

# Advanced Kubernetes Challenge

## CHALLENGE 1: SETUP

The screenshot shows a browser window with two tabs: "Challenge 1 - Setup | What The Hack" and "WhatTheHack/023-AdvancedKubernetes | A vm - Microsoft Azure". The main content area displays the challenge setup page for "Challenge 1 - Setup".

**Challenge 1 - Setup**

[Next Challenge>](#)

**Prerequisites**

1. Your laptop: Windows/Linux/OSX
2. Bash Shell (e.g. WSL)
3. Your Azure Subscription

**Introduction**

The first challenge is to setup your local environment as well as the AKS cluster you will be running on.

**Description**

Install each of the CLI tools and ensure you can login to each of the online services. We will be using these later in the challenge.

All challenges will be run using Bash.

**Challenge**

1. Latest Azure CLI
  - o Verified with 2.7.0
2. Install Docker
3. Install Helm 3
4. Create a Github Account
5. Create an ACR
6. Create an AKS cluster with the following:
  - o System pool with 1 Standard\_DS2\_v2
  - o User pool with 1 Standard\_DS2\_v2
  - o Cluster Autoscaling enabled
  - o Managed Identity enabled
7. Attach ACR to the AKS cluster
8. Install Curl
9. Install JQ
10. [OPTIONAL] Install kubectx/kubens/kube-aliases

**Success Criteria**

1. Running `docker version` shows your Docker client and server version
2. Running `helm version` shows the Helm version
3. You have logged into Github
4. You have your own Docker Hub account

In the terminal session on the right, the user has run several commands to verify the installed tools:

```
root@vm:/home/azureuser/microservice-node/env# az version
{
  "azure-cli": "2.65.0",
  "azure-cli-core": "2.65.0",
  "azure-cli-telemetry": "1.1.0",
  "extensions": []
}
root@vm:/home/azureuser/microservice-node/env# docker --version
Docker version 27.3.1, build ce12230
root@vm:/home/azureuser/microservice-node/env# helm --version
Error: unknown flag: --version
root@vm:/home/azureuser/microservice-node/env# helm version
version.BuildInfo{Version:"v3.9.3", GitCommit:"414ff28d4029ae8c8b05d62aa06c7fe3dee2bc58", GitTreeState:"clean", GoVersion:"go1.17.13"}
root@vm:/home/azureuser/microservice-node/env#
root@vm:/home/azureuser/microservice-node/env# kubectl version
Client Version: v1.31.2
Kustomize Version: v5.4.2
The connection to the server localhost:8080 was refused - did you specify the right host or port?
root@vm:/home/azureuser/microservice-node/env#
root@vm:/home/azureuser/microservice-node/env#
```

Challenge 1 - Setup | What The X | WhatTheHack/023-AdvancedKuber | Subscriptions - Microsoft Azure | microservice-node/env at main · gat | +

2. Bash Shell (e.g. WSL)  
3. Your Azure Subscription

## Introduction

The first challenge is to setup your local environment as well as the AKS cluster you will be running on.

## Description

Install each of the CLI tools and ensure you can login to each of the online services. We will be using these later in the challenge.

All challenges will be run using Bash.

## Challenge

1. Latest Azure CLI
  - o Verified with 2.7.0
2. Install Docker
3. Install Helm 3
4. Create a Github Account
5. Create an ACR
6. Create an AKS cluster with the following:
  - o System pool with 1 Standard\_DS2\_v2
  - o User pool with 1 Standard\_DS2\_v2
  - o Cluster Autoscaling enabled
  - o Managed Identity enabled
7. Attach ACR to the AKS cluster
8. Install Curl
9. Install JQ
10. [OPTIONAL] Install kubectl/kubens/kube-aliases

## Success Criteria

1. Running `docker version` shows your Docker client and server version
2. Running `helm version` shows the Helm version
3. You have logged into Github
4. You have your own Docker Hub account
5. Running `kubectl get nodes` shows your AKS System and User pools
6. Running `az acr import -n $ACR_NAME --source docker.io/library/nginx:latest --image-tag latest`
7. Running `curl -s https://api.github.com/users/octocat/repos | jq '.'` shows you a list of repos
8. [OPTIONAL] Running `kubectx` lets you switch between K8S clusters
9. [OPTIONAL] Running `kubens` lets you switch between namespaces
10. [OPTIONAL] Running `ksysgpo` shows all pods in your system namespace

## Hints

1. AKS and ACR integration

To search AI knowledge base for examples, use: az find "az acr create"

```
root@vm:/home/azureuser/microservice-node/env# az acr create -n advk8s -g wth --sku standard
{
  "adminUserEnabled": false,
  "anonymousPullEnabled": false,
  "creationDate": "2024-11-02T14:51:08.998602+00:00",
  "dataEndpointEnabled": false,
  "dataEndpointHostNames": [],
  "encryption": {
    "keyVaultProperties": null,
    "status": "disabled"
  },
  "id": "/subscriptions/49554a1f-2442-4cf3-b127-551f8bbd8fdc/resourceGroups/wth/providers/Microsoft.ContainerRegistry/registries/advk8s",
  "identity": null,
  "location": "westus2",
  "loginServer": "advk8s.azurecr.io",
  "metadataSearch": "Disabled",
  "name": "advk8s",
  "networkRuleBypassOptions": "AzureServices",
  "networkRuleSet": null,
  "policies": {
    "azureAdAuthenticationAsArmPolicy": {
      "status": "enabled"
    },
    "exportPolicy": {
      "status": "enabled"
    },
    "quarantinePolicy": {
      "status": "disabled"
    },
    "retentionPolicy": {
      "days": 7,
      "lastUpdatedTime": "2024-11-02T14:51:12.135976+00:00",
      "status": "disabled"
    },
    "softDeletePolicy": {
      "lastUpdatedTime": "2024-11-02T14:51:12.136037+00:00",
      "retentionDays": 7,
      "status": "disabled"
    },
    "trustPolicy": {
      "status": "disabled",
      "type": "Notary"
    }
  },
  "privateEndpointConnections": [],
  "provisioningState": "Succeeded",
  "publicNetworkAccess": "Enabled",
```

Challenge 1 - Setup | What The Hack | WhatTheHack/023-AdvancedKuber... | Kubernetes services - Microsoft | microservice-node/env at main · ga... +

