



Names: Shady Mohamed Abdel Gawad **ID:** 20200246

Names: Mohamed Ayman Eze El regal **ID:** 20200432

Subject: Topics in Information Technology-2

(Cloud Computing)

Under the supervision of: Dr/ Nour Mahmoud

Lab Assignment: 1

Group: S1

1. Part 1 search report:

1. What are the differences between virtualization and containerization?

Resource Isolation: Virtualization isolates the entire operating system for each virtual machine (VM). Containers share the host operating system's kernel, isolating applications from each other but not the underlying system.

Operating System: VMs require a full guest operating system to be installed. Containers leverage the host's kernel, reducing resource consumption.

Performance: VMs can be slower to start and consume more resources due to guest OS overhead. Containers are lightweight and start up faster.

Use Cases: VMs are ideal for running applications requiring specific operating systems or for complete system isolation. Containers excel at deploying microservices and portable applications within a shared environment.

2. What is the architecture of a virtual machine and a container?

Virtual Machine Architecture:

Physical Hardware: The underlying physical server.

Hypervisor: Software that creates and manages VMs, providing a virtual layer above the hardware.

Guest Operating System: The operating system installed within each VM.

Application: The software running on the guest OS.

Container Architecture:

Physical Hardware: The underlying physical server.

Host Operating System: The main operating system running on the server.

Container Engine (e.g., Docker): Software that manages containers.

Container Image: A lightweight, executable package containing the application and its dependencies.

Container: An isolated instance of a container image running on the host OS.

3. What is Docker?

Docker is a popular open-source platform for developing, deploying, and managing applications using containers. It provides tools for building container images, running containers, and sharing them across environments.

4. What is a Docker file, Docker image, and Docker container?

Docker file: A text file containing instructions for building a Docker image. It specifies the base operating system, dependencies, and application code to be included in the image.

Docker Image: A read-only template that encapsulates the application code, configuration files, and dependencies required to run the application. It's essentially a blueprint for creating containers.

Docker Container: A running instance of a Docker image. It provides an isolated environment for the application to execute. Multiple containers can be created from a single image.

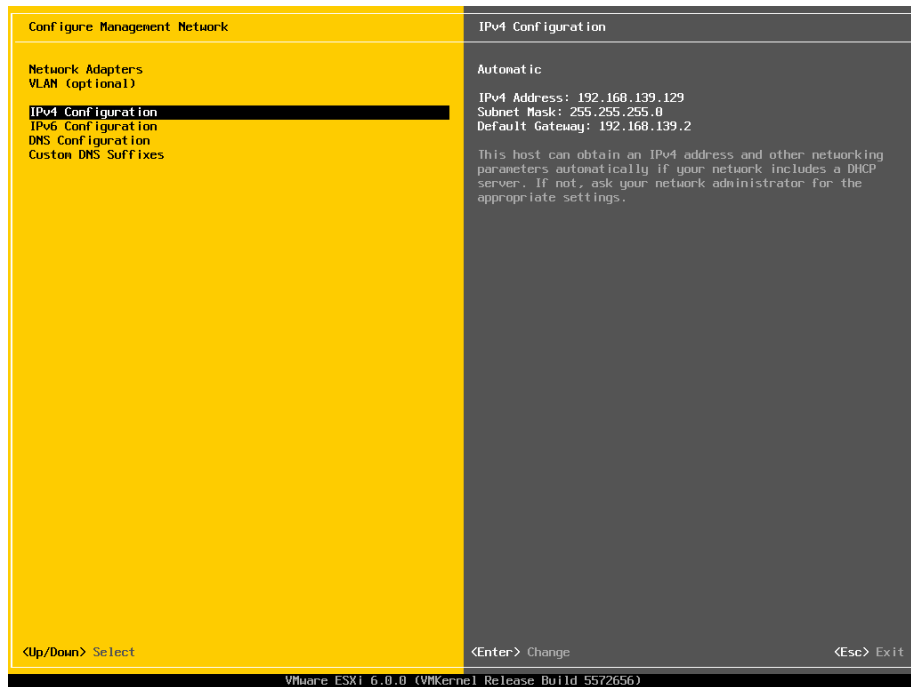
5. Differences between Kubernetes and Docker?

Purpose: Docker focuses on building, sharing, and running individual containers. Kubernetes is a container orchestration platform that manages the deployment, scaling, and networking of containerized applications. It allows you to automate container deployments across clusters of machines.

