Mohit Narang

2103124

Experiment 10

Aim: To study and implement container orchestration using Kubernetes

Theory:

Explain need of container orchestration tool

Container orchestration tools are essential in managing and scaling containerized applications effectively. Here's why they're needed:

1. Managing Complexity: As the number of containers increases, managing them manually becomes impractical. Orchestration tools provide a centralized platform to manage containerized applications, reducing the complexity of deployment, scaling, and maintenance.

2. Scaling: Container orchestration tools enable automatic scaling of containers based on application demand. They can dynamically allocate resources to ensure optimal performance without manual intervention.

3. High Availability: Orchestration tools ensure high availability by automatically restarting containers that fail or distributing application components across multiple nodes to prevent single points of failure.

4. Load Balancing: They provide built-in load balancing capabilities to distribute incoming traffic across multiple instances of an application, ensuring efficient resource utilization and improved performance.

5. Resource Utilization: Orchestration tools optimize resource utilization by efficiently scheduling containers on host machines, minimizing idle resources and maximizing utilization.

6. Service Discovery: They facilitate service discovery by automatically registering and tracking the IP addresses and ports of containers, allowing other services to easily locate and communicate with them.

7. Rolling Updates and Rollbacks: Orchestration tools enable seamless rolling updates and rollbacks of containerized applications, minimizing downtime and ensuring continuous delivery of new features or bug fixes.

8. Health Monitoring: They monitor the health of containers and nodes, automatically replacing or rescheduling unhealthy containers to maintain the desired state of the application.

9. Security: Orchestration tools offer features for securing containerized applications, such as network policies, secret management, and role-based access control (RBAC), helping to enforce security best practices.
10. Portability: They promote application portability by abstracting away infrastructure details, allowing applications to run consistently across different environments, including on-premises data centers and cloud platforms.

What is Kubernetes? Describe its features.

Kubernetes is an open-source container orchestration platform originally developed by Google, now maintained by the Cloud Native Computing Foundation (CNCF). It automates the deployment, scaling, and management of containerized applications. Below are some of its key features:

1. Automated Deployment and Scaling: Kubernetes automates the deployment of containerized applications, ensuring that the desired state of the application is maintained. It can automatically scale applications based on CPU or memory usage, or custom metrics defined by the user.
2. Service Discovery and Load Balancing: Kubernetes provides built-in service discovery and load balancing for containers. It assigns a unique IP address and DNS name to each service, enabling other services to discover and communicate with them. Load balancing ensures that incoming traffic is distributed evenly across multiple instances of an application.
3. Self-healing: Kubernetes monitors the health of containers and nodes in the cluster. If a container or node fails, Kubernetes automatically restarts the container on a healthy node to ensure that the desired state of the application is maintained. It can also replace containers that fail liveness probes or become unresponsive.
4. Rolling Updates and Rollbacks: Kubernetes supports rolling updates and rollbacks of containerized applications. It can gradually update or rollback application versions without downtime by incrementally updating or reverting containers.
5. Storage Orchestration: Kubernetes provides storage orchestration capabilities, allowing containers to mount persistent storage volumes. It supports various storage solutions, including local storage, network-attached storage (NAS), and cloud storage providers.
6. Secrets and Configuration Management: Kubernetes enables secure management of sensitive information, such as passwords, API tokens, and SSL certificates, using Kubernetes Secrets. It also supports configuration management through Confirms, which allow you to decouple configuration details from container images.

7. Batch Execution and Cron Jobs: Kubernetes supports batch execution of jobs and cron-like scheduled tasks. It allows you to run containerized batch workloads, such as data processing jobs or periodic tasks, reliably and efficiently.

8. Resource Management: Kubernetes provides resource management features to allocate compute resources, such as CPU and memory, to containers. It allows you to specify resource requests and limits for containers, ensuring fair resource allocation and preventing resource starvation.

9. Multi-tenancy: Kubernetes supports multi-tenancy by providing features like namespaces, which allow you to create isolated virtual clusters within a Kubernetes cluster. This enables teams to share a single Kubernetes cluster while maintaining isolation and resource segregation.

10. Extensibility and Ecosystem: Kubernetes has a rich ecosystem of plugins and extensions, allowing you to extend its functionality to meet specific requirements. It supports custom resource definitions (CRDs), which enable you to define custom resources and controllers to manage them.

Explain Kubernetes Components, its working and architecture.

Kubernetes is composed of several key components that work together to manage containerized applications efficiently. Below are the main components along with an explanation of their roles and how they work together:

1. Master Node Components:

    - API Server: The API server is the central management point for the Kubernetes cluster. It exposes the Kubernetes API, which clients use to interact with the cluster. All other components communicate with the API server to perform operations like deploying applications, scaling, and monitoring.

    - Scheduler: The scheduler is responsible for scheduling pods (a group of one or more containers) onto individual nodes in the cluster. It considers factors such as resource requirements, node capacity, and affinity/anti-affinity rules when making scheduling decisions.

    - Controller Manager: The controller manager runs various controllers that are responsible for maintaining the desired state of the cluster. These controllers include the Replication Controller, ReplicaSet Controller, Endpoints Controller, and others. They continuously monitor the cluster and take action to ensure that the actual state matches the desired state.

    - etcd: etcd is a distributed key-value store that serves as the cluster's persistent storage. It stores configuration data, state information, and metadata about the cluster, allowing Kubernetes components to retrieve and update information reliably.

2. Node Components:

    - Kubelet: The kubelet is an agent that runs on each node in the cluster and is responsible for managing containers on that node. It receives pod specifications from the API server and ensures that the containers described in those specifications are running and healthy.

    - Kube-proxy: Kube-proxy is a network proxy that runs on each node and implements Kubernetes services abstraction. It maintains network rules to enable communication with pods from outside the cluster and provides load balancing for services.

    - Container Runtime: The container runtime is responsible for running containers on each node. Kubernetes supports various container runtimes, including Docker, containerd, and CRI-O. The kubelet interacts with the container runtime to start, stop, and manage containers.

3. Networking:

    - Pod Network: Pods in a Kubernetes cluster communicate with each other over a pod network. Kubernetes does not mandate a specific networking solution but provides an interface (CNI) for network plugins to integrate with the cluster. Common pod network solutions include Calico, Flannel, and Weave.

    - Service Network: Kubernetes services have an IP address and are accessible internally within the cluster. The kube-proxy component ensures that requests to a service's IP address are properly load balanced and routed to the appropriate pods.

4. Add-ons:

    - Kubernetes clusters often include additional add-ons for monitoring, logging, and other purposes. These add-ons may include tools like Prometheus for monitoring, Fluentd or Elasticsearch for logging, and Grafana for visualization.

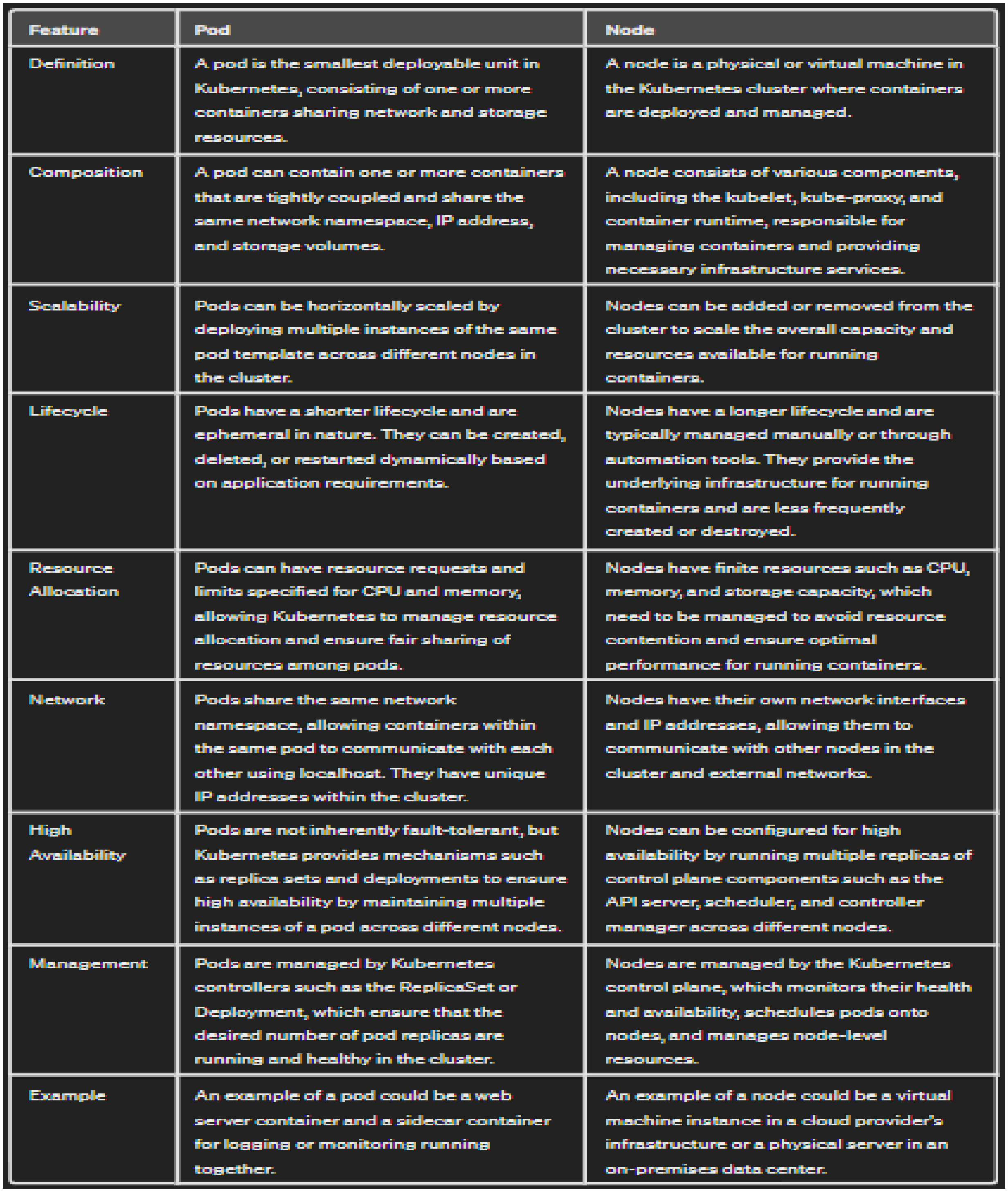
Architecture:

- Cluster: A Kubernetes cluster consists of one or more master nodes and multiple worker nodes.

- Master-Worker Architecture: The master nodes are responsible for managing the cluster, while the worker nodes host the running applications.

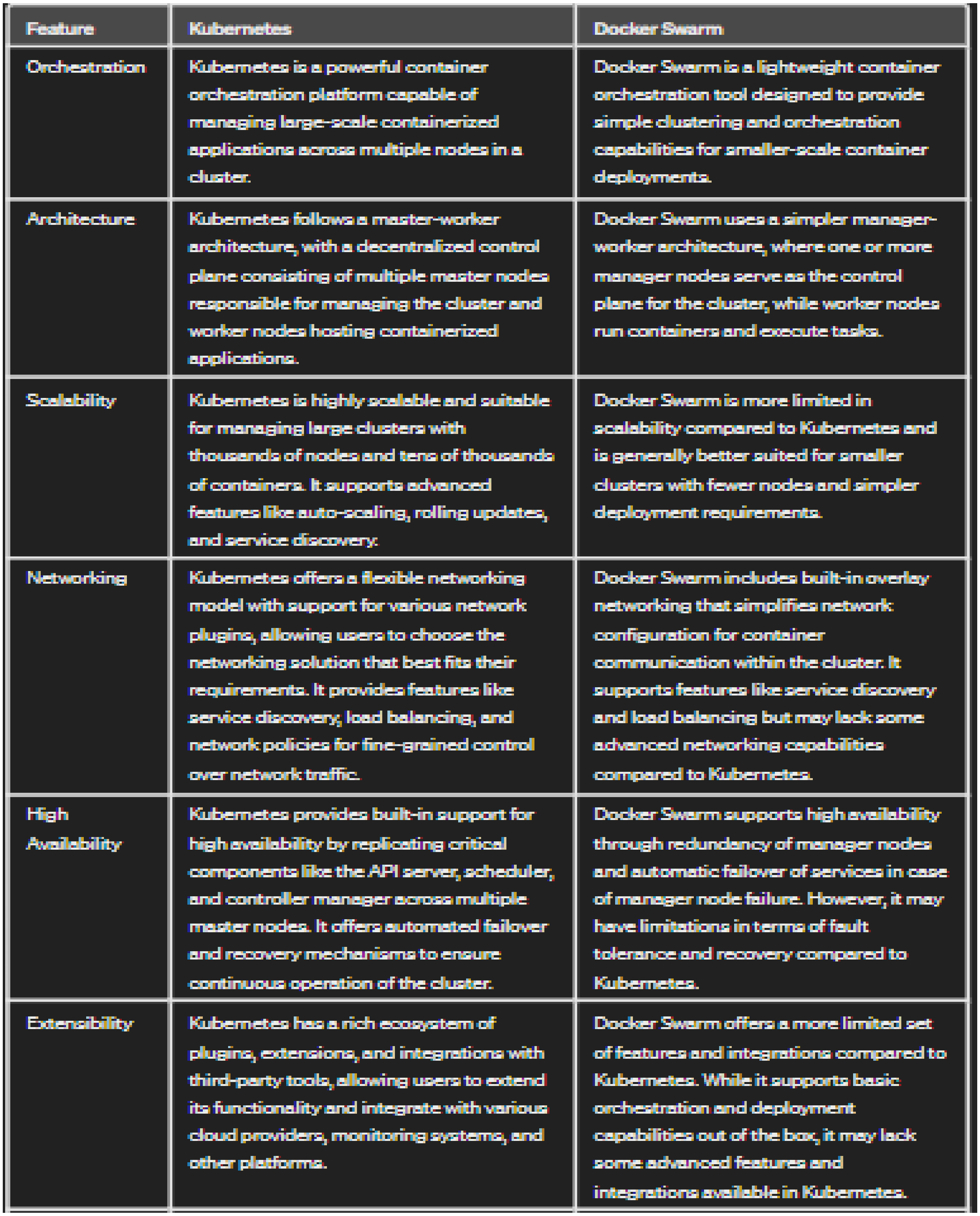
- High Availability: To achieve high availability, Kubernetes master components can be replicated and distributed across multiple nodes, and worker nodes can be added or removed dynamically as needed.
- Decentralized Control Plane: Kubernetes follows a decentralized control plane architecture, where each master component is responsible for a specific aspect of cluster management. This architecture ensures scalability and fault tolerance.