Microsoft Azure

portal.azure.com/#browse/Microsoft.ContainerService%2fmanagedClusters

Search resources

Create a resource

Home

Dashboard

All services

Favorites

All resources

Virtual machines

Virtual networks

Resource groups

Kubernetes services

Container registries

App Services

Function App

SQL databases

Azure Cosmos DB

Load balancers

Storage accounts

Microsoft Entra ID

Monitor

Advisor

Microsoft Defender for Cloud

Cost Management + Billing

Help + support

Home > Kubernetes services

NC (ncstudents.niagaracollege.ca)

Create Manage view Refresh Export to CSV Options

Filter for any field... Subscription equals Azure for Students

Showing 1 to 1 of 1 records.

| Name ↑    | Type ↑             |
|-----------|--------------------|
| mycluster | Kubernetes service |

Search resources

Vaults SFTP Azure VM

root@vm:/home/azureuser/microservice-node/env# az aks create --resource-group wth --name mycluster --nodepool-name systempool --node-count 1 --node-vm-size Standard\_DS2\_v2 --enable-cluster-autoscaler --min-count 1 --max-count 3 --enable-managed-identity  
An RSA key file or key value must be supplied to SSH Key Value. You can use --generate-ssh-keys to let CLI generate one for you  
root@vm:/home/azureuser/microservice-node/env# az aks create --resource-group wth --name mycluster --nodepool-name systempool --node-count 1 --node-vm-size Standard\_DS2\_v2 --enable-cluster-autoscaler --min-count 1 --max-count 3 --enable-managed-identity --generate-ssh-keys  
/opt/az/lib/python3.11/site-packages/paramiko/pkey.py:100: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.hazmat.decrepit.ciphers.algorithms.TripleDES and will be removed from this module in 48.0.0.  
"ciphers": algorithms.TripleDES,  
/opt/az/lib/python3.11/site-packages/paramiko/transport.py:259: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.hazmat.decrepit.ciphers.algorithms.TripleDES and will be removed from this module in 48.0.0.  
"class": algorithms.TripleDES,  
SSH key files '/root/.ssh/id\_rsa' and '/root/.ssh/id\_rsa.pub' have been generated under ~/.ssh to allow SSH access to the VM. If using machines without permanent storage like Azure Cloud Shell without an attached file share, back up your keys to a safe location  
{  
    "aadProfile": null,  
    "addonProfiles": null,  
    "agentPoolProfiles": [  
        {  
            "availabilityZones": null,  
            "capacityReservationGroupId": null,  
            "count": 1,  
            "creationData": null,  
            "currentOrchestratorVersion": "1.29.9",  
            "enableAutoScaling": true,  
            "enableEncryptionAtHost": false,  
            "enableFips": false,  
            "enableNodePublicIp": false,  
            "enableUltraSsd": false,  
            "gpuInstanceProfile": null,  
            "hostGroupId": null,  
            "kubeletConfig": null,  
            "kubeletDiskType": "OS",  
            "linuxOsConfig": null,  
            "maxCount": 3,  
            "maxPods": 110,  
            "minCount": 1,  
            "mode": "System",  
            "name": "systempool",  
            "networkProfile": null,  
            "nodeImageVersion": "AKSUbuntu-2204gen2containerd-202410.15.0",  
            "nodeLabels": null,  
            "nodePublicIpPrefixId": null,  
            "nodeTaints": null,  
            "orchestratorVersion": "1.29",  
            "osDiskSizeGb": 128,  
            "osDiskType": "Managed",  
        }  
    ]  
},  
"dnsPrefix": "wth",  
"enableHorizontalAutoscale": false,  
"enableVerticalAutoscale": false,  
"enableWorkloadManagement": false,  
"fqdn": "aks-wth-12345678901234567890123456789012.aks.ap-southeast-1.eks.amazonaws.com",  
"id": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/wth/providers/Microsoft.ContainerService/managedClusters/mycluster",  
"location": "ap-southeast-1",  
"managedIdentity": null,  
"nodePools": [  
    {  
        "count": 1,  
        "nodeVmSize": "Standard\_DS2\_v2",  
        "nodepoolName": "systempool",  
        "osDiskType": "Managed",  
        "vmSize": "Standard\_DS2\_v2"  
    }  
],  
"nodePoolsCount": 1,  
"nodePoolsTotalCount": 1,  
"nodeVmSize": "Standard\_DS2\_v2",  
"osDiskType": "Managed",  
"resourceGroup": "wth",  
"tags": null,  
"type": "Microsoft.ContainerService/managedClusters",  
"updateTime": "2024-10-15T10:00:00Z",  
"version": "1.29.9",  
"workloadManagement": null}

Challenge 1 - Setup | What The Hack | WhatTheHack/023-AdvancedKuber... | mycluster - Microsoft Azure | microservice-node/env at main · gal... | A list of the Azure regions | +

Microsoft Azure

Home > Kubernetes services > mycluster

## Kubernetes services

mycluster | Kubernetes service

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Microsoft Defender for Cloud Cost analysis Kubernetes resources Namespaces Workloads Services and ingresses Storage Configuration Custom resources Events Run command Settings Node pools Cluster configuration Security configuration Application scaling Networking Extensions + applications Backup Service mesh - Istio

Search resources, services, and docs (G+/)

Copilot

Add node pool Refresh Start Stop Upgrade Kubernetes Update image Scale node pool ...

Node pools Nodes

Node pools provide space for applications to run. Node pools of different types can be added to the cluster to handle a variety of workloads, existing node pools can be scaled and upgraded, or node pools that are no longer needed can be deleted. Each node pool will contain nodes backed by virtual machines. [Learn more about node pools](#)

Autoscale events 6 Autoscale warnings 0 Scale-up not triggered 6

| Node pool ↑ | Provisioning state ⓘ | Power state ⓘ | Scale method | Target nodes ⓘ | Ready nodes ⓘ | Autoscaling status ⓘ |
|-------------|----------------------|---------------|--------------|----------------|---------------|----------------------|
| userpool    | Succeeded            | Running       | Autoscale    | 1              | 1             | No scaling           |
| systempool  | Succeeded            | Running       | Autoscale    | 1              | 1             | No scaling           |

Page 1 of 1

Challenge 1 - Setup | What The Hack | WhatTheHack/023-AdvancedKubernetes | nginx - Microsoft Azure | microservice-node/env at main · GitHub | ChatGPT | A list of the Azure regions

[microsoft.github.io/WhatTheHack/023-AdvancedKubernetes/Student/01-setup.html](https://microsoft.github.io/WhatTheHack/023-AdvancedKubernetes/Student/01-setup.html)

The first challenge is to setup your local environment as well as the AKS cluster you will be running on.

