Experiment No: 10

Aim: To Study and implement container orchestration using kubernetes.

Theory:

1) Explain need of container orchestration tool

⇒ Container orchestration tools are needed to manage, deploy, scale, and automate containers effectively.

- They ensure containers run as intended, manage resource allocation, handle networking, storage, and maintain high availability across a cluster of machines.
- Container orchestration tools automate various tasks involved in managing containers, such as deployment, scaling, load balancing, health monitoring, and recovery.
- As the number of containers and the size of the infrastructure grow, manual management becomes impractical.

2) What is Kubernetes? Describe its features

⇒ Kubernetes is an open-source container orchestration platform designed to automate the deployment, scaling and management of containerized applications.

- It's features are :-

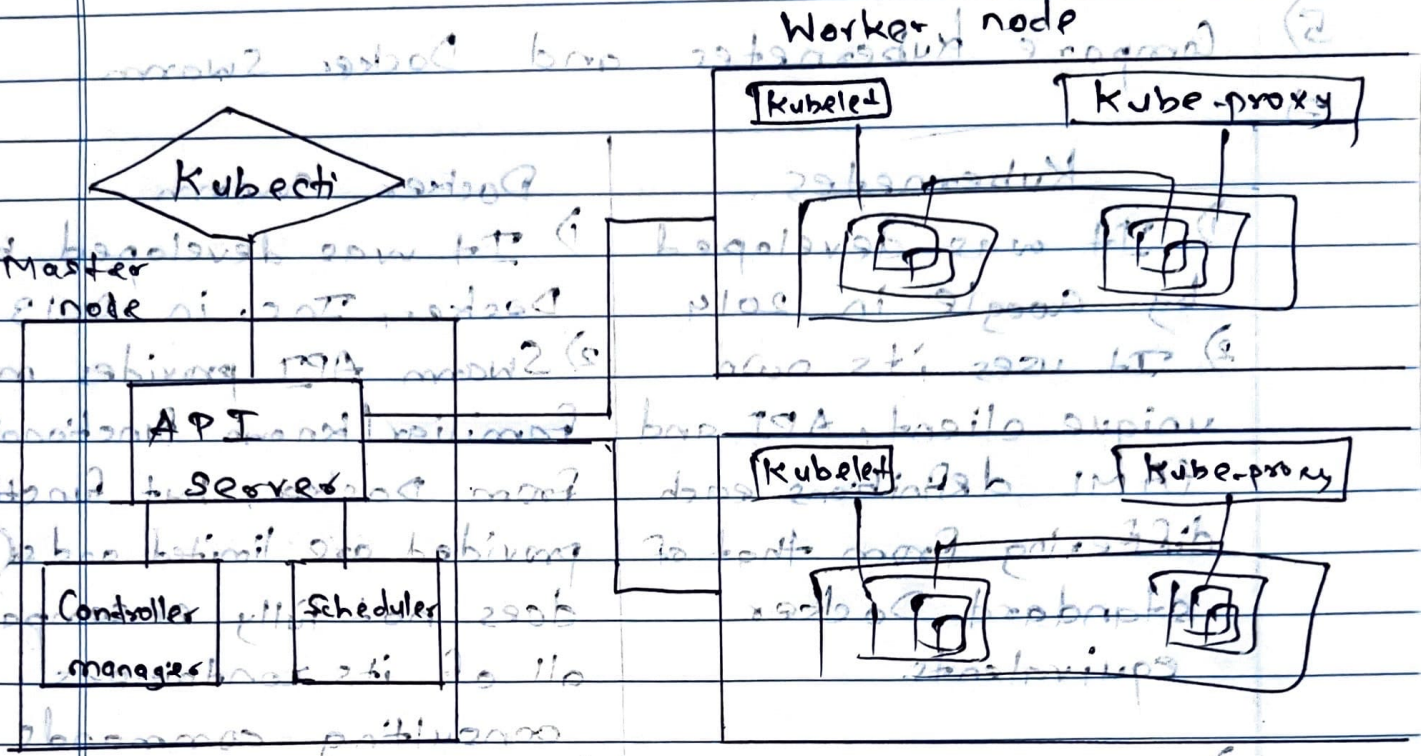
i) Automated rollouts and rollbacks

ii) Service discovery and load balancing,

iii) Storage orchestration
iv) Self-healing
v) Secret and configuration management
vi) Automatic bin packing
vii) Batch execution
viii) Horizontal scaling
ix) IPv4/IPv6 dual-stack
x) Designed for extensibility.