Difference between POD and node?



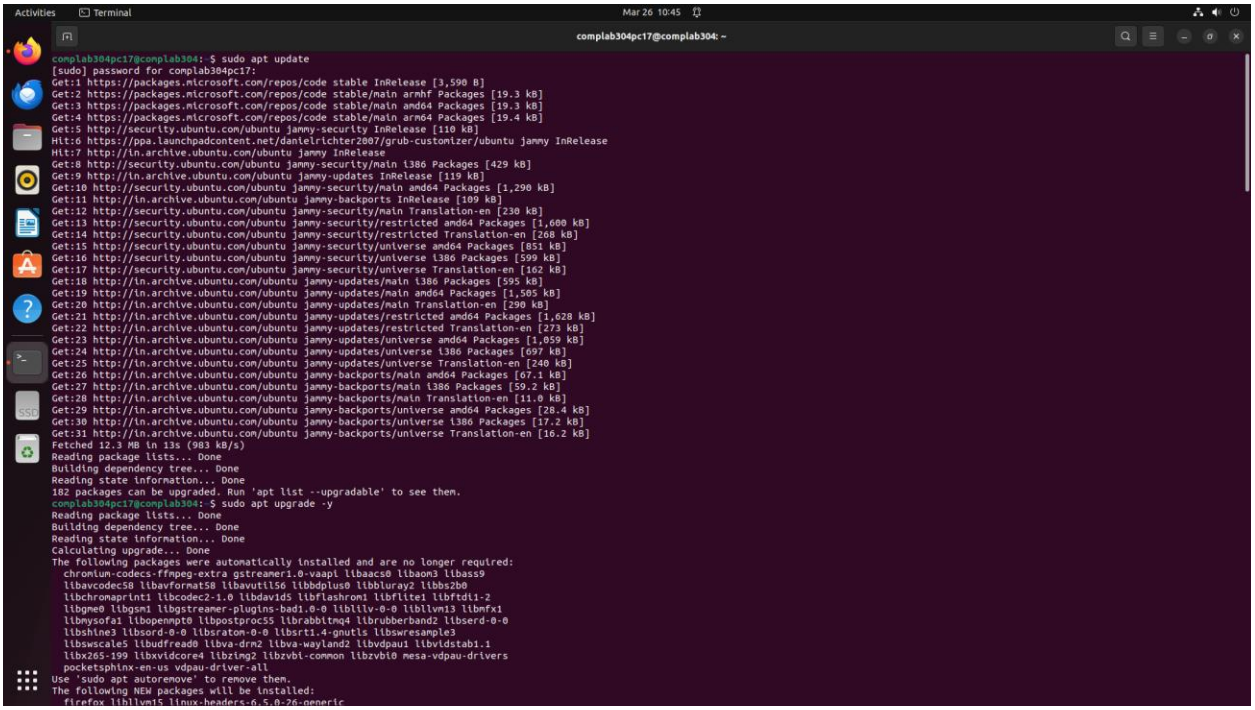
Feature	Pod	Node
Definition	A pod is the smallest deployable unit in Kubernetes, consisting of one or more containers sharing network and storage resources.	A node is a physical or virtual machine in the Kubernetes cluster where containers are deployed and managed.
Composition	A pod can contain one or more containers that are tightly coupled and share the same network namespace, IP address, and storage volumes.	A node consists of various components, including the kubelet, kube-proxy, and container runtime, responsible for managing containers and providing necessary infrastructure services.
Scalability	Pods can be horizontally scaled by deploying multiple instances of the same pod template across different nodes in the cluster.	Nodes can be added or removed from the cluster to scale the overall capacity and resources available for running containers.
Lifecycle	Pods have a shorter lifecycle and are ephemeral in nature. They can be created, deleted, or restarted dynamically based on application requirements.	Nodes have a longer lifecycle and are typically managed manually or through automation tools. They provide the underlying infrastructure for running containers and are less frequently created or destroyed.
Resource Allocation	Pods can have resource requests and limits specified for CPU and memory, allowing Kubernetes to manage resource allocation and ensure fair sharing of resources among pods.	Nodes have finite resources such as CPU, memory, and storage capacity, which need to be managed to avoid resource contention and ensure optimal performance for running containers.
Network	Pods share the same network namespace, allowing containers within the same pod to communicate with each other using localhost. They have unique IP addresses within the cluster.	Nodes have their own network interfaces and IP addresses, allowing them to communicate with other nodes in the cluster and external networks.
High Availability	Pods are not inherently fault-tolerant, but Kubernetes provides mechanisms such as replica sets and deployments to ensure high availability by maintaining multiple instances of a pod across different nodes.	Nodes can be configured for high availability by running multiple replicas of control plane components such as the API server, scheduler, and controller manager across different nodes.
Management	Pods are managed by Kubernetes controllers such as the ReplicaSet or Deployment, which ensure that the desired number of pod replicas are running and healthy in the cluster.	Nodes are managed by the Kubernetes control plane, which monitors their health and availability, schedules pods onto nodes, and manages node-level resources.
Example	An example of a pod could be a web server container and a sidecar container for logging or monitoring running together.	An example of a node could be a virtual machine instance in a cloud provider's infrastructure or a physical server in an on-premises data center.

Compare Kubernetes and Docker Swarm

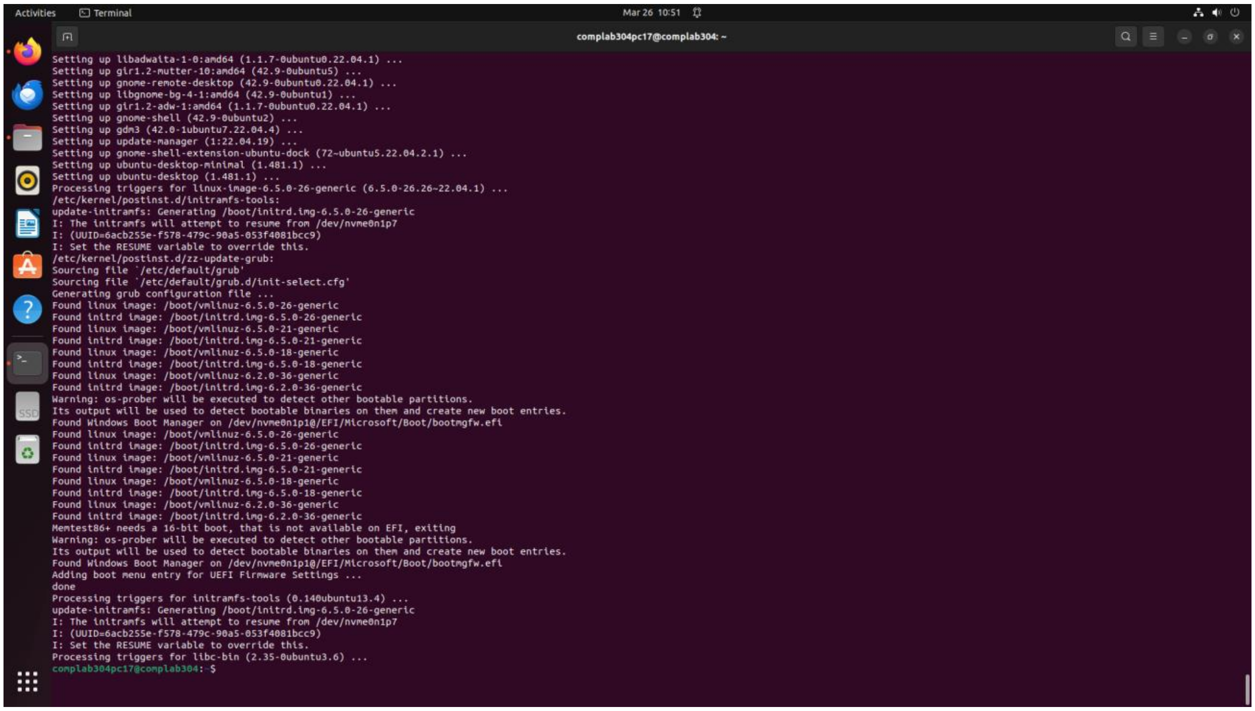


Feature	Kubernetes	Docker Swarm
Orchestration	Kubernetes is a powerful container orchestration platform capable of managing large-scale containerized applications across multiple nodes in a cluster.	Docker Swarm is a lightweight container orchestration tool designed to provide simple clustering and orchestration capabilities for smaller-scale container deployments.
Architecture	Kubernetes follows a master-worker architecture, with a decentralized control plane consisting of multiple master nodes responsible for managing the cluster and worker nodes hosting containerized applications.	Docker Swarm uses a simpler manager-worker architecture, where one or more manager nodes serve as the control plane for the cluster, while worker nodes run containers and execute tasks.
Scalability	Kubernetes is highly scalable and suitable for managing large clusters with thousands of nodes and tens of thousands of containers. It supports advanced features like auto-scaling, rolling updates, and service discovery.	Docker Swarm is more limited in scalability compared to Kubernetes and is generally better suited for smaller clusters with fewer nodes and simpler deployment requirements.
Networking	Kubernetes offers a flexible networking model with support for various network plugins, allowing users to choose the networking solution that best fits their requirements. It provides features like service discovery, load balancing, and network policies for fine-grained control over network traffic.	Docker Swarm includes built-in overlay networking that simplifies network configuration for container communication within the cluster. It supports features like service discovery and load balancing but may lack some advanced networking capabilities compared to Kubernetes.
High Availability	Kubernetes provides built-in support for high availability by replicating critical components like the API server, scheduler, and controller manager across multiple master nodes. It offers automated failover and recovery mechanisms to ensure continuous operation of the cluster.	Docker Swarm supports high availability through redundancy of manager nodes and automatic failover of services in case of manager node failure. However, it may have limitations in terms of fault tolerance and recovery compared to Kubernetes.
Extensibility	Kubernetes has a rich ecosystem of plugins, extensions, and integrations with third-party tools, allowing users to extend its functionality and integrate with various cloud providers, monitoring systems, and other platforms.	Docker Swarm offers a more limited set of features and integrations compared to Kubernetes. While it supports basic orchestration and deployment capabilities out of the box, it may lack some advanced features and integrations available in Kubernetes.

