

# Deploy C8000v High Availability Configuration on AWS

## Contents

---

### [Introduction](#)

### [Prerequisites](#)

[Requirements](#)

[Components Used](#)

### [Topology](#)

[Network Diagram](#)

[Table Summary](#)

### [Restrictions](#)

### [Configuration](#)

[Step 1. Select a Region](#)

[Step 2. Create the VPC](#)

[Step 3. Create a Security Group for the VPC](#)

[Step 4. Create an IAM Role with a Policy and Associate to the VPC](#)

[Step 5. Create and Attach a Trust Policy to an IAM Role](#)

[Step 6. Configure and Launch the C8000v Instances](#)

[Step 6.1. Configure the Key Pair for Remote Access](#)

[Step 6.2. Create and Configure the Subnets for the AMI](#)

[Step 6.3. Configure the AMI Interfaces](#)

[Step 6.4. Set the IAM Instance Profile to the AMI](#)

[Step 6.5. \(Optional\) Set the Credentials on the AMI](#)

[Step 6.6. Finish the Instance Configuration](#)

[Step 6.7. Disable Source/Destination Check on the ENIs](#)

[Step 6.8. Create and Associate an Elastic IP to the Public ENI of the Instance](#)

[Step 7. Repeat Step 6 to Create the Second C8000v Instance for HA](#)

[Step 8. Repeat Step 6 to Create a VM \(Linux/Windows\) from the AMI Marketplace](#)

[Step 9. Create and Configure an Internet Gateway \(IGW\) for the VPC](#)

[Step 10. Create and Configure Route Tables on AWS for Public and Private subnets](#)

[Step 10.1. Create and Configure the Public Route Table](#)

[Step 10.2. Create and Configure the Private Route Table](#)

[Step 11. Check and Configure Basic Network Configuration, Network Address Translation\(NAT\), GRE Tunnel with BFD and Routing Protocol](#)

[Step 12. Configure High Availability \(Cisco IOS® XE Denali 16.3.1a or later\)](#)

### [Verification](#)

### [Troubleshoot](#)

---

## Introduction

This document describes how to setup a High Availability environment with Catalyst 8000v routers on Amazon Web Services cloud.

# Prerequisites

## Requirements

Cisco recommends that you have previous knowledge of the these topics:

- General knowledge of AWS Console and its components
- Understanding of Cisco IOS® XE software
- Basic knowledge of HA feature.

## Components Used

These components are required for this configuration example:

- An Amazon AWS account with administrator role
- Two C8000v devices running Cisco IOS® XE 17.15.3a and 1 Ubuntu 22.04 LTS VM AMIs in the same region

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

## Topology

There are various scenarios of HA deployment based on the network requirements. For this example, HA redundancy is configured with these settings:

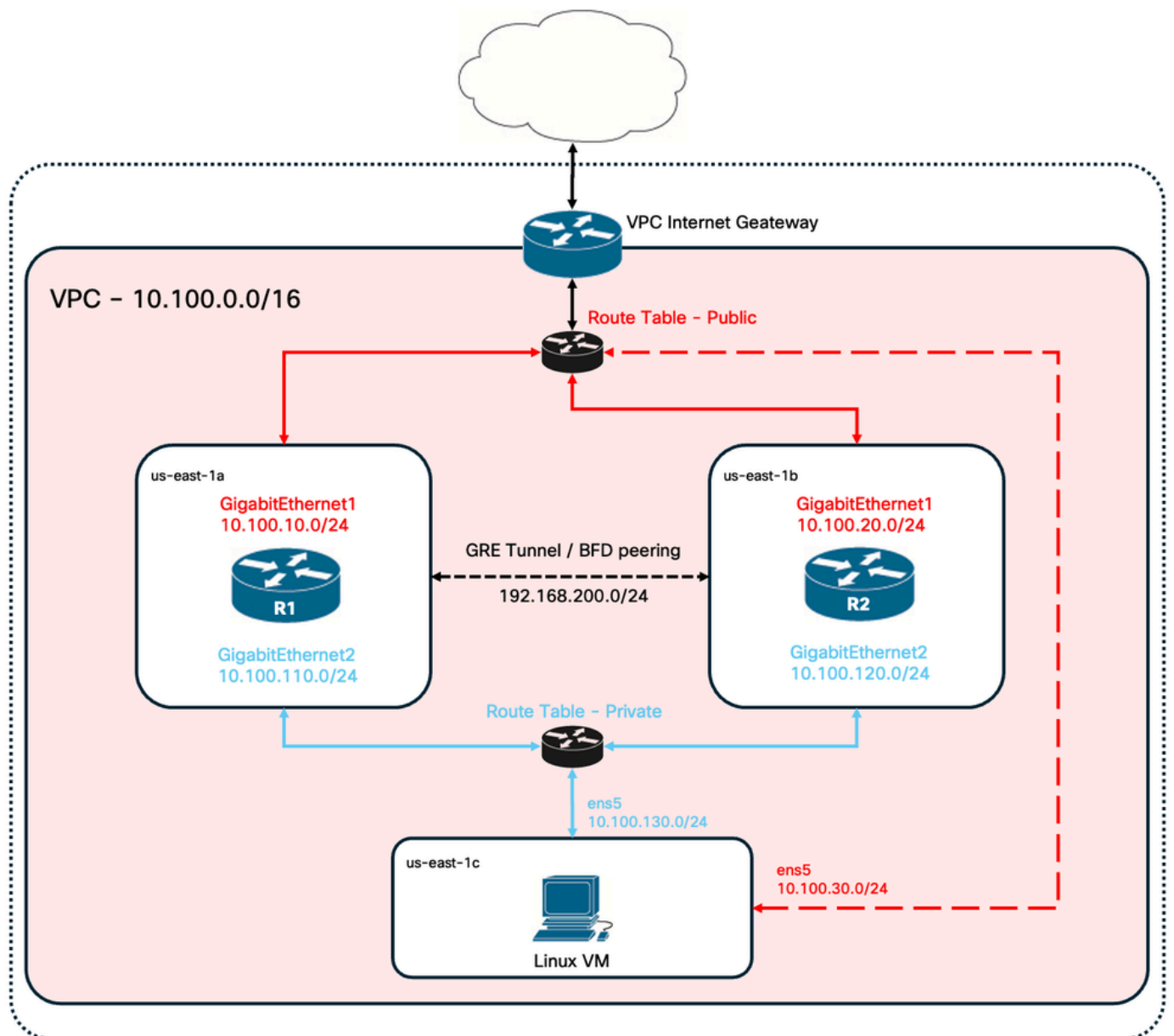
- 1x - Region
- 1x - VPC
- 3x - Availability Zones
- 6x - Network Interfaces/Subnets (3x Public Facing/3x Private Facing)
- 2x - Route Tables ( Public & Private )
- 2x - C8000v routers (Cisco IOS® XEDenali 17.15.3a)
- 1x - VM (Linux/Windows)

There are 2 C8000v routers in an HA pair, in two different availability zones. Think of each availability zone as a separate datacenter for additional hardware resiliency.

The third zone is a VM, which simulates a device in a private datacenter. For now, internet access is enabled through the public interface so that you can access and configure the VM. Generally, all normal traffic must flow through the private route table.

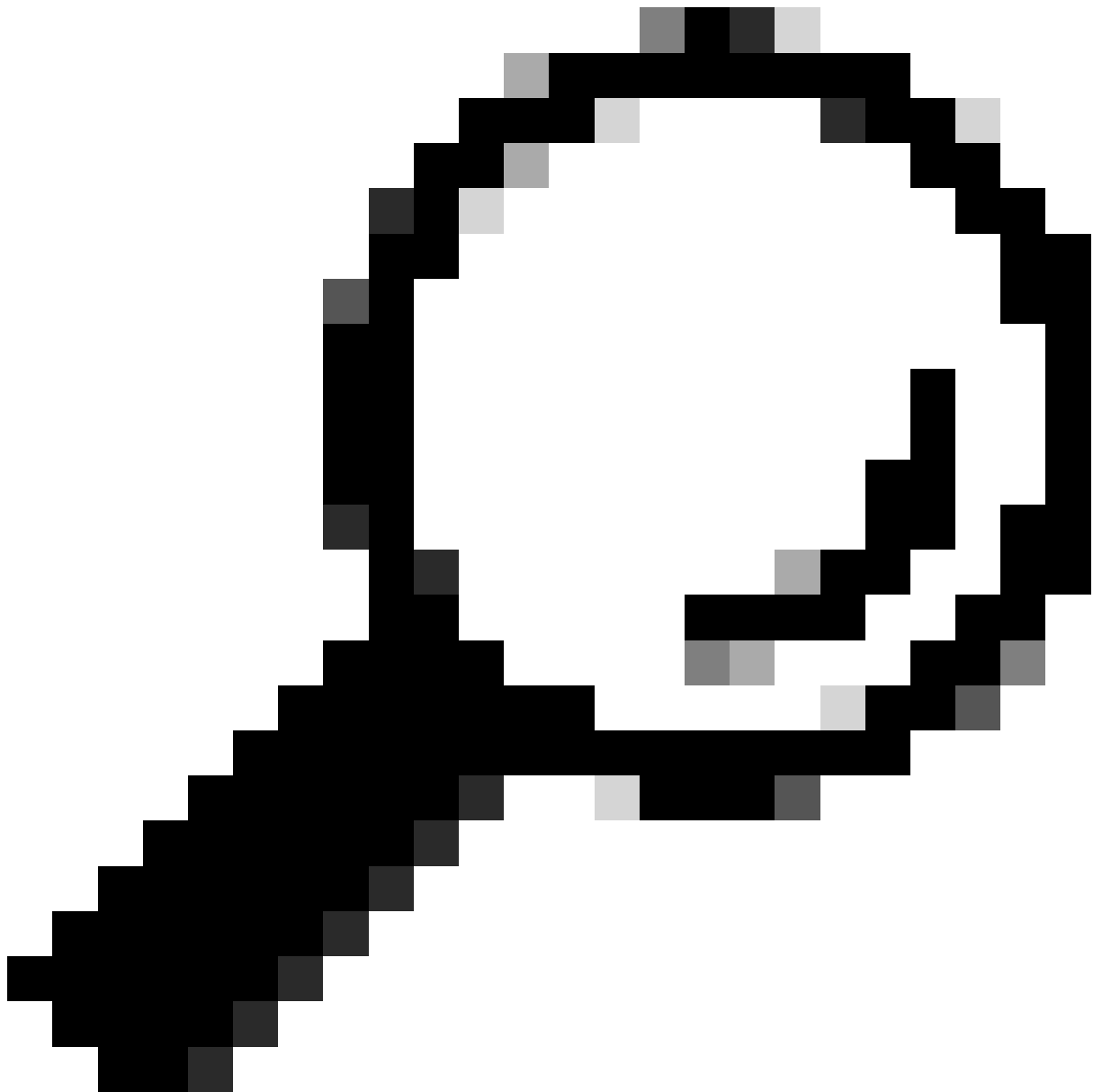
To simulate traffic, initiate a ping from the Virtual Machine's private interface, traversing the private route table through R1 to reach 8.8.8.8. In the event of a failover, verify that the private route table has automatically updated to route traffic through the private interface of the R2 Router.

## Network Diagram



## Table Summary

To summarize the topology, here is the table with the most important values from each component in the lab. The information provided in this table is exclusive for this lab.