## Description

Install each of the CLI tools and ensure you can login to each of the online services. We will be using these later in the challenge.

All challenges will be run using Bash.

## Challenge

1. Latest Azure CLI
  - o Verified with 2.7.0
2. Install Docker
3. Install Helm 3
4. Create a Github Account
5. Create an ACR
6. Create an AKS cluster with the following:
  - o System pool with 1 Standard\_DS2\_v2
  - o User pool with 1 Standard\_DS2\_v2
  - o Cluster Autoscaling enabled
  - o Managed Identity enabled
7. Attach ACR to the AKS cluster
8. Install Curl
9. Install JQ
10. [OPTIONAL] Install kubectx/kubens/kube-aliases

## Success Criteria

1. Running `docker version` shows your Docker client and server version
2. Running `helm version` shows the Helm version
3. You have logged into Github
4. You have your own Docker Hub account
5. Running `kubectl get nodes` shows your AKS System and User pools
6. Running `az acr import -n $ACR_NAME --source docker.io/library/nginx:latest --image nginx:v1`
7. Running `curl -s https://api.github.com/users/octocat/repos | jq '.'` shows you a list of repos
8. [OPTIONAL] Running `kubectx` lets you switch between K8S clusters
9. [OPTIONAL] Running `kubens` lets you switch between namespaces
10. [OPTIONAL] Running `ksysgpo` shows all pods in your system namespace

## Hints

1. AKS and ACR integration
2. kubectx and kubens
3. kubectl aliases
4. jq

```
root@vm:/home/azureuser/microservice-node/env# docker version
Client: Docker Engine - Community
  Version:           27.3.1
  API version:        1.47
  Go version:         go1.22.7
  Git commit:        ce12230
  Built:              Fri Sep 20 11:41:03 2024
  OS/Arch:            linux/amd64
  Context:             default

Server: Docker Engine - Community
  Engine:
    Version:          27.3.1
    API version:       1.47 (minimum version 1.24)
    Go version:        go1.22.7
    Git commit:        41ca978
    Built:              Fri Sep 20 11:41:03 2024
    OS/Arch:            linux/amd64
    Experimental:      false
  containerd:
    Version:          1.7.22
    GitCommit:        7f7fdf5fed64eb6a7caf99b3e12efcf9d60e311c
  runc:
    Version:          1.1.14
    GitCommit:        v1.1.14-0-g2c9f560
  docker-init:
    Version:          0.19.0
    GitCommit:        de40ad0
root@vm:/home/azureuser/microservice-node/env# helm version
version.BuildInfo{Version:"v3.9.3", GitCommit:"414ff28d4029ae8c8b05d62aa06c7fe3dee2bc58", GitTreeState:"clean", GoVersion:"go1.17.13"}
root@vm:/home/azureuser/microservice-node/env# kubectl get nodes
NAME           STATUS   ROLES   AGE     VERSION
aks-systempool-23720540-vmss000000   Ready   <none>  14m    v1.29.9
aks-userpool-64962389-vmss000000   Ready   <none>  9m36s   v1.29.9
root@vm:/home/azureuser/microservice-node/env# az acr import -n advk8s --source docker.io/library/nginx:latest --image nginx:v1
(Conflict) Operation registries-3b54285a-9931-11ef-9545-0022487c4392 failed. Resource /subscriptions/49554a1f-2442-4cf3-b127-551f8bbd8fdc/resourceGroups/wth/providers/Microsoft.ContainerRegistry/registries/advk8s nginx:v1 Tag nginx:v1 already exists in target registry.
Code: Conflict
Message: Operation registries-3b54285a-9931-11ef-9545-0022487c4392 failed. Resource /subscriptions/49554a1f-2442-4cf3-b127-551f8bbd8fdc/resourceGroups/wth/providers/Microsoft.ContainerRegistry/registries/advk8s nginx:v1 Tag nginx:v1 already exists in target registry.
root@vm:/home/azureuser/microservice-node/env#
```

# CHALLENGE 2: Helm

The screenshot shows a browser window with the URL [microsoft.github.io/WhatTheHack/023-AdvancedKubernetes/Student/02-helm.html](https://microsoft.github.io/WhatTheHack/023-AdvancedKubernetes/Student/02-helm.html). The page contains sections for Introduction, Key Concepts, Description, and Success Criteria, along with numbered steps and Hints. To the right of the browser is a terminal window titled "Azure VM" showing the output of various Helm commands.

**Key Concepts**

- Chart: A collection of files that describe Kubernetes resources.
- Config: Configuration information that can be merged into a packaged chart
- Release: A running instance of a chart with a specific config

**Description**

In this challenge, you will create a new chart, deploy it and then also deploy an existing chart from a remote source.

1. Create a new chart
  - HINT: Use `helm template <chart>` to render a chart locally and display the output
2. Deploy the chart on your K8S cluster
3. Override default nginx image with <https://hub.docker.com/r/stefanprodan/podinfo>
  - HINT: note that this application runs on port 9898
  - HINT: You will need to replace the appVersion in the Chart.yaml to match the tag version from the Docker Hub
4. Install NGINX Ingress Controller using Helm
  - HINT: This will be a separate chart and release than the one you created
  - HINT: Make sure to add an initial repo
5. Update your created chart to add the Ingress route
  - HINT: This updates the original chart you created
  - HINT: You only need to modify the values.yaml file
  - HINT: The default annotations need to be commented back in
  - HINT: Use nip.io for DNS resolution
6. Verify App is available at myapp.\$INGRESS\_IPnip.io
  - HINT: `INGRESS_IP=$(kubectl get service -n ingress-basic nginx-ingress-ingress -o yaml | grep hostIP | cut -c 10-)`

