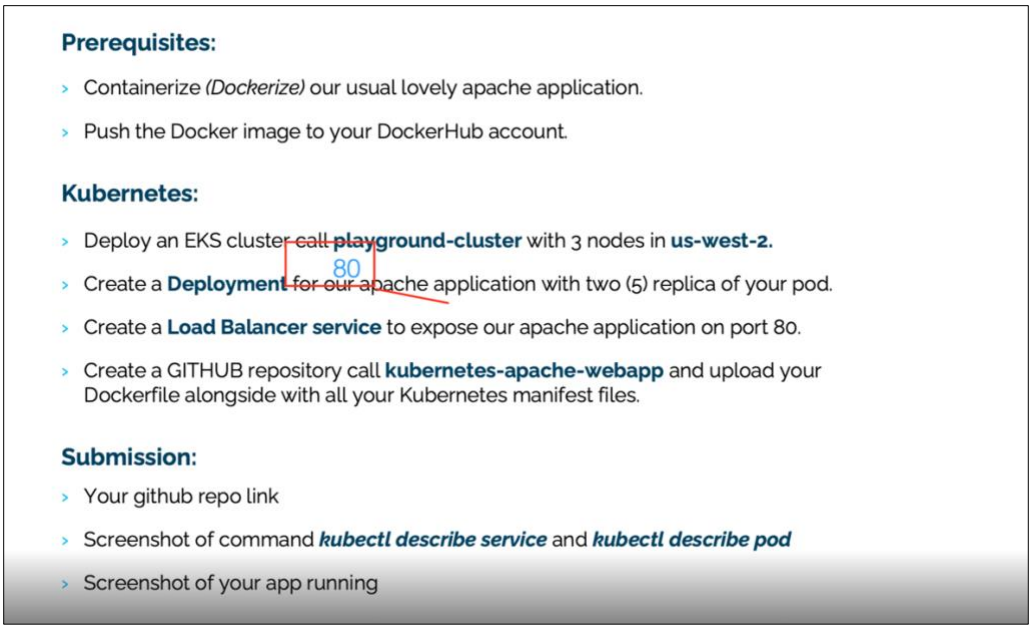
Kurbernetes project:

Project instruction



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 "$(cat kubectl.sha256)  kubectl" | shasum -a 256 --che
ck
kubectl: OK
geo-Wamba22:~ wamba$ 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 ou
tput from kubectl version --short.  Use --output=yaml|json to get the full versi
on.
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

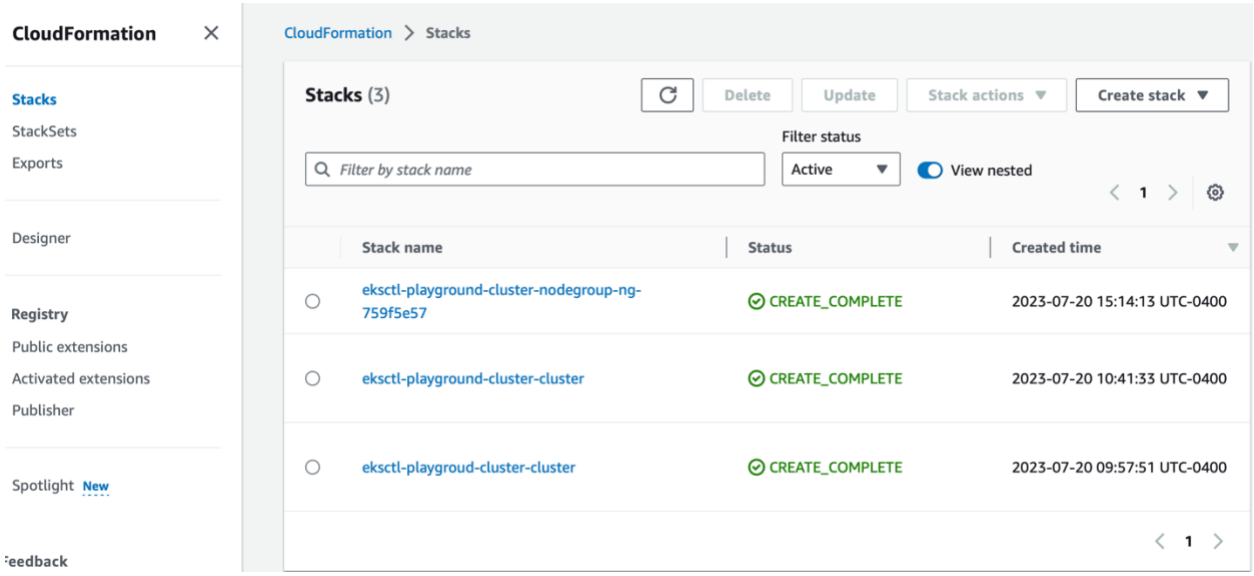
```
geo-Wamba22:~ wamba$ eksctl create cluster --name playground-cluster --region us
-west-2 --nodes-min 3 --nodes-max 5 --nodes 3
2023-07-20 10:41:32 [ℹ]  eksctl version 0.147.0
2023-07-20 10:41:32 [ℹ]  using region us-west-2
2023-07-20 10:41:33 [ℹ]  setting availability zones to [us-west-2a us-west-2c 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 cluste
r itself and the initial managed nodegroup
2023-07-20 10:41:33 [ℹ]  if you encounter any issues, check CloudFormation conso
le 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 {pub
licAccess=true, privateAccess=false} for cluster "playground-cluster" in "us-wes
t-2"
2023-07-20 10:41:33 [ℹ]  CloudWatch logging will not be enabled for cluster "pla
yground-cluster" in "us-west-2"
2023-07-20 10:41:33 [ℹ]  you can enable it with 'eksctl utils update-cluster-log
ging --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-clust
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-clu
ster-cluster"
2023-07-20 10:42:34 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
```