3) Explain Kubernetes components, its working and architecture.

=> Kubernetes components include the Master and Worker nodes.
- The Master node oversees the cluster and manages its state through various components such as the API server, controller manager, scheduler and etcd.
- Worker nodes host the containers and run the necessary kubernetes components like kubelet, kube-proxy and container runtime.
- Kubernetes has a client-server architecture and has master and worker nodes, with the masters being installed on a single Linux system and the nodes on many Linux workstations.



4) Difference between POD and node

⇒

POD	Node
1) A pod is smallest deployable unit in Kubernetes	1) A node is a physical or virtual machine in a Kubernetes cluster.
2) It represents a single instance of a running process in your cluster.	2) It is the underlying infrastructure where POD runs
3) PODs are ephemeral and can be created, destroyed or replicated dynamically based on workload requirement.	3) Nodes are responsible running and managing PODs providing the necessary computing, networking and storage resource.

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

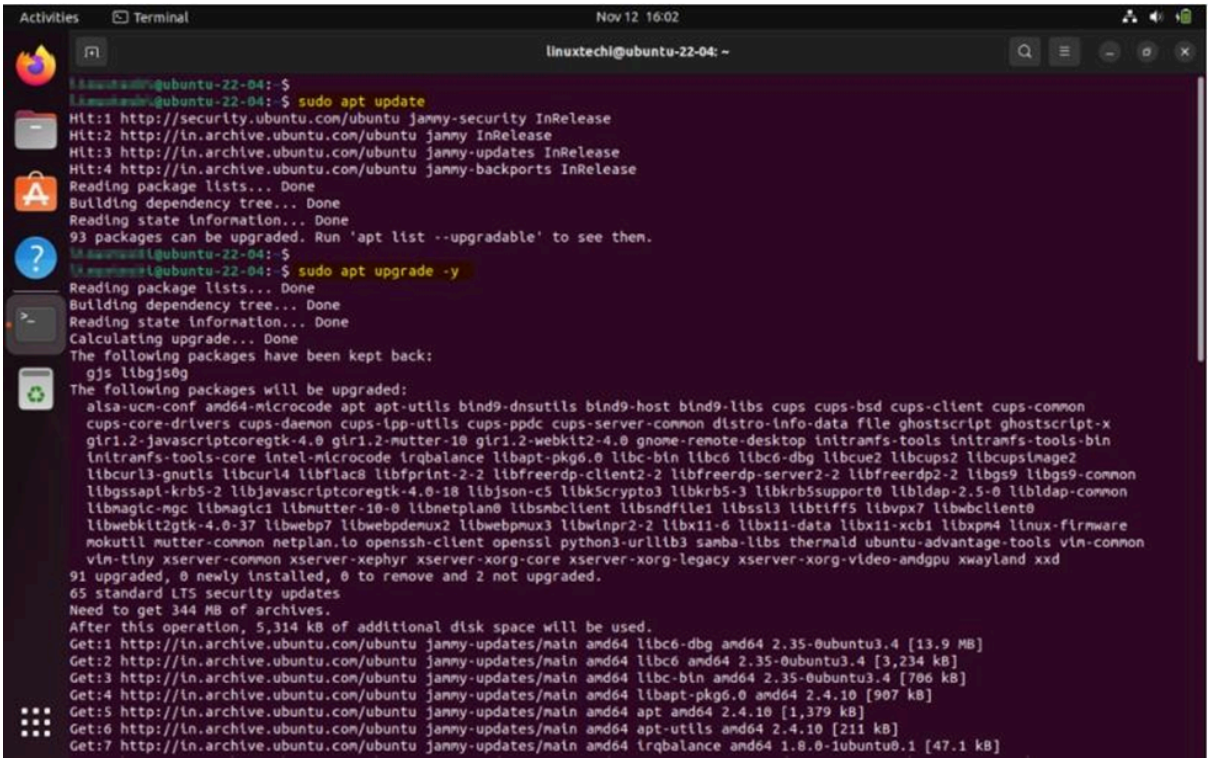
5) Compare Kubernetes and Docker Swarm

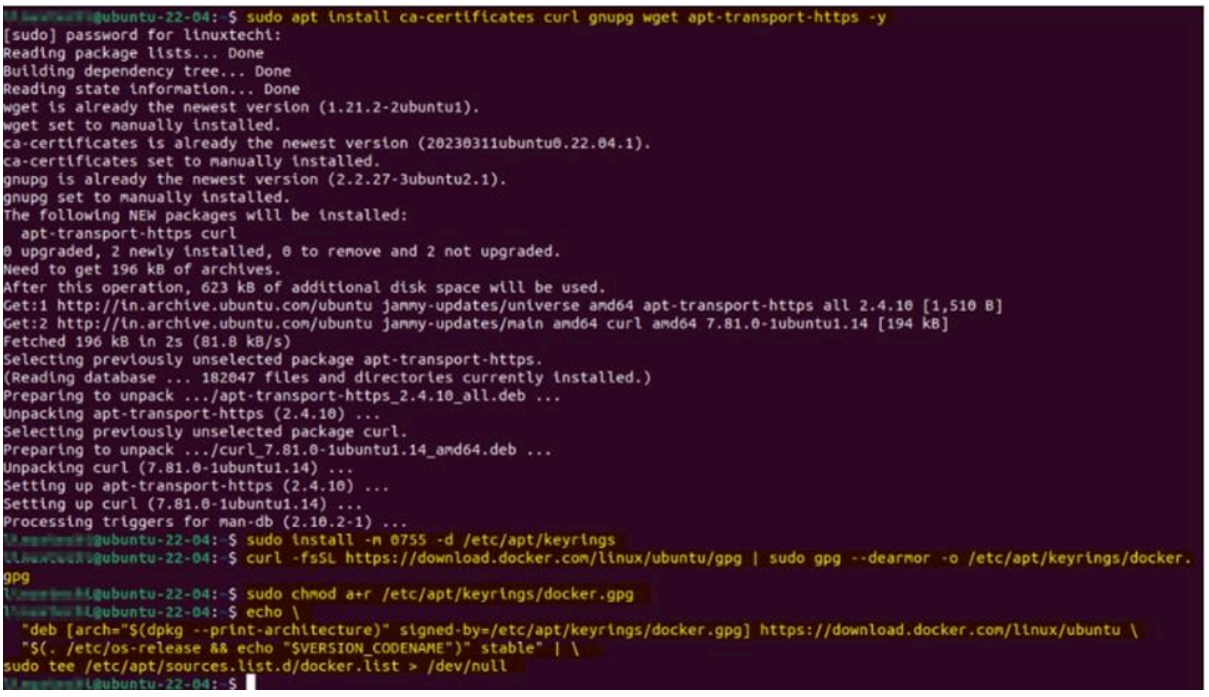
=>

Kubernetes	Docker Swarm
1) It was developed by Google in 2014	1) It was developed by Docker, Inc. in 2013
2) It uses its own unique client, API and YAML definitions each differing from that of standard Docker equivalents.	2) Swarm API provides many familiar/known functionalities from Docker, but functionalities provided are limited and so it does not fully encompass all of its containers consulting commands.
3) It supports a more complex, flexible architecture with stronger service guarantees due to which performance slows down.	3) On other hand, supports a simpler architecture, so in terms of sheer speed, it always has an added advantage.
4) It supports auto-scaling.	4) It cannot do auto-scaling.

A SP 13/3/2024

Output:


Activities
Terminal
Nov 12 16:02
linuxtechi@ubuntu-22-04: ~
@ubuntu-22-04:~$
@ubuntu-22-04:~$ sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
93 packages can be upgraded. Run 'apt list --upgradable' to see them.
@ubuntu-22-04:~$
@ubuntu-22-04:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
gjs libgjs0g
The following packages will be upgraded:
alsa-ucm-conf amd64-microcode apt apt-utils bind9-dnsutils bind9-host bind9-libs cups cups-bsd cups-client cups-common
cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc cups-server-common distro-info-data file ghostscript ghostscript-x
gir1.2-javascriptcoregtk-4.0 gir1.2-mutter-10 gir1.2-webkit2-4.0 gnome-remote-desktop initramfs-tools initramfs-tools-bin
initramfs-tools-core intel-microcode irqbalance libapt-pkg6.0 libc-bin libc6 libc6-dbg libcue2 libcups2 libcupsimage2
libcurl3-gnutls libcurl4 libflac8 libfprint-2-2 libfreerdp-client2-2 libfreerdp-server2-2 libfreerdp2-2 libgs9 libgs9-common
libgssapi-krb5-2 libjavascriptcoregtk-4.0-18 libjson-c5 libk5crypto3 libkrb5-3 libkrb5support0 libldap-2.5-0 libldap-common
libmagic-mgc libmagic1 libmutter-10-0 libnetplan0 libsmbclient libsndfile1 libssl3 libtiff5 libvpx7 libwbclient0
libwebkit2gtk-4.0-37 libwebp7 libwebpdemux2 libwebpmux3 libwinpr2-2 libx11-6 libx11-data libx11-xcb1 libxpm4 linux-firmware
mokutil mutter-common netplan.io openssh-client openssl python3-urllib3 samba-libs thermald ubuntu-advantage-tools vim-common
vim-tiny xserver-common xserver-xephyr xserver-xorg-core xserver-xorg-legacy xserver-xorg-video-amdgpu xwayland xxd
91 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
65 standard LTS security updates
Need to get 344 MB of archives.
After this operation, 5,314 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc6-dbg amd64 2.35-0ubuntu3.4 [13.9 MB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc6 amd64 2.35-0ubuntu3.4 [3,234 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc-bin amd64 2.35-0ubuntu3.4 [706 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libapt-pkg6.0 amd64 2.4.10 [907 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apt amd64 2.4.10 [1,379 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apt-utils amd64 2.4.10 [211 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 irqbalance amd64 1.8.0-1ubuntu0.1 [47.1 kB]


