Install Docker on Ubuntu and kubectl

```
root@ip-172-31-34-212:~# # 1. Remove the "bad" files
rm kubectl kubectl.sha256

# 2. Download the binary
curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

# 3. Download the checksum
curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sh
a256"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   138  100   138    0     0    411      0 --:--:-- --:--:-- --:--:--   411
100 55.8M  100 55.8M    0     0  64.8M      0 --:--:-- --:--:-- --:--:--  137M
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   138  100   138    0     0    410      0 --:--:-- --:--:-- --:--:--   410
100    64  100    64    0     0    157      0 --:--:-- --:--:-- --:--:--   157
root@ip-172-31-34-212:~# echo "$(cat kubectl.sha256)  kubectl" | sha256sum -c
kubectl: OK
root@ip-172-31-34-212:~# sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
root@ip-172-31-34-212:~# kubectl version -client
error: extra arguments: [-client]
root@ip-172-31-34-212:~# kubectl version -client
error: unknown shorthand flag: 'c' in -client
See 'kubectl version --help' for usage.
root@ip-172-31-34-212:~# kubectl version --client
Client Version: v1.35.0
Kustomize Version: v5.7.1
root@ip-172-31-34-212:~# kubectl version --client --output=yaml
clientVersion:
  buildDate: "2025-12-17T12:41:05Z"
  compiler: gc
  gitCommit: 66452049f3d692768c39c797b21b793dce80314e
  gitTreeState: clean
  gitVersion: v1.35.0
  goVersion: go1.25.5
  major: "1"
  minor: "35"
  platform: linux/amd64
kustomizeVersion: v5.7.1
```

```
tlxp@ip-172-31-34-212:~$ docker ps
permission denied while trying to connect to the docker API at unix:///var/run/docker.sock
tlxp@ip-172-31-34-212:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
tlxp@ip-172-31-34-212:~$ sudo docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED          STATUS                      PORTS     NAMES
2c11e80d15a7   hello-world   "/hello"   33 seconds ago   Exited (0) 32 seconds ago             condescending_mat
umoto
tlxp@ip-172-31-34-212:~$ sudo docker images
                                                                                    i Info →  U In Us
IMAGE                ID             DISK USAGE   CONTENT SIZE   EXTRA
hello-world:latest   d4aaab6242e0       25.9kB         9.52kB    U
tlxp@ip-172-31-34-212:~$ sudo su -
root@ip-172-31-34-212:~# ^C
root@ip-172-31-34-212:~# docker rm 2c11e80d15a7
2c11e80d15a7
root@ip-172-31-34-212:~#
```

MiniKube Installation

```
root@ip-172-31-34-212:~# 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  133M  100  133M    0     0  12.8M      0  0:00:10  0:00:10 --:--:-- 16.5M
root@ip-172-31-34-212:~# sudo install minikube-linux-amd64 /usr/local/bin/minikube
root@ip-172-31-34-212:~# minikube version
minikube version: v1.37.0
commit: 65318f4cfff9c12cc87ec9eb8f4cdd57b25047f3
```

```
tlxp@ip-172-31-34-212:~$ minikube delete --profile=minikube
🔥  Deleting "minikube" in docker ...
🔥  Deleting container "minikube" ...
🔥  Removing /home/tlxp/.minikube/machines/minikube ...
💀  Removed all traces of the "minikube" cluster.
tlxp@ip-172-31-34-212:~$ ^C
tlxp@ip-172-31-34-212:~$ minikube start --profile=minikube
😄  minikube v1.37.0 on Ubuntu 22.04 (xen/amd64)
✨  Automatically selected the docker driver

🧯  The requested memory allocation of 3072MiB does not leave room for system overhead (total system memory: 3912M
iB). You may face stability issues.
💡  Suggestion: Start minikube with less memory allocated: 'minikube start --memory=3072mb'

📌  Using Docker driver with root privileges
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.48 ...
🔥  Creating docker container (CPUs=2, Memory=3072MB) ...