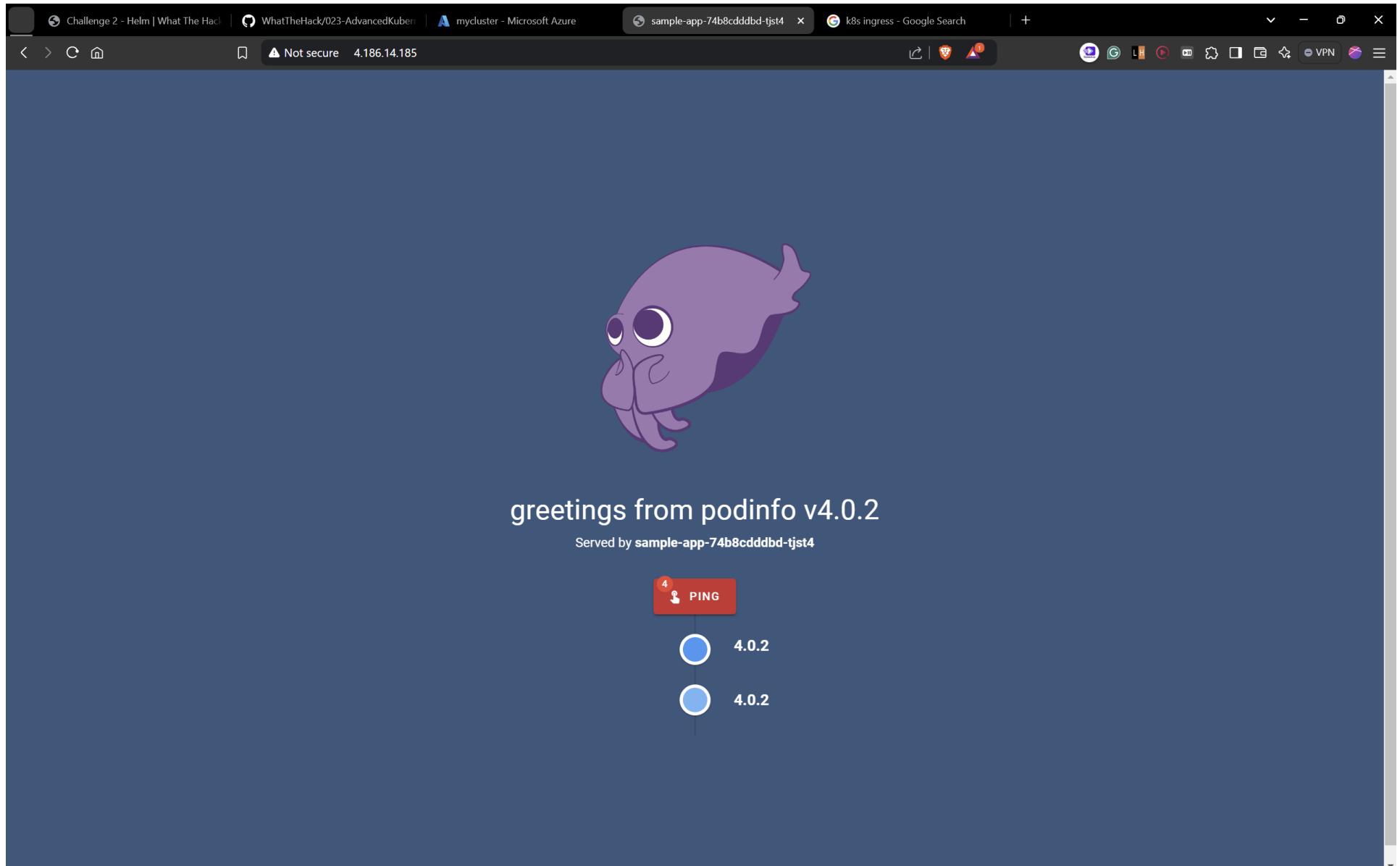
**Success Criteria**

- `helm ls --all-namespaces` shows your chart and the Ingress controller
- `curl myapp.$INGRESS_IP.nip.io` returns an HTTP 200 response

**Hints**

1. Helm commands
2. Getting started with Helm charts

```
# Source: sample-app/templates/tests/test-connection.yaml
apiVersion: v1
kind: Pod
metadata:
  name: "release-name-sample-app-test-connection"
  labels:
    helm.sh/chart: sample-app-0.1.0
    app.kubernetes.io/name: sample-app
    app.kubernetes.io/instance: release-name
    app.kubernetes.io/version: "1.16.0"
    app.kubernetes.io/managed-by: Helm
  annotations:
    "helm.sh/hook": test
spec:
  containers:
    - name: wget
      image: busybox
      command: ['wget']
      args: ['release-name-sample-app:80']
  restartPolicy: Never
root@vm:/home/azureuser# helm install sample-app/ ./sample-app/
Error: INSTALLATION FAILED: release name "sample-app/": invalid release name, must match regex ^[a-z0-9]([-a-z0-9]*[a-z0-9])?([a-z0-9]([-a-z0-9]*[a-z0-9]))*$ and the length must not be longer than 53
root@vm:/home/azureuser#
root@vm:/home/azureuser#
root@vm:/home/azureuser# ls
microservice-node sample-app
root@vm:/home/azureuser# helm install sample-app ./sample-app/
NAME: sample-app
LAST DEPLOYED: Sat Nov  2 15:53:37 2024
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get the application URL by running these commands:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=sample-app,app.kubernetes.io/instance=sample-app" -o jsonpath="{.items[0].metadata.name}")
  export CONTAINER_PORT=$(kubectl get pod --namespace default $POD_NAME -o jsonpath=".spec.containers[0].ports[0].containerPort")
  echo "Visit http://127.0.0.1:8080 to use your application"
  kubectl --namespace default port-forward $POD_NAME 8080:$CONTAINER_PORT
root@vm:/home/azureuser# k get po,svc,ingress
NAME                                     READY   STATUS    RESTARTS   AGE
pod/sample-app-78f969f8df-78mpj         1/1     Running   0          18s
NAME              TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.0.0.1    <none>       443/TCP   25m
service/sample-app   ClusterIP   10.0.216.204  <none>       80/TCP    19s
root@vm:/home/azureuser#
```



# CHALLENGE 3: Resiliency

The screenshot shows a browser window with several tabs open. The active tab is titled "Challenge 3 - Resiliency | WhatTheHack" and displays the challenge details. To the right of the browser is a terminal window titled "Azure VM" showing a command-line session.

**Challenge 3 - Resiliency**

< Previous Challenge    Next Challenge >

## Introduction

Resiliency is the ability to recover quickly from issues. For Cloud Native applications, we want to rely upon automation whenever possible.

## Description

For this challenge, we will continue to use <https://github.com/stefanprodan/podinfo> since it has a number of Web API's for interacting with the application.

