

[Menu](#)[Browse Library](#) ▼

Training Library / Deploy a Stateful Application in a Kubernetes Cluster

Monitoring Your Kubernetes Cluster Using Kubernetes Dashboard

1h 27m 45s left

Open Cloud



100%

Environment

Setup completed

Average setup time: 2m

35s



100%

Open Code

Environment

Setup completed

Credentials

Account ID ⓘ

062069998 [Copy](#)

Username ⓘ

student [Copy](#)

Password ⓘ

Ca1_fl11s [Copy](#)

Region ⓘ

US West 2 [Copy](#)

PEM ⓘ

PPK ⓘ

[Download](#) [Download](#)

Bastion Public Ip ⓘ

35.155.21 [Copy](#)

Cluster SSH ⓘ

Introduction

Earlier you saw that the Kubernetes Dashboard pod is running in the kube-system namespace. The Kubernetes Dashboard provides a website for monitoring and managing your Kubernetes cluster. In this lab step, you will familiarize yourself with the Kubernetes Dashboard and also demonstrate how to use a port forward to connect to the control plane node. To gain full access to the dashboard you will first bind the cluster-admin role to the dashboard's service account.

Instructions

1. In your SSH shell, enter the following to create a cluster role binding for the kubernetes-dashboard service account:

[Copy code](#)

```
1 cat << EOF > dashboard-admin.yaml
2 apiVersion: v1
3 kind: ServiceAccount
4 metadata:
5   name: dashboard-admin
6   namespace: kubernetes-dashboard
7 ---
8 apiVersion: rbac.authorization.k8s.io/v1
9 kind: ClusterRoleBinding
10 metadata:
11   name: dashboard-admin
12 roleRef:
13   apiGroup: rbac.authorization.k8s.io
14   kind: ClusterRole
15   name: cluster-admin
16 subjects:
17 - kind: ServiceAccount
18   name: dashboard-admin
19   namespace: kubernetes-dashboard
20 EOF
```

The roleRef reference the cluster-admin role and the subjects list the kubernetes-dashboard ServiceAccount.

Support

[Skip to content](#)

Press [option](#) + [Q](#) to open this menu



Bridge
Connection
Completed

Lab Steps

1 Connecting to the
Kubernetes Cluster

2 Inspecting the
Kubernetes Cluster

3 Deploying a
Stateful Application
in the Kubernetes
Cluster

4 Working with the
Stateful Application

5 **Monitoring Your
Kubernetes Cluster
Using Kubernetes
Dashboard**

🔗 Need help? Contact our
support team

```
1 | kubectl create -f dashboard-admin.yaml
```

3. Enter the following to create a token for the kubernetes dashboard service account which you will need to sign in to the dashboard later:

Copy code

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

4. Run a Kubernetes proxy server on the bastion in order to proxy requests to the control plane node from outside the private subnet:

Copy code

```
1 | kubectl port-forward -n kubernetes-dashboard --address 0.0.0.0 s
```

Forwarding from 0.0.0.0:8001 -> 8443

The `--address` option allows connections from anywhere. This is only acceptable for demonstration purposes.

5. Open a new browser tab, enter the following url to open the Kubernetes Dashboard for the cluster:

Copy code

```
https://35.155.211.176:8001
```



Menu



Browse Library ▾



Your connection is not private

Attackers might be trying to steal your information from **54.212.99.198** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_INVALID



To get Chrome's highest level of security, [turn on enhanced protection](#)

Advanced

Reload

Your browser will likely flag the TLS certificate as being invalid. This is expected since the cluster used a self-signed certificate.

6. Allow your browser to view the website. in recent versions of Chrome, you can do so by typing *thisisunsafe*

Kubernetes Dashboard

☒ Token

Every Service Account has a Secret with valid Bearer Token that can be used to log in to Dashboard. To find out more about how to configure and use Bearer Tokens, please refer to the [Authentication](#) section.

☐ Kubeconfig

Please select the kubeconfig file that you have created to configure access to the cluster. To find out more about how to configure and use kubeconfig file, please refer to the [Configure Access to Multiple Clusters](#) section.

Enter token *

Sign in

The dashboard asks you for a method to authenticate, you will use the token you output before a token to sign in.

7. Copy the token you output earlier at the step number 3 (in the web terminal, you only need to highlight the token and it is automatically copied to click **Sign in**.

