Cloudera AI

AWS Requirements

Date published: 2020-07-16
Date modified: 2025-10-31

https://docs.cloudera.com/

Legal Notice

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

AWS Account Prerequisites for Cloudera AI Workbenches

To successfully provision an Cloudera AI Workbench, there are many prerequisites that you must ensure are met. Carefully go through this section step by step.

1. Review the AWS Account Prerequisites for Cloudera

   Verify that the AWS account that you would like to use for Cloudera has the required resources and that you have the permissions required to manage these resources.

   Instructions: AWS Account Requirements
2. Review the Cloudera AI-Specific AWS Resource Requirements

   Provisioning an Cloudera AI Workbench will require access to the following AWS resources. Make sure your AWS account has access to these resources.

   - AWS Services used by Cloudera AI

     a. Compute - Amazon Elastic Kubernetes Service (EKS)
     b. Load Balancing - Amazon Network Load Balancer (NLB)
     c. Key Management - AWS Key Management Service (KMS)
     d. DNS - Amazon Route 53, hosted by Cloudera
     e. Persistent Storage - Amazon Elastic Block Store (EBS)
     f. Project File Storage - Amazon Elastic File System (EFS) for project file storage
     g. Command Line Interface - AWS Command Line Interface (CLI).
     h. Security Token Service - AWS Security Token Service (STS)
   - VPC Requirements - You can either use an existing VPC or allow Cloudera to create one for you.

     - Option 1. Using your own VPC

       - Recommended requirements: Divide the address space according to the following recommended sizes:

         - 3 x /19 private subnets. Each subnet should be created in a separate Availability Zone for the EKS worker nodes.
         - 3 x /24 public subnets. These should also be created in three separate Availability Zones, using the same zones as the private subnets.
         - Ensure the CIDR block for the subnets is sized appropriately.
         - You must enable Amazon DNS with the VPC. Corporate DNS is not supported. For guidelines on how verify your DNS settings, refer to sections 1-3 in AWS environment requirements checklist for the Data Warehouse service.

         Note: There is no way to increase the subnet size without recreating the environment and VPC.

       Private subnets should have routable IPs over your internal VPN. If IPs are not routable, private Cloudera AI endpoints will need to be accessed via a SOCKS proxy. Cloudera recommends creating routable IPs by setting up VPN connections between networks, and not using any public load balancers.

         If a fully-private network configuration is not feasible, use of a SOCKS proxy to access Cloudera AI is possible, but is not recommended.

         Tag the VPC and the subnets as shared so that Kubernetes can find them. For load balancers to be able to choose the subnets correctly, you are also required to tag private subnets with the kubernetes.io/role/internal-elb:1 tag, and public subnets with the kubernetes.io/role/elb:1 tag.

     - Option 2. Cloudera creates a new VPC

       If you choose to allow Cloudera to create a new VPC, three subnets will be created automatically. One subnet is created for each availability zone assuming three AZs per region; If a region has two AZs instead of three, then still three subnets are created, two in the same AZ.

       You will be asked to specify a valid CIDR in IPv4 range that will be used to define the range of private IPs for EC2 instances provisioned into these subnets.
     - Related AWS documentation: Amazon EKS - Cluster VPC Considerations, Creating a VPC for your Amazon EKS Cluster
   - Ports Requirements

     HTTPS access to Cloudera AI Workbenches is available over port 443 for the following cases:

     - internal only - should be accessible from your organization's network, but not the public internet
     - internet facing - should be accessible from the public internet as well as your internal organization's network

     This is in addition to the ports requirements noted here for Cloudera's default security group: Management Console - Security groups.
   - Firewall requirements

     Installations must comply with firewall requirements set by cloud providers at all times. Ensure that ports required by the provider are not closed. For example, Kubernetes services have requirements documented in *Amazon EKS security group considerations*.

     Also, for information on repositories that must be accessible to set up Cloudera AI Workbenches, see *Outbound network access destinations for AWS*.

3. **Review the default AWS service limits and your current AWS account limits**

   By default, AWS imposes certain default limits for AWS services, per-user account. Make sure you review your account's current usage status and resource limits before you start provisioning additional resources for Cloudera and Cloudera AI.

   For example, depending on your AWS account, you might only be allowed to provision a certain number of CPU instances, or you might not have default access to GPU instances at all. Make sure to review your AWS service limits before your proceed.

   Related AWS documentation: AWS Service Limits, Amazon EC2 Resource Limits.
4. **Review supported AWS regions**

   Cloudera supports the following AWS regions: Supported AWS regions. However, the Cloudera AI service requires AWS Elastic Kubernetes Service (EKS). Make sure you select a region that includes EKS.

   Related AWS documentation: Region Table (AWS Documentation).
5. **Set up an AWS Cloud Credential**

   Create a role-based AWS credential that allows Cloudera to authenticate with your AWS account and has authorization to provision AWS resources on your behalf. Role-based authentication uses an IAM role with an attached IAM policy that has the minimum permissions required to use Cloudera.

   Once you have created this IAM policy, register it in Cloudera as a cloud credential. Then, reference this credential when you are registering the environment in the next step.

   Instructions: Introduction to the role-based provisioning credential for AWS

6. Register an AWS Environment

   A Cloudera User with the role of Power User must register an environment for their organization. An environment determines the specific cloud provider region and virtual network in which resources can be provisioned, and includes the credential that should be used to access the cloud provider account.

   Instructions: Register an AWS Environment
7. Ensure private subnets have outbound internet connectivity

   Also, ensure that your private subnets have outbound internet connectivity. Check the route tables of private subnets to verify the internet routing. Worker nodes must be able to download Docker images for Kubernetes, billing and metering information, and to perform API server registration.
8. Ensure the Amazon Security Token Service (STS) is activated

   To successfully activate an environment in the Data Warehouse service, you must ensure the Amazon STS is activated in your AWS VPC:

   a. In the AWS Management Console home page, select IAM under Security, Identity, & Compliance.
   b. In the Identity and Access Management (IAM) dashboard, select Account settings in the left navigation menu.
   c. On the Account settings page, scroll down to the section for Security Token Service (STS).
   d. In the Endpoints section, locate the region in which your environment is located and make sure that the STS service is activated.
9. Cloudera AI Role Requirements

   There are two Cloudera user roles associated with the Cloudera AI service: MLAdmin and MLUser. Any Cloudera user with the EnvironmentAdmin (or higher) access level must assign these roles to users who require access to the Cloudera AI service within their environment.

