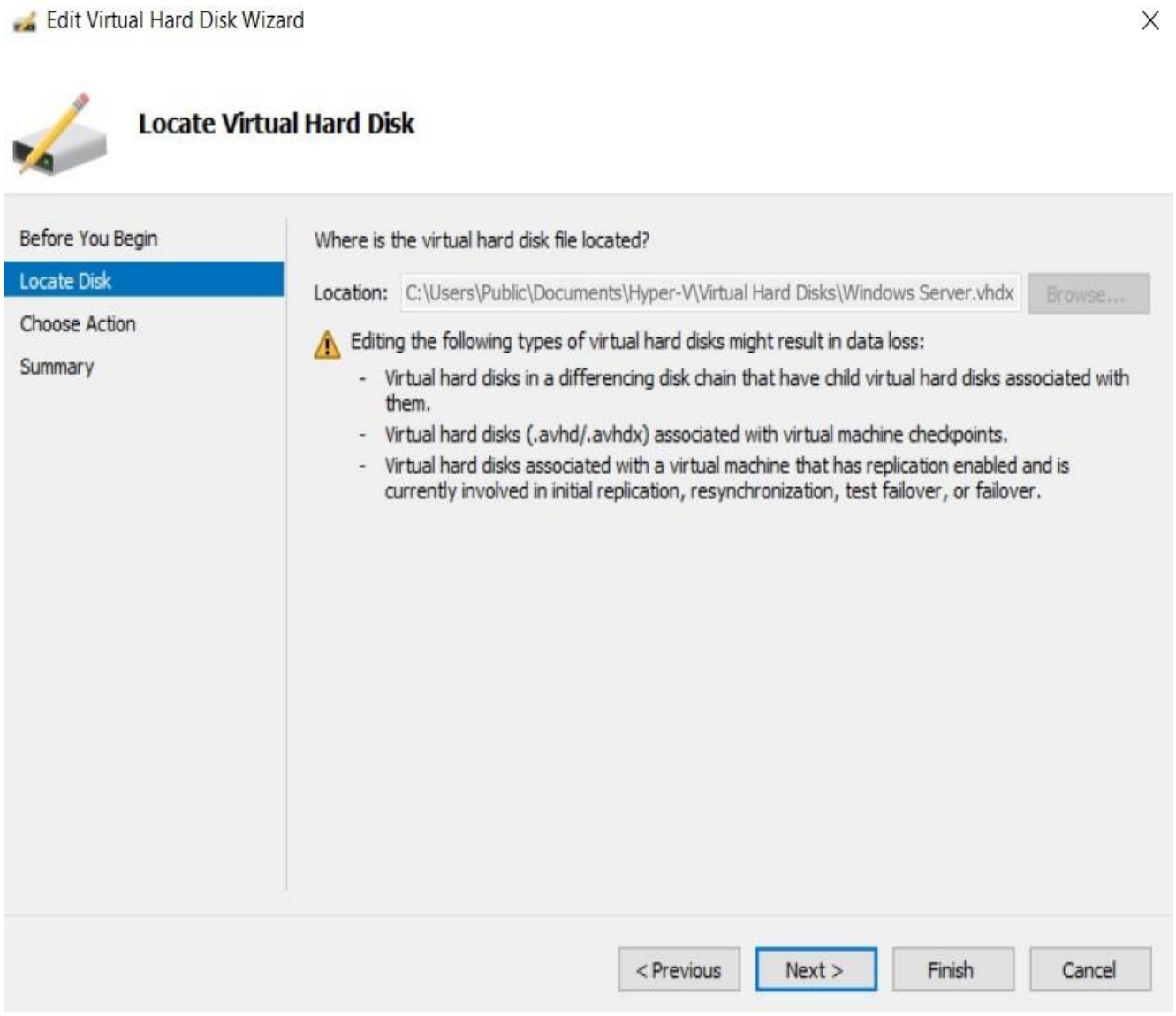


3. Click **Hard Drive** to get access to the virtual hard disk attached to this VM.

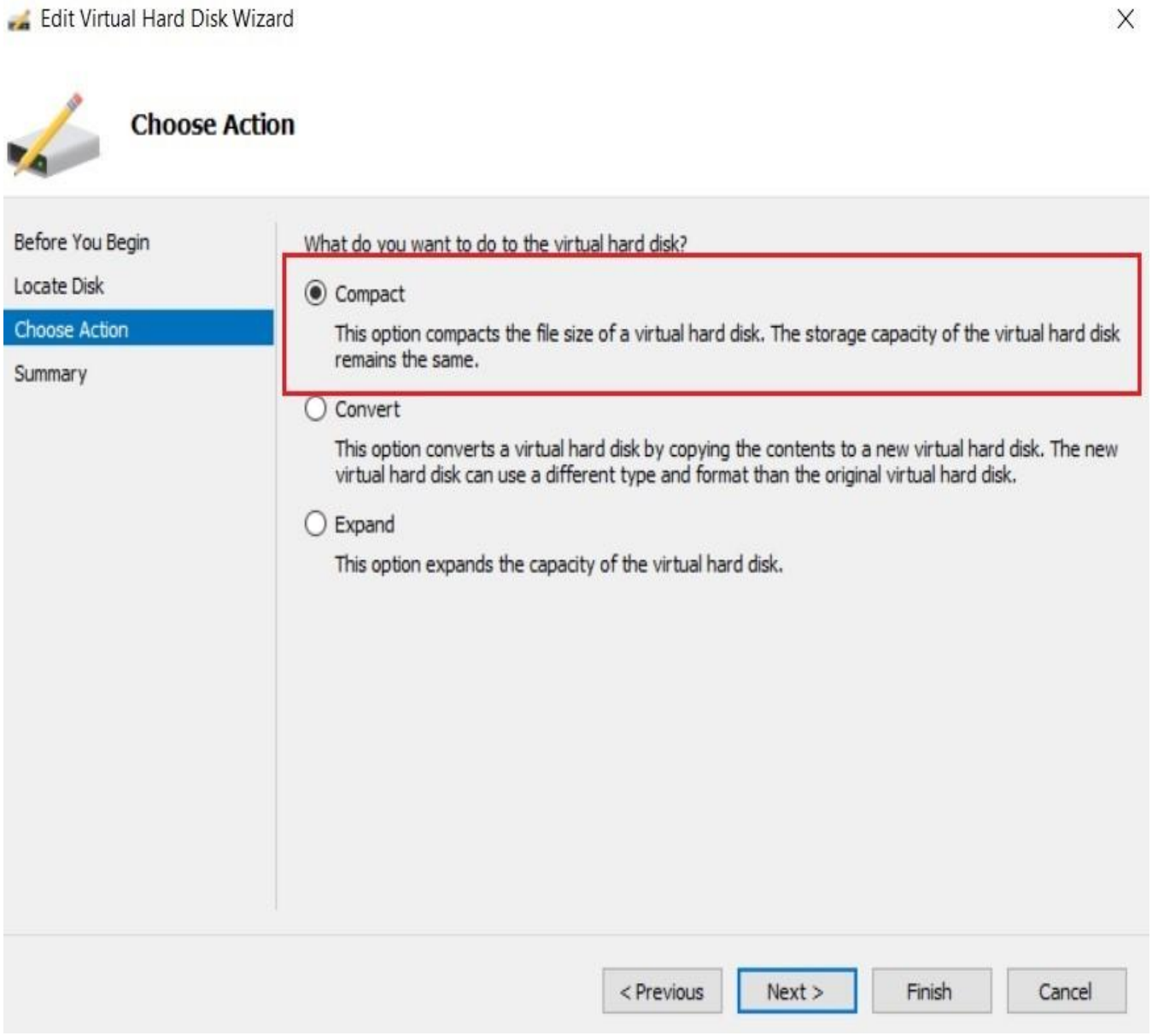
4. Click **Edit** to launch the Edit Virtual Hard Disk Wizard.



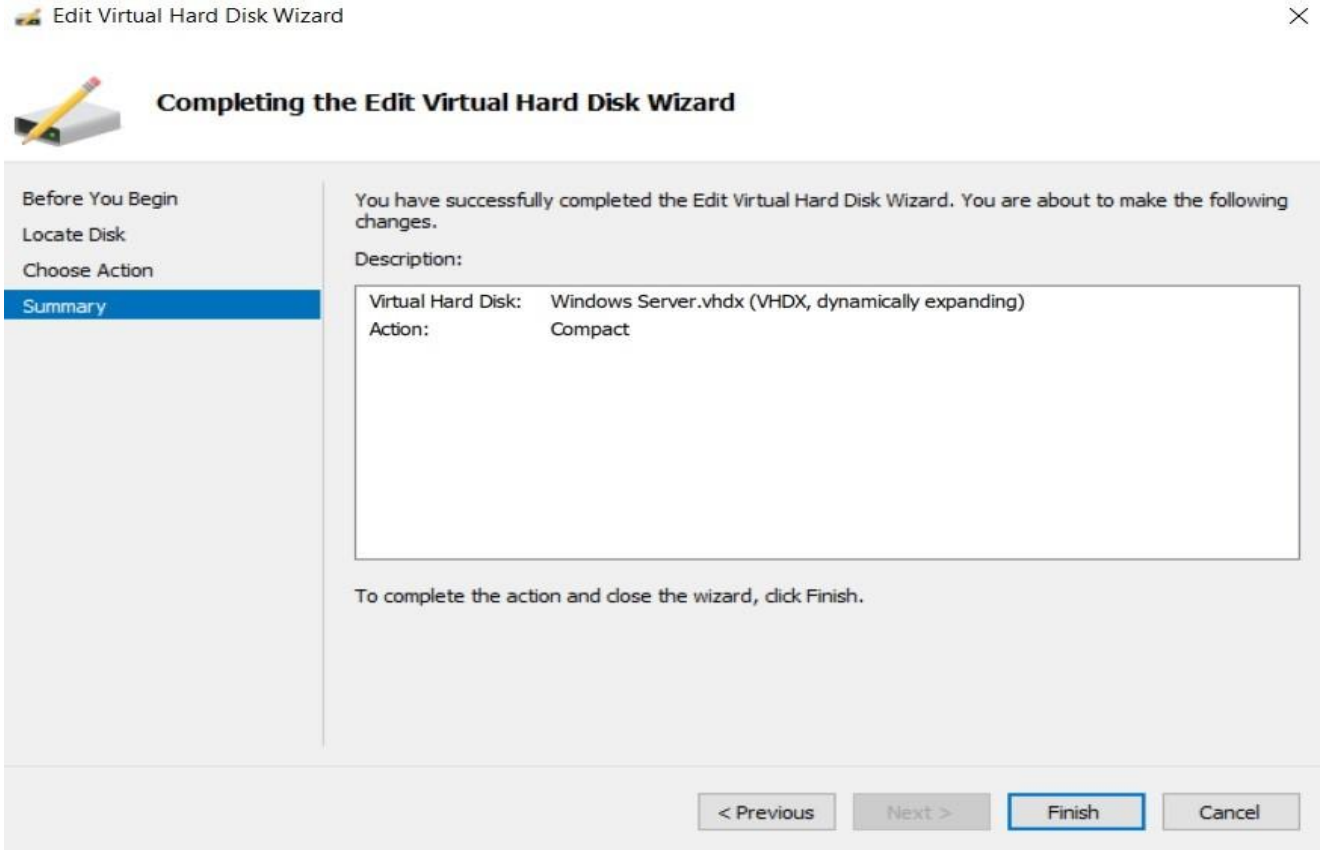
5. Skip the Locate Disk step as you have already selected the required virtual hard disk file, and click **Next.**



6. Select the **Compact** action.



7. In the Summary section, you can verify the changes that are about to be made to the virtual hard disk. Click **Finish** to complete the action and close the wizard.



8. Click **Inspect** to verify that the disk file size has been actually reduced.

Wrapping Up

So, why do you need to shrink or compact virtual hard disks? The answer is simple. The size of the virtual hard disk can be reduced only manually. If you decide to simply delete content, you will clear up space on the virtual hard disk, and not on the physical disk. To free up physical disk space, you need to either shrink or compact virtual hard disks, depending on your needs.
Before you can start with shrinking or compacting the virtual hard disk, it is always better to prepare for the worst-case scenario. An unexpected system error or bug can make your infrastructure entirely inaccessible. For this purpose, you need to ensure that data stored on the disk is securely protected and can be successfully recovered in case of disk failure.

NAKIVO Backup & Replication is a simple, yet powerful data protection solution, which can help you protect your VMware, Hyper-V, Nutanix, and AWS EC2 environments from anywhere and at any time.

RAID 0, RAID 1, RAID 5, RAID 10 Explained with Diagrams

by RAMESH NATARAJAN *on* AUGUST 10, 2010

RAID stands for Redundant Array of Inexpensive (Independent) Disks.

On most situations you will be using one of the following four levels of RAIDs.

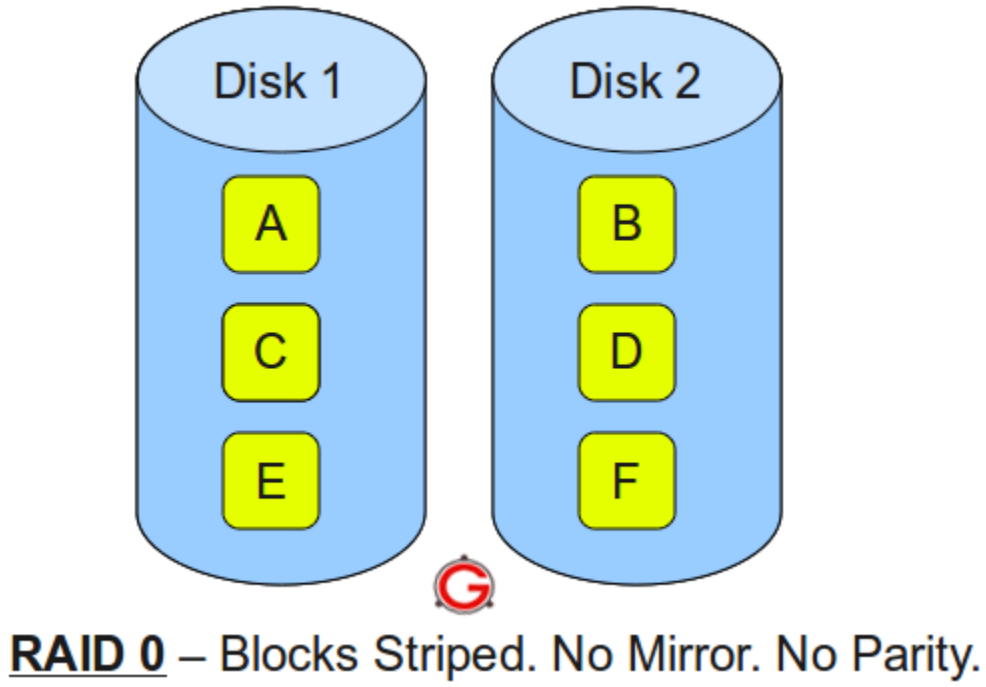
- RAID 0
- RAID 1
- RAID 5
- RAID 10 (also known as RAID 1+0)

This article explains the main difference between these raid levels along with an easy to understand diagram.

In all the diagrams mentioned below:

- A, B, C, D, E and F – represents blocks
- p1, p2, and p3 – represents parity

RAID LEVEL 0



Following are the key points to remember for RAID level 0.

- Minimum 2 disks.
- Excellent performance (as blocks are striped).
- No redundancy (no mirror, no parity).
- Don't use this for any critical system.

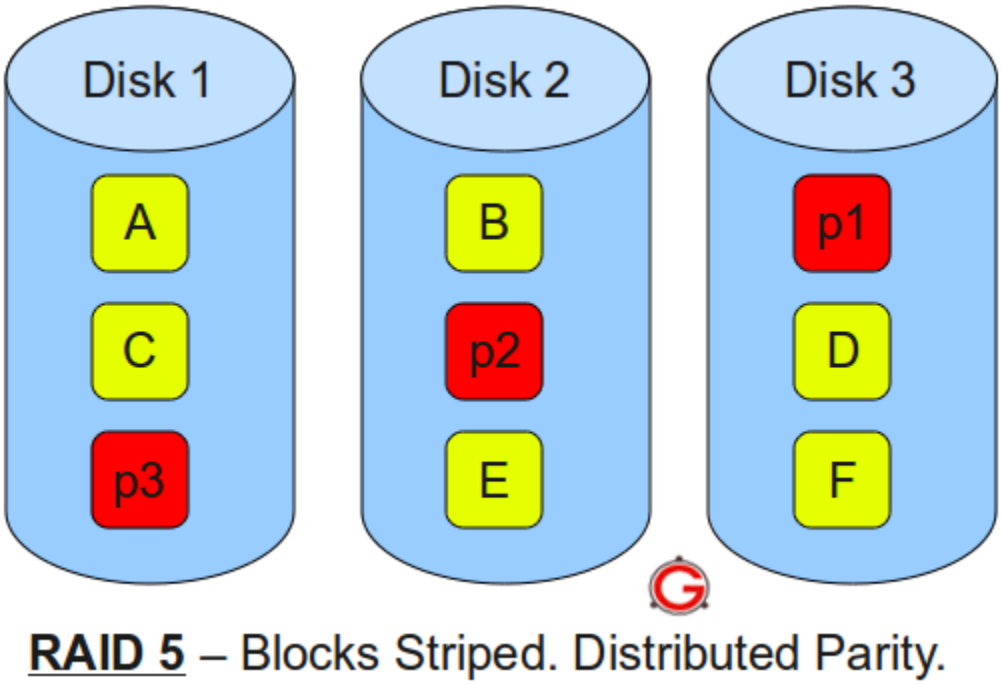
RAID LEVEL 1

Following are the key points to remember for RAID level 1.

- Minimum 2 disks.
- Good performance (no striping. no parity).
- Excellent redundancy (as blocks are mirrored).

RAID LEVEL 5

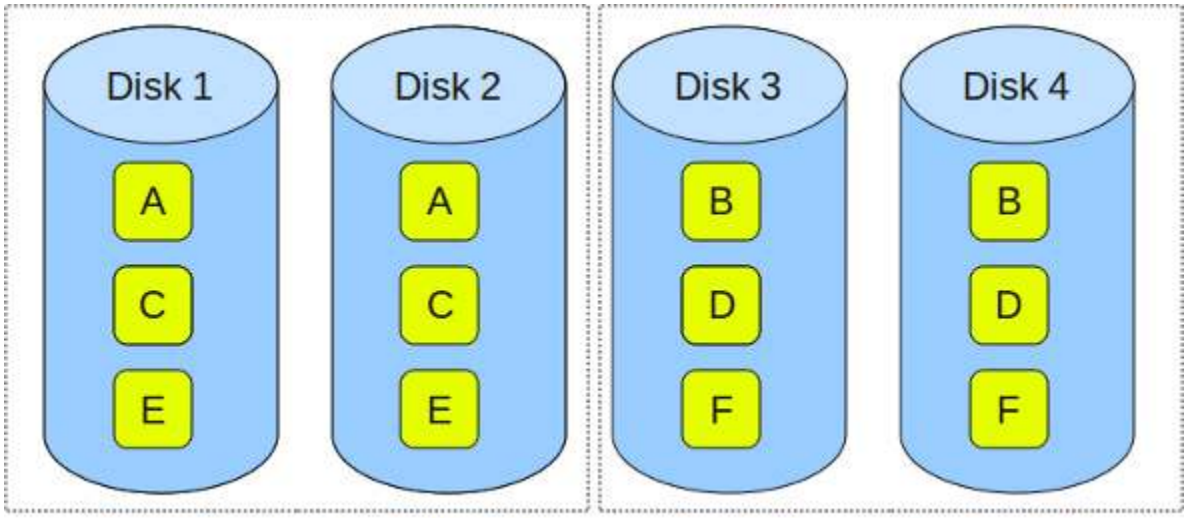
RAID LEVEL 5



Following are the key points to remember for RAID level 5.

- Minimum 3 disks.
- Good performance (as blocks are striped).
- Good redundancy (distributed parity).
- Best cost effective option providing both performance and redundancy. Use this for DB that is heavily read oriented. Write operations will be slow.

RAID LEVEL 10



RAID 10 – Blocks Mirrored. (and Blocks Striped)

Following are the key points to remember for RAID level 10.

- Minimum 4 disks.
- This is also called as “stripe of mirrors”
- Excellent redundancy (as blocks are mirrored)
- Excellent performance (as blocks are striped)
- If you can afford the dollar, this is the BEST option for any mission critical applications (especially databases).

3 . A. Desktop Virtualization using VNC

Introduction:

VNC or Virtual Network Computing is a platform-independent protocol that enables users to connect to a remote computer system and use its resources from a Graphical User Interface (GUI).

It's like remote controlling an application: the client computer's keystrokes or mouse clicks are transmitted over the network to the remote computer. VNC also allows clipboard sharing between both computers. If you come from a Microsoft Windows server background, VNC is much like the Remote Desktop Service, except it's also available for OS X, Linux, and other operating systems.

Like everything else in the networking world, VNC is based on the client server model: VNC server runs on a remote computer — your Droplet — which serves incoming client requests.

Goals:

In this tutorial we will learn how to install and configure a VNC server on CentOS 7. We will install the TigerVNC server which is freely available from the TigerVNC GitHub repository.

To demonstrate how VNC works, we will also install the GNOME desktop on your CentOS server. We will create two user accounts and configure VNC access for them. We will then test their connectivity to the remote desktop, and finally, learn how to secure the remote connection through an SSH tunnel.

The commands, packages, and files shown in this tutorial were tested on a minimal installation of CentOS 7. We would recommend the following:

- **Distro**: CentOS 7, 64-bit
- **Resource Requirements**: A Droplet with 2 GB RAM
- To follow this tutorial, you should use a sudo user. To understand how sudo privileges work, you can refer to this DigitalOcean tutorial
  Warning: You should not run any commands, queries, or configurations from this tutorial on a production Linux server. This could result in security issues and downtime.

Step 1 Creating Two User Accounts

First, we will create two user accounts. These accounts will remotely connect to our CentOS 7 server from VNC clients.