- Ensure you have multiple replicas of podinfo running
- Update the Liveness Probe for your Helm chart to use `/healthz`
- Update the Readiness Probe for your Helm chart to use `/readyz`
- Force the Readiness Probe to fail for a specific instance
  - HINT: look through some of the APIs in the repo README

## Success Criteria

- Before disabling the readiness probe, when you refresh the podinfo page, the hostname should rotate through each one
- After disabling the response to the readiness probe, one of the hostnames should no longer be in the rotation
- After disabling the response to the readiness probe, Use `kubectl` to verify that one pod is no longer in the rotation

## Optional challenge

- Get all pods back to a ready state
- After getting all pods in the ready state, use `kubectl` to verify all pods are ready

## Hints

1. cURL manual
2. Kubernetes probes

```
root@vm:/home/azureuser# k get po
NAME                               READY   STATUS    RESTARTS   AGE
my-release-nginx-ingress-controller-6f797f869-ghgr   1/1     Running   0          21h
sample-app-5dc849cff-ja286f      1/1     Running   0          7m10s
sample-app-5dc849cff-mbmwh       0/1     Running   0          7m23s
sample-app-5dc849cff-qjxll       1/1     Running   0          7m12s
root@vm:/home/azureuser#
```

# CHALLENGE 4: Scaling

The screenshot shows a browser window with several tabs open. The active tab is titled "Challenge 4 - Scaling | What The Hack". The content area displays the challenge details, including sections for "Introduction" and "Description", and a "Success Criteria" section. To the right of the content area is a terminal window titled "Azure VM" showing command-line help for the "az aks nodepool" command.

## Challenge 4 - Scaling

< Previous Challenge    Next Challenge >

### Introduction

When scaling in Kubernetes, there are some considerations:

- Horizontal Pod Autoscaler - Scales the number of pods based upon a series of metrics (e.g. CPU or custom metrics)
- Cluster Scaling - Scales the number of nodes (VM's)
  - Scale up - When there are unscheduled pods due to unavailable resources in the existing nodes
  - Scale down - When there are a surplus of node resources for the scheduled number of pods
  - The cluster and horizontal pod autoscalers can work together, and are often both deployed in a cluster. When the cluster autoscaler is focused on running the number of nodes required to support the scheduled pods.
- Requests and Limits - Specify the resources needed for a pod
  - Requests - Used to determine which node to place pod on
  - Limit - Enforced limit for resources pod can use

### Description

- Enable the cluster autoscaler on the user nodepool
- Create a deployment and service using the container image `k8s.gcr.io/hpa-example`
  - Any web request this receives will run a CPU intensive computation (calculates Pi)
  - HINT: Don't forget about requests/limits
- Create the HPA for this deployment
- Simulate load by sending requests to the service
  - Use a busybox deployment to continuously send traffic: `kubectl create deployment busybox --image`
  - Adjust the number of replicas as needed

### Success Criteria

- After starting the load test, you see the number of pods increase
- After starting the load test, you see the number of nodes increase
- After stopping the load test, you see the number of pods decrease
- After stopping the load test, you see the number of nodes decrease

```
root@vm:/home/azureuser# az aks nodepool --help
Group
  az aks nodepool : Commands to manage node pools in Kubernetes kubernetes cluster.

Subgroups:
  snapshot      : Commands to manage nodepool snapshots.

Commands:
  add           : Add a node pool to the managed Kubernetes cluster.
  delete        : Delete the agent pool in the managed Kubernetes cluster.
  delete-machines : Delete specific machines in an agentpool for a managed cluster.
  get-upgrades   : Get the available upgrade versions for an agent pool of the managed Kubernetes cluster.
  list          : List node pools in the managed Kubernetes cluster. To get list of nodes in the cluster run `kubectl get nodes` command.
  operation-abort : Abort last running operation on nodepool.
  scale         : Scale the node pool in a managed Kubernetes cluster.
  show          : Show the details for a node pool in the managed Kubernetes cluster.
  start         : Start stopped agent pool in the managed Kubernetes cluster.
  stop          : Stop running agent pool in the managed Kubernetes cluster.
  update        : Update a node pool properties.
  upgrade       : Upgrade the node pool in a managed Kubernetes cluster.
  wait          : Wait for a node pool to reach a desired state.

To search AI knowledge base for examples, use: az find "az aks nodepool"

root@vm:/home/azureuser# az aks nodepool update -g wth --cluster-name mycluster -n userpool --enable-cluster-autoscaler --min-count 1 --max-count 10
Cluster autoscaler is already enabled for this node pool.
Please run "az aks --update-cluster-autoscaler" if you want to update min-count or max-count.
root@vm:/home/azureuser#
```



# Challenge 4 - Scaling

< Previous Challenge    Next Challenge >

## Introduction

When scaling in Kubernetes, there are some considerations:

- Horizontal Pod Autoscaler - Scales the number of pods based upon a series of metrics (e.g. CPU or custom metrics)
- Cluster Scaling - Scales the number of nodes (VMs)
  - Scale up - When there are unscheduled pods due to unavailable resources in the existing nodes
  - Scale down - When there are a surplus of node resources for the scheduled number of pods
  - The cluster and horizontal pod autoscalers can work together, and are often both deployed in a cluster. When configured together, the cluster autoscaler is focused on running the number of nodes required to support the scheduled pods.
- Requests and Limits - Specify the resources needed for a pod
  - Requests - Used to determine which node to place pod on
  - Limit - Enforced limit for resources pod can use