   Furthermore, if you want to allow users to log in to provisioned Cloudera AI Workbenches and run workloads on them, this will need to be configured separately.

   Instructions: Configuring User Access to Cloudera AI Workbenches

Related Information

Amazon EKS security group considerations

Outbound network access destinations for AWS

Limitations on AWS

This section lists some resource limits that Cloudera AI and AWS impose on workloads running in Cloudera AI Workbenches.

- Certificate creation (for TLS) uses LetsEncrypt which is limited to 2000 certs/week. As such a single tenant in Cloudera can create a maximum of 2000 Cloudera AI Workbenches per week.
- Cloudera AI imposes a limit (50) on the number of pods a user can create at any point within a specific Cloudera AI Workbench. This limit is not configurable.
- Cloudera AI allows you to provision a maximum of 100 compute nodes per Cloudera AI Workbench. This does not include any additional infrastructure nodes Cloudera AI might need to provision to run the service.
- Amazon EKS imposes a limit on the number of pods you can run simultaneously on a node. This limit varies depending on your instance type. For details, see ENI Max Pods.
- Cloudera AI creates one Amazon Elastic File System per Cloudera AI Workbench. The number of Elastic File Systems in a region is limited to 1000 per account. Therefore, the number of Cloudera AI Workbenches in a region is limited to 1000 at any given time for a given account.

Related Information

Supported AWS regions

Network Planning for Cloudera AI on AWS

Before attempting to deploy your AWS virtual network and set up your AWS Environment and Cloudera AI Workbenches, you should plan the network.

Setting up Cloudera AI for AWS is comprehensively covered in the Getting Started guide. See *AWS account requirements* for more information.

Related Information

AWS account requirements

AWS IAM restricted roles and policies for compute and Cloudera AI

AWS IAM write permissions are used by the Cloudera AI compute infrastructure to create and delete roles and instance profiles.

Some customers may not be willing to provide IAM write permissions in the role’s policy. Instead, customers can set up static pre-created roles and instance profiles defined and used by the Cloudera AI compute infrastructure to provision clusters.

Note:

- The compute infrastructure is only able to use the pre-created roles and instance profile if the entitlement LIFTIE_USE_PRECREATED_IAM_RESOURCES is set for the tenant in use.
- The pre-created roles and instance profiles should continue to exist for the lifetime of the cluster.

The two main tasks are:

1. Create roles and an instance profile.
2. Create restricted IAM policies for use by the compute infrastructure.

After these two tasks are completed, you can create the cross-account credential, if needed.

See the following topics for the procedures for creating the roles and policies.

Create IAM roles and instance profile pair

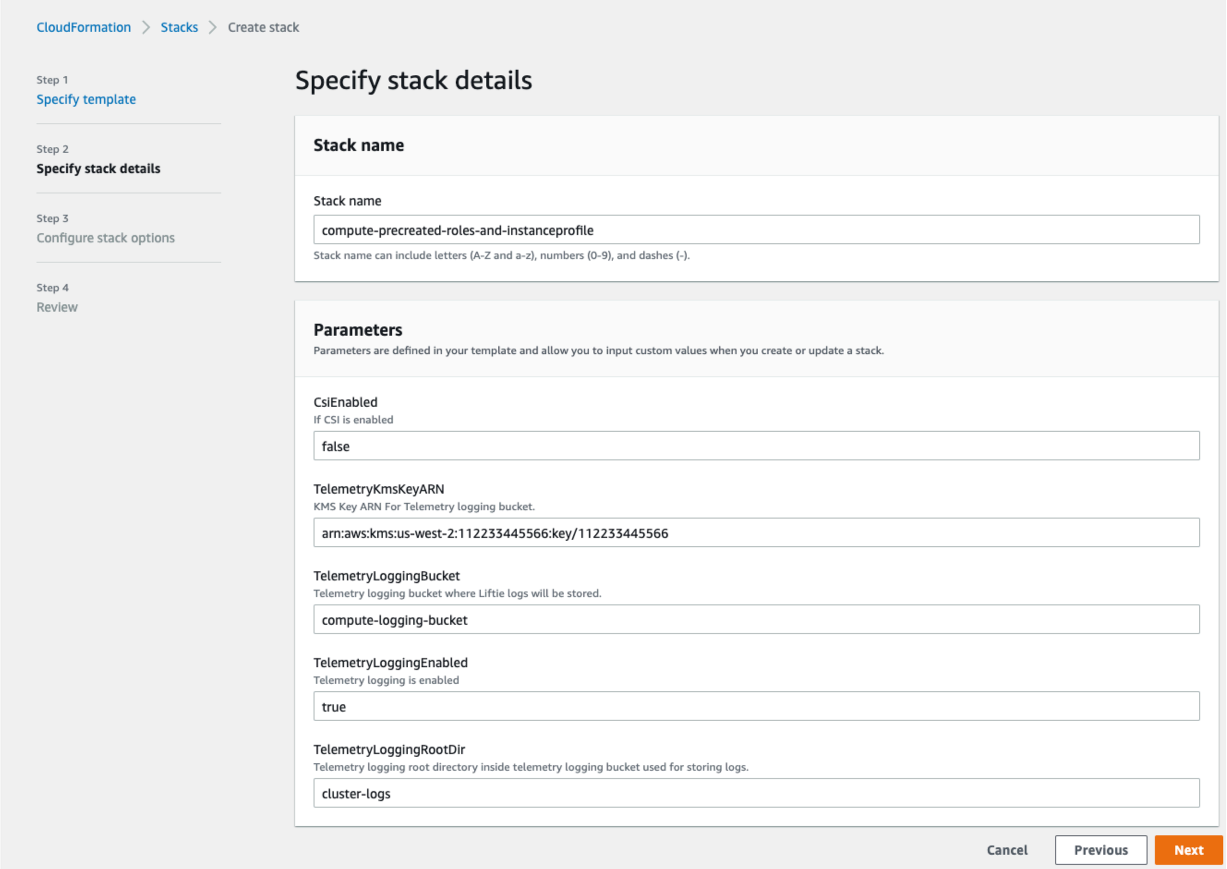
This step describes the roles and instance profiles that you create and attach to EKS master and worker instances at runtime. This step is needed in customer environments where write permissions are not provided to Cloudera AI. The roles created here are used exclusively within the customer's account.

