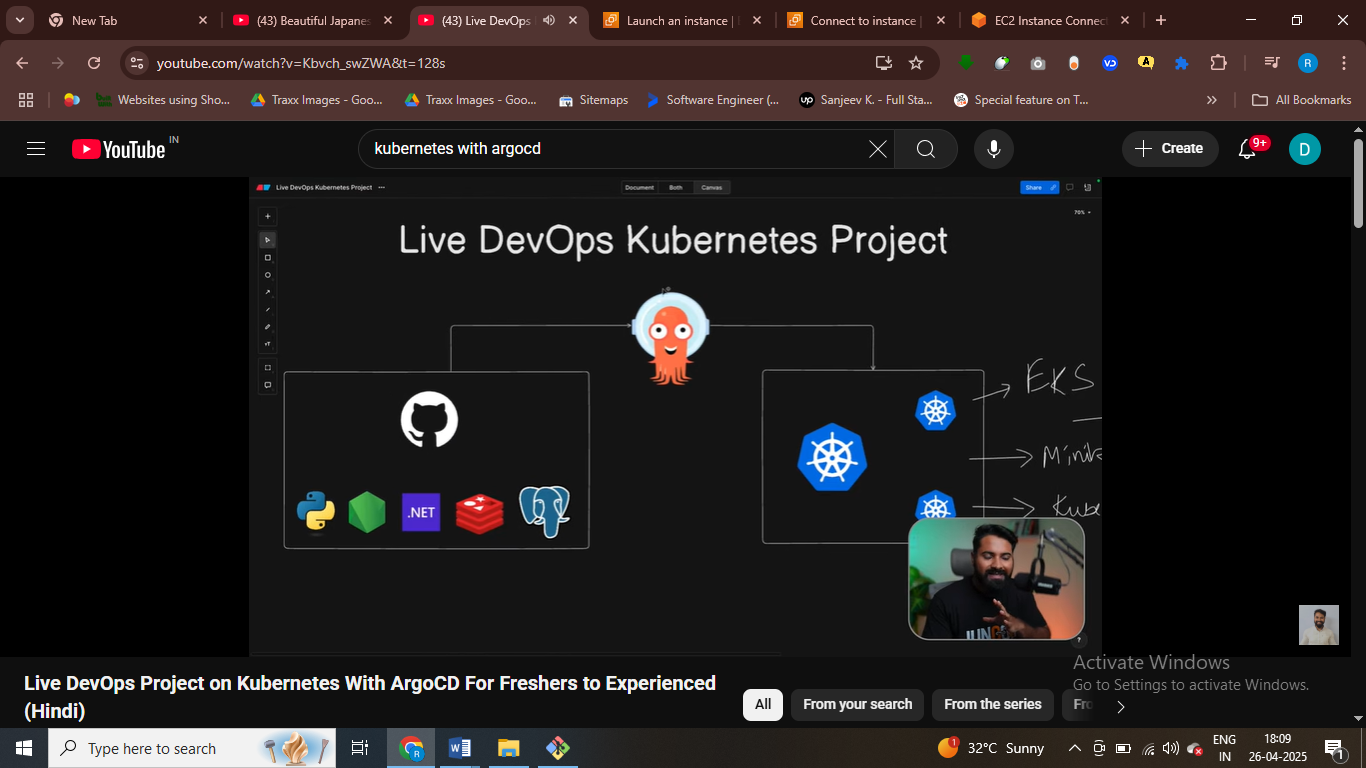
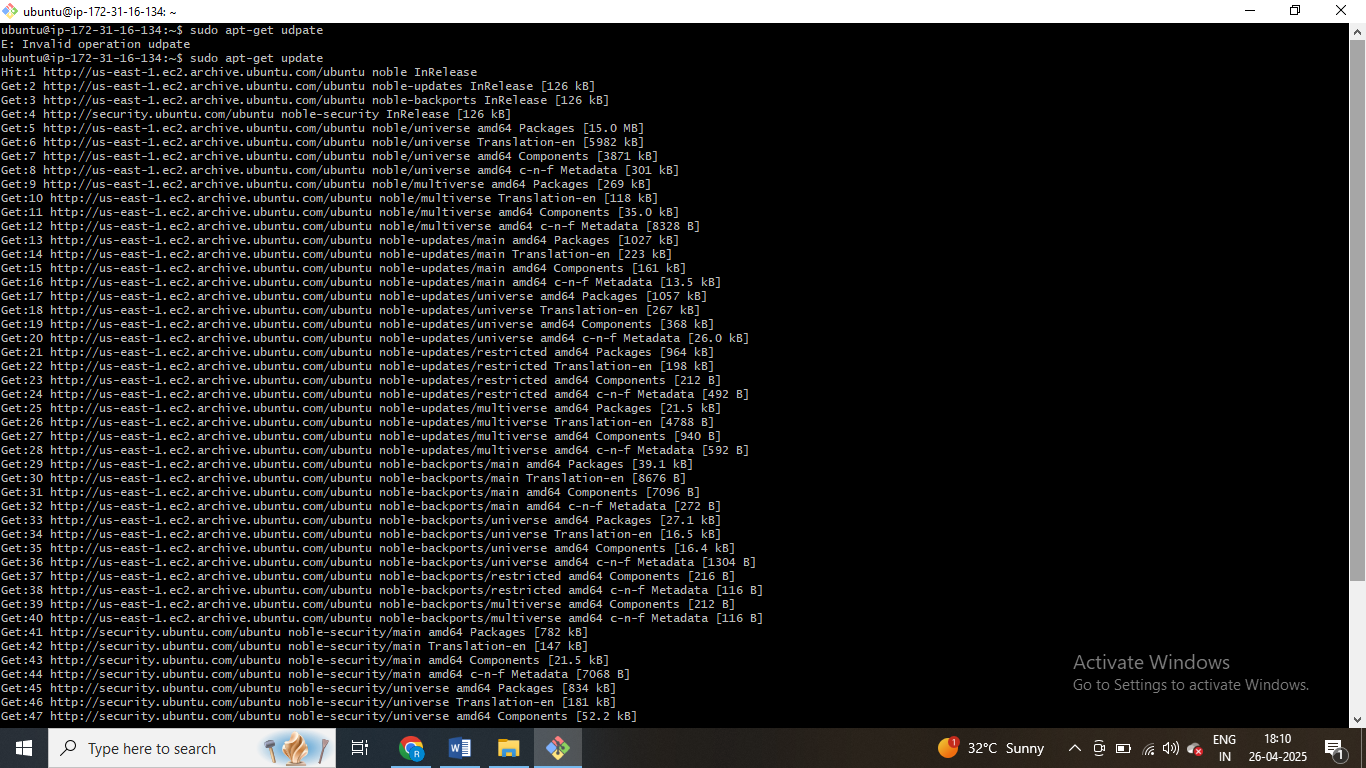
https://github.com/LondheShubham153/k8s-kind-voting-app/tree/main

sabse pahle argocd setup karna hai phir iss kuberntes cluster ko run karenge

First you need to install docker



ubuntu@ip-172-31-16-134:~$ sudo apt-get install docker.io

Reading package lists... Done

vubuntu@ip-172-31-16-134:~$ sudo usermod -aG docker $USER && newgrp docker

ubuntu@ip-172-31-16-134:~$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

ubuntu@ip-172-31-16-134:~$

install kuberntes in dockre

#!/bin/bash

# For AMD64 / x86\_64

[ $(uname -m) = x86\_64 ] && curl -Lo ./kind https://kind.sigs.k8s.io/dl/v0.20.0/kind-linux-amd64

chmod +x ./kind

sudo cp ./kind /usr/local/bin/kind

rm -rf kind

No VM guests are running outdated hypervisor (qemu) binaries on this host.

ubuntu@ip-172-31-16-134:~$ sudo usermod -aG docker $USER && newgrp docker

ubuntu@ip-172-31-16-134:~$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

ubuntu@ip-172-31-16-134:~$ mkdir k8-install

ubuntu@ip-172-31-16-134:~$ cd k8-install

ubuntu@ip-172-31-16-134:~/k8-install$ vim install\_kind.sh

ubuntu@ip-172-31-16-134:~/k8-install$ chmod +x install\_kind.sh

ubuntu@ip-172-31-16-134:~/k8-install$ ./install\_kind.sh

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 97 100 97 0 0 2472 0 --:--:-- --:--:-- --:--:-- 2487

0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0

100 6304k 100 6304k 0 0 26.2M 0 --:--:-- --:--:-- --:--:-- 26.2M

ubuntu@ip-172-31-16-134:~/k8-install$

#contorl panel that container everything

kind: Cluster

apiVersion: kind.x-k8s.io/v1alpha4

nodes:

- role: control-plane

image: kindest/node:v1.30.0

- role: worker

image: kindest/node:v1.30.0

- role: worker

image: kindest/node:v1.30.0

100 6304k 100 6304k 0 0 26.2M 0 --:--:-- --:--:-- --:--:-- 26.2M

ubuntu@ip-172-31-16-134:~/k8-install$ vim config.yaml

ubuntu@ip-172-31-16-134:~/k8-install$ kind create cluster --config=config.yml

ERROR: failed to create cluster: error reading file: open config.yml: no such file or directory

ubuntu@ip-172-31-16-134:~/k8-install$ vim config.yaml

ubuntu@ip-172-31-16-134:~/k8-install$ kind create cluster --config=config.yml --name=my-cluster

ERROR: failed to create cluster: error reading file: open config.yml: no such file or directory

ubuntu@ip-172-31-16-134:~/k8-install$ mv config.yaml config.yml

ubuntu@ip-172-31-16-134:~/k8-install$ kind create cluster --config=config.yml --name=my-cluster

Creating cluster "my-cluster" ...