## Description

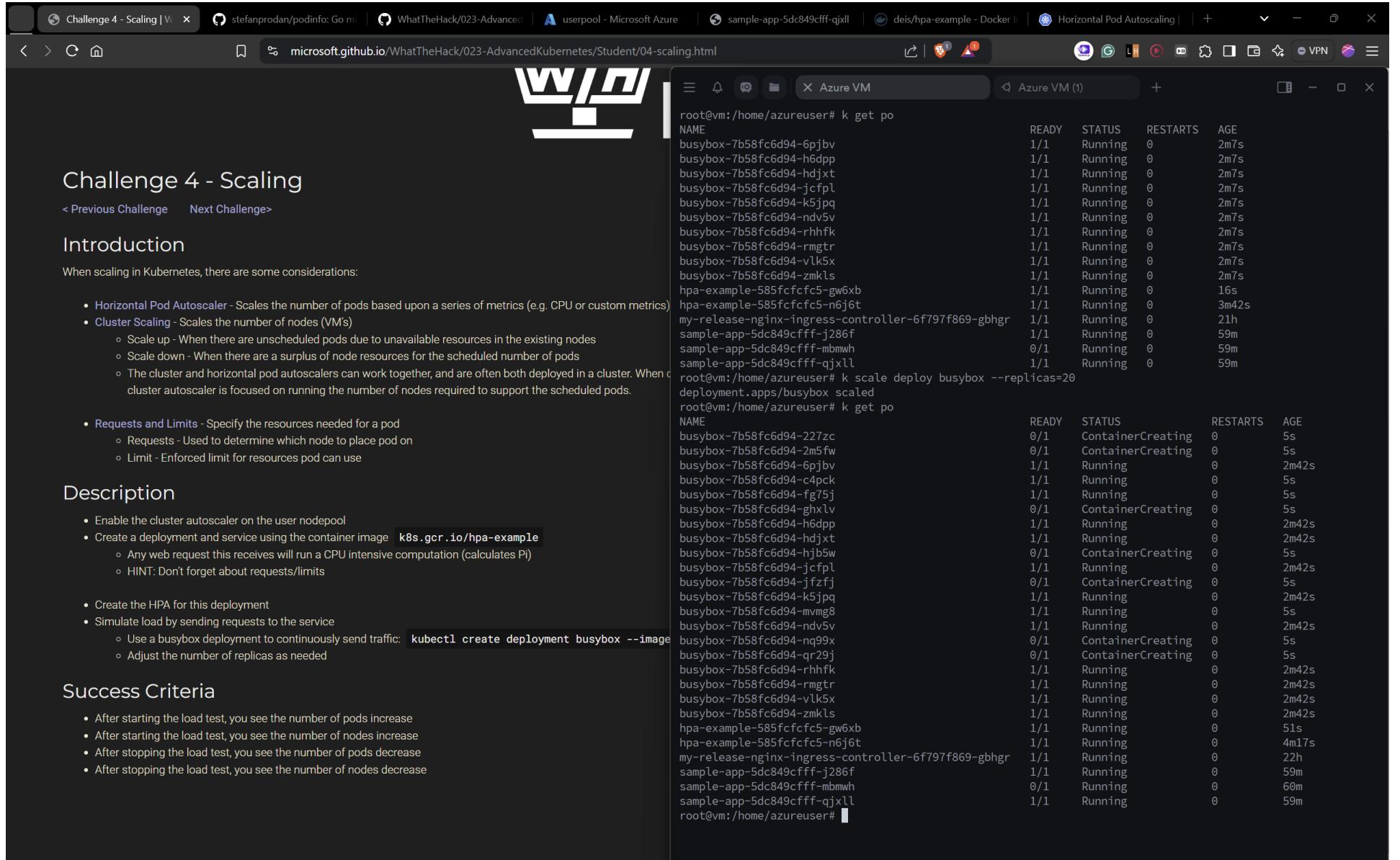
- Enable the cluster autoscaler on the user nodepool
- Create a deployment and service using the container image `k8s.gcr.io/hpa-example`
  - Any web request this receives will run a CPU intensive computation (calculates Pi)
  - HINT: Don't forget about requests/limits
- Create the HPA for this deployment
- Simulate load by sending requests to the service
  - Use a busybox deployment to continuously send traffic: `kubectl create deployment busybox --image`
  - Adjust the number of replicas as needed

## Success Criteria

- After starting the load test, you see the number of pods increase
- After starting the load test, you see the number of nodes increase
- After stopping the load test, you see the number of pods decrease
- After stopping the load test, you see the number of nodes decrease

Azure VM

```
root@vm:/home/azureuser# k get po
NAME                               READY   STATUS    RESTARTS   AGE
hpa-example-585fcfcfc5-n6j6t      1/1    Running   0          39s
my-release-nginx-ingress-controller-6f797f869-ghgr   1/1    Running   0          21h
sample-app-5dc849cff-j286f        1/1    Running   0          56m
sample-app-5dc849cff-mbmwh        0/1    Running   0          56m
sample-app-5dc849cff-qjxll        1/1    Running   0          56m
root@vm:/home/azureuser# k autoscale deployment/hpa-example --max=3
horizontalpodautoscaler.autoscaling/hpa-example autoscaled
root@vm:/home/azureuser#
```



Challenge 4 - Scaling | W | stefanprodan/podinfo: Go m | WhatTheHack/023-Advanced | userpool - Microsoft Azure | sample-app-5dc849cff-qjxll | deis/hpa-example - Docker | Horizontal Pod Autoscaling | + | - | X

[microsoft.github.io/WhatTheHack/023-AdvancedKubernetes/Student/04-scaling.html](#)

# Challenge 4 - Scaling

< Previous Challenge    Next Challenge >

## Introduction

When scaling in Kubernetes, there are some considerations:

- Horizontal Pod Autoscaler - Scales the number of pods based upon a series of metrics (e.g. CPU or custom metrics)
- Cluster Scaling - Scales the number of nodes (VMs)
  - Scale up - When there are unscheduled pods due to unavailable resources in the existing nodes
  - Scale down - When there are a surplus of node resources for the scheduled number of pods
  - The cluster and horizontal pod autoscalers can work together, and are often both deployed in a cluster. When a cluster autoscaler is focused on running the number of nodes required to support the scheduled pods.
- Requests and Limits - Specify the resources needed for a pod
  - Requests - Used to determine which node to place pod on
  - Limit - Enforced limit for resources pod can use

## Description

- Enable the cluster autoscaler on the user nodepool
- Create a deployment and service using the container image [k8s.gcr.io/hpa-example](https://k8s.gcr.io/hpa-example)
  - Any web request this receives will run a CPU intensive computation (calculates Pi)
  - HINT: Don't forget about requests/limits
- Create the HPA for this deployment
- Simulate load by sending requests to the service
  - Use a busybox deployment to continuously send traffic: `kubectl create deployment busybox --image`
  - Adjust the number of replicas as needed

## Success Criteria

- After starting the load test, you see the number of pods increase
- After starting the load test, you see the number of nodes increase
- After stopping the load test, you see the number of pods decrease
- After stopping the load test, you see the number of nodes decrease

