Olimberio, David Christian C.

IV – ACSAD

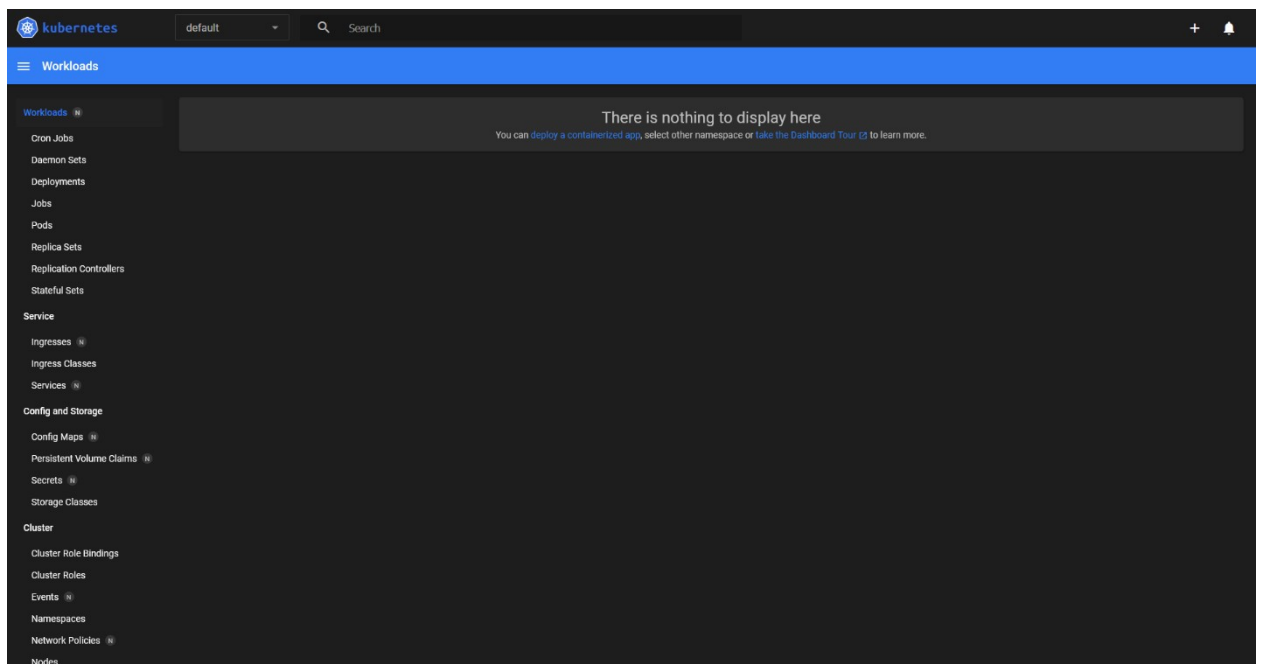# Assignment# 5 - Kubernetes Home Lab Activity

HELLO MINIKUBE

```
PS C:\WINDOWS\system32> minikube start
* minikube v1.37.0 on Microsoft Windows 11 Pro 10.0.26200.7171 Build 26200.7171
* Using the docker driver based on existing profile
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.48 ...
* Updating the running docker "minikube" container ...
! Failing to connect to https://registry.k8s.io/ from inside the minikube container
* To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking
/proxy/
* Preparing Kubernetes v1.34.0 on Docker 28.4.0 ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

- Minikube dashboard

```
PS C:\WINDOWS\system32> minikube dashboard
* Enabling dashboard ...
  - Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
  - Using image docker.io/kubernetesui/dashboard:v2.7.0
* Some dashboard features require the metrics-server addon. To enable all features please run:

        minikube addons enable metrics-server

* Verifying dashboard health ...
* Launching proxy ...
* Verifying proxy health ...
* Opening http://127.0.0.1:63700/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy/ in your default browser...
```

- Minikube status

```
PS C:\WINDOWS\system32> minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

Create a Deployment

```
No resources found in default namespace.
PS C:\Users\david> kubectl create deployment hello-node --image=registry.k8s.io/e2e-test-images/agnhost:2.53 -- /agnhost netexec --http-port=8080
deployment.apps/hello-node created
```

- View the deployments and view the pods

```
PS C:\Users\david> kubectl get deployments
NAME         READY   UP-TO-DATE   AVAILABLE   AGE
hello-node   1/1     1            1           21s
PS C:\Users\david> kubectl get pods
NAME                         READY   STATUS    RESTARTS   AGE
hello-node-6c9b5f4b59-c2bj5  1/1     Running   0          27s
```