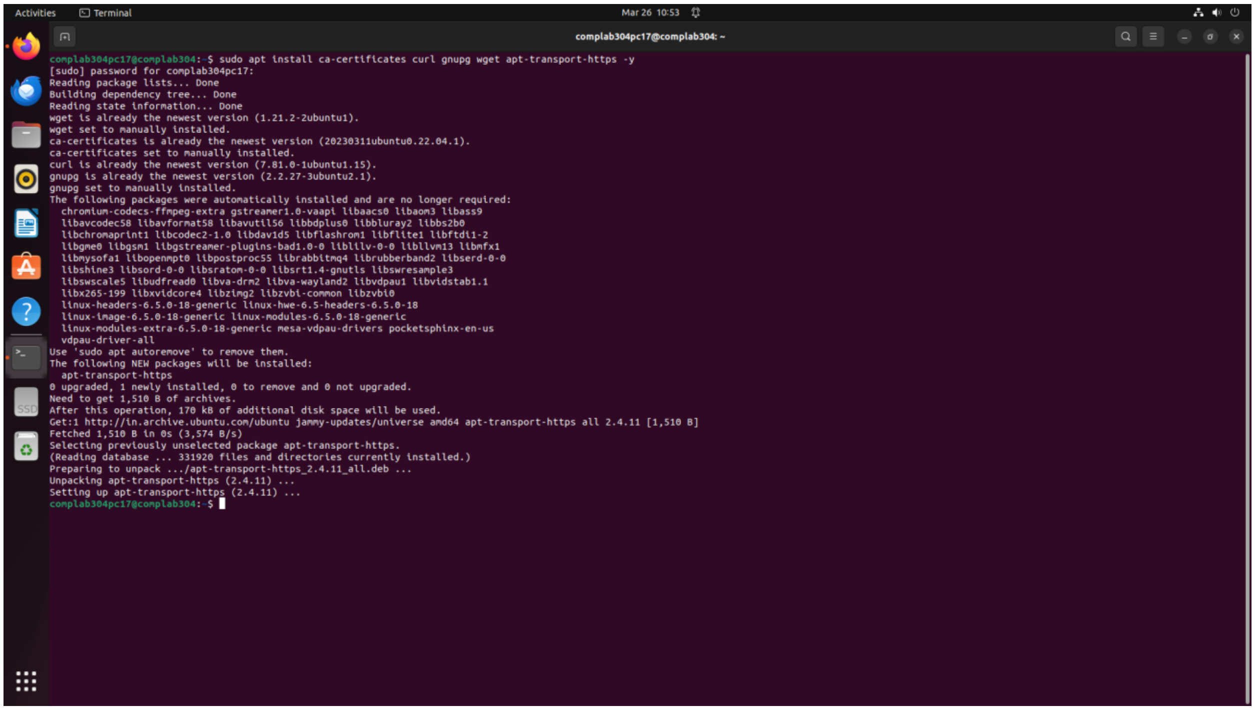
Steps t be followed



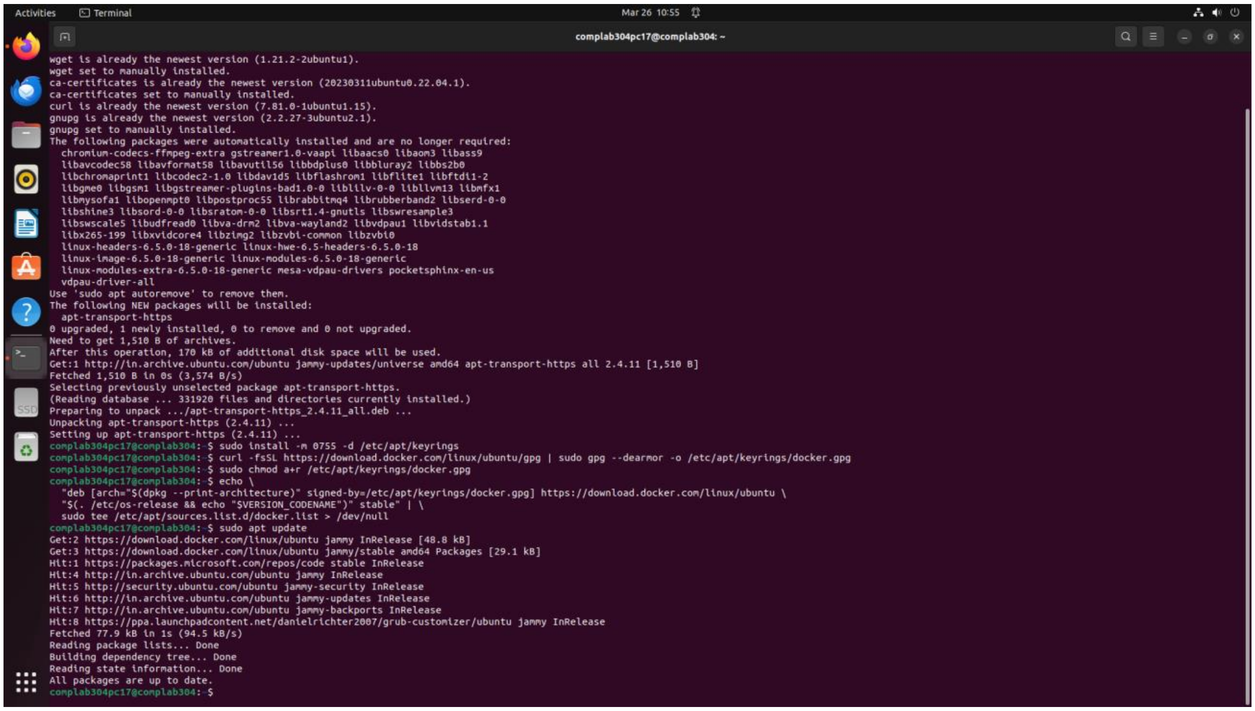
```
complab304pc17@complab304:~$ sudo apt update
[sudo] password for complab304pc17:
Get:1 https://packages.microsoft.com/repos/code stable InRelease [3,590 B]
Get:2 https://packages.microsoft.com/repos/code stable/main armhf Packages [19.3 kB]
Get:3 https://packages.microsoft.com/repos/code stable/main amd64 Packages [19.3 kB]
Get:4 https://packages.microsoft.com/repos/code stable/main arm64 Packages [19.4 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:6 https://ppa.launchpadcontent.net/danielrichter2007/grub-customizer/ubuntu jammy InRelease
Hit:7 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [429 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1,290 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [230 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1,600 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [268 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [851 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/universe i386 Packages [599 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [162 kB]
Get:18 http://in.archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [595 kB]
Get:19 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1,505 kB]
Get:20 http://in.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [290 kB]
Get:21 http://in.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1,628 kB]
Get:22 http://in.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [273 kB]
Get:23 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1,059 kB]
Get:24 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages [697 kB]
Get:25 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [240 kB]
Get:26 http://in.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.1 kB]
Get:27 http://in.archive.ubuntu.com/ubuntu jammy-backports/main i386 Packages [59.2 kB]
Get:28 http://in.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.0 kB]
Get:29 http://in.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28.4 kB]
Get:30 http://in.archive.ubuntu.com/ubuntu jammy-backports/universe i386 Packages [17.2 kB]
Get:31 http://in.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.2 kB]
Fetched 12.3 MB in 13s (983 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
182 packages can be upgraded. Run 'apt list --upgradable' to see them.
complab304pc17@complab304:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libaacs0 libaom3 libass9
  libavcodec58 libavformat58 libavutil56 libbdplus0 libbluray2 libbs2b0
  libchromaprint1 libcodec2-1.0 libdav1d5 libflashrom1 libflite1 libftdi1-2
  libgme0 libgsm1 libgstreamer-plugins-bad1.0-0 liblilv-0-0 libllvm13 libmfx1
  libmysofa1 libopenmpt0 libpostproc55 librabbitmq4 librubberband2 libserd-0-0
  libshine3 libsord-0-0 libsratom-0-0 libsrt1.4-gnutls libswresample3
  libswscale5 libudfread0 libva-drm2 libva-wayland2 libvdpau1 libvidstab1.1
  libx265-199 libxvidcore4 libzimg2 libzvbi-common libzvbi0 mesa-vdpau-drivers
  pocketsphinx-en-us vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  firefox libllvm15 linux-headers-6.5.0-26-generic
```



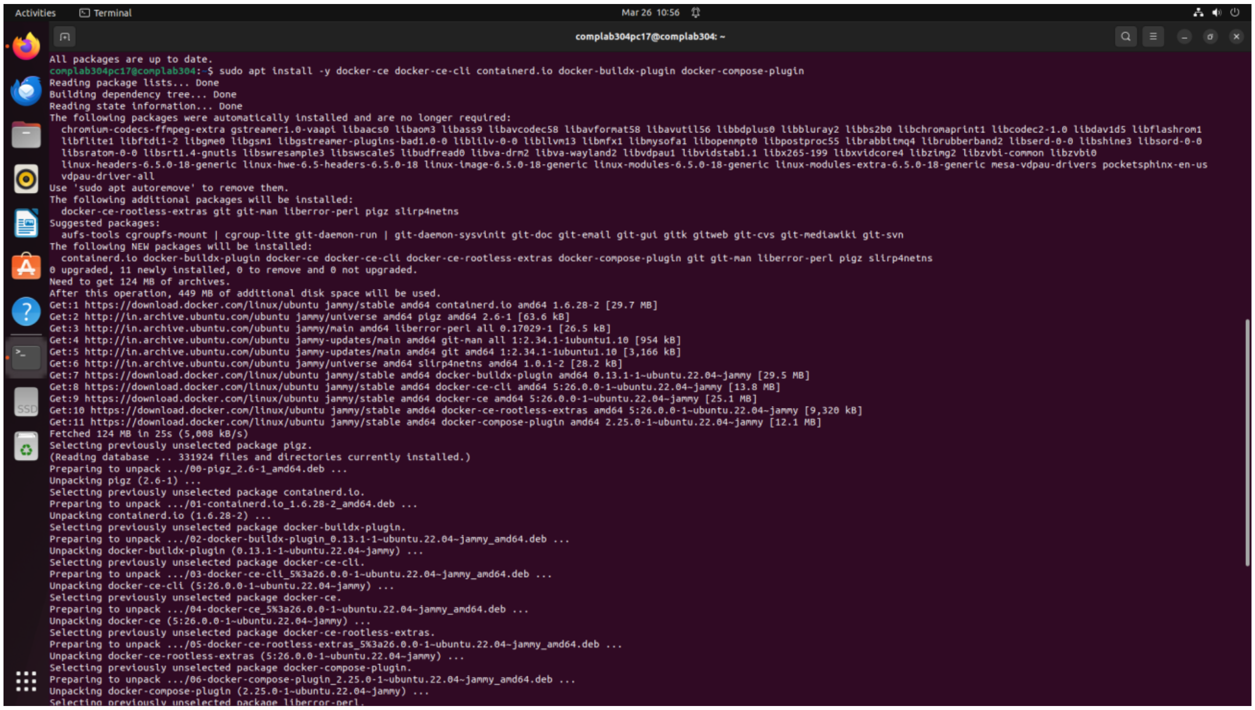
```
Setting up libadwaita-1-0:amd64 (1.1.7-0ubuntu0.22.04.1) ...
Setting up gir1.2-mutter-10:amd64 (42.9-0ubuntu5) ...
Setting up gnome-remote-desktop (42.9-0ubuntu0.22.04.1) ...
Setting up libgnome-bg-4-1:amd64 (42.9-0ubuntu1) ...
Setting up gir1.2-adw-1:amd64 (1.1.7-0ubuntu0.22.04.1) ...
Setting up gnome-shell (42.9-0ubuntu2) ...
Setting up gdm3 (42.0-1ubuntu7.22.04.4) ...
Setting up update-manager (1:22.04.19) ...
Setting up gnome-shell-extension-ubuntu-dock (72~ubuntu5.22.04.2.1) ...
Setting up ubuntu-desktop-minimal (1.481.1) ...
Setting up ubuntu-desktop (1.481.1) ...
Processing triggers for linux-image-6.5.0-26-generic (6.5.0-26.26~22.04.1) ...
/etc/kernel/postinst.d/initramfs-tools:
update-initramfs: Generating /boot/initrd.img-6.5.0-26-generic
I: The initramfs will attempt to resume from /dev/nvme0n1p7
I: (UUID=6acb255e-f578-479c-90a5-053f4081bcc9)
I: Set the RESUME variable to override this.
/etc/kernel/postinst.d/zz-update-grub:
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.5.0-26-generic
Found initrd image: /boot/initrd.img-6.5.0-26-generic
Found linux image: /boot/vmlinuz-6.5.0-21-generic
Found initrd image: /boot/initrd.img-6.5.0-21-generic
Found linux image: /boot/vmlinuz-6.5.0-18-generic
Found initrd image: /boot/initrd.img-6.5.0-18-generic
Found linux image: /boot/vmlinuz-6.2.0-36-generic
Found initrd image: /boot/initrd.img-6.2.0-36-generic
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows Boot Manager on /dev/nvme0n1p1@/EFI/Microsoft/Boot/bootmgfw.efi
Found linux image: /boot/vmlinuz-6.5.0-26-generic
Found initrd image: /boot/initrd.img-6.5.0-26-generic
Found linux image: /boot/vmlinuz-6.5.0-21-generic
Found initrd image: /boot/initrd.img-6.5.0-21-generic
Found linux image: /boot/vmlinuz-6.5.0-18-generic
Found initrd image: /boot/initrd.img-6.5.0-18-generic
Found linux image: /boot/vmlinuz-6.2.0-36-generic
Found initrd image: /boot/initrd.img-6.2.0-36-generic
Memtest86+ needs a 16-bit boot, that is not available on EFI, exiting
Warning: os-prober will be executed to detect other bootable partitions.
Its output will be used to detect bootable binaries on them and create new boot entries.
Found Windows Boot Manager on /dev/nvme0n1p1@/EFI/Microsoft/Boot/bootmgfw.efi
Adding boot menu entry for UEFI Firmware Settings ...
done
Processing triggers for initramfs-tools (0.140ubuntu13.4) ...
update-initramfs: Generating /boot/initrd.img-6.5.0-26-generic
I: The initramfs will attempt to resume from /dev/nvme0n1p7
I: (UUID=6acb255e-f578-479c-90a5-053f4081bcc9)
I: Set the RESUME variable to override this.
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
complab304pc17@complab304:~$
```