- joevnc
- janevnc
  Run the following command to add a user account for **joevnc**:

```
sudo useradd -c "User Joe Configured for VNC Access" joevnc
```

Then run the passwd command to change **joevnc**'s password:

```
sudo passwd joevnc
```

The output will ask us for new password. Once supplied, the account will be ready for login:

```
Changing password for user joevnc.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

Next, create an account for **janevnc**:

```
sudo useradd -c "User Jane Configured for VNC Access" janevnc
```

Step2: Installing GNOME Desktop

Now we will install GNOME desktop. GNOME is a collaborative effort: it's a collection of free and open source software that makes up a very popular desktop environment. There are other desktop environments like KDE, but GNOME is more popular. Our VNC users will use GNOME to interact with the server from its desktop:

```
sudo yum groupinstall -y "GNOME Desktop"
```

Depending on the speed of your network, this can take a few minutes.

Once the package group is installed, reboot the server:

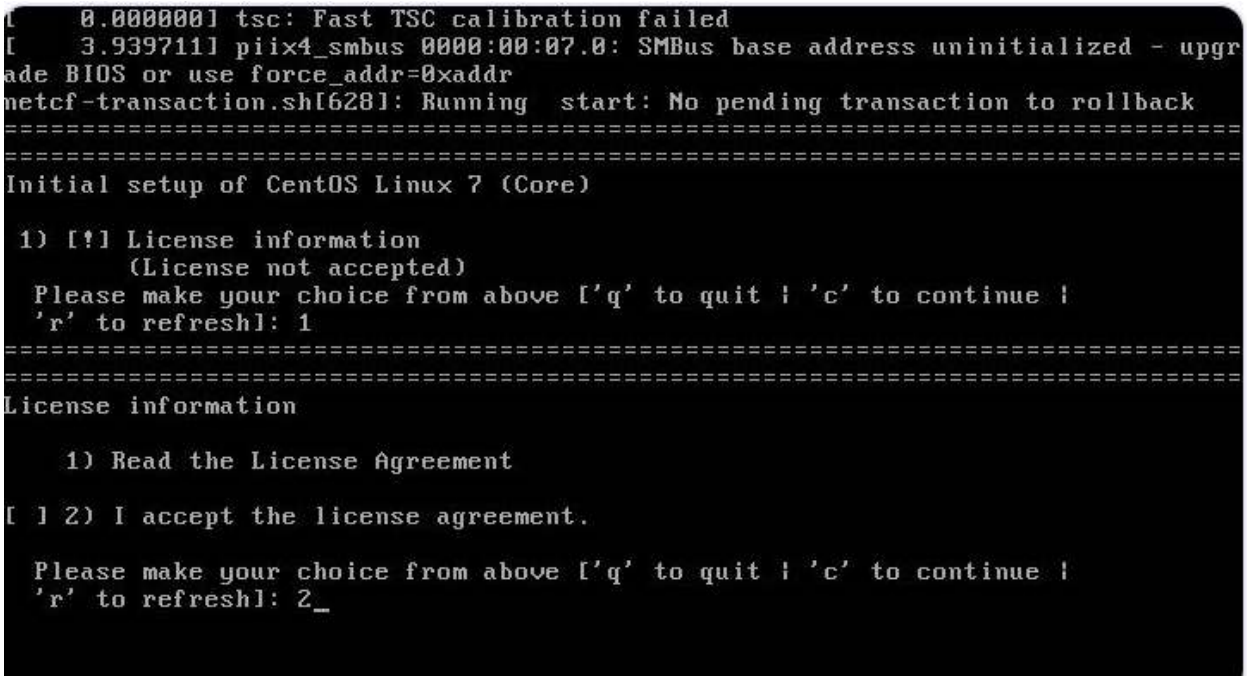
```
sudo reboot
```

Troubleshooting –Server Stuck at boot Phase

Depending on how your server has been set up, when the machine starts up it may remain in the boot phase showing a message like this:

```
Initial setup of CentOS Linux 7 (core)
1) [!] License information (Licence not accepted)
Please make your choice from above ['q' to quit | 'c' to continue |
'r' to refresh]:
```

To get past this, press **1** (license read), then **2** (accept licence), and then **C** (to continue). You may have to press **C** two or more times. The image below shows this:



If you don't see this error and the boot process is smooth, all the better – you can move on to the next step.

Step3:

TigerVNC is the software that will allow us to make a remote desktop connection.

Install the Tiger VNC server:

```
sudo yum install -y tigervnc-server
```

This should show output like the following:

```
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile

. . .

Running transaction
  Installing : tigervnc-server-1.2.80-0.30.20130314svn5065.el7.x86_64
1/1
```

```
  Verifying  : tigervnc-server-1.2.80-0.30.20130314svn5065.el7.x86_64
1/1

Installed:
  tigervnc-server.x86_64 0:1.2.80-0.30.20130314svn5065.el7

Complete!
```

Now we have VNC server and the GNOME desktop installed. We have also created two user accounts for connecting through VNC.

Step4: Configuring VNC Service for Two Clients:

VNC server doesn't start automatically when it's first installed. To check this, run the following command:

```
sudo systemctl status vncserver@:.service
```

The output will be like this:

```
vncserver@:.service - Remote desktop service (VNC)
   Loaded: loaded (/usr/lib/systemd/system/vncserver@.service; disabled)
   Active: inactive (dead)
```

You can also run this command:

```
udo systemctl is-enabled vncserver@.service
```

This should show output like this:

```
disabled
```

Note:

So why is it disabled? That's because each user will start a separate instance of the VNC service daemon. In other words, VNC doesn't run as one single process that serves every user request. Each user connecting via VNC will have to start a new instance of the daemon (or the system administrator can automate this).

CentOS 7 uses the systemd daemon to initiate other services. Each service that natively runs under systemd has a *service unit file* that's placed under the `/lib/systemd/system` directory by the yum installer. Processes that get started automatically at boot time have a link to this service unit file placed in the `/etc/systemd/system/` directory.

In our case, a generic service unit file was created in the `/lib/systemd/system/` directory, but no link was made under `/etc/systemd/system/`. To test this, run the following commands:

```
sudo ls -l /lib/systemd/system/vnc*
```

You should see:

```
-rw-r--r--. 1 root root 1744 Jun 10
16:15/lib/systemd/system/vncserver@.service
```

Then check under `/etc/systemd/system/`:

```
sudo ls -l /etc/systemd/system/*.wants/vnc*
```

This one doesn't exist:

```
ls: cannot access /etc/systemd/system/*.wants/vnc*: No such file or
directory
```

So, the first step is to start two new instances of VNC server for our two users. To do this, we will need to make two copies of the generic VNC service unit file under `/etc/system/system`. In the code snippet below, you're making two copies with two different names:

```
sudo cp /lib/systemd/system/vncserver@.service
/etc/systemd/system/vncserver@:4.service

sudo cp /lib/systemd/system/vncserver@.service
/etc/systemd/system/vncserver@:5.service
```

So why did we add two numbers (along with the colon) in the copied file names?

Again, that comes back to the concept of individual VNC services. VNC by itself runs on port 5900. Since each user will run their own VNC server, each user will have to connect via a separate port. The addition of a number in the file name tells VNC to run that service as a *sub-port* of `5900`. So in our case, **joevnc**'s VNC service will run on port 5904 (5900 + 4) and **janevnc**'s will run on 5905 (5900 + 5).
Next edit the service unit file for each client. Open the `/etc/systemd/system/vncserver@:4.service` file with the **vi** editor:

```
sudo vi /etc/systemd/system/vncserver@:4.service
```

A look at the “Quick HowTo” section tells us we have already completed the first step. Now we need to go through the remaining steps. The comments also tell us that VNC is a non-trusted connection. We will talk about this later.

For now, edit the `[Service]` section of the file, replacing instances of `<USER>` with `joevnc`. Also, add the `-geometry 1280x1024` clause at the end of the `ExecStart` parameter. This just tells VNC the screen size it should start in. You will modify two lines in total. Here’s what the edited file should look like (note that the entire file is not shown):

```
# The vncserver service unit file
#
# Quick HowTo:
# 1. Copy this file to /etc/systemd/system/vncserver@:<display>.service
# 2. Edit <USER> and vncserver parameters appropriately
#   ("runuser -l <USER> -c /usr/bin/vncserver %i -arg1 -arg2")
# 3. Run `systemctl daemon-reload`
# 4. Run `systemctl enable vncserver@:<display>.service`
#

. . .

[Unit]
Description=Remote desktop service (VNC)
After=syslog.target network.target

[Service]
Type=forking
# Clean any existing files in /tmp/.X11-unix environment
ExecStartPre=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 ||
:'
ExecStart=/sbin/runuser -l joevnc -c "/usr/bin/vncserver %i -geometry
1280x1024"
PIDFile=/home/joevnc/.vnc/%H%i.pid
ExecStop=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 || :'

[Install]
WantedBy=multi-user.target
```

Save the file and exit vi.

Similarly, open the `/etc/systemd/system/vncserver@:5.service` file in vi and make the changes for user **janevnc**:

```
sudo vi /etc/systemd/system/vncserver@:5.service
```

Here's just the `[Service]` section with the changes marked:

```
[Service]
Type=forking
# Clean any existing files in /tmp/.X11-unix environment
ExecStartPre=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 ||
:'
ExecStart=/sbin/runuser -l janevnc -c "/usr/bin/vncserver %i -geometry
1280x1024"
PIDFile=/home/janevnc/.vnc/%H%i.pid
ExecStop=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 || :'
```

Next, run the following commands to reload the systemd daemon and also to make sure VNC starts up for two users at boot time.

```
sudo systemctl daemon-reload
```

Enable the first server instance:

```
sudo systemctl enable vncserver@:4.service
```

Output:

```
ln -s '/etc/systemd/system/vncserver@:4.service'
'/etc/systemd/system/multi-user.target.wants/vncserver@:4.service'
```

Enable the second server instance:

```
sudo systemctl enable vncserver@:5.service
```

Output:

```
ln -s '/etc/systemd/system/vncserver@:5.service'
'/etc/systemd/system/multi-user.target.wants/vncserver@:5.service'
```

Now you've configured two VNC server instances.

Step 5: Configuring your firewall

Next, we will need to configure the firewall to allow VNC traffic through ports **5904** and **5905** only. CentOS 7 uses Dynamic Firewall through the firewalld daemon; the service doesn't need to restart for changes to take effect.

The firewalld service should start automatically at system boot time, but it's always good to check:

```
sudo firewall-cmd --state
```

This should show:

```
running
```

If the state is "not running" for any reason, execute the following command to make sure it's running:

```
sudo systemctl start firewalld
```

Now add the rules for ports 5904 and 5905:

```
sudo firewall-cmd --permanent --zone=public --add-port=5904-5905/tcp
```

Output:

```
success
```

Reload the firewall:

```
sudo firewall-cmd --reload
```

Output:

```
success
```

Step6

Setting VNC Password

We are one step away from seeing VNC in action. In this step, the users will need to set their **VNC passwords**. These are *not* the users' Linux passwords, but the passwords to log in to the VNC sessions.

Open another terminal connection to the CentOS 7 server, and this time log in as **joevnc**.

```
ssh joevnc@your_server_ip
```

Execute the following command:

```
vncserver
```

As shown in the output below, the server will ask **joevnc** to set up a VNC password. After typing in the password, the program also shows a number of files being created in the user's home directory:

```
you will require a password to access your desktops.

Password:
Verify:
xauth:  file /home/joevnc/.Xauthority does not exist

New 'localhost.localdomain:1 (joevnc)' desktop is localhost.localdomain:1

Creating default startup script /home/joevnc/.vnc/xstartup
Starting applications specified in /home/joevnc/.vnc/xstartup
Log file is /home/joevnc/.vnc/localhost.localdomain:1.log
```

Let's look at the line `New 'localhost.localdomain:1 (joevnc)' desktop is localhost.localdomain:1`. **localhost.localdomain** was the server name in our example; in your case it could be different. Note the number after the server name: (**1**, separated by a colon). It's not the number in **joevnc**'s service unit file (which was **4**). That's because this is the *display number* **joevnc**'s session will run on in this server, not the port number of the service (`5904`) itself.
Next open a new terminal session and log in as **janevnc**. Here as well, start the VNC server and set a password for **janevnc**:

```
vncserver
```

You should see similar output showing that **janevnc**'s session will run on display **2**. Finally, reload the services from the **main terminal session**:

```
sudo systemctl daemon-reload
sudo systemctl restart vncserver@:4.service
sudo systemctl restart vncserver@:5.service
```

Ex.No 3 B.

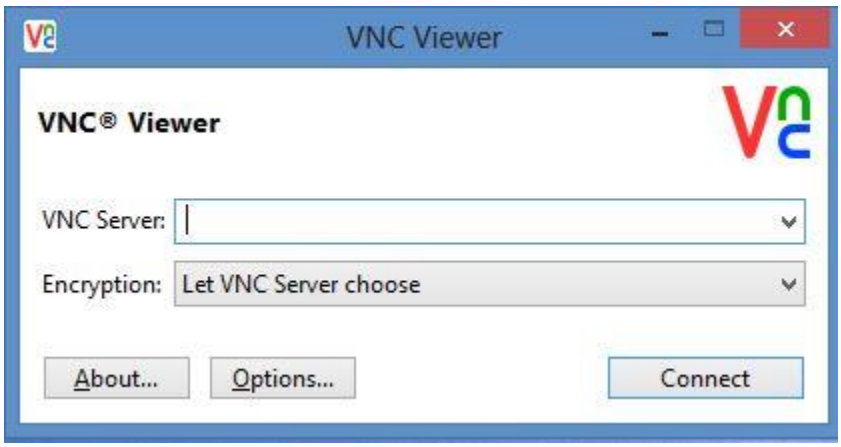
Desktop Virtualization using Chrome remote Desktop

Step1:

Connecting a Remote Desktops with a VNC client.

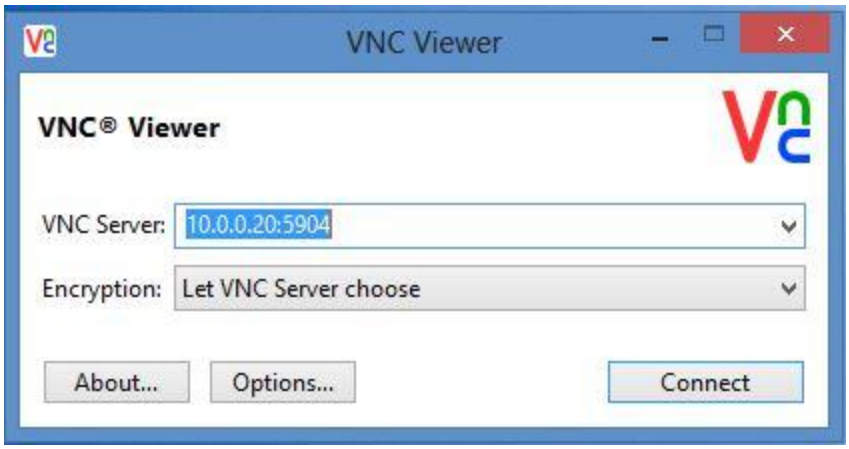
For this tutorial, we will assume users **joevnc** and **janevnc** are trying to connect to the CentOS 7 server from their Windows computers.
They will each need a VNC client for Windows to log into the remote desktop. This client is just like a terminal client like PuTTY, except it shows graphical output. There are various VNC client available, but the one we will use is RealVNC, available here. VNC Viewer for Mac OS X is available for download on the same page, and the Mac version is fairly similar to the Windows one.

When VNC Viewer is started, it shows a dialogue box like this:



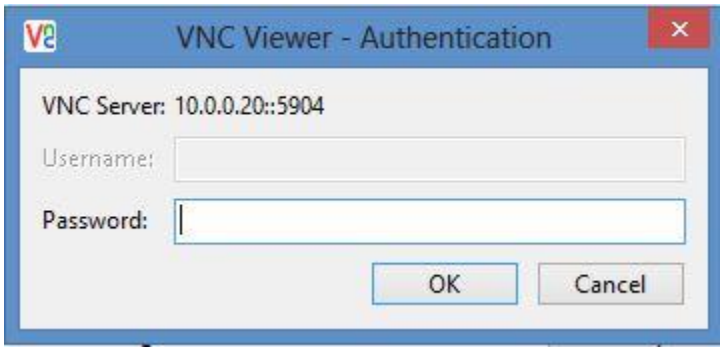
In the **VNC Server** field, add the IP address of your CentOS 7 server. Specify the port number 5904 after the server's IP, separate by a colon (**:**). We used 5904 because that's the VNC service port for **joevnc**.

We have also decided to let VNC Viewer choose the encryption method. This option will only encrypt the password sent across the network. Any subsequent communication with the server will be unencrypted. (We'll set up a secure SSH tunnel in the final step.)



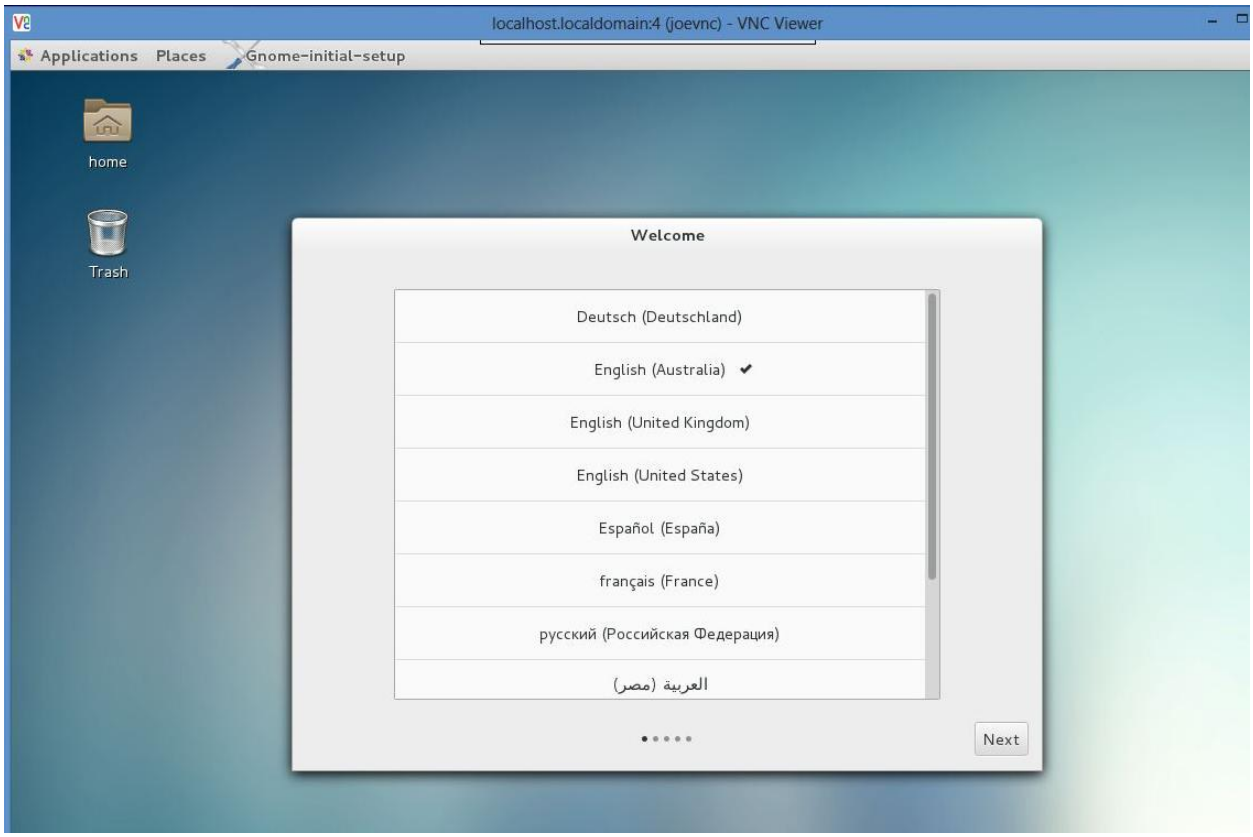
In fact, a warning message shows just that:

Accept the warning for now. A password prompt is displayed:



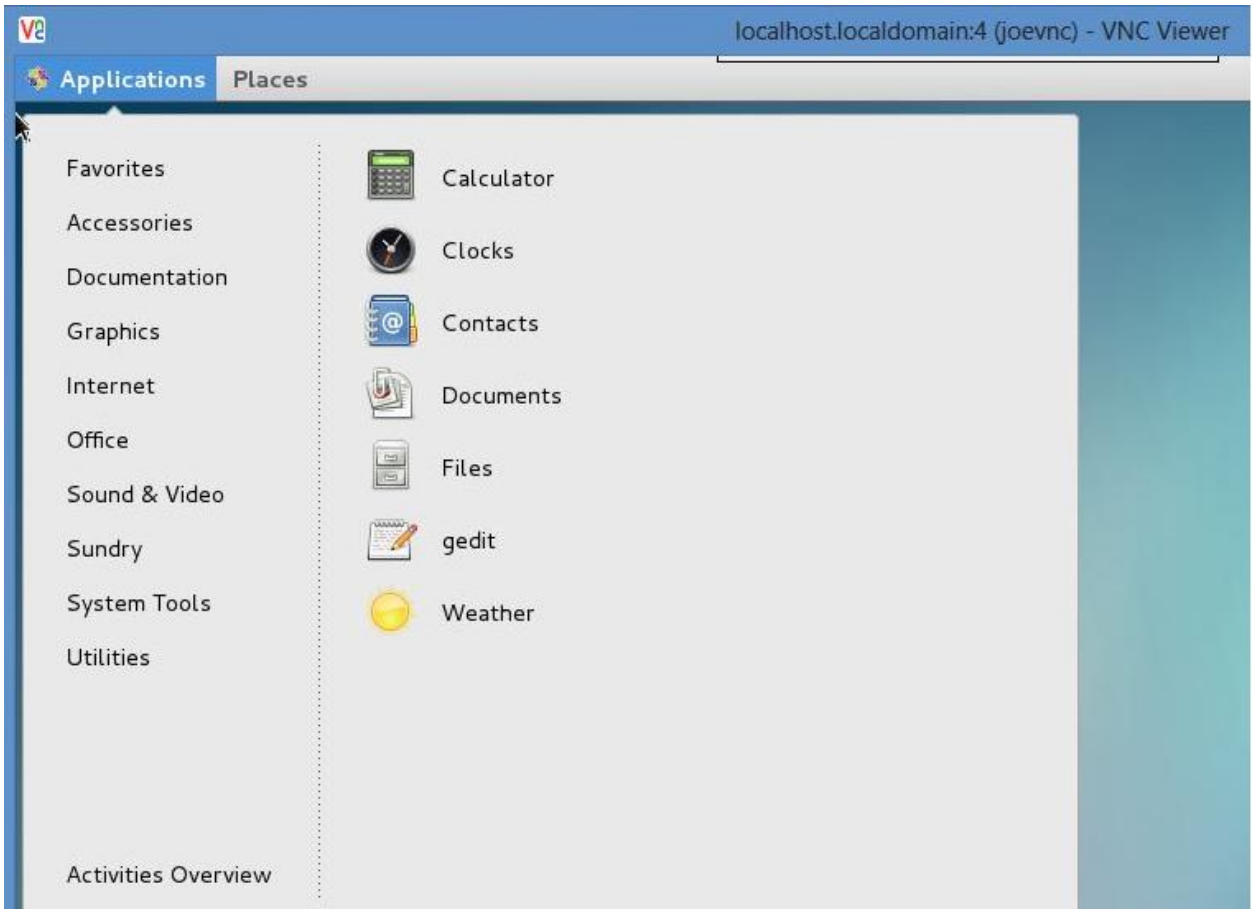
Enter **joevnc**'s VNC password that you set earlier.

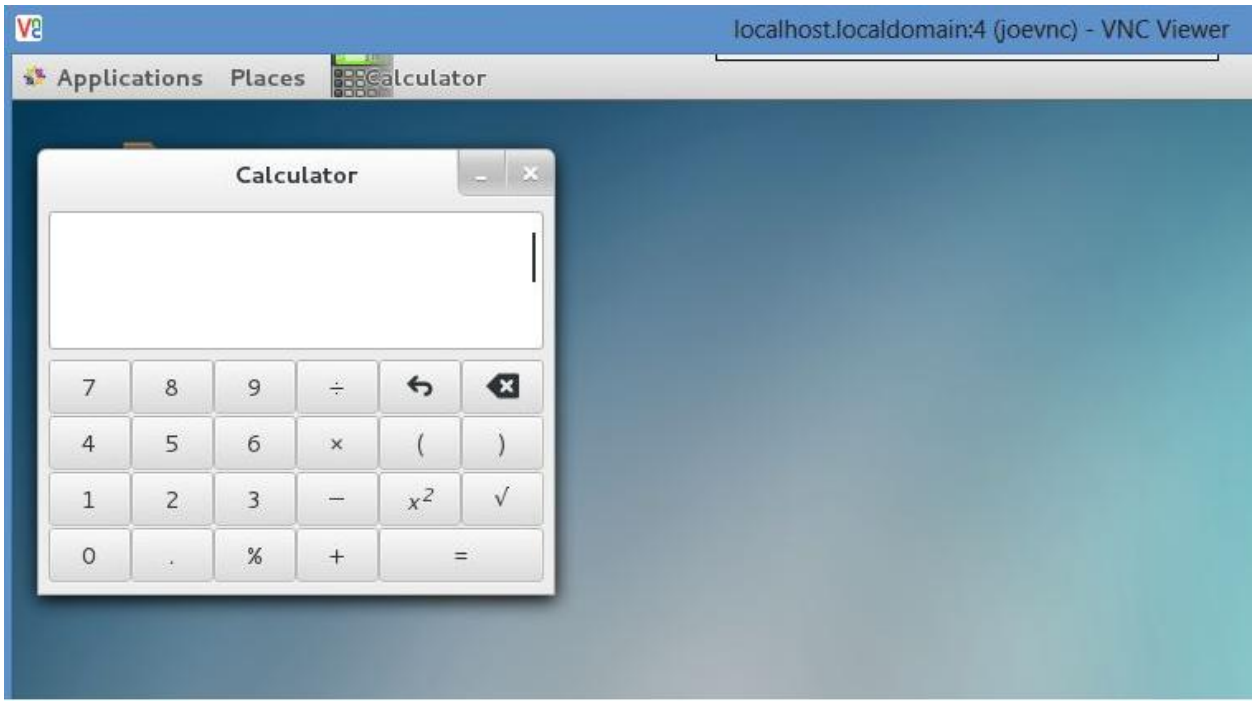
A new window opens showing the GNOME desktop for our remote CentOS server:



Accept the default welcome message.

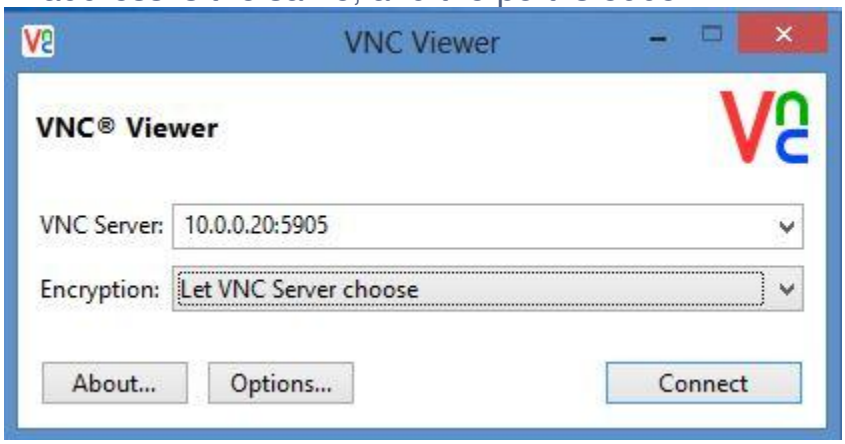
Now **joevnc** can start a graphical tool like the GNOME calculator:


localhost.localdomain:4 (joevnc) - VNC Viewer
Applications
Places
Favorites
Accessories
Documentation
Graphics
Internet
Office
Sound & Video
Sundry
System Tools
Utilities
Activities Overview
Calculator
Clocks
Contacts
Documents
Files
gedit
Weather


localhost.localdomain:4 (joevnc) - VNC Viewer
Applications
Places
Calculator
Calculator
7
8
9
÷
4
5
6
×
(
)
1
2
3
−
x²
√
0
.
%
+
=

You can leave this desktop connection open.

Now **janevnc** can also start another VNC session with the CentOS server. The IP address is the same, and the port is 5905:



When **janevnc** logs in via VNC Viewer, an empty desktop with a welcome message is shown, just like it was shown for **joevnc**. In other words, the two users are not sharing the desktop instances. **joevnc**'s desktop should still be showing the calculator.

To close the remote desktop session, simply closing the window will do. However, this doesn't stop the user's VNC service in the background on the server. If the service is not stopped or restarted and the machine had no reboots, the same desktop session would be presented at the next logon.

Close the VNC Viewer windows for **joevnc** and **janevnc**. Close their terminal sessions, too. From the main terminal window, check to see if the VNC services are still running:

