

# Lab Exercise 12 - Start and Access

## Kubernetes Dashboard

### Objective

To enable Kubernetes in Docker Desktop, deploy the Kubernetes Dashboard, and access it securely using a web browser on Windows.

### Prerequisites

- Windows 10 / 11
- Docker Desktop installed
- Docker Desktop Kubernetes enabled
- Internet connection
- kubectl (comes bundled with Docker Desktop)

### Step 1: Enable Kubernetes in Docker Desktop

1. Open **Docker Desktop**
2. Go to **Settings**
3. Select **Kubernetes**
4. Check **Enable Kubernetes**
5. Click **Apply & Restart**

Wait until Kubernetes status shows **Running** (green).

The screenshot shows the Docker Desktop interface with the "Containers" tab selected. The sidebar on the left includes links for Ask Gordon (Beta), Containers (selected), Images, Volumes, Builds, Models (Beta), MCP Toolkit (Beta), Docker Hub, Docker Scout, and Extensions. The main area displays a table of running containers:

	Name	Container ID	Image	Port(s)	Last Update	Created
<input type="checkbox"/>	epic_hoover	d71aa15596a9	combined-example		N/A	14 days ago
<input type="checkbox"/>	objective_cerf	bb9f785d7054	combined-example		N/A	14 days ago
<input type="checkbox"/>	blissful_chatterjee	26705d4f1cde	entrypoint-example		N/A	14 days ago
<input type="checkbox"/>	infallible_jumiere	612bc5166a4a	entrypoint-example		N/A	14 days ago
<input type="checkbox"/>	romantic_heisenberg	ac7fe851bb82	entrypoint-example		N/A	14 days ago
<input type="checkbox"/>	nice_ride	1398a5418814	cmd-example		N/A	14 days ago
<input type="checkbox"/>	serene_shamir	4a4f7eb0ad30	cmd-example		N/A	14 days ago
<input type="checkbox"/>	busy_chebyshev	0c03404f1e0a	cmd-example		N/A	14 days ago
<input type="checkbox"/>	affectionate_bardeen	4790d53ba747	nginxlatest		N/A	3 months ago
<input type="checkbox"/>	loving_baner	46006d4200	nginx.html.nginxlatest		N/A	2 months ago

At the bottom, status indicators show "Engine running", "Kubernetes starting", and "New version available". A context menu is open over the "loving\_baner" container, showing options for "Edit", "Logs", "Delete", and "Reset cluster".

The screenshot shows the Docker Desktop interface with the "Settings" tab selected. The sidebar includes links for General, Resources, Docker Engine, Builders, **Kubernetes** (selected), Software updates, Extensions, Beta features, and Notifications. The main area displays the Kubernetes configuration:

**Kubernetes**

Enable Kubernetes  
Start a Kubernetes single or multi-node cluster when starting Docker Desktop.