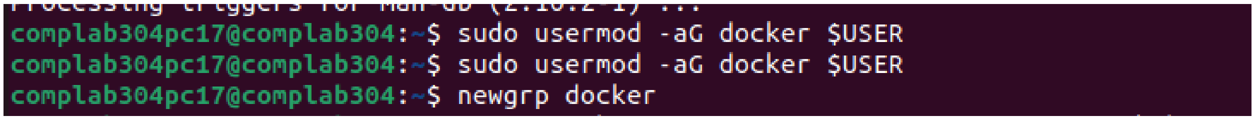
```
complab304pc17@complab304:~$ sudo apt install ca-certificates curl gnupg wget apt-transport-https -y
[sudo] password for complab304pc17:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
wget is already the newest version (1.21.2-2ubuntu1).
wget set to manually installed.
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.15).
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libaacs0 libaom3 libass9
  libavcodec58 libavformat58 libavutil56 libbdplus0 libbluray2 libbs2b0
  libchromaprint1 libcodec2-1.0 libdav1d5 libflashrom1 libflite1 libftdi1-2
  libgme0 libgsm1 libgstreamer-plugins-bad1.0-0 liblilv-0-0 libllvm13 libmfx1
  libmysofa1 libopenmpt0 libpostproc55 librabbitmq4 librubberband2 libserd-0-0
  libshine3 libsord-0-0 libsratom-0-0 libsrt1.4-gnutls libswresample3
  libswscale5 libudfread0 libva-drm2 libva-wayland2 libvdpau1 libvidstab1.1
  libx265-199 libxvidcore4 libzimg2 libzvbi-common libzvbi0
  linux-headers-6.5.0-18-generic linux-hwe-6.5-headers-6.5.0-18
  linux-image-6.5.0-18-generic linux-modules-6.5.0-18-generic
  linux-modules-extra-6.5.0-18-generic mesa-vdpau-drivers pocketsphinx-en-us
  vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1,510 B of archives.
After this operation, 170 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.11 [1,510 B]
Fetched 1,510 B in 0s (3,574 B/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 331920 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.11_all.deb ...
Unpacking apt-transport-https (2.4.11) ...
Setting up apt-transport-https (2.4.11) ...
complab304pc17@complab304:~$ 
```



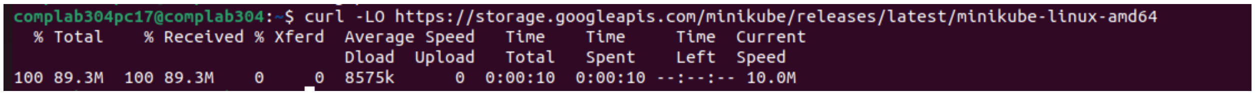
```
wget is already the newest version (1.21.2-2ubuntu1).
wget set to manually installed.
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.15).
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libaacs0 libaom3 libass9
  libavcodec58 libavformat58 libavutil56 libbdplus0 libbluray2 libbs2b0
  libchromaprint1 libcodec2-1.0 libdav1d5 libflashrom1 libflite1 libftdi1-2
  libgme0 libgsm1 libgstreamer-plugins-bad1.0-0 liblilv-0-0 libllvm13 libmfx1
  libmysofa1 libopenmpt0 libpostproc55 librabbitmq4 librubberband2 libserd-0-0
  libshine3 libsord-0-0 libsratom-0-0 libsrt1.4-gnutls libswresample3
  libswscale5 libudfread0 libva-drm2 libva-wayland2 libvdpau1 libvidstab1.1
  libx265-199 libxvidcore4 libzimg2 libzvbi-common libzvbi0
  linux-headers-6.5.0-18-generic linux-hwe-6.5-headers-6.5.0-18
  linux-image-6.5.0-18-generic linux-modules-6.5.0-18-generic
  linux-modules-extra-6.5.0-18-generic mesa-vdpau-drivers pocketsphinx-en-us
  vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1,510 B of archives.
After this operation, 170 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.11 [1,510 B]
Fetched 1,510 B in 0s (3,574 B/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 331920 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.11_all.deb ...
Unpacking apt-transport-https (2.4.11) ...
Setting up apt-transport-https (2.4.11) ...
complab304pc17@complab304:~$ sudo install -m 0755 -d /etc/apt/keyrings
complab304pc17@complab304:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
complab304pc17@complab304:~$ sudo chmod a+r /etc/apt/keyrings/docker.gpg
complab304pc17@complab304:~$ echo \
  "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
  "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
complab304pc17@complab304:~$ sudo apt update
Get:2 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:3 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [29.1 kB]
Hit:1 https://packages.microsoft.com/repos/code stable InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:6 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:7 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:8 https://ppa.launchpadcontent.net/danielrichter2007/grub-customizer/ubuntu jammy InRelease
Fetched 77.9 kB in 1s (94.5 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
complab304pc17@complab304:~$
```



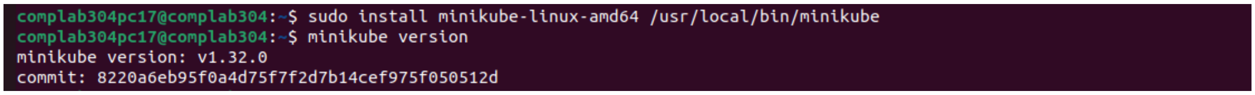
```
All packages are up to date.
complab304pc17@complab304:~$ sudo apt install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi libaacs0 libaom3 libass9 libavcodec58 libavformat58 libavutil56 libbdplus0 libbluray2 libbs2b0 libchromaprint1 libcodec2-1.0 libdav1d5 libflashrom1
  libflite1 libftdi1-2 libgme0 libgsm1 libgstreamer-plugins-bad1.0-0 liblilv-0-0 libllvm13 libmfx1 libmysofa1 libopenmpt0 libpostproc55 librabbitmq4 librubberband2 libserd-0-0 libshine3 libsord-0-0
  libsratom-0-0 libsrt1.4-gnutls libswresample3 libswscale5 libudfread0 libva-drm2 libva-wayland2 libvdpau1 libvidstab1.1 libx265-199 libxvidcore4 libzimg2 libzvbi-common libzvbi0
  linux-headers-6.5.0-18-generic linux-hwe-6.5-headers-6.5.0-18 linux-image-6.5.0-18-generic linux-modules-6.5.0-18-generic linux-modules-extra-6.5.0-18-generic mesa-vdpau-drivers pocketsphinx-en-us
  vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  docker-ce-rootless-extras git git-man liberror-perl pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin git git-man liberror-perl pigz slirp4netns
0 upgraded, 11 newly installed, 0 to remove and 0 not upgraded.
Need to get 124 MB of archives.
After this operation, 449 MB of additional disk space will be used.
Get:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.28-2 [29.7 MB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.10 [954 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.10 [3,166 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 slirp4netns amd64 1.0.1-2 [28.2 kB]
Get:7 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-buildx-plugin amd64 0.13.1-1~ubuntu.22.04~jammy [29.5 MB]
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-cli amd64 5:26.0.0-1~ubuntu.22.04~jammy [13.8 MB]
Get:9 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce amd64 5:26.0.0-1~ubuntu.22.04~jammy [25.1 MB]
Get:10 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-rootless-extras amd64 5:26.0.0-1~ubuntu.22.04~jammy [9,320 kB]
Get:11 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-compose-plugin amd64 2.25.0-1~ubuntu.22.04~jammy [12.1 MB]
Fetched 124 MB in 25s (5,008 kB/s)
Selecting previously unselected package pigz.
(Reading database ... 331924 files and directories currently installed.)
Preparing to unpack .../00-pigz_2.6-1_amd64.deb ...
Unpacking pigz (2.6-1) ...
Selecting previously unselected package containerd.io.
Preparing to unpack .../01-containerd.io_1.6.28-2_amd64.deb ...
Unpacking containerd.io (1.6.28-2) ...
Selecting previously unselected package docker-buildx-plugin.
Preparing to unpack .../02-docker-buildx-plugin_0.13.1-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-buildx-plugin (0.13.1-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../03-docker-ce-cli_5%3a26.0.0-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-ce-cli (5:26.0.0-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../04-docker-ce_5%3a26.0.0-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-ce (5:26.0.0-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-ce-rootless-extras.
Preparing to unpack .../05-docker-ce-rootless-extras_5%3a26.0.0-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-ce-rootless-extras (5:26.0.0-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package docker-compose-plugin.
Preparing to unpack .../06-docker-compose-plugin_2.25.0-1~ubuntu.22.04~jammy_amd64.deb ...
Unpacking docker-compose-plugin (2.25.0-1~ubuntu.22.04~jammy) ...
Selecting previously unselected package liberror-perl.
```



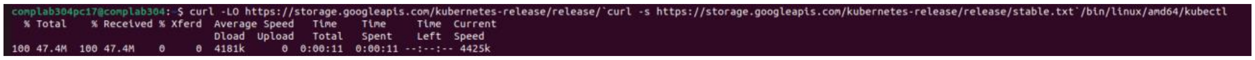
```
Processing triggers for man-db (2.10.2-1) ...
complab304pc17@complab304:~$ sudo usermod -aG docker $USER
complab304pc17@complab304:~$ sudo usermod -aG docker $USER
complab304pc17@complab304:~$ newgrp docker
```



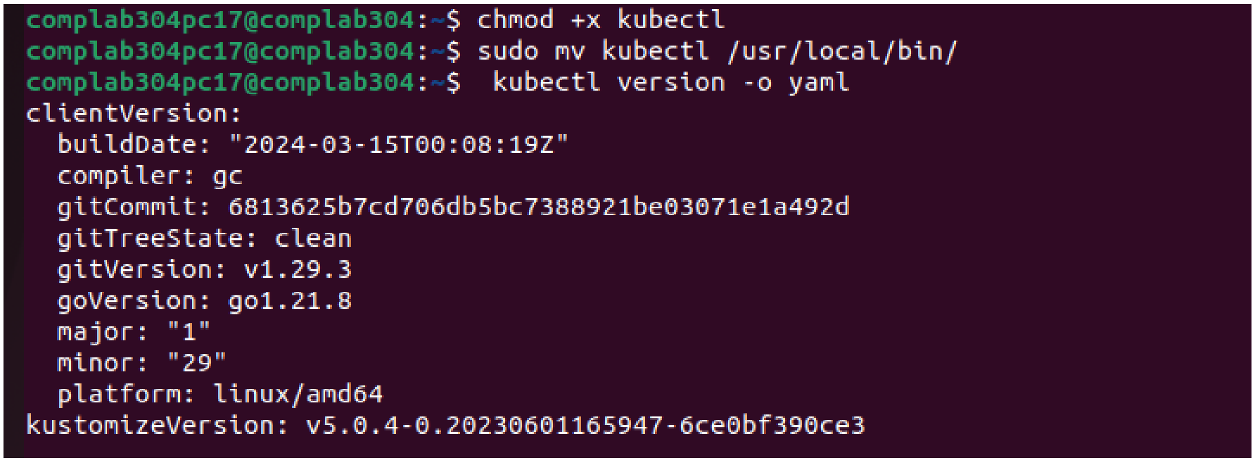
```
complab304pc17@complab304:~$ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 89.3M  100 89.3M    0     0  8575k      0  0:00:10  0:00:10 --:--:-- 10.0M
```



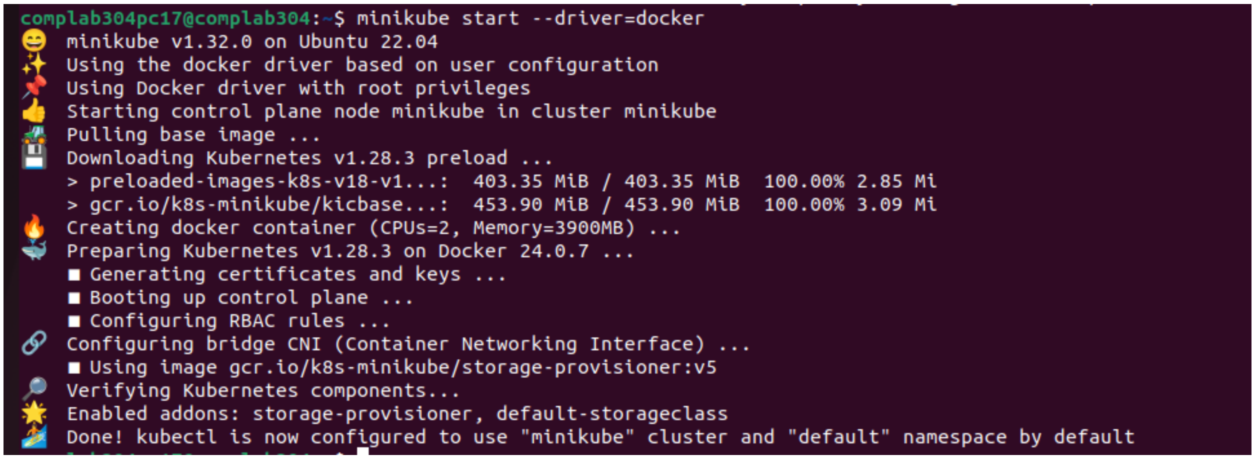
```
complab304pc17@complab304:~$ sudo install minikube-linux-amd64 /usr/local/bin/minikube
complab304pc17@complab304:~$ minikube version
minikube version: v1.32.0
commit: 8220a6eb95f0a4d75f7f2d7b14cef975f050512d
```



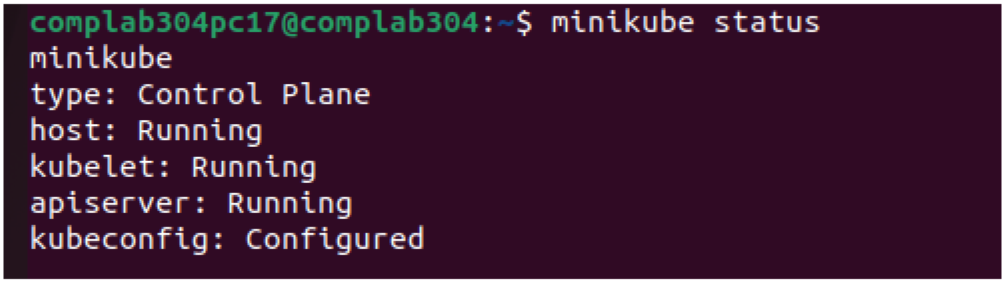
```
complab304pc17@complab304:~$ curl -LO https://storage.googleapis.com/kubernetes-release/release/`curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt`/bin/linux/amd64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 47.4M  100 47.4M    0     0  4181k      0  0:00:11  0:00:11 --:--:-- 4425k
```