**Tip:** Using this table helps maintain a clear overview of key variables throughout the guide. Collecting the information in this format is recommended to streamline the process.

| Device   | Availability Zone | Interfaces       | IP Addresses   | RTB                             | ENI                   |
|----------|-------------------|------------------|----------------|---------------------------------|-----------------------|
| R1       | us-east-1a        | GigabitEthernet1 | 10.100.10.254  | rtb-0d0e48f25c9b00635 (public)  | eni-0645a881c13823696 |
|          |                   | GigabitEthernet2 | 10.100.110.254 | rtb-093df10a4de426eb8 (private) | eni-070e14fbfde0d8e3b |
| R2       | us-east-1b        | GigabitEthernet1 | 10.100.20.254  | rtb-0d0e48f25c9b00635 (public)  | eni-0a7817922ffbb317b |
|          |                   | GigabitEthernet2 | 10.100.120.254 | rtb-093df10a4de426eb8 (private) | eni-0239fda341b4d7e41 |
| Linux VM | us-east-1c        | ens5             | 10.100.30.254  | rtb-0d0e48f25c9b00635 (public)  | eni-0b28560781b3435b1 |

|  |  |      |                |                                    |                       |
|--|--|------|----------------|------------------------------------|-----------------------|
|  |  | ens6 | 10.100.130.254 | rtb-093df10a4de426eb8<br>(private) | eni-05d025e88b6355808 |
|--|--|------|----------------|------------------------------------|-----------------------|

## Restrictions

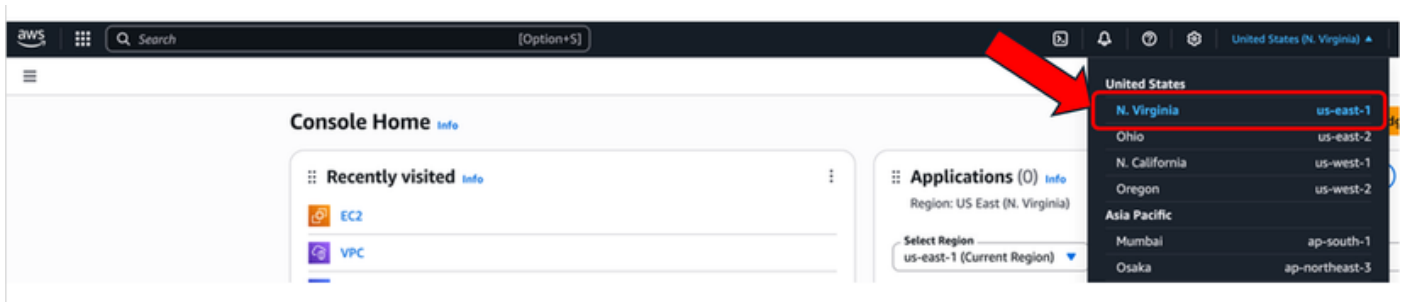
- On any subnet created, do not use the first available address of that subnet. These IP addresses are used by the AWS services internally.
- Do not configure the public interfaces of the C8000v devices inside a VRF. HA does not work properly if this is set.

## Configuration

The general flow of configuration is focused to create the requested VMs in the proper region and move your way down to the most specific configuration such as routes and interfaces of each one of them. However, it is recommended to understand the topology first and configure it in any order desired.

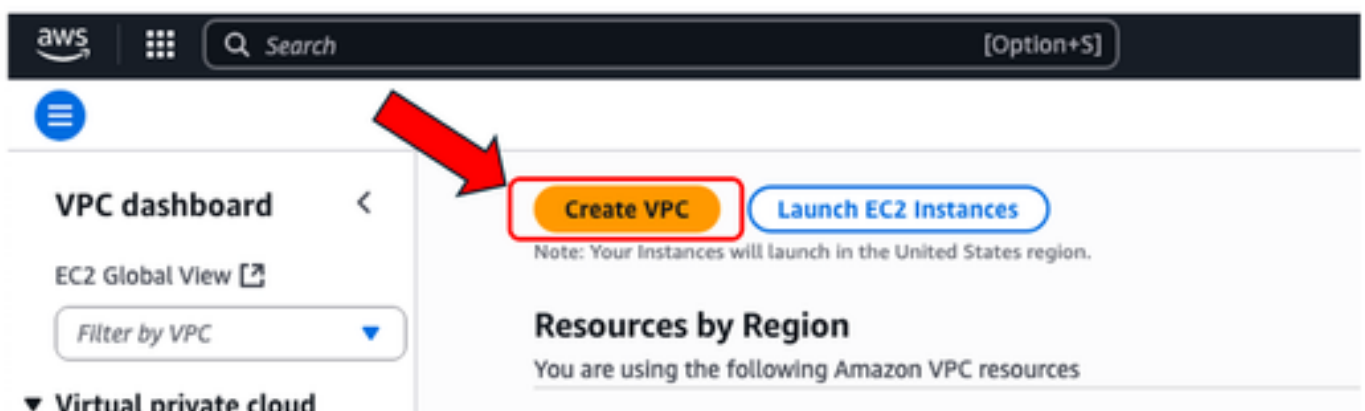
### Step 1. Select a Region

For this deployment guide, the **US West (North Virginia) - us-east-1** region is selected as the VPC region.



### Step 2. Create the VPC

On the AWS Console, navigate to **VPC > VPC Dashboard > Create VPC**.



When you create the VPC, select the **VPC only** option. You can assign a /16 network to use as you please.

On this deployment guide, the 10.100.0.0/16 network is selected:

**Create VPC** [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

**VPC settings**

**Resources to create** [Info](#)  
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

**Name tag - optional**  
Creates a tag with a key of 'Name' and a value that you specify.

HA

**IPv4 CIDR block** [Info](#)  
☒ IPv4 CIDR manual input  
☐ IPAM-allocated IPv4 CIDR block

**IPv4 CIDR**  
10.100.0.0/16  
CIDR block size must be between /16 and /28.

**IPv6 CIDR block** [Info](#)  
☒ No IPv6 CIDR block  
☐ IPAM-allocated IPv6 CIDR block  
☐ Amazon-provided IPv6 CIDR block  
☐ IPv6 CIDR owned by me

**Tenancy** [Info](#)  
Default

**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

**Key**  **Value - optional**  [Remove tag](#)

[Add tag](#)  
You can add 49 more tags.

[Cancel](#) [Preview code](#) [Create VPC](#)

After clicking **Create VPC**, the VPC-0d30b9fa9511f3639 with HA tag is now created:

**VPC dashboard** [EC2 Global View](#)

Filter by VPC

**Virtual private cloud**  
Your VPCs  
Subnets  
Route tables  
Internet gateways  
Egress-only Internet gateways  
Carrier gateways  
DHCP option sets  
Elastic IPs  
Managed prefix lists  
NAT gateways  
Peering connections  
Route servers [New](#)

**Security**  
Network ACLs  
Security groups

**PrivateLink and Lattice**  
Getting started [Updated](#)  
Endpoints [Updated](#)  
Frontend services

**You successfully created vpc-0d30b9fa9511f3639 / HA**

**vpc-0d30b9fa9511f3639 / HA** [Actions](#)

**Details** [Info](#)

VPC ID: vpc-0d30b9fa9511f3639  
DNS resolution: Enabled  
Main network ACL: acl-02519ab1454b877a8  
IPv6 CIDR (Network border group): -

**State** [Info](#)  
Available  
Tenancy: default  
Default VPC: No  
Network Address Usage metrics: Disabled

**Block Public Access** [Info](#)  
Off

**DHCP option set**  
dopt-e75e6f80

**IPv4 CIDR**  
10.100.0.0/16

**Route 53 Resolver DNS Firewall rule groups**  
-

**DNS hostnames**  
Disabled

**Main route table**  
rtb-0d0e48f25c9b00635

**IPv6 pool**  
-

**Owner ID**  
073713984176

**Resource map** [Info](#)

**VPC** [Show details](#)  
Your AWS virtual network  
HA

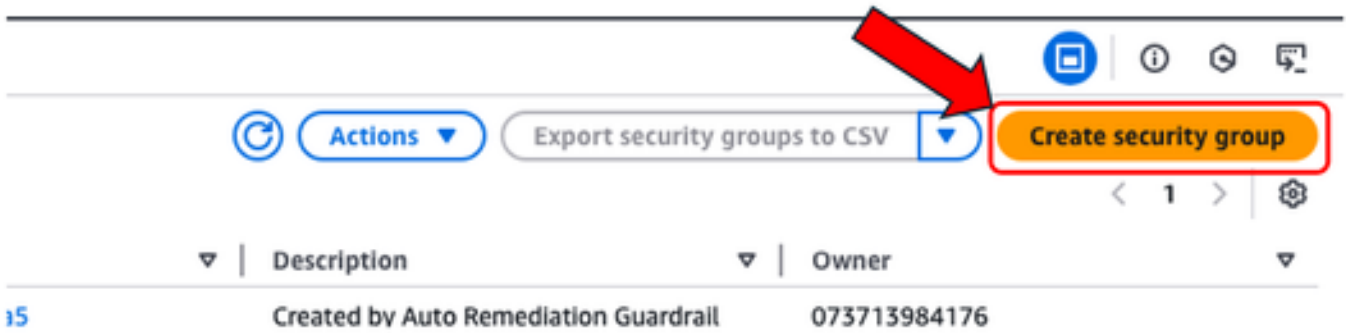
**Subnets (0)**  
Subnets within this VPC

**Route tables (1)**  
Route network traffic to resources  
rtb-0d0e48f25c9b00635

**Network connections (0)**  
Connections to other networks

### Step 3. Create a Security Group for the VPC

In AWS, Security Groups function like ACLs, allowing or denying traffic to configured VMs within a VPC. On the AWS Console, navigate to **VPC > VPC Dashboard > Security > Security Groups** section and click **Create security group**.



Under Inbound Rules, define what traffic you wish to allow for. For this example, **All Traffic** is selected by using the **0.0.0.0/0** network.

[VPC](#) > [Security Groups](#) > Create security group

### Create security group info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

**Basic details**

**Security group name** info

Name cannot be edited after creation.

**Description** info

**VPC** info

**Inbound rules** info

| Type                     | Protocol | Port range | Source      | Description - optional |             |
|--------------------------|----------|------------|-------------|------------------------|-------------|
| All traffic              | All      | All        | Anywhere... |                        | 0.0.0.0/0   |
|                          |          |            |             |                        | 0.0.0.0/0 X |
| <a href="#">Add rule</a> |          |            |             |                        |             |

**Outbound rules** info

| Type                     | Protocol | Port range | Destination | Description - optional |             |
|--------------------------|----------|------------|-------------|------------------------|-------------|
| All traffic              | All      | All        | Custom      |                        | 0.0.0.0/0   |
|                          |          |            |             |                        | 0.0.0.0/0 X |
| <a href="#">Add rule</a> |          |            |             |                        |             |

**Tags - optional**

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags

[Cancel](#) [Create security group](#)

## Step 4. Create an IAM Role with a Policy and Associate to the VPC

IAM grants to your AMIs the required access to Amazon APIs. The C8000v is used as a proxy to call AWS API commands to modify the route table in AWS. By default, EC2 instances are not allowed access to APIs. For this reason, a new IAM role must be created which is going to be applied during AMI creations.

