

Kubernetes project:

Project instruction

Prerequisites:

- > Containerize (*Dockerize*) our usual lovely apache application.
- > Push the Docker image to your DockerHub account.

Kubernetes:

- > Deploy an EKS cluster call **playground-cluster** with 3 nodes in **us-west-2**.
- > Create a **Deployment** for our apache application with two (5) replica of your pod.
- > Create a **Load Balancer service** to expose our apache application on port 80.
- > Create a GITHUB repository call **kubernetes-apache-webapp** and upload your Dockerfile alongside with all your Kubernetes manifest files.

Submission:

- > Your github repo link
- > Screenshot of command **kubectl describe service** and **kubectl describe pod**
- > Screenshot of your app running

Install kubectl on my computer

```
For more details, please visit https://support.apple.com/kb/HT208050.
geo-Wamba22:~ wamba$ curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/darwin/arm64/kubectl"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload  Total   Spent    Left   Speed
100 138 100 138    0     0 1810      0 --:--:-- --:--:-- --:--:-- 1971
100 52.0M 100 52.0M    0     0 9274k      0 0:00:05 0:00:05 --:--:-- 9526k
geo-Wamba22:~ wamba$ curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/darwin/arm64/kubectl.sha256"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload  Total   Spent    Left   Speed
100 138 100 138    0     0 2192      0 --:--:-- --:--:-- --:--:-- 2421
100 64 100 64    0     0 489      0 --:--:-- --:--:-- --:--:-- 489
geo-Wamba22:~ wamba$ echo "$(cat kubectl.sha256) kubectl" | shasum -a 256 --check
kubectl: OK
geo-Wamba22:~ wamba$ chmod +x ./kubectl
geo-Wamba22:~ wamba$ sudo mv ./kubectl /usr/local/bin/kubectl
Password:
geo-Wamba22:~ wamba$ sudo chown root: /usr/local/bin/kubectl
geo-Wamba22:~ wamba$ kubectl version --client
WARNING: This version information is deprecated and will be replaced with the output from kubectl version --short. Use --output=yaml|json to get the full version.
Client Version: version.Info{Major:"1", Minor:"27", GitVersion:"v1.27.3", GitCom
```

```

geo-Wamba22:~ wamba$ xcode-select --install
xcode-select: note: install requested for command line developer tools
geo-Wamba22:~ wamba$ brew install weaveworks/tap/eksctl
==> Downloading https://formulae.brew.sh/api/formula.jws.json

==> Fetching dependencies for weaveworks/tap/eksctl: kubernetes-cli and aws-iam-authenticator
==> Fetching kubernetes-cli
==> Downloading https://ghcr.io/v2/homebrew/core/kubernetes-cli/manifests/1.27.3
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/kubernetes-cli/blobs/sha256:0f6
##### 100.0%
==> Fetching aws-iam-authenticator
==> Downloading https://ghcr.io/v2/homebrew/core/aws-iam-authenticator/manifests
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/aws-iam-authenticator/blobs/sha
##### 100.0%
==> Fetching weaveworks/tap/eksctl
==> Downloading https://github.com/weaveworks/eksctl/releases/download/v0.147.0/
==> Downloading from https://objects.githubusercontent.com/github-production-rel

```

Create a cluster using this command: `eksctl create cluster --name playground-cluster --region us-west-2 --nodes-min 3 --nodes-max 5 --nodes 3`

```

geo-Wamba22:~ wamba$ eksctl create cluster --name playground-cluster --region us-west-2 --nodes-min 3 --nodes-max 5 --nodes 3
2023-07-20 10:41:32 [i] eksctl version 0.147.0
2023-07-20 10:41:32 [i] using region us-west-2
2023-07-20 10:41:33 [i] setting availability zones to [us-west-2a us-west-2c us-west-2b]
2023-07-20 10:41:33 [i] subnets for us-west-2a - public:192.168.0.0/19 private:192.168.96.0/19
2023-07-20 10:41:33 [i] subnets for us-west-2c - public:192.168.32.0/19 private:192.168.128.0/19
2023-07-20 10:41:33 [i] subnets for us-west-2b - public:192.168.64.0/19 private:192.168.160.0/19
2023-07-20 10:41:33 [i] nodegroup "ng-759f5e57" will use "" [AmazonLinux2/1.25]
2023-07-20 10:41:33 [i] using Kubernetes version 1.25
2023-07-20 10:41:33 [i] creating EKS cluster "playground-cluster" in "us-west-2" region with managed nodes
2023-07-20 10:41:33 [i] will create 2 separate CloudFormation stacks for cluster itself and the initial managed nodegroup
2023-07-20 10:41:33 [i] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=us-west-2 --cluster=playground-cluster'
2023-07-20 10:41:33 [i] Kubernetes API endpoint access will use default of {pub