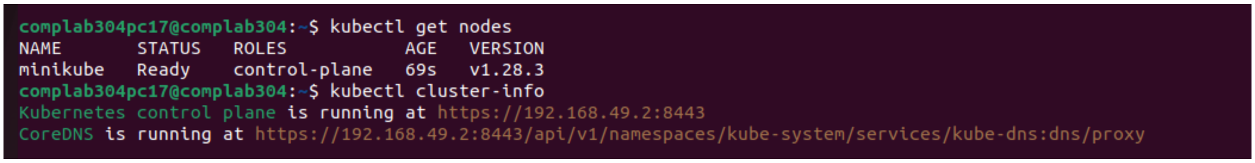
```
complab304pc17@complab304:~$ chmod +x kubectl
complab304pc17@complab304:~$ sudo mv kubectl /usr/local/bin/
complab304pc17@complab304:~$  kubectl version -o yaml
clientVersion:
  buildDate: "2024-03-15T00:08:19Z"
  compiler: gc
  gitCommit: 6813625b7cd706db5bc7388921be03071e1a492d
  gitTreeState: clean
  gitVersion: v1.29.3
  goVersion: go1.21.8
  major: "1"
  minor: "29"
  platform: linux/amd64
kustomizeVersion: v5.0.4-0.20230601165947-6ce0bf390ce3
```



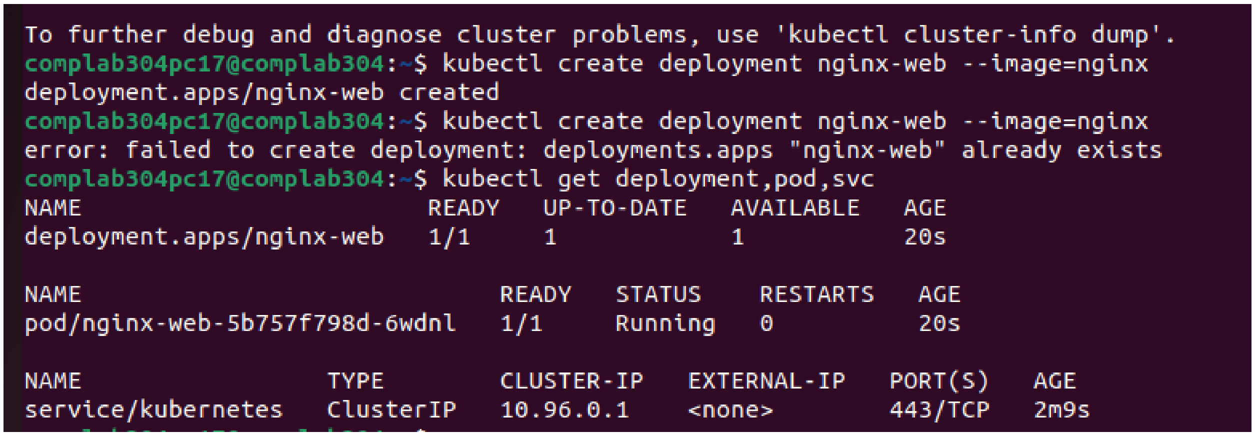
```
complab304pc17@complab304:~$ minikube start --driver=docker
😄  minikube v1.32.0 on Ubuntu 22.04
✨  Using the docker driver based on user configuration
📌  Using Docker driver with root privileges
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
💾  Downloading Kubernetes v1.28.3 preload ...
    > preloaded-images-k8s-v18-v1...:  403.35 MiB / 403.35 MiB  100.00% 2.85 Mi
    > gcr.io/k8s-minikube/kicbase...:  453.90 MiB / 453.90 MiB  100.00% 3.09 Mi
🔥  Creating docker container (CPUs=2, Memory=3900MB) ...
🐳  Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
    ■ Generating certificates and keys ...
    ■ Booting up control plane ...
    ■ Configuring RBAC rules ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
    ■ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🔎  Verifying Kubernetes components...
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```



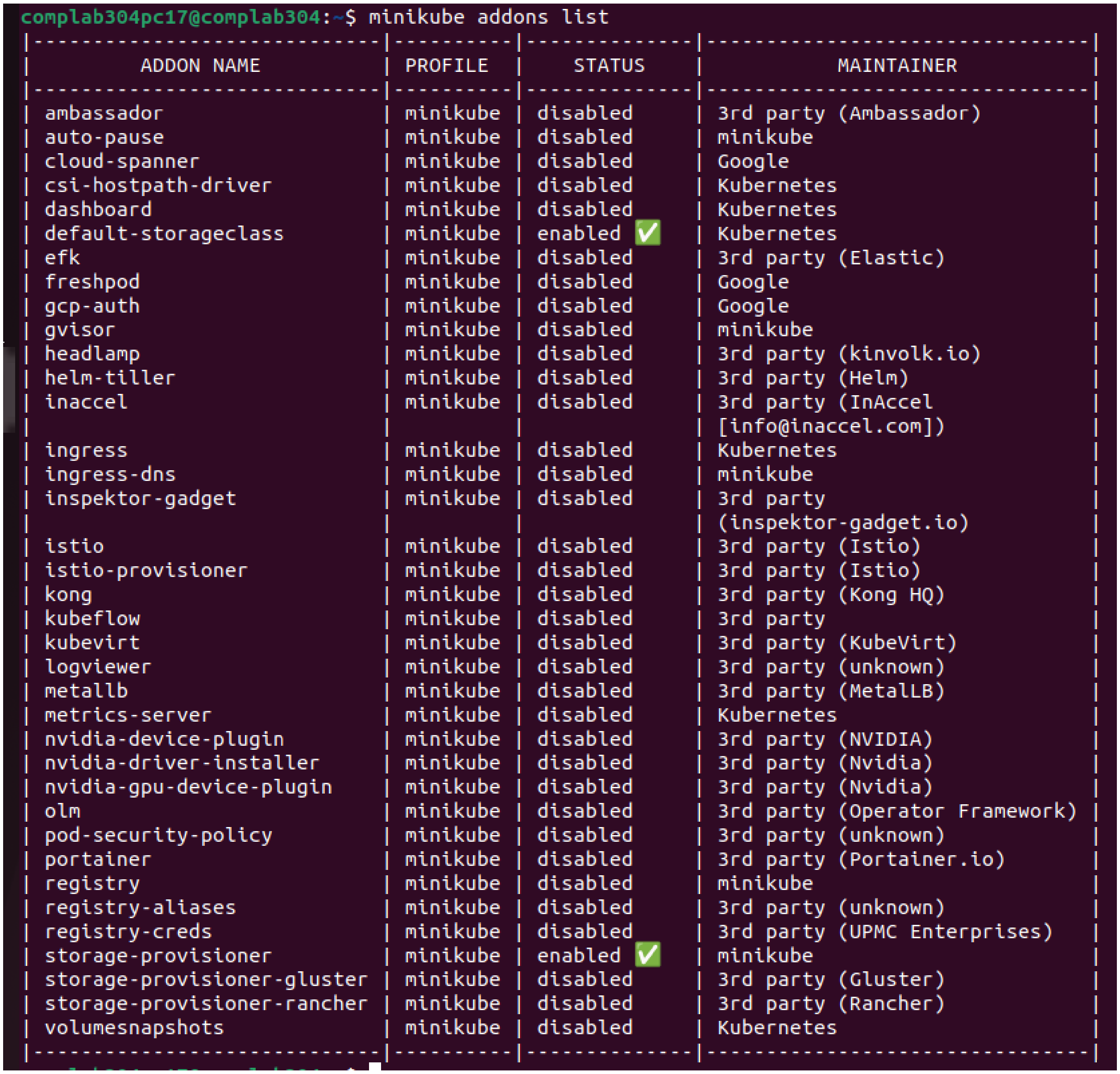
```
complab304pc17@complab304:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```



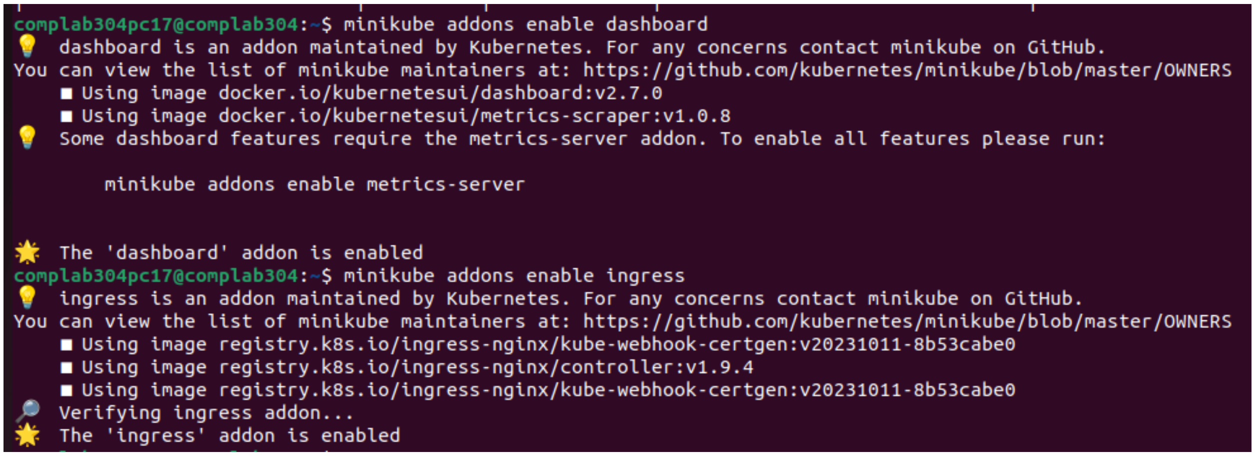
```
complab304pc17@complab304:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   69s   v1.28.3
complab304pc17@complab304:~$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy
```