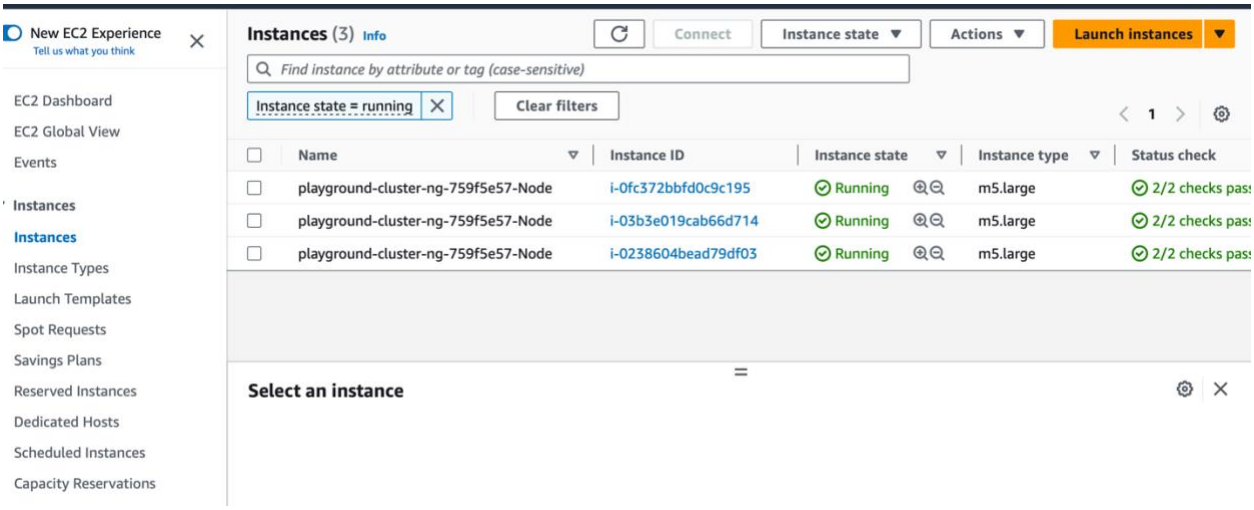
```
er"
2023-07-20 10:41:33 [ℹ]  deploying stack "eksctl-playground-cluster-cluster"
2023-07-20 10:42:03 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-cluster"
2023-07-20 10:42:34 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-cluster"
2023-07-20 10:43:34 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-cluster"
2023-07-20 11:38:06 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-cluster"
2023-07-20 15:14:12 [ℹ]  building managed nodegroup stack "eksctl-playground-clu
ster-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-clu
ster-nodegroup-ng-759f5e57"
2023-07-20 15:14:43 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-nodegroup-ng-759f5e57"
2023-07-20 15:15:24 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-nodegroup-ng-759f5e57"
2023-07-20 15:17:20 [ℹ]  waiting for CloudFormation stack "eksctl-playground-clu
ster-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
```