📦  Preparing Kubernetes v1.34.0 on Docker 28.4.0 ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  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
tlxp@ip-172-31-34-212:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   17s   v1.34.0
tlxp@ip-172-31-34-212:~$
```

Checking basic configuration of Kubernetes

```
tlxp@ip-172-31-34-212:~$ kubectl get pods
NAME                     READY   STATUS              RESTARTS   AGE
nginx-66686b6766-z2scz   0/1     ContainerCreating   0          2s
```

```
tlxp@ip-172-31-34-212:~$ kubectl get pods --all-namespaces
NAMESPACE     NAME                                   READY   STATUS    RESTARTS      AGE
default       nginx-66686b6766-z2scz                 1/1     Running   0             14s
kube-system   coredns-66bc5c9577-69d2s               1/1     Running   0             16m
kube-system   etcd-minikube                          1/1     Running   0             16m
kube-system   kube-apiserver-minikube                1/1     Running   0             16m
kube-system   kube-controller-manager-minikube       1/1     Running   0             16m
kube-system   kube-proxy-7cm55                       1/1     Running   0             16m
kube-system   kube-scheduler-minikube                1/1     Running   0             16m
kube-system   storage-provisioner                    1/1     Running   1 (16m ago)   16m
tlxp@ip-172-31-34-212:~$ kubectl get pods -A
NAMESPACE     NAME                                   READY   STATUS    RESTARTS      AGE
default       nginx-66686b6766-z2scz                 1/1     Running   0             42s
kube-system   coredns-66bc5c9577-69d2s               1/1     Running   0             16m
kube-system   etcd-minikube                          1/1     Running   0             17m
kube-system   kube-apiserver-minikube                1/1     Running   0             17m
kube-system   kube-controller-manager-minikube       1/1     Running   0             17m
kube-system   kube-proxy-7cm55                       1/1     Running   0             17m
kube-system   kube-scheduler-minikube                1/1     Running   0             17m
kube-system   storage-provisioner                    1/1     Running   1 (16m ago)   17m
tlxp@ip-172-31-34-212:~$ kubectl get pods -o wide --all-namespaces
NAMESPACE     NAME                                   READY   STATUS    RESTARTS      AGE   IP             NODE
NOMINATED NODE   READINESS GATES
default       nginx-66686b6766-z2scz                 1/1     Running   0             54s   10.244.0.3     minikube
<none>           <none>
kube-system   coredns-66bc5c9577-69d2s               1/1     Running   0             17m   10.244.0.2     minikube
<none>           <none>
kube-system   etcd-minikube                          1/1     Running   0             17m   192.168.49.2   minikube
<none>           <none>
kube-system   kube-apiserver-minikube                1/1     Running   0             17m   192.168.49.2   minikube
<none>           <none>
kube-system   kube-controller-manager-minikube       1/1     Running   0             17m   192.168.49.2   minikube
<none>           <none>
kube-system   kube-proxy-7cm55                       1/1     Running   0             17m   192.168.49.2   minikube
<none>           <none>
kube-system   kube-scheduler-minikube                1/1     Running   0             17m   192.168.49.2   minikube
<none>           <none>
kube-system   storage-provisioner                    1/1     Running   1 (16m ago)   17m   192.168.49.2   minikube
<none>           <none>
tlxp@ip-172-31-34-212:~$ alias k=kubectl
tlxp@ip-172-31-34-212:~$
```

Create a Pod from YAML

```
tlxp@ip-172-31-34-212:~$ cat ~/yaml_files/my-5gc-pod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: my-5gc-pod
  labels:
    app: my-5gc
spec:
  containers:
    - name: my-5gc-container
      image: nginx:latest
      ports:
        - containerPort: kubectl apply -f ~/yaml_files/my-5gc-pod.yaml/yaml_files/my-5gc-pod.yaml
pod/my-5gc-pod unchanged
tlxp@ip-172-31-34-212:~$ kubectl get pods
NAME                    READY   STATUS    RESTARTS   AGE
my-5gc-pod              1/1     Running   0          3m27s
nginx-66686b6766-z2scz  1/1     Running   0          5m10s
tlxp@ip-172-31-34-212:~$ kubectl apply -f ~/yaml_files/my-5gc-pod.yaml
pod/my-5gc-pod unchanged
```

```
tlxp@ip-172-31-34-212:~$ kubectl describe pod my-5gc-pod
Name:             my-5gc-pod
Namespace:        default
Priority:         0
Service Account:  default
Node:             minikube/192.168.49.2
Start Time:       Wed, 31 Dec 2025 06:22:55 +0000
Labels:           app=my-5gc
Annotations:      <none>
Status:           Running
IP:               10.244.0.4
IPs:
  IP:  10.244.0.4
Containers:
  my-5gc-container:
    Container ID:   docker://a7e891426f1cb28ecea69ce8409b1f964c6f29a4a84fb49b4f0293700ab2ec4b
    Image:          nginx:latest
    Image ID:       docker-pullable://nginx@sha256:ca871a86d45a3ec6864dc45f014b11fe626145569ef0e74deaffc95a3b15b4
30
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Wed, 31 Dec 2025 06:22:58 +0000
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-hdqjz (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       True
  ContainersReady             True
  PodScheduled                True
Volumes:
  kube-api-access-hdqjz:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    Optional:                false
```

```
tlxp@ip-172-31-34-212:~$ kubectl exec -it my-5gc-pod -- /bin/bash
root@my-5gc-pod:/# hostname
my-5gc-pod
root@my-5gc-pod:/# exit
exit
tlxp@ip-172-31-34-212:~$
```

Exploring K8s Services

```
tlxp@ip-172-31-34-212:~$ kubectl get pod my-5gc-pod
NAME         READY   STATUS    RESTARTS   AGE
my-5gc-pod   1/1     Running   0          10m
tlxp@ip-172-31-34-212:~$ kubectl expose pod my-5gc-pod --type=ClusterIP --port=80 --target-port=80 --name=my
5gc-svc-clusterip
service/my exposed
5gc-svc-clusterip: command not found
tlxp@ip-172-31-34-212:~$ kubectl get svc
NAME                  TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes            ClusterIP      10.96.0.1       <none>        443/TCP        28m
my                    ClusterIP      10.98.202.47    <none>        80/TCP         15s
my-5gc-svc-clusterip  ClusterIP      10.97.91.231    <none>        80/TCP         5m45s
my-5gc-svc-lb         LoadBalancer   10.97.6.134     <pending>     80:32455/TCP   4m16s
my-5gc-svc-nodeport   NodePort       10.98.59.219    <none>        80:31397/TCP   4m50s
tlxp@ip-172-31-34-212:~$ kubectl expose pod my-5gc-pod --type=ClusterIP --port=80 --target-port=80 --name=my-5gc-
svc-clusterip
Error from server (AlreadyExists): services "my-5gc-svc-clusterip" already exists
tlxp@ip-172-31-34-212:~$ kubectl delete svc my
service "my" deleted from default namespace
tlxp@ip-172-31-34-212:~$ kubectl expose pod my-5gc-pod --type=ClusterIP --port=80 --target-port=80 --name=my-5gc-
svc-clusterip
Error from server (AlreadyExists): services "my-5gc-svc-clusterip" already exists
tlxp@ip-172-31-34-212:~$ kubectl expose pod my-5gc-pod --type=NodePort --port=80 --target-port=80 --name=my-5gc-s
vc-nodeport
Error from server (AlreadyExists): services "my-5gc-svc-nodeport" already exists
tlxp@ip-172-31-34-212:~$ kubectl get svc
NAME                  TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
kubernetes            ClusterIP      10.96.0.1       <none>        443/TCP        32m
my-5gc-svc-clusterip  ClusterIP      10.97.91.231    <none>        80/TCP         8m55s
my-5gc-svc-lb         LoadBalancer   10.97.6.134     <pending>     80:32455/TCP   7m26s
my-5gc-svc-nodeport   NodePort       10.98.59.219    <none>        80:31397/TCP   8m
tlxp@ip-172-31-34-212:~$ kubectl expose pod my-5gc-pod --type=LoadBalancer --port=80 --target-port=80 -name=my-5g
c-svc-lb
Error from server (NotFound): namespaces "ame=my-5gc-svc-lb" not found