Browse to the IAM dashboard, and navigate to **Access Management > Roles > Create Role**. This process consists of 3 steps:

The screenshot shows the AWS IAM console 'Roles' page. On the left, there's a navigation menu with 'Identity and Access Management (IAM)' and 'Access management' sections. The main area displays a list of roles with columns for 'Role name', 'Trusted entities', and 'Last activity'. The 'Create role' button is highlighted in the top right corner.

| Role name                                      | Trusted entities                                  | Last activity  |
|--|---|----------------|
| admin  | Identity Provider: arn:aws:iam::0737              | 52 days ago    |
| AmazonAppStreamServiceAccess                   | AWS Service: appstream                            | -              |
| ApplicationAutoScalingForAmazonAppStreamAccess | AWS Service: application-autoscaling              | -              |
| aws-codestar-service-role                      | AWS Service: codestar                             | -              |
| aws-elasticbeanstalk-ec2-role                  | AWS Service: ec2                                  | -              |
| aws-elasticbeanstalk-service-role              | AWS Service: elasticbeanstalk                     | -              |
| AWSCloud9SSMAccessRole                         | AWS Service: ec2, and 1 more                      | -              |
| AWSServiceRoleForAmazonSSM                     | AWS Service: ssm (Service-Linked Role)            | 42 minutes ago |
| AWSServiceRoleForAPIGateway                    | AWS Service: ops.apigateway (Service-Linked Role) | -              |

First, select the **AWS Service** option on the **Trusted entity type** section and **EC2** as the service assigned for this policy.

The screenshot shows the 'Create role' wizard, Step 1: Select trusted entity. The 'AWS service' option is selected under 'Trusted entity type'. Under 'Use case', 'EC2' is selected as the service, and 'EC2' is selected as the use case.

**Trusted entity type**

- ☒ **AWS service**  
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- ☐ **AWS account**  
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- ☐ **Web identity**  
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- ☐ **SAML 2.0 federation**  
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- ☐ **Custom trust policy**  
Create a custom trust policy to enable others to perform actions in this account.

**Use case**  
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

**Service or use case**  
EC2

**Choose a use case for the specified service.**

**Use case**

- ☒ **EC2**  
Allows EC2 instances to call AWS services on your behalf.
- ☐ **EC2 Role for AWS Systems Manager**  
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- ☐ **EC2 Spot Fleet Role**  
Allows EC2 Spot Fleet to request and terminate Spot instances on your behalf.
- ☐ **EC2 - Spot Fleet Auto Scaling**  
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
- ☐ **EC2 - Spot Fleet Tagging**  
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
- ☐ **EC2 - Spot Instances**  
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
- ☐ **EC2 - Spot Fleet**  
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
- ☐ **EC2 - Scheduled Instances**  
Allows EC2 Scheduled Instances to manage instances on your behalf.

**Next**

When finished, click **Next**:



IAM > Roles > Create role

Step 1

Step 2

Step 3

Step 4

Select trusted entity

Add permissions

Name, review, and create

Add permissions

Info

Permissions policies (1062)

Info

Choose one or more policies to attach to your new role.

Filter by Type

All types

Search

1 2 3 4 5 6 7 ... 54

| <input type="checkbox"/> | Policy name   | Type                       | Description                                 |
|--------------------------|---|----------------------------|---|
| <input type="checkbox"/> | <a href="#">AdministratorAccess</a>                         | AWS managed - job function | Provides full access to AWS services an...  |
| <input type="checkbox"/> | <a href="#">AdministratorAccess-Amplify</a>                 | AWS managed                | Grants account administrative permis...     |
| <input type="checkbox"/> | <a href="#">AdministratorAccess-AWSElasticBeanstalk</a>     | AWS managed                | Grants account administrative permis...     |
| <input type="checkbox"/> | <a href="#">AIOpsAssistantPolicy</a>                        | AWS managed                | Provides ReadOnly permissions requir...     |
| <input type="checkbox"/> | <a href="#">AIOpsConsoleAdminPolicy</a>                     | AWS managed                | Grants full access to Amazon AI Opera...    |
| <input type="checkbox"/> | <a href="#">AIOpsOperatorAccess</a>                         | AWS managed                | Grants access to the Amazon AI Opera...     |
| <input type="checkbox"/> | <a href="#">AIOpsReadOnlyAccess</a>                         | AWS managed                | Grants ReadOnly permissions to the A...     |
| <input type="checkbox"/> | <a href="#">AlexaForBusinessDeviceSetup</a>                 | AWS managed                | Provide device setup access to AlexaFo...   |
| <input type="checkbox"/> | <a href="#">AlexaForBusinessFullAccess</a>                  | AWS managed                | Grants full access to AlexaForBusiness ...  |
| <input type="checkbox"/> | <a href="#">AlexaForBusinessGatewayExecution</a>            | AWS managed                | Provide gateway execution access to A...    |
| <input type="checkbox"/> | <a href="#">AlexaForBusinessLifesizeDelegatedAccessP...</a> | AWS managed                | Provide access to Lifesize AVS devices      |
| <input type="checkbox"/> | <a href="#">AlexaForBusinessPolyDelegatedAccessPolicy</a>   | AWS managed                | Provide access to Poly AVS devices          |
| <input type="checkbox"/> | <a href="#">AlexaForBusinessReadOnlyAccess</a>              | AWS managed                | Provide read only access to AlexaForB...    |
| <input type="checkbox"/> | <a href="#">AmazonAPIGatewayAdministrator</a>               | AWS managed                | Provides full access to create/edit/dele... |
| <input type="checkbox"/> | <a href="#">AmazonAPIGatewayInvokeFullAccess</a>            | AWS managed                | Provides full access to invoke APIs in A... |
| <input type="checkbox"/> | <a href="#">AmazonAPIGatewayPushToCloudWatchLogs</a>        | AWS managed                | Allows API Gateway to push logs to us...    |
| <input type="checkbox"/> | <a href="#">AmazonAppFlowFullAccess</a>                     | AWS managed                | Provides full access to Amazon AppFlo...    |
| <input type="checkbox"/> | <a href="#">AmazonAppFlowReadOnlyAccess</a>                 | AWS managed                | Provides read only access to Amazon A...    |
| <input type="checkbox"/> | <a href="#">AmazonAppStreamFullAccess</a>                   | AWS managed                | Provides full access to Amazon AppStr...    |
| <input type="checkbox"/> | <a href="#">AmazonAppStreamPCAAccess</a>                    | AWS managed                | Amazon AppStream 2.0 access to AWS...       |

Set permissions boundary - optional

Cancel

Previous

Next

Finally, set the **Role Name** and click the **Create Role** button.

Step 1: Select trusted entity

Step 2: Add permissions

Step 3: Name, review, and create

### Name, review, and create

**Role details**

**Role name**  
Enter a meaningful name to identify this role.

route-table-change

Maximum 64 characters. Use alphanumeric and "+,=,\_,@,-" characters.

**Description**  
Add a short explanation for this role.

Allows EC2 instances to make changes on the route table

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: \_+=, @-/\[\]{}#%\*!~:;,"'

**Step 1: Select trusted entities** [Edit](#)

**Trust policy**

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
8       ],
9       "Principal": {
10        "Service": [
11          "ec2.amazonaws.com"
12        ]
13      }
14    }
15  ]
16 }

```

**Step 2: Add permissions** [Edit](#)

**Permissions policy summary**

| Policy name | Type | Attached as |
|-------------|------|-------------|
|             |      |             |

**Step 3: Add tags**

**Add tags - optional** [Info](#)

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tags.

[Cancel](#) [Previous](#) [Create role](#)

## Step 5. Create and Attach a Trust Policy to an IAM Role

Once the Role is created, a Trust Policy must be made to acquire the skill of modifying the AWS routing tables when needed. Move to the **Policies** section on the IAM dashboard. Click **Create Policy** button. This process consists of 2 steps:

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles
- Policies**
- Identity providers
- Account settings
- Root access management [New](#)

Access reports

- Access Analyzer

### Policies (1367) [Info](#)

A policy is an object in AWS that defines permissions.

Filter by Type

Search

All types

< 1 2 3 4 5 6 7 ... 69 > [Settings](#)

|                       | Policy name   | Type                       | Used as          |
|-----------------------|---|----------------------------|------------------|
| <input type="radio"/> | <a href="#">AccessAnalyzerServiceRolePolicy</a>         | AWS managed                | None             |
| <input type="radio"/> | <a href="#">AdministratorAccess</a>                     | AWS managed - job function | Permissions poli |
| <input type="radio"/> | <a href="#">AdministratorAccess-Amplify</a>             | AWS managed                | None             |
| <input type="radio"/> | <a href="#">AdministratorAccess-AWSElasticBeanstalk</a> | AWS managed                | None             |
| <input type="radio"/> | <a href="#">AIOpsAssistantPolicy</a>                    | AWS managed                | None             |
| <input type="radio"/> | <a href="#">AIOpsConsoleAdminPolicy</a>                 | AWS managed                | None             |
| <input type="radio"/> | <a href="#">AIOpsOperatorAccess</a>                     | AWS managed                | None             |
| <input type="radio"/> | <a href="#">AIOpsReadOnlyAccess</a>                     | AWS managed                | None             |

[Create policy](#)

First, make sure that the **Policy Editor** is using **JSON** and apply the commands that are shown below. Once

configured, click **Next**:

Step 1: Specify permissions

Step 2: Review and create

### Specify permissions

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

**Policy editor**

Visual | **JSON** | Actions

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Action": [  
7         "ec2:AssociateRouteTable",  
8         "ec2:CreateRoute",  
9         "ec2:CreateRouteTable",  
10        "ec2:DeleteRoute",  
11        "ec2:DeleteRouteTable",  
12        "ec2:DescribeRouteTables",  
13        "ec2:DescribeVpcs",  
14        "ec2:ReplaceRoute",  
15        "ec2:DisassociateRouteTable",  
16        "ec2:ReplaceRouteTableAssociation"  
17      ],  
18      "Resource": "*"   
19    }  
20  ]  
21 }
```

+ Add new statement

JSON Ln 21, Col 1

5825 of 6144 characters remaining

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

Edit statement

Select a statement

Select an existing statement in the policy or add a new statement.

+ Add new statement

Cancel **Next**

This is the text code used in the image:

```
{  
"Version": "2012-10-17",  
"Statement": [  
{  
"Effect": "Allow",  
"Action": [  
"ec2:AssociateRouteTable",  
"ec2:CreateRoute",  
"ec2:CreateRouteTable",  
"ec2:DeleteRoute",  
"ec2:DeleteRouteTable",  
"ec2:DescribeRouteTables",  
"ec2:DescribeVpcs",  
"ec2:ReplaceRoute",  
"ec2:DisassociateRouteTable",  
"ec2:ReplaceRouteTableAssociation"  
],  
"Resource": "*"   
},  
],  
}
```

Later, set the **Policy Name** and click **Create Policy**.

IAM > Policies > Create policy

Step 1  
Specify permissions

Step 2  
**Review and create**

### Review and create

Review the permissions, specify details, and tags.

#### Policy details

**Policy name**  
Enter a meaningful name to identify this policy.

C8000v-HA

Maximum 128 characters. Use alphanumeric and '+', '@', '-', '\_' characters.