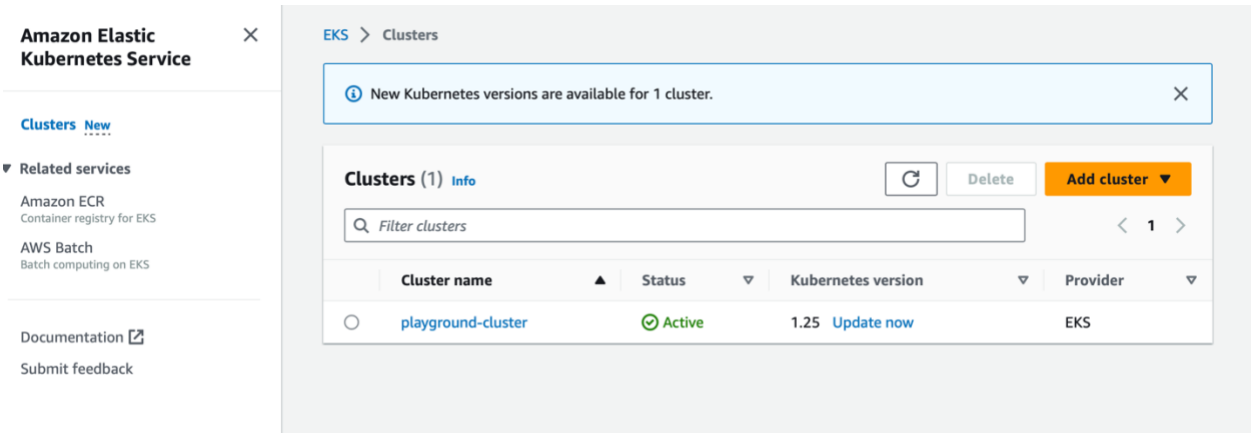
CloudFormation stack complete

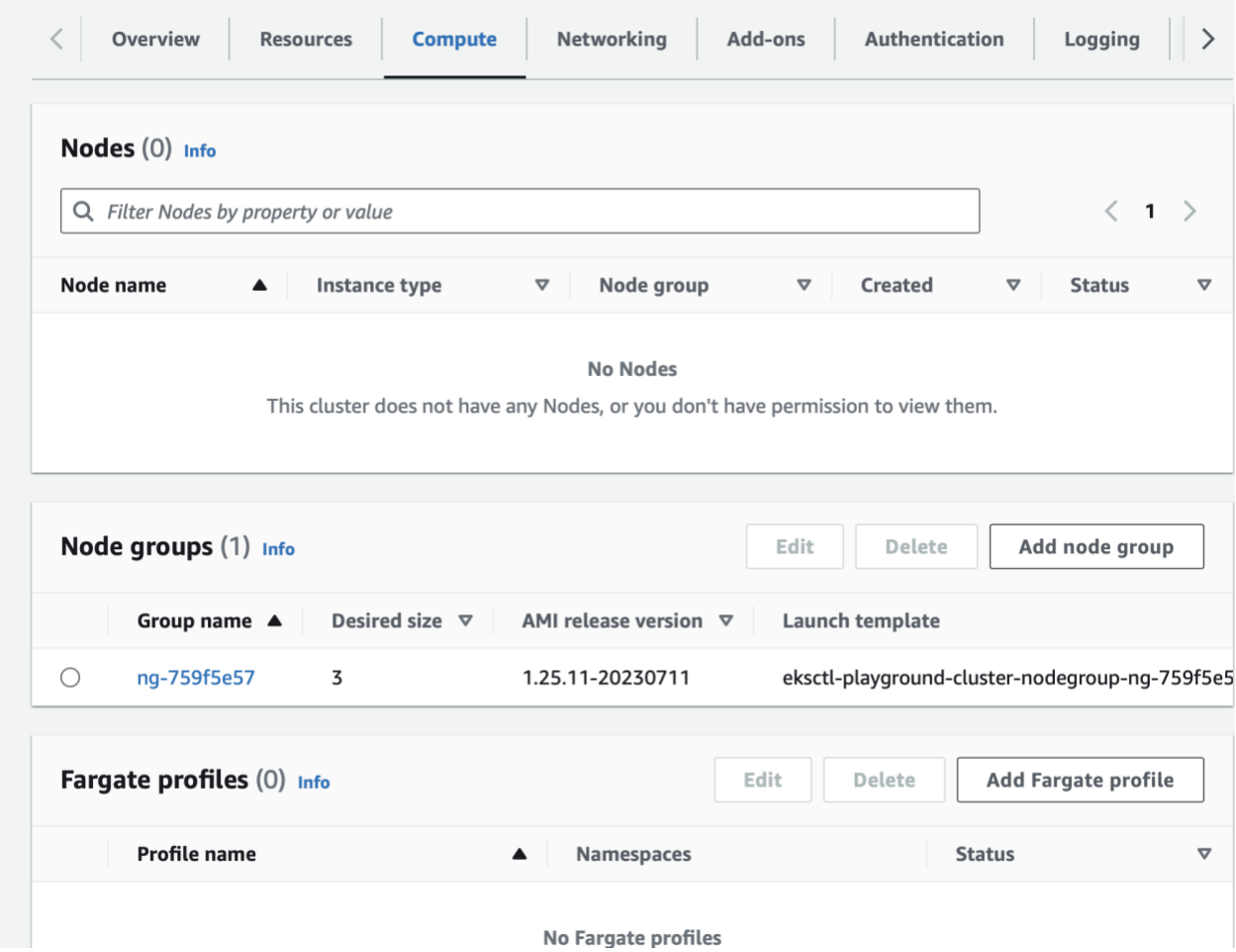


Nodes groups created from cloudformation



Cluster created.





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 kubenetes
geo-Wamba22:kubenetes wamba$ ls
apache-LB.yml           apache-deployment.yml   nginx-pod.yml
geo-Wamba22:kubenetes wamba$ kubectl get pod
No resources found in default namespace.
geo-Wamba22:kubenetes wamba$ kubectl create -f apache-deployment.yml
deployment.apps/apache-deployment created