- Get events

```
PS C:\Users\david> kubectl get events
LAST SEEN   TYPE     REASON                  OBJECT                              MESSAGE
2m15s       Normal   Scheduled               pod/hello-node-6c9b5f4b59-c2bj5     Successfully assigned default/hello-node-6c9b5f4b59-c2bj5 to minikube
2m15s       Normal   Pulling                 pod/hello-node-6c9b5f4b59-c2bj5     Pulling image "registry.k8s.io/e2e-test-images/agnhost:2.53"
115s        Normal   Pulled                  pod/hello-node-6c9b5f4b59-c2bj5     Successfully pulled image "registry.k8s.io/e2e-test-images/agnhost:2.53" in 19.677s (19.677s including wai
ting). Image size: 139374622 bytes.
115s        Normal   Created                 pod/hello-node-6c9b5f4b59-c2bj5     Created container: agnhost
115s        Normal   Started                 pod/hello-node-6c9b5f4b59-c2bj5     Started container agnhost
2m15s       Normal   SuccessfulCreate        replicaset/hello-node-6c9b5f4b59    Created pod: hello-node-6c9b5f4b59-c2bj5
2m15s       Normal   ScalingReplicaSet       deployment/hello-node               Scaled up replica set hello-node-6c9b5f4b59 from 0 to 1
2m56s       Normal   Starting                node/minikube                       Starting kubelet.
2m56s       Normal   NodeHasSufficientMemory node/minikube                       Node minikube status is now: NodeHasSufficientMemory
2m56s       Normal   NodeHasNoDiskPressure   node/minikube                       Node minikube status is now: NodeHasNoDiskPressure
2m56s       Normal   NodeHasSufficientPID    node/minikube                       Node minikube status is now: NodeHasSufficientPID
2m56s       Normal   NodeAllocatableEnforced node/minikube                       Updated Node Allocatable limit across pods
2m50s       Normal   Starting                node/minikube                       Starting kubelet.
2m50s       Normal   NodeAllocatableEnforced node/minikube                       Updated Node Allocatable limit across pods
2m50s       Normal   NodeHasSufficientMemory node/minikube                       Node minikube status is now: NodeHasSufficientMemory
2m50s       Normal   NodeHasNoDiskPressure   node/minikube                       Node minikube status is now: NodeHasNoDiskPressure
2m50s       Normal   NodeHasSufficientPID    node/minikube                       Node minikube status is now: NodeHasSufficientPID
2m45s       Normal   RegisteredNode          node/minikube                       Node minikube event: Registered Node minikube in Controller
2m42s       Normal   Starting                node/minikube
PS C:\Users\david>
```

- View the configuration

```
PS C:\Users\david> kubectl config view
apiVersion: v1
clusters:
- cluster:
    certificate-authority: C:\Users\david\.minikube\ca.crt
    extensions:
    - extension:
        last-update: Mon, 17 Nov 2025 14:31:06 +08
        provider: minikube.sigs.k8s.io
        version: v1.37.0
      name: cluster_info
    server: https://127.0.0.1:54931
  name: minikube
contexts:
- context:
    cluster: minikube
    extensions:
    - extension:
        last-update: Mon, 17 Nov 2025 14:31:06 +08
        provider: minikube.sigs.k8s.io
        version: v1.37.0
      name: context_info
    namespace: default
    user: minikube
  name: minikube
current-context: minikube
kind: Config
users:
- name: minikube
  user:
    client-certificate: C:\Users\david\.minikube\profiles\minikube\client.crt
    client-key: C:\Users\david\.minikube\profiles\minikube\client.key
PS C:\Users\david>
```

- **View application logs for a container in a pod**

```
PS C:\Users\david> kubectl logs hello-node-6c9b5f4b59-c2bj5
I1117 06:32:00.861285       1 log.go:245] Started HTTP server on port 8080
I1117 06:32:00.862000       1 log.go:245] Started UDP server on port  8081
PS C:\Users\david>
```

Create a Service

- Expose the Pod to the public internet

```
PS C:\Users\david> kubectl expose deployment hello-node --type=LoadBalancer --port=8080
service/hello-node exposed
```