@ubuntu-22-04:~$ sudo apt install ca-certificates curl gnupg wget apt-transport-https -y
[sudo] password for linuxtechi:
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.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following NEW packages will be installed:
apt-transport-https curl
0 upgraded, 2 newly installed, 0 to remove and 2 not upgraded.
Need to get 196 kB of archives.
After this operation, 623 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.10 [1,510 B]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.14 [194 kB]
Fetched 196 kB in 2s (81.8 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 182047 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.10_all.deb ...
Unpacking apt-transport-https (2.4.10) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.81.0-1ubuntu1.14_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.14) ...
Setting up apt-transport-https (2.4.10) ...
Setting up curl (7.81.0-1ubuntu1.14) ...
Processing triggers for man-db (2.10.2-1) ...
@ubuntu-22-04:~$ sudo install -m 0755 -d /etc/apt/keyrings
@ubuntu-22-04:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.
gpg
@ubuntu-22-04:~$ sudo chmod a+r /etc/apt/keyrings/docker.gpg
@ubuntu-22-04:~$ echo \
"deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
"$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
@ubuntu-22-04:~$

```
[redacted]@ubuntu-22-04:~$ 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 additional packages will be installed:
  docker-ce-rootless-extras git git-man liberror-perl libslirp0 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 libslirp0 pigz slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 2 not upgraded.
Need to get 118 MB of archives.
After this operation, 430 MB of additional disk space will be used.
Get:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.24-1 [28.6 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 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-buildx-plugin amd64 0.11.2-1~ubuntu.22.04~jammy [28.2 MB]
Get:6 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.10 [3,166 kB]
Get:7 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-cli amd64 5:24.0.7-1~ubuntu.22.04~jammy [13.3 MB]
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce amd64 5:24.0.7-1~ubuntu.22.04~jammy [22.6 MB]
Get:9 http://in.archive.ubuntu.com/ubuntu jammy/main amd64 libslirp0 amd64 4.6.1-1build1 [61.5 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu jammy/universe amd64 slirp4netns amd64 1.0.1-2 [28.2 kB]
Get:11 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-rootless-extras amd64 5:24.0.7-1~ubuntu.22.04~jammy [9,
030 kB]
Get:12 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-compose-plugin amd64 2.21.0-1~ubuntu.22.04~jammy [11.9 MB]
Fetched 118 MB in 8s (14.4 MB/s)
```

```
[redacted]@ubuntu-22-04:~$ sudo usermod -aG docker $USER
[redacted]@ubuntu-22-04:~$ newgrp docker
[redacted]@ubuntu-22-04:~$ systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2023-11-12 16:20:53 IST; 1min 54s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 3114 (dockerd)
      Tasks: 8
     Memory: 27.3M
        CPU: 1.065s
     CGroup: /system.slice/docker.service
             └─3114 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Nov 12 16:20:51 ubuntu-22-04 systemd[1]: Starting Docker Application Container Engine...
Nov 12 16:20:51 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:51.771454830+05:30" level=info msg="Starting up"
Nov 12 16:20:51 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:51.774072108+05:30" level=info msg="detected 127.0.0.53 nameserv
Nov 12 16:20:52 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:52.051794664+05:30" level=info msg="Loading containers: start."
Nov 12 16:20:53 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:53.028261474+05:30" level=info msg="Loading containers: done."
Nov 12 16:20:53 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:53.231111803+05:30" level=info msg="Docker daemon" commit=311b9f
Nov 12 16:20:53 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:53.233235025+05:30" level=info msg="Daemon has completed initial
Nov 12 16:20:53 ubuntu-22-04 dockerd[3114]: time="2023-11-12T16:20:53.351693568+05:30" level=info msg="API listen on /run/docker.so
Nov 12 16:20:53 ubuntu-22-04 systemd[1]: Started Docker Application Container Engine.
```

```
[redacted]@ubuntu-22-04:~$ 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   9.8M      0  0:00:09  0:00:09 --:--:-- 13.4M
[redacted]@ubuntu-22-04:~$
[redacted]@ubuntu-22-04:~$ sudo install minikube-linux-amd64 /usr/local/bin/minikube
[redacted]@ubuntu-22-04:~$
[redacted]@ubuntu-22-04:~$ minikube version
minikube version: v1.32.0
commit: 8220a6eb95f0a4d75f7f2d7b14cef975f050512d
[redacted]@ubuntu-22-04:~$
[redacted]@ubuntu-22-04:~$
```

```
linuxbuzz@ubuntu-22-04:~$ 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 47.5M  100 47.5M    0     0  8524k      0  0:00:05  0:00:05 --:--:--  9.8M
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ chmod +x kubectl
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ sudo mv kubectl /usr/local/bin/
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ kubectl version -o yaml
clientVersion:
  buildDate: "2023-10-18T11:42:52Z"
  compiler: gc
  gitCommit: a8a1abc25cad87333840cd7d54be2efaf31a3177
  gitTreeState: clean
  gitVersion: v1.28.3
  goVersion: go1.20.10
  major: "1"
  minor: "28"
  platform: linux/amd64
kustomizeVersion: v5.0.4-0.20230601165947-6ce0bf390ce3

The connection to the server localhost:8080 was refused - did you specify the right host or port?
linuxbuzz@ubuntu-22-04:~$
```

```
linuxbuzz@ubuntu-22-04:~$ minikube start --driver=docker
😄  minikube v1.32.0 on Ubuntu 22.04 (vbox/amd64)
✨  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% 7.23 Mi
    > gcr.io/k8s-minikube/kicbase...:  453.90 MiB / 453.90 MiB  100.00% 7.36 Mi
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  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
linuxbuzz@ubuntu-22-04:~$
```

```
linuxbuzz@ubuntu-22-04:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

linuxbuzz@ubuntu-22-04:~$
```

```
linuxbuzz@ubuntu-22-04:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE     VERSION
minikube   Ready    control-plane   5m49s   v1.28.3
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ 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'.
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$
```

```
linuxbuzz@ubuntu-22-04:~$ kubectl create deployment nginx-web --image=nginx
deployment.apps/nginx-web created
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ kubectl expose deployment nginx-web --type NodePort --port=80
service/nginx-web exposed
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ kubectl get deployment,pod,svc
NAME                            READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-web       1/1     1            1           40s

NAME                                READY   STATUS    RESTARTS   AGE
pod/nginx-web-5b757f798d-qnbzq      1/1     Running   0          39s

NAME                 TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
service/kubernetes   ClusterIP   10.96.0.1        <none>        443/TCP        12m
service/nginx-web    NodePort    10.104.243.119   <none>        80:30523/TCP   27s
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$
```