✓ Ensuring node image (kindest/node:v1.30.0) 🖼

✓ Preparing nodes 📦 📦 📦

✓ Writing configuration 📜

✓ Starting control-plane 🕹️

✓ Installing CNI 🔌

✓ Installing StorageClass 💾

✓ Joining worker nodes 🚜

Set kubectl context to "kind-my-cluster"

You can now use your cluster with:

kubectl cluster-info --context kind-my-cluster

Have a question, bug, or feature request? Let us know! https://kind.sigs.k8s.io/#community 🙂

ubuntu@ip-172-31-16-134:~/k8-install$

# control panel will contain everything like api server , etcd , and everything

# it will manage the worker nodes

Have a question, bug, or feature request? Let us know! https://kind.sigs.k8s.io/#community 🙂

#!/bin/bash

# Variables

VERSION="v1.30.0"

URL="https://dl.k8s.io/release/${VERSION}/bin/linux/amd64/kubectl"

INSTALL\_DIR="/usr/local/bin"

# Download and install kubectl

curl -LO "$URL"

chmod +x kubectl

sudo mv kubectl $INSTALL\_DIR/

kubectl version --client

# Clean up

rm -f kubectl

echo "kubectl installation complete."

ubuntu@ip-172-31-16-134:~/k8-install$ vim install\_kubectl.sh

ubuntu@ip-172-31-16-134:~/k8-install$ chmod +x install\_kubectl.sh

ubuntu@ip-172-31-16-134:~/k8-install$ ./install\_kubectl.sh

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 138 100 138 0 0 2061 0 --:--:-- --:--:-- --:--:-- 2090

100 49.0M 100 49.0M 0 0 63.9M 0 --:--:-- --:--:-- --:--:-- 63.9M

Client Version: v1.30.0

Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3

kubectl installation complete.

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get nodes

NAME STATUS ROLES AGE VERSION

my-cluster-control-plane Ready control-plane 6m50s v1.30.0

my-cluster-worker Ready <none> 6m28s v1.30.0

my-cluster-worker2 Ready <none> 6m27s v1.30.0

ubuntu@ip-172-31-16-134:~/k8-install$

# we need to set up three nodes kuberntes cluster

NOW SETUP ARGOCD

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get nodes

NAME STATUS ROLES AGE VERSION

my-cluster-control-plane Ready control-plane 6m50s v1.30.0

my-cluster-worker Ready <none> 6m28s v1.30.0

my-cluster-worker2 Ready <none> 6m27s v1.30.0

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl create namespace argocd

namespace/argocd created

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml

customresourcedefinition.apiextensions.k8s.io/applications.argoproj.io created

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get pods -n argocd

NAME READY STATUS RESTARTS AGE

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get svc -n argocd

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

argocd-applicationset-controller ClusterIP 10.96.70.84 <none> 7000/TCP,8080/TCP 119s

argocd-dex-server ClusterIP 10.96.32.31 <none> 5556/TCP,5557/TCP,5558/TCP 119s

argocd-metrics ClusterIP 10.96.137.254 <none> 8082/TCP 119s

argocd-notifications-controller-metrics ClusterIP 10.96.166.161 <none> 9001/TCP 119s

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl patch svc argocd-server -n argocd -p '{"spec": {"type": "NodePort"}}'

service/argocd-server patched

ubuntu@ip-172-31-16-134:~/k8-install$ Forwarding from 127.0.0.1:8443 -> 8080

Forwarding from [::1]:8443 -> 8080

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl port-forward -n argocd service/argocd-server 8443:443 --address=0.0.0.0 &

[2] 8842

[1] Killed kubectl port-forward -n argocd service/argocd-server --address=0.0.0.0 8443:443

ubuntu@ip-172-31-16-134:~/k8-install$ Forwarding from 0.0.0.0:8443 -> 8080

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

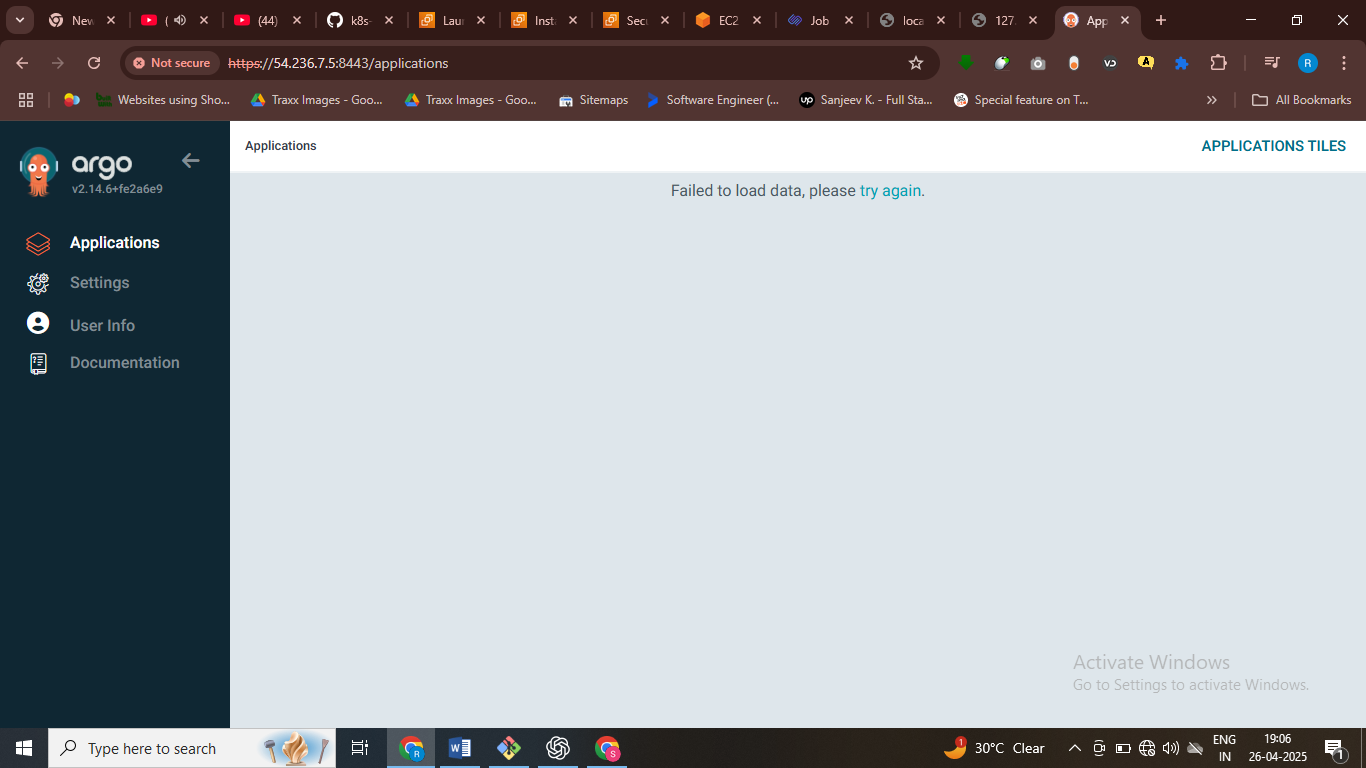
Handling connection for 8443

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get secret -n argocd argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d && echo