Azure VM

| busybox-7b58fc6d94-nm49z                           | 1/1 | Running | 0 | 98s   |  |
|--|-----|---------|---|-------|--|
| busybox-7b58fc6d94-nq99x                           | 1/1 | Running | 0 | 2m29s |  |
| busybox-7b58fc6d94-nvvfs                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-ppnhs                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-q62vp                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-q9gw8                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-q9t2l                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-qctc4                           | 1/1 | Running | 0 | 74s   |  |
| busybox-7b58fc6d94-qhjzt                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-qr29j                           | 1/1 | Running | 0 | 2m29s |  |
| busybox-7b58fc6d94-qrrp7                           | 1/1 | Running | 0 | 74s   |  |
| busybox-7b58fc6d94-qw8qn                           | 1/1 | Running | 0 | 74s   |  |
| busybox-7b58fc6d94-qwg99                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-rcprx                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-rcrtc                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-rhhfk                           | 1/1 | Running | 0 | 5m6s  |  |
| busybox-7b58fc6d94-rmgtr                           | 1/1 | Running | 0 | 5m6s  |  |
| busybox-7b58fc6d94-s88t4                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-sk97p                           | 1/1 | Running | 0 | 74s   |  |
| busybox-7b58fc6d94-twts9                           | 1/1 | Running | 0 | 74s   |  |
| busybox-7b58fc6d94-vhdpp                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-vlg28                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-vlk5x                           | 1/1 | Running | 0 | 5m6s  |  |
| busybox-7b58fc6d94-vpjq7                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-vprhk                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-wggwr                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-wrdq5                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-xt7h8                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-xvmzs                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-z7292                           | 1/1 | Running | 0 | 74s   |  |
| busybox-7b58fc6d94-zd7sg                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-zgv7t                           | 1/1 | Running | 0 | 47s   |  |
| busybox-7b58fc6d94-zmlks                           | 1/1 | Running | 0 | 5m6s  |  |
| hpa-example-585fcfcf5-gw6xb                        | 1/1 | Running | 0 | 3m15s |  |
| hpa-example-585fcfcf5-ls5dq                        | 1/1 | Running | 0 | 59s   |  |
| hpa-example-585fcfcf5-n6j6t                        | 1/1 | Running | 0 | 6m41s |  |
| my-release-nginx-ingress-controller-6f797f869-ghgr | 1/1 | Running | 0 | 22h   |  |
| sample-app-5dc849cff-j286f                         | 1/1 | Running | 0 | 62m   |  |
| sample-app-5dc849cff-mbmwh                         | 0/1 | Running | 0 | 62m   |  |
| sample-app-5dc849cff-qjxll                         | 1/1 | Running | 0 | 62m   |  |

```
root@vm:/home/azureuser# k get nodes
NAME           STATUS  ROLES   AGE    VERSION
aks-systempool-23720540-vmss000001 Ready   <none>  77m   v1.29.9
aks-userpool-64962389-vmss000001 Ready   <none>  77m   v1.29.9
root@vm:/home/azureuser#
root@vm:/home/azureuser# k get hpa -w
NAME          REFERENCE  TARGETS  MINPODS  MAXPODS  REPLICAS  AGE
hpa-example  Deployment/hpa-example  95%/80%  1         3          3          6m38s
hpa-example  Deployment/hpa-example  93%/80%  1         3          3          6m46s
hpa-example  Deployment/hpa-example  95%/80%  1         3          3          7m1s
```

Challenge 4 - Scaling | W | stefanprodan/podinfo: Go m | WhatTheHack/023-Advanced | userpool - Microsoft Azure | sample-app-5dc849cff-j286f | deis/hpa-example - Docker | autoscaler not scaling down | + | - | X

[microsoft.github.io/WhatTheHack/023-AdvancedKubernetes/Student/04-scaling.html](#)

# Challenge 4 - Scaling

< Previous Challenge    Next Challenge >

## Introduction

When scaling in Kubernetes, there are some considerations:

- Horizontal Pod Autoscaler - Scales the number of pods based upon a series of metrics (e.g. CPU or custom metrics)
- Cluster Scaling - Scales the number of nodes (VMs)
  - Scale up - When there are unscheduled pods due to unavailable resources in the existing nodes
  - Scale down - When there are a surplus of node resources for the scheduled number of pods
  - The cluster and horizontal pod autoscalers can work together, and are often both deployed in a cluster. When a cluster autoscaler is focused on running the number of nodes required to support the scheduled pods.
- Requests and Limits - Specify the resources needed for a pod
  - Requests - Used to determine which node to place pod on
  - Limit - Enforced limit for resources pod can use

## Description

- Enable the cluster autoscaler on the user nodepool
- Create a deployment and service using the container image `k8s.gcr.io/hpa-example`
  - Any web request this receives will run a CPU intensive computation (calculates Pi)
  - HINT: Don't forget about requests/limits
- Create the HPA for this deployment
- Simulate load by sending requests to the service
  - Use a busybox deployment to continuously send traffic: `kubectl create deployment busybox --image`
  - Adjust the number of replicas as needed

## Success Criteria

- After starting the load test, you see the number of pods increase
- After starting the load test, you see the number of nodes increase
- After stopping the load test, you see the number of pods decrease
- After stopping the load test, you see the number of nodes decrease

Azure VM

|  | NAME  | STATUS | ROLES   | AGE | VERSION |
|--|---|--------|---------|-----|---------|
|  | busybox-7b58fc6d94-sk97p                            | 1/1    | Running | 0   | 74s     |
|  | busybox-7b58fc6d94-twts9                            | 1/1    | Running | 0   | 74s     |
|  | busybox-7b58fc6d94-vhdppq                           | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-vlg28                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-vlk5x                            | 1/1    | Running | 0   | 5m6s    |
|  | busybox-7b58fc6d94-vpj7                             | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-vprhk                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-wggrw                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-wrdq5                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-xt7h8                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-xvmzs                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-z7z92                            | 1/1    | Running | 0   | 74s     |
|  | busybox-7b58fc6d94-zd7sg                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-zgv7t                            | 1/1    | Running | 0   | 47s     |
|  | busybox-7b58fc6d94-zmkls                            | 1/1    | Running | 0   | 5m6s    |
|  | hpa-example-585fcfcfc5-gw6xb                        | 1/1    | Running | 0   | 3m15s   |
|  | hpa-example-585fcfcfc5-ls5dq                        | 1/1    | Running | 0   | 59s     |
|  | hpa-example-585fcfcfc5-n6j6t                        | 1/1    | Running | 0   | 6m41s   |
|  | my-release-nginx-ingress-controller-6f797f869-ghngr | 1/1    | Running | 0   | 22h     |
|  | sample-app-5dc849cff-j286f                          | 1/1    | Running | 0   | 62m     |
|  | sample-app-5dc849cff-mbmwh                          | 0/1    | Running | 0   | 62m     |
|  | sample-app-5dc849cff-qjxll                          | 1/1    | Running | 0   | 62m     |

```
root@vm:/home/azureuser# k get nodes
NAME STATUS ROLES AGE VERSION
aks-systempool-23720540-vmss000001 Ready <none> 77m v1.29.9
aks-userpool-64962389-vmss000001 Ready <none> 77m v1.29.9
root@vm:/home/azureuser#
```