tlxp@ip-172-31-34-212:~$ kubectl expose pod my-5gc-pod --type=LoadBalancer --port=80 --target-port=80 --name=my-5
gc-svc-lb
Error from server (AlreadyExists): services "my-5gc-svc-lb" already exists
tlxp@ip-172-31-34-212:~$ kubectl get svc my-5gc-svc-lb
NAME           TYPE           CLUSTER-IP     EXTERNAL-IP   PORT(S)        AGE
my-5gc-svc-lb  LoadBalancer   10.97.6.134    <pending>     80:32455/TCP   10m
```

```
tlxp@ip-172-31-34-212:~$ nano my-5gc-externalname.yaml
tlxp@ip-172-31-34-212:~$ ls
kubectl  kubectl.sha256  minikube-linux-amd64  my-5gc-externalname.yaml  ttyd.log  yaml_files
tlxp@ip-172-31-34-212:~$ kubectl apply -f my-5gc-externalname.yaml
error: error when retrieving current configuration of:
Resource: "/v1, Resource=services", GroupVersionKind: "/v1, Kind=Service"
Name: "", Namespace: "default"
from server for: "my-5gc-externalname.yaml": resource name may not be empty
tlxp@ip-172-31-34-212:~$ nano my-5gc-externalname.yaml
tlxp@ip-172-31-34-212:~$ kubectl apply -f my-5gc-externalname.yaml
service/my-5gc-externalname created
tlxp@ip-172-31-34-212:~$
```

K8s Workloads

Create Multiple Pods using a ReplicaSet (File: my-5gc-replicaset.yaml)

```
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl apply -f my-5gc-replicaset.yaml
replicaset.apps/my-5gc-rs created
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get replicasets
NAME              DESIRED   CURRENT   READY   AGE
my-5gc-rs         3         3         3       8s
nginx-66686b6766  1         1         1       35m
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods -l app=my-5gc
NAME              READY   STATUS    RESTARTS   AGE
my-5gc-rs-g48x6   1/1     Running   0          16s
my-5gc-rs-mcm4m   1/1     Running   0          16s
my-5gc-rs-tvn9l   1/1     Running   0          16s
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl delete pod my-5gc-rs-g48x6
pod "my-5gc-rs-g48x6" deleted from default namespace
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods -l app=my-5gc
NAME              READY   STATUS              RESTARTS   AGE
my-5gc-rs-mcm4m   1/1     Running             0          83s
my-5gc-rs-tsdfd   0/1     ContainerCreating   0          3s
my-5gc-rs-tvn9l   1/1     Running             0          83s
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl delete -f my-5gc-replicaset.yaml
replicaset.apps "my-5gc-rs" deleted from default namespace
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods
NAME                    READY   STATUS    RESTARTS   AGE
nginx-66686b6766-z2scz  1/1     Running   0          36m
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get replicasets
NAME              DESIRED   CURRENT   READY   AGE
nginx-66686b6766  1         1         1       36m
tlxp@ip-172-31-34-212:~/yaml_files$ ls
my-5gc-clusterip.yaml   my-5gc-ingress.yaml    my-5gc-pod.yaml         nginx-deployment.yaml
my-5gc-configmap.yaml   my-5gc-namespace.yaml  my-5gc-replicaset.yaml
my-5gc-deployment.yaml  my-5gc-nodeport.yaml   my-5gc-secret.yaml
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl apply -f my-5gc-deployment.yaml
deployment.apps/my-5gc-deployment created
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
my-5gc-deployment  3/3     3            3           8s
nginx              1/1     1            1           37m
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get ReplicaSet
NAME                          DESIRED   CURRENT   READY   AGE
my-5gc-deployment-6d6cf88fd   3         3         3       17s
nginx-66686b6766              1         1         1       37m
```

```
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods -l app=my-5gc
NAME                                READY   STATUS    RESTARTS   AGE
my-5gc-deployment-6d6cf88fd-9jlbs   1/1     Running   0          42s
my-5gc-deployment-6d6cf88fd-mv55d   1/1     Running   0          42s
my-5gc-deployment-6d6cf88fd-vm4js   1/1     Running   0          42s