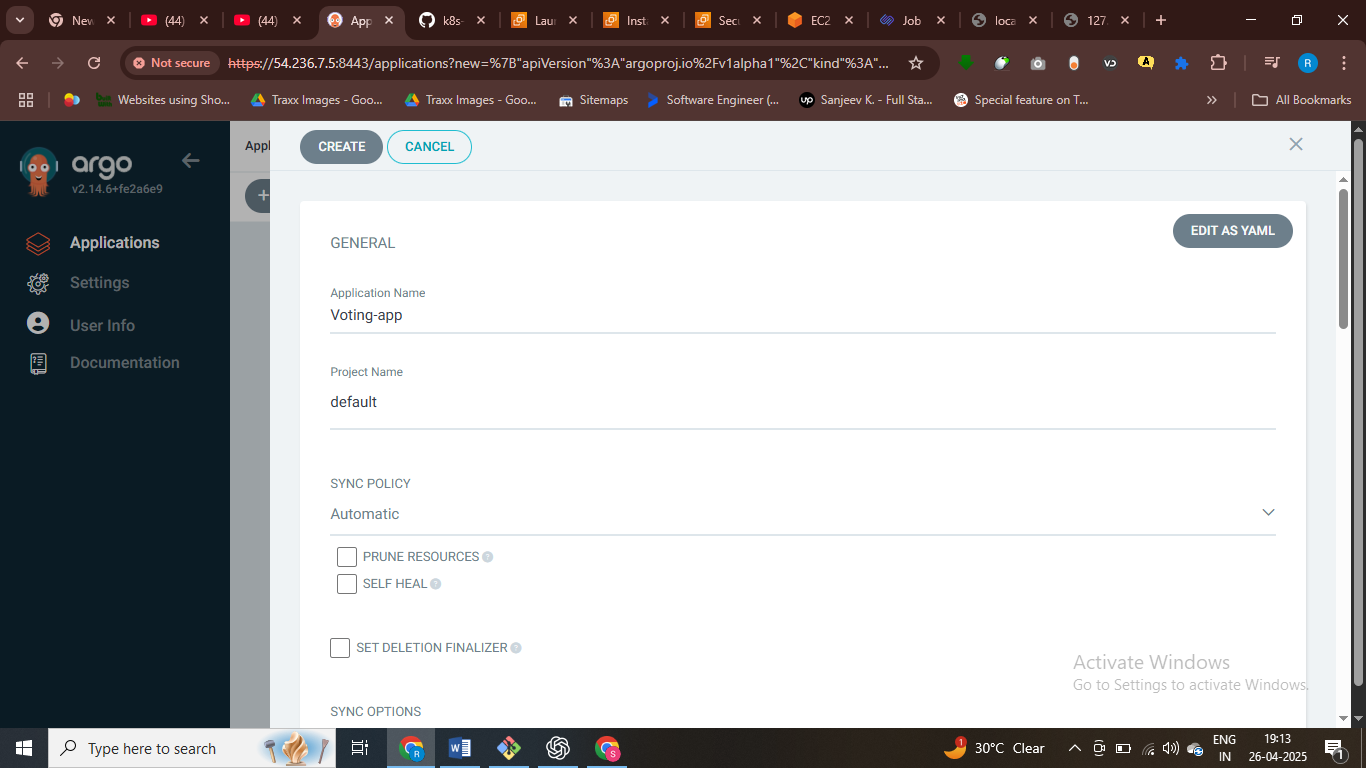
z0cypMHFdMhbx6fl

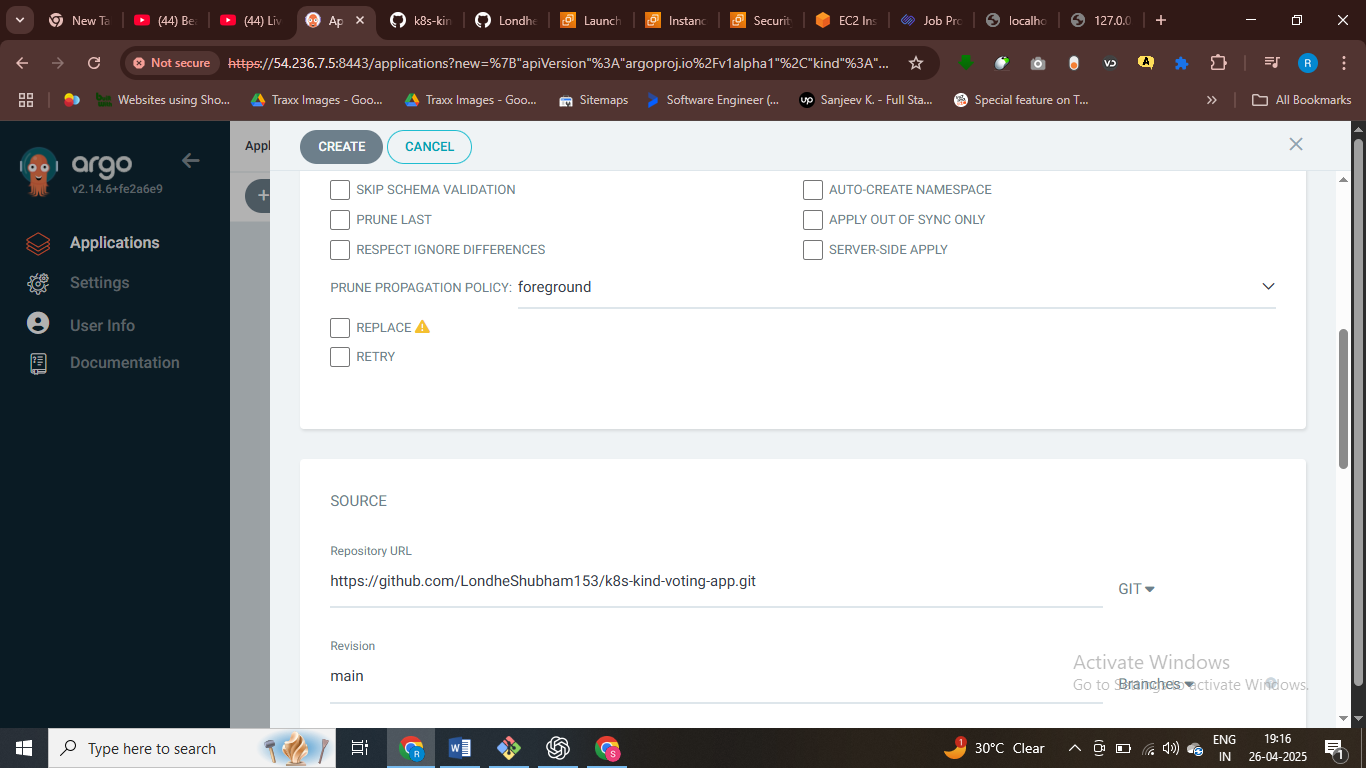
ubuntu@ip-172-31-16-1

#argo cd main cluster add karna hote hai , aur appn add karna hote hai , aur github ka sath connect karke appn deplioy karo