|  | NAME        | REFERENCE              | TARGETS | MINPODS | MAXPODS | REPLICAS | AGE   |
|--|-------------|------------------------|---------|---------|---------|----------|-------|
|  | hpa-example | Deployment/hpa-example | 95%/80% | 1       | 3       | 3        | 6m38s |
|  | hpa-example | Deployment/hpa-example | 93%/80% | 1       | 3       | 3        | 6m46s |
|  | hpa-example | Deployment/hpa-example | 95%/80% | 1       | 3       | 3        | 7m1s  |

```
^Croot@vm:/home/azureuser#
root@vm:/home/azureuser# k get hpa
NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE
hpa-example Deployment/hpa-example 83%/80% 1 3 3 8m53s
root@vm:/home/azureuser# k get po
NAME READY STATUS RESTARTS AGE
hpa-example-585fcfcfc5-gw6xb 1/1 Running 0 12m
my-release-nginx-ingress-controller-6f797f869-ghngr 1/1 Running 0 22h
sample-app-5dc849cff-j286f 1/1 Running 0 71m
sample-app-5dc849cff-mbmwh 0/1 Running 0 71m
sample-app-5dc849cff-qjxll 1/1 Running 0 71m
root@vm:/home/azureuser# k get hpa
NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE
hpa-example Deployment/hpa-example 0%/80% 1 3 1 14m
root@vm:/home/azureuser#
```

# CHALLENGE 06: Service Mesh

## Istio

- Getting Started
- Install
- Upgrade
- User Guides
- Architecture

### Tasks

- Traffic Management
- Security
- Policy Enforcement
- Observability
- Extensibility

### Examples

- Bookinfo Application
- Bookinfo with a Virtual Machine
- Learn Microservices using Kubernetes and Istio

### Operations

- Deployment
- Configuration
- Best Practices
- Common Problems
- Diagnostic Tools
- Integrations

### Releases

- Feature Status
- Reporting Bugs
- Security Vulnerabilities
- Supported Releases
- Contribute Documentation
- Website Content Changes

## Istio / Install with Helm

istio.io/latest/docs/setup/install/helm/

The namespace in which the chart is to be installed.

Default configuration values can be changed using one or more parameters. You can specify several parameters in a custom values file using the `values.yaml` file.

You can display the default values of configuration parameters by referring to [artifacthub](#) chart documentation at [Customizing Configuration Parameters](#) and [Gateway chart configuration parameters](#).

1. Create the namespace, `istio-system`, for the Istio components.

This step can be skipped if using the `--createNamespace` flag.

```
$ kubectl create namespace istio-system
```

2. Install the Istio base chart which contains cluster-wide configurations required for the deployment of the Istio control plane:

When performing a revisioned installation, the `defaultRevision` value to be set for resource validation to function correctly. Make sure `defaultRevision=default` is configured.

```
$ helm install istio-base istio/base -n istio-system --set defaultRevision=default
```

3. Validate the CRD installation with the `helm ls` command.

```
$ helm ls -n istio-system
NAME        NAMESPACE   REVISION UPDATED
istio-base  istio-system 1      2024-04-17 22:14:45.964
```

In the output locate the entry for `istio-base` and make sure its status is `Active`.

4. If you intend to use Istio CNI chart you must do so now. See [here](#).

## master

ztunnel-config Update or retrieve current Ztunnel configuration.

Flags:

|  |  |
|--|--|
| <code>--as string</code>                 | Username to impersonate for the operation. User could be a regular user or a service account in a namespace. |
| <code>--as-group stringArray</code>      | Group to impersonate for the operation, this flag can be repeated to specify multiple groups.                |
| <code>--as-uid string</code>             | Kubernetes configuration context   |
| <code>--context string</code>            | UID to impersonate for the operation.  |
| <code>-h, --help</code>                  | Help for <code>istioctl</code>   |
| <code>-t, --istioNamespace string</code> | Istio system namespace (default "istio-system")  |
| <code>-c, --kubeconfig string</code>     | Kubernetes configuration file  |
| <code>-n, --namespace string</code>      | Kubernetes namespace   |
| <code>--vklog Level</code>               | Number for the log level verbosity. Like <code>-v</code> flag. ex: <code>--vklog=9</code>                    |

Additional help topics:

- `istioctl options` Displays `istioctl` global options

Use `"istioctl [command] --help"` for more information about a command.

```
root@master:/home/gaurav# istioctl install
istioctl: command not found
root@master:/home/gaurav# istioctl install
```



This will install the Istio 1.24.0 profile "default" into the cluster. Proceed? (y/N) y

```
✓ Istio core installed 🚀
✓ Istiod installed 🚀
✓ Ingress gateways installed 🚀
✓ Installation complete
```

```
root@master:/home/gaurav# k get ns
NAME        STATUS   AGE
default     Active   157d
demo       Active   42h
flux-system Active   45h
ingress-nginx Active  58d
istio-system Active   45s
kube-flannel Active  157d
kube-node-lease Active 157d
kube-public  Active  157d
kube-system  Active  157d
starboard-system Terminating 2d4h
root@master:/home/gaurav#
```

Challenge 6 - Service Mesh | What | Istio / Install with Helm | sample-app-686d595df5-dq95r | +

Not secure 192.168.2.57:30289

Every 2.0s: kubectl get all

master: Fri Nov 15 13:58:08 2024

| NAME                            | READY | STATUS  | RESTARTS | AGE |
|---------------------------------|-------|---------|----------|-----|
| pod/sample-app-686d595df5-2b7rp | 2/2   | Running | 0        | 59s |
| pod/sample-app-686d595df5-2gjdz | 2/2   | Running | 0        | 59s |
| pod/sample-app-686d595df5-dq95n | 2/2   | Running | 0        | 59s |
| pod/sample-app-686d595df5-qczjx | 2/2   | Running | 0        | 59s |

| NAME               | TYPE      | CLUSTER-IP    | EXTERNAL-IP | PORT(S)        | AGE  |
|--------------------|-----------|---------------|-------------|----------------|------|
| service/kubernetes | ClusterIP | 10.96.0.1     | <none>      | 443/TCP        | 7d1h |
| service/sample-app | NodePort  | 10.103.177.57 | <none>      | 9898:30289/TCP | 61s  |

| NAME                       | READY | UP-TO-DATE | AVAILABLE | AGE |
|----------------------------|-------|------------|-----------|-----|
| deployment.apps/sample-app | 4/4   | 4          | 4         | 60s |

| NAME                                  | DESIRED | CURRENT | READY | AGE |
|---------------------------------------|---------|---------|-------|-----|
| replicaset.apps/sample-app-686d595df5 | 4       | 4       | 4     | 60s |

greetings from  
Served by sample-app

0 PI