```
sudo systemctl status vncserver@:4.service
```

The output shows that the remote desktop is still running:

```
vncserver@:4.service - Remote desktop service (VNC)
   Loaded: loaded (/etc/systemd/system/vncserver@:4.service; enabled)
   Active: active (running) since Sat 2014-11-01 12:06:49 EST; 58min ago
  Process: 2014 ExecStart=/sbin/runuser -l joevnc -c /usr/bin/vncserver %i
-geometry 1280x1024 (code=exited, status=0/SUCCESS)

. . .
```

Check the second service:

```
sudo systemctl status vncserver@:5.service
```

This one is running, too:

```
vncserver@:5.service - Remote desktop service (VNC)
   Loaded: loaded (/etc/systemd/system/vncserver@:5.service; enabled)
   Active: active (running) since Sat 2014-11-01 12:42:56 EST; 22min ago
  Process: 3748 ExecStart=/sbin/runuser -l janevnc -c /usr/bin/vncserver
%i -geometry 1280x1024 (code=exited, status=0/SUCCESS)


. . .
```

If you wanted to log back into **joevnc**'s desktop at this point, you'd see the same calculator app open.

This presents some interesting challenges for system administrators. If you have a number of users connecting to the server via VNC, you may want to devise some way to stop their VNC services when no longer needed. This may save some valuable system resources.

Troubleshooting — VNC Service Crashes

As you test and play around with VNC, you may sometimes find the service has crashed and is unrecoverable. When you try to check the status:

```
sudo systemctl status vncserver@:4.service
```

This long error message may come up:

```
vncserver@:4.service - Remote desktop service (VNC)
   Loaded: loaded (/etc/systemd/system/vncserver@:4.service;
enabled)
   Active: failed (Result: exit-code) since Fri 2014-11-07 00:02:38
EST; 2min 20s ago
  Process: 2221 ExecStart=/sbin/runuser -l joevnc -c
/usr/bin/vncserver %i -geometry 1280x1024 (code=exited, status=2)
  Process: 1257 ExecStartPre=/bin/sh -c /usr/bin/vncserver -kill %i
> /dev/null 2>&1 || : (code=exited, status=0/SUCCESS)
```

Trying to start the service doesn't work:

```
sudo systemctl start vncserver@:4.service
```

Failed startup:

```
Job for vncserver@:4.service failed. See 'systemctl status
vncserver@:4.service' and 'journalctl -xn' for details.
```

Usually the reason is simple enough. Check `/var/log/messages`:

```
sudo tail  /var/log/messages
```

The related error will look like this:

```
Nov  7 00:08:36 localhost runuser: Warning: localhost.localdomain:4
is taken because of /tmp/.X11-unix/X4
Nov  7 00:08:36 localhost runuser: Remove this file if there is no
X server localhost.localdomain:4
Nov  7 00:08:36 localhost runuser: A VNC server is already running
as :4
Nov  7 00:08:36 localhost systemd: vncserver@:4.service: control
process exited, code=exited status=2
Nov  7 00:08:36 localhost systemd: Failed to start Remote desktop
service (VNC).
Nov  7 00:08:36 localhost systemd: Unit vncserver@:4.service
entered failed state.
Nov  7 00:08:36 localhost systemd: Failed to mark scope session-
c3.scope as abandoned : Stale file handle
```

The remedy is to delete the file under /tmp folder:

```
sudo rm -i /tmp/.X11-unix/X4
```

Output:

```
rm: remove socket '/tmp/.X11-unix/X4'? y
```

Then start the VNC service:

```
sudo systemctl start vncserver@:4.service
```

General Troubleshooting

Although relatively rare, you may encounter other errors when working with VNC. For example, your remote desktop screen can go blank or hang, the session might crash with a cryptic error message, VNC Viewer may not connect properly or transmit commands to the GUI to launch applications, etc.

We recommend checking the `/var/log/messages` file to get a better understanding. At times you may need to reboot your server, or in extreme cases recreate the VNC service.

System resources can also be a culprit; you may have to add extra RAM to your Droplet, etc.

Step 2:

Securing VNC Sessions Through SSH Tunneling

So far both **joevnc** and **janevnc** have been accessing their remote desktops through unencrypted channels. As we saw before, VNC Viewer warns us about this at connection time; only the password is encrypted as the sessions begins. Any subsequent network traffic and data transfer is open for anyone to intercept in the middle.

About SSH Tunnelling

This is where Secure Shell (SSH) sessions can help. With SSH, VNC can run within the context of an SSH encrypted session. This is known as *tunnelling*. In effect, VNC traffic piggybacks on the SSH protocol, resulting in all of its communication with the server being encrypted. It's called *tunnelling* because SSH is providing wraparound protection over VNC and VNC is running as if in a tunnel within SSH. SSH tunnelling can be used for other protocols like POP, X, or IMAP as well.

SSH tunnelling works with *port forwarding* which is basically a means of translating access from one particular port to a different port on another machine. With port forwarding, when a client application connects to Port A running on machine A, it's transparently forwarded to port B running on machine B. The client application is unaware of this translation and thinks it's connecting to the original port. Port forwarding is one of the features of SSH protocol.

With port forwarding, we can set our local VNC client to connect to port **5900** on the local client computer, and this can be mapped to port **5905** on the remote server. This is example is for **janevnc**'s connection, but you could easily follow the same steps for any other clients.

When the VNC client application starts, it can be pointed to port **5900** on **localhost**, and our port forwarding will transparently transport it to port **5905** on the remote server.

Note: You'll have to start an SSH section **each time** to make the connection secure.

OS X

On your Mac, open **Terminal**.

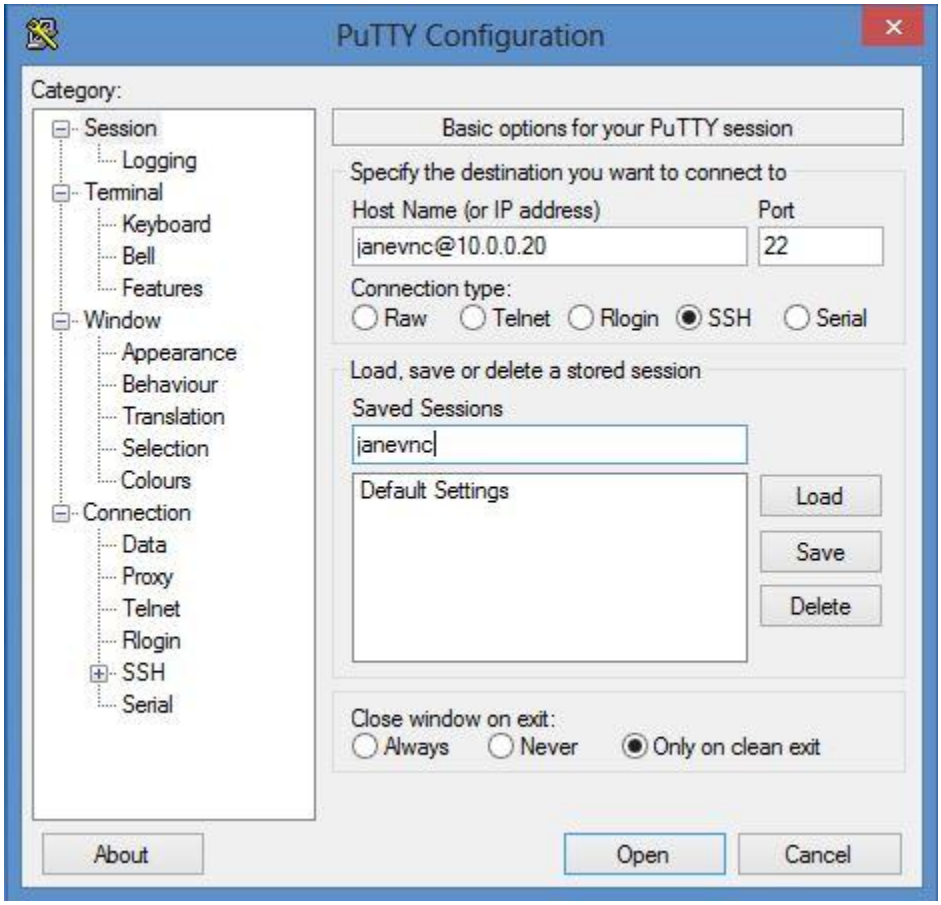
Enter the following connection information, being sure to replace `your_server_ip` with your remote server's IP address:

```
ssh -L 5900:your_server_ip:5905 janevnc@your_server_ip -N
```

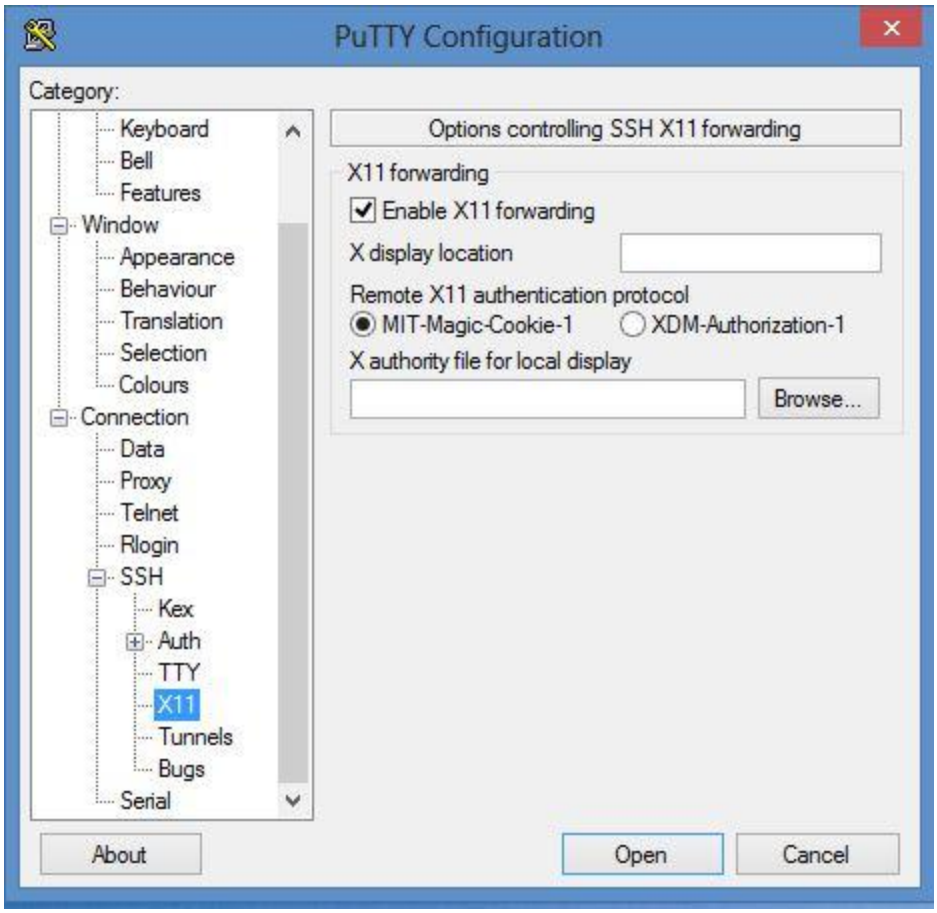
Enter **janevnc**'s UNIX password. The connection will appear to hang; you can keep it running for as long as you use the remote desktop.

Now skip ahead to the VNC Viewer instructions.

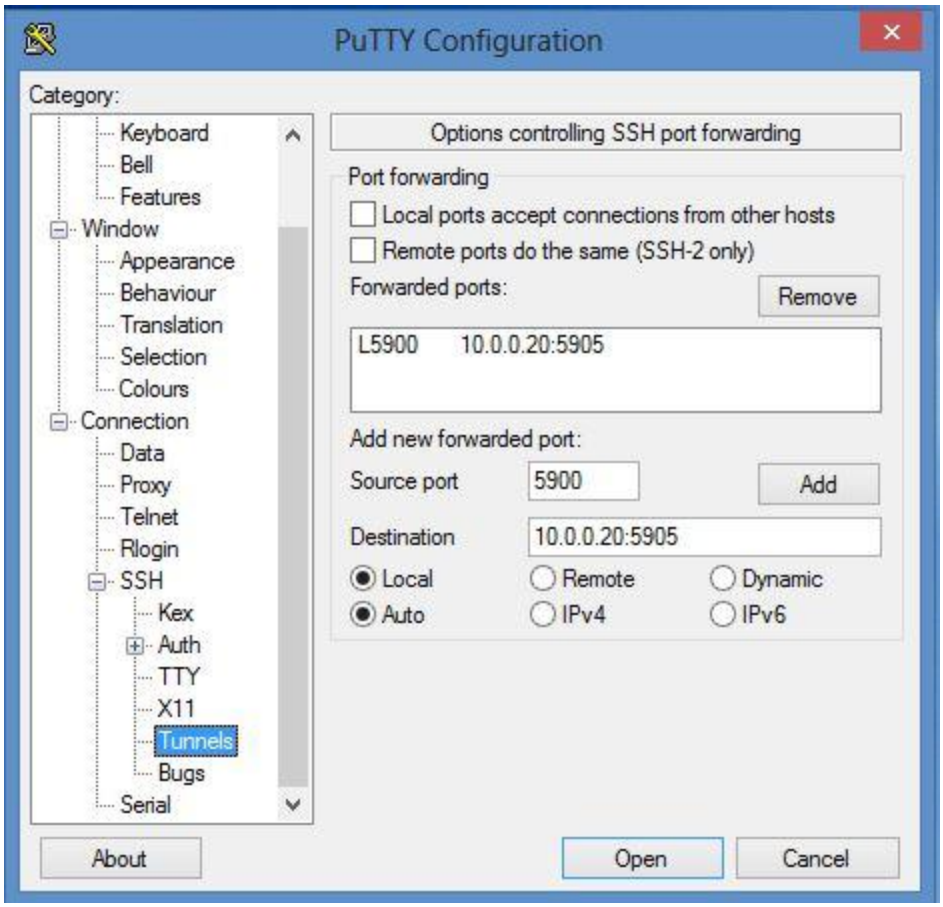
Windows
For securing **janevnc**'s VNC session, we will assume the local Windows computer has PuTTY installed. PuTTY is free and can be downloaded from here.
If **janevnc**'s VNC and terminal sessions are not closed already, close them now.
Start PuTTY. In the session screen, ensure you specify the server IP address and give a descriptive name to the connection, then click the **Save** button to save the connection details. Note how we have specified `username@your_server_ip` in the **Hostname** field:



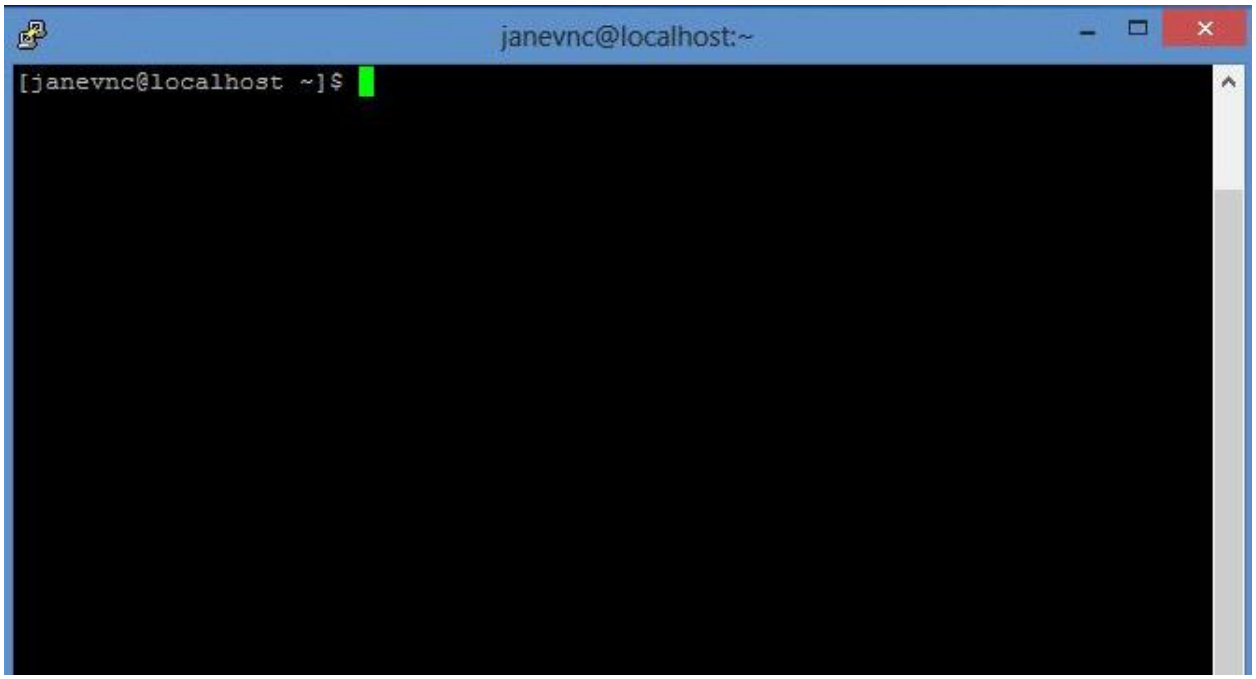
Next, expand the **SSH** menu item in the left navigation pane, and select the **X11** item. This shows the X11 forwarding properties for the session. Ensure the checkbox for **Enable X11 forwarding** is checked. This ensures that SSH encrypts X Windows traffic that flows between the server and client:



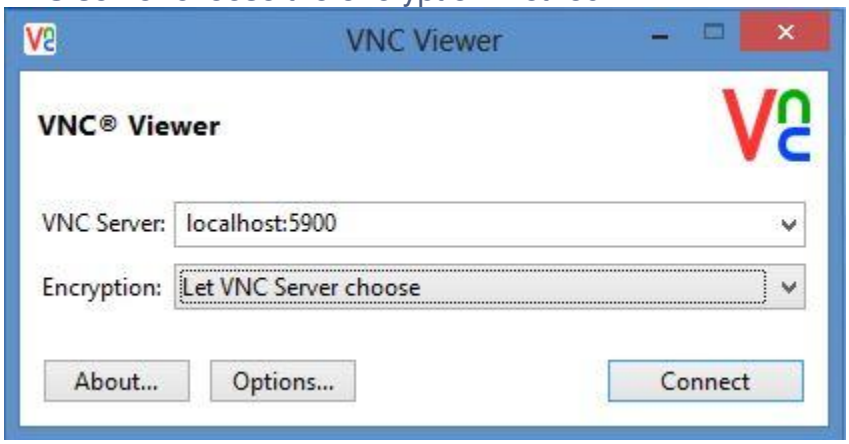
Finally, select **SSH > Tunnels**. Type `5900` in the **Source** port field. In the **Destination** field, specify your server's name or IP address, followed by a colon and the VNC port number for the intended user. In our case, we have specified `your_server_ip:5905`.
Alternately, you could use port **5902**. The **2** in this case would be the display number for **janevnc** (remember the message displayed when **janevnc** ran the `vncserver` command).
Click the **Add** button and the mapping will be added under **Forwarded ports**. This is where we are adding port forwarding for the SSH session; when the user connects to localhost at port **5900**, the connection will be automatically tunnelled through SSH to the remote server's port **5905**.



Go back to the **Sessions** items and save the session for **janevnc**. Click the **Open** button and a new terminal session will open for **janevnc**. Log in as **janevnc** with the appropriate UNIX password:



VNC Viewer
Next start VNC Viewer again. This time, in the **VNC Server** address, type <^> and let VNC server choose the encryption method:

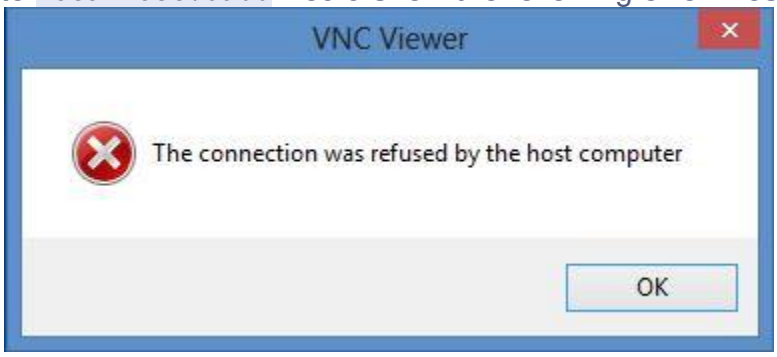


Click the **Connect** button.

You will still get the dialogue box warning you about an unencrypted session, but this time you can safely ignore it. VNC Viewer doesn't know about the port it's being forwarded to (this was set in the SSH session just started) and assumes you are trying to connect to the local machine.

Accepting this warning will show the familiar password prompt. Enter **janevnc**'s VNC password to access the remote desktop.
So how do you know the session was encrypted? If you think about it, we had set port forwarding in the SSH session. If an SSH session wasn't established, port forwarding wouldn't have worked. In fact, if you close the terminal window and log out of the PuTTY

session then try to connect with VNC Viewer alone, a connection attempt to `localhost:5900` would show the following error message:



So, if the `localhost:5900` connection works, you can be confident that the connection is encrypted.

Remember that you will want to establish the SSH connection first every time you use VNC, to make sure your connection is always encrypted.

Conclusion

Accessing your CentOS Linux system from a GUI front end can make system administration much simpler. You can connect from any client operating system and don't have to depend on web-based hosting control panels. VNC has a much smaller footprint compared to most control panels.

Although we have shown how two ordinary users can connect with their VNC clients, that's hardly practical in serious production environments. In reality, users will have customized applications or browsers for accessing the server. Running a number of VNC services for each user also creates an unnecessary burden on system resources, not to mention the inherent risks associated with it.

If you decide to install and run VNC on your production Linux server, we strongly recommend using it for administrative purposes only.

OR

3 A & B

Important Terminology

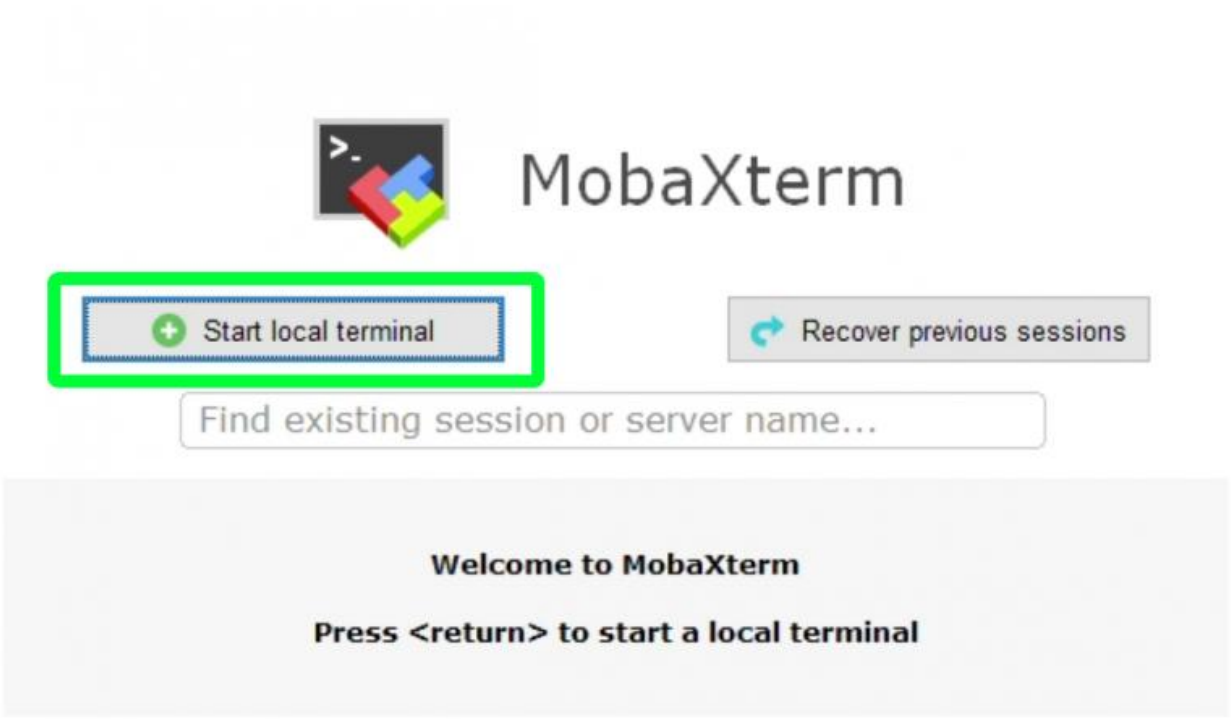
Here are some key terms to remember when connecting with VNC

- **VNC session password** – this password is used only with VNC. This is *not tied to your MCECS login*. Because the encryption on this password is extremely weak, do not use a password that you want to keep private as your session password.
- **Display Number** – when you start a VNC session, it will be assigned a number between 1 and 99 that will identify it on the Linux system you are remotely connected to.
- **Port Number** – This is equal to your *Display Number plus 5900*. This number is used by the VNC viewer software to remotely connect with the VNC session running on MCECS Linux systems.
- **Host Address** – this is the full domain name of the system you want to remotely connect with. This is usually in the form of somecomputer.cs.pdx.edu, somecomputer.ece.pdx.edu, or somecomputer.cecs.pdx.edu

Starting the VNC session

Step 1 – Opening a terminal

Open MobaXterm and click on the *Start local terminal* button, as highlighted in the image below.



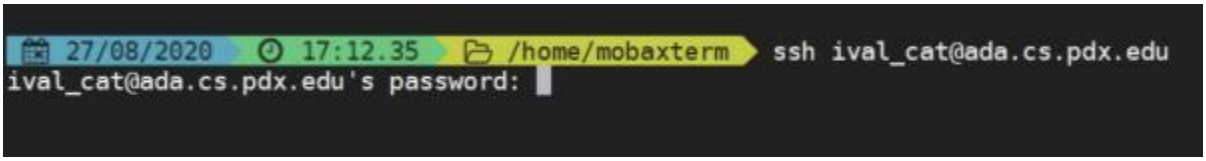
Step 2 – Connecting to Linux system with SSH

In the command line, enter the following command

```
ssh your_username@host_name
```

Replace *your_username* with your **MCECS username** and replace *host_name* with the **address of the MCECS Linux machine or server** you want to connect with (for example, mo.ece.pdx.edu or rita.cecs.pdx.edu).

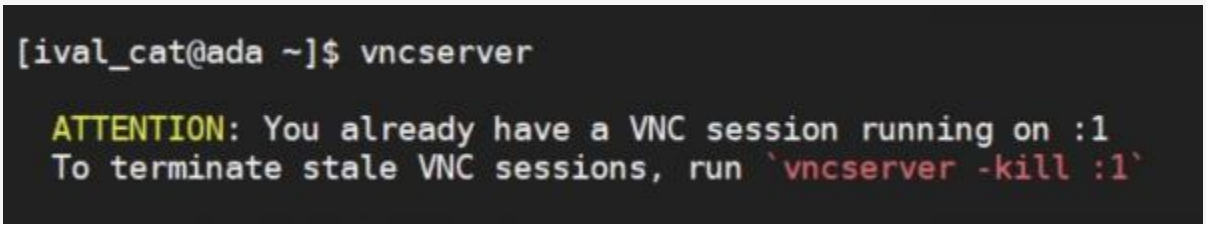
Enter your MCECS account password when prompted, and log in to the host system.



Step 3 – Starting VNC session process

Start a VNC session by entering the command **vncserver**

NOTE: If you see the following message after entering **vncserver**, this means you have a VNC session already running on this system. Go to the end of this article for more information on how to check for existing VNC sessions and also how to terminate them.



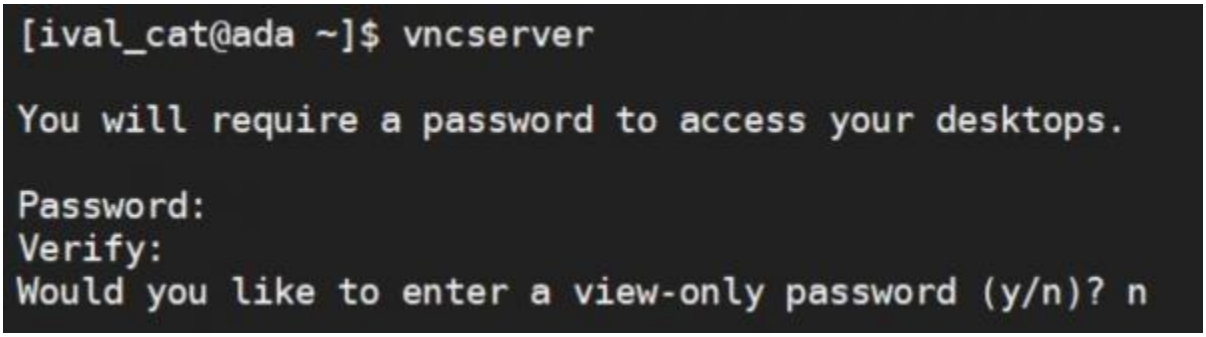
Step 4 – Creating VNC session password

You should now see a prompt to enter a password like in the image below. This will be your **VNC session password**.

Be aware of the following:

- The session password needs to be **at least 6 characters long**.
- This password is only used to log in to your VNC session and is **not tied to your MCECS account password**.
- This password is stored with very poor encryption, so it is advised that you **do not use a sensitive password for your VNC session password**.

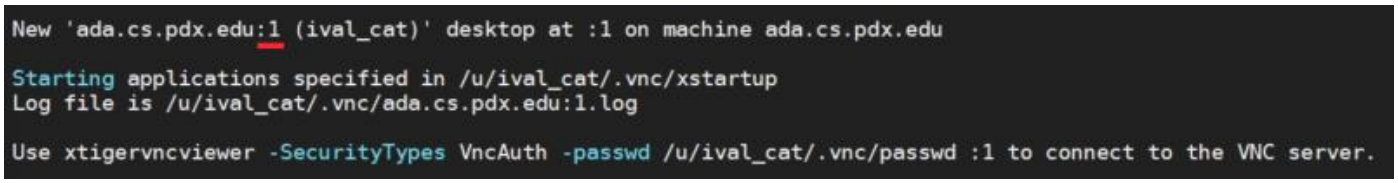
You will also be prompted to enter a **view-only password**, which can be used by other people to observe your VNC session. If you are unsure about this feature, enter **n** for "no" and avoid creating one.



NOTE: It is possible you may not see a password prompt. If you have previously used VNC, the new process will sometimes use your previous session password. If you have forgotten your previous session password, run the command vncpasswd to change it.

Step 5 – Getting Display/Port numbers

Your VNC session has been created, and you should see a message similar to the sample output below



The number that appears after the host address is the **display number** (it is underlined in red in the image above). By adding this number to 5900, this will give you the **port number** used to connect your VNC viewer to the VNC session.

For example, if your display number is 4, your port number is 5904. If your display number is 12, your port number is 5912.

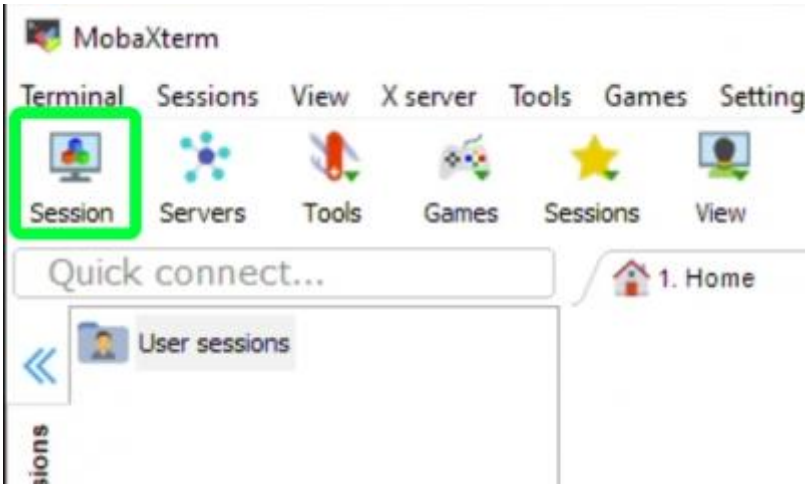
NOTE: Your display number **may not be the same as the sample image above**. Make sure to read the output message in your terminal and look for the number after the semicolon for your true display number.

The VNC session is now running on the remote Linux host system and is ready to connect with your VNC viewing software. You can exit and close this terminal if you want, as the VNC session will continue to run in the background. Be aware that the CAT will kill any VNC session that has been idle for more than 48 hours.

Connecting with your VNC session with MobaXterm's VNC viewer

Step 1 – Opening new VNC viewer session

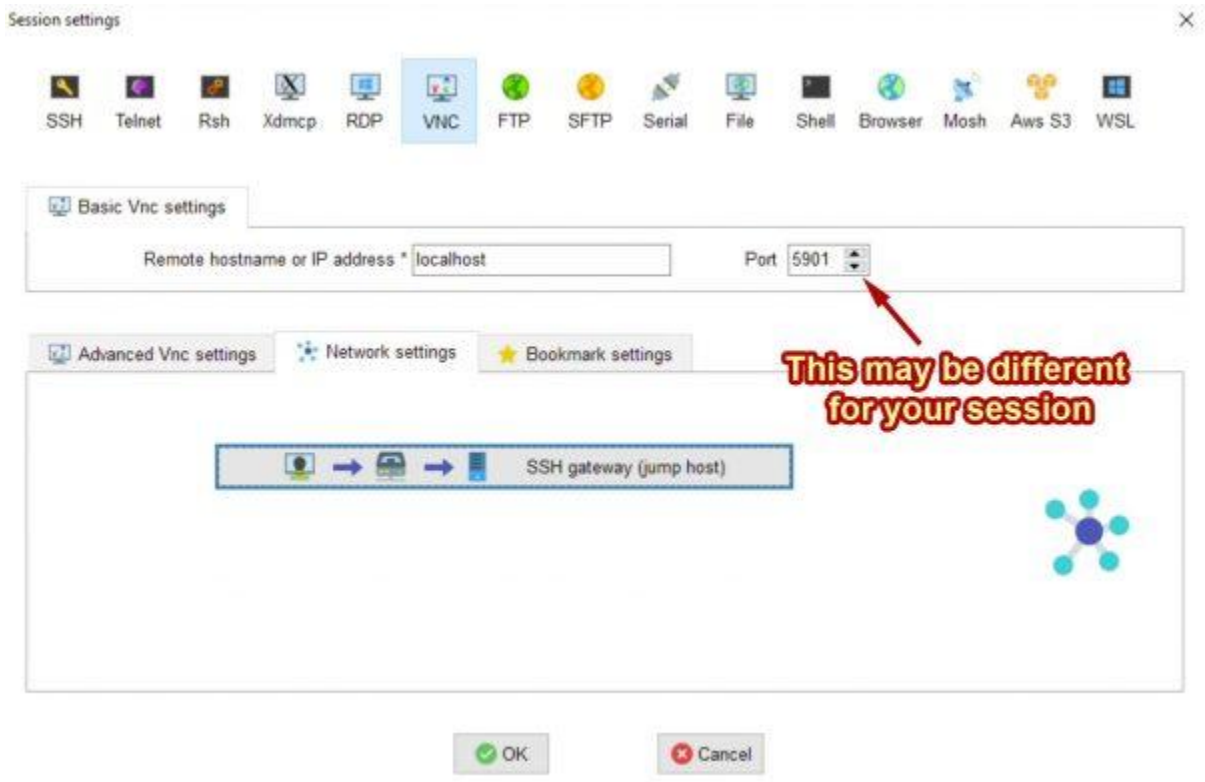
In MobaXterm, click on the Session button in the upper left hand corner



Step 2 – VNC viewer setup

In the window that pops up, look for the VNC icon in the top row and click on it

- In the *Remote hostname or IP address* box, enter **localhost**
- In the *Port* box, enter your **Port Number**. Recall that this is 5900 plus the Display Number that appeared after running the vncserver command



Step 3 – SSH Gateway setup

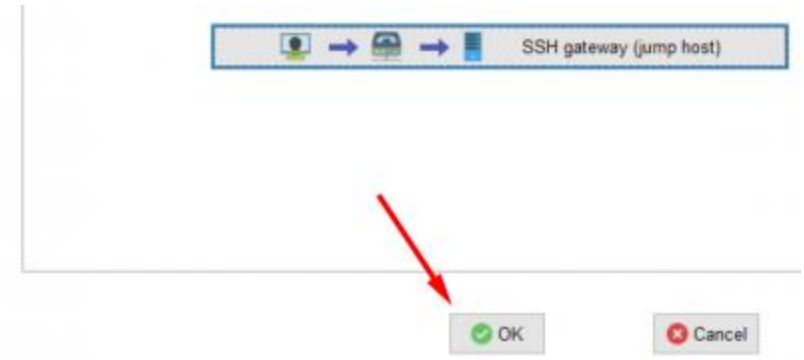
In the lower area, click on the *Network Settings* tab, and then click on the *SSH gateway (jumphost)* button. The button is highlighted in the blue box in the image above.

In the window that pops up, enter the following

- In the *Gateway host* box, enter the **address of the host machine** that your VNC session is running on (e.g. ada.cs.pdx.edu, mo.ece.pdx.edu, etc).
- In the *Username* box, enter your **MCECS username**.
- In *Port*, leave it set at **22**
- **Do not** check off the box for *Use SSH key*.

Afterwards, click the *OK* button with the green checkmark to save these settings and close this configuration window.

When you return to the previous menu, click the *OK* button again and connect MobaXterm's VNC Viewer with the remote VNC session

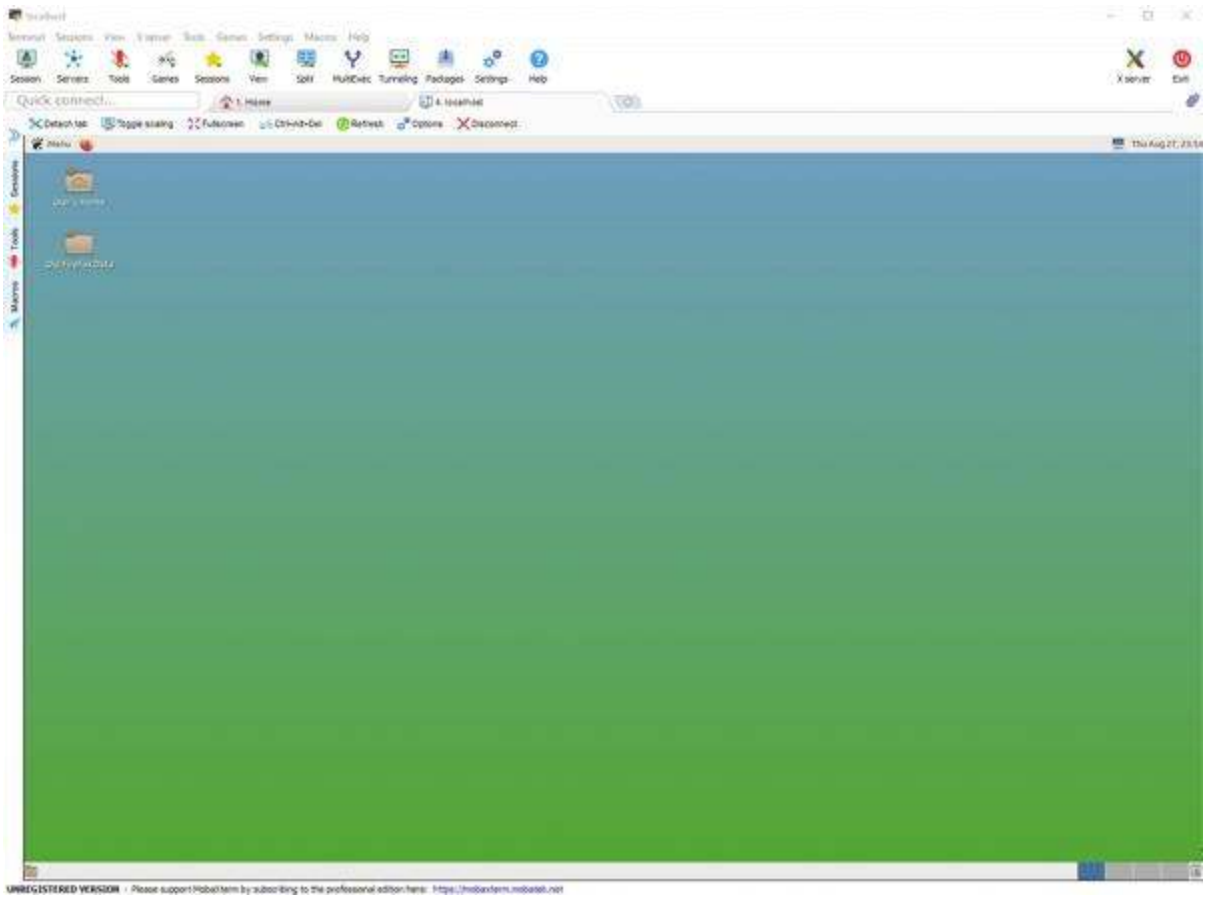


Step 4 – VNC password guide

When you see the following window asking for the *password for MCECS username on the host address*, enter your **MCECS login password**. This window may or may not appear, depending on how recently you used MobaXterm to view a VNC session previously.

When you see the following window asking for the *password for localhost*, enter your **VNC session password**.

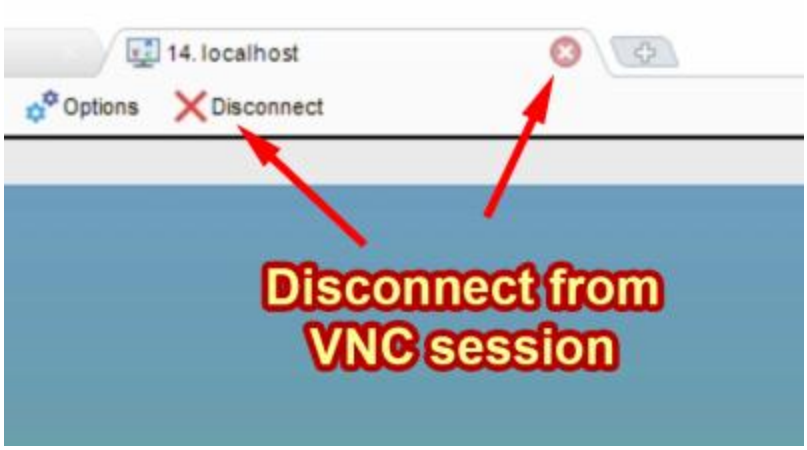
A new tab should now appear in MobaXterm with a Linux graphical interface. Congratulations! You are now remotely connected with a Linux system via VNC.



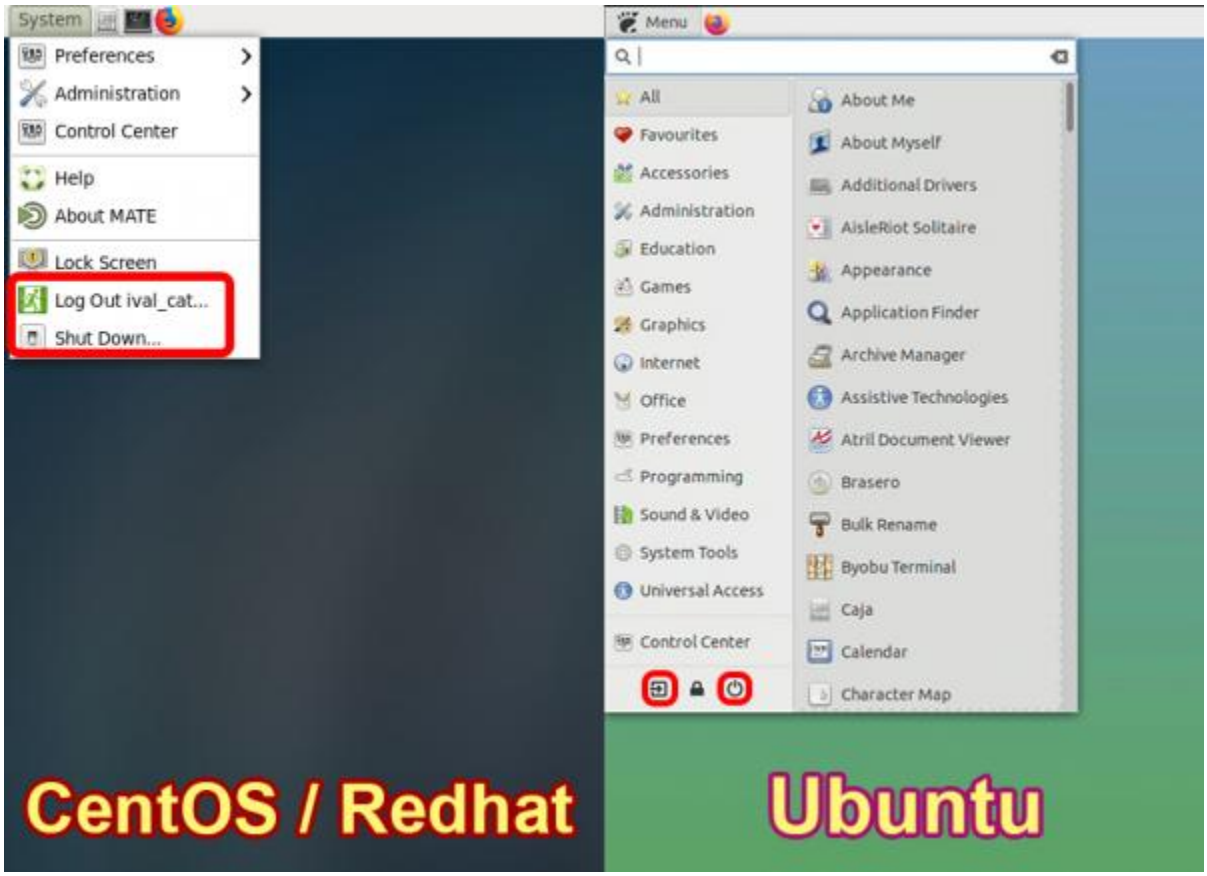
Disconnecting vs. Terminating your VNC session

It is possible to disconnect from your VNC session and reconnect with it later on to pick up where you left off. In MobaXterm, if you close the tab or click on the Disconnect button, your VNC session will not end and will continue to run on the host system. To reconnect with your session, simply follow the instructions above for **Connecting with your VNC session with MobaXterm's VNC viewer** using the same session password and port number as before**.**

Be aware that on CAT-supported systems, VNC sessions are terminated if they have been idle for more than 48 hours.



If you want to kill the VNC session, you can use the *Log Off* or *Shutdown* option in the Linux graphical interface. The location of these options will vary depending on the version of Linux on the host system and your personal settings.



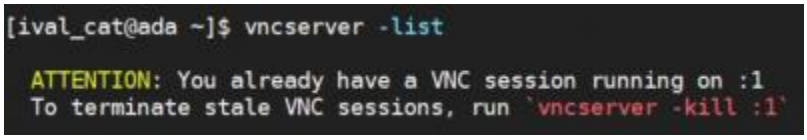
Alternatively, you can kill VNC sessions using the command **vncserver -kill :X**, where X is replaced with your session's Display Number.