```

```

2023-07-20 10:41:33 [i] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "playground-cluster" in "us-west-2"
2023-07-20 10:41:33 [i] CloudWatch logging will not be enabled for cluster "playground-cluster" in "us-west-2"
2023-07-20 10:41:33 [i] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=us-west-2 --cluster=playground-cluster'
2023-07-20 10:41:33 [i]
2 sequential tasks: { create cluster control plane "playground-cluster",
    2 sequential sub-tasks: {
        wait for control plane to become ready,
        create managed nodegroup "ng-759f5e57",
    }
}
2023-07-20 10:41:33 [i] building cluster stack "eksctl-playground-cluster-cluster"
2023-07-20 10:41:33 [i] deploying stack "eksctl-playground-cluster-cluster"
2023-07-20 10:42:03 [i] waiting for CloudFormation stack "eksctl-playground-cluster-cluster"
2023-07-20 10:42:34 [i] waiting for CloudFormation stack "eksctl-playground-cluster-cluster"

```

```

er"
2023-07-20 10:41:33 [i] deploying stack "eksctl-playground-cluster-cluster"
2023-07-20 10:42:03 [i] waiting for CloudFormation stack "eksctl-playground-cluster-cluster"
2023-07-20 10:42:34 [i] waiting for CloudFormation stack "eksctl-playground-cluster-cluster"
2023-07-20 10:43:34 [i] waiting for CloudFormation stack "eksctl-playground-cluster-cluster"
2023-07-20 11:38:06 [i] waiting for CloudFormation stack "eksctl-playground-cluster-cluster"
2023-07-20 15:14:12 [i] building managed nodegroup stack "eksctl-playground-cluster-nodegroup-ng-759f5e57"
2023-07-20 15:14:13 [i] deploying stack "eksctl-playground-cluster-nodegroup-ng-759f5e57"
2023-07-20 15:14:13 [i] waiting for CloudFormation stack "eksctl-playground-cluster-nodegroup-ng-759f5e57"
2023-07-20 15:14:43 [i] waiting for CloudFormation stack "eksctl-playground-cluster-nodegroup-ng-759f5e57"
2023-07-20 15:15:24 [i] waiting for CloudFormation stack "eksctl-playground-cluster-nodegroup-ng-759f5e57"
2023-07-20 15:17:20 [i] waiting for CloudFormation stack "eksctl-playground-cluster-nodegroup-ng-759f5e57"
2023-07-20 15:17:20 [i] waiting for the control plane to become ready
2023-07-20 15:17:20 [✓] saved kubeconfig as "/Users/wamba/.kube/config"
2023-07-20 15:17:20 [i] no tasks

```

```

2023-07-20 15:17:20 [✓] saved kubeconfig as "/Users/wamba/.kube/config"
2023-07-20 15:17:20 [i] no tasks
2023-07-20 15:17:20 [✓] all EKS cluster resources for "playground-cluster" have
  been created
2023-07-20 15:17:21 [i] nodegroup "ng-759f5e57" has 3 node(s)
2023-07-20 15:17:21 [i] node "ip-192-168-30-199.us-west-2.compute.internal" is
  ready
2023-07-20 15:17:21 [i] node "ip-192-168-38-0.us-west-2.compute.internal" is re
  ady
2023-07-20 15:17:21 [i] node "ip-192-168-87-176.us-west-2.compute.internal" is
  ready
2023-07-20 15:17:21 [i] waiting for at least 3 node(s) to become ready in "ng-7
  59f5e57"
2023-07-20 15:17:21 [i] nodegroup "ng-759f5e57" has 3 node(s)
2023-07-20 15:17:21 [i] node "ip-192-168-30-199.us-west-2.compute.internal" is
  ready
2023-07-20 15:17:21 [i] node "ip-192-168-38-0.us-west-2.compute.internal" is re
  ady
2023-07-20 15:17:21 [i] node "ip-192-168-87-176.us-west-2.compute.internal" is
  ready
2023-07-20 15:17:22 [i] kubectl command should work with "/Users/wamba/.kube/co
  nfig", try 'kubectl get nodes'
2023-07-20 15:17:22 [✓] EKS cluster "playground-cluster" in "us-west-2" region
  is ready

