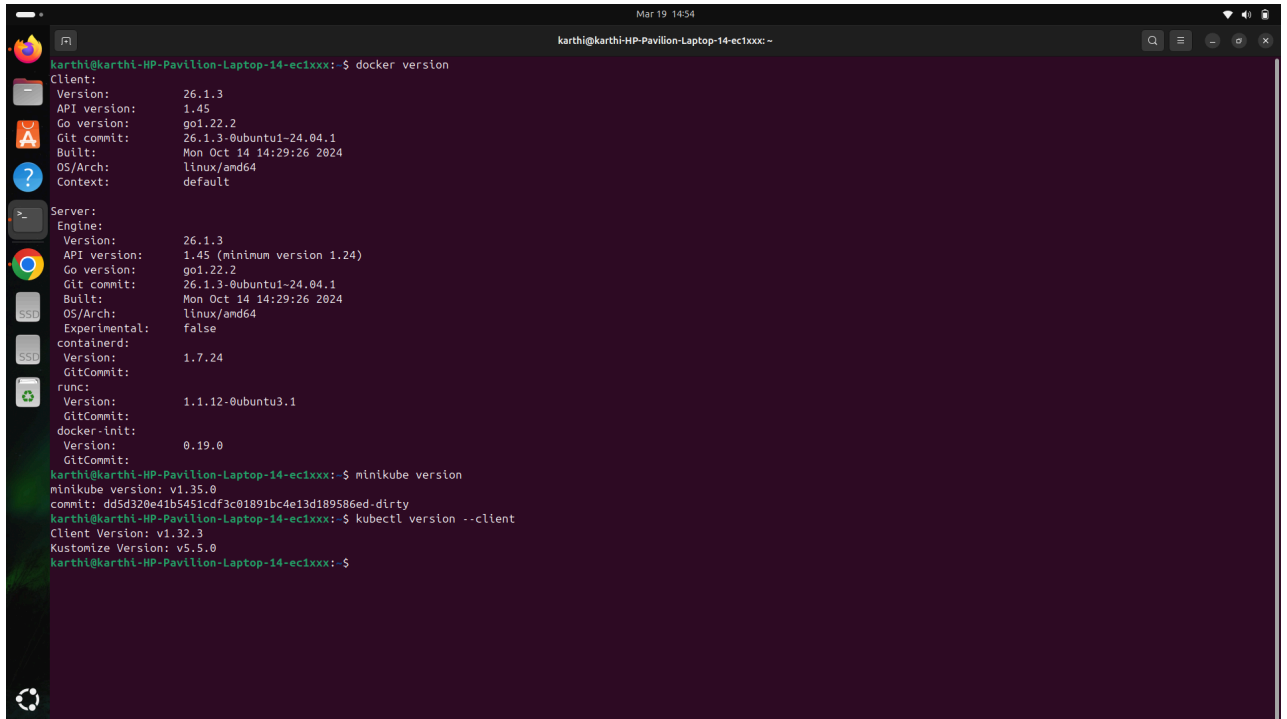


## DAY - 3

# Deploy an Nginx Container in Kubernetes and expose it with a NodePort Service

A terminal window with a dark purple background. The prompt is 'karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~'. The user has entered 'docker version' and 'minikube version'. The output shows Docker client and server details, and Minikube version v1.35.0. The terminal window has a title bar with 'Mar 19 14:54' and window control buttons.

```
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ docker version
Client:
Version:      26.1.3
API version:  1.45
Go version:   go1.22.2
Git commit:   26.1.3-0ubuntu1-24.04.1
Built:        Mon Oct 14 14:29:26 2024
OS/Arch:      linux/amd64
Context:      default

Server:
Engine:
Version:      26.1.3
API version:  1.45 (minimum version 1.24)
Go version:   go1.22.2
Git commit:   26.1.3-0ubuntu1-24.04.1
Built:        Mon Oct 14 14:29:26 2024
OS/Arch:      linux/amd64
Experimental: false
Containerd:
Version:      1.7.24
GitCommit:
runc:
Version:      1.1.12-0ubuntu3.1
GitCommit:
docker-init:
Version:      0.19.0
GitCommit:

karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ minikube version
minikube version: v1.35.0
commit: d05d320e41b5451cd3c01891bc4e13d189586ed-dirty
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ kubectl version --client
Client Version: v1.32.3
Kustomize Version: v5.5.0
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$
```

```
sudo nano nginx-deployment.yml
```

nginx-deployment.yml

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-login

spec:

replicas: 1

selector:

matchLabels:

app: nginx-login

template:

metadata:

labels:

app: nginx-login

spec:

containers:

- name: nginx-login

image: nginx:latest # Public Nginx image from Docker Hub

ports:

- containerPort: 80

---

apiVersion: v1

kind: Service

metadata:

name: nginx-login-service

spec:

type: NodePort

selector:

app: nginx-login

ports:

- protocol: TCP

port: 80

targetPort: 80

nodePort: 30008 # NodePort exposes service on this port

```
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ minikube start
🐳 minikube v1.35.0 on Ubuntu 24.04
🌟 Using the docker driver based on existing profile
👍 Starting "minikube" primary control-plane node in "minikube" cluster
📡 Pulling base image v0.0.46 ...
🔄 Restarting existing docker container for "minikube" ...
🔧 Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
🔍 Verifying Kubernetes components...
   ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟 Enabled addons: default-storageclass, storage-provisioner
🏡 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ 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'.
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ kubectl apply -f nginx-deployment.yml
deployment.apps/nginx-login created
service/nginx-login-service created
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ kubectl get pods
NAME                                READY   STATUS              RESTARTS   AGE
nginx-login-b6fd8c4cf-8dbnh         0/1     ContainerCreating   0           14s
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ kubectl get svc
NAME                TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
kubernetes          ClusterIP   10.96.0.1    <none>        443/TCP          24h
nginx-login-service NodePort    10.99.136.215 <none>        80:30008/TCP    22s
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ minikube ip
192.168.49.2
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$
```

`kubectl apply -f nginx-deployment.yml`

Check if the pods are running:

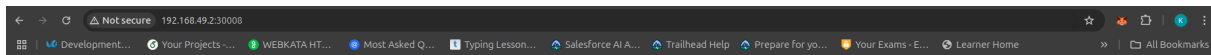
`kubectl get pods`

Check if the service is created:

`kubectl get svc`

Find the ip:

`minikube ip`



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

Thank you for using nginx.

```
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$ curl http://192.168.49.2:30008
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
karthi@karthi-HP-Pavilion-Laptop-14-ec1xxx:~$
```