- Get services

```
PS C:\Users\david> kubectl get services
NAME          TYPE           CLUSTER-IP       EXTERNAL-IP    PORT(S)          AGE
hello-node    LoadBalancer   10.101.151.241   <pending>      8080:32712/TCP   63s
kubernetes    ClusterIP      10.96.0.1        <none>         443/TCP          7m4s
PS C:\Users\david>
```

- Open the service

```
PS C:\Users\david> minikube service hello-node
```

NAMESPACE	NAME	TARGET PORT	URL
default	hello-node	8080	http://192.168.49.2:32712

```
* Starting tunnel for service hello-node./
```

NAMESPACE	NAME	TARGET PORT	URL
default	hello-node		http://127.0.0.1:57407

```
* Starting tunnel for service hello-node.
* Opening service default/hello-node in default browser...
! Because you are using a Docker driver on windows, the terminal needs to be open to run it.
```

```
NOW: 2025-11-17 06:38:39.158166497 +0000 UTC m=+398.346827119
```

ENABLE ADDONS

- View the adds on list

```
* Stopping tunnel for service netto node.
PS C:\Users\david> minikube addons list
```

ADDON NAME	PROFILE	STATUS	MAINTAINER
ambassador	minikube	disabled	3rd party (Ambassador)
amd-gpu-device-plugin	minikube	disabled	3rd party (AMD)
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)
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
kubetail	minikube	disabled	3rd party (kubetail.com)
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)
volcano	minikube	disabled	third-party (volcano)
volumesnapshots	minikube	disabled	Kubernetes
yakd	minikube	disabled	3rd party (marcnuri.com)

```
PS C:\Users\david> |
```

- Enabling metric server

```
PS C:\Users\david> minikube addons enable metrics-server
* metrics-server 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/metrics-server/metrics-server:v0.8.0
* The 'metrics-server' addon is enabled
```

- View the Pod and Service you created by installing that addon

```
PS C:\Users\david> kubectl get pod,svc -n kube-system
NAME                                      READY   STATUS    RESTARTS      AGE
pod/coredns-66bc5c9577-dmtc9              1/1     Running   0             10m
pod/etcd-minikube                        1/1     Running   0             10m
pod/kube-apiserver-minikube              1/1     Running   0             10m
pod/kube-controller-manager-minikube     1/1     Running   0             10m
pod/kube-proxy-dm9rw                     1/1     Running   0             10m
pod/kube-scheduler-minikube              1/1     Running   0             10m
pod/metrics-server-85b7d694d7-xbpzs      0/1     Running   0             55s
pod/storage-provisioner                  1/1     Running   1 (10m ago)   10m

NAME                     TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)                  AGE
service/kube-dns         ClusterIP   10.96.0.10      <none>        53/UDP,53/TCP,9153/TCP   10m
service/metrics-server   ClusterIP   10.99.140.182   <none>        443/TCP                  55s
```

- Check the output from metrics-server:

```
PS C:\Users\david> kubectl top pods
NAME                           CPU(cores)   MEMORY(bytes)
hello-node-6c9b5f4b59-c2bj5    1m           6Mi
```

- Disable metrics-server:

```
PS C:\Users\david> minikube addons disable metrics-server
* "The 'metrics-server' addon is disabled
```

Clean Up

```
PS C:\Users\david> kubectl delete service hello-node
service "hello-node" deleted from default namespace
PS C:\Users\david> kubectl delete deployment hello-node
deployment.apps "hello-node" deleted from default namespace
PS C:\Users\david> minikube stop
* Stopping node "minikube"  ...
* Powering off "minikube" via SSH ...
* 1 node stopped.
PS C:\Users\david> minikube delete
* Deleting "minikube" in docker ...
* Deleting container "minikube" ...
* Removing C:\Users\david\.minikube\machines\minikube ...
* Removed all traces of the "minikube" cluster.
PS C:\Users\david>
```

GET A SHELL TO A RUNNING CONTAINER

- Creating pod

```
PS C:\Users\david> kubectl apply -f https://k8s.io/examples/application/shell-demo.yaml
pod/shell-demo created
PS C:\Users\david>
```

- Verifying that the container is running

```
PS C:\Users\david> kubectl get pod shell-demo
NAME          READY   STATUS    RESTARTS   AGE
shell-demo    1/1     Running   0          43s
```

- Get a shell to the running container:

```
PS C:\Users\david> kubectl exec --stdin --tty shell-demo -- /bin/bash
root@minikube:/#
```

```
root@minikube:/# ls /
bin  boot  dev  docker-entrypoint.d  docker-entrypoint.sh  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@minikube:/#
```