geo-Wamba22:kubenetes 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:kubenetes 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:kubenetes wamba$ kubectl get deployment
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
apache-deployment   5/5     5            5           4m8s
geo-Wamba22:kubenetes 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 unavai
lable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=apache
  Containers:
   apache-container:
    Image:        emireine/myfirst_image:v4
```

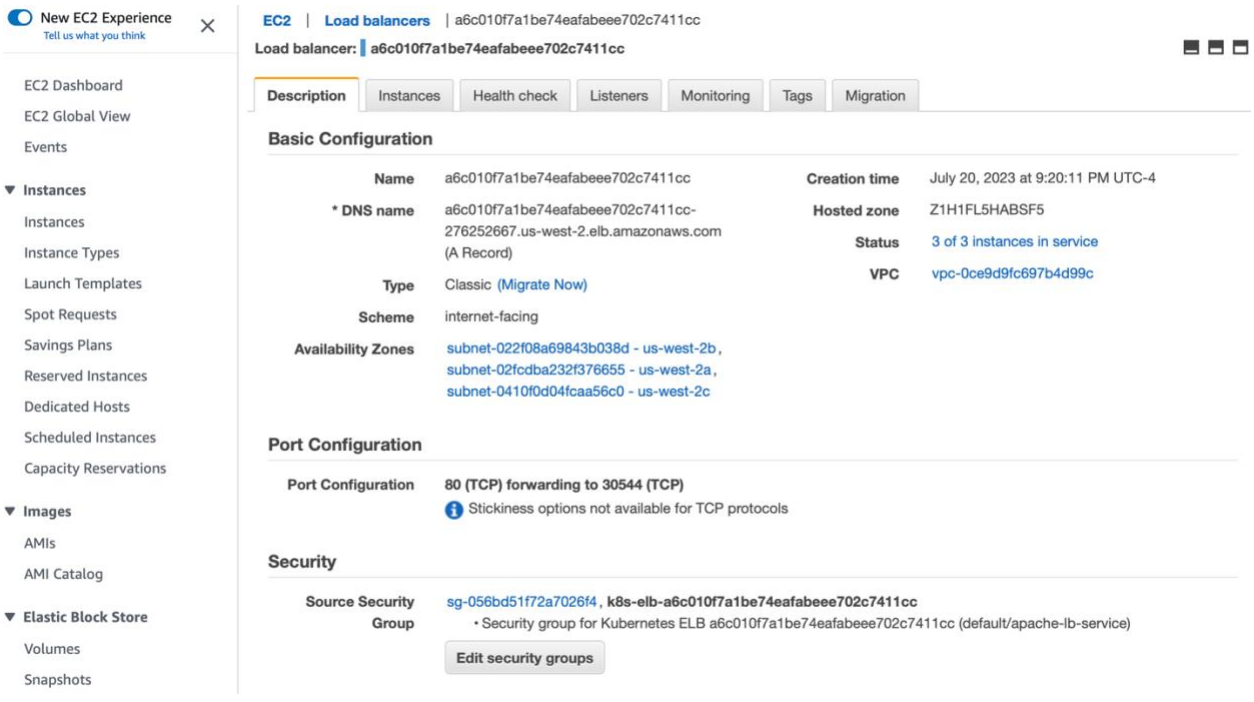
Describe our deployment.