**Description - optional**  
Add a short explanation for this policy.

Allows EC2 instances to make route changes on AWS

Maximum 1,000 characters. Use alphanumeric and '+', '@', '-', '\_' characters.

#### Permissions defined in this policy

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

Search

**Allow (1 of 441 services)** Show remaining 440 services

| Service | Access level         | Resource      | Request condition |
|---------|----------------------|---------------|-------------------|
| EC2     | Limited: List, Write | All resources | None              |

#### Add tags - optional

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous **Create policy**

Once the policy is created, filter and select the **policy** then click **Attach** option on the **Actions** drop down menu.

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles
- Policies**

### Policies (1/1368)

A policy is an object in AWS that defines permissions.

Filter by Type: All types 1 match

Search: C800

| Policy name      | Type             | Used as | Description                              |
|------------------|------------------|---------|--|
| <b>C8000v-HA</b> | Customer managed | None    | Allows EC2 instances to make route ch... |

Actions: Attach, Detach

A new window is open. In the IAM Entities section, filter and select the **IAM Role** created and click **Attach policy**.

Attach as a permissions policy

To define permissions for an IAM identity (user, user group, or role), attach a policy to it.

#### IAM Entities (1/69)

Entities are IAM users, user groups and roles.

Filter by Entity type: All types 1 match

Search: route

| Entity name               | Entity type |
|---------------------------|-------------|
| <b>route-table-change</b> | Roles       |

Cancel **Attach policy**

## Step 6. Configure and Launch the C8000v Instances

Each C8000v router is going to have 2 interfaces (1 public, 1 private) and is going to be created on its own Availability Zone.

On the **EC2 Dashboard**, click **Launch Instances**:

The screenshot shows the AWS Management Console EC2 Dashboard. The left sidebar contains the navigation menu with the following items: EC2, Dashboard, EC2 Global View, Events, Instances (expanded), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and AMI Catalog. The 'Launch instance' button is highlighted with a red box. The main content area shows the 'Resources' section with a table of EC2 resources: Instances (running) 1, Dedicated Hosts 0, Key pairs 17, Security groups 19, Auto Scaling Groups, Elastic IPs, Load balancers, and Snapshots. Below this is the 'Launch instance' section with a description and a 'Launch instance' button highlighted with a red box. A note at the bottom states: 'Note: Your instances will launch in the United States (N. Virginia) Region'.

Filter the AMI database with the name **Cisco Catalyst 8000v for SD-WAN & Routing**. On the **AWS Marketplace AMIs** list, click **Select**.

The screenshot shows the 'Choose an Amazon Machine Image (AMI)' page. The search bar contains the text 'Cisco Catalyst 8000v for SD-WAN & Routing'. Below the search bar, there are four tabs: 'Quick Start AMIs (0)', 'My AMIs (691)', 'AWS Marketplace AMIs (1)', and 'Community AMIs (500)'. The 'AWS Marketplace AMIs (1)' tab is selected. Below the tabs, there is a list of results. The first result is 'Cisco Catalyst 8000v for SD-WAN & Routing' by Cisco Systems, Inc. The 'Select' button is highlighted with a red box. The page also includes a 'Refine results' section on the left with filters for Categories, Publisher, Pricing model, Operating system, and Architecture.

Select the corresponding **size** for the AMI. For this example, the **c5n.large** size is selected. This can depend on the capacity required for your network. Once selected, click **Subscribe now**.

Published by me | AWS & trusted third-party AMIs | Published by anyone

### Cisco Catalyst 8000V for SD-WAN & Routing

Cisco Systems, Inc. [0 AWS reviews](#)  
[Bring Your Own License](#)

Overview | Product details | **Pricing** | Usage | Support

**Bring Your Own License**  
Available for customers with current licenses purchased via other channels.

|  |                          |
|--|--------------------------|
| ▶ Cisco Catalyst 8000V for SD-WAN & Routing<br>EC2 - c5n.large <i>vendor recommended</i> | \$0/Hour<br>\$0.108/Hour |
|--|--------------------------|

▶ EBS volume

Cancel [Subscribe on instance launch](#) [Subscribe now](#)

## Step 6.1. Configure the Key Pair for Remote Access

Once subscribed to the AMI, a new window with multiple options is displayed. On the **Key pair (login)** section, if a keypair is not present, click **Create new key pair**. You can reuse a single key for every device created.

▼ **Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

fsimmond-pem ▼ [Create new key pair](#)

A new pop-up window is displayed. For this example, a .pem key file with ED25519 encryption is created. Once everything is set, click **Create key pair**.

## Create key pair

✕

### Key pair name

Key pairs allow you to connect to your instance securely.

fsimmond-ed-pem

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

### Key pair type



☐ RSA  
RSA encrypted private and public key pair

☒ ED25519  
ED25519 encrypted private and public key pair

### Private key file format

☒ .pem  
For use with OpenSSH

☐ .ppk  
For use with PuTTY

 When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#) 

Cancel

Create key pair

## Step 6.2. Create and Configure the Subnets for the AMI

On the **Network Settings** section, click **Edit**. Some new options inside the section are now available:

1. Select the desired **VPC** for this work. For this example, the VPC called **HA** is selected.
2. On the **Firewall (security groups)** section, select **Select existing security group**.
3. Once option 2 is selected, the **Common security groups** option is available. Filter and select the desired security group. For this example, the **All traffic HA** security group is selected.
4. (Optional) If no subnets for these device are created, click the **Create new subnet**.

**▼ Network settings** [Info](#)

**VPC - required** [Info](#)

vpc-0d30b9fa9511f3639 (HA)  
10.100.0.0/16

**Subnet** [Info](#)

subnet-0b664f8e74443d28f public-R1-C8000v  
VPC: vpc-0d30b9fa9511f3639 Owner: 073713984176 Availability Zone: us-east-1a  
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.100.10.0/24

**Auto-assign public IP** [Info](#)

Disable

**Firewall (security groups)** [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group ☒ Select existing security group

**Common security groups** [Info](#)

Select security groups

All traffic HA sg-029461ba80052f10c X  
VPC: vpc-0d30b9fa9511f3639

Security groups that you add or remove here will be added to or removed from all your network interfaces.

**► Advanced network configuration**

A new tab on the web browser is open, leading you to the **Create subnet** section:

1. Select the corresponding **VPC** for this configuration from the drop-down list.
2. Set a **name** for the new subnet.
3. Define the **Availability Zone** for this subnet. (Please refer to the Topology section of this document for more information of the setting)
4. Set the **subnet block** that belongs on the VPC CIDR block.
5. Additionally, all the subnets that are going to be used can be created by clicking the **Add new subnet** section and repeat the steps from 2 to 4 for each subnet.
6. Once finished, click **Create subnet**. Navigate to the previous page to continue with the settings.



☰ VPC > Subnets > Create subnet

### Create subnet Info

**VPC**

VPC ID

vpce-0d30b9fa9511f3639 (HA) 1

Associated VPC CIDRs

IPv4 CIDRs

10.100.0.0/16

**Subnet settings**

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

**Subnet name**

Associate a tag with a key or "Name" and a value that you specify.

public-R1-C8000v 2

The name can be up to 256 characters long.

**Availability Zone** Info

Select the Availability Zone for your subnet, or let Amazon choose one for you.

United States (N. Virginia) / us-east-1a 3

**IPv4 VPC CIDR block** Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

10.100.0.0/16

**IPv4 subnet CIDR block**

10.100.10.0/24 4 256 IPs

< > ^ v

▼ **Tags - optional**

| Key    | Value - optional   |        |
|--------|--------------------|--------|
| Q Name | Q public-R1-C8000v | Remove |

[Add new tag](#)

You can add 49 more tags.

[Remove](#)

[Add new subnet](#) 5

6

[Cancel](#) [Create subnet](#)

On the **Subnet** subsection from the **Network Settings** section, click **Refresh** icon to get the created subnets on the drop-down list.

### Step 6.3. Configure the AMI Interfaces

On the **Network Settings** section, expand the **Advanced Network configuration** subsection. These options are displayed:

▼ Advanced network configuration

### Network interface 1

Device index | Info  
0

Network interface | Info  
New interface ▼

Description | Info  
Public-R1

Subnet | Info  
subnet-0b664f8e74443d28f  
IP addresses available: 249

Security groups | Info  
Select security groups ▼

Auto-assign public IP | Info  
Disable ▼

Primary IP | Info  
10.100.10.254

Secondary IP | Info  
Select ▼

IPv6 IPs | Info  
Select ▼  
The selected subnet does not support IPv6 IPs.

IPv4 Prefixes | Info  
Select ▼

IPv6 Prefixes | Info  
Select ▼  
The selected subnet does not support IPv6 prefixes because it does not have an IPv6 CIDR.

Assign Primary IPv6 IP | Info  
Select ▼  
A primary IPv6 address is only compatible with subnets that support IPv6.

Delete on termination | Info  
No ▼

Interface type | Info  
Select ▼  
The selected instance type does not support multiple network cards.

Network card index | Info  
Select ▼

ENA Express | Info  
Select ▼  
The selected instance type does not support ENA Express.

ENA Express UDP | Info  
Select ▼  
The selected instance type does not support ENA Express.

ENA queues | Info  
Select ▼  
The selected instance type does not support ENA queues.

Idle connection tracking timeout | Info  
☐ Enable

Add network interface

On this menu, set the **Description**, **Primary IP**, **Delete on termination** parameters. For the **Primary IP** parameter, use any IP address except for the first available address of the subnet. This is used internally by AWS.

The **Delete on termination** parameter on this example is set as **No**. However, this can be set to **yes** depending of your environment.

Due to this topology, a second interface is needed for the private subnet. Click **Add network interface** and this prompt is displayed. However, the interface provides the option to select the subnet this time:

Network interface 2

Remove

Device index | Info  
1

Network interface | Info  
New interface ▼

Description | Info  
Private-R1

Subnet | Info  
subnet-0a5f13361443951d2 ▼  
IP addresses available: 250

Security groups | Info  
Select security groups ▼

Auto-assign public IP | Info  
Select ▼

Primary IP | Info  
10.100.110.254

Secondary IP | Info  
Select ▼

IPv6 IPs | Info  
Select ▼  
The selected subnet does not support IPv6 IPs.

Once all the parameters are set as was made on the Network Interface 1, continue with the next steps.

## Step 6.4. Set the IAM Instance Profile to the AMI

Under the **Advanced details** section, select the created IAM role on the **IAM instance profile** parameter:

▼ **Advanced details** [Info](#)

Domain join directory | [Info](#)

Select [Create new directory](#)

**IAM instance profile** | [Info](#)

route-table-change  
arn:aws:iam::073713984176:instance-profile/route-table-change [Create new IAM profile](#)

Hostname type | [Info](#)

IP name

DNS Hostname | [Info](#)

- ☒ Enable IP name IPv4 (A record) DNS requests
- ☐ Enable resource-based IPv4 (A record) DNS requests
- ☐ Enable resource-based IPv6 (AAAA record) DNS requests

## Step 6.5. (Optional) Set the Credentials on the AMI

Under the **Advanced details** section, navigate to the **User data - optional** section and apply this setting to set a username and password while the instance is created:

```
ios-config-1="username <username> priv 15 pass <password>"
```



**Note:** The username provided by AWS to SSH into the C8000v can be incorrectly listed as root. Change this to ec2-user if necessary.