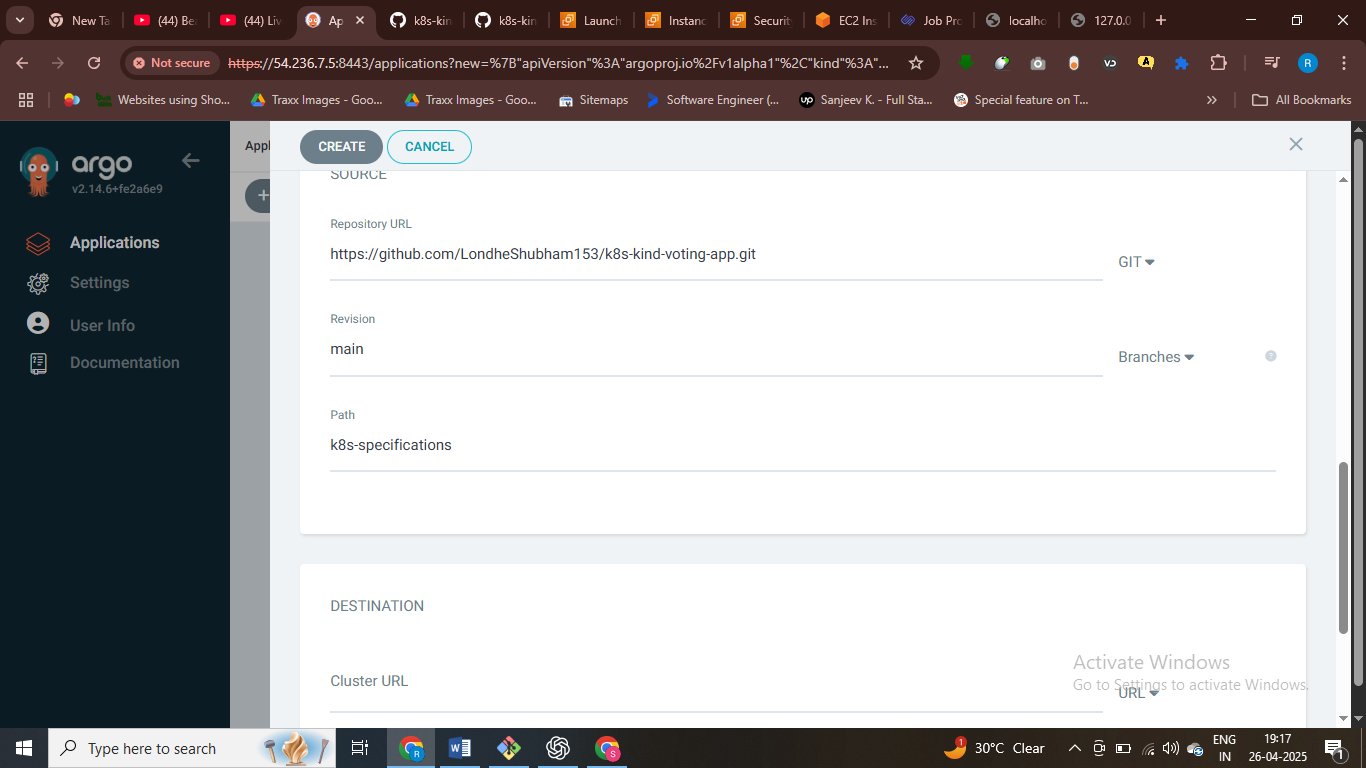


Crate new app

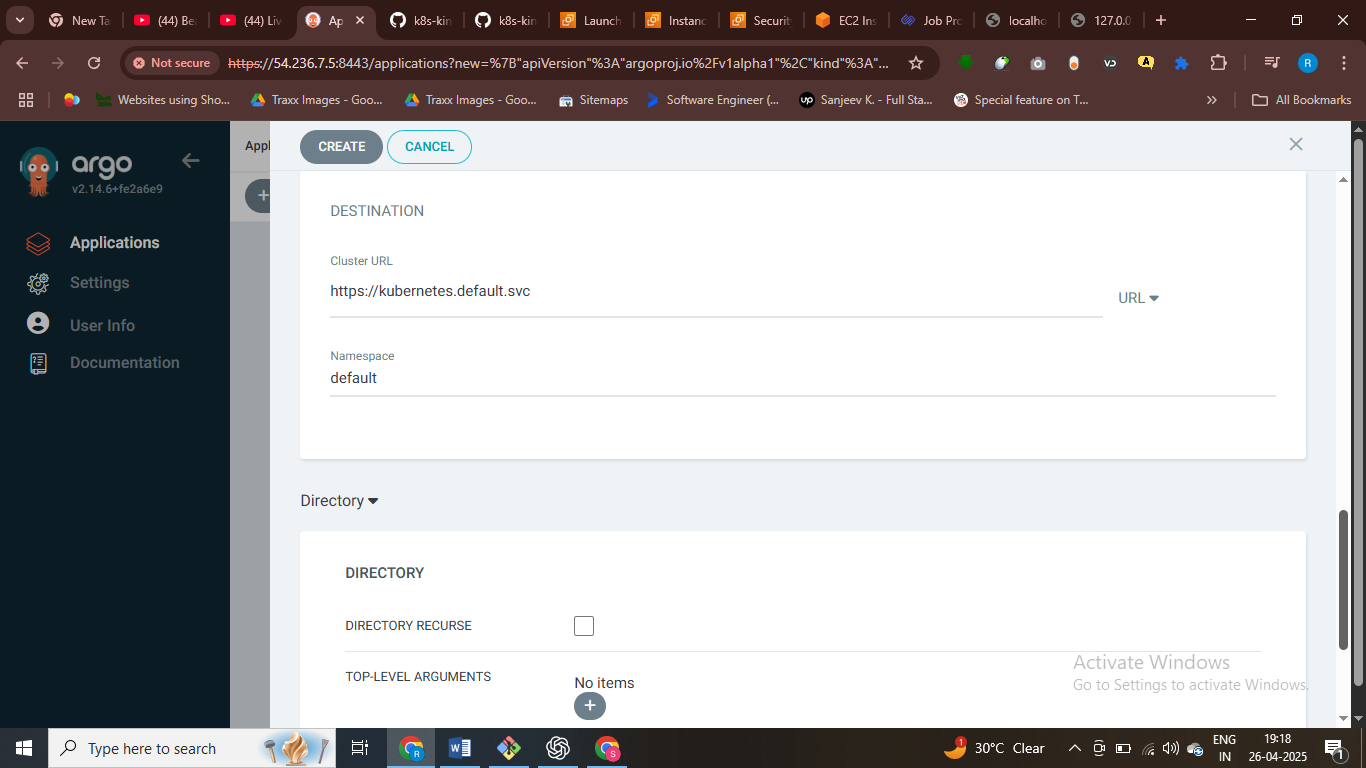




# add repo url

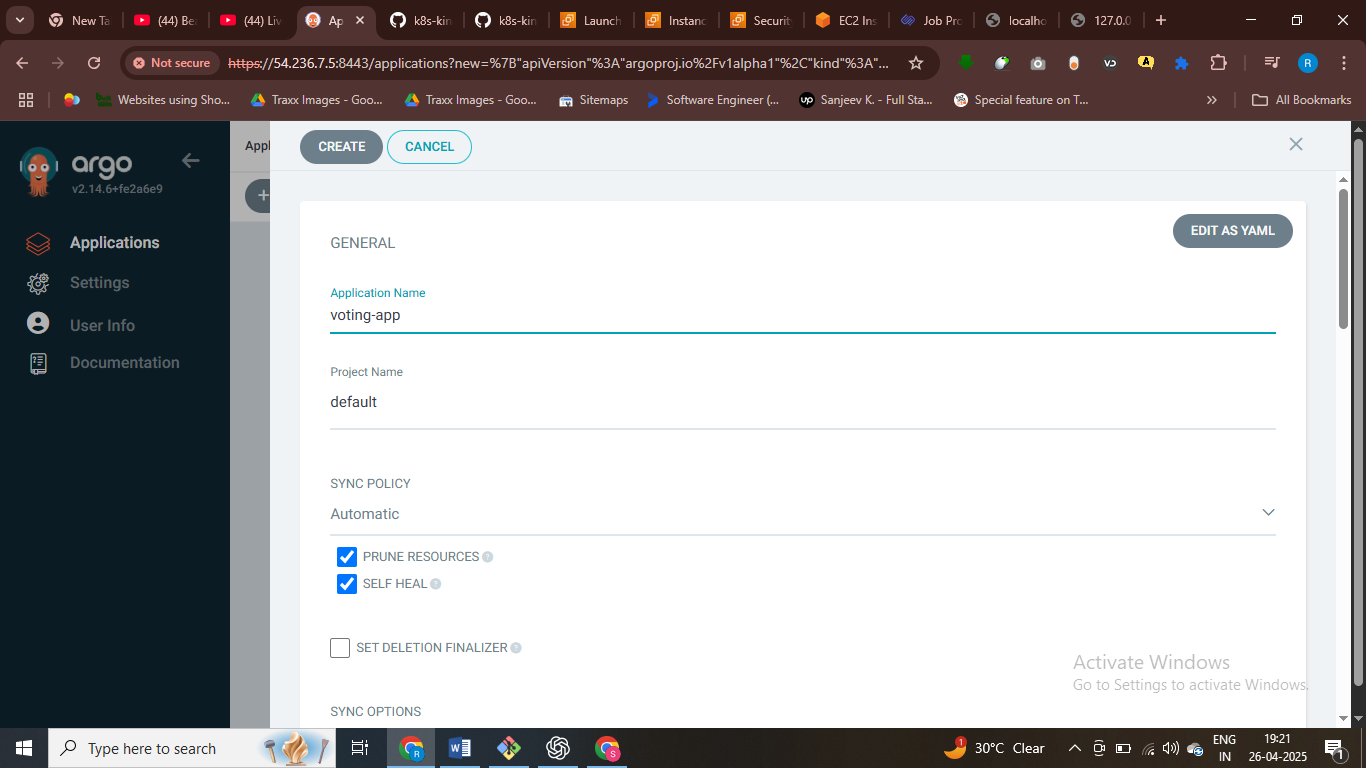


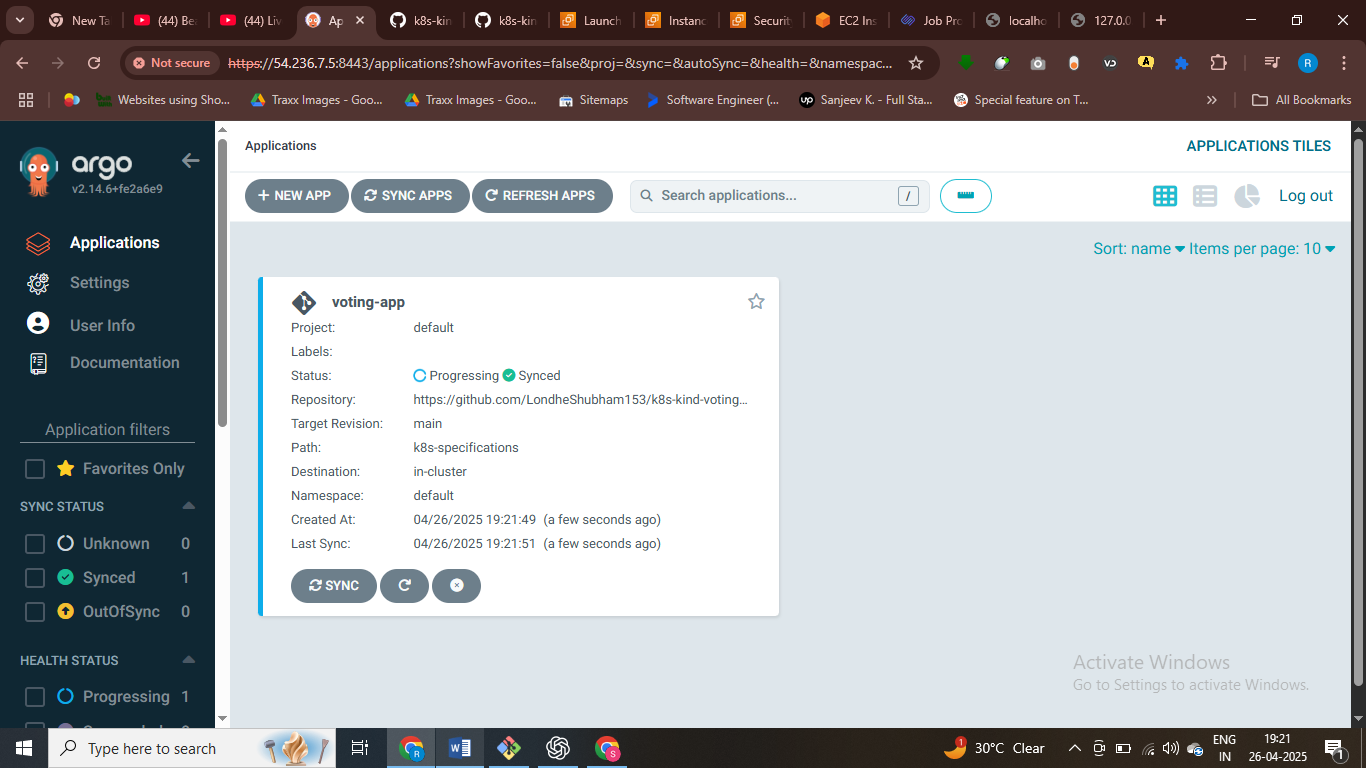
$jha kuberntes file rakhi hai who path diya hamne