```
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:kubenetes wamba$
```

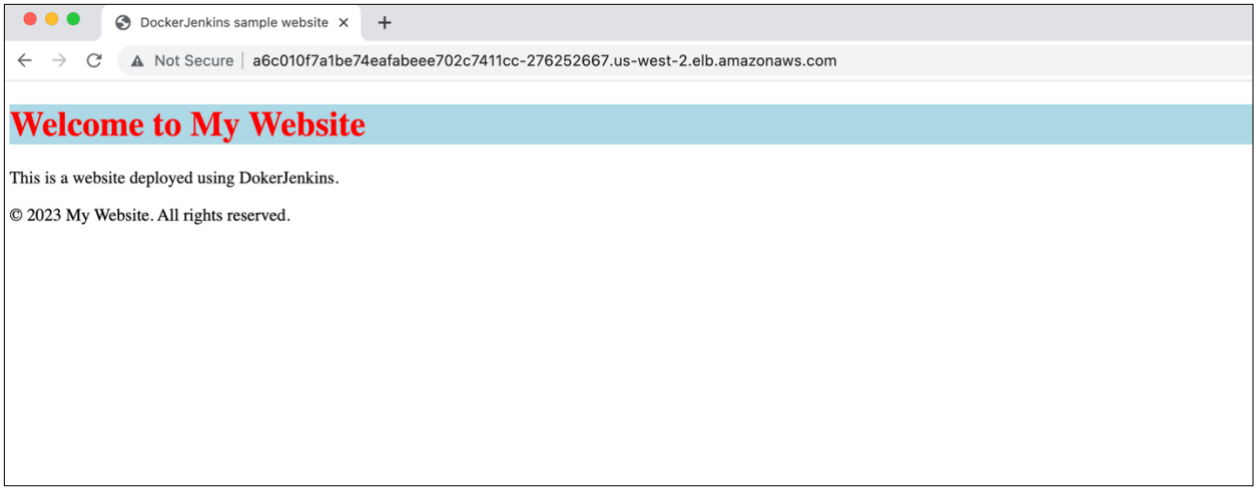
Configure your loadbalancer manifestfile and create it

```
geo-Wamba22:kubenetes wamba$ kubectl create -f apache-LB.yml
service/apache-lb-service created
geo-Wamba22:kubenetes 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:kubenetes wamba$
```

Load balancer created.



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


DockerJenkins sample website
Not Secure
a6c010f7a1be74eafabeee702c7411cc-276252667.us-west-2.elb.amazonaws.com
Welcome to My Website
This is a website deployed using DokerJenkins.
© 2023 My Website. All rights reserved.