Use the following CloudFormation template to create:

- IAM role called cdp-eks-master-role
- IAM role and Instance Profile pair called cdp-liftie-instance-profile

1. To apply the template, you need to provide values for the following parameters in the AWS console CloudFormation wizard:

   - Stack Name: Provide an appropriate name. (Example : compute-precreated-roles-and-instanceprofile)
   - TelemetryLoggingBucket: Name of the log bucket (just the name, not s3://) (Example : compute-logging-bucket)
   - TelemetryLoggingEnabled: Set it to true
   - TelemetryLoggingRootDir: Keep the default value (which is cluster-logs)
   - TelemetryKMSKeyARN: If the telemetry bucket is encrypted, give the KMS Key ARN. Default value is null.



Specify stack details

Stack name

Stack name: compute-precreated-roles-and-instanceprofile

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

CsiEnabled
If CSI is enabled
false

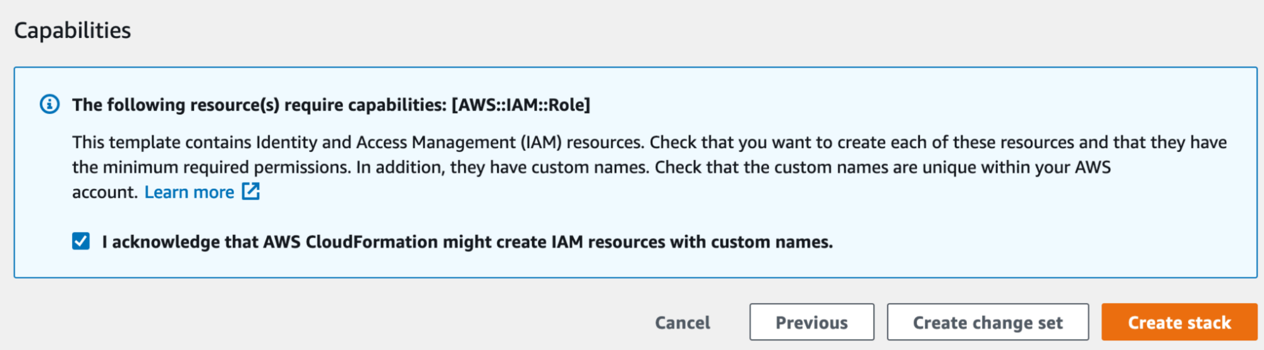
TelemetryKmsKeyARN
KMS Key ARN For Telemetry logging bucket.
arn:aws:kms:us-west-2:112233445566:key/112233445566

TelemetryLoggingBucket
Telemetry logging bucket where Liftie logs will be stored.
compute-logging-bucket

TelemetryLoggingEnabled
Telemetry logging is enabled
true

TelemetryLoggingRootDir
Telemetry logging root directory inside telemetry logging bucket used for storing logs.
cluster-logs

2. On the last page of the wizard, select the checkbox to allow creation of IAM resources with special names. If not selected, CloudFormation prepends the provided name with random prefixes to ensure uniqueness.



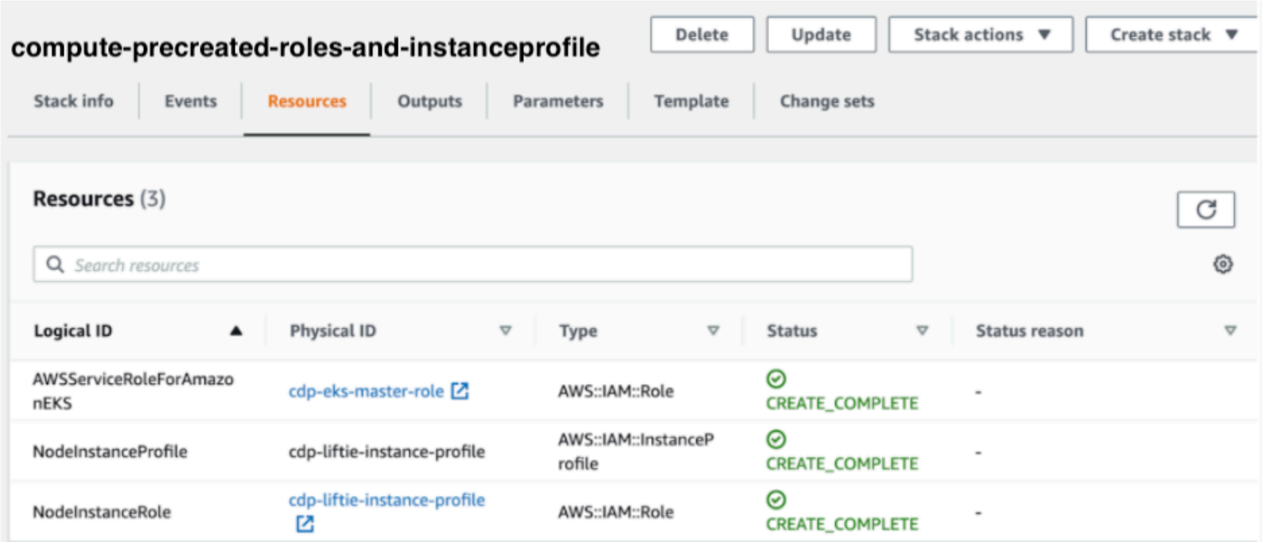
Capabilities

The following resource(s) require capabilities: [AWS::IAM::Role]

This template contains Identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. Learn more

☑ **I acknowledge that AWS CloudFormation might create IAM resources with custom names.**

The result of this procedure resembles the following:



Logical ID ▲	Physical ID	Type	Status	Status reason
AWSServiceRoleForAmazonEKS	cdp-eks-master-role	AWS::IAM::Role	✓ CREATE_COMPLETE	-
NodeInstanceProfile	cdp-liftie-instance-profile	AWS::IAM::InstanceProfile	✓ CREATE_COMPLETE	-
NodeInstanceRole	cdp-liftie-instance-profile	AWS::IAM::Role	✓ CREATE_COMPLETE	-

Use the following CloudFormation template for this process.