```

CloudFormation stack complete

CloudFormation

Stacks

StackSets

Exports

Designer

Registry

Public extensions

Activated extensions

Publisher

Spotlight [New](#)

Feedback

CloudFormation > Stacks

Stacks (3)

Refresh

Delete

Update

Stack actions

Create stack

Filter by stack name

Filter status: Active

View nested

< 1 >

Settings

	Stack name	Status	Created time
<input type="radio"/>	eksctl-playground-cluster-nodegroup-ng-759f5e57	CREATE_COMPLETE	2023-07-20 15:14:13 UTC-0400
<input type="radio"/>	eksctl-playground-cluster-cluster	CREATE_COMPLETE	2023-07-20 10:41:33 UTC-0400
<input type="radio"/>	eksctl-playground-cluster-cluster	CREATE_COMPLETE	2023-07-20 09:57:51 UTC-0400

< 1 >

Nodes groups created from cloudformation

New EC2 Experience ×
Tell us what you think

EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Scheduled Instances
Capacity Reservations

Instances (3) [Info](#) ↻ Connect Instance state ▾ Actions ▾ Launch instances ▾

Instance state = running × Clear filters < 1 > ⚙

<input type="checkbox"/>	Name ▾	Instance ID	Instance state ▾	Instance type ▾	Status check
<input type="checkbox"/>	playground-cluster-ng-759f5e57-Node	i-0fc372bbfd0c9c195	Running	m5.large	2/2 checks passed
<input type="checkbox"/>	playground-cluster-ng-759f5e57-Node	i-03b3e019cab66d714	Running	m5.large	2/2 checks passed
<input type="checkbox"/>	playground-cluster-ng-759f5e57-Node	i-0238604bead79df03	Running	m5.large	2/2 checks passed

Select an instance ⚙ ×

Cluster created.

Amazon Elastic Kubernetes Service ×

Clusters New

▼ Related services
Amazon ECR
Container registry for EKS
AWS Batch
Batch computing on EKS

Documentation [🔗](#)
Submit feedback

EKS > Clusters

ⓘ New Kubernetes versions are available for 1 cluster. ×

Clusters (1) [Info](#) ↻ Delete Add cluster ▾

< 1 >

<input type="radio"/>	Cluster name ▲	Status ▾	Kubernetes version ▾	Provider ▾
<input type="radio"/>	playground-cluster	Active	1.25 Update now	EKS

<

Overview

Resources

Compute

Networking

Add-ons

Authentication

Logging

>

Nodes (0)

Info

Q

Filter Nodes by property or value

< 1 >

Node name

▲

Instance type

▼

Node group

▼

Created

▼

Status

▼

No Nodes

This cluster does not have any Nodes, or you don't have permission to view them.

Node groups (1)

Info

Edit

Delete

Add node group

Group name

▲

Desired size

▼

AMI release version

▼

Launch template

ng-759f5e57

3

1.25.11-20230711

eksctl-playground-cluster-nodegroup-ng-759f5e5

Fargate profiles (0)

Info

Edit

Delete

Add Fargate profile

Profile name

▲

Namespaces

Status

▼

No Fargate profiles

Make our deployment manifest file to deploy our nodes for our Apache app: make sur at the container field you give the right name of the docker image that you want to use on your POD as container and tag it

```
geo-Wamba22:~ wamba$ cd kubernetes
geo-Wamba22:kubernetes wamba$ ls
apache-LB.yml      apache-deployment.yml  nginx-pod.yml
geo-Wamba22:kubernetes wamba$ kubectl get pod
No resources found in default namespace.
geo-Wamba22:kubernetes wamba$ kubectl create -f apache-deployment.yml
deployment.apps/apache-deployment created
geo-Wamba22:kubernetes wamba$ kubectl get pod
NAME                                READY   STATUS    RESTARTS   AGE
apache-deployment-558bc587f7-7nrgz  1/1     Running   0           57s
apache-deployment-558bc587f7-bnbw2  1/1     Running   0           57s
apache-deployment-558bc587f7-gm5dz  1/1     Running   0           57s
apache-deployment-558bc587f7-j25s2  1/1     Running   0           57s
apache-deployment-558bc587f7-wr965  1/1     Running   0           57s
geo-Wamba22:kubernetes wamba$
```