---

### Step 6.6. Finish the Instance Configuration

Once everything is configured, click **Launch Instance**:

## ▼ Summary

Number of instances | [Info](#)

1

### Software Image (AMI)

Cisco Catalyst 8000V for SD-WA...[read more](#)

ami-03cc286883c62bdee

### Virtual server type (instance type)


c5n.large

### Firewall (security group)

All traffic HA

### Storage (volumes)

1 volume(s) - 16 GiB

 **Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.



[Cancel](#)

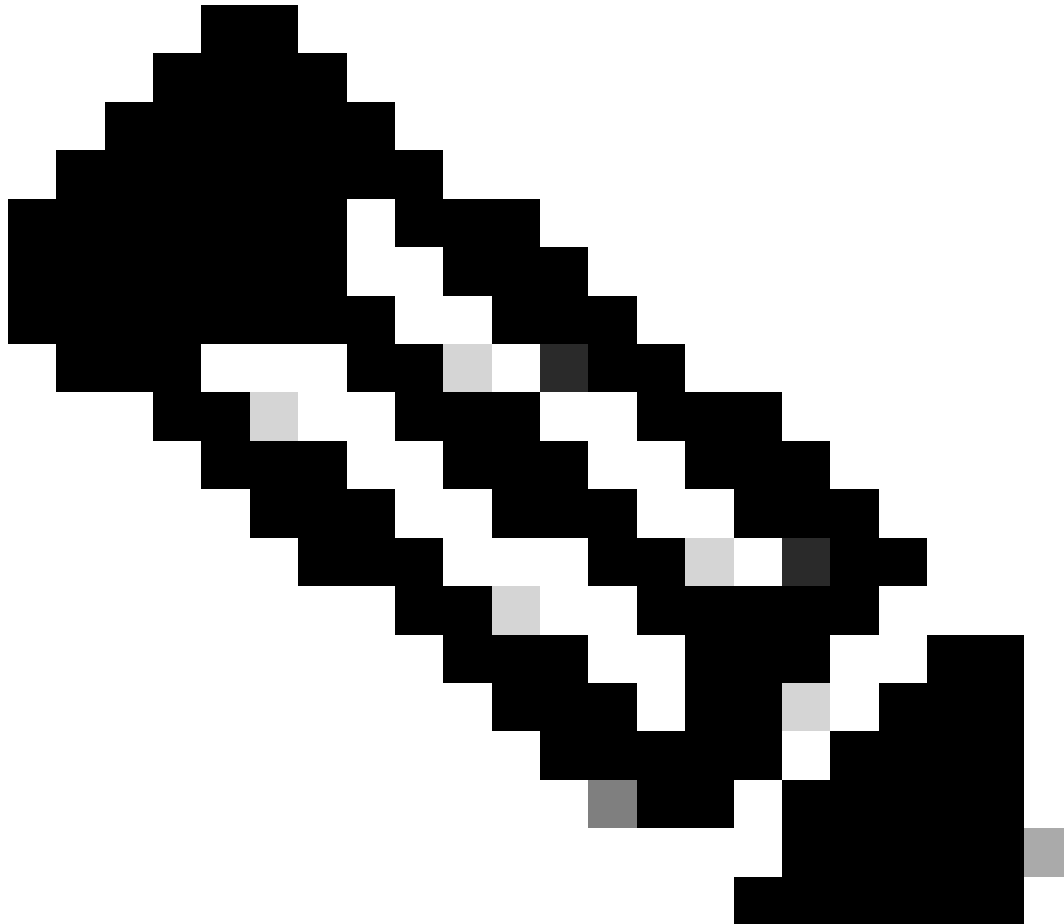
[Launch instance](#)

 [Preview code](#)

### Step 6.7. Disable Source/Destination Check on the ENIs

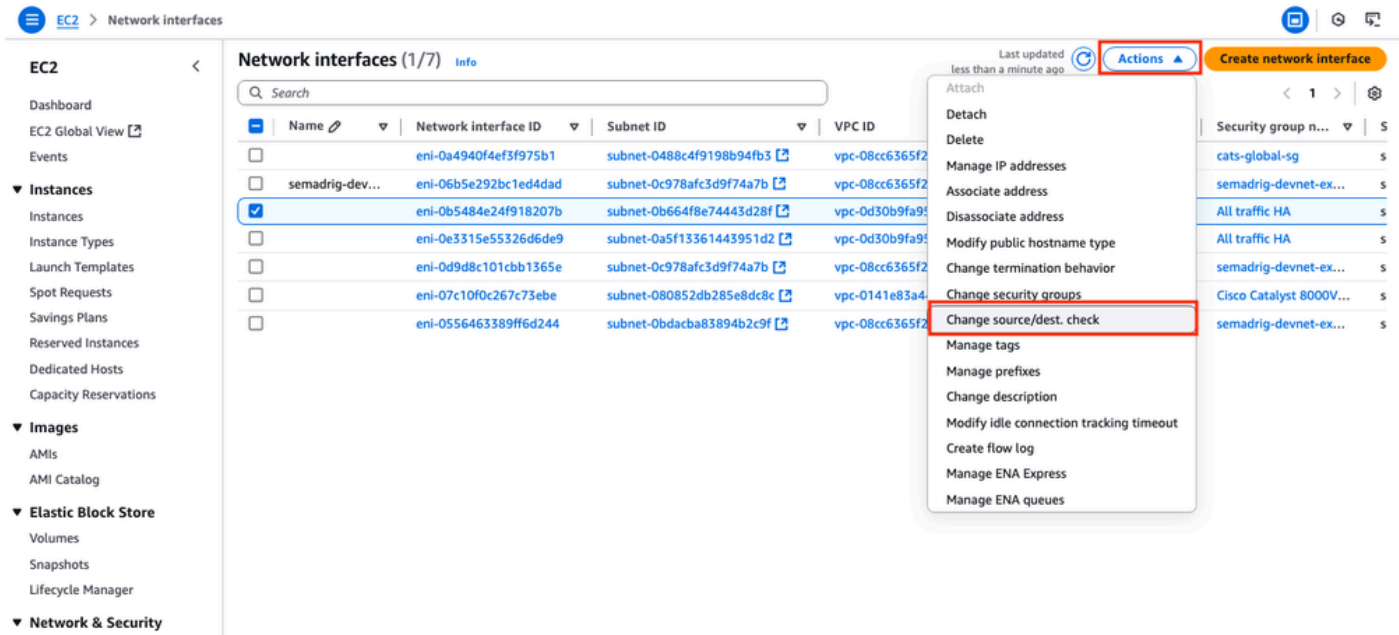
Once the Instance is created, disable the src/dst check functionality on AWS to get the connectivity between interfaces in the same subnet. On the **EC2 Dashboard > Network & Security > Network interfaces** section, select the **ENIs** and click **Actions > Change source/dest. check.**

---

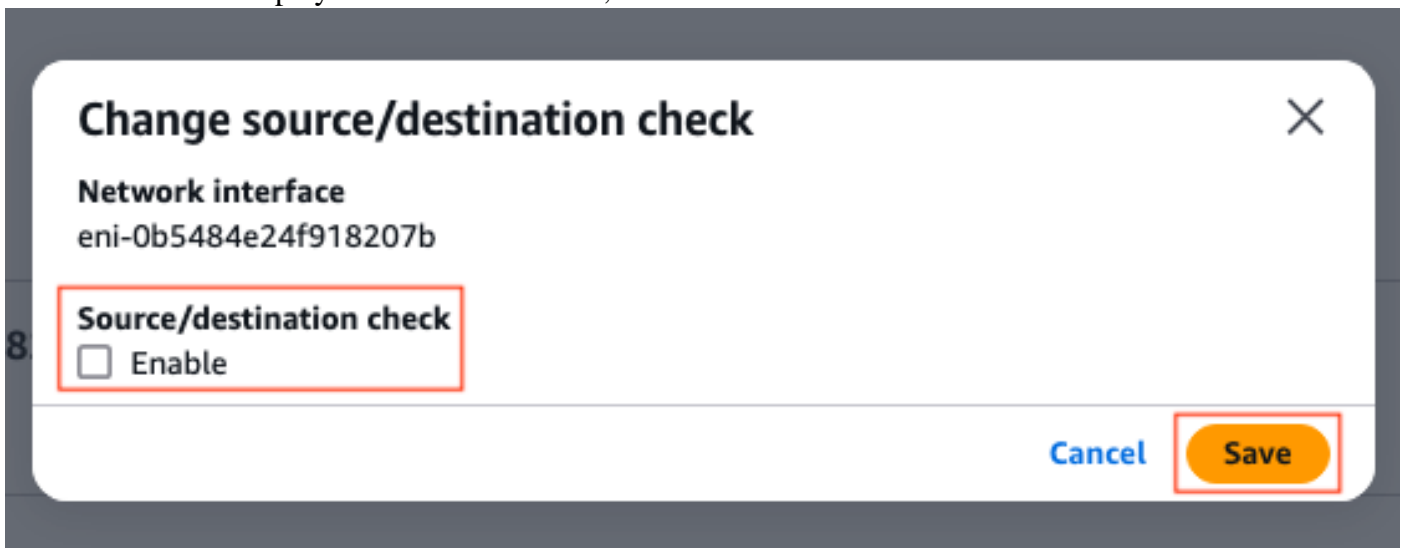


**Note:** You must select the ENIs one by one to get this option available.

---



A new window is displayed. On the new menu, disable the **Enable** checkbox and click **Save**.



## Step 6.8. Create and Associate an Elastic IP to the Public ENI of the Instance

On the **EC2 Dashboard** > **Network & Security** > **Elastic IPs** section, click **Allocate Elastic IP address**.



The page leads you to the another section. For this example, the **Amazon pool of IPv4 addresses** option is selected along with the availability zone **us-east-1**. Once finished, click **Allocate**.

EC2 > Elastic IP addresses > Allocate Elastic IP address

### Allocate Elastic IP address Info

#### Elastic IP address settings Info

**Public IPv4 address pool**

- ☒ Amazon's pool of IPv4 addresses
  - Public IPv4 address that you bring to your AWS account with BYOIP. (option disabled because no pools found) [Learn more](#)
  - Customer-owned pool of IPv4 addresses created from your on-premises network for use with an Outpost. (option disabled because no customer owned pools found) [Learn more](#)
  - Allocate using an IPv4 IPAM pool (option disabled because no public IPv4 IPAM pools with AWS service as EC2 were found)

**Network border group Info**

us-east-1

**Global static IP addresses**

AWS Global Accelerator can provide global static IP addresses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using the Amazon global network. [Learn more](#)

[Create accelerator](#)

**Tags - optional**

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

[Add new tag](#)

You can add up to 50 more tag

[Cancel](#) [Allocate](#)

When the IP address is created, assign the IP address to the public interface of the instance. On the **EC2 Dashboard > Network & Security > Elastic IPs** section, click **Actions > Associate Elastic IP address**.