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl scale deployment my-5gc-deployment --replicas=6
deployment.apps/my-5gc-deployment scaled
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
my-5gc-deployment-6d6cf88fd-8d7dq   1/1     Running   0          7s
my-5gc-deployment-6d6cf88fd-9jlbs   1/1     Running   0          2m49s
my-5gc-deployment-6d6cf88fd-f5rqb   1/1     Running   0          7s
my-5gc-deployment-6d6cf88fd-mv55d   1/1     Running   0          2m49s
my-5gc-deployment-6d6cf88fd-tlqcm   1/1     Running   0          7s
my-5gc-deployment-6d6cf88fd-vm4js   1/1     Running   0          2m49s
nginx-66686b6766-z2scz              1/1     Running   0          40m
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get deployment my-5gc-deployment
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
my-5gc-deployment   6/6     6            6           2m57s
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl delete deployment my-5gc-deployment
deployment.apps "my-5gc-deployment" deleted from default namespace
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get deployment my-5gc-deployment
Error from server (NotFound): deployments.apps "my-5gc-deployment" not found
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get deployments
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
nginx   1/1     1            1           41m
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-66686b6766-z2scz   1/1     Running   0          41m
tlxp@ip-172-31-34-212:~/yaml_files$
```

Kubernetes Networking Exercise

```
tlxp@ip-172-31-34-212:~/yaml_files$ nano nginx-deployment.yaml
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx-deploy created
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods -l app=nginx
NAME                            READY   STATUS    RESTARTS   AGE
nginx-66686b6766-z2scz          1/1     Running   0          58m
nginx-deploy-77bf8679f9-brd9w   1/1     Running   0          61s
nginx-deploy-77bf8679f9-ps2zb   1/1     Running   0          61s
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl expose deployment nginx-deploy \  --port=80 \  --target-port=80 \ --t
ype=ClusterIP \  --name=nginx-service
Error from server (AlreadyExists): services "nginx-service" already exists
Error from server (NotFound): deployments.apps " " not found
Error from server (NotFound): deployments.apps " " not found
Error from server (NotFound): deployments.apps " --type=ClusterIP" not found
Error from server (NotFound): deployments.apps " " not found
```

```
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl expose deployment nginx-deploy \
  --port=80 \
  --target-port=80 \
  --type=ClusterIP \
  --name=nginx-service
Error from server (AlreadyExists): services "nginx-deploy" already exists
Error from server (NotFound): deployments.apps " " not found
--port=80: command not found
--target-port=80: command not found
--type=ClusterIP: command not found
--name=nginx-service: command not found
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl delete service nginx-deploy
service "nginx-deploy" deleted from default namespace
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl expose deployment nginx-deploy --port=80 --target-port=80 --type=Clus
terIP --name=nginx-service
Error from server (AlreadyExists): services "nginx-service" already exists
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get svc nginx-service
NAME            TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   AGE
nginx-service   ClusterIP   10.98.185.126   <none>        80/TCP    41m
```

```
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl run tester --image=busybox --restart=Never -it --rm --command -- sh
All commands and output from this session will be recorded in container logs, including credentials and sensitive
 information passed through the command prompt.
If you don't see a command prompt, try pressing enter.