```
[ival_cat@ada ~]$ vncserver -kill :1
Killing Xtigervnc process ID 22671... success!
```

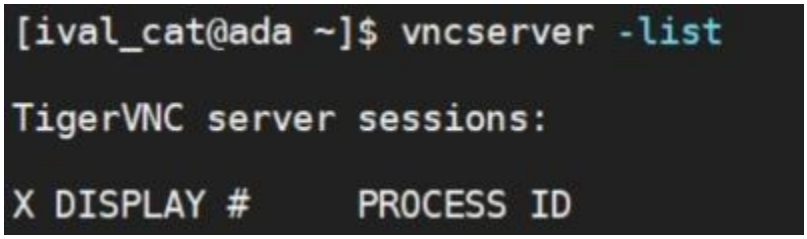
Checking for existing VNC sessions

If you want to check for existing VNC sessions or find its display number, run the command **vncserver -list**

If there is an existing session, you will see the following output

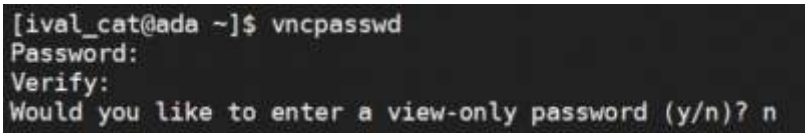


If there are no sessions running, you will see the following output



Changing your VNC password

If you want to change your VNC session password, run the command **vncpasswd** and follow the prompts. The session password can be changed even if you have VNC currently running, allowing you to use the new password even after starting a session.



Ex No:4

VMware ESXi 6.5 **01 Dec**
Network Server **2016**
VMware, Inc. **145109**

YES CERTIFIED with the following products:

Virtual Machines (Guests OS):

2 SUSE® Linux Enterprise Server 11 for AMD64 & Intel64
SP: Service Pack 4 for SUSE SLES 11 Arch: 64 Mode: Fully Virtualized

2 SUSE® Linux Enterprise Server 11 for AMD64 & Intel64
SP: Service Pack 4 for SUSE SLES 11 Arch: 32pae Mode: Fully Virtualized

Product Description

VMware ESXi is a complete, scalable and robust virtualization platform. Designed to reduce capital and operation costs. Maximize IT efficiency while giving you agility through automation and the freedom to choose applications, SUSE® Linux Enterprise OS and hardware.

Tested Configuration:

Computer Type:	Virtual Machine
Mother Board Revision:	Intel 440 BX Desktop Reference Platform / Motherboard Rev B0
BIOS/uEFI:	BIOS: 6.00 (04/05/2016)
CPU:	8 Intel Xeon® Processor E7-8890 v2 2.80 GHz
RAM:	Host platform: 6 TB, multiple 64-bit VM guests: 3000 GB, 32-bit guests: 16 GB, single max VM guest: 6128 GB
Ports and Bus Types:	Serial Parallel Port 3 PCI-ISA 4 32-Bit PCI

	PCI Express X8
Video Adapter:	VMware® VMware SVGA II
Host Bus Adapter:	VMware® Paravirtual SCSI (PVSCSI) adapter , SCSI VMware® Virtual IDE Device for SUSE Linux , IDE
Hard Disk Drive:	VMware® Virtual Hard Disk rev:1.0 (SCSI) , SCSI
CD/DVD:	VMware® Virtual IDE CDR10 , IDE
Test Kit:	System Certification Kit 7.6.0-44.1

Config Notes

1. VMware recommends to install deployPkg with open-vm-tools if creating a Template http://kb.vmware.com/kb/2075048.
2. Virtual machine guest: System certification testing was performed with a virtual machine configured with up to 6128 GB of memory on a SLES 11 SP4 x86-64 guest.
3. Virtual machine guest: System certification testing was performed with a SLES 11 SP4 64-bit virtual machine configured with up to 128 CPUs.
4. VMware recommends using the Open VM Tools redistributed by the operating system vendors. For additional information, see knowledge base article 2073803 at http://kb.vmware.com/kb/2073803. VMware guest OS installation information can be found at:http://partnerweb.vmware.com/GOSIG/SLE_11.html
5. The VMware Memory Ballooning driver included in SLES 11 SP4 is auto loaded to improve the virtual machine memory performance.

Adapters and Drivers

VMware® VMware SVGA II

Driver Type: Video Driver	Driver Name: vmware_drv.so
Driver Date: 17-Jun-	Driver Size: 5716

2015

Driver Version: 11.0.3

Driver Type: Video Driver	Driver Name: vmware_drv.so
Driver Date: 17-Jun-2015	Driver Size: 10560

Driver Version: 11.0.3

VMware® VMXNET3 Ethernet Adapter

Driver Type: LAN Driver	Driver Name: vmxnet3.ko
Driver Date: 24-Jun-2015	Driver Size: 77407

Driver Version: 1.1.30.0-k

Driver Type: LAN Driver	Driver Name: vmxnet3.ko
Driver Date: 24-Jun-2015	Driver Size: 58587

Driver Version: 1.1.30.0-k

VMware® Paravirtual SCSI (PVSCSI) adapter

Driver Type: HBA Driver	Driver Name: vmw_pvscsi.ko
Driver Date: 24-Jun-2015	Driver Size: 39527

Driver Version: 1.0.1.0-k

Driver Type: HBA Driver	Driver Name: vmw_pvscsi.ko
Driver Date: 24-Jun-2015	Driver Size: 28699

Driver Version: 1.0.1.0-k

VMware® Virtual IDE Device for SUSE Linux

Driver Type: HBA Driver	Driver Name: ata_piix.ko
Driver Date: 24-Jun-2015	Driver Size: 51119

Driver Version: 2.13

Driver Type: HBA	Driver Name: ata_generic.ko

Driver	
Driver Date: 24-Jun-2015	Driver Size: 13383
Driver Version: 0.2.15	
Driver Type: HBA Driver	Driver Name: ata_generic.ko
Driver Date: 24-Jun-2015	Driver Size: 10087
Driver Version: 0.2.15	
Driver Type: HBA Driver	Driver Name: ata_piix.ko
Driver Date: 24-Jun-2015	Driver Size: 39751
Driver Version: 2.13	

The term *YES CERTIFIED* applies only to the exact configuration documented in this bulletin. For more information on hardware exchange policies, please access the following document and view the Hardware Component Exchange Guide.

Supported virtual installations and virtualization products

You can install
Symantec Endpoint Protection
 on the supported operating systems that run in virtual environments. Install
Symantec Endpoint Protection
 on the guest operating system, and not the host.
The following virtualization products support the
Symantec Endpoint Protection Manager
, console, and
Symantec Endpoint Protection
 client software for Windows and Linux:

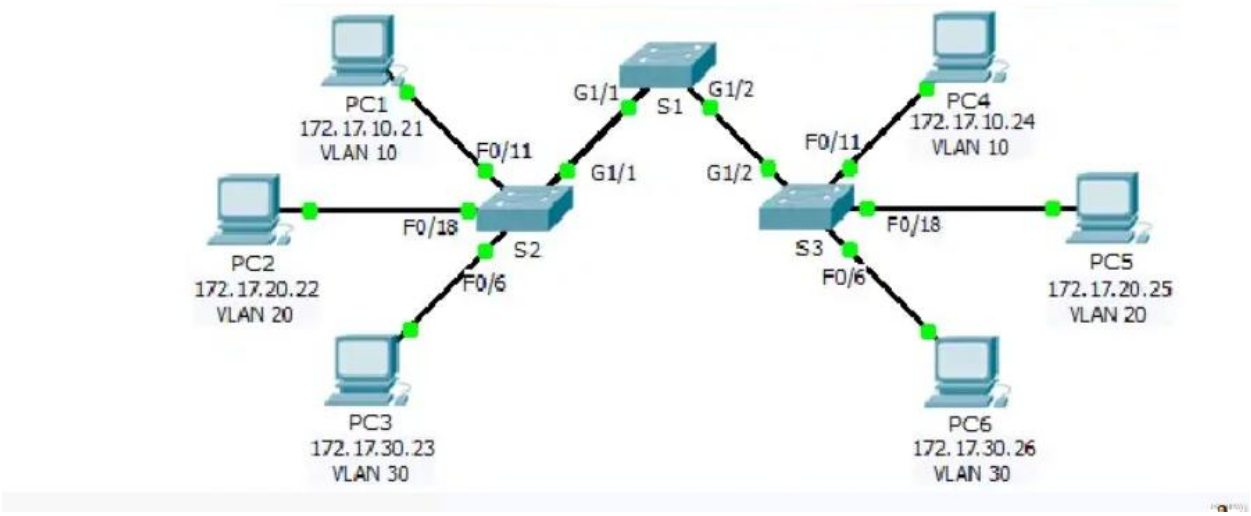
- Microsoft Azure
- Amazon Web Service (AWS) EC2
- Amazon WorkSpaces

- Citrix Studio Version 2009.0.0
- Nutanix AOS 5.15 (LTS)
- Oracle Cloud Infrastructure (OCI)
- VMware WS 5.0 (workstation) or later
- VMware GSX 3.2 (enterprise) or later
- VMware ESX 2.5 (workstation) or later
- VMware ESXi 4.1 - 5.5
- VMware ESXi 6.0
- VMware ESXi 6.0 Update 1
- VMware ESXi 6.0 Update 2
- VMware ESXi 6.0 Update 3 (As of 14.0.1)
- VMware ESXi 6.5 (As of 14.0.1)
- VMware ESXi 6.5U1 (As of 14.2)
- VMware ESXi 6.5U2 (As of 14.2)
- VMware ESXi 6.7 (As of 14.2)
- VMware ESXi 7.0 Update 2 (As of 14.3 RU2)
- Microsoft Virtual Server 2005
- Windows Server 2008 Hyper-V
- Windows Server 2008 R2 Hyper-V
- Windows Server 2012 Hyper-V
- Windows Server 2012 R2 Hyper-V
- Windows Server 2016 Hyper-V (As of 14.2 MP1)
- Windows Server 2019 Hyper-V Core Edition (As of 14.2 MP1)
- Citrix XenServer 5.6 or later
- Virtual Box, supplied by Oracle

Symantec Endpoint Protection
includes many features that enhance performance in virtual environments.

Packet Tracer – Configuring VLANs

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	VLAN
PC1	NIC	172.17.10.21	255.255.255.0	10
PC2	NIC	172.17.20.22	255.255.255.0	20
PC3	NIC	172.17.30.23	255.255.255.0	30
PC4	NIC	172.17.10.24	255.255.255.0	10
PC5	NIC	172.17.20.25	255.255.255.0	20
PC6	NIC	172.17.30.26	255.255.255.0	30

Objectives

Part 1: Verify the Default VLAN Configuration

Part 2: Configure VLANs

Part 3: Assign VLANs to Ports

Background

VLANs are helpful in the administration of logical groups, allowing members of a group to be easily moved, changed, or added. This activity focuses on creating and naming VLANs, and assigning access ports to specific VLANs.

Part 1: View the Default VLAN Configuration

Step 1: Display the current VLANs.

On S1, issue the command that displays all VLANs configured. By default, all interfaces are assigned to VLAN 1.

Step 2: Verify connectivity between PCs on the same network.

Notice that each PC can ping the other PC that shares the same network.

- PC1 can ping PC4
- PC2 can ping PC5
- PC3 can ping PC6

Pings to PCs in other networks fail.

What benefit will configuring VLANs provide to the current configuration?

Part 2: Configure VLANs

Step 1: Create and name VLANs on S1.

Create the following VLANs. Names are case-sensitive:

- VLAN 10: Faculty/Staff
- VLAN 20: Students
- VLAN 30: Guest(Default)
- VLAN 99: Management&Native

Step 2: Verify the VLAN configuration.

Which command will only display the VLAN name, status, and associated ports on a switch?

Step 3: Create the VLANs on S2 and S3.

Using the same commands from Step 1, create and name the same VLANs on S2 and S3.

Step 4: Verify the VLAN configuration.

Part 3: Assign VLANs to Ports

Step 1: Assign VLANs to the active ports on S2.

Assign the VLANs to the following ports:

- VLAN 10: Fast Ethernet 0/11
- VLAN 20: Fast Ethernet 0/18
- VLAN 30: Fast Ethernet 0/6

Step 2: Assign VLANs to the active ports on S3.

S3 uses the same VLAN access port assignments as S2.

Step 3: Verify loss of connectivity.

Previously, PCs that shared the same network could ping each other successfully. Try pinging between PC1 and PC4. Although the access ports are assigned to the appropriate VLANs, were the pings successful? Why?

_____

_____

What could be done to resolve this issue?

_____

Suggested Scoring Rubric

Activity Section	Question Location	Possible Points	Earned Points
Part 1: Verify the Default VLAN Configuration	Step 2	**4**	
Part 2: Configure VLANs	Step 2	**2**	
Part 3: Assign VLANs to Ports	Step 3	**4**	
	Packet Tracer Score	**90**	
	Total Score	**100**	

OR

Title: Configuration of Virtual Local Area Network (VLAN) and Inter-VLAN.

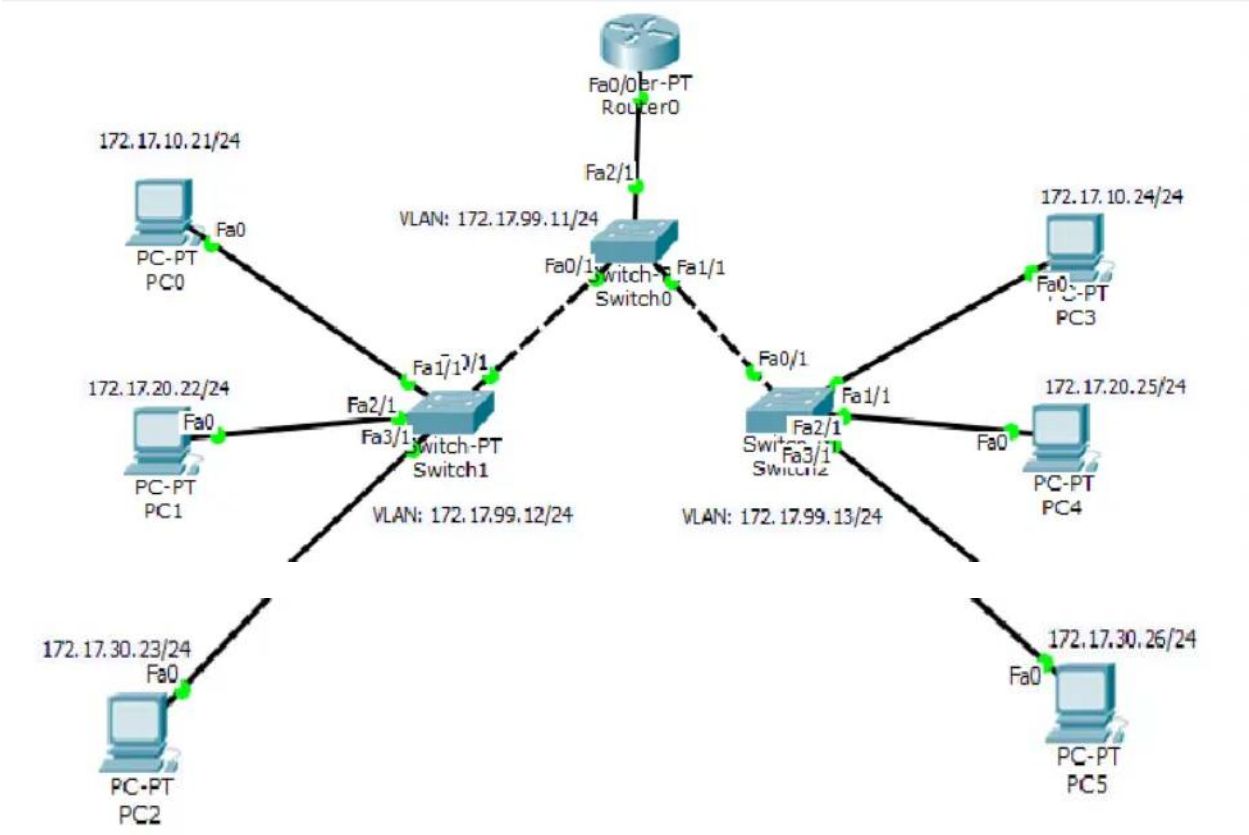
Objective:

- Configure and maintain Virtual Local Area Network (VLAN).
- Assign switch ports to a VLAN also assign the management VLAN.
- Configure trunking and verify that the switches can communicate to each other.
- Configuration of Inter-VLAN routing with sub-interfaces corresponding to the configured VLANs.

Tools: CISCO Packet Tracer 6.3.

Simulation:

VLAN is a virtual LAN. In technical terms, a VLAN is a broadcast domain created by switches. For configuring VLAN and Inter-VLAN, create a topology as in following figure.



Addressing Table

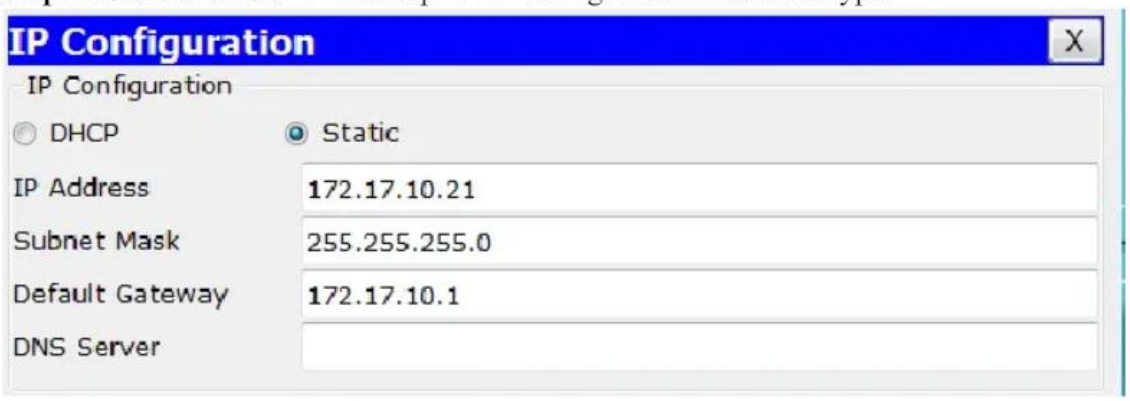
Device	Interface	IP Configuration	Default Gateway
S1	VLAN 99	172.17.99.11/24	-
S2	VLAN 99	172.17.99.12/24	-
S3	VLAN 99	172.17.99.13/24	-
PC0	-	172.17.10.21/24	172.17.10.1
PC1	-	172.17.20.22/24	172.17.20.1
PC2	-	172.17.30.23/24	172.17.30.1
PC3	-	172.17.10.24/24	172.17.10.1
PC4	-	172.17.20.25/24	172.17.20.1
PC5	-	172.17.30.26/24	172.17.30.1

For VLAN configuration, we following the steps one by one,

1. Configure the PCs IP Address.
2. User ports of S1 and S2 are enabled as access ports
3. Create VLAN on switch S0, S1 and S2.
4. Assign switch ports to VLANs on S1 and S2
5. Assign the management VLAN
6. Configure trunking and native VLAN for the Switches

These steps help to easy configuration.

Step 1: Go to the PC0 => Desktop => IP configuration mode and type



Same as configure all other five PCs. [PC1, PC2, PC3, PC4, PC5]

Step 2: Enabling the access ports. Just write these commands,

Switch 1 (S1)

Step 2: Switchport enable

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End
with CNTL/Z.
Switch(config)#int f1/1
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#int f2/1
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#int f3/1
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#
```

