AWSでのC8000vハイアベイラビリティコンフィギュレーションのデプロイ

内容

はじめに

このドキュメントでは、Amazon Web Servicesクラウド上のCatalyst 8000vルータでハイアベイラビリティ環境をセットアップする方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- AWSコンソールとそのコンポーネントに関する一般的な知識
- Cisco IOS® XEソフトウェアについて
- HA機能に関する基礎知識

使用するコンポーネント

この設定例では、次のコンポーネントが必要です。

- 管理者ロールを持つAmazon AWSアカウント
- Cisco IOS® XE 17.15.3aを実行するC8000vデバイス2台とUbuntu 22.04 LTS VM 1台 同じリージョン内のAMI

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

トポロジ

HA導入には、ネットワーク要件に基づくさまざまなシナリオがあります。この例では、HA冗長性は次の設定で設定されます。

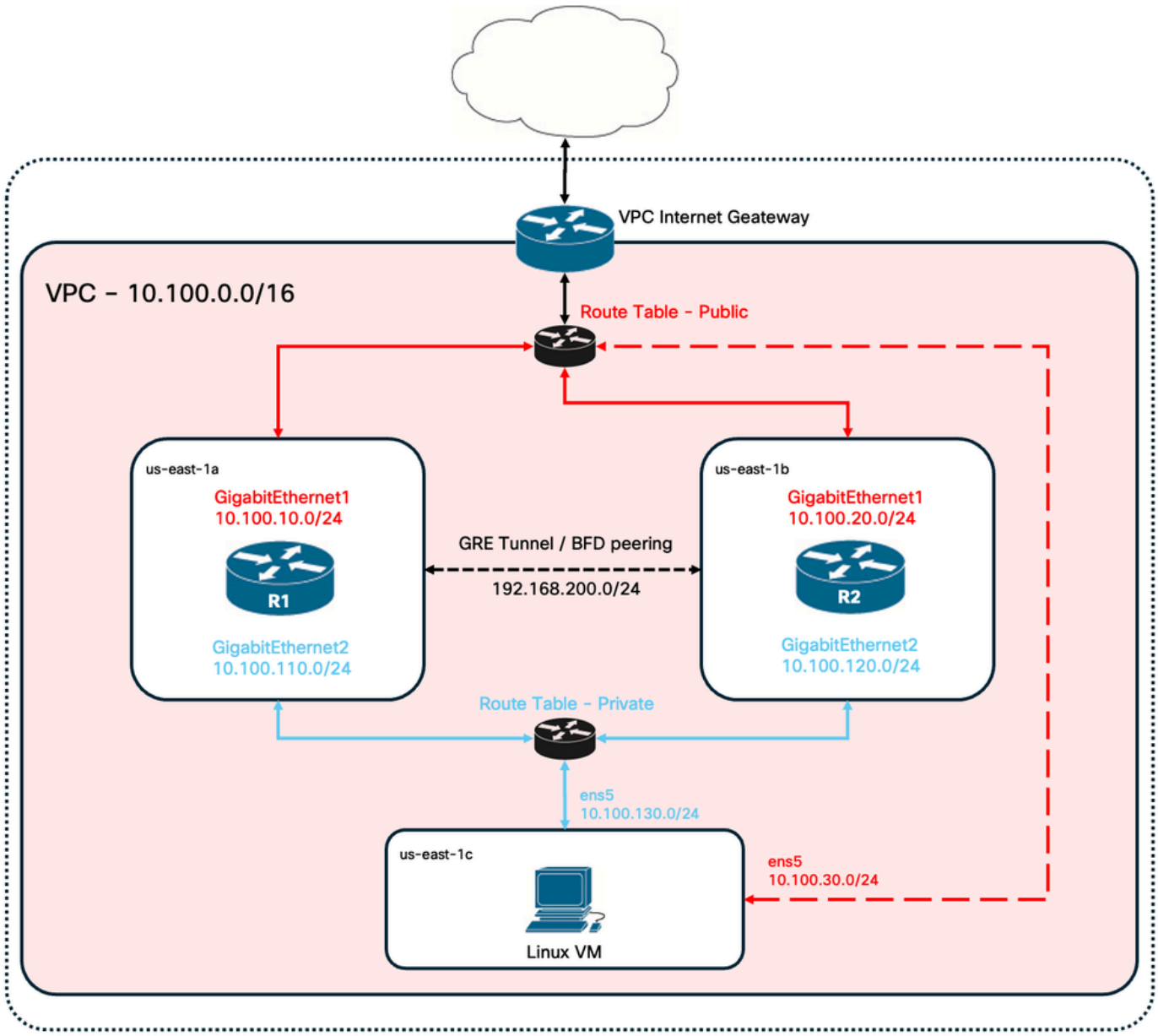
- 1x – 地域
- 1x - VPC
- 3x：アベイラビリティゾーン
- 6x – ネットワークインターフェイス/サブネット（3xパブリック側/3xプライベート側）
- 2x – ルートテーブル（パブリックおよびプライベート）
- 2x - C8000vルータ(Cisco IOS® XEDenali 17.15.3a)
- 1x - VM(Linux/Windows)