Elastic IP addresses (1/1) Info

Find elastic IP addresses by attribute or tag

Public IPv4 address [Clear filters](#)

| <input checked="" type="checkbox"/> | Name | Allocated IPv4 addr... | Type      | Allocation ID              | Reverse DNS record |   |
|-------------------------------------|------|------------------------|-----------|----------------------------|--------------------|---|
| <input checked="" type="checkbox"/> | -    | -                      | Public IP | eipalloc-0948346735ab2017c | -                  | <div><div>Actions</div><div><a href="#">View details</a><br/><a href="#">Release Elastic IP addresses</a><br/><a href="#">Associate Elastic IP address</a><br/><a href="#">Disassociate Elastic IP address</a><br/><a href="#">Update reverse DNS</a><br/><a href="#">Enable transfers</a><br/><a href="#">Disable transfers</a><br/><a href="#">Accept transfers</a></div></div> |

In this new section, select the **Network interface** option and look for the public **ENI** of the corresponding interface. Associate the corresponding public IP address and click **Associate**.





**Note:** To get the proper ENI ID, navigate to the **EC2 Dashboard > Instances** section. Then select the instance and check the **Networking** section. Look for the IP address of your public interface to get the ENI value on the same row.

---

## Associate Elastic IP address [Info](#)

Choose the instance or network interface to associate to this Elastic IP address ([See IP address](#))

**Elastic IP address:** [aa-2111-1000-2100](#)

**Resource type**  
Choose the type of resource with which to associate the Elastic IP address.  
☐ Instance  
☒ Network interface

⚠ If you associate an Elastic IP address with an instance that already has an Elastic IP address associated, the previously associated Elastic IP address will be disassociated, but the address will still be allocated to your account. [Learn more](#)

If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address.

**Network interface**

**Private IP address**  
The private IP address with which to associate the Elastic IP address.

**Reassociation**  
Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.  
☒ Allow this Elastic IP address to be reassociated

[Cancel](#) [Associate](#)

## Step 7. Repeat Step 6 to Create the Second C8000v Instance for HA

Please refer to the **Topology** section of this document to have the corresponding information for each interface and repeat the same steps from 6.1 to 6.6.

## Step 8. Repeat Step 6 to Create a VM (Linux/Windows) from the AMI Marketplace

For this example, Ubuntu server 22.04.5 LTS is selected from the AMI Marketplace as the internal host.

ens5 is created by default for the public interface. For this example, create a second interface (ens6 on the device) for the private subnet.

<#root>

```
ubuntu@ip-10-100-30-254:~$ sudo apt install net-tools
```

```
...
```

```
ubuntu@ip-10-100-30-254:~$ ifconfig
```

```
ens5: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
inet
```

```
10.100.30.254
```

```
netmask 255.255.255.0 broadcast 10.100.30.255
inet6 fe80::51:19ff:fea2:1151 prefixlen 64 scopeid 0x20<link>
ether 02:51:19:a2:11:51 txqueuelen 1000 (Ethernet)
RX packets 1366 bytes 376912 (376.9 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 1417 bytes 189934 (189.9 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

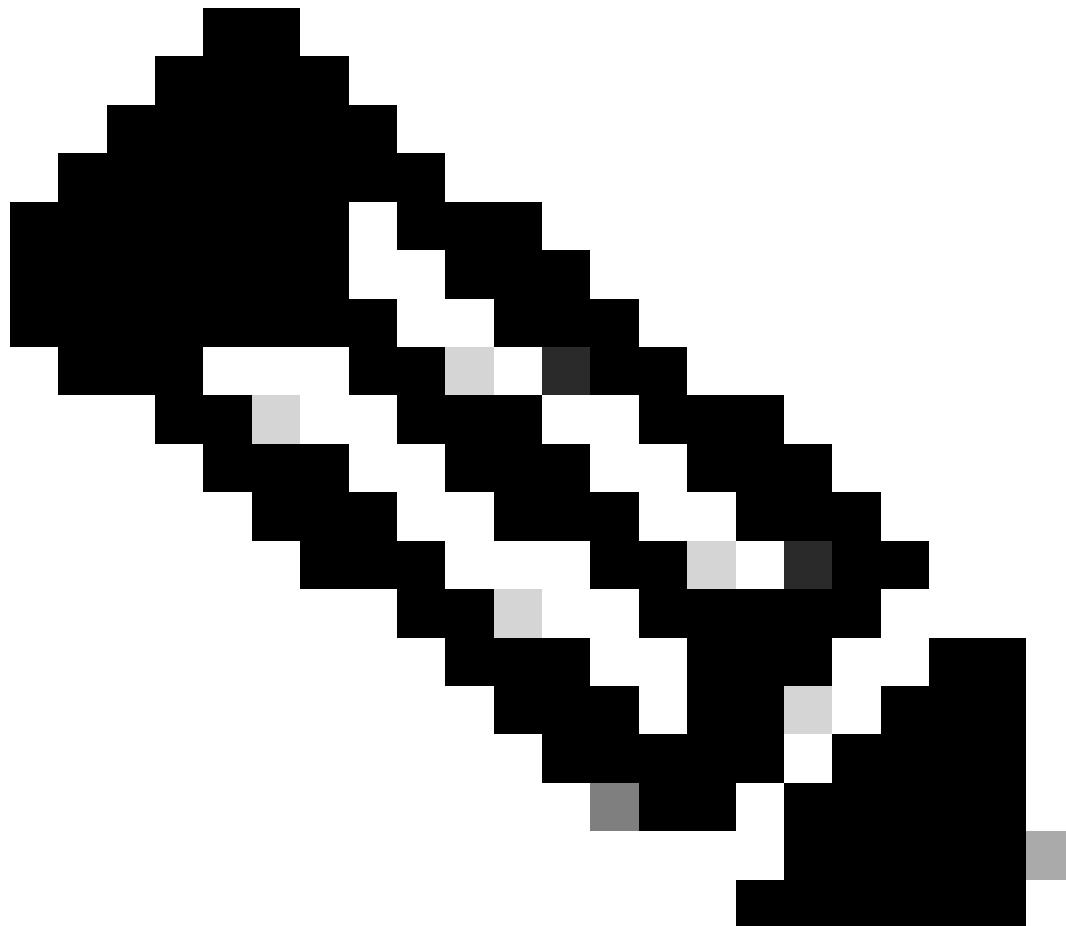
```
ens6: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
inet
```

```
10.100.130.254
```

```
netmask 255.255.255.0 broadcast 10.100.130.255
inet6 fe80::3b:7eff:fead:dbe5 prefixlen 64 scopeid 0x20<link>
```

```
ether 02:3b:7e:ad:db:e5 txqueuelen 1000 (Ethernet)
RX packets 119 bytes 16831 (16.8 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 133 bytes 13816 (13.8 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

---

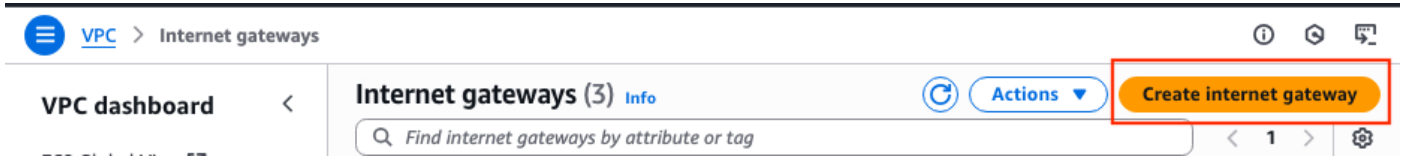


**Note:** If any change is made on the interfaces, flap the interface or reload the VM to get these changes applied.

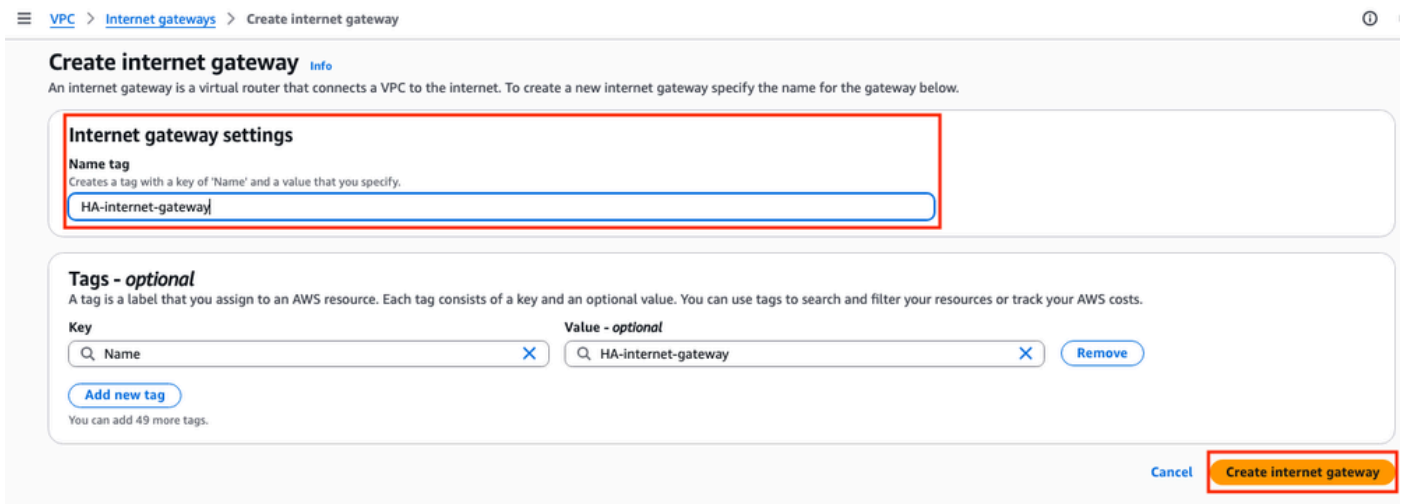
---

## Step 9. Create and Configure an Internet Gateway (IGW) for the VPC

On the **VPC Dashboard > Virtual private cloud > Internet gateways** section, click **Create internet gateway**.



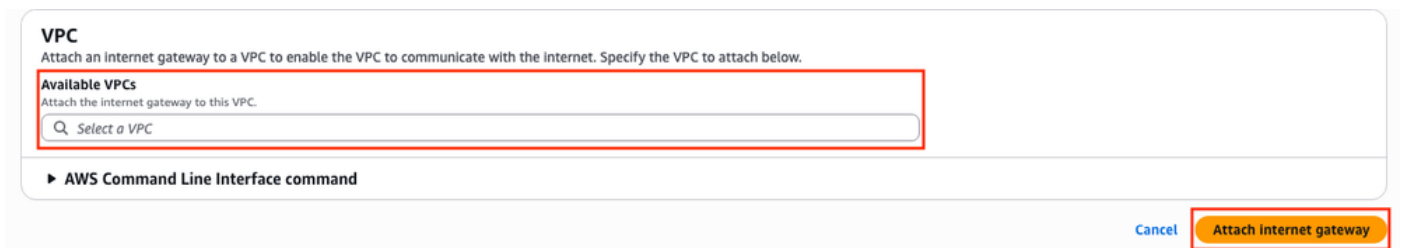
In this new section, create a **name tag** for this gateway and click **Create internet gateway**.



