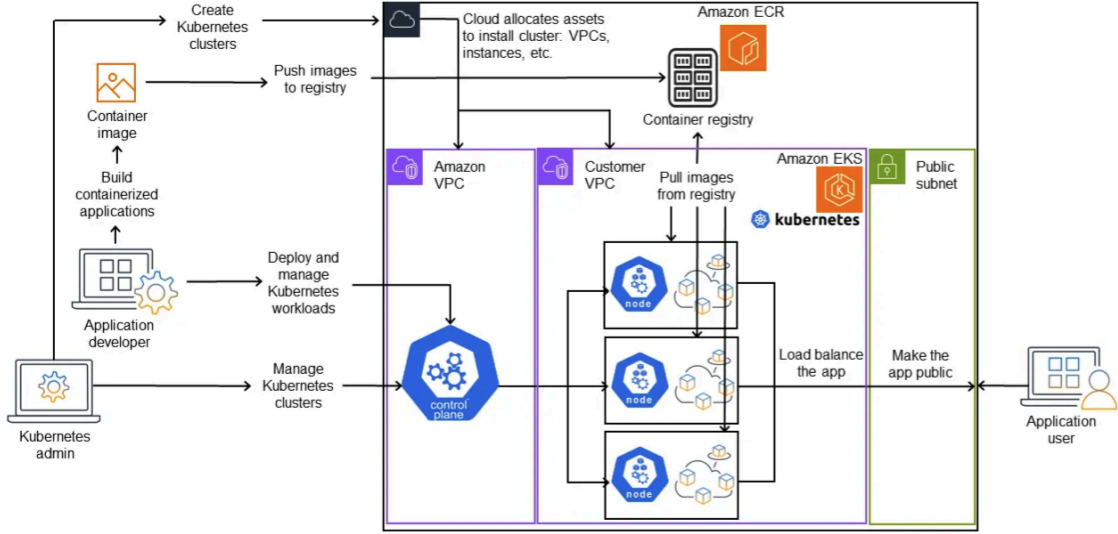


# EKS cluster kurulumu ve örnek deployment

## A Kubernetes cluster in action



1. Create EKS Cluster:

Amazon Elastic Kubernetes Service (EKS), Kubernetes'i AWS üzerinde çalıştırmayı kolaylaştıran yönetilen bir Kubernetes hizmetidir.

## Ön Gereksinimler

- **AWS Hesabı:** EKS cluster oluşturmak için bir AWS hesabınız olmalı.
- **AWS CLI:** AWS CLI'yi kurun ve yapılandırın.
- **kubectl:** Kubernetes cluster'ları ile etkileşim için kubectl'yi kurun.
- **eksctl:** EKS cluster'larını oluşturmak için eksctl'yi kurun.
- **IAM Yetkileri:** EKS cluster'ı oluşturmak için gerekli IAM yetkilerine sahip olun.

## EKS Cluster Kurulumu

EKS Cluster oluşturun:

```
eksctl create cluster \
  --name my-eks-cluster \
  --region us-east-1 \
  --nodegroup-name standard-workers \
  --node-type t3.medium \
  --nodes 2
```

Kubeconfig güncelleyin:

```
aws eks --region us-east-1 update-kubeconfig --name my-eks-cluster
```

## EKS cluster için bir IAM OIDC sağlayıcısı oluşturun

```
eksctl utils associate-iam-oidc-provider \
  --region us-east-1 \
  --cluster my-eks-cluster \
  --approve
```

## AWS load balancer için IAM politikasını indirin:

```
curl -o iam-policy.json
https://raw.githubusercontent.com/kubernetes-sigs/aws-load-balancer-controller/v2.2.1/docs/install/iam_policy.json
```

## AWSLoadBalancerControllerIAMPolicy adlı bir IAM politikası oluşturun

```
aws iam create-policy \
  --policy-name AWSLoadBalancerControllerIAMPolicy \
  --policy-document file://iam-policy.json
```

Döndürülen politika ARN'sini not edin.

```
(base) eks ➜ aws iam create-policy \
  --policy-name AWSLoadBalancerControllerIAMPolicy \
  --policy-document file://iam-policy.json

{
  "Policy": {
    "PolicyName": "AWSLoadBalancerControllerIAMPolicy",
    "PolicyId": "ANPA54T5LD3FIY25UFT43",
    "Arn": "arn:aws:iam::954819354314:policy/AWSLoadBalancerControllerIAMPolicy",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "PermissionsBoundaryUsageCount": 0,
    "IsAttachable": true,
    "CreateDate": "2024-09-13T06:45:46+00:00",
    "UpdateDate": "2024-09-13T06:45:46+00:00"
  }
}
```

**Önceki adımdaki ARN'yi kullanarak AWS Yük Dengeleyici Denetleyicisi için bir IAM rolü ve ServiceAccount oluşturun:**

```
eksctl create iamserviceaccount \
  --cluster=my-eks-cluster \
  --namespace=kube-system \
  --name=aws-load-balancer-controller \

--attach-policy-arn=arn:aws:iam::954819354314:policyAWSLoadBalancer
ControllerIAMPolicy \
  --override-existing-serviceaccounts \
  --approve
```

## Örnek Uygulama Dağıtımı

Deployment manifesti oluşturun (nginx-deployment.yaml):

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:latest
          ports:
            - containerPort: 80
```

Deployment'u EKS cluster'a yükleyin

```
kubectl apply -f nginx-deployment.yaml
```

Deployment'u görüntüleyin:

```
kubectl get deployments
```

## Servis Dağıtımı

Servis manifesti oluşturun (nginx-service.yaml):

```
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: LoadBalancer
```

Servisi deploy edin:

```
kubectl apply -f nginx-service.yaml
```

Servisi görüntüleyin:

```
kubectl get services
```

```
(base) eks ➤ kubectl get services
NAME                TYPE          CLUSTER-IP    EXTERNAL-IP                                PORT(S)          AGE
kubernetes          ClusterIP     10.100.0.1    <none>                                     443/TCP          21m
nginx-service       LoadBalancer 10.100.26.172  a32abc9bdfef648569fface9dd59ada0-2085278652.us-east-1.elb.amazonaws.com  80:32727/TCP    12m
```

oluşturulan servise erişmek için External IP altındaki load balancer adresini browserunuza kopyalayın.



## Uygulama Ölçekleme

Deployment'u manuel olarak ölçeklendirin:

```
kubectl scale deployment/nginx-deployment --replicas=3
```

Ölçeklendirmeyi kontrol edin

```
kubectl get deployments
kubectl get pods
```

Oluşturulan eks cluster silmek için

```
eksctl delete cluster --name my-cluster --region us-east-1
```