HAペアの2台のC8000vルータは、2つの異なるアベイラビリティゾーンにあります。各アベイラビリティゾーンを、ハードウェアの復元力を高める個別のデータセンターと考えてください。

3番目のゾーンはVMで、プライベートデータセンターのデバイスをシミュレートします。ここでは、パブリックインターフェイスからインターネットアクセスを有効にして、VMにアクセスして設定できるようにします。一般に、通常のトラフィックはすべて、プライベートルートテーブルを通過する必要があります。

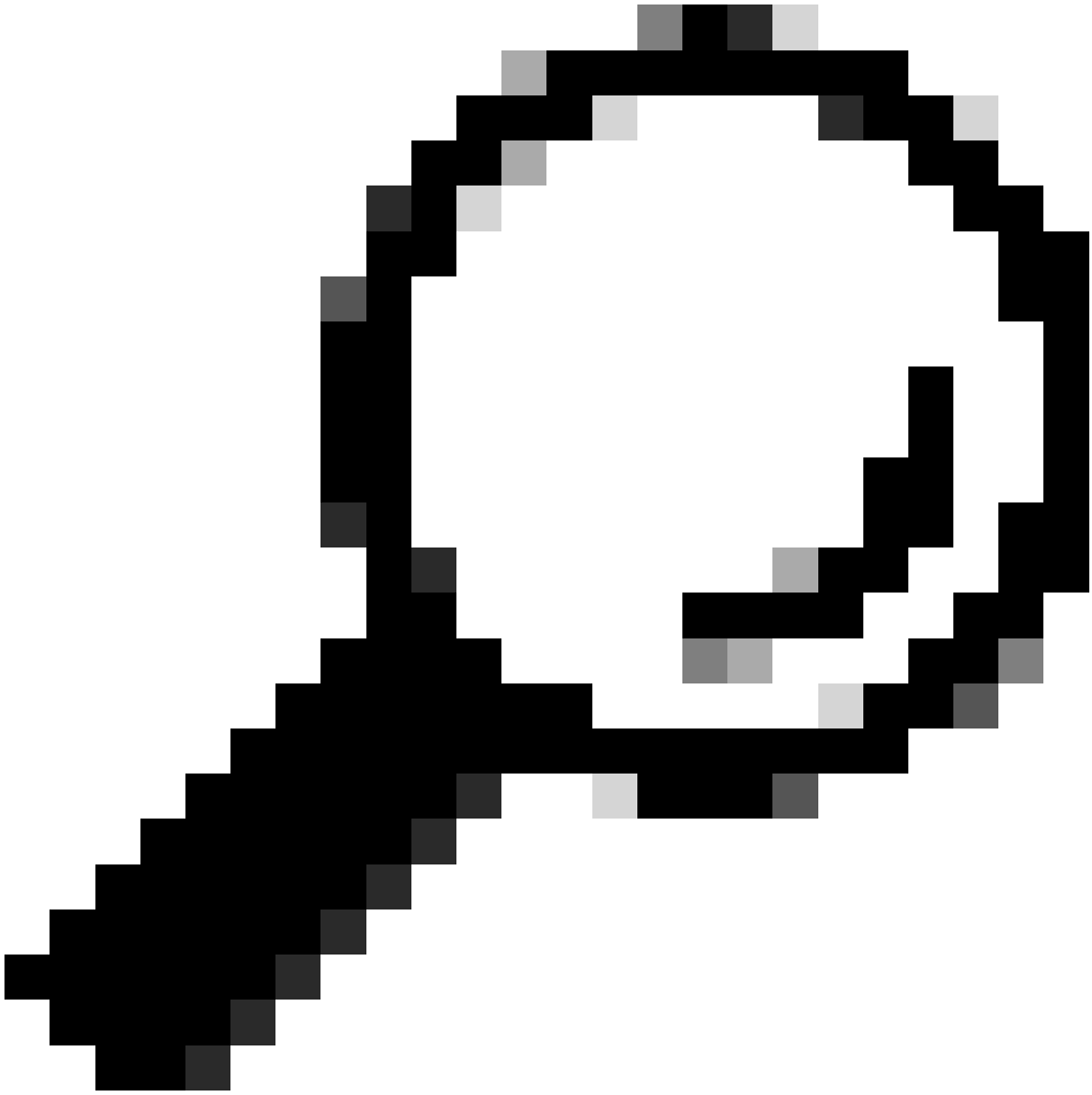
トラフィックをシミュレートするには、仮想マシンのプライベートインターフェイスからpingを開始し、R1経由でプライベートルートテーブルを通過して8.8.8.8に到達します。フェールオーバーが発生した場合は、R2ルータのプライベートインターフェイスを介してトラフィックをルーティングするように、プライベートルートテーブルが自動的に更新されていることを確認します。