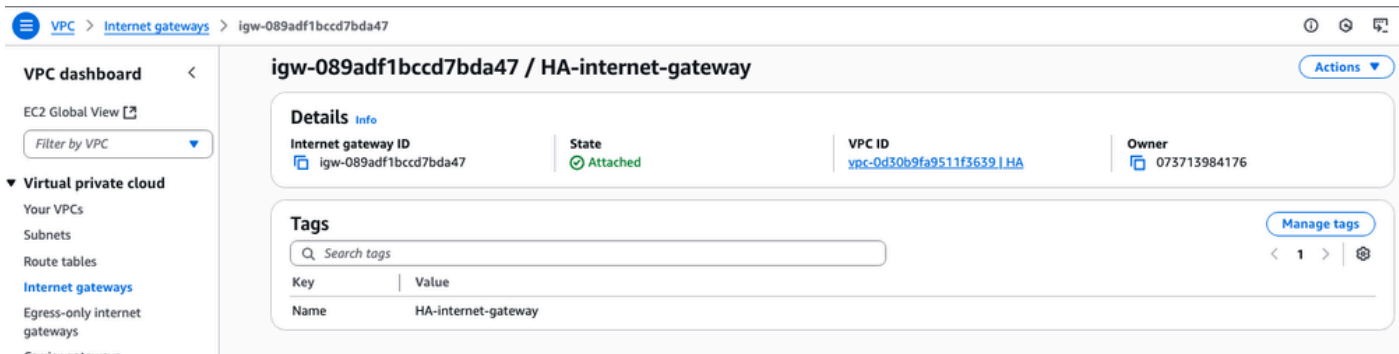
Once the IGW is created, attach it to your corresponding VPC. Navigate to the **VPC Dashboard > Virtual Private Cloud > Internet Gateway** section and select the corresponding IGW. Click **Actions > Attach to VPC**.



In this new section, select the VPC called **HA**. For this example, click **Attach internet gateway**.



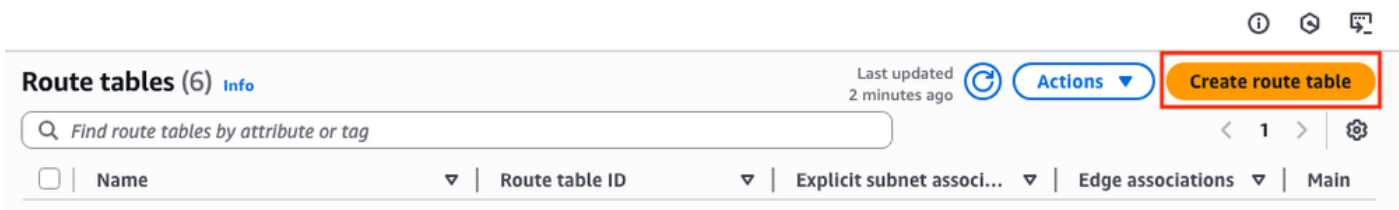
The IGW must indicate the Attached state as it is shown:



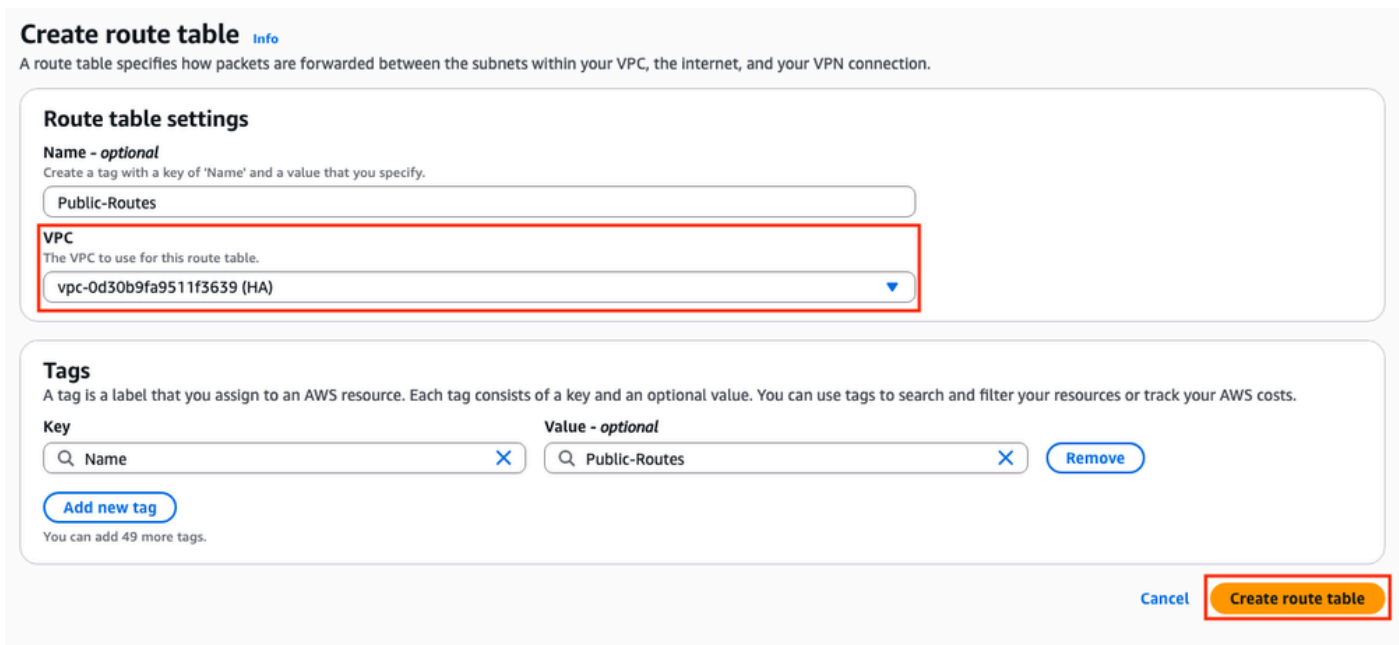
## Step 10. Create and Configure Route Tables on AWS for Public and Private subnets

### Step 10.1. Create and Configure the Public Route Table

In order to establish the HA on this topology, associate all the public and private subnets on their corresponding route tables. In the **VPC Dashboard > Virtual Private Cloud > Route tables** section, click **Create route table**.



In the new section, select the corresponding **VPC** for this topology. Once selected, click **Create route table**.



In the **Route tables** section, select the **created** table and click **Actions > Edit Subnet associations**.

VPC > Route tables

Route tables (1/1) Info

Find route tables by attribute or tag

public-routes X Clear filters

Public-Routes

Route table ID: [rtb-0d0e48f25c9b00635](#)

Explicit subnet associ...: 3 subnets

Actions

- View details
- Set main route table
- Edit subnet associations
- Edit edge associations
- Edit route propagation
- Edit routes
- Manage tags
- Delete route table

Then, select the corresponding **subnets** and click **Save associations**.

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (3/6)

Filter subnet associations

public X Clear filters

| Name                         | Subnet ID                | IPv4 CIDR      | IPv6 CIDR | Route table ID                        |
|------------------------------|--------------------------|----------------|-----------|---------------------------------------|
| public-R1-C8000v             | subnet-0b664f8e74443d28f | 10.100.10.0/24 | -         | rtb-0d0e48f25c9b00635 / Public-Routes |
| Public-VM-linux-1c - fsmmond | subnet-04fb9de939e3778bb | 10.100.30.0/24 | -         | rtb-0d0e48f25c9b00635 / Public-Routes |
| public-R2-C8000v             | subnet-02d8108842f9f3129 | 10.100.20.0/24 | -         | rtb-0d0e48f25c9b00635 / Public-Routes |

Selected subnets

subnet-0b664f8e74443d28f / public-R1-C8000v X subnet-04fb9de939e3778bb / Public-VM-linux-1c - fsmmond X subnet-02d8108842f9f3129 / public-R2-C8000v X

Cancel Save associations

Once the subnets are associated, click the **Route table ID** hyperlink to add the proper routes for the table. Then, click **Edit Routes**:

Route tables (1/1) Info

Find route tables by attribute or tag

public-routes X Clear filters

| Name          | Route table ID                        |
|---------------|---------------------------------------|
| Public-Routes | <a href="#">rtb-0d0e48f25c9b00635</a> |

In order to get Internet access, click **Add route** and link this **public route table** with the **IGW** created on Step 9 with these parameters. Once selected, click **Save changes**:

## Edit routes

| Destination   | Target           | Status | Propagated |
|---------------|------------------|--------|------------|
| 10.100.0.0/16 | local            | Active | No         |
| 0.0.0.0/0     | Internet Gateway | Active | No         |

Buttons: Add route, Cancel, Preview, Save changes

## Step 10.2. Create and Configure the Private Route Table

Now that the public route table is created, replicate Steps 10 for the private route and private subnets except for the addition of the Internet Gateway on its routes. For this example, the routing table looks like this since the traffic for 8.8.8.8 must go through the private subnet in this example:

## Edit routes

| Destination   | Target            | Status | Propagated |
|---------------|-------------------|--------|------------|
| 10.100.0.0/16 | local             | Active | No         |
| 8.8.8.8/32    | Network Interface | -      | No         |

Buttons: Add route, Cancel, Preview, Save changes

## Step 11. Check and Configure Basic Network Configuration, Network Address Translation (NAT), GRE Tunnel with BFD and Routing Protocol

Once the Instances and its routing configuration on AWS is prepared, configure the devices:

C8000v R1 configuration:

```
interface Tunnel1
ip address 192.168.200.1 255.255.255.0
bfd interval 500 min_rx 500 multiplier 3
tunnel source GigabitEthernet1
tunnel destination <Public IPv4 address of C8000v R2>
!
interface GigabitEthernet1
ip address 10.100.10.254 255.255.255.0
ip nat outside
negotiation auto
!
interface GigabitEthernet2
ip address 10.100.110.254 255.255.255.0
ip nat inside
negotiation auto
!
router eigrp 1
bfd interface Tunnel1
network 192.168.200.0
passive-interface GigabitEthernet1
!
ip access-list standard 10
10 permit 10.100.130.0 0.0.0.255
```

```

!
ip nat inside source list 10 interface GigabitEthernet1 overload
!
ip route 0.0.0.0 0.0.0.0 GigabitEthernet1 10.100.10.1
ip route 10.100.130.0 255.255.255.0 GigabitEthernet2 10.100.110.1

```

C8000v R2 configuration:

```

interface Tunnel1
ip address 192.168.200.2 255.255.255.0
bfd interval 500 min_rx 500 multiplier 3
tunnel source GigabitEthernet1
tunnel destination<Public IPv4 address of C8000v R1>
!
interface GigabitEthernet1
ip address 10.100.20.254 255.255.255.0
ip nat outside
negotiation auto
!
interface GigabitEthernet2
ip address 10.100.120.254 255.255.255.0
negotiation auto
!
router eigrp 1
bfd interface Tunnel1
network 192.168.200.0
passive-interface GigabitEthernet1
!
ip route 0.0.0.0 0.0.0.0 GigabitEthernet1 10.100.20.1
ip route 10.100.130.0 255.255.255.0 GigabitEthernet2 10.100.120.1
!
ip access-list standard 10
10 permit 10.100.130.0 0.0.0.255
!
ip nat inside source list 10 interface GigabitEthernet1 overload
!

ip route 0.0.0.0 0.0.0.0 GigabitEthernet1 10.100.20.1
ip route 10.100.130.0 255.255.255.0 GigabitEthernet2 10.100.120.1