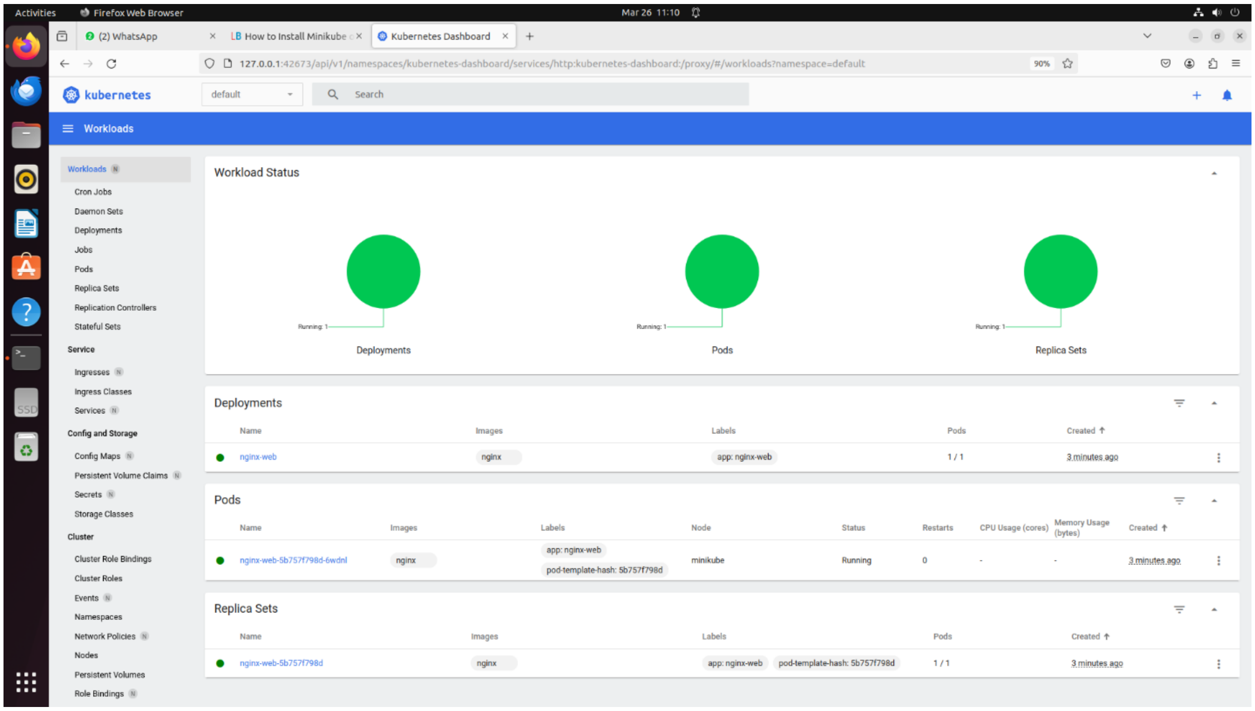
```
To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
complab304pc17@complab304:~$ kubectl create deployment nginx-web --image=nginx
deployment.apps/nginx-web created
complab304pc17@complab304:~$ kubectl create deployment nginx-web --image=nginx
error: failed to create deployment: deployments.apps "nginx-web" already exists
complab304pc17@complab304:~$ kubectl get deployment,pod,svc
NAME                        READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-web   1/1     1            1           20s

NAME                             READY   STATUS    RESTARTS   AGE
pod/nginx-web-5b757f798d-6wdnl   1/1     Running   0          20s

NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   2m9s
```



```
complab304pc17@complab304:~$ minikube addons list
|-----------------------------|----------|--------------|--------------------------------|
|         ADDON NAME          | PROFILE  |    STATUS    |           MAINTAINER           |
|-----------------------------|----------|--------------|--------------------------------|
| ambassador                  | minikube | disabled     | 3rd party (Ambassador)         |
| auto-pause                  | minikube | disabled     | minikube                       |
| cloud-spanner               | minikube | disabled     | Google                         |
| csi-hostpath-driver         | minikube | disabled     | Kubernetes                     |
| dashboard                   | minikube | disabled     | Kubernetes                     |
| default-storageclass        | minikube | enabled ✅   | Kubernetes                     |
| efk                         | minikube | disabled     | 3rd party (Elastic)            |
| freshpod                    | minikube | disabled     | Google                         |
| gcp-auth                    | minikube | disabled     | Google                         |
| gvisor                      | minikube | disabled     | minikube                       |
| headlamp                    | minikube | disabled     | 3rd party (kinvolk.io)         |
| helm-tiller                 | minikube | disabled     | 3rd party (Helm)               |
| inaccel                     | minikube | disabled     | 3rd party (InAccel             |
|                             |          |              | [info@inaccel.com])            |
| ingress                     | minikube | disabled     | Kubernetes                     |
| ingress-dns                 | minikube | disabled     | minikube                       |
| inspektor-gadget            | minikube | disabled     | 3rd party                      |
|                             |          |              | (inspektor-gadget.io)          |
| istio                       | minikube | disabled     | 3rd party (Istio)              |
| istio-provisioner           | minikube | disabled     | 3rd party (Istio)              |
| kong                        | minikube | disabled     | 3rd party (Kong HQ)            |
| kubeflow                    | minikube | disabled     | 3rd party                      |
| kubevirt                    | minikube | disabled     | 3rd party (KubeVirt)           |
| logviewer                   | minikube | disabled     | 3rd party (unknown)            |
| metallb                     | minikube | disabled     | 3rd party (MetalLB)            |
| metrics-server              | minikube | disabled     | Kubernetes                     |
| nvidia-device-plugin        | minikube | disabled     | 3rd party (NVIDIA)             |
| nvidia-driver-installer     | minikube | disabled     | 3rd party (Nvidia)             |
| nvidia-gpu-device-plugin    | minikube | disabled     | 3rd party (Nvidia)             |
| olm                         | minikube | disabled     | 3rd party (Operator Framework) |
| pod-security-policy         | minikube | disabled     | 3rd party (unknown)            |
| portainer                   | minikube | disabled     | 3rd party (Portainer.io)       |
| registry                    | minikube | disabled     | minikube                       |
| registry-aliases            | minikube | disabled     | 3rd party (unknown)            |
| registry-creds              | minikube | disabled     | 3rd party (UPMC Enterprises)   |
| storage-provisioner         | minikube | enabled ✅   | minikube                       |
| storage-provisioner-gluster | minikube | disabled     | 3rd party (Gluster)            |
| storage-provisioner-rancher | minikube | disabled     | 3rd party (Rancher)            |
| volumesnapshots             | minikube | disabled     | Kubernetes                     |
|-----------------------------|----------|--------------|--------------------------------|
```



```
complab304pc17@complab304:~$ minikube addons enable dashboard
💡  dashboard is an addon maintained by Kubernetes. For any concerns contact minikube on GitHub.
You can view the list of minikube maintainers at: https://github.com/kubernetes/minikube/blob/master/OWNERS
    ▪ Using image docker.io/kubernetesui/dashboard:v2.7.0
    ▪ Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
💡  Some dashboard features require the metrics-server addon. To enable all features please run:

        minikube addons enable metrics-server


🌟  The 'dashboard' addon is enabled
complab304pc17@complab304:~$ minikube addons enable ingress
💡  ingress is an addon maintained by Kubernetes. For any concerns contact minikube on GitHub.
You can view the list of minikube maintainers at: https://github.com/kubernetes/minikube/blob/master/OWNERS
    ▪ Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
    ▪ Using image registry.k8s.io/ingress-nginx/controller:v1.9.4
    ▪ Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
🔎  Verifying ingress addon...
🌟  The 'ingress' addon is enabled
```



Deployments

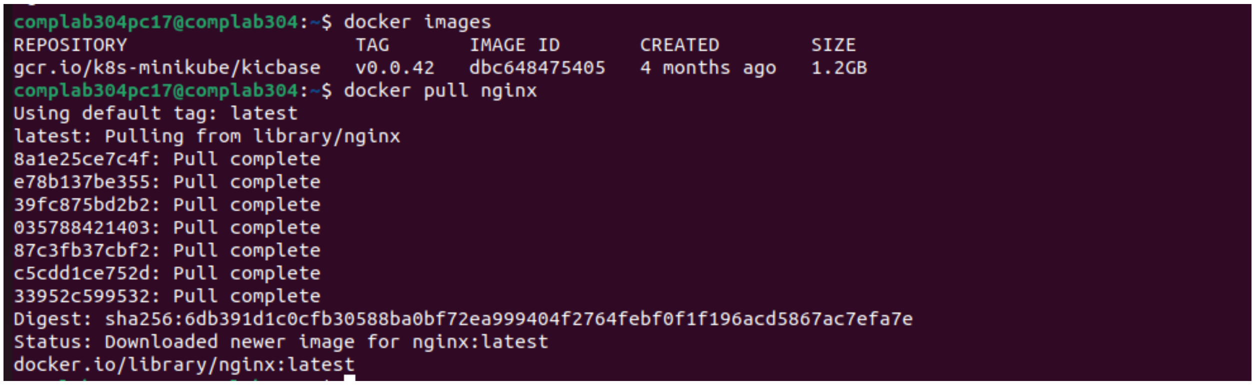
Name	Images	Labels	Pods	Created ↑
● nginx-web	nginx	app: nginx-web	1 / 1	3 minutes ago

Pods

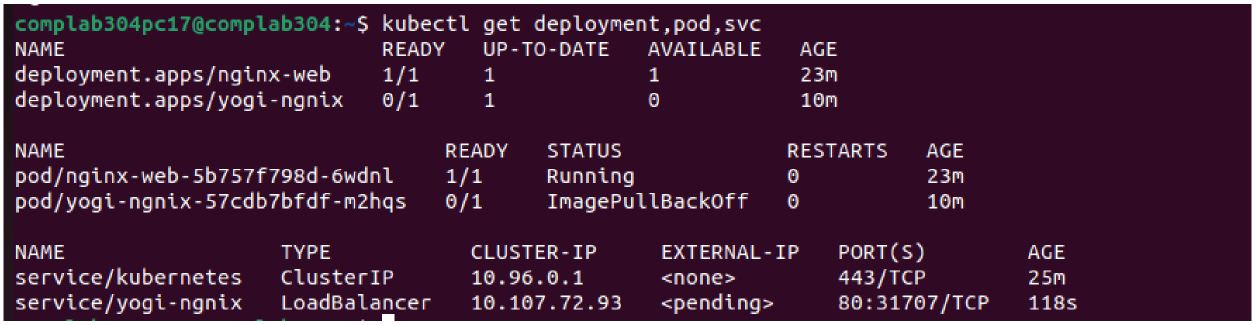
Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
● nginx-web-5b757f798d-6wdnl	nginx	app: nginx-web pod-template-hash: 5b757f798d	minikube	Running	0	-	-	3 minutes ago

Replica Sets

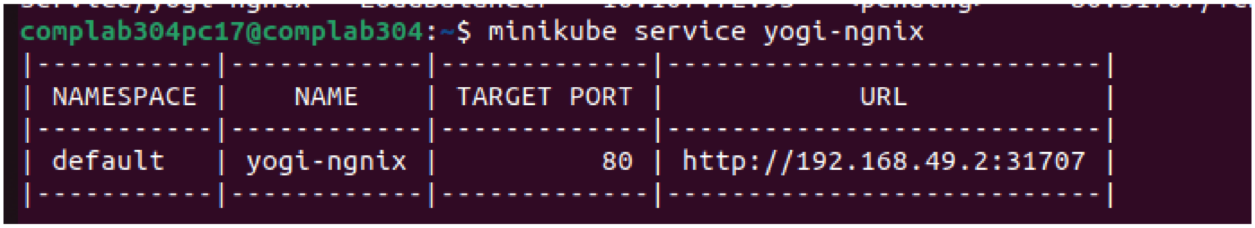
Name	Images	Labels	Pods	Created ↑
● nginx-web-5b757f798d	nginx	app: nginx-web pod-template-hash: 5b757f798d	1 / 1	3 minutes ago



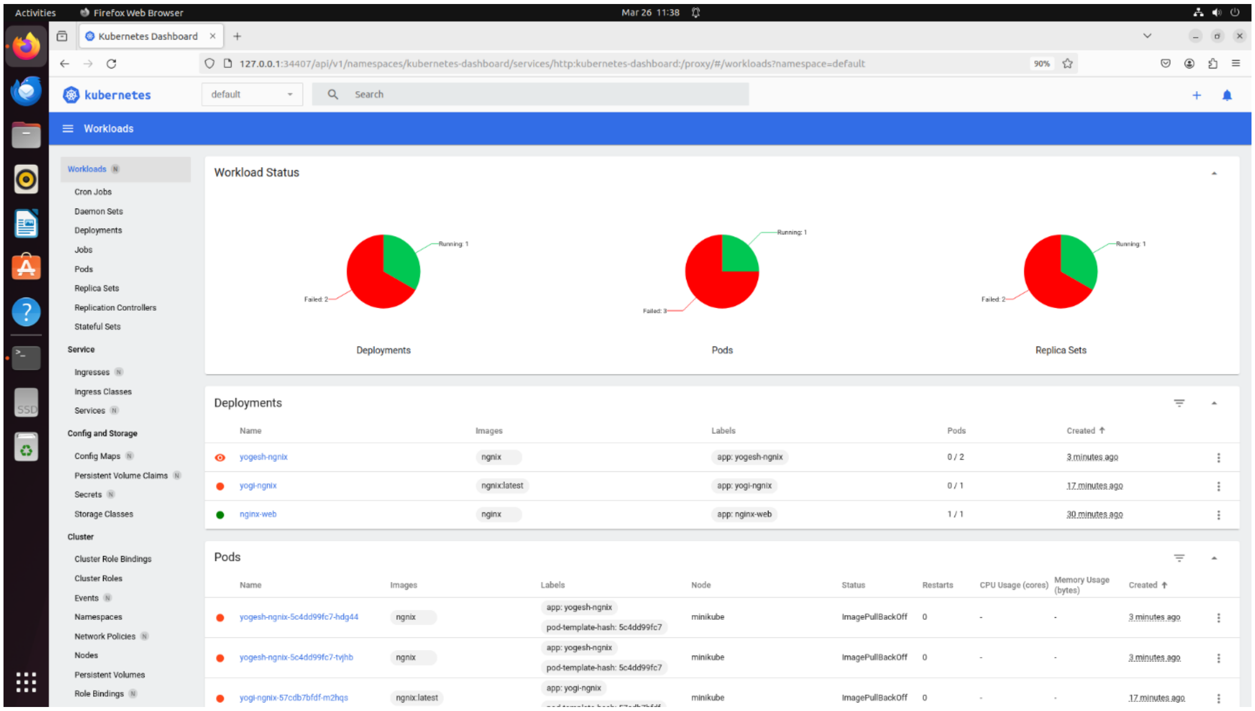
```
complab304pc17@complab304:~$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED        SIZE
gcr.io/k8s-minikube/kicbase   v0.0.42   dbc648475405   4 months ago   1.2GB
complab304pc17@complab304:~$ docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
8a1e25ce7c4f: Pull complete
e78b137be355: Pull complete
39fc875bd2b2: Pull complete
035788421403: Pull complete
87c3fb37cbf2: Pull complete
c5cdd1ce752d: Pull complete
33952c599532: Pull complete
Digest: sha256:6db391d1c0cfb30588ba0bf72ea999404f2764febf0f1f196acd5867ac7efa7e
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
```



```
complab304pc17@complab304:~$ kubectl get deployment,pod,svc
NAME                          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-web     1/1     1            1           23m
deployment.apps/yogi-ngnix    0/1     1            0           10m

NAME                                READY   STATUS             RESTARTS   AGE
pod/nginx-web-5b757f798d-6wdnl      1/1     Running            0          23m
pod/yogi-ngnix-57cdb7bfdf-m2hqs     0/1     ImagePullBackOff   0          10m

NAME                    TYPE           CLUSTER-IP     EXTERNAL-IP   PORT(S)        AGE
service/kubernetes      ClusterIP      10.96.0.1      <none>        443/TCP        25m
service/yogi-ngnix      LoadBalancer   10.107.72.93   <pending>     80:31707/TCP   118s
```



```
complab304pc17@complab304:~$ minikube service yogi-ngnix
|-----------|------------|-------------|---------------------------|
| NAMESPACE |    NAME    | TARGET PORT |            URL            |
|-----------|------------|-------------|---------------------------|
| default   | yogi-ngnix |          80 | http://192.168.49.2:31707 |
|-----------|------------|-------------|---------------------------|
```