ネットワーク図



表の要約

トポロジを要約するために、この表にはラボの各コンポーネントの最も重要な値を示します。この表に記載された情報は、このラボ演習の情報とは排他的です。



ヒント:この表を使用すると、ガイド全体を通じて主要な変数の概要を明確に把握するのに役立ちます。プロセスを合理化するために、この形式で情報を収集することをお勧めします。

デバイス	アベイラビリティーゾーン	インターフェイス	[IP アドレス（IP Addresses）]	RTB	エニ
R1	us-east-1a	GigabitEthernet1	10.100.10.254	rtb-0d0e48f25c9b00635（パブリック）	eni-0645a881c13823696
		GigabitEthernet2	10.100.110.254	rtb-093df10a4de426eb8（プライベート）	eni-070e14fbfde0d8e3b

R2	us-east-1b	GigabitEthernet1	10.100.20.254	rtb-0d0e48f25c9b00635（パブリック）	eni-0a7817922ffbb317b
		GigabitEthernet2	10.100.120.254	rtb-093df10a4de426eb8（プライベート）	eni-0239fda341b4d7e41
Linux仮想マシン	米国東部–1c	ens5	10.100.30.254	rtb-0d0e48f25c9b00635（パブリック）	eni-0b28560781b3435b1
		ens6	10.100.130.254	rtb-093df10a4de426eb8（プライベート）	eni-05d025e88b6355808

制約事項

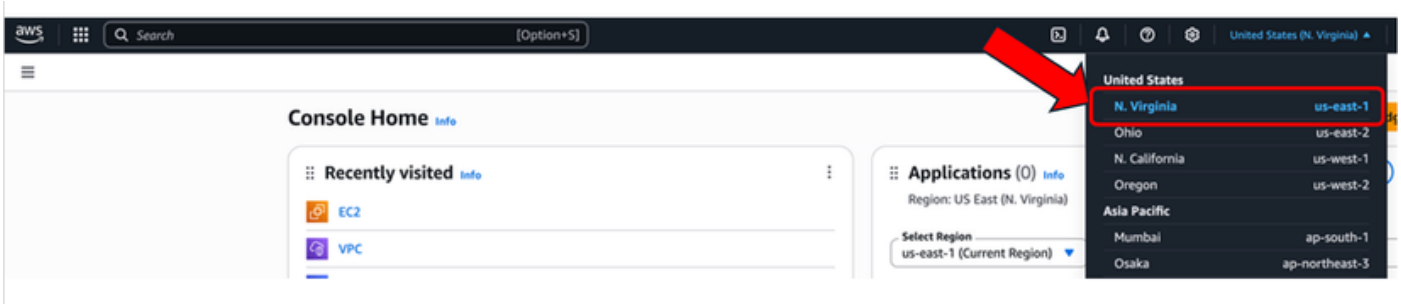
- 作成されたサブネットでは、そのサブネットの最初に使用可能なアドレスを使用しないでください。これらのIPアドレスは、AWSサービスによって内部的に使用されます。
- VRF内でC8000vデバイスのパブリックインターフェイスを設定しないでください。これが設定されていると、HAが正しく動作しません。

コンフィギュレーション

設定の一般的なフローは、要求されたVMを適切な領域に作成し、各VMのルートやインターフェイスなど、最も具体的な設定に移行することに重点を置いています。ただし、最初にトポロジを理解し、必要な順序でトポロジを設定することをお勧めします。