Switch 2 (S2)

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End
with CNTL/Z.
Switch(config)#int f1/1
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#int f2/1
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#int f3/1
Switch(config-if)#switchport mode access
Switch(config-if)#exit
Switch(config)#
```

Step 3: Create VLANs. The commands are same in all switches. So, repeat these commands in S0, S1, and S2.

Switch(S0,S1,S2)

```
Switch(config)#vlan 10
Switch(config-vlan)#name A
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name B
Switch(config-vlan)#exit
Switch(config)#vlan 30
Switch(config-vlan)#name C
Switch(config-vlan)#exit
Switch(config)#vlan 99
Switch(config-vlan)#name M
Switch(config-vlan)#exit
Switch(config)#
```

Step 4: Assign switch ports to VLANs.

Switch 1: (S1)

```
Switch(config)#int f1/1
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
Switch(config)#int f2/1
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
Switch(config)#int f3/1
Switch(config-if)#switchport access vlan 30
Switch(config-if)#exit
Switch(config)#
```

Switch 2: (S2)

```
Switch(config)#int f1/1
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
Switch(config)#int f2/1
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
Switch(config)#int f3/1
Switch(config-if)#switchport access vlan 30
Switch(config-if)#exit
Switch(config)#
```

Step 5: Assign Management VLAN in all switches (S0, S1, S2)

Switch 1 (S1)

```
Switch(config)#int vlan 99
Switch(config-if)#ip address 172.17.99.12
255.255.255.0
Switch(config-if)#exit
```

Switch 2 (S2)

```
Switch(config)#int vlan 99
Switch(config-if)#ip address 172.17.99.13 255.255.255.0
Switch(config-if)#exit
```

Switch 0 (S0)

```
Switch(config)#int vlan 99
Switch(config-if)#ip address 172.17.99.11
255.255.255.0
Switch(config-if)#exit
```

Step 6: Configuring trunk in S0, S1 and S2

Switch (S1 & S2)

```
Switch(config)#int f0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk native vlan 99
Switch(config-if)#exit
```

Switch 0 (S0)

```
Switch(config)#int f0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk native vlan 99
Switch(config-if)#exit
Switch(config)#
Switch(config)#int f1/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk native vlan 99
Switch(config-if)#exit
```

Yes!!! VLAN is done.

Now we configure the Inter-VLAN from the router. At first, enable the router interface by "no shutdown" command like as-

Router 0 (R0)

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End
with CNTL/Z.
Router(config)#int f0/0
Router(config-if)#no shut
```

Now assign default gateway and virtual gateway for VLANs by following commands,

Router 0 (R0)

```
Router(config-if)#int f0/0.1
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.1, changed state to up

Router(config-subif)#encapsulation dot1
Router(config-subif)#encapsulation dot1Q 1
Router(config-subif)#ip address 172.17.1.1 255.255.255.0
Router(config-subif)#exit
```

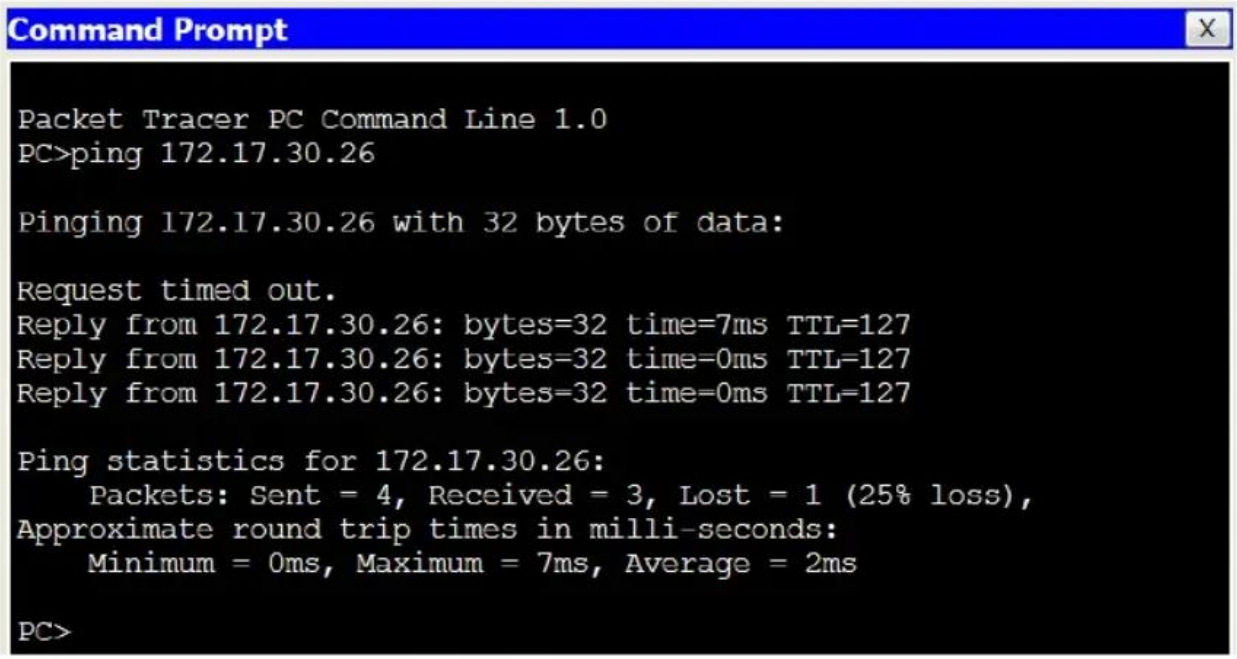
```
Router(config)#int f0/0.10
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.10, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.10, changed state to up

Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip address 172.17.10.1 255.255.255.0
Router(config-subif)#exit
Router(config-subif)#int f0/0.20
Router(config-subif)#encapsul
Router(config-subif)#encapsulation dot1
Router(config-subif)#encapsulation dot1Q 20
Router(config-subif)#ip address 172.17.20.1 255.255.255.0
Router(config-subif)#exit
Router(config)#int f0/0.30

Router(config-subif)#encapsulation dot1Q 30
Router(config-subif)#ip address 172.17.30.1 255.255.255.0
Router(config-subif)#exit
Router(config)#int f0/0.99
Router(config-subif)#encapsulation dot1Q 99 native
Router(config-subif)#ip address 172.17.99.1 255.255.255.0
Router(config-subif)#exit
Router(config)#
```

Finally trunk the switch ports which connect with router.

Switch 0 (R0)

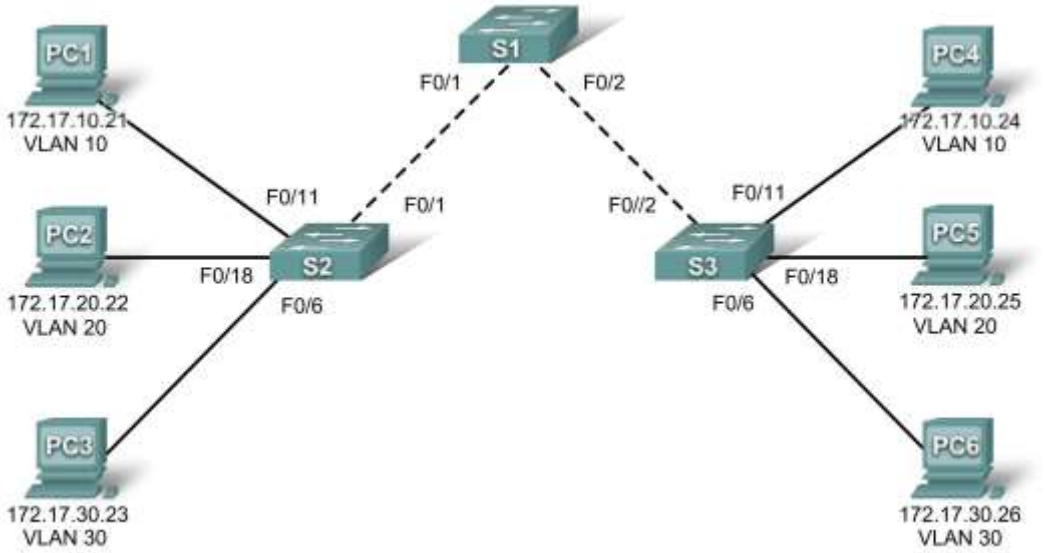
```
Switch(config)#int f1/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk native vlan 99
Switch(config-if)#exit
```

That's it.
Fully complete the configuration of VLAN and Inter-VLAN. For testing the connectivity among PCs, as usual we use the ping command in source PC's command prompt to write destination PC's IP address. For example here ping from PC0 to PC6 and observe the output.



Lab 5: Basic VLAN Configuration

Topology Diagram



Addressing Table

Device (Hostname)	Interface	IP Address	Subnet Mask	Default Gateway
S1	**VLAN 99**	172.17.99.11	255.255.255.0	N/A
S2	**VLAN 99**	172.17.99.12	255.255.255.0	N/A
S3	**VLAN 99**	172.17.99.13	255.255.255.0	N/A
PC1	**NIC**	172.17.10.21	255.255.255.0	172.17.10.1
PC2	**NIC**	172.17.20.22	255.255.255.0	172.17.20.1
PC3	**NIC**	172.17.30.23	255.255.255.0	172.17.30.1
PC4	**NIC**	172.17.10.24	255.255.255.0	172.17.10.1
PC5	**NIC**	172.17.20.25	255.255.255.0	172.17.20.1
PC6	**NIC**	172.17.30.26	255.255.255.0	172.17.30.1

Initial Port Assignments (Switches 2 and 3)

Ports	Assignment	Network
Fa0/1 – 0/5	802.1q Trunks (Native VLAN 99)	172.17.99.0 /24
Fa0/6 – 0/10	VLAN 30 – Guest (Default)	172.17.30.0 /24
Fa0/11 – 0/17	VLAN 10 – Faculty/Staff	172.17.10.0 /24
Fa0/18 – 0/24	VLAN 20 – Students	172.17.20.0 /24

Learning Objectives

Upon completion of this lab, you will be able to:

- Cable a network according to the topology diagram
- Erase the startup configuration and reload a switch to the default state
- Perform basic configuration tasks on a switch
- Create VLANs
- Assign switch ports to a VLAN
- Add, move, and change ports
- Verify VLAN configuration
- Enable trunking on inter-switch connections
- Verify trunk configuration
- Save the VLAN configuration

Task 1: Prepare the Network

Step 1: Cable a network that is similar to the one in the topology diagram.

You can use any current switch in your lab as long as it has the required interfaces shown in the topology.

Note: If you use 2900 or 2950 switches, the outputs may appear different. Also, certain commands may be different or unavailable.

Step 2: Clear any existing configurations on the switches, and initialize all ports in the shutdown state.

If necessary, refer to Lab 2.5.1, Appendix 1, for the procedure to clear switch configurations.

It is a good practice to disable any unused ports on the switches by putting them in shutdown. Disable all ports on the switches:

```
Switch#config term
Switch(config)#interface range fa0/1-24
Switch(config-if-range)#shutdown
Switch(config-if-range)#interface range gi0/1-2
Switch(config-if-range)#shutdown
```

Task 2: Perform Basic Switch Configurations

Step 1: Configure the switches according to the following guidelines.

- Configure the switch hostname.
- Disable DNS lookup.
- Configure an EXEC mode password of **class**.
- Configure a password of **cisco** for console connections.
- Configure a password of **cisco** for vty connections.

Step 2: Re-enable the user ports on S2 and S3.

```
S2(config)#interface range fa0/6, fa0/11, fa0/18 S2(config-if-range)#switchport
mode access S2(config-if-range)#no shutdown

S3(config)#interface range fa0/6, fa0/11, fa0/18
S3(config-if-range)#switchport mode access
S3(config-if-range)#no shutdown
```


Task 3: Configure and Activate Ethernet Interfaces

Step 1: Configure the PCs.

You can complete this lab using only two PCs by simply changing the IP addressing for the two PCs specific to a test you want to conduct. For example, if you want to test connectivity between PC1 and PC2, then configure the IP addresses for those PCs by referring to the addressing table at the beginning of the lab. Alternatively, you can configure all six PCs with the IP addresses and default gateways.

Task 4: Configure VLANs on the Switch

Step 1: Create VLANs on switch S1.

Use the **vlan** *vlan-id* command in global configuration mode to add a VLAN to switch S1. There are four VLANS configured for this lab: VLAN 10 (faculty/staff); VLAN 20 (students); VLAN 30 (guest); and VLAN 99 (management). After you create the VLAN, you will be in vlan configuration mode, where you can assign a name to the VLAN with the **name** *vlan name* command.

```
S1(config)#vlan 10
S1(config-vlan)#name faculty/staff
S1(config-vlan)#vlan 20
S1(config-vlan)#name students
S1(config-vlan)#vlan 30
S1(config-vlan)#name guest
S1(config-vlan)#vlan 99
S1(config-vlan)#name management
S1(config-vlan)#end
S1#
```

Step 2: Verify that the VLANs have been created on S1.

Use the **show vlan brief** command to verify that the VLANs have been created.

```
S1#show vlan brief

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------------------------
1    default                           active    Fa0/1, Fa0/2, Fa0/4, Fa0/5
                                                 Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                                  Fa0/10, Fa0/11, Fa0/12, Fa0/13
                          Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                 Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                 Fa0/22, Fa0/23, Fa0/24, Gi0/1
                                                 Gi0/2
10   faculty/staff                     active
20   students                          active
30   guest                             active
99   management                        active
```

Step 3: Configure and name VLANs on switches S2 and S3.

Create and name VLANs 10, 20, 30, and 99 on S2 and S3 using the commands from Step 1. Verify the correct configuration with the **show vlan brief** command.

What ports are currently assigned to the four VLANs you have created?

Step 4: Assign switch ports to VLANs on S2 and S3.

Refer to the port assignment table on page 1. Ports are assigned to VLANs in interface configuration mode, using the **switchport access vlan** *vlan-id* command. You can assign each port individually or you can use the **interface range** command to simplify this task, as shown here. The commands are shown for S3 only, but you should configure both S2 and S3 similarly. Save your configuration when done.


```
S3(config)#interface range fa0/6-10
S3(config-if-range)#switchport access vlan 30
S3(config-if-range)#interface range fa0/11-17
S3(config-if-range)#switchport access vlan 10
S3(config-if-range)#interface range fa0/18-24
S3(config-if-range)#switchport access vlan 20
S3(config-if-range)#end
S3#copy running-config startup-config Destination
filename [startup-config]? [enter]
Building configuration...
[OK]
```

Step 5: Determine which ports have been added.

Use the **show vlan id** *vlan-number* command on S2 to see which ports are assigned to VLAN 10.

Which ports are assigned to VLAN 10?

_____

Note: The **show vlan name** *vlan-name* displays the same output.

You can also view VLAN assignment information using the **show interfaces** *interface* **switchport** command.

Step 6: Assign the management VLAN.