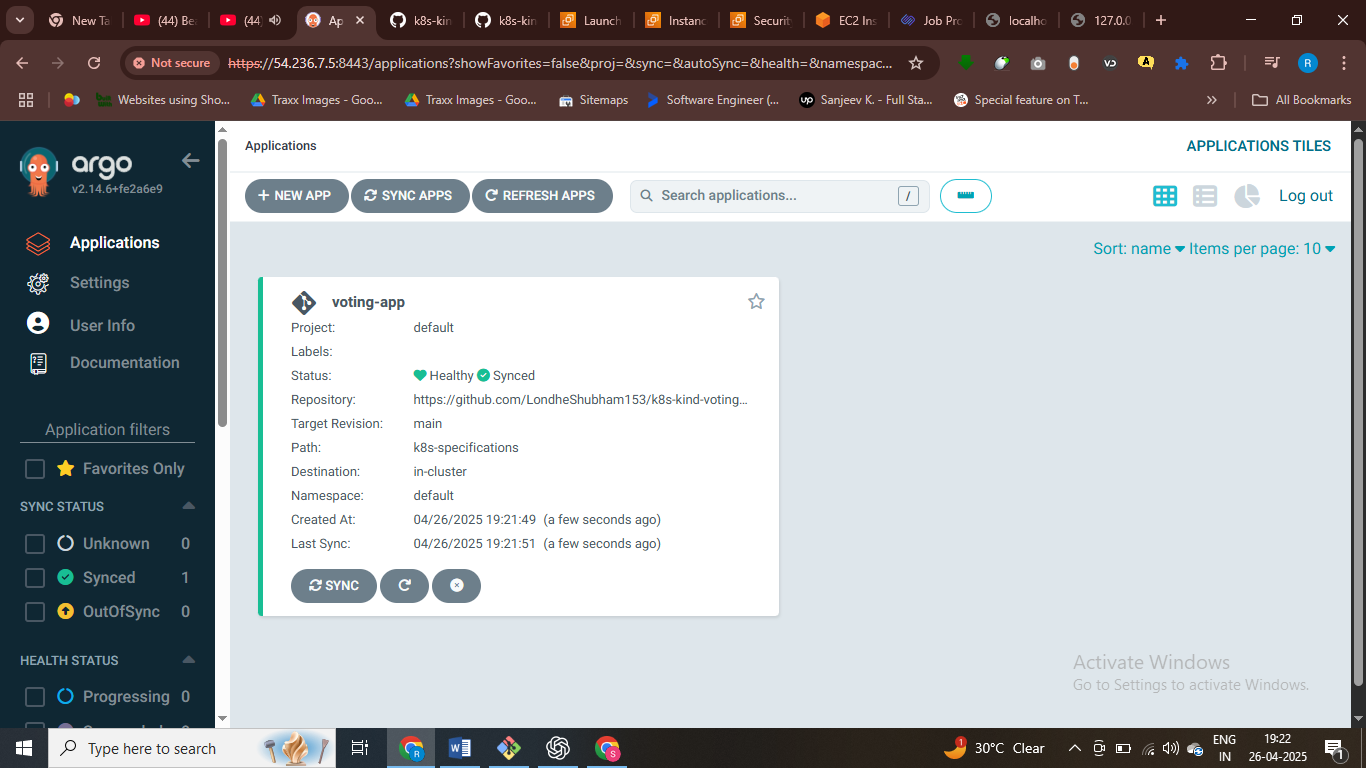


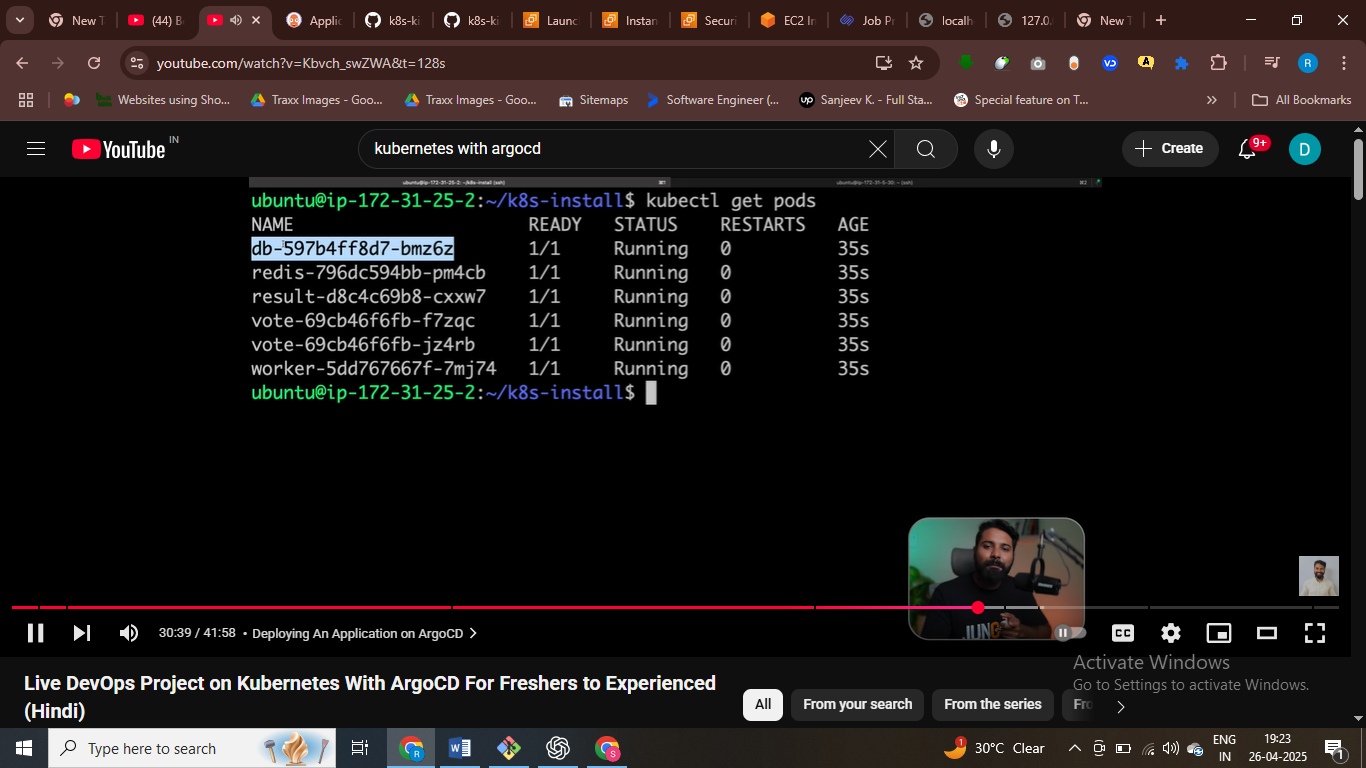
#ye source code yha deploy hoga

Now click on create

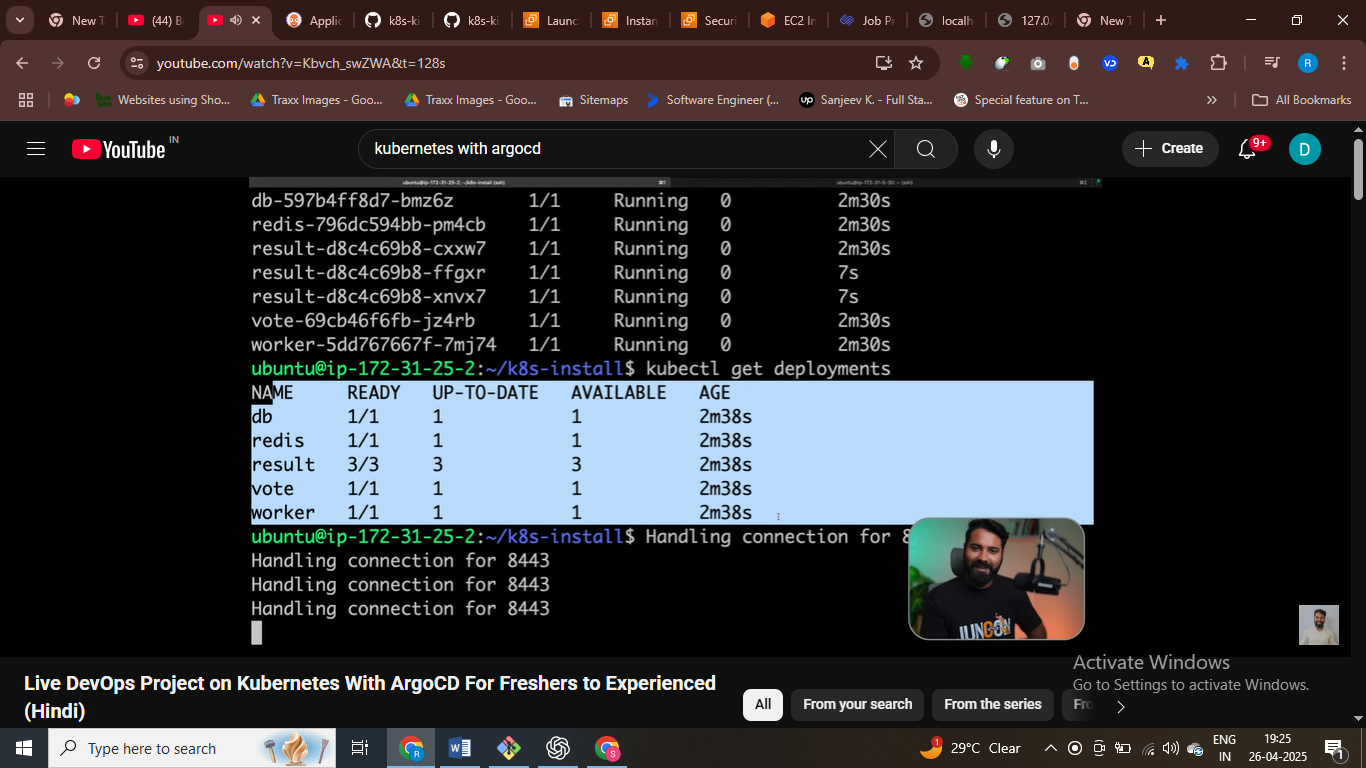