CloudFormation Template (format: YAML)

```
AWSTemplateFormatVersion: 2010-09-09
Description: Creates Liftie IAM resources
Parameters:
  TelemetryLoggingEnabled:
    Description: Telemetry logging is enabled
    Type: String
  TelemetryLoggingBucket:
    Description: Telemetry logging bucket where Liftie logs will be stored.
    Type: String
  TelemetryKmsKeyARN:
    Description: KMS Key ARN For Telemetry logging bucket.
    Type: String
    Default: ""
  TelemetryLoggingRootDir:
    Description: Telemetry logging root directory inside telemetry logging
bucket used for storing logs.
    Default: "cluster-logs"
    Type: String
Conditions:
  TelemetryLoggingEnabled:
    Fn::Equals:
      - {Ref: TelemetryLoggingEnabled}
      - true
  KMSKeyARNForTelemetryLoggingBucketIsEmpty: !Not [!Equals [!Ref Telemetry
KmsKeyARN, ""]]
Resources:
  AWSServiceRoleForAmazonEKS:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              Service:
                - eks.amazonaws.com
            Action:
              - sts:AssumeRole
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AmazonEKSServicePolicy
        - arn:aws:iam::aws:policy/AmazonEKSClusterPolicy
      RoleName: cdp-eks-master-role
  NodeInstanceRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              Service:
                - ec2.amazonaws.com
            Action:
              - sts:AssumeRole
      Path: "/"
      ManagedPolicyArns:
        - arn:aws:iam::aws:policy/AmazonEKSWorkerNodePolicy
        - arn:aws:iam::aws:policy/AmazonEKS_CNI_Policy
        - arn:aws:iam::aws:policy/AmazonEC2ContainerRegistryReadOnly
      RoleName: cdp-liftie-instance-profile
      Policies:
        - PolicyName: ssm-required
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - ssm:GetParameters
                Resource:
                  - "*"
        - PolicyName: cluster-autoscaler
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - autoscaling:DescribeAutoScalingGroups
                  - autoscaling:DescribeAutoScalingInstances
                  - autoscaling:DescribeLaunchConfigurations
                  - autoscaling:DescribeScalingActivities
                  - autoscaling:DescribeTags
                  - ec2:DescribeImages
                  - ec2:DescribeInstanceTypes
                  - ec2:DescribeLaunchTemplateVersions
                  - ec2:GetInstanceTypesFromInstanceRequirements
                  - eks:DescribeNodegroup
                  Resource:
                  - "*"
                - Effect: Allow
                  Action:
                     - autoscaling:SetDesiredCapacity
                     - autoscaling:TerminateInstanceInAutoScalingGroup
                  Resource:
                    - "*"
                  Condition:
                    StringEquals:
                      "aws:ResourceTag/k8s.io/cluster-autoscaler/enabled":
 "true"
        - PolicyName: ebs-csi
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - ec2:CreateSnapshot
                  - ec2:AttachVolume
                  - ec2:DetachVolume
                  - ec2:ModifyVolume
                  - ec2:DescribeAvailabilityZones
                  - ec2:DescribeInstances
                  - ec2:DescribeSnapshots
                  - ec2:DescribeTags
                  - ec2:DescribeVolumes
                  - ec2:DescribeVolumesModifications
                Resource: "*"
              - Effect: Allow
                Action:
                  - ec2:CreateTags
                Resource:
                  - "arn:aws:ec2:*:*:volume/*"
                  - "arn:aws:ec2:*:*:snapshot/*"
                Condition:
                  StringEquals:
                    "ec2:CreateAction":
                      - CreateVolume
                      - CreateSnapshot
              - Effect: Allow
                Action:
                  - ec2:DeleteTags
                Resource:
                  - "arn:aws:ec2:*:*:volume/*"
                  - "arn:aws:ec2:*:*:snapshot/*"
              - Effect: Allow
                Action:
                  - ec2:CreateVolume
                Resource: "*"
                Condition:
                  StringLike:
                    "aws:RequestTag/ebs.csi.aws.com/cluster": "true"
              - Effect: Allow
                Action:
                  - ec2:CreateVolume
                Resource: "*"
                Condition:
                  StringLike:
                    "aws:RequestTag/CSIVolumeName": "*"
              - Effect: Allow
                Action:
                  - ec2:CreateVolume
                Resource: "*"
                Condition:
                  StringLike:
                    "aws:RequestTag/kubernetes.io/cluster/*": "owned"
              - Effect: Allow
                Action:
                  - ec2:DeleteVolume
                Resource: "*"
                Condition:
                  StringLike:
                    "ec2:ResourceTag/ebs.csi.aws.com/cluster": "true"
              - Effect: Allow
                Action:
                  - ec2:DeleteVolume
                Resource: "*"
                Condition:
                  StringLike:
                    "ec2:ResourceTag/CSIVolumeName": "*"
```

```
              - Effect: Allow
                Action:
                  - ec2:DeleteVolume
                Resource: "*"
                Condition:
                  StringLike:
                    "ec2:ResourceTag/kubernetes.io/created-for/pvc/name": "
*"
              - Effect: Allow
                Action:
                  - ec2:DeleteSnapshot
                Resource: "*"
                Condition:
                  StringLike:
                    "ec2:ResourceTag/CSIVolumeSnapshotName": "*"
              - Effect: Allow
                Action:
                  - ec2:DeleteSnapshot
                Resource: "*"
                Condition:
                  StringLike:
                    "ec2:ResourceTag/ebs.csi.aws.com/cluster": "true"
        - PolicyName: efs-csi
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - elasticfilesystem:DescribeAccessPoints
                  - elasticfilesystem:DescribeFileSystems
                  - elasticfilesystem:DescribeMountTargets
                  - elasticfilesystem:TagResource
                Resource: "*"
              - Effect: Allow
                Action:
                  - elasticfilesystem:CreateAccessPoint
                Resource: "*"
                Condition:
                  StringLike:
                    "aws:RequestTag/efs.csi.aws.com/cluster": "true"
              - Effect: Allow
                Action:
                  - elasticfilesystem:DeleteAccessPoint
                Resource: "*"
                Condition:
                  StringEquals:
                    "aws:ResourceTag/efs.csi.aws.com/cluster": "true"
        - !If
          - TelemetryLoggingEnabled
          - PolicyName: telemetry-s3-list-bucket
            PolicyDocument:
              Version: 2012-10-17
              Statement:
                - Effect: Allow
                  Action:
                    - s3:ListBucket
                  Resource:
                    - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}'
                    - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}/${Telem
etryLoggingRootDir}/*'
          - !Ref 'AWS::NoValue'
        - !If
          - TelemetryLoggingEnabled
          - PolicyName: telemetry-s3-read-write
```

```
            PolicyDocument:
              Version: 2012-10-17
              Statement:
                - Effect: Allow
                  Action:
                    - s3:*Object
                    - s3:AbortMultipartUpload
                    - s3:GetBucketAcl
                  Resource:
                    - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}'
                    - !Sub 'arn:aws:s3:::${TelemetryLoggingBucket}/${Tele
metryLoggingRootDir}/*'
          - !Ref 'AWS::NoValue'
        - !If
          - KMSKeyARNForTelemetryLoggingBucketIsEmpty
          - PolicyName: s3-kms-read-write-policy
            PolicyDocument:
              Version: 2012-10-17
              Statement:
                - Effect: Allow
                  Action:
                    - kms:CreateGrant
                    - kms:Decrypt
                    - kms:GenerateDataKey
                    - kms:GenerateDataKeyWithoutPlainText
                  Resource:
                    - !Sub ${TelemetryKmsKeyARN}
          - !Ref 'AWS::NoValue'
        - PolicyName: calico-cni
          PolicyDocument:
            Version: 2012-10-17
            Statement:
              - Effect: Allow
                Action:
                  - ec2:ModifyInstanceAttribute
                Resource:
                  - "*"
  NodeInstanceProfile:
    Type: AWS::IAM::InstanceProfile
    Properties:
      Path: /
      InstanceProfileName: cdp-liftie-instance-profile
      Roles:
        - !Ref NodeInstanceRole
```

