

```
1 # Cloudformation Infrastructure as Code to create EKS cluster
2 AWSTemplateFormatVersion: 2010-09-09
3 Parameters:
4   WorkerInstanceType:
5     Type: String
6     Default: t2.micro
7     Description: "Worker node instance type"
8   VpcId:
9     Type: AWS::EC2::VPC::Id
10    Description: "ID of the VPC to deploy the EKS cluster into"
11  SubnetIds:
12    Type: List<AWS::EC2::Subnet::Id>
13    Description: "List of subnet IDs in different AZs to deploy the EKS
cluster into"
14
15 Resources:
16   # creates eks cluster
17   MyCluster:
18     Type: AWS::EKS::Cluster
19     Properties:
20       Name: !Sub ${AWS::StackName}-cluster
21       Version: "1.26"
22       RoleArn: !GetAtt MyClusterRole.Arn
23       ResourcesVpcConfig:
24         SecurityGroupIds:
25           - !GetAtt MyClusterSecurityGroup.GroupId
26         SubnetIds: !Ref SubnetIds
27       Tags:
28         - Key: Name
29           Value: !Sub ${AWS::StackName}-cluster
30
31   # creates security group for eks cluster
32   MyClusterSecurityGroup:
33     Type: AWS::EC2::SecurityGroup
34     Properties:
35       GroupDescription: "My Kubernetes cluster security group"
36       VpcId: !Ref VpcId
37       SecurityGroupIngress:
38         - IpProtocol: tcp
39           FromPort: 0
40           ToPort: 65535
41           CidrIp: "0.0.0.0/0"
42       SecurityGroupEgress:
43         - IpProtocol: tcp
44           FromPort: 0
45           ToPort: 65535
46           CidrIp: "0.0.0.0/0"
47       Tags:
48         - Key: Name
49           Value: !Sub ${AWS::StackName}-cluster-sg
50
51   # creates role for eks cluster
52   MyClusterRole:
53     Type: AWS::IAM::Role
54     Properties:
55       AssumeRolePolicyDocument:
56         Version: "2012-10-17"
57         Statement:
58           - Effect: Allow
```

```

59     Principal:
60       Service: eks.amazonaws.com
61     Action: sts:AssumeRole
62   ManagedPolicyArns:
63     - arn:aws:iam::aws:policy/AmazonEKSClusterPolicy
64     - arn:aws:iam::aws:policy/AmazonEKSServicePolicy
65   Tags:
66     - Key: Name
67       Value: !Sub ${AWS::StackName}-cluster-role
68
69 # creates node group for eks cluster
70 MyNodeGroup:
71   Type: AWS::EKS::Nodegroup
72   Properties:
73     ClusterName: !Ref MyCluster
74     NodeRole: !GetAtt MyNodeRole.Arn
75     Subnets: !Ref SubnetIds
76     InstanceTypes:
77       - !Ref WorkerInstanceType
78     ScalingConfig:
79       DesiredSize: 1
80       MaxSize: 2
81       MinSize: 1
82     AmiType: AL2_x86_64
83
84 # creates role for node group
85 MyNodeRole:
86   Type: AWS::IAM::Role
87   Properties:
88     AssumeRolePolicyDocument:
89       Version: "2012-10-17"
90       Statement:
91         - Effect: Allow
92           Principal:
93             Service: ec2.amazonaws.com
94           Action: sts:AssumeRole
95     ManagedPolicyArns:
96       - arn:aws:iam::aws:policy/AmazonEKSWorkerNodePolicy
97       - arn:aws:iam::aws:policy/AmazonEC2ContainerRegistryReadOnly
98       - arn:aws:iam::aws:policy/AmazonEKS_CNI_Policy
99     Tags:
100       - Key: Name
101         Value: !Sub ${AWS::StackName}-node-role
102
103 # create ecr repository
104 MyECR:
105   Type: AWS::ECR::Repository
106   Properties:
107     RepositoryName: !Sub ${AWS::StackName}-ecr
108     RepositoryPolicyText:
109       Version: "2008-10-17"
110       Statement:
111         - Sid: AllowPushPull
112           Effect: Allow
113           Principal:
114             AWS: !GetAtt MyNodeRole.Arn
115           Action:
116             - ecr:GetDownloadUrlForLayer
117             - ecr:BatchGetImage
118             - ecr:BatchCheckLayerAvailability

```

```
119         - ecr:PutImage
120         - ecr:InitiateLayerUpload
121         - ecr:UploadLayerPart
122         - ecr:CompleteLayerUpload
123
124     # roles to let admin user see the pod, service details etc in console
125
126 Outputs:
127     ECRUrl:
128         Description: "ECR URL"
129         Value: !GetAtt MyECR.RepositoryUri
130
```