**Cluster**

**docker-desktop**  
kubeadm, 1 node, v1.32.2

**Cluster settings**

Choose cluster provisioning method

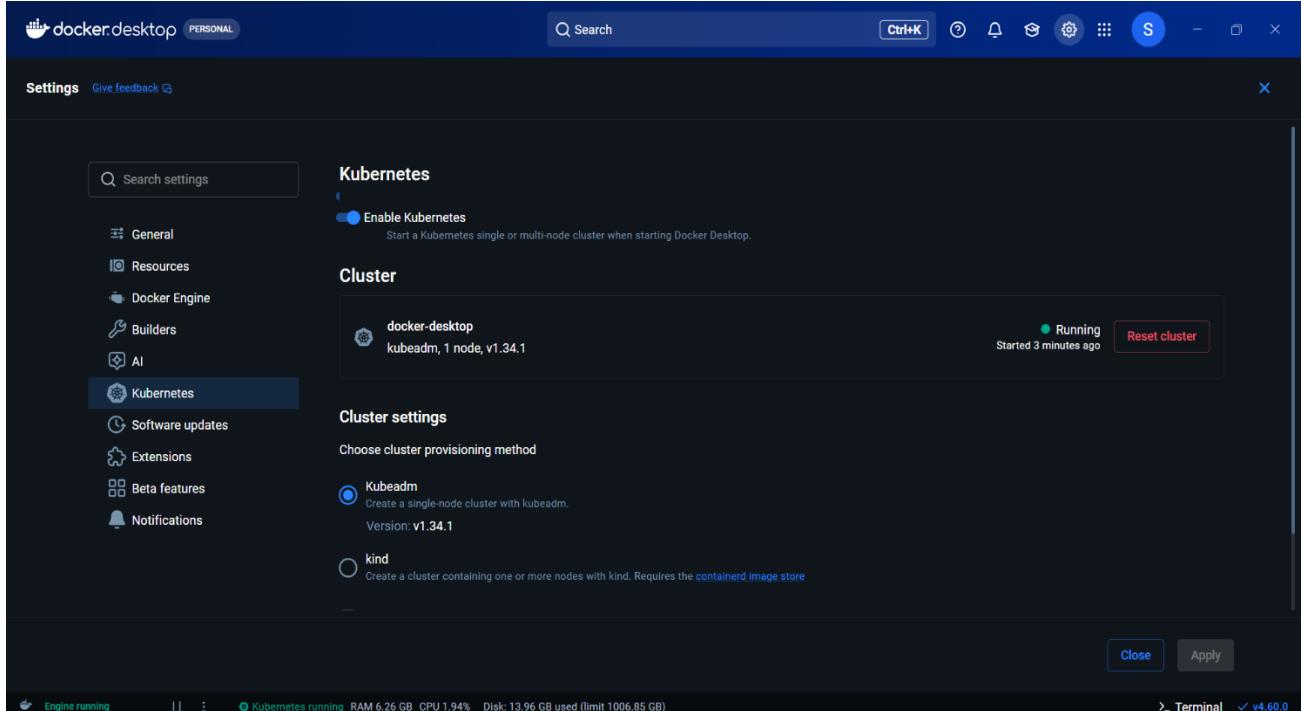
Kubeadm  
Create a single-node cluster with kubeadm.  
Version: v1.32.2

kind  
Create a cluster containing one or more nodes with kind. Requires the containerd image store

Show custom containers (advanced)

Buttons at the bottom right: Close, Apply, and a red-bordered "Reset cluster" button.

At the bottom, status indicators show "Engine running", "Kubernetes starting", and "New version available".



## Step 2: Verify Kubernetes Cluster

Open **PowerShell** or **Command Prompt** and run:

- `kubectl version --client`
- Check cluster status:
- `kubectl cluster-info`

Check nodes:

```
kubectl get nodes
```

Expected output: Node status should be **Ready**

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\HP> docker --version
Docker version 29.2.0, build 0b9d198
PS C:\Users\HP> kubectl version --client
Client Version: v1.34.1
Kustomize Version: v5.7.1
```

```
PS C:\Users\HP> kubectl cluster-info
Kubernetes control plane is running at https://kubernetes.docker.internal:6443
CoreDNS is running at https://kubernetes.docker.internal:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
PS C:\Users\HP> kubectl get nodes
NAME           STATUS    ROLES     AGE      VERSION
docker-desktop   Ready    control-plane   14m    v1.32.2
PS C:\Users\HP> |
```

### Step 3: Deploy Kubernetes Dashboard

Apply the official Kubernetes Dashboard manifest:

```
kubectl apply -f
```

```
https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml
```

Verify namespace creation:

```
kubectl get ns
```

You should see:

```
kubernetes-dashboard
```

```
PS C:\Users\HP> kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.7.0/aio/deploy/recommended.yaml
namespace/kubernetes-dashboard created
serviceaccount/kubernetes-dashboard created
service/kubernetes-dashboard created
secret/kubernetes-dashboard-certs created
secret/kubernetes-dashboard-csrf created
secret/kubernetes-dashboard-key-holder created
configmap/kubernetes-dashboard-settings created
role.rbac.authorization.k8s.io/kubernetes-dashboard created
clusterrole.rbac.authorization.k8s.io/kubernetes-dashboard created
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created
clusterrolebinding.rbac.authorization.k8s.io/kubernetes-dashboard created
deployment.apps/kubernetes-dashboard created
service/dashboard-metrics-scraper created
deployment.apps/dashboard-metrics-scraper created
PS C:\Users\HP> kubectl get ns
NAME          STATUS  AGE
default        Active  15m
kube-node-lease  Active  15m
kube-public    Active  15m
kube-system    Active  15m
kubernetes-dashboard  Active  9s
```

## **Step 4: Verify Dashboard Pods**

Check dashboard pods:

```
kubectl get pods -n kubernetes-dashboard
```

Expected status: Running

```
PS C:\Users\HP> kubectl get pods -n kubernetes-dashboard
NAME                               READY   STATUS    RESTARTS   AGE
dashboard-metrics-scraper-5bd45c9dd6-gn6cg   1/1     Running   0          92s
kubernetes-dashboard-79cbcf9fb6-pv62z        1/1     Running   0          92s
PS C:\Users\HP> |
```

## **Step 5: Create Admin User for Dashboard Access**

Create a service account:

```
kubectl create serviceaccount dashboard-admin -n kubernetes-dashboard
```

Create cluster role binding:

```
kubectl create clusterrolebinding dashboard-admin-binding --clusterrole=cluster-admin   --
serviceaccount=kubernetes-dashboard:dashboard-admin
```

```
PS C:\Users\HP> kubectl create serviceaccount dashboard-admin -n kubernetes-dashboard
serviceaccount/dashboard-admin created
PS C:\Users\HP> kubectl create clusterrolebinding dashboard-admin-binding --clusterrole=cluster-admin   --
serviceaccount=kubernetes-dashboard:dashboard-admin
clusterrolebinding.rbac.authorization.k8s.io/dashboard-admin-binding created
PS C:\Users\HP> |
```

## **Step 6: Generate Dashboard Login Token**

Run the following command to get the token:

```
kubectl -n kubernetes-dashboard create token dashboard-admin
```

Copy the generated token (you will paste it in the browser later).

```
PS C:\Users\HP> kubectl -n kubernetes-dashboard create token dashboard-admin
eyJhbGciOiJSUzI1NiIsImtpZCI6IlJwYzJMSjBxT3lW0lxat4b1BNZ0ZFeGLYjNZNldNdFptemU2SG9oMUUifQ.eyJhdWQiOlsiaHR0cHM6Ly9rdWJlc5ldGVzLmRlZmF1bhQuc3ZjLmNsdxN0ZXIubG9jYWwiXSwizXhwIjoxNzcwNzQ3MjAyLCJpYXQiOjE3NzA3NDM2MDIsImLzcyI6Imh0dHbz018va3ViZXJuZXRLcy5kZWzhWx0LnN2Y5jbHVzdGVyLmxvY2FsliwianRpIjoindU2TMwZDAtMTFmNy00ZDgyLTlh0WytNjRi0TA3YjBlyMz2Iiwiia3ViZXJuZXRLcy5pbvI6eyJuyWlc3BhY2Ui0iJrdWJlc5ldGVzLWRhc2hib2FyZCIsInNlcnPzY2VhY2NvdW50Ijp7Im5hbWUi0i0jYXN0Ym9hcmQtYWRtaW4iLCJ1aWQioiit4MDllYjN1Mi05Njg2LTQ3Y2ItOTE3MC02NjI1NjIxOGNjZDYifxF0sIn5izi6Mt3MDc0MzYwMiic3ViIjoic3LzdGVtOnNlcnPzY2VhY2NvdW500mt1YmVybV0ZMtzGFzaGJvYXJkOmRhc2hib2FyZC1hZG1pbjJ9.Gw04YPnu0yUZHmBr-d_yNv7CbslycugZSl0oih8wQ3BA2oFBqc-4BvrGeqBcQsMMKCjIHpQuFv-Gs6nVbX0wfLOvt51KZX_OFjx2Xn1_Dg-TRpE4jQyHx3_iTGUZM14iqs50p972p3DXPy8AsFhf_05RnWt1VJ4uX0uGD2CY2tw_kApe8n5lcc978W71651bXByram-SMKBFVJtqe7GJ0q1eAnqrqoyAx4L2E1USJ2Y0XAmFh3Kzr3uaUpxSMskprar6qkJh1drm_BC_cQG3EFLb0kxngdgXvMf6lw9XYGa19Wop5rZVjRrkItqChekjUqwUeYH8n_sj30Up18BQ
PS C:\Users\HP> |
```

## Step 7: Start Kubernetes Dashboard

Run the proxy command:

```
kubectl proxy
```

Keep this terminal **running**.

```
PS C:\Users\HP> kubectl proxy
Starting to serve on 127.0.0.1:8001
|
```

## Step 8: Access Kubernetes Dashboard in Browser

Open a web browser and paste the following URL:

```
http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-
dashboard:/proxy/
```

## Step 9: Login to Dashboard

1. Select **Token** authentication
2. Paste the token generated earlier
3. Click **Sign In**

You should now see the **Kubernetes Dashboard UI**.

## Step 10: Explore Dashboard

You can now view:

- Nodes
- Pods
- Deployments
- Services
- Namespaces
- ConfigMaps and Secrets