```

## Step 12. Configure High Availability (Cisco IOS® XE Denali 16.3.1a or later)

Now that the redundancy and connection among VMs are set, configure the HA settings to define the routing changes. Set the **Route-table-id**, **Network-interface-id** and **CIDR** values that must be set after an AWS HA error such as BFD peer down.

```

Router(config)# redundancy
Router(config-red)# cloud provider aws (node-id)
bfd peer <IP address of the remote device>
route-table <Route table ID>
cidr ip <traffic to be monitored/prefix>
eni <Elastic network interface (ENI) ID>
region <region-name>

```

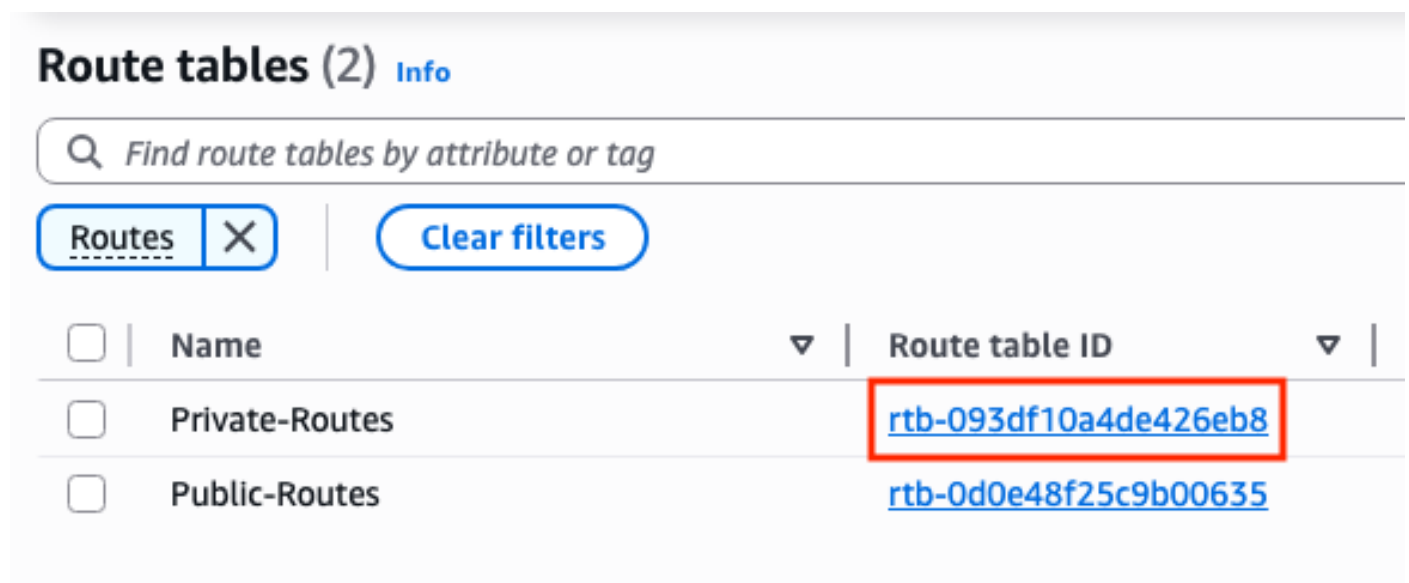


The **bfd peer** parameter is related to the Tunnel peer IP address. This can be checked using the **show bfd neighbor** output:

```
R1(config)#do sh bfd neighbors
```

```
IPv4 Sessions
NeighAddr LD/RD RH/RS State Int
192.168.200.2 4097/4097 Up Up Tu1
```

The **route-table** parameter is related to the Private Route Table ID located in the **VPC Dashboard > Virtual Private Cloud > Route Tables** section. Copy the corresponding **Route Table ID**.



**Route tables (2)** [Info](#)

**Routes**

| <input type="checkbox"/> | Name           | Route table ID                        |
|--------------------------|----------------|---------------------------------------|
| <input type="checkbox"/> | Private-Routes | <a href="#">rtb-093df10a4de426eb8</a> |
| <input type="checkbox"/> | Public-Routes  | <a href="#">rtb-0d0e48f25c9b00635</a> |

The **cidr ip** parameter is related with the route prefix added on the Private Route Table (routes created on Step 10.2):

# rtb-093df10a4de426eb8 / Private-Routes

## Details [Info](#)

### Route table ID

 rtb-093df10a4de426eb8

### VPC

 vpc-0d30b9fa9511f3639 | HA

### Main

 Yes

### Owner ID

 073713984176

## Routes

## Subnet associations

## Edge associations

## Route propagation

## Tags

## Routes (2)



### Destination



### Target

8.8.8.8/32

 eni-0239fda341b4d7e41 


10.100.0.0/16


local


The **eni** parameter is related with the ENI ID of the corresponding private interface of the Instance that is being configured. For this example, the ENI ID of the interface GigabitEthernet2 of the instance is used:


Instances (1/3) [Info](#)


Last updated 1 minute ago

 [Connect](#)


[Instance state](#) 





















[Actions](#) 

[Launch instances](#) 



[fsimmond](#) 

[Clear filters](#)

< 1 > 

|  | Name  | Instance ID                         | Instance state   | Instance type  | Status check  | Alarm status  | Availability Zone |
|---|--|-------------------------------------|---|---|---|---|-------------------|
|  | C8000v-R2-fsimmond   | <a href="#">i-0a1a91794f919f641</a> |  Stopped   | c5n.large   | -   | <a href="#">View alarms</a>  | us-east-1b        |
|  | Ubuntu VM - fsimmond   | <a href="#">i-03a306e81a0b99864</a> |  Stopped   | m5.large  | -   | <a href="#">View alarms</a>  | us-east-1c        |
|  | C8000v-R1-fsimmond   | <a href="#">i-0b9a50a09b089b03a</a> |  Running   | c5n.large   |  3/3 checks passed | <a href="#">View alarms</a>  | us-east-1a        |

i-0b9a50a09b089b03a (C8000v-R1-fsimmond)

Details

Status and alarms



Monitoring



Security


**Networking**

Storage

Tags

VPC ID  
 [vpc-0d30b9fa9511f3639](#) (HA) 

Subnet ID  
 [subnet-0b664f8e74443d28f](#) (public-R1-C8000v) 




Availability zone  
 [us-east-1a](#)

Outpost ID  
-

► IP addresses [Info](#)

► Hostname and DNS [Info](#)

▼ Network Interfaces (2) [Info](#)

| Interface ID  | Device Index | Card Index | Description | Public IPv4 address  | Private IPv4 address | Private IPv4 DNS | IPv6 |
|---|--------------|------------|-------------|--|----------------------|------------------|------|
|  <a href="#">eni-0645a881c13823696</a> | 0            | 0          | -           |  <a href="#">10.100.100.254</a> | 10.100.10.254        | -                | -    |
|  <a href="#">eni-070e14fbfde0d8e3b</a> | 1            | 0          | -           | -  | 10.100.110.254       | -                | -    |

The **region** parameter is related with the code name found in the AWS documentation for the region where the VPC is located. For this example, the **us-east-1** region is used.

However, this list can change or grow. To find the latest updates, visit Amazon [Region and Availability Zones](#) document.

Taking all this information into account, here is the configuration example for each router in the VPC:

Configuration Example for C8000v R1:

```
redundancy
cloud provider aws 1
bfd peer 192.168.200.2
route-table rtb-093df10a4de426eb8
cidr ip 8.8.8.8/32
eni eni-070e14fbfde0d8e3b
region us-east-1
```

Configuration Example for C8000v R2:

```
redundancy
cloud provider aws 1
bfd peer 192.168.200.1
route-table rtb-093df10a4de426eb8
cidr ip 8.8.8.8/32
eni eni-0239fda341b4d7e41
region us-east-1
```

## Verification

1. Check that status of the C8000v R1 instance. Confirm that the Tunnel and the Cloud redundancy is up and running.

```
R1#show bfd neighbors
```

```
IPv4 Sessions
NeighAddr LD/RD RH/RS State Int
192.168.200.2 4097/4097 Up Up Tu1
```

```
R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H Address Interface Hold Uptime SRTT RTO Q Seq
(sec) (ms) Cnt Num
0 192.168.200.2 Tu1 10 00:16:52 2 1470 0 2
```

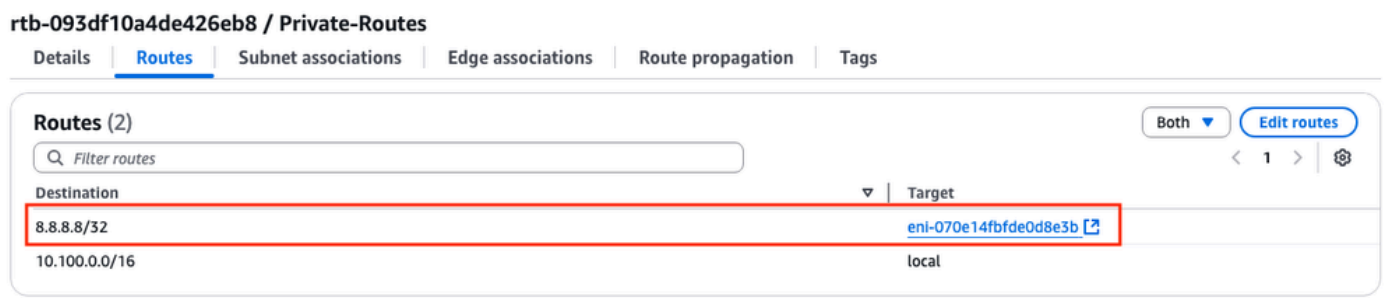
```
R1#show redundancy cloud provider aws 1
Provider : AWS node 1
BFD peer = 192.168.200.2
```

```
BFD intf = Tunnel1
route-table = rtb-093df10a4de426eb8
cidr = 8.8.8.8/32
eni = eni-070e14fbfde0d8e3b
region = us-east-1
```

2. Run a continuous ping to 8.8.8.8 from the Host VM that is behind the routers. Please make sure that the ping is going through the private interface:

```
ubuntu@ip-10-100-30-254:~$ ping -I ens6 8.8.8.8
PING 8.8.8.8 (8.8.8.8) from 10.100.130.254 ens6: 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=1.36 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=1.30 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=114 time=1.34 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=114 time=1.28 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=114 time=1.31 ms
```

3. Open the **AWS WebGUI** and check the **status** of the **routing table**. The current ENI belongs to the private interface of the R1 instance:



4. Trigger the route change by shutting down the **Tunnel1** interface on the **R1 Instance** to simulate a HA failover event:

```
R1#config t
R1(config)#interface tunnel1
R1(config-if)#shut
```

5. Check again at the **route table** on **AWS**, the ENI ID has changed to the R2 private interface ENI ID:



# Troubleshoot

These are most of the common points that are often forgotten/misconfigured while recreating the deployment:

- Ensure resources are associated. When creating VPC, Subnets, Interfaces, Route Tables, and so on, many of these are not associated with each other automatically. They have no knowledge of each other.
- Ensure that the Elastic IP and any Private IP is associated with the correct Interfaces, with the right subnets, added to the correct Route Table, connected to the correct router, and the correct VPC and Zone, linked with the IAM Role and security groups.
- Disable Source/Dest check per ENI.  
In case you already checked all the points discussed in this section and the issue is still present, please gather these outputs, test the HA failover, if possible, and open a case with Cisco TAC:

```
show redundancy cloud provider aws <node-id>
debug redundancy cloud all
debug ip http all
```