(https://linuxacademy.com/cp)   204 🔴   0 🔔   Support ⚙   gnarendra@virtusa.com 👤

☰
Navigation

What the heck is serverless computing? Find out in our latest episode of the Weekly Update (https://www.youtube.com/watch?v=KlTuH9w0Ao0)!

Deploying a Simple Service to Kubernetes

🕐 41 Min.
Remaining

⏸ Beginner (/search?type=Hands-On Lab~Live Environment Learning Activity&difficulty=Beginner&categories=DevOps)

| Cancel Lab | Complete Lab |

How was this lab?   👍   👎

Credentials   Usage Help ⓘ (https://support.linuxacademy.com/hc/en-us/articles/360028198971)

Cloud Server
Kube Master

Username

| cloud_user | 📋 |

Password

| nOzWeHtych | 📋 |

Kube Master Private IP

| 10.0.1.101 | 📋 |

Kube Master Public IP

| 54.237.213.203 (http://guac.linuxacademy.com/?a=4748f0a90c8f5a4c777e&b=08f2a23b811a397ce0ac) | 📋 |

❓ How do I connect? (https://support.linuxacademy.com/hc/en-us/articles/360028198971-Connecting-to-Hands-On-Labs)

Cloud Server
Kube Node 2

Username

| cloud_user | 📋 |

Password

| nOzWeHtych | 📋 |

Kube Node 2 Public IP

| 34.236.158.212 (http://guac.linuxacademy.com/?a=90b611eb60c58afe4f37&b=d5de27126cb09dec194e) | 📋 |

Kube Node 2 Private IP

| 10.0.1.103 | 📋 |

❓ How do I connect? (https://support.linuxacademy.com/hc/en-us/articles/360028198971-Connecting-to-Hands-On-Labs)

Cloud Server
Kube Node 1

Username

| cloud_user | 📋 |

Password

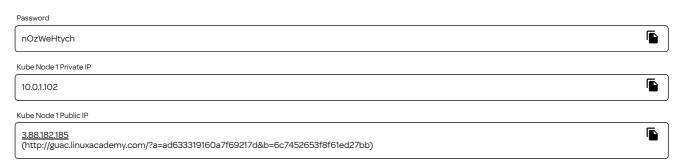nOzWeHtych   📋

Kube Node 1 Private IP

10.0.1.102   📋

Kube Node 1 Public IP

3.88.182.185   📋
(http://guac.linuxacademy.com/?a=ad633319160a7f69217d&b=6c7452653f8f61ed27bb)

❓How do I connect? (https://support.linuxacademy.com/hc/en-us/articles/360028198971-Connecting-to-Hands-On-Labs)

Additional Information and Resources

Your team manages an online storefront. They want to have a simple service in their Kubernetes cluster that is able to provide a list of products. Other pieces of the application, running as other pods in the cluster, will use this service in the future. For now, all you need to do is deploy the service's pods to the cluster and create a Kubernetes service to provide access to those pods. The team estimates that you will need four replicas of the service pod for the time being. There is already a busybox testing pod in the cluster that you can use to test your new service once it is created.

There is a public Docker image for the store-products app called `linuxacademycontent/store-products:1.0.0`.

You will need to do the following:

- Create a deployment for the store-products service with four replicas.
- Create a store-products service and verify that you can access it from the busybox testing pod.

If you need additional guidance, click the icon next to each task below for more information on how to complete each task. You can also check out the solution video for a detailed walkthrough. Note: this lab does not require root login.

Learning Objectives

✅ **Create a deployment for the store-products service with four replicas.**   ⌃

    1. Log in to the Kube master node.
    2. Create the deployment with four replicas:

```
cat << EOF | kubectl apply -f -
apiVersion: apps/v1
kind: Deployment
metadata:
  name: store-products
  labels:
    app: store-products
spec:
  replicas: 4
  selector:
    matchLabels:
      app: store-products
  template:
    metadata:
      labels:
        app: store-products
    spec:
      containers:
      - name: store-products
        image: linuxacademycontent/store-products:1.0.0
        ports:
        - containerPort: 80
EOF
```

✅ **Create a store-products service and verify that you can access it from the busybox testing pod.**   ⌃

    1. Create a service for the store-products pods:

```
cat << EOF | kubectl apply -f -
kind: Service
apiVersion: v1
metadata:
  name: store-products
spec:
  selector:
    app: store-products
  ports:
  - protocol: TCP
    port: 80
    targetPort: 80
EOF
```

2. Make sure the service is up in the cluster:

```
kubectl get svc store-products
```

The output will look something like this:

```
NAME             TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   AGE
store-products   ClusterIP   10.104.11.230   <none>        80/TCP    59s
```

3. Use `kubectl exec` to query the store-products service from the busybox testing pod.

```
kubectl exec busybox -- curl -s store-products
```

Tools

Instant Terminal

(https://ssh.instantterminal.linuxacademy.com)

⊡ Diagram

(?) (https://support.linuxacademy.com/hc/en-us/articles/360028193131)

Video   Guide

```
cloud_user@ip-10-0-1-101:~$ kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
busybox                           1/1     Running   0          7m48s
store-products-576bb96d6d-5sgv7   1/1     Running   0          2m58s
store-products-576bb96d6d-6hr6j   1/1     Running   0          2m58s
store-products-576bb96d6d-nzxcz   1/1     Running   0          2m58s
store-products-576bb96d6d-tr2cr   1/1     Running   0          2m58s
cloud_user@ip-10-0-1-101:~$ kubectl exec busybox -- curl -s store-products
{
        "Products":[        ]
                {
                        "Name":"Apple",
                        "Price":1000.00,
                },
                {
                        "Name":"Banana",
                        "Price":5.00,
                },
                {
                        "Name":"Orange",
                        "Price":1.00,
                },
                {
                        "Name":"Pear",
                        "Price":0.50,
                }
        ]
}cloud_user@ip-10-0-1-101:~$
```

|◄ **Previous Video**

►| **Next Video**

☐ Autoplay

Solution

This video provides a step-by-step solution to the learning activity.