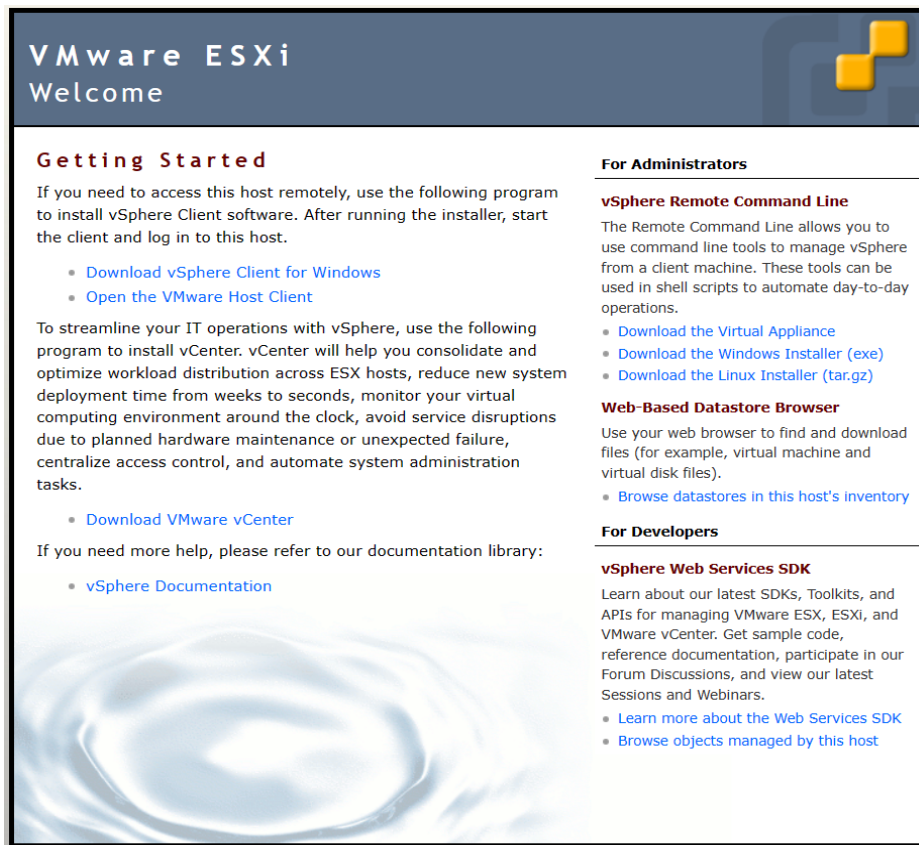
Scope: Docker deals with individual containers. Kubernetes manages the entire lifecycle of containerized applications at scale.

Complexity: Docker is simpler to learn and use. Kubernetes requires a deeper understanding of container orchestration principles.

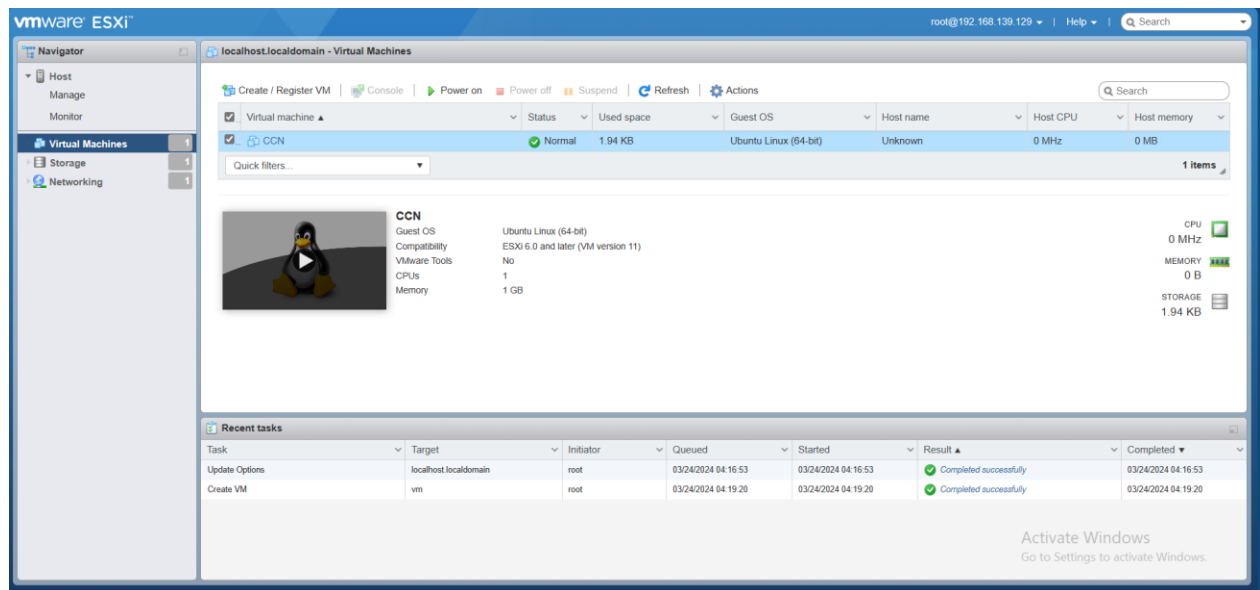
2. Part 2 Lab task:



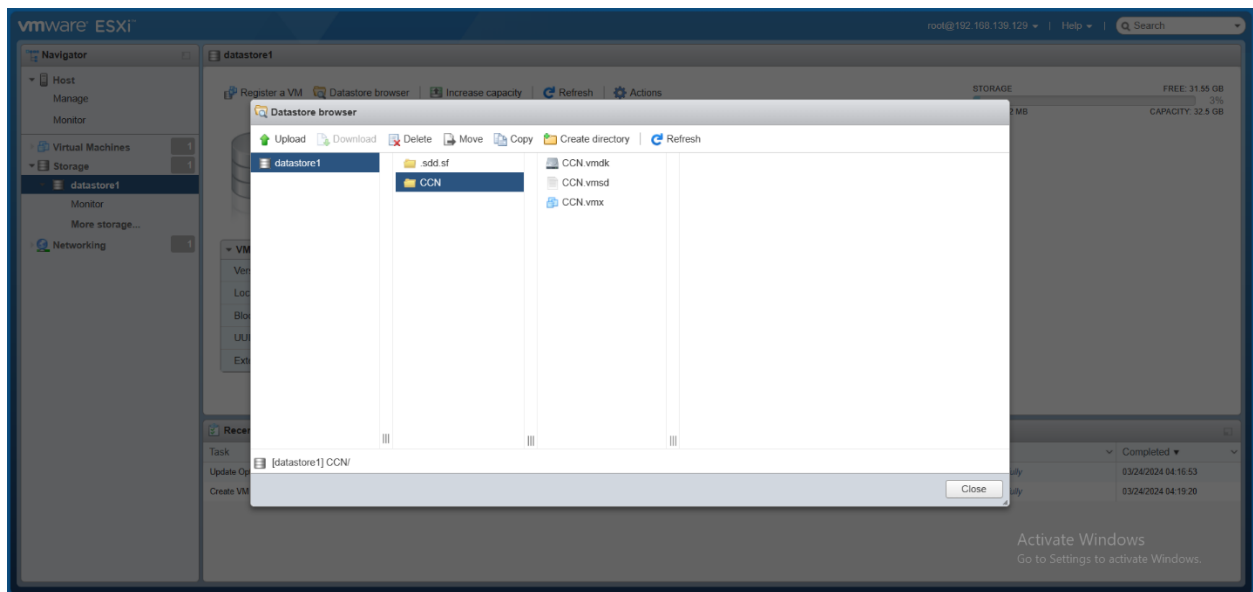
ESXi configuration



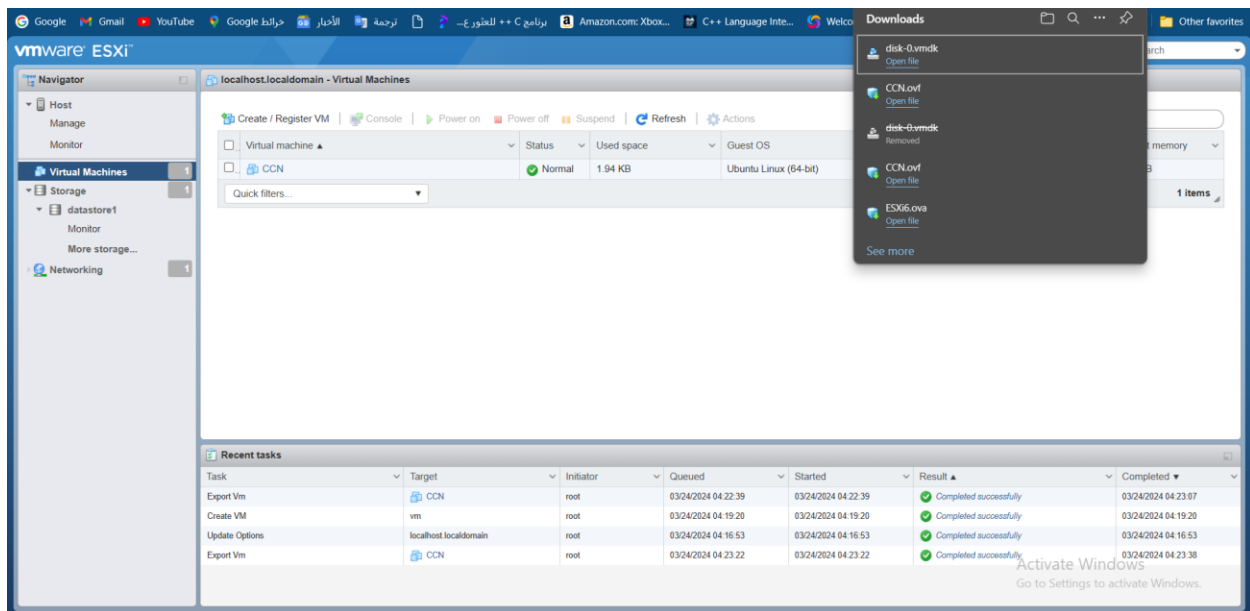
Remote connection to ESXi host



Create Empty VM on ESXi



Datastore



Export VM