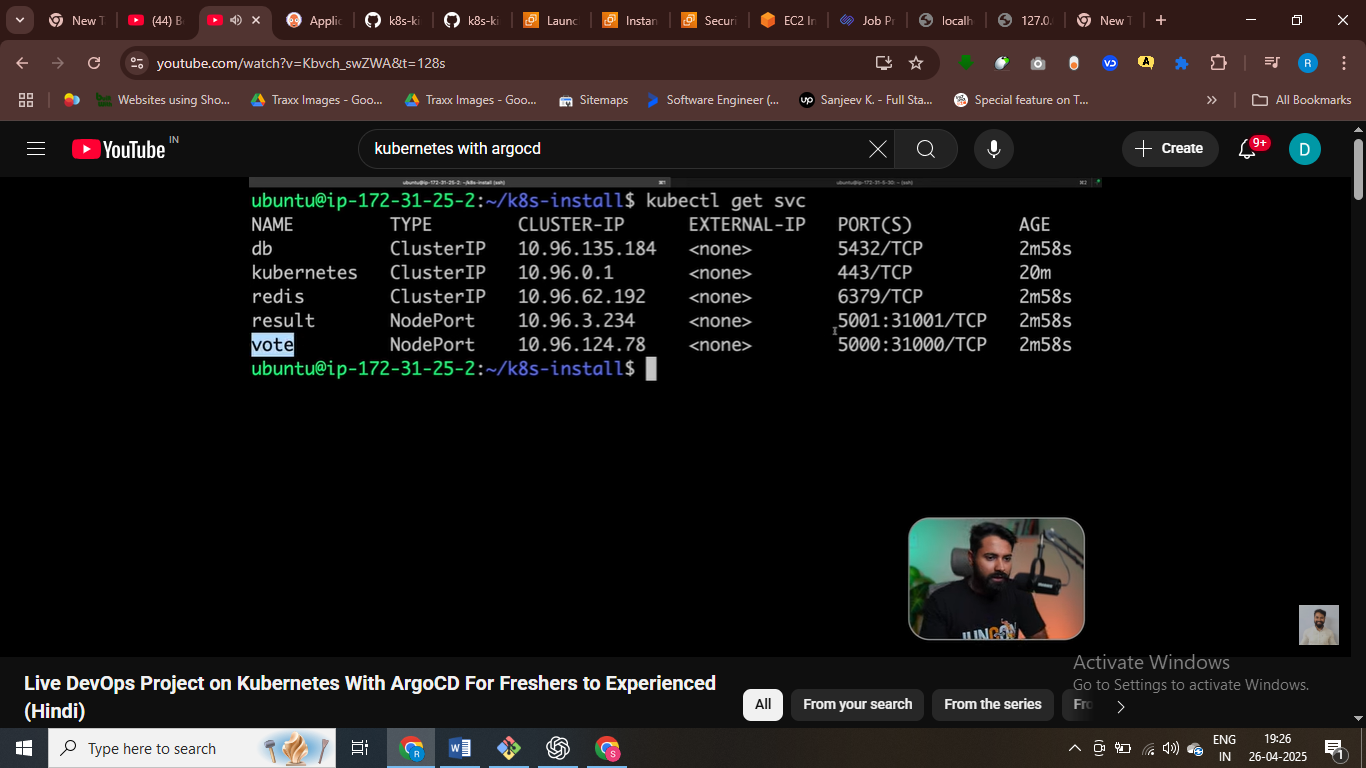
pod is creatin

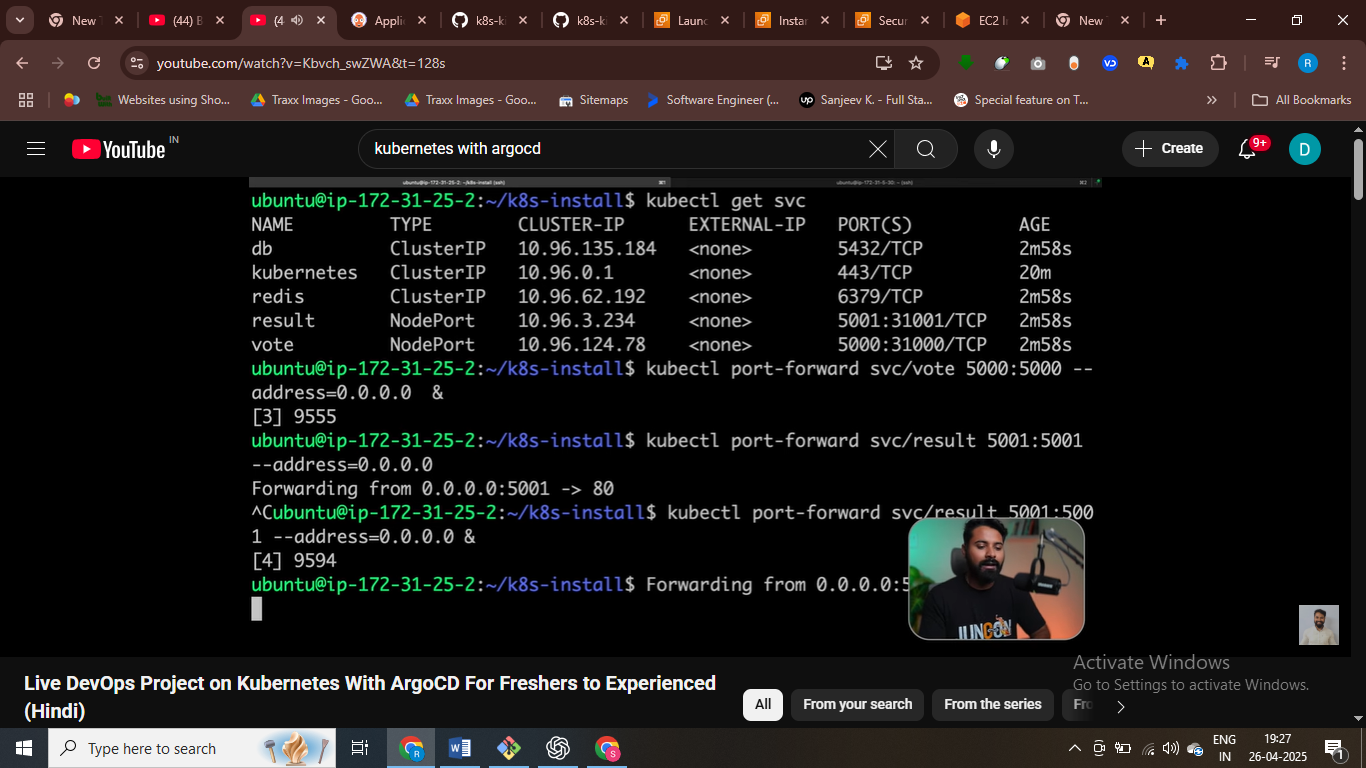


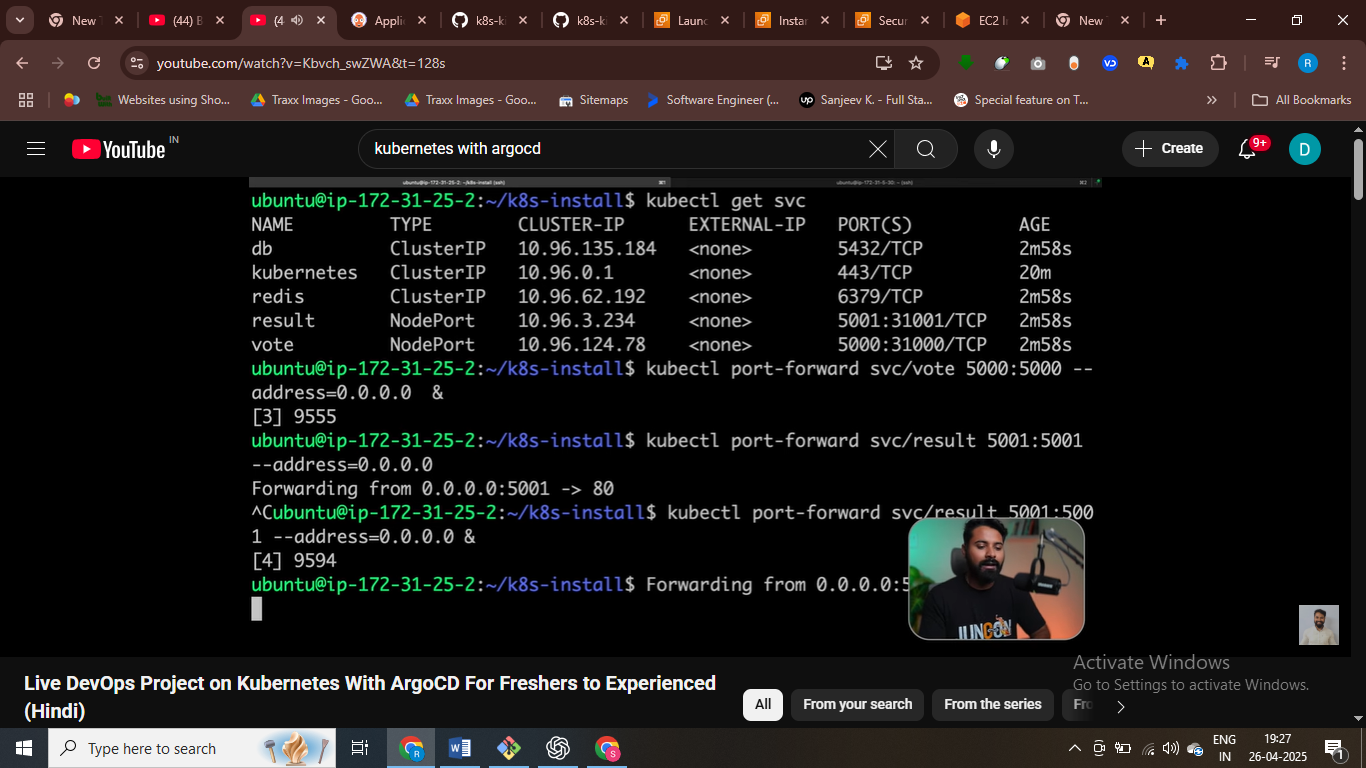
# after this pod is running , and deploy our appn to argocd