Writing the root page for nginx

- In your shell, create an index.html file in the /usr/share/nginx/html directory:

```
PS C:\Users\david> kubectl exec --stdin --tty shell-demo -- /bin/bash
root@minikube:/# echo 'Hello shell demo' > /usr/share/nginx/html/index.html
root@minikube:/# apt-get update
Get:1 http://deb.debian.org/debian trixie InRelease [140 kB]
Get:2 http://deb.debian.org/debian trixie-updates InRelease [47.3 kB]
Get:3 http://deb.debian.org/debian-security trixie-security InRelease [43.4 kB]
Get:4 http://deb.debian.org/debian trixie/main amd64 Packages [9670 kB]
Get:5 http://deb.debian.org/debian trixie-updates/main amd64 Packages [5412 B]
Get:6 http://deb.debian.org/debian-security trixie-security/main amd64 Packages [71.8 kB]
Fetched 9978 kB in 2s (5678 kB/s)
Reading package lists... Done
root@minikube:/# apt-get install curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcurl4t64
The following packages will be upgraded:
  curl libcurl4t64
2 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
Need to get 661 kB of archives.
After this operation, 0 B of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian trixie/main amd64 curl amd64 8.14.1-2+deb13u2 [270 kB]
Get:2 http://deb.debian.org/debian trixie/main amd64 libcurl4t64 amd64 8.14.1-2+deb13u2 [391 kB]
Fetched 661 kB in 0s (1900 kB/s)
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 79, <STDIN> line 2.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine module) (@INC entries checked: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.40.1 /usr/local/share/perl/5.40.1 /usr/lib/x86_64-linux-gnu/perl
5.40 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.40 /usr/share/perl/5.40 /usr/local/lib/site_perl) at /usr/share/perl5/Debconf/FrontEnd/Readline.pm line 8, <STDIN> line 2.)
debconf: falling back to frontend: Teletype
(Reading database ... 6699 files and directories currently installed.)
Preparing to unpack .../curl_8.14.1-2+deb13u2_amd64.deb ...
Unpacking curl (8.14.1-2+deb13u2) over (8.14.1-2) ...
Preparing to unpack .../libcurl4t64_8.14.1-2+deb13u2_amd64.deb ...
Unpacking libcurl4t64:amd64 (8.14.1-2+deb13u2) over (8.14.1-2) ...
Setting up libcurl4t64:amd64 (8.14.1-2+deb13u2) ...
Setting up curl (8.14.1-2+deb13u2) ...
Processing triggers for libc-bin (2.41-12) ...
root@minikube:/#
```

- The output shows the text that you wrote to the index.html file:

```
root@minikube:/# curl http://localhost/
Hello shell demo
```

- Exiting

```
Hello shell demo
root@minikube:/# exit
exit
```

Running individual command in a container

```
exit
PS C:\Users\david> kubectl exec shell-demo -- env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=minikube
KUBERNETES_PORT_443_TCP_ADDR=10.96.0.1
KUBERNETES_SERVICE_HOST=10.96.0.1
KUBERNETES_SERVICE_PORT=443
KUBERNETES_SERVICE_PORT_HTTPS=443
KUBERNETES_PORT=tcp://10.96.0.1:443
KUBERNETES_PORT_443_TCP=tcp://10.96.0.1:443
KUBERNETES_PORT_443_TCP_PROTO=tcp
KUBERNETES_PORT_443_TCP_PORT=443
NGINX_VERSION=1.29.3
NJS_VERSION=0.9.4
NJS_RELEASE=1~trixie
PKG_RELEASE=1~trixie
DYNPKG_RELEASE=1~trixie
HOME=/root
```

- Experimenting commands

```
command terminated with exit code 127
PS C:\Users\david> kubectl exec shell-demo -- ls /
bin
boot
dev
docker-entrypoint.d
docker-entrypoint.sh
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
```

Opening a shell when a Pod has more than one container

- If a Pod has more than one container, use --container or -c to specify a container in the kubectl exec command. For example, suppose you have a Pod named my-pod, and the Pod has two containers named *main-app* and *helper-app*. The following command would open a shell to the *main-app* container.

```
PS C:\Users\david> kubectl exec -i -t my-pod --container main-app -- /bin/bash
Error from server (NotFound): pods "my-pod" not found
PS C:\Users\david> kubectl exec -i -t my-pod --container main-app -- /bin/bash
Error from server (NotFound): pods "my-pod" not found
```