A management VLAN is any VLAN that you configure to access the management capabilities of a switch. VLAN 1 serves as the management VLAN if you did not specifically define another VLAN. You assign the management VLAN an IP address and subnet mask. A switch can be managed via HTTP, Telnet, SSH, or SNMP. Because the out-of-the-box configuration of a Cisco switch has VLAN 1 as the default VLAN, VLAN 1 is a bad choice as the management VLAN. You do not want an arbitrary user who is connecting to a switch to default to the management VLAN. Recall that you configured the management VLAN as VLAN 99 earlier in this lab.

From interface configuration mode, use the **ip address** command to assign the management IP address to the switches.

```
S1(config)#interface vlan 99
S1(config-if)#ip address 172.17.99.11 255.255.255.0
S1(config-if)#no shutdown

S2(config)#interface vlan 99
S2(config-if)#ip address 172.17.99.12 255.255.255.0
S2(config-if)#no shutdown

S3(config)#interface vlan 99
S3(config-if)#ip address 172.17.99.13 255.255.255.0
S3(config-if)#no shutdown
```

Assigning a management address allows IP communication between the switches, and also allows any host connected to a port assigned to VLAN 99 to connect to the switches. Because VLAN 99 is configured as the management VLAN, any ports assigned to this VLAN are considered management ports and should be secured to control which devices can connect to these ports.

Step 7: **Configure trunking and the native VLAN for the trunking ports on all switches.**

Trunks are connections between the switches that allow the switches to exchange information for all VLANS. By default, a trunk port belongs to all VLANs, as opposed to an access port, which can only belong to a single VLAN. If the switch supports both ISL and 802.1Q VLAN encapsulation, the trunks must specify which method is being used. Because the 2960 switch only supports 802.1Q trunking, it is not specified in this lab.

A native VLAN is assigned to an 802.1Q trunk port. In the topology, the native VLAN is VLAN 99. An 802.1Q trunk port supports traffic coming from many VLANs (tagged traffic) as well as traffic that does not come from a VLAN (untagged traffic). The 802.1Q trunk port places untagged traffic on the native VLAN. Untagged traffic is generated by a computer attached to a switch port that is configured with the native

VLAN. One of the IEEE 802.1Q specifications for Native VLANs is to maintain backward compatibility with untagged traffic common to legacy LAN scenarios. For the purposes of this lab, a native VLAN serves as a common identifier on opposing ends of a trunk link. It is a best practice to use a VLAN other than VLAN 1 as the native VLAN.

Use the **interface range** command in global configuration mode to simplify configuring trunking.

```
S1(config)#interface range fa0/1-5
S1(config-if-range)#switchport mode trunk
S1(config-if-range)#switchport trunk native vlan 99
S1(config-if-range)#no shutdown
S1(config-if-range)#end
```

```
S2(config)# interface range fa0/1-5
S2(config-if-range)#switchport mode trunk
S2(config-if-range)#switchport trunk native vlan 99
S2(config-if-range)#no shutdown
S2(config-if-range)#end
```

```
S3(config)# interface range fa0/1-5
S3(config-if-range)#switchport mode trunk
S3(config-if-range)#switchport trunk native vlan 99
S3(config-if-range)#no shutdown
S3(config-if-range)#end
```

Verify that the trunks have been configured with the **show interface trunk** command.

```
S1#show interface trunk

Port        Mode         Encapsulation  Status        Native vlan
Fa0/1       on           802.1q         trunking      99
Fa0/2       on           802.1q         trunking      99

Port      Vlans allowed on trunk
Fa0/1       1-4094 Fa0/2
1-4094

Port        Vlans allowed and active in management domain
Fa0/1       1,10,20,30,99
Fa0/2       1,10,20,30,99

Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       1,10,20,30,99
Fa0/2       1,10,20,30,99
```

Step 8: Verify that the switches can communicate.

From S1, ping the management address on both S2 and S3.

```
S1#ping 172.17.99.12
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.99.12, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/9 ms
```

```
S1#ping 172.17.99.13
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.99.13, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```


Step 9: Ping several hosts from PC2.

Ping from host PC2 to host PC1 (172.17.10.21). Is the ping attempt successful?

Ping from host PC2 to the switch VLAN 99 IP address 172.17.99.12. Is the ping attempt successful?
Ping from host PC2 to host PC5. Is the ping attempt successful?

Step 10: Move PC1 into the same VLAN as PC2.

The port connected to PC2 (S2 Fa0/18) is assigned to VLAN 20, and the port connected to PC1 (S2 Fa0/11) is assigned to VLAN 10. Reassign the S2 Fa0/11 port to VLAN 20. You do not need to first remove a port from a VLAN to change its VLAN membership. After you reassign a port to a new VLAN, that port is automatically removed from its previous VLAN.

```
S2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
S2(config)#interface fastethernet 0/11
S2(config-if)#switchport access vlan 20
S2(config-if)#end
```

Ping from host PC2 to host PC1. Is the ping attempt successful?

Even though the ports used by PC1 and PC2 are in the same VLAN, they are still in different subnetworks, so they cannot communicate directly.

Step 11: Change the IP address and network on PC1.

Change the IP address on PC1 to 172.17.20.22. The subnet mask and default gateway can remain the same. Once again, ping from host PC2 to host PC1, using the newly assigned IP address.

Is the ping attempt successful?

Why was this attempt successful?

_____

_____.

Task 5: Document the Switch Configurations

On each switch, capture the running configuration to a text file and save it for future reference.

Task 6: Clean Up

Erase the configurations and reload the switches. Disconnect and store the cabling. For PC hosts that are normally connected to other networks (such as the school LAN or to the Internet), reconnect the appropriate cabling and restore the TCP/IP settings.

Final Switch Configurations

S1

```
hostname S1 !
enable secret class no
ip domain-lookup
! interface FastEthernet0/1
switchport trunk native vlan 99
switchport mode trunk
! interface FastEthernet0/2
switchport trunk native vlan 99
switchport mode trunk
```

```
! interface FastEthernet0/3
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/4
switchport trunk native vlan 99
switchport mode trunk
! interface FastEthernet0/5
switchport trunk native vlan 99
switchport mode trunk !
interface FastEthernet0/6
shutdown
!
<all remaining FastEthernet and GigabitEthernet interface are shutdown>
!
interface Vlan1
no ip address  no
ip route-cache
! interface
Vlan99
 ip address 172.17.99.11 255.255.255.0  no
ip route-cache
! line con 0  exec-
timeout 0 0
password cisco
logging synchronous
login line vty 0 4
exec-timeout 0 0
password cisco
logging synchronous
login line vty 5 15
exec-timeout 0 0
password cisco
logging synchronous
login ! end
```

S2

```
hostname S2 !
enable secret class no
ip domain-lookup
!
interface FastEthernet0/1
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/2
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/3
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/4
switchport trunk native vlan 99
switchport mode trunk
!
```

```
interface FastEthernet0/5
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/6
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/7
switchport access vlan 30
switchport mode access  shutdown
!
interface FastEthernet0/8
switchport access vlan 30
switchport mode access  shutdown
!
interface FastEthernet0/9
switchport access vlan 30
switchport mode access  shutdown
!
interface FastEthernet0/10
switchport access vlan 30
switchport mode access  shutdown
!
interface FastEthernet0/11
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/12
switchport access vlan 10
switchport mode access  shutdown
!
interface FastEthernet0/13
switchport access vlan 10
switchport mode access  shutdown
!
interface FastEthernet0/14
switchport access vlan 10
switchport mode access  shutdown
!
interface FastEthernet0/15
switchport access vlan 10
switchport mode access  shutdown
!
interface FastEthernet0/16
switchport access vlan 10
switchport mode access  shutdown
!
interface FastEthernet0/17
switchport access vlan 10
switchport mode access  shutdown
!
interface FastEthernet0/18
switchport access vlan 20
switchport mode access
!
```

```
interface FastEthernet0/19
switchport access vlan 20
switchport mode access  shutdown
!
interface FastEthernet0/20
switchport access vlan 20
switchport mode access  shutdown
!
interface FastEthernet0/21
switchport access vlan 20
switchport mode access  shutdown
!
interface FastEthernet0/22
switchport access vlan 20
switchport mode access  shutdown
!
interface FastEthernet0/23
switchport access vlan 20
switchport mode access  shutdown
!
interface FastEthernet0/24
switchport access vlan 20
switchport mode access  shutdown
! interface
GigabitEthernet0/1  shutdown
!
interface GigabitEthernet0/2
shutdown
!
interface Vlan1
no ip address  no
ip route-cache
shutdown
!
interface Vlan99  ip address
172.17.99.12 255.255.255.0  no ip
route-cache !
ip http server !
control-plane
! !
line con 0  exec-
timeout 0 0
password cisco
logging synchronous
login line vty 0 4
exec-timeout 0 0
password cisco
logging synchronous
login
line vty 5 15  exec-
timeout 0 0
password cisco
logging synchronous
login ! ! end
```

```
S3 hostname S3 no ip
domain-lookup
enable secret class
!
interface FastEthernet0/1
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/2
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/3
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/4
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/5
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/6
switchport access vlan 30
switchport mode access
!
interface FastEthernet0/7
switchport access vlan 30
!
interface FastEthernet0/8
switchport access vlan 30
!
interface FastEthernet0/9
switchport access vlan 30
!
interface FastEthernet0/10
switchport access vlan 30
!
interface FastEthernet0/11
switchport access vlan 10
switchport mode access
!
interface FastEthernet0/12
switchport access vlan 10
!
interface FastEthernet0/13
switchport access vlan 10
!
interface FastEthernet0/14
switchport access vlan 10
!
interface FastEthernet0/15
switchport access vlan 10
!
interface FastEthernet0/16
switchport access vlan 10
```