# after that scaling our deployments and replica sets







#access appn on browser

Cmds and all

ubuntu@ip-172-31-16-134:~$ sudo apt-get udpate

E: Invalid operation udpate

ubuntu@ip-172-31-16-134:~$ sudo apt-get update

Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease

Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]

Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]

Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]

Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]

Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]

Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]

Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]

Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]

Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]

Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]

Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]

Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1027 kB]

Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [223 kB]

Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [161 kB]

Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [13.5 kB]

Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1057 kB]

Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [267 kB]

Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [368 kB]

Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [26.0 kB]

Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [964 kB]

Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [198 kB]

Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]

Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [492 B]

Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [21.5 kB]

Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [4788 B]

Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]

Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [592 B]

Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Packages [39.1 kB]

Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main Translation-en [8676 B]

Get:31 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7096 B]

Get:32 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [272 B]

Get:33 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [27.1 kB]

Get:34 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [16.5 kB]

Get:35 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [16.4 kB]

Get:36 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1304 B]

Get:37 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]

Get:38 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]

Get:39 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]

Get:40 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]

Get:41 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [782 kB]

Get:42 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [147 kB]

Get:43 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [21.5 kB]

Get:44 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [7068 B]

Get:45 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [834 kB]

Get:46 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [181 kB]

Get:47 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.2 kB]

Get:48 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [17.0 kB]

Get:49 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [931 kB]

Get:50 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [190 kB]

Get:51 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]

Get:52 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 c-n-f Metadata [468 B]

Get:53 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [17.6 kB]

Get:54 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [3792 B]

Get:55 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]

Get:56 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [380 B]

Fetched 33.6 MB in 8s (4421 kB/s)

Reading package lists... Done

ubuntu@ip-172-31-16-134:~$ sudo apt-get install docker.io

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

The following additional packages will be installed:

bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan

Suggested packages:

ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-buildx docker-compose-v2 docker-doc rinse zfs-fuse | zfsutils

The following NEW packages will be installed:

bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan

0 upgraded, 8 newly installed, 0 to remove and 68 not upgraded.

Need to get 78.6 MB of archives.

After this operation, 302 MB of additional disk space will be used.

Do you want to continue? [Y/n] y

Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]

Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge-utils amd64 1.7.1-1ubuntu2 [33.9 kB]

Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]

Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 containerd amd64 1.7.24-0ubuntu1~24.04.2 [37.0 MB]

Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 dns-root-data all 2024071801~ubuntu0.24.04.1 [5918 B]

Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 dnsmasq-base amd64 2.90-2build2 [375 kB]

Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 docker.io amd64 26.1.3-0ubuntu1~24.04.1 [32.4 MB]

Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]

Fetched 78.6 MB in 1s (88.8 MB/s)

Preconfiguring packages ...

Selecting previously unselected package pigz.

(Reading database ... 70560 files and directories currently installed.)

Preparing to unpack .../0-pigz\_2.8-1\_amd64.deb ...

Unpacking pigz (2.8-1) ...

Selecting previously unselected package bridge-utils.

Preparing to unpack .../1-bridge-utils\_1.7.1-1ubuntu2\_amd64.deb ...

Unpacking bridge-utils (1.7.1-1ubuntu2) ...

Selecting previously unselected package runc.

Preparing to unpack .../2-runc\_1.1.12-0ubuntu3.1\_amd64.deb ...

Unpacking runc (1.1.12-0ubuntu3.1) ...

Selecting previously unselected package containerd.

Preparing to unpack .../3-containerd\_1.7.24-0ubuntu1~24.04.2\_amd64.deb ...

Unpacking containerd (1.7.24-0ubuntu1~24.04.2) ...

Selecting previously unselected package dns-root-data.

Preparing to unpack .../4-dns-root-data\_2024071801~ubuntu0.24.04.1\_all.deb ...

Unpacking dns-root-data (2024071801~ubuntu0.24.04.1) ...

Selecting previously unselected package dnsmasq-base.

Preparing to unpack .../5-dnsmasq-base\_2.90-2build2\_amd64.deb ...

Unpacking dnsmasq-base (2.90-2build2) ...

Selecting previously unselected package docker.io.

Preparing to unpack .../6-docker.io\_26.1.3-0ubuntu1~24.04.1\_amd64.deb ...

Unpacking docker.io (26.1.3-0ubuntu1~24.04.1) ...

Selecting previously unselected package ubuntu-fan.

Preparing to unpack .../7-ubuntu-fan\_0.12.16\_all.deb ...

Unpacking ubuntu-fan (0.12.16) ...

Setting up dnsmasq-base (2.90-2build2) ...

Setting up runc (1.1.12-0ubuntu3.1) ...

Setting up dns-root-data (2024071801~ubuntu0.24.04.1) ...

Setting up bridge-utils (1.7.1-1ubuntu2) ...

Setting up pigz (2.8-1) ...

Setting up containerd (1.7.24-0ubuntu1~24.04.2) ...

Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service.

Setting up ubuntu-fan (0.12.16) ...

Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /usr/lib/systemd/system/ubuntu-fan.service.

Setting up docker.io (26.1.3-0ubuntu1~24.04.1) ...

info: Selecting GID from range 100 to 999 ...

info: Adding group `docker' (GID 113) ...

Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.

Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.

Processing triggers for dbus (1.14.10-4ubuntu4.1) ...

Processing triggers for man-db (2.12.0-4build2) ...

Scanning processes...

Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.

ubuntu@ip-172-31-16-134:~$ sudo usermod -aG docker $USER && newgrp docker

ubuntu@ip-172-31-16-134:~$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

ubuntu@ip-172-31-16-134:~$ mkdir k8-install

ubuntu@ip-172-31-16-134:~$ cd k8-install

ubuntu@ip-172-31-16-134:~/k8-install$ vim install\_kind.sh

ubuntu@ip-172-31-16-134:~/k8-install$ chmod +x install\_kind.sh

ubuntu@ip-172-31-16-134:~/k8-install$ ./install\_kind.sh

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 97 100 97 0 0 2472 0 --:--:-- --:--:-- --:--:-- 2487

0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0

100 6304k 100 6304k 0 0 26.2M 0 --:--:-- --:--:-- --:--:-- 26.2M

ubuntu@ip-172-31-16-134:~/k8-install$ vim config.yaml

ubuntu@ip-172-31-16-134:~/k8-install$ kind create cluster --config=config.yml

ERROR: failed to create cluster: error reading file: open config.yml: no such file or directory

ubuntu@ip-172-31-16-134:~/k8-install$ vim config.yaml

ubuntu@ip-172-31-16-134:~/k8-install$ kind create cluster --config=config.yml --name=my-cluster

ERROR: failed to create cluster: error reading file: open config.yml: no such file or directory

ubuntu@ip-172-31-16-134:~/k8-install$ mv config.yaml config.yml

ubuntu@ip-172-31-16-134:~/k8-install$ kind create cluster --config=config.yml --name=my-cluster

Creating cluster "my-cluster" ...

✓ Ensuring node image (kindest/node:v1.30.0) 🖼

✓ Preparing nodes 📦 📦 📦

✓ Writing configuration 📜

✓ Starting control-plane 🕹️

✓ Installing CNI 🔌

✓ Installing StorageClass 💾

✓ Joining worker nodes 🚜

Set kubectl context to "kind-my-cluster"

You can now use your cluster with:

kubectl cluster-info --context kind-my-cluster

Have a question, bug, or feature request? Let us know! https://kind.sigs.k8s.io/#community 🙂

ubuntu@ip-172-31-16-134:~/k8-install$ vim install\_kubectl.sh

ubuntu@ip-172-31-16-134:~/k8-install$ chmod +x install\_kubectl.sh

ubuntu@ip-172-31-16-134:~/k8-install$ ./install\_kubectl.sh

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 138 100 138 0 0 2061 0 --:--:-- --:--:-- --:--:-- 2090

100 49.0M 100 49.0M 0 0 63.9M 0 --:--:-- --:--:-- --:--:-- 63.9M

Client Version: v1.30.0

Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3

kubectl installation complete.

