Bring Up and Configure an EKS Cluster

This module describes the steps to bring up and configure an EKS Cluster.

Bring up an EKS Cluster

Create a Cluster Role

You must create a cluster role that has the permission to access EKS resources.

Copy the following contents to a file named `eks-cluster-role-trust-policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "eks.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Create an AWS Identity and Access Management (IAM) role with this policy document.

```
aws iam create-role \
  --role-name xrd-eks-cluster-role \
  --assume-role-policy-document "file://eks-cluster-role-trust-policy.json"
```

Make a note of the role Amazon Resource Names (ARN), `<cluster-role-arn>`.

Attach the EKS managed IAM policy to the role.

```
aws iam attach-role-policy \
  --policy-arn arn:aws:iam::aws:policy/AmazonEKSClusterPolicy \
  --role-name xrd-eks-cluster-role
```

Create a Worker Node Role

You must create a role for the EKS worker nodes to connect to the EKS cluster.

Copy the following contents to the file named `eks-node-role-trust-policy.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Note

- 2012-10-17 is the latest trust policy version.
- The **Statement** allows any EC2 resource to assume the worker node role and therefore obtain the associated permissions, which enable a node to join to an EKS cluster.

Create the IAM role using the following command:

```
aws iam create-role \
  --role-name xrd-eks-node-role \
  --assume-role-policy-document "file://eks-node-role-trust-policy.json"
```

Make a note of the node role ARN, `<node-role-arn>`.

Attach the required IAM policies to the role. For example,

```
aws iam attach-role-policy \
  --policy-arn arn:aws:iam::aws:policy/AmazonEKSWorkerNodePolicy \
  --role-name xrd-eks-node-role
aws iam attach-role-policy \
  --policy-arn arn:aws:iam::aws:policy/AmazonEC2ContainerRegistryReadOnly \
  --role-name xrd-eks-node-role
aws iam attach-role-policy \
  --policy-arn arn:aws:iam::aws:policy/AmazonEKS_CNI_Policy \
  --role-name xrd-eks-node-role
```

Create Worker Node Profile

An instance profile is required to apply the Worker node role to EC2 instances.

To create an instance profile, use the following command:

```
aws iam create-instance-profile --instance-profile-name xrd-eks-node-profile
```

Make a note of the profile ARN, `<node-profile-arn>`.

Add the role to the new profile using the following command:

```
aws iam add-role-to-instance-profile \
  --instance-profile-name xrd-eks-node-profile \
  --role-name xrd-eks-node-role
```

Create EKS Cluster

Use the following command to create an EKS cluster:

```
aws eks create-cluster \
  --name xrd-cluster \
  --role-arn <cluster-role-arn> \
  --resources-vpc-config
"subnetIds=<private-subnet-1>,<private-subnet-2>,securityGroupIds=<sg-id>,endpointPublicAccess=true,endpointPrivateAccess=true"
 \
  --kubernetes-version <k8s-version>
```

This command execution completes quickly, but the Control plane takes around 20-30 minutes to come up completely in AWS. If you want the AWS CLI tool to monitor and wait for the cluster to become active, use the following command:

```
aws eks wait cluster-active --name xrd-cluster
```

To check the status of the cluster manually, run the following command:

```
aws eks describe-cluster --name xrd-cluster
```

In the output, the "status" will be `CREATING` until the cluster comes up completely, and then the status changes to `ACTIVE`.

Note This sample configuration sets up an EKS cluster with both public and private endpoints. The EKS cluster control plane can now receive traffic from the internet.

You can restrict the IP range allowed to access the public endpoint using the `publicAccessCirds` configuration item, or you can remove the public endpoint entirely by setting the `endpointPublicAccess` item to `false`.

For more details, see Amazon EKS cluster Endpoint Access Control.

Configure EKS Cluster

You must configure the EKS cluster after it becomes `ACTIVE`. The configuration steps are as follows:

1. First, generate `kubectl` configuration for the cluster by running the following command:

   ```
   aws eks update-kubeconfig --name xrd-cluster
   ```

2. To verify the configuration, run the following command:

   ```
   kubectl get svc
   ```

   The output must be similar to the following:

   ```
   NAME             TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
   svc/kubernetes   ClusterIP   172.20.0.1   <none>        443/TCP   1m
   ```

EKS Cluster configuration includes:

- Role authentication configuration—Allows worker nodes to join the cluster.
- AWS cluster networking configuration—Prevents AWS from using more than one Elastic Network Adapter (ENA) for Pod IPs. This configuration ensures that maximum number of Elastic Network Interfaces (ENIs) are available for use in XRd.

Role Authentication Configuration

1. Copy the following contents to a file named `aws-auth-cm.yaml`:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: aws-auth
  namespace: kube-system
data:
  mapRoles: |
    - rolearn: <node-role-arn>
      username: system:node:{{EC2PrivateDNSName}}
      groups:
        - system:bootstrappers
        - system:nodes
```

2. Replace `<node-role-arn>` with the node role ARN created earlier.
3. Apply this configuration to the cluster using the following command:

```
kubectl apply -f aws-auth-cm.yaml
```

4. Then, apply the networking configuration to the cluster using the following command:

```
kubectl set env ds aws-node -n kube-system MAX_ENI=1
```

EBS CSI Driver

EKS clusters at version 1.23 or later require installation of the Amazon Elastic Block Store (Amazon EBS) Container Storage Interface (CSI) driver to manage Amazon EBS volumes for persistent volumes in Cisco IOS XRd. Configuration persistence over container restart necessitates this installation.

Install the EBS CSI driver as an EKS addon, following the recommended approach outlined in the AWS documentation.

Note The addon state will be 'Degraded' until worker nodes are brought up in the cluster, as a minimum number of EBS CSI containers are expected to be running in the cluster.

AWS Cluster Networking Configuration

Note This configuration is required only during XRd Control Plane deployment.

When deploying XRd Control Plane in the cluster, you need the Multus Container Network Interface (CNI) plugin to configure container networking.

To install Multus, run the following command:

```
kubectl apply -f
https://raw.githubusercontent.com/aws/amazon-vpc-cni-k8s/master/config/multus/v4.0.2-eksbuild.1/multus-daemonset-thick.yml
```