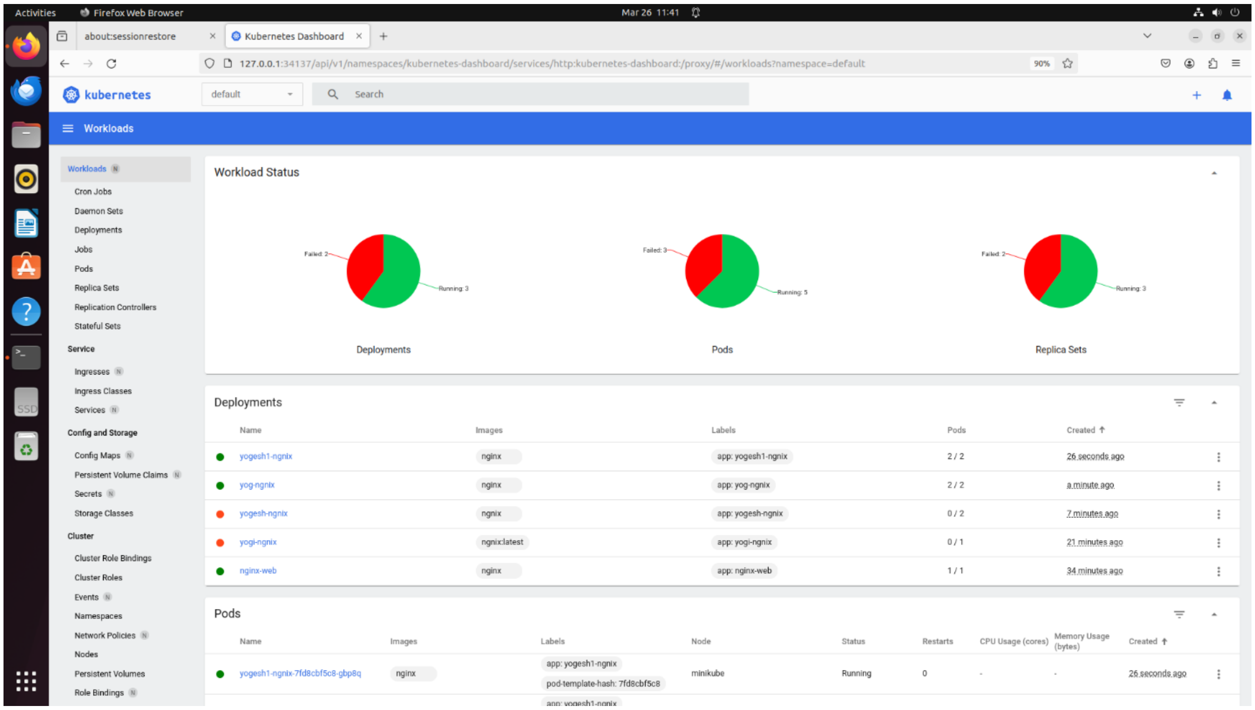


Deployments

	Name	Images	Labels	Pods	Created ↑
●	yogesh-ngnix	ngnix	app: yogesh-ngnix	0 / 2	3 minutes ago
●	yogi-ngnix	ngnix:latest	app: yogi-ngnix	0 / 1	17 minutes ago
●	nginx-web	nginx	app: nginx-web	1 / 1	30 minutes ago

Pods

	Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
●	yogesh-ngnix-5c4dd99fc7-hdg44	ngnix	app: yogesh-ngnix pod-template-hash: 5c4dd99fc7	minikube	ImagePullBackOff	0	-	-	3 minutes ago
●	yogesh-ngnix-5c4dd99fc7-tvjhb	ngnix	app: yogesh-ngnix pod-template-hash: 5c4dd99fc7	minikube	ImagePullBackOff	0	-	-	3 minutes ago
●	yogi-ngnix-57cdb7bfdf-m2hqs	ngnix:latest	app: yogi-ngnix	minikube	ImagePullBackOff	0	-	-	17 minutes ago

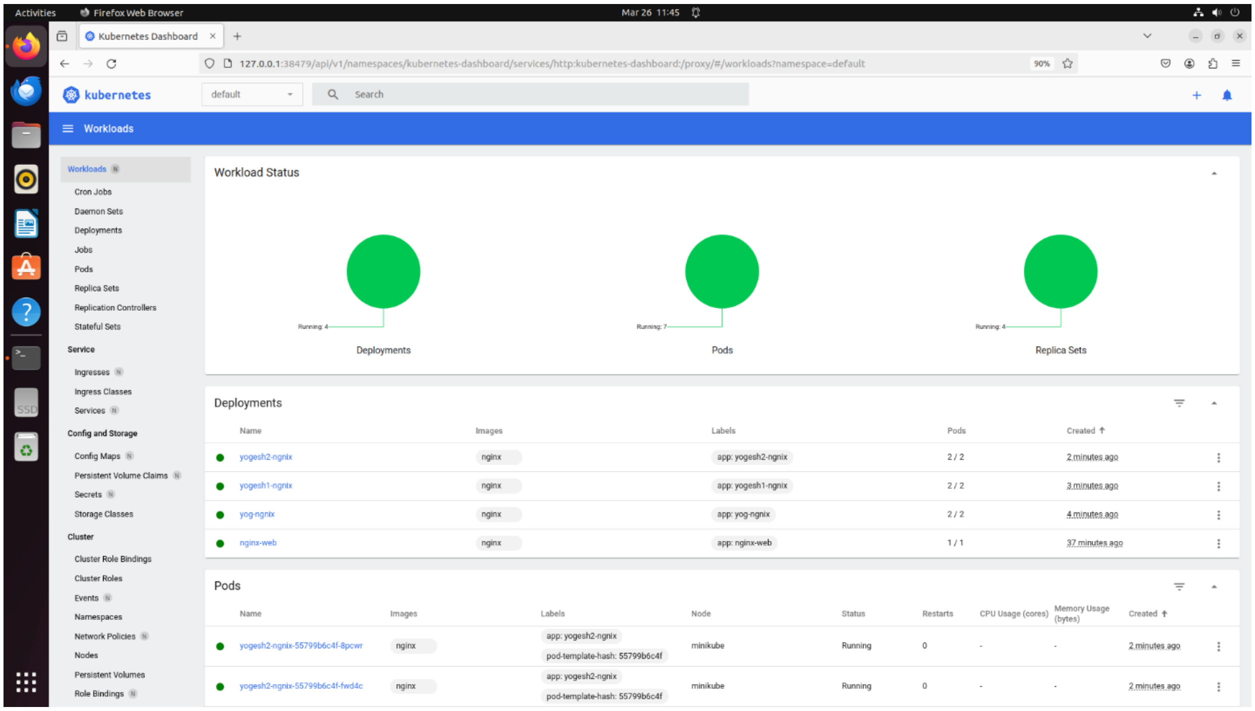


Deployments

	Name	Images	Labels	Pods	Created ↑
●	yogesh1-ngnix	nginx	app: yogesh1-ngnix	2 / 2	26 seconds ago
●	yog-ngnix	nginx	app: yog-ngnix	2 / 2	a minute ago
●	yogesh-ngnix	nginx	app: yogesh-ngnix	0 / 2	7 minutes ago
●	yogi-ngnix	nginx:latest	app: yogi-ngnix	0 / 1	21 minutes ago
●	nginx-web	nginx	app: nginx-web	1 / 1	34 minutes ago

Pods

	Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
●	yogesh1-ngnix-7fd8cbf5c8-gbp8q	nginx	app: yogesh1-ngnix pod-template-hash: 7fd8cbf5c8	minikube	Running	0	-	-	26 seconds ago
			app: yogesh1-ngnix						

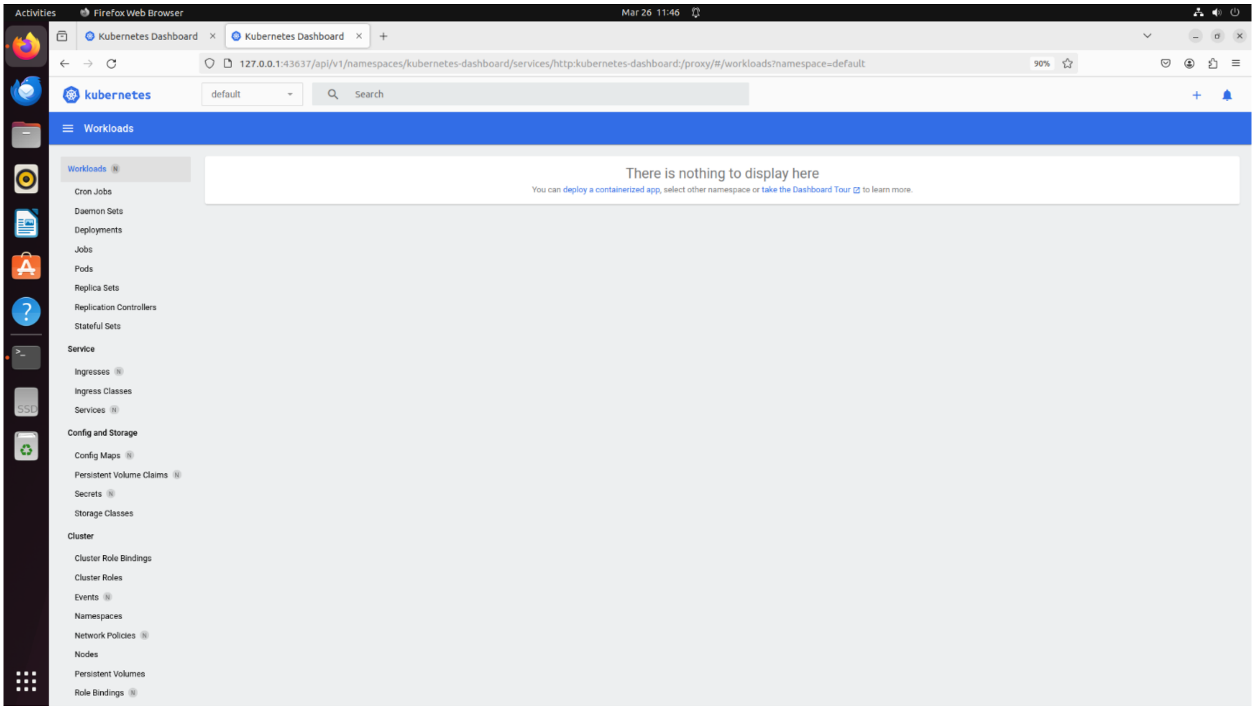


Deployments

	Name	Images	Labels	Pods	Created ↑
●	yogesh2-ngnix	nginx	app: yogesh2-ngnix	2 / 2	2 minutes ago
●	yogesh1-ngnix	nginx	app: yogesh1-ngnix	2 / 2	3 minutes ago
●	yog-ngnix	nginx	app: yog-ngnix	2 / 2	4 minutes ago
●	nginx-web	nginx	app: nginx-web	1 / 1	37 minutes ago

Pods

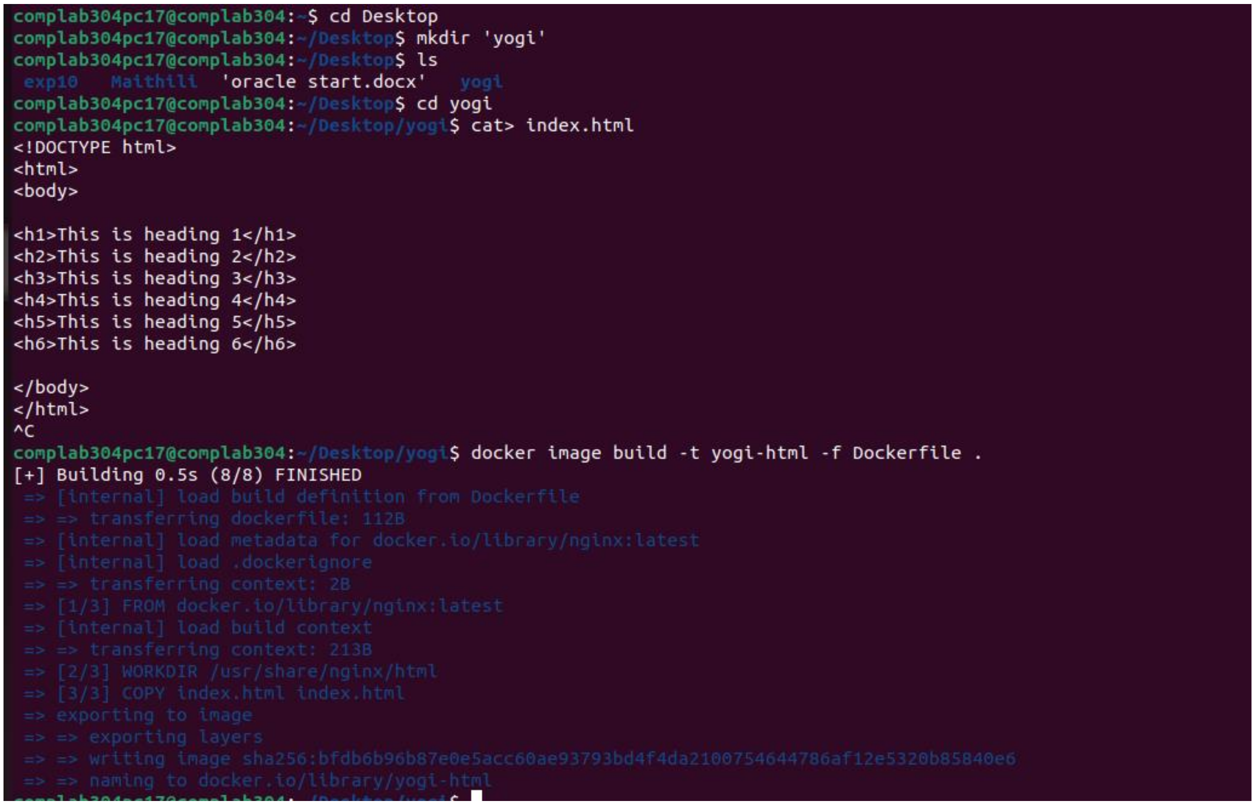
	Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
●	yogesh2-ngnix-55799b6c4f-8pcwr	nginx	app: yogesh2-ngnix pod-template-hash: 55799b6c4f	minikube	Running	0	-	-	2 minutes ago
●	yogesh2-ngnix-55799b6c4f-fwd4c	nginx	app: yogesh2-ngnix pod-template-hash: 55799b6c4f	minikube	Running	0	-	-	2 minutes ago