/ # wget -qO- http://nginx-service
<!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>
/ # exit
pod "tester" deleted from default namespace
tlxp@ip-172-31-34-212:~/yaml_files$
```

ConfigMap and Secrets

```
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl apply -f my-5gc-configmap.yaml
configmap/my-5gc-config created
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pod
NAME                            READY   STATUS    RESTARTS   AGE
nginx-66686b6766-z2scz          1/1     Running   0          115m
nginx-deploy-77bf8679f9-brd9w   1/1     Running   0          57m
nginx-deploy-77bf8679f9-ps2zb   1/1     Running   0          57m
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get config
error: the server doesn't have a resource type "config"
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get configmap
NAME               DATA   AGE
kube-root-ca.crt   1      131m
my-5gc-config      2      17s
tlxp@ip-172-31-34-212:~/yaml_files$ cat my-5gc-secret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: my-5gc-secret
type: Opaque
data:
  DB_PASSWORD: cGFzc3dvcmQxMjM=  # base64 encoded 'password123'tlxp@ip-172-31-34-212:~/yaml_files$ kubectl apply
kubectl apply -f my-5gc-secret.yaml
secret/my-5gc-secret unchanged
```

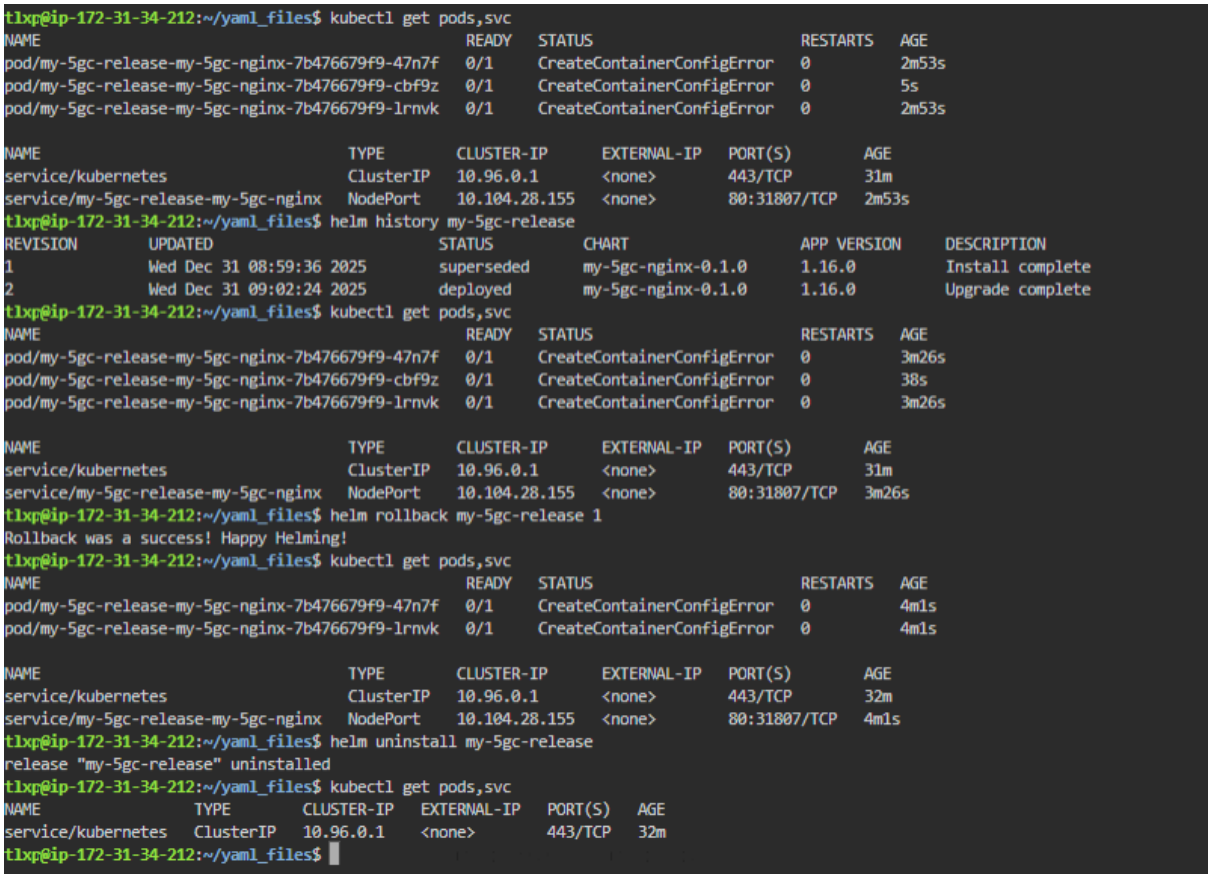
```
  DB_PASSWORD: cGFzc3dvcmQxMjM=  # base64 encoded 'password123'tlxp@ip-172-31-34-212:~/yaml_files$ kubectl apply
kubectl apply -f my-5gc-secret.yaml
secret/my-5gc-secret unchanged
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get secret my-5gc-secret -o yaml
apiVersion: v1
data:
  DB_PASSWORD: cGFzc3dvcmQxMjM=
kind: Secret
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","data":{"DB_PASSWORD":"cGFzc3dvcmQxMjM="},"kind":"Secret","metadata":{"annotations":{},"
name":"my-5gc-secret","namespace":"default"},"type":"Opaque"}
  creationTimestamp: "2025-12-31T07:26:37Z"
  name: my-5gc-secret
  namespace: default
  resourceVersion: "5295"
  uid: 396cc781-9c85-4797-bdf7-3117a43c3b54
type: Opaque
tlxp@ip-172-31-34-212:~/yaml_files$
```

Application Packaging & Deployment with Helm

```
tlxp@ip-172-31-34-212:~/yaml_files$ curl https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11929  100 11929    0     0   155k      0 --:--:-- --:--:-- --:--:--  157k
Helm v3.19.4 is already latest
tlxp@ip-172-31-34-212:~/yaml_files$ helm version
version.BuildInfo{Version:"v3.19.4", GitCommit:"7cfb6e486dac026202556836bb910c37d847793e", GitTreeState:"clean", GoVersion:"go1.24.1
1"}
tlxp@ip-172-31-34-212:~/yaml_files$ helm create my-5gc-nginx
Creating my-5gc-nginx