Verify if our deployment was created.


```

apache-deployment-558bc587f7-7nrgz 1/1 Running 0 57s
apache-deployment-558bc587f7-bnbw2 1/1 Running 0 57s
apache-deployment-558bc587f7-gm5dz 1/1 Running 0 57s
apache-deployment-558bc587f7-j25s2 1/1 Running 0 57s
apache-deployment-558bc587f7-wr965 1/1 Running 0 57s
geo-Wamba22:kubernetes wamba$ kubectl get deployment
NAME READY UP-TO-DATE AVAILABLE AGE
apache-deployment 5/5 5 5 4m8s
geo-Wamba22:kubernetes wamba$ kubectl describe deployment apache-deployment
Name: apache-deployment
Namespace: default
CreationTimestamp: Thu, 20 Jul 2023 20:43:22 -0400
Labels: app=apache
Annotations: deployment.kubernetes.io/revision: 1
Selector: app=apache
Replicas: 5 desired | 5 updated | 5 total | 5 available | 0 unavailable
StrategyType: RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels: app=apache
  Containers:
    apache-container:
      Image: emireine/myfirst_image:v4

```

Describe our deployment.

```

MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels: app=apache
  Containers:
    apache-container:
      Image: emireine/myfirst_image:v4
      Port: 80/TCP
      Host Port: 0/TCP
      Environment: <none>
      Mounts: <none>
      Volumes: <none>
Conditions:
  Type Status Reason
  ----
  Available True MinimumReplicasAvailable
  Progressing True NewReplicaSetAvailable
OldReplicaSets: <none>
NewReplicaSet: apache-deployment-558bc587f7 (5/5 replicas created)
Events:
  Type Reason Age From Message
  ----
  Normal ScalingReplicaSet 6m25s deployment-controller Scaled up replica set
  apache-deployment-558bc587f7 to 5
geo-Wamba22:kubernetes wamba$

```


Configure your loadbalancer manifestfile and create it

```
geo-Wamba22:kubernetes wamba$ kubectl create -f apache-LB.yml
service/apache-lb-service created
geo-Wamba22:kubernetes wamba$ kubectl get service
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP
PORT(S)                            AGE
apache-lb-service                  LoadBalancer        10.100.68.205    a6c010f7a1be74eafabeee702c741
1cc-276252667.us-west-2.elb.amazonaws.com  80:30544/TCP        5m18s
kubernetes                         ClusterIP            10.100.0.1       <none>
443/TCP                            10h
geo-Wamba22:kubernetes wamba$
```

Load balancer created.

The screenshot displays the AWS Management Console interface for an Elastic Load Balancing (ELB) instance. The left sidebar shows the navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Instances, Images, and Elastic Block Store. The main content area shows the configuration details for the load balancer `a6c010f7a1be74eafabeee702c7411cc`.

Basic Configuration

Property	Value
Name	a6c010f7a1be74eafabeee702c7411cc
* DNS name	a6c010f7a1be74eafabeee702c7411cc-276252667.us-west-2.elb.amazonaws.com (A Record)
Type	Classic (Migrate Now)
Scheme	internet-facing
Availability Zones	subnet-022f08a69843b038d - us-west-2b, subnet-02fcd8a232f376655 - us-west-2a, subnet-0410f0d04fcaa56c0 - us-west-2c
Creation time	July 20, 2023 at 9:20:11 PM UTC-4
Hosted zone	Z1H1FL5HABSF5
Status	3 of 3 instances in service
VPC	vpc-0ce9d9fc697b4d99c

Port Configuration

Port Configuration: 80 (TCP) forwarding to 30544 (TCP)

Stickiness options not available for TCP protocols

Security

Source Security Group: sg-056bd51f72a7026f4, k8s-elb-a6c010f7a1be74eafabeee702c7411cc

• Security group for Kubernetes ELB a6c010f7a1be74eafabeee702c7411cc (default/apache-lb-service)

[Edit security groups](#)

Then use the load balancer DNS to verify if your website is running.