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get nodes

NAME STATUS ROLES AGE VERSION

my-cluster-control-plane Ready control-plane 6m50s v1.30.0

my-cluster-worker Ready <none> 6m28s v1.30.0

my-cluster-worker2 Ready <none> 6m27s v1.30.0

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl create namespace argocd

namespace/argocd created

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml

customresourcedefinition.apiextensions.k8s.io/applications.argoproj.io created

customresourcedefinition.apiextensions.k8s.io/applicationsets.argoproj.io created

customresourcedefinition.apiextensions.k8s.io/appprojects.argoproj.io created

serviceaccount/argocd-application-controller created

serviceaccount/argocd-applicationset-controller created

serviceaccount/argocd-dex-server created

serviceaccount/argocd-notifications-controller created

serviceaccount/argocd-redis created

serviceaccount/argocd-repo-server created

serviceaccount/argocd-server created

role.rbac.authorization.k8s.io/argocd-application-controller created

role.rbac.authorization.k8s.io/argocd-applicationset-controller created

role.rbac.authorization.k8s.io/argocd-dex-server created

role.rbac.authorization.k8s.io/argocd-notifications-controller created

role.rbac.authorization.k8s.io/argocd-redis created

role.rbac.authorization.k8s.io/argocd-server created

clusterrole.rbac.authorization.k8s.io/argocd-application-controller created

clusterrole.rbac.authorization.k8s.io/argocd-applicationset-controller created

clusterrole.rbac.authorization.k8s.io/argocd-server created

rolebinding.rbac.authorization.k8s.io/argocd-application-controller created

rolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created

rolebinding.rbac.authorization.k8s.io/argocd-dex-server created

rolebinding.rbac.authorization.k8s.io/argocd-notifications-controller created

rolebinding.rbac.authorization.k8s.io/argocd-redis created

rolebinding.rbac.authorization.k8s.io/argocd-server created

clusterrolebinding.rbac.authorization.k8s.io/argocd-application-controller created

clusterrolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created

clusterrolebinding.rbac.authorization.k8s.io/argocd-server created

configmap/argocd-cm created

configmap/argocd-cmd-params-cm created

configmap/argocd-gpg-keys-cm created

configmap/argocd-notifications-cm created

configmap/argocd-rbac-cm created

configmap/argocd-ssh-known-hosts-cm created

configmap/argocd-tls-certs-cm created

secret/argocd-notifications-secret created

secret/argocd-secret created

service/argocd-applicationset-controller created

service/argocd-dex-server created

service/argocd-metrics created

service/argocd-notifications-controller-metrics created

service/argocd-redis created

service/argocd-repo-server created

service/argocd-server created

service/argocd-server-metrics created

deployment.apps/argocd-applicationset-controller created

deployment.apps/argocd-dex-server created

deployment.apps/argocd-notifications-controller created

deployment.apps/argocd-redis created

deployment.apps/argocd-repo-server created

deployment.apps/argocd-server created

statefulset.apps/argocd-application-controller created

networkpolicy.networking.k8s.io/argocd-application-controller-network-policy created

networkpolicy.networking.k8s.io/argocd-applicationset-controller-network-policy created

networkpolicy.networking.k8s.io/argocd-dex-server-network-policy created

networkpolicy.networking.k8s.io/argocd-notifications-controller-network-policy created

networkpolicy.networking.k8s.io/argocd-redis-network-policy created

networkpolicy.networking.k8s.io/argocd-repo-server-network-policy created

networkpolicy.networking.k8s.io/argocd-server-network-policy created

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get pods -n argocd

NAME READY STATUS RESTARTS AGE

argocd-application-controller-0 0/1 Running 0 25s

argocd-applicationset-controller-cc68b7b7b-l7fdt 1/1 Running 0 26s

argocd-dex-server-555b55c97d-wrx4m 0/1 PodInitializing 0 26s

argocd-notifications-controller-65655df9d5-8lr6d 1/1 Running 0 26s

argocd-redis-764b74c9b9-6vr8f 1/1 Running 0 26s

argocd-repo-server-7dcbcd967b-m2r7d 0/1 PodInitializing 0 26s

argocd-server-5b9cc8b776-nh6hl 0/1 CreateContainerConfigError 0 26s

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get svc -n argocd

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

argocd-applicationset-controller ClusterIP 10.96.70.84 <none> 7000/TCP,8080/TCP 119s

argocd-dex-server ClusterIP 10.96.32.31 <none> 5556/TCP,5557/TCP,5558/TCP 119s

argocd-metrics ClusterIP 10.96.137.254 <none> 8082/TCP 119s

argocd-notifications-controller-metrics ClusterIP 10.96.166.161 <none> 9001/TCP 119s

argocd-redis ClusterIP 10.96.57.18 <none> 6379/TCP 119s

argocd-repo-server ClusterIP 10.96.138.76 <none> 8081/TCP,8084/TCP 118s

argocd-server ClusterIP 10.96.210.124 <none> 80/TCP,443/TCP 118s

argocd-server-metrics ClusterIP 10.96.84.197 <none> 8083/TCP 118s

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl patch svc argocd-server -n argocd -p '{"spec": {"type": "NodePort"}}'

service/argocd-server patched

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl port-forward -n argocd service/argocd-server 8443:443 &

[1] 8201

ubuntu@ip-172-31-16-134:~/k8-install$ Forwarding from 127.0.0.1:8443 -> 8080

Forwarding from [::1]:8443 -> 8080

^C

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl port-forward -n argocd service/argocd-server 8443:443 &

[2] 8238

ubuntu@ip-172-31-16-134:~/k8-install$ Unable to listen on port 8443: Listeners failed to create with the following errors: [unable to create listener: Error listen tcp4 127.0.0.1:8443: bind: address already in use unable to create listener: Error listen tcp6 [::1]:8443: bind: address already in use]

error: unable to listen on any of the requested ports: [{8443 8080}]

^C

[2]+ Exit 1 kubectl port-forward -n argocd service/argocd-server 8443:443

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl port-forward -n argocd service/argocd-server 8443:443 &

[2] 8394

ubuntu@ip-172-31-16-134:~/k8-install$ Unable to listen on port 8443: Listeners failed to create with the following errors: [unable to create listener: Error listen tcp4 127.0.0.1:8443: bind: address already in use unable to create listener: Error listen tcp6 [::1]:8443: bind: address already in use]

error: unable to listen on any of the requested ports: [{8443 8080}]

^C

[2]+ Exit 1 kubectl port-forward -n argocd service/argocd-server 8443:443

ubuntu@ip-172-31-16-134:~/k8-install$ sudo lsof -i :8443

COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME

kubectl 8201 ubuntu 8u IPv4 184710 0t0 TCP localhost:8443 (LISTEN)

kubectl 8201 ubuntu 9u IPv6 185469 0t0 TCP ip6-localhost:8443 (LISTEN)

ubuntu@ip-172-31-16-134:~/k8-install$ kill -9 8201

ubuntu@ip-172-31-16-134:~/k8-install$ kill -9 8201

bash: kill: (8201) - No such process

[1]+ Killed kubectl port-forward -n argocd service/argocd-server 8443:443

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl port-forward -n argocd service/argocd-server 8443:443

Forwarding from 127.0.0.1:8443 -> 8080

Forwarding from [::1]:8443 -> 8080

^Cubuntu@ip-172-31-16-134:~/k8-installkubectl port-forward -n argocd service/argocd-server --address=0.0.0.0 8443:443 & &

[1] 8700

ubuntu@ip-172-31-16-134:~/k8-install$ Forwarding from 0.0.0.0:8443 -> 8080

kubectl port-forward -n argocd service/argocd-server --address=0.0.0.0 8443:443 &

[2] 8707

ubuntu@ip-172-31-16-134:~/k8-install$ Unable to listen on port 8443: Listeners failed to create with the following errors: [unable to create listener: Error listen tcp4 0.0.0.0:8443: bind: address already in use]

error: unable to listen on any of the requested ports: [{8443 8080}]

^C

[2]+ Exit 1 kubectl port-forward -n argocd service/argocd-server --address=0.0.0.0 8443:443

ubuntu@ip-172-31-16-134:~/k8-install$ sudo lsof -i :8443

COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME

kubectl 8700 ubuntu 8u IPv4 333495 0t0 TCP \*:8443 (LISTEN)

ubuntu@ip-172-31-16-134:~/k8-install$ kill -9 8700

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl port-forward -n argocd service/argocd-server 8443:443 --address=0.0.0.0 &

[2] 8842

[1] Killed kubectl port-forward -n argocd service/argocd-server --address=0.0.0.0 8443:443

ubuntu@ip-172-31-16-134:~/k8-install$ Forwarding from 0.0.0.0:8443 -> 8080

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

Handling connection for 8443

^C

ubuntu@ip-172-31-16-134:~/k8-install$ kubectl get secret -n argocd argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d && echo

z0cypMHFdMhbx6fl

ubuntu@ip-172-31-16-134:~/k8-install$ Read from remote host ec2-54-236-7-5.compute-1.amazonaws.com: Connection reset by peer

Connection to ec2-54-236-7-5.compute-1.amazonaws.com closed.

client\_loop: send disconnect: Connection reset by peer

user121@DESKTOP-JFJ21I5 MINGW64 ~/Downloads

$