tlxp@ip-172-31-34-212:~/yaml_files$ ls
my-5gc-clusterip.yaml   my-5gc-ingress.yaml    my-5gc-nodeport.yaml    my-5gc-secret.yaml
my-5gc-configmap.yaml   my-5gc-namespace.yaml  my-5gc-pod.yaml         nginx-deployment.yaml
my-5gc-deployment.yaml  my-5gc-nginx           my-5gc-replicaset.yaml
tlxp@ip-172-31-34-212:~/yaml_files$ nano my-5gc-nginx/values.yaml
```

```
tlxp@ip-172-31-34-212:~/yaml_files$ helm install my-5gc-release ./my-5gc-nginx
Error: INSTALLATION FAILED: cannot re-use a name that is still in use
tlxp@ip-172-31-34-212:~/yaml_files$ helm delete my-5gc-release ./my-5gc-nginx
release "my-5gc-release" uninstalled
Error: uninstall: Release name is invalid: ./my-5gc-nginx
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get cm,secret
NAME                         DATA   AGE
configmap/kube-root-ca.crt   1      31m
tlxp@ip-172-31-34-212:~/yaml_files$ helm install my-5gc-release ./my-5gc-nginx
NAME: my-5gc-release
LAST DEPLOYED: Wed Dec 31 08:59:36 2025
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services my-5gc-release-my-5gc-nginx)
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0].address}")
  echo http://$NODE_IP:$NODE_PORT
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods,svc
NAME                                                READY   STATUS                       RESTARTS   AGE
pod/my-5gc-release-my-5gc-nginx-7b476679f9-47n7f    0/1     CreateContainerConfigError   0          40s
pod/my-5gc-release-my-5gc-nginx-7b476679f9-lrnvk    0/1     CreateContainerConfigError   0          40s

NAME                                  TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)        AGE
service/kubernetes                    ClusterIP   10.96.0.1       <none>        443/TCP        29m
service/my-5gc-release-my-5gc-nginx   NodePort    10.104.28.155   <none>        80:31807/TCP   40s
tlxp@ip-172-31-34-212:~/yaml_files$ nano my-5gc-nginx/values.yaml
tlxp@ip-172-31-34-212:~/yaml_files$ helm upgrade my-5gc-release ./my-5gc-nginx
Release "my-5gc-release" has been upgraded. Happy Helming!
NAME: my-5gc-release
LAST DEPLOYED: Wed Dec 31 09:02:24 2025
NAMESPACE: default
STATUS: deployed
REVISION: 2
NOTES:
1. Get the application URL by running these commands:
  export NODE_PORT=$(kubectl get --namespace default -o jsonpath="{.spec.ports[0].nodePort}" services my-5gc-release-my-5gc-nginx)
  export NODE_IP=$(kubectl get nodes --namespace default -o jsonpath="{.items[0].status.addresses[0].address}")
  echo http://$NODE_IP:$NODE_PORT
tlxp@ip-172-31-34-212:~/yaml_files$ kubectl get pods,svc
```



Service Mesh with Linkerd (Basic Hands-on)