[Skip to content](#)

Press  +  to open this menu



Menu



Browse Library ▾



Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses

Services

Config and Storage

Config Maps

Persistent Volume Claims

Secrets

Storage Classes

Cluster

Cluster Role Bindings

Pods

Stateful Sets

Pods

Name	Namespace	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
mysql-4	default	app: mysql controller-revision-hash: mysql-8448f759 statefulset.kubernetes.io/pod-name: mysql-4	ip-10-0-0-11.us-west-2.compute.int	Running	0		192.64Mi	28 minutes ago
mysql-3	default	app: mysql controller-revision-hash: mysql-8448f759 statefulset.kubernetes.io/pod-name: mysql-3	ip-10-0-0-11.us-west-2.compute.int	Running	0		182.30Mi	29 minutes ago

From the dashboard, you can get the same information found using `kubectl` commands but with an easy-to-navigate interface.

8. Navigate to Workloads > Stateful Sets > mysql:

Workloads

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses

Services

Config and Storage

Config Maps

Persistent Volume Claims

Secrets

Storage Classes

Cluster

Images

mysql:5.7 gcr.io/google-samples/xtrabackup:1.0

Init images

mysql:5.7 gcr.io/google-samples/xtrabackup:1.0

Pods status

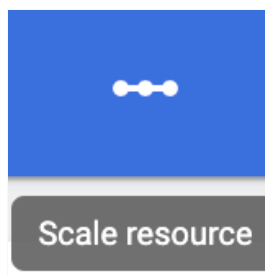
Running 5 Desired 5

Pods

Name	Namespace	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Created ↑
mysql-4	default	app: mysql controller-revision-hash: mysql-77775f95c4 statefulset.kubernetes.io/pod-name: mysql-4	ip-10-0-11-19f-west-2.compute.int	Running	0		191.91Mi	8 minutes ago
mysql-3	default	app: mysql controller-revision-hash: mysql-77775f95c4 statefulset.kubernetes.io/pod-name: mysql-3	ip-10-0-11-19f-west-2.compute.int	Running	0		182.71Mi	8 minutes ago

The **CPU** and **Memory** column graphs are visible because [metrics server](#) is running in your Kubernetes cluster.

9. Click on the Scale button near the top-right corner of the mysql Stateful Sets view:



Skip to content

Press **option** + to open this menu

11. Scroll down to the **Pods** section and observe the `mysql-4` pod is removed first, followed by the `mysql-3` pod.

Pods are removed in the reverse order that they are created in StatefulSets.

12. Navigate to **Cluster > Persistent Volumes**:

Name	Capacity	Access Modes	Reclaim Policy	Status	Claim	Storage Class	Reason	Created ↑
pvc-8a54c3f1-e3e5-4645-9412-32d6660e4db0	storage: 2Gi	ReadWriteOnce	Delete	Bound	default/data-mysql-4	general	-	21 minutes ago
pvc-e28b52c4-96e9-4ad6-ae82-376c91db0220	storage: 2Gi	ReadWriteOnce	Delete	Bound	default/data-mysql-3	general	-	22 minutes ago
pvc-a7fd27a8-10dd-496f-8d47-ba4b47696699	storage: 2Gi	ReadWriteOnce	Delete	Bound	default/data-mysql-2	general	-	36 minutes ago
pvc-9de1fc48-bd37-49f6-9541-1dfe87d447af	storage: 2Gi	ReadWriteOnce	Delete	Bound	default/data-mysql-1	general	-	37 minutes ago
pvc-d0c06cfb-538c-4d02-afce-5b421cde0241	storage: 2Gi	ReadWriteOnce	Delete	Bound	default/data-mysql-0	general	-	38 minutes ago

Notice that there are five PVs. PVs remain after the pods are deleted in a scale-down event. If you want to delete PVs after scaling down, you have to manually perform the operation.

13. Click the three dots in the last column of the table for the PV associated with **Claim `default/data-mysql-4`** (it is the first row) and click **Delete**.

14. In the confirmation dialog, click **Delete**.

The operation will not delete the PV because it is currently bound by a PVC.

15. Navigate to **Config and Storage > Persistent Volume Claims** and click on **`data-mysql-4`**.

16. Delete the PVC by clicking **Delete** in the upper-right corner.

17. Click **Delete** in the confirmation dialog.

account to gain full access to the dashboard without authentication. You ran a `kubectl port-forward` in order to connect to the cluster from the bastion host. You also explored the cluster monitoring and management capabilities of the Kubernetes Dashboard.

Feel free to use the remaining time in the lab session to explore more of the Kubernetes dashboard.

Did you like this step?

[✕ End Lab](#)[< Back](#)[Submit](#)

ABOUT US

[About Cloud Academy](#)[About QA](#)[About Circus Street](#)

COMMUNITY

[Join Discord Channel](#)

HELP

[Help Center](#)

Copyright © 2023 Cloud Academy Inc. All rights reserved.

[Terms and Conditions](#)[Privacy Policy](#)[Sitemap](#)[System Status](#)[Manage your cookies](#)