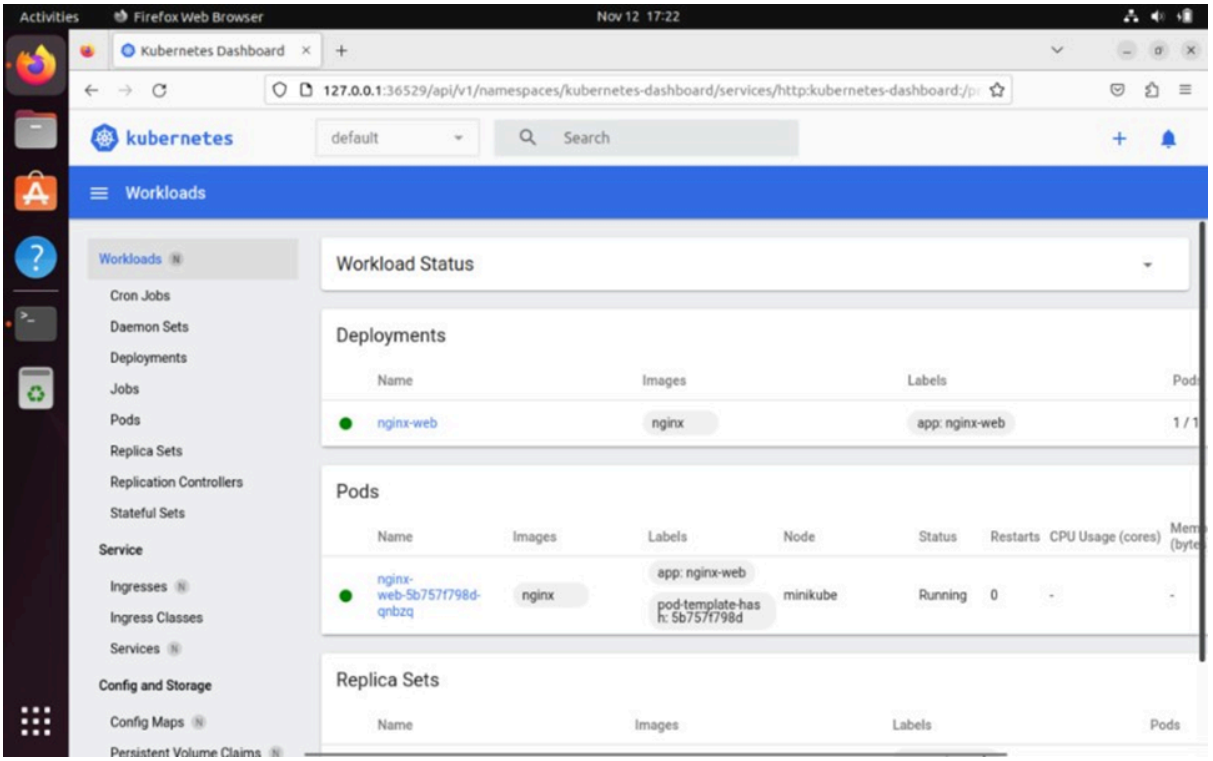
```
linuxbuzz@ubuntu-22-04:~$ minikube addons list
|-----------------------------|----------|--------------|-------------------------------|
|         ADDON NAME          | PROFILE  |    STATUS    |           MAINTAINER          |
|-----------------------------|----------|--------------|-------------------------------|
| 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)           |
```

```
linuxbuzz@ubuntu-22-04:~$ 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
linuxbuzz@ubuntu-22-04:~$
linuxbuzz@ubuntu-22-04:~$ 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/controller:v1.9.4
    ▪ Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
    ▪ Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
🔎  Verifying ingress addon...
🌟  The 'ingress' addon is enabled
linuxbuzz@ubuntu-22-04:~$
```

```
linuxbuzz@ubuntu-22-04:~$ minikube dashboard
🤔  Verifying dashboard health ...
🚀  Launching proxy ...
🤔  Verifying proxy health ...
🎉  Opening http://127.0.0.1:36529/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in your default
 browser...
```


Activities
Firefox Web Browser
Nov 12 17:22
Kubernetes Dashboard
127.0.0.1:36529/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/p
kubernetes
default
Search
Workloads
Workloads
Cron Jobs
Daemon Sets
Deployments
Jobs
Pods
Replica Sets
Replication Controllers
Stateful Sets
Service
Ingresses
Ingress Classes
Services
Config and Storage
Config Maps
Persistent Volume Claims
Workload Status
Deployments
Name
Images
Labels
Pod
nginx-web
nginx
app: nginx-web
1 / 1
Pods
Name
Images
Labels
Node
Status
Restarts
CPU Usage (cores)
Mem (byte
nginx-web-5b757f798d-qnbzq
nginx
app: nginx-web
pod-template-hash: 5b757f798d
minikube
Running
0
-
-
Replica Sets
Name
Images
Labels
Pods