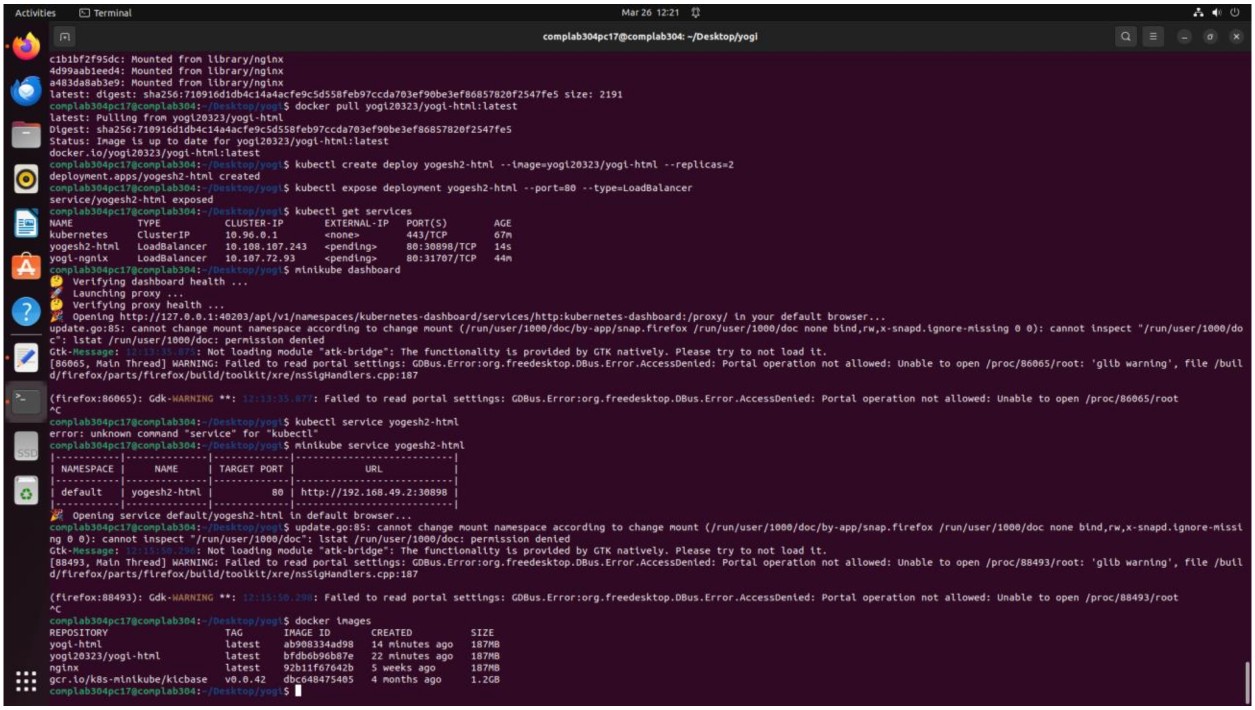
127.0.0.1:43637/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/#/workloads?namespace=default

There is nothing to display here

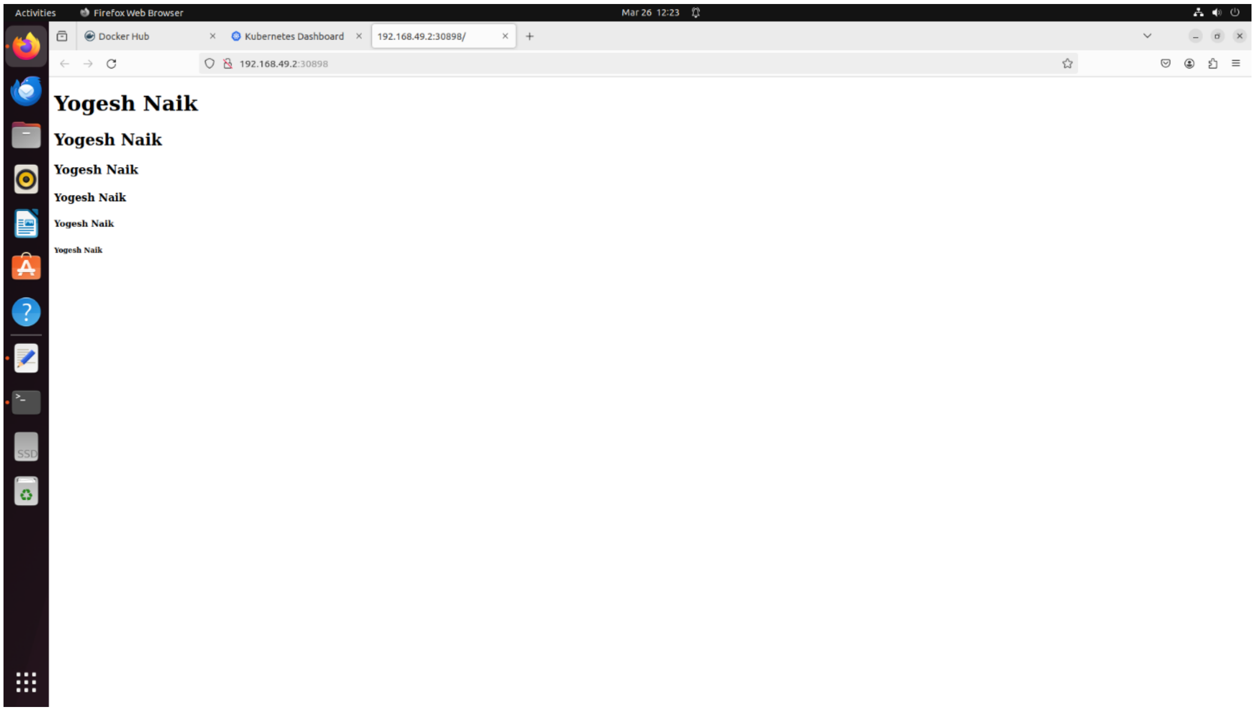
You can deploy a containerized app, select other namespace or take the Dashboard Tour to learn more.



```
complab304pc17@complab304:~$ cd Desktop
complab304pc17@complab304:~/Desktop$ mkdir 'yogi'
complab304pc17@complab304:~/Desktop$ ls
 exp10   Maithili  'oracle start.docx'   yogi
complab304pc17@complab304:~/Desktop$ cd yogi
complab304pc17@complab304:~/Desktop/yogi$ cat> index.html
<!DOCTYPE html>
<html>
<body>

<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>

</body>
</html>
^C
complab304pc17@complab304:~/Desktop/yogi$ docker image build -t yogi-html -f Dockerfile .
[+] Building 0.5s (8/8) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 112B
 => [internal] load metadata for docker.io/library/nginx:latest
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [1/3] FROM docker.io/library/nginx:latest
 => [internal] load build context
 => => transferring context: 213B
 => [2/3] WORKDIR /usr/share/nginx/html
 => [3/3] COPY index.html index.html
 => exporting to image
 => => exporting layers
 => => writing image sha256:bfdb6b96b87e0e5acc60ae93793bd4f4da2100754644786af12e5320b85840e6
 => => naming to docker.io/library/yogi-html
complab304pc17@complab304:~/Desktop/yogi$
```



```
c1b1bf2f95dc: Mounted from library/nginx
4d99aab1eed4: Mounted from library/nginx
a483da8ab3e9: Mounted from library/nginx
latest: digest: sha256:710916d1db4c14a4acfe9c5d558feb97ccda703ef90be3ef86857820f2547fe5 size: 2191
complab304pc17@complab304:~/Desktop/yogi$ docker pull yogi20323/yogi-html:latest
latest: Pulling from yogi20323/yogi-html
Digest: sha256:710916d1db4c14a4acfe9c5d558feb97ccda703ef90be3ef86857820f2547fe5
Status: Image is up to date for yogi20323/yogi-html:latest
docker.io/yogi20323/yogi-html:latest
complab304pc17@complab304:~/Desktop/yogi$ kubectl create deploy yogesh2-html --image=yogi20323/yogi-html --replicas=2
deployment.apps/yogesh2-html created
complab304pc17@complab304:~/Desktop/yogi$ kubectl expose deployment yogesh2-html --port=80 --type=LoadBalancer
service/yogesh2-html exposed
complab304pc17@complab304:~/Desktop/yogi$ kubectl get services
NAME           TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes     ClusterIP      10.96.0.1        <none>        443/TCP        67m
yogesh2-html   LoadBalancer   10.108.107.243   <pending>     80:30898/TCP   14s
yogi-ngnix     LoadBalancer   10.107.72.93     <pending>     80:31707/TCP   44m
complab304pc17@complab304:~/Desktop/yogi$ minikube dashboard
🤔  Verifying dashboard health ...
🚀  Launching proxy ...
🤔  Verifying proxy health ...
🎉  Opening http://127.0.0.1:40203/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in your default browser...
update.go:85: cannot change mount namespace according to change mount (/run/user/1000/doc/by-app/snap.firefox /run/user/1000/doc none bind,rw,x-snapd.ignore-missing 0 0): cannot inspect "/run/user/1000/do
c": lstat /run/user/1000/doc: permission denied
Gtk-Message: 12:13:35.875: Not loading module "atk-bridge": The functionality is provided by GTK natively. Please try to not load it.
[86065, Main Thread] WARNING: Failed to read portal settings: GDBus.Error:org.freedesktop.DBus.Error.AccessDenied: Portal operation not allowed: Unable to open /proc/86065/root: 'glib warning', file /buil
d/firefox/parts/firefox/build/toolkit/xre/nsSigHandlers.cpp:187

(firefox:86065): Gdk-WARNING **: 12:13:35.877: Failed to read portal settings: GDBus.Error:org.freedesktop.DBus.Error.AccessDenied: Portal operation not allowed: Unable to open /proc/86065/root
^C
complab304pc17@complab304:~/Desktop/yogi$ kubectl service yogesh2-html
error: unknown command "service" for "kubectl"
complab304pc17@complab304:~/Desktop/yogi$ minikube service yogesh2-html
|-----------|--------------|-------------|---------------------------|
| NAMESPACE |     NAME     | TARGET PORT |            URL            |
|-----------|--------------|-------------|---------------------------|
| default   | yogesh2-html |          80 | http://192.168.49.2:30898 |
|-----------|--------------|-------------|---------------------------|
🎉  Opening service default/yogesh2-html in default browser...
complab304pc17@complab304:~/Desktop/yogi$ update.go:85: cannot change mount namespace according to change mount (/run/user/1000/doc/by-app/snap.firefox /run/user/1000/doc none bind,rw,x-snapd.ignore-missi
ng 0 0): cannot inspect "/run/user/1000/doc": lstat /run/user/1000/doc: permission denied
Gtk-Message: 12:15:50.296: Not loading module "atk-bridge": The functionality is provided by GTK natively. Please try to not load it.
[88493, Main Thread] WARNING: Failed to read portal settings: GDBus.Error:org.freedesktop.DBus.Error.AccessDenied: Portal operation not allowed: Unable to open /proc/88493/root: 'glib warning', file /buil
d/firefox/parts/firefox/build/toolkit/xre/nsSigHandlers.cpp:187

(firefox:88493): Gdk-WARNING **: 12:15:50.298: Failed to read portal settings: GDBus.Error:org.freedesktop.DBus.Error.AccessDenied: Portal operation not allowed: Unable to open /proc/88493/root
^C
complab304pc17@complab304:~/Desktop/yogi$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
yogi-html                     latest    ab908334ad98   14 minutes ago   187MB
yogi20323/yogi-html           latest    bfdb6b96b87e   22 minutes ago   187MB
nginx                         latest    92b11f67642b   5 weeks ago      187MB
gcr.io/k8s-minikube/kicbase   v0.0.42   dbc648475405   4 months ago     1.2GB
complab304pc17@complab304:~/Desktop/yogi$ 
```



Yogesh Naik

Yogesh Naik

Yogesh Naik

Yogesh Naik

Yogesh Naik

Yogesh Naik