```
!
interface FastEthernet0/17
switchport access vlan 10
!
interface FastEthernet0/18
switchport access vlan 20
switchport mode access
!
interface FastEthernet0/19
switchport access vlan 20
!
interface FastEthernet0/20
switchport access vlan 20
!
interface FastEthernet0/21
switchport access vlan 20
!
interface FastEthernet0/22
switchport access vlan 20
!
interface FastEthernet0/23
switchport access vlan 20
!
interface FastEthernet0/24
switchport access vlan 20
!
interface GigabitEthernet0/1
!
interface GigabitEthernet0/2
!
interface Vlan1
no ip address  no
ip route-cache
shutdown
!
interface Vlan99  ip address
172.17.99.13 255.255.255.0  no ip
route-cache
!
line con 0
password cisco
login line vty
0 4  password
cisco  login
line vty 5 15
password cisco
login ! end
```


Experiment No. 6

1. **Aim:** Installation and Configuration of virtualization using KVM

2. **Objectives:** From this experiment, the student will be able to,
   - Understand the concepts of virtualization.
   - Understand KVM architecture and its configuration.

3. **Outcomes:** The learner will be able**,**

   - To analyze user models and develop user centric interfaces
   - To analyze the local and global impact of computing on individuals, organizations, and society.
   - To engage in life-long learning development and higher studies.
   - To understand, identify, analyze and design the problem, implement and validate the solution including both hardware and software.

4. **Hardware / Software Required:** Ubuntu operating system, open source software KVM, Internet.

5. **Theory:** Virtualization is software that separates physical infrastructures to create various dedicated resources. It is the fundamental technology that powers cloud computing.

   The technology behind virtualization is known as a virtual machine monitor (VMM) or virtual manager, which separates compute environments from the actual physical infrastructure.

   Virtualization makes servers, workstations, storage and other systems independent of the physical hardware layer. This is done by installing a Hypervisor on top of the hardware layer, where the systems are then installed.

   There are three areas of IT where virtualization is making headroads, network virtualization, storage virtualization and server virtualization:

   • Network virtualization is a method of combining the available resources in a network by splitting up the available bandwidth into channels, each of which is independent from the others, and each of which can be assigned (or reassigned) to a particular server or device in real time. The idea is that virtualization disguises the true complexity of the network by separating it into manageable parts, much like your partitioned hard drive makes it easier to manage your files.

   • Storage virtualization is the pooling of physical storage from multiple network storage devices into what appears to be a single storage device that is

managed from a central console. Storage virtualization is commonly used in storage area networks (SANs).

• Server virtualization is the masking of server resources (including the number and identity of individual physical servers, processors, and operating systems) from server users. The intention is to spare the user from having to understand and manage complicated details of server resources while increasing resource sharing and utilization and maintaining the capacity to expand later.

Virtualization can be viewed as part of an overall trend in enterprise IT that includes autonomic computing, a scenario in which the IT environment will be able to manage itself based on perceived activity, and utility computing, in which computer processing power is seen as a utility that clients can pay for only as needed. The usual goal of virtualization is to centralize administrative tasks while improving scalability and work loads.

6. Procedure:

Installation Steps :

1. #sudo grep -c "svm\|vmx" /proc/cpuinfo
2. #sudo apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager
3. #sudoadduserrait

#sudoadduserraitlibvirtd

After running this command, log out and log back in as rait

4. Run following command after logging back in as rait and you should see an empty list of virtual machines. This indicates that everything is working correctly.

#virsh -c qemu:///system list

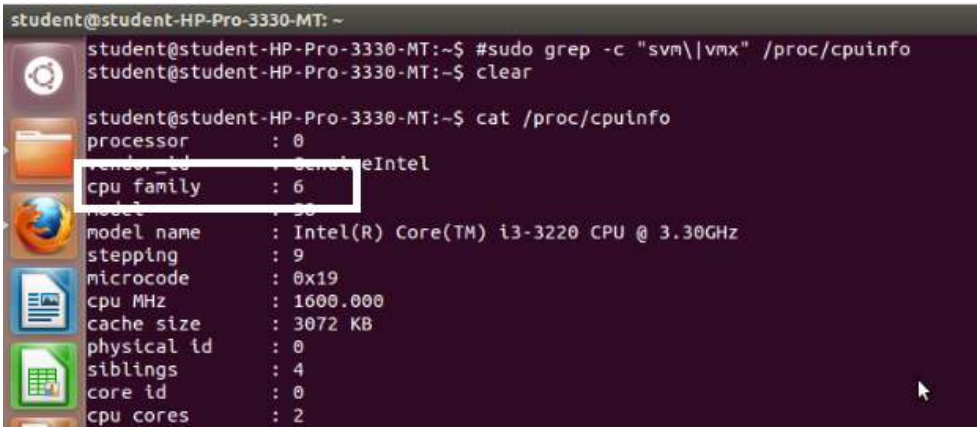
5. Open Virtual Machine Manager application and Create Virtual Machine

#virt-manager

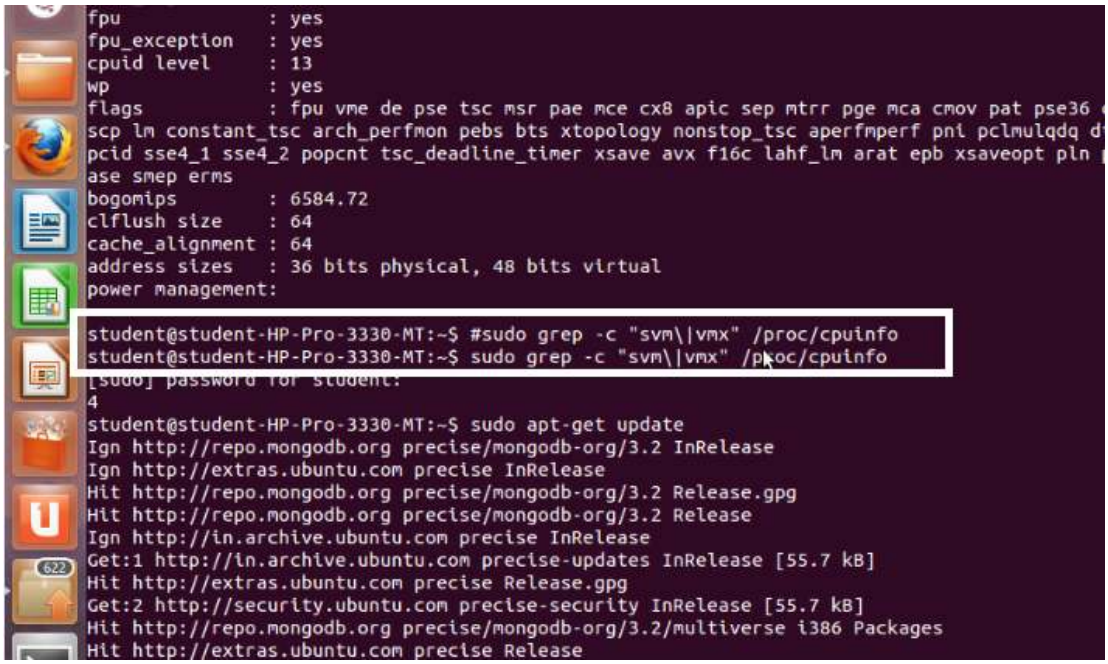
7. Result:

SNAPSHOTS_____

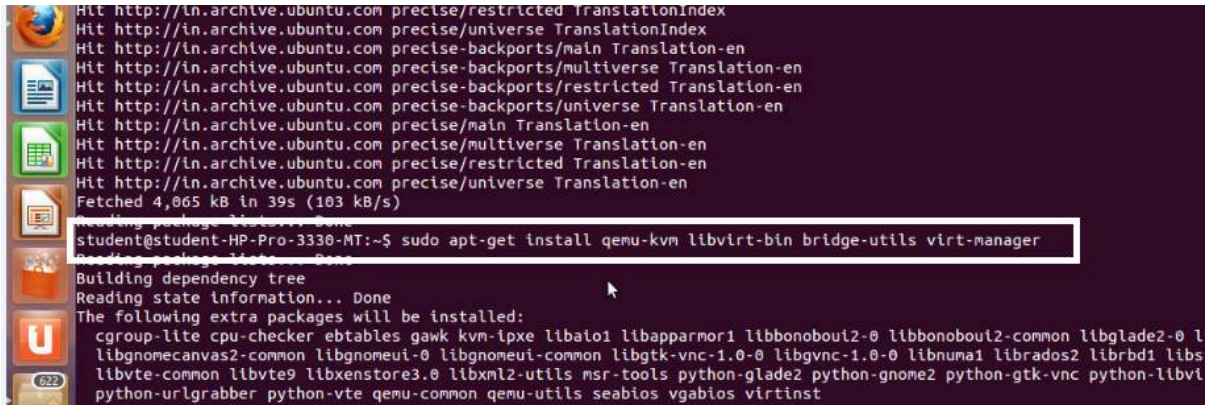
Step 1 : #sudo grep -c "svm\|vmx" /proc/cpuinfo



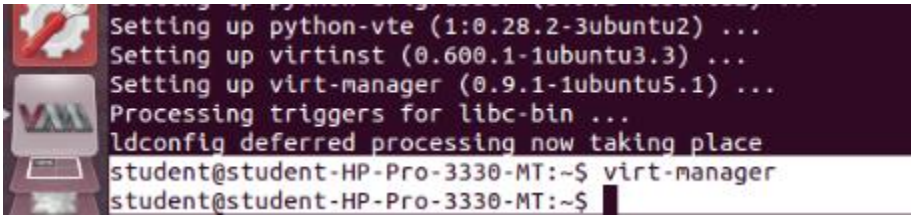
Step 2 : #sudo apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager



Step 3 : #sudoadduserrait
After running this command, log out and log back in as rait

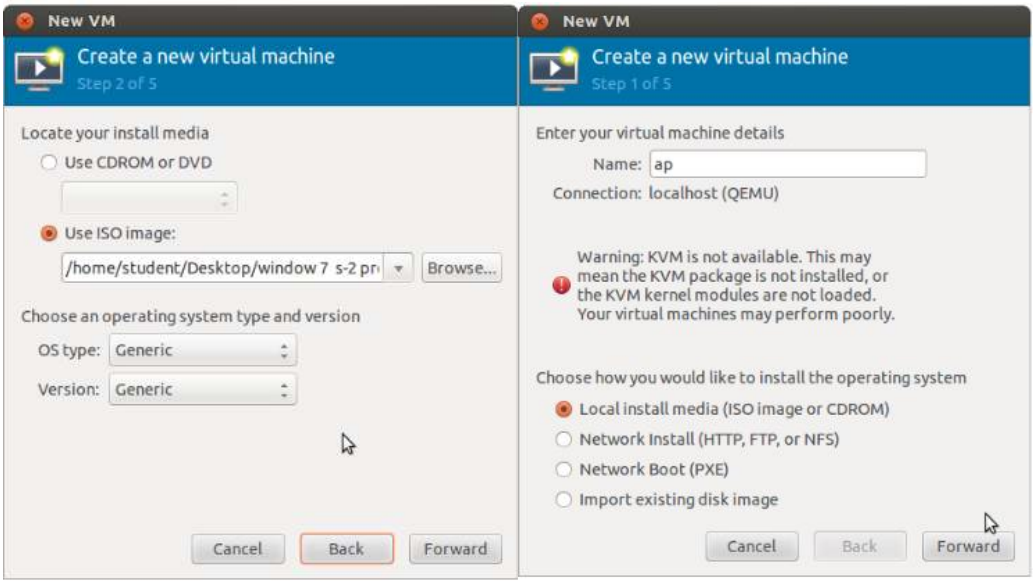


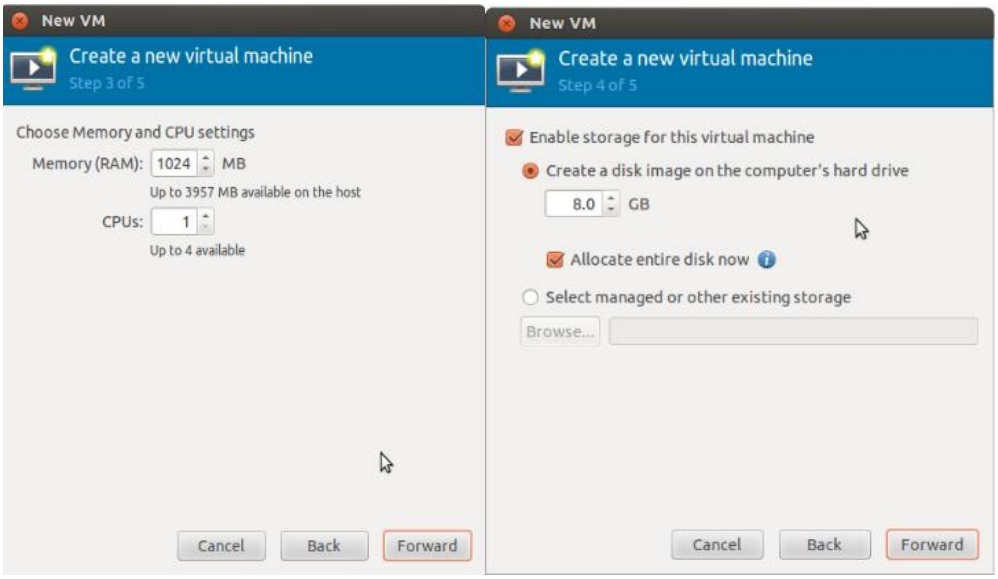
Step 4 : #sudoadduserraitlibvirtd
After running this command, log out and log back in as rait



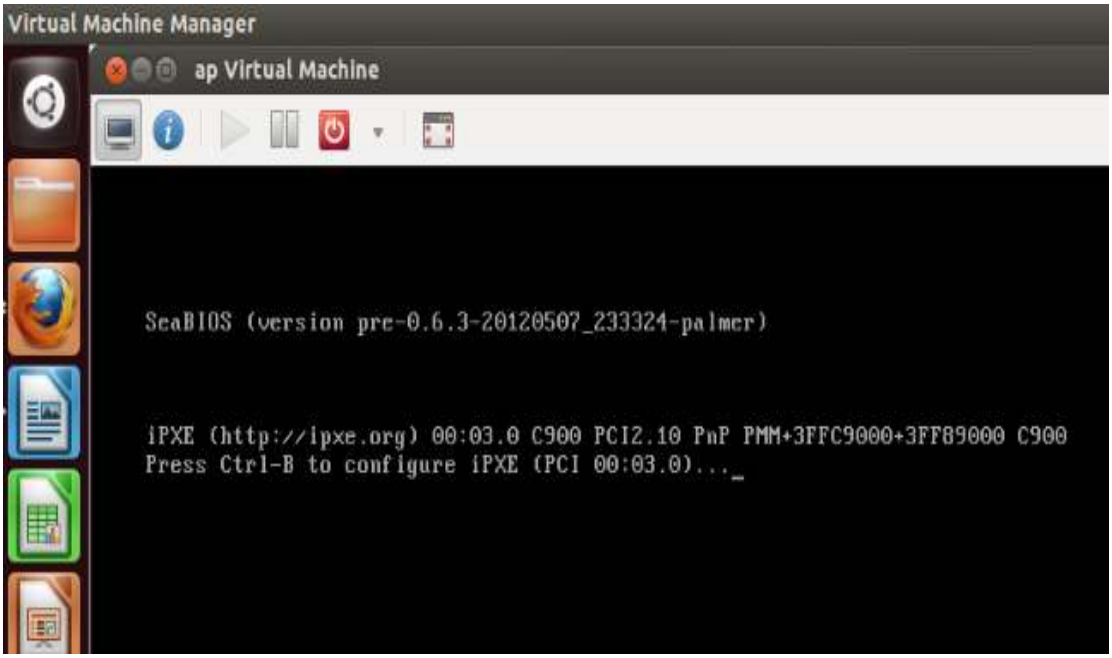
Step 5 : Open Virtual Machine Manager application and Create Virtual Machine #virt-manager as shown below

Step 6 : Create a new virtual machine as shown below

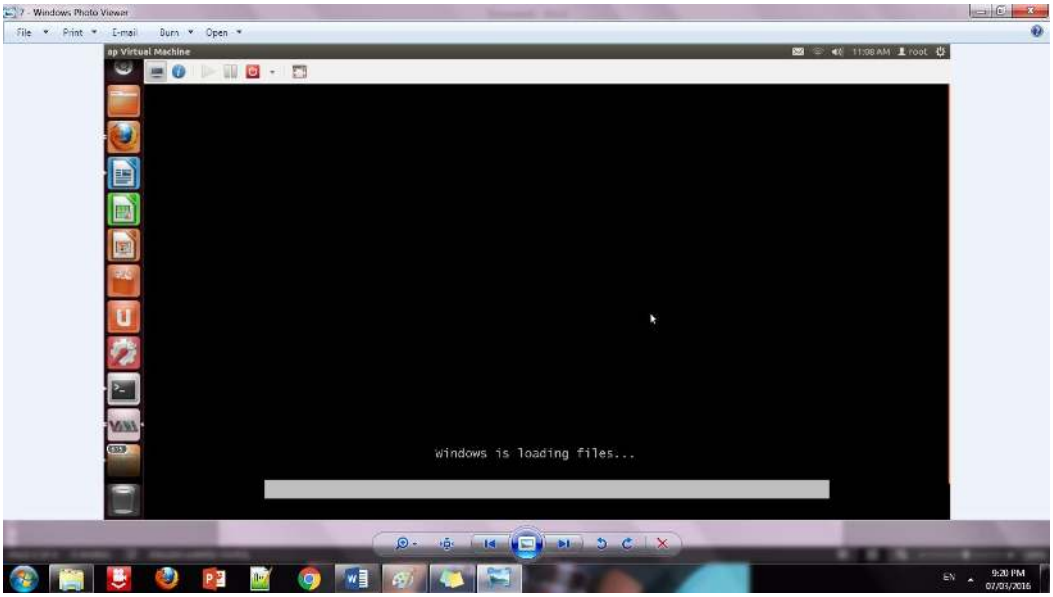




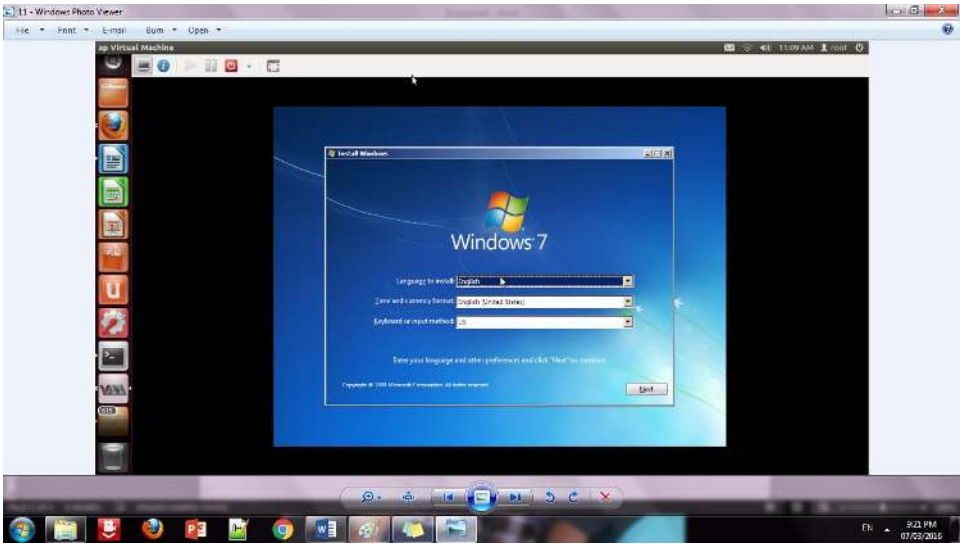
Step 7 : Install windows operating system on virtual machine



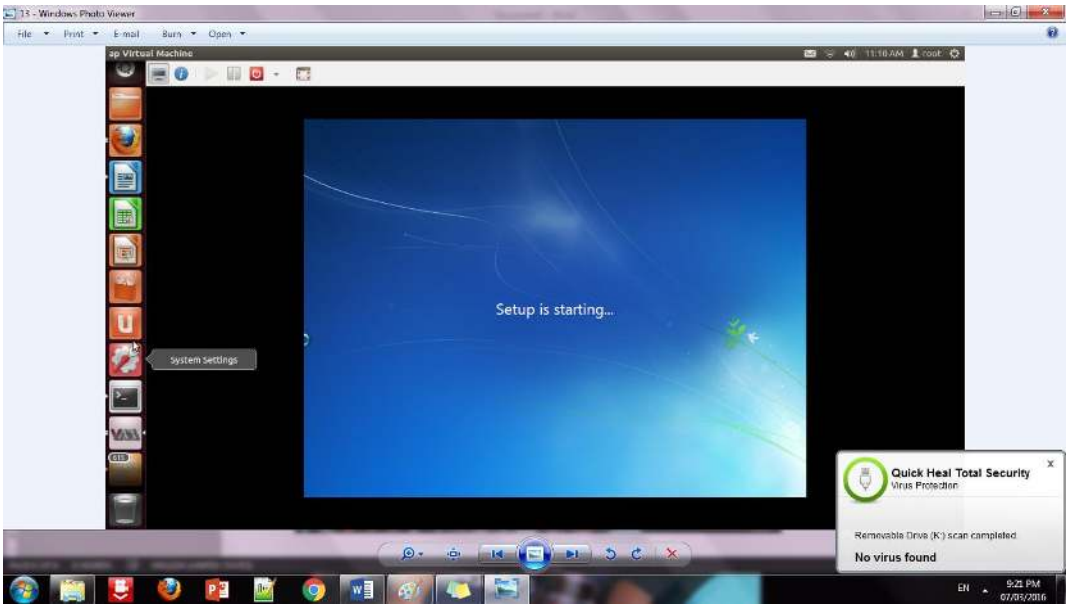
Step 8: Installation of windows on virtual machine



Step 9: Installation of windows 7 on virtual machine



Step 10: Initialization of windows on virtual machine



8. Conclusion:

Installation and configuration of KVM have been done successfully onto Ubantu and users added. Like this we can create as many virtual machines as possible on OS and can install any windows onto it

9. Viva Questions:

- What is virtualization ?
- What is the benefit of desktop virtualization?
- What are the different virtual machines available?

10. References:

1. Enterprise Cloud Computing by Gautam Shroff, Cambridge,2010
2. Cloud Security by Ronald Krutz and Russell Dean Vines, Wiley - India, 2010 ISBN:978-0-470-58987-8
3. Getting Started with OwnCloud by Aditya Patawar , Packt Publishing Ltd, 2013

Ex No:7 Create Nested Virtual Machine(VM under another VM)

Aim:

Enable Nested Virtualization

How to Set Up Hyper-V Nested Virtualization [Step-by-Step]

1. Creating A NAT-Enabled Virtual Switch.
2. Creating the First-Level Guest Virtual Machine. ...
3. Enabling Nested Virtualization.
4. Installing The Guest Operating System.
5. Connecting To The Network.
6. Installing Hyper-V On The First-Level Virtual Machine.
7. Conclusion.

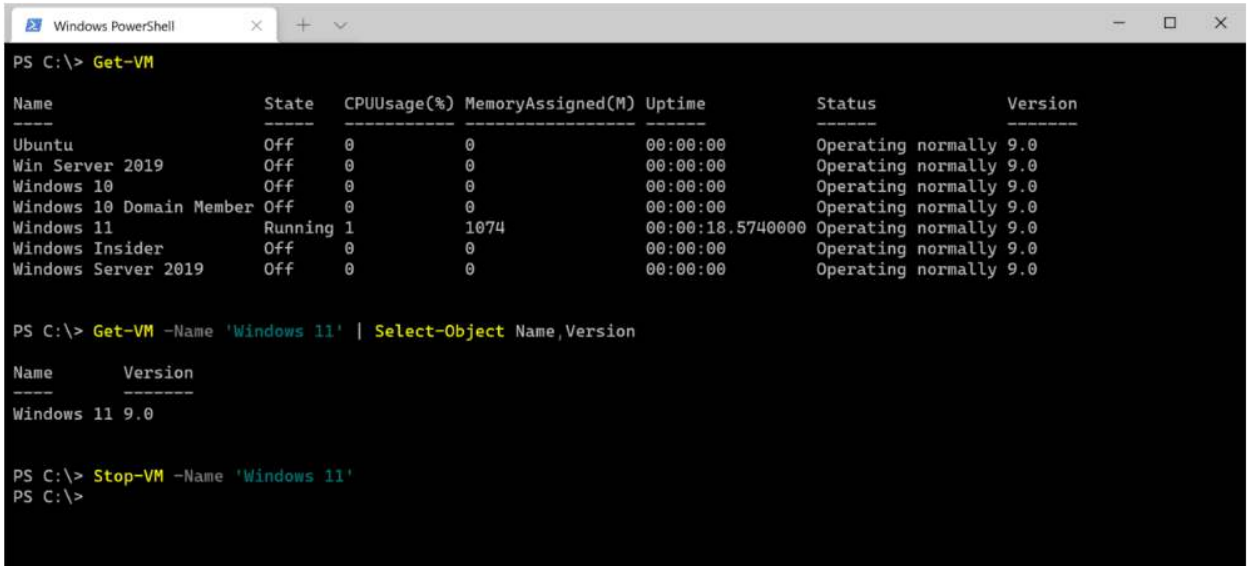
Enable Nested Virtualization in Windows 10 Hyper-V

Hyper-V is the built-in hypervisor that comes free in Windows and Windows Server. It is used to run on Windows. Virtualization is also used for other features, like Virtualization-Based Security (VBS), Windows Sandbox, and Windows Defender Application Guard (WDAG). Developers sometimes use virtualization with Visual Studio to run device emulators.

Microsoft introduced support for nested virtualization in Windows Server 2016. Nested virtualization lets you turn on . So, you can think of it like a VM running inside a VM.

There are a few prerequisites that you need to meet before you can use nested virtualization. The VM configuration must be version 8.0 or higher. And nested virtualization is only supported on Intel CPUs with virtualization (VT-x) and Extended Page Tables (EPT).

```
Stop-VM -Name 'Windows 11'
```

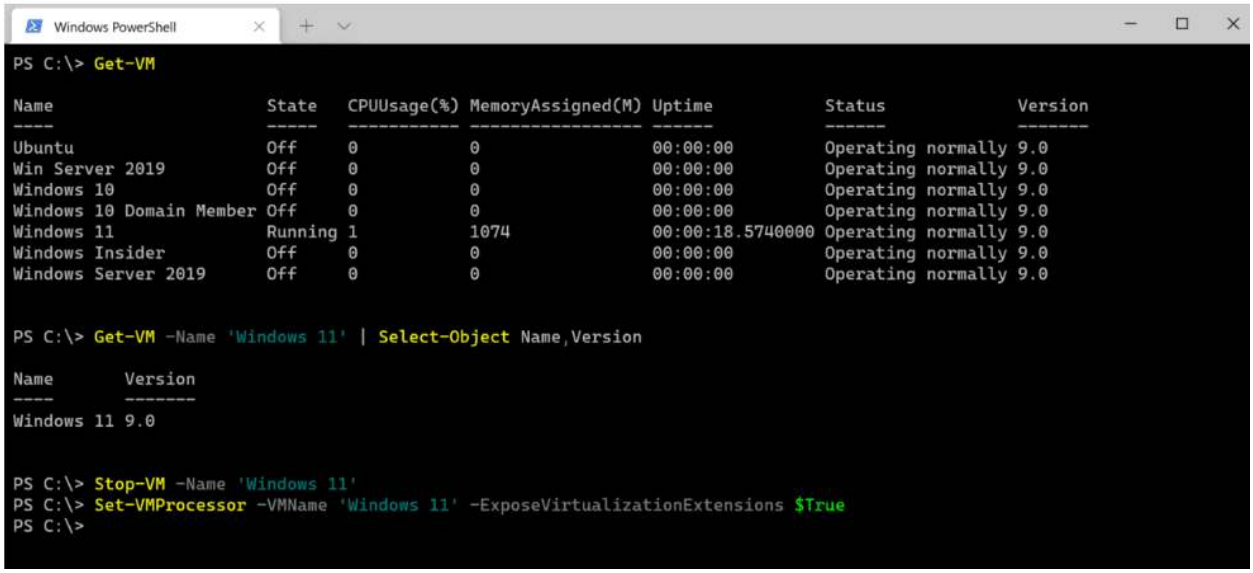


How to Enable Nested Virtualization in Hyper-V (Image Credit: Russell Smith)

```
Update-VMVersion -Name 'Windows 11'
```

Now that the state of your VM is set to 'Off', you can enable nested virtualization. The only way to enable it is using PowerShell. In a PowerShell terminal window, run the command below, replacing *Windows 11* with the name of your VM.

```
Set-VMProcessor -VMName 'Windows 11' -ExposeVirtualizationExtensions $True
```

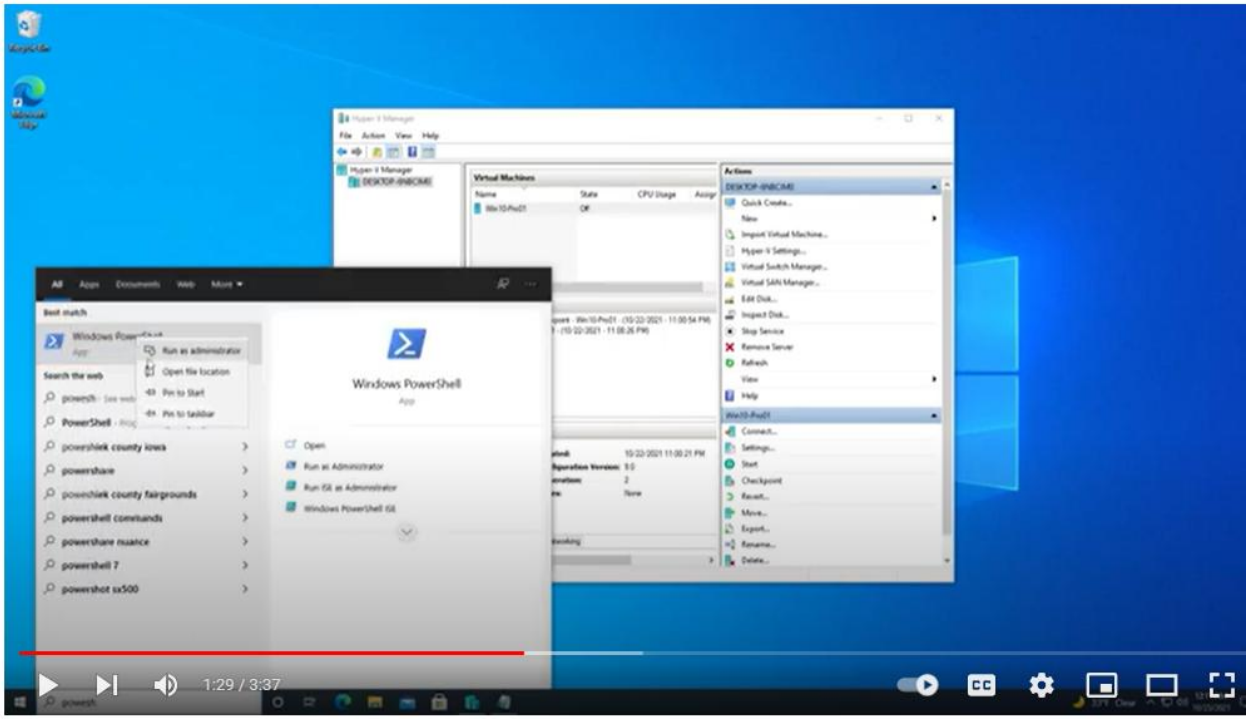


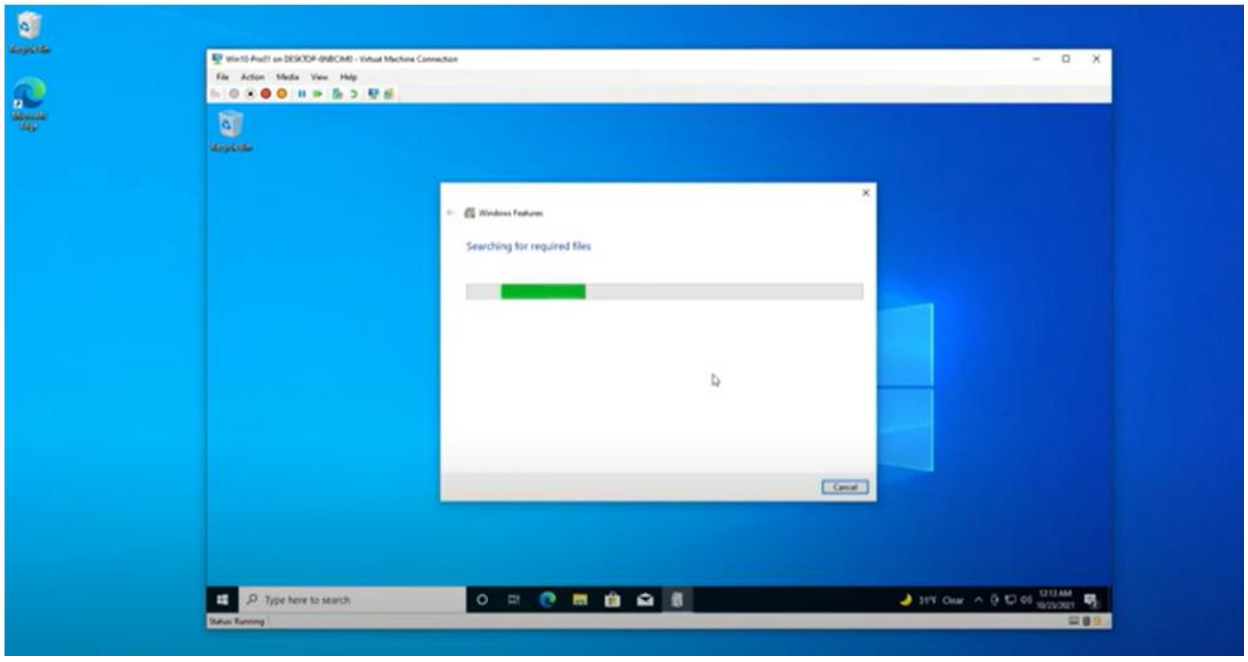
How to Enable Nested Virtualization in Hyper-V (Image Credit: Russell Smith)

Now all that's left to do is start the VM, which you can do either using Hyper-V Manager or using PowerShell as shown below:

```
Start-VM -Name 'Windows 11'
```

When the VM has started, log in to Windows and you will be able to install the Hyper-V feature or server role, and any other features that depend on it.


Windows PowerShell
1:29 / 3:37


Windows Features
Searching for required files
Cancel
Type here to search