Create role and policy used to deploy Cloudera environments for Cloudera AI

The Cloudera AI control plane requires a role and policies to create Cloudera environments. In this step, you create a common policy for creating environments, as well as a policy that is specific to Cloudera AI environments.

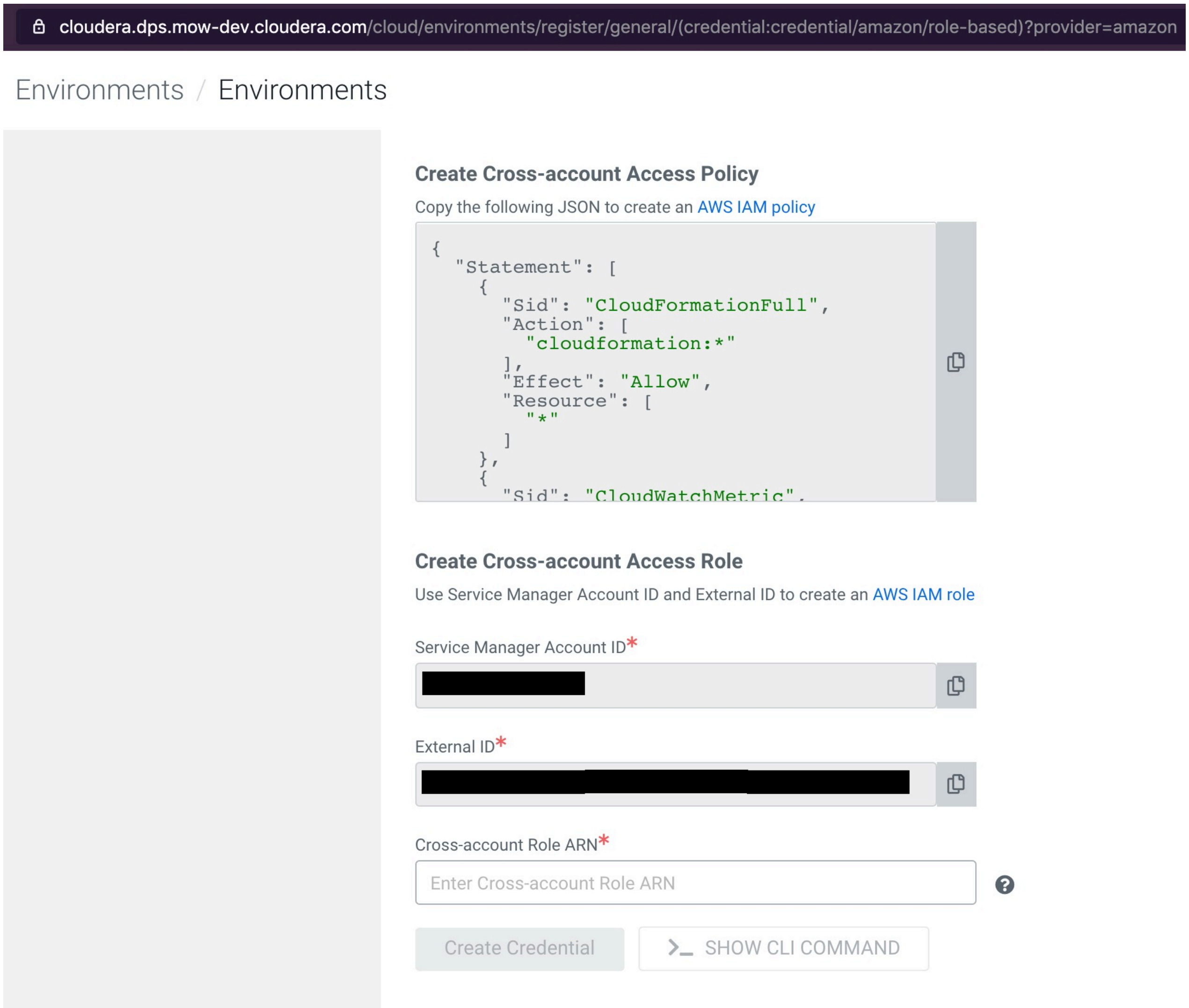
The following two policies are created in this step:

- Compute infrastructure restricted IAM policy - A common policy for all data services deployed on Cloudera.
- Cloudera AI restricted IAM policy - A policy with additional permissions for Cloudera AI.

There are two options for the timing of attaching the role: during environment creation, or prior to enabling the Cloudera AI data service.

Option #1: During environment creation

The Cloudbreak environment creation UI should be set up as shown here:


cloudera.dps.mow-dev.cloudera.com/cloud/environments/register/general/(credential:credential/amazon/role-based)?provider=amazon
Environments / Environments

Create Cross-account Access Policy

Copy the following JSON to create an AWS IAM policy

```
{
  "Statement": [
    {
      "Sid": "CloudFormationFull",
      "Action": [
        "cloudformation:*"
      ],
      "Effect": "Allow",
      "Resource": [
        "*"
      ]
    },
    {
      "Sid": "CloudWatchMetric",
```

Create Cross-account Access Role

Use Service Manager Account ID and External ID to create an AWS IAM role

Service Manager Account ID*

[redacted]

External ID*

[redacted]

Cross-account Role ARN*

Enter Cross-account Role ARN

Note:

- For the AWS IAM policy mentioned in the “Create Cross-account Access Policy” section, use the Compute infrastructure Restricted IAM and Cloudera AI Restricted IAM policies below and create as new policies in AWS IAM. There may be one or more restricted policies already attached to the cross-account role, in addition to the Compute infrastructure and Cloudera AI restricted policies. For example, there may also be a Cloudera Data Hub restricted policy.
- For the “Create Cross-account Access Role” section, create the cross-account role as instructed (or update the role if one already exists) and attach the newly created Compute infrastructure Restricted IAM policy and Cloudera AI Restricted IAM policy. Finally, update the cross-account role to use it.

Option #2: Prior to enabling Cloudera AI data service

If the Cloudbreak environment has already been created, you can create and attach the Compute infrastructure Restricted IAM policy and Cloudera AI restricted IAM policy to the existing cross-account role associated with the environment.

To view the existing cross-account role, in the Environments section of the Cloudera Management Console, on the Summary tab, see Credentials.

Note: There may be one or more restricted policies already attached to the cross-account role, in addition to the Compute infrastructure and Cloudera AI restricted policies. For example, there might be a Data Hub restricted policy. These should be left in place.

Compute (Liftie) Restricted IAM policy

Replace the following placeholders in the JSON file:

- [YOUR-ACCOUNT-ID] with your account ID in use.
- [YOUR-IAM-ROLE-NAME] with the IAM restricted role associated with this policy.
- [YOUR-SUBNET-ARN-*] supplied during the Cloudbreak Environment(s) creation. Note: Please provide all the subnets present in all the Cloudbreak Environment(s) that you intend to use it for the experience. If at any point a new Cloudbreak Environment is created or an existing one is updated for subnets, the same should be updated here.
- [YOUR-IDBROKER-ROLE-NAME] with the ID Broker Role name in use.
- [YOUR-LOG-ROLE-NAME] with the Log Role name in use.
- [YOUR-KMS-CUSTOMER-MANAGED-KEY-ARN] with KMS key ARN.
- [YOUR-ACCOUNT-REGION] with the AWS region.

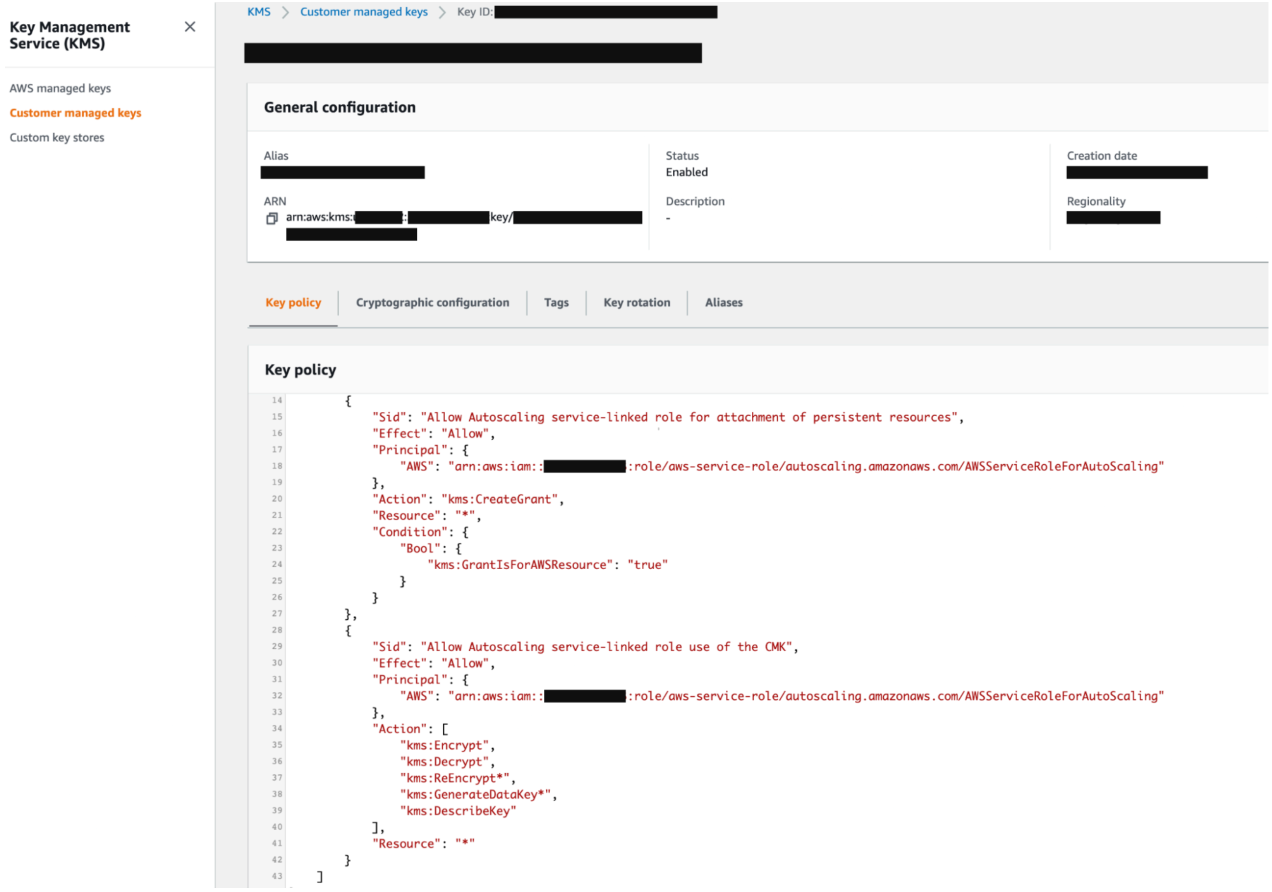
```
{
  "Version": "2012-10-17",
  "Id": "ComputePolicy_v12",
  "Statement": [
    {
      "Sid": "SimulatePrincipalPolicy",
      "Effect": "Allow",
      "Action": [
        "iam:SimulatePrincipalPolicy"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-IAM-ROLE-NAME]"
      ]
    },
    {
      "Sid": "RestrictedPermissionsViaClouderaRequestTag",
      "Effect": "Allow",
      "Action": [
        "cloudformation:CreateStack",
        "cloudformation:CreateChangeSet",
        "ec2:createTags",
        "eks:TagResource"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "aws:RequestTag/Cloudera-Resource-Name": [
            "crn:cdp:*"
          ]
        }
      }
    },
    {
      "Sid": "RestrictedPermissionsViaClouderaResourceTag",
      "Effect": "Allow",
      "Action": [
        "autoscaling:DetachInstances",
        "autoscaling:ResumeProcesses",
        "autoscaling:SetDesiredCapacity",
        "autoscaling:SuspendProcesses",
        "autoscaling:UpdateAutoScalingGroup",
        "autoscaling:DeleteTags",
        "autoscaling:TerminateInstanceInAutoScalingGroup",
        "cloudformation:DeleteStack",
        "cloudformation:DescribeStacks"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/Cloudera-Resource-Name": [
            "crn:cdp:*"
          ]
        }
      }
    },
    {
      "Sid": "RestrictedPermissionsViaCloudFormation",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateSecurityGroup",
        "ec2:DeleteSecurityGroup",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:RevokeSecurityGroupIngress",
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:RevokeSecurityGroupEgress",
        "ec2:CreateLaunchTemplate",
        "ec2:DeleteLaunchTemplate",
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:DeleteAutoScalingGroup",
        "autoscaling:CreateOrUpdateTags",
        "autoscaling:CreateLaunchConfiguration",
        "eks:CreateCluster",
        "eks:DeleteCluster"
      ],
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:CalledVia": [
            "cloudformation.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "RestrictedEC2PermissionsViaClouderaResourceTag",
      "Effect": "Allow",
      "Action": [
        "ec2:RebootInstances",
        "ec2:StartInstances",
        "ec2:StopInstances",
        "ec2:TerminateInstances"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "ForAnyValue:StringLike": {
          "ec2:ResourceTag/Cloudera-Resource-Name": [
            "crn:cdp:*"
          ]
        }
      }
    },
    {
      "Sid": "RestrictedIamPermissionsToClouderaResources",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-IDBROKER-ROLE-NAME]",
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-LOG-ROLE-NAME]",
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/liftie-*-eks-service-role",
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/liftie-*-eks-worker-nodes",
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/cdp-eks-master-role",
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/cdp-liftie-instance-profile"
      ]
    },
    {
      "Sid": "RestrictedKMSPermissionsUsingCustomerProvidedKey",
      "Effect": "Allow",
      "Action": [
        "kms:CreateGrant",
        "kms:DescribeKey",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*"
      ],
      "Resource": [
        "[YOUR-KMS-CUSTOMER-MANAGED-KEY-ARN]"
      ]
    },
    {
      "Sid": "AllowCreateDeleteTagsForSubnets",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2:DeleteTags"
      ],
      "Resource": [
        "arn:aws:ec2:[YOUR-SUBNET-REGION]:[YOUR-ACCOUNT-ID]:subnet/*"
      ]
    },
    {
      "Sid": "OtherPermissionsViaCloudFormation",
      "Effect": "Allow",
      "Action": [
        "autoscaling:DescribeScheduledActions",
        "autoscaling:DescribeTags",
        "autoscaling:DescribeAutoScalingInstances",
        "autoscaling:DescribeLaunchConfigurations",
        "autoscaling:DeleteLaunchConfiguration",
        "autoscaling:DescribeScalingActivities",
        "dynamodb:DescribeTable",
        "ec2:DeletePlacementGroup",
        "ec2:DescribeAccountAttributes",
        "ec2:DescribeImages",
        "ec2:DescribeInstanceStatus",
        "ec2:DescribeInstances",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeLaunchTemplateVersions",
        "ec2:DescribeLaunchTemplates",
        "ec2:DescribePlacementGroups",
        "ec2:DescribeRegions",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVolumes"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:CalledVia": [
            "cloudformation.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "ModifyInstanceAttribute",
      "Effect": "Allow",
      "Action": [
        "ec2:ModifyInstanceAttribute"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringEquals": {
          "ec2:Attribute": "SourceDestCheck"
        }
      }
    },
    {
      "Sid": "OtherPermissionsViaClouderaResourceTag",
      "Effect": "Allow",
      "Action": [
        "cloudformation:DescribeChangeSet",
        "cloudformation:DeleteChangeSet",
        "cloudformation:ExecuteChangeSet",
        "cloudformation:CancelUpdateStack",
        "cloudformation:ContinueUpdateRollback",
        "cloudformation:ListStacks",
        "cloudformation:DescribeStackEvents",
        "cloudformation:DescribeStackResource",
        "cloudformation:DescribeStackResources",
        "cloudwatch:deleteAlarms",
        "cloudwatch:putMetricAlarm",
        "logs:DescribeLogStreams",
        "logs:FilterLogEvents",
        "ec2:AttachVolume",
        "ec2:CreateNetworkInterface",
        "ec2:CreateVolume",
        "ec2:DeleteVolume",
        "ec2:RunInstances",
        "eks:ListUpdates",
        "eks:UpdateClusterConfig",
        "eks:UpdateClusterVersion",
        "eks:DescribeUpdate",
        "iam:GetRolePolicy",
        "iam:ListInstanceProfiles",
        "iam:ListRoleTags",
        "iam:RemoveRoleFromInstanceProfile",
        "iam:TagRole",
        "iam:UntagRole"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/Cloudera-Resource-Name": [
            "crn:cdp:*"
          ]
        }
      }
    },
    {
      "Sid": "OtherPermissions",
      "Effect": "Allow",
      "Action": [
        "autoscaling:DescribeAutoScalingGroups",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:CreateLaunchTemplateVersion",
        "ec2:CreatePlacementGroup",
        "ec2:DeleteKeyPair",
        "ec2:DeleteNetworkInterface",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeInstanceTypes",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcAttribute",
        "ec2:DescribeVpcs",
        "ec2:ImportKeyPair",
        "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
        "ec2:GetInstanceTypesFromInstanceRequirements",
        "eks:DescribeCluster",
        "eks:CreateAccessEntry",
        "eks:DeleteAccessEntry",
        "eks:ListAccessEntries",
        "eks:DescribeAccessEntry",
        "eks:AssociateAccessPolicy",
        "eks:DisassociateAccessPolicy",
        "eks:ListAssociatedAccessPolicies",
        "elasticloadbalancing:DescribeLoadBalancers",
        "iam:GetRole",
        "iam:ListRoles",
        "iam:GetInstanceProfile"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Sid": "AllowSsmParams",
      "Effect": "Allow",
      "Action": [
        "ssm:DescribeParameters",
        "ssm:GetParameter",
        "ssm:GetParameters",
        "ssm:GetParameterHistory",
        "ssm:GetParametersByPath"
      ],
      "Resource": [
        "arn:aws:ssm:*:*:parameter/aws/service/eks/optimized-ami/*"
      ]
    },
    {
      "Sid": "CfDeny",
      "Effect": "Deny",
      "Action": [
        "cloudformation:*"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "ForAnyValue:StringLike": {
          "cloudformation:ImportResourceTypes": [
            "*"
          ]
        }
      }
    },
    {
      "Sid": "ForAutoscalingLinkedRole",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/autoscaling-
plans.amazonaws.com/AWSServiceRoleForAutoScalingPlans_EC2AutoScaling"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "autoscaling-plans.amazonaws.com"
        }
      }
    },
    {
      "Sid": "ForEksLinkedRole",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/eks.amazonaws.
com/AWSServiceRoleForEKS"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "eks.amazonaws.com"
        }
      }
    }
  ]
}
```

Supporting Customer Managed CMKs

Along with providing the KMS Customer Managed Customer Master Key (CMK) for volume encryption in the policy section with Sid: RestrictedKMSPermissionsUsingCustomerProvidedKey, you need to verify that the policy for the Customer Managed Customer Master Key (CMK) at KMS (this is not an IAM policy) has the following three permission blocks defined for AWSServiceRoleForAutoScaling.

```
{
  "Statement": [
    {
      "Sid": "AllowAutoscalingServiceLinkedRoleForAttachmentOfPersistentRes
ources",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/auto
scaling.amazonaws.com/AWSServiceRoleForAutoScaling"
      },
      "Action": "kms:CreateGrant",
      "Resource": "*",
      "Condition": {
        "Bool": {
          "kms:GrantIsForAWSResource": "true"
        }
      }
    },
    {
      "Sid": "AllowAutoscalingServiceLinkedRoleUseOfTheCMK",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/autos
caling.amazonaws.com/AWSServiceRoleForAutoScaling"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
      ],
      "Resource": "*"
    },
    {
      "Sid": "Allow EKS access to EBS.",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "kms:CreateGrant",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:CallerAccount": "[YOUR-ACCOUNT-ID]",
          "kms:viaService": "ec2.[YOUR-ACCOUNT-REGION].amazonaws.com"
        }
      }
    }
  ]
}
```

After the policy is attached, the KMS service page will show the CMS as having the policy attached, similar to this screen shot:



Key Management Service (KMS)
AWS managed keys
Customer managed keys
Custom key stores
KMS > Customer managed keys > Key ID: [redacted]

[redacted]

General configuration

Alias
[redacted]

ARN
arn:aws:kms:[redacted]:[redacted]key/[redacted]
[redacted]

Status
Enabled

Description
-

Creation date
[redacted]

Regionality
[redacted]

Key policy
Cryptographic configuration
Tags
Key rotation
Aliases

Key policy

```
        {
            "Sid": "Allow Autoscaling service-linked role for attachment of persistent resources",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::[redacted]:role/aws-service-role/autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling"
            },
            "Action": "kms:CreateGrant",
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        },
        {
            "Sid": "Allow Autoscaling service-linked role use of the CMK",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::[redacted]:role/aws-service-role/autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        }
    ]
```

Cloudera AI restricted IAM policy

Replace the following placeholders in the JSON file:

- [YOUR-ACCOUNT-ID] with your account ID in use.
- [YOUR-IAM-ROLE-NAME] with the IAM restricted role with which this policy would be associated with.

```
{
  "Version": "2012-10-17",
  "Id": "CMLPolicy_v1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:SimulatePrincipalPolicy",
      "Resource": "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/[YOUR-IAM-ROLE-NAME]"
    },
    {
      "Sid": "RestrictedPermissionsViaClouderaRequestTag",
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:CreateFileSystem"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "aws:RequestTag/Cloudera-Resource-Name": "crn:cdp:*"
        }
      }
    },
    {
      "Sid": "OtherPermissions",
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:DescribeMountTargets",
        "elasticfilesystem:DeleteAccessPoint",
        "elasticfilesystem:CreateMountTarget",
        "elasticfilesystem:DescribeAccessPoints",
        "elasticfilesystem:DescribeFileSystems",
        "elasticfilesystem:DeleteMountTarget",
        "elasticfilesystem:CreateAccessPoint",
        "elasticfilesystem:DeleteFileSystem",
        "elasticfilesystem:DescribeMountTargetSecurityGroups"
      ],
      "Resource": "*"
    },
    {
      "Sid": "ForEFSLinkedRole",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::[YOUR-ACCOUNT-ID]:role/aws-service-role/elasticfilesys
tem.amazonaws.com/AWSServiceRoleForAmazonElasticFileSystem"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "elasticfilesystem.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowAttachRolePolicy",
      "Effect": "Allow",
      "Action": "iam:AttachRolePolicy",
      "Resource": "*"
    }
  ]
}
```

Use a non-transparent proxy with Cloudera AI on AWS environments

Cloudera AI can use non-transparent proxies if the environment is configured to use a network proxy in Management Console.

Enterprise customers frequently need to deploy Cloudera in a virtual network that does not have direct internet access. Specifically, the proxy server may be located in a different virtual network, in order to filter traffic for allowed domains or IPs.

Transparent and non-transparent network proxies differ in the following ways.

Transparent network proxy

- Proxy is unknown to clients and requires no additional client configuration.
- Usually, connections by way of transparent proxies are configured in route tables on your AWS VPC.

Non-transparent proxy

- Clients are aware of non-transparent proxies and each client must be specifically configured to use the non-transparent proxy connection.
- You pass connection or security information (username/password) along with the connection request sent by clients.

You can configure an AWS environment to use non-transparent proxy connections when activating environments for Cloudera AI.

Use a non-transparent proxy in a different VPC

If the customer wants to copy the hostname for the non-transparent proxy and the non-transparent proxy is configured in a different VPC, then Cloudera needs the CIDR of the non-transparent proxy to allow the inbound access. To configure this, in the Provision Cloudera AI Workbench UI, select Use hostname for non-transparent proxy and enter the CIDR range in Inbound Proxy CIDR Ranges.

Related Information

Using a non-transparent proxy

Certified scale limitations for Cloudera AI Workbenches on AWS

The listed scale limitations have been certified with Cloudera AI Workbenches on AWS.

The Cloudera AI Workbench is certified to support the following:

- 1000 logged in users using the product concurrently
- 850 concurrent interactive Sessions
- 500 concurrent cron jobs executing at the same frequency
- 500 concurrent Applications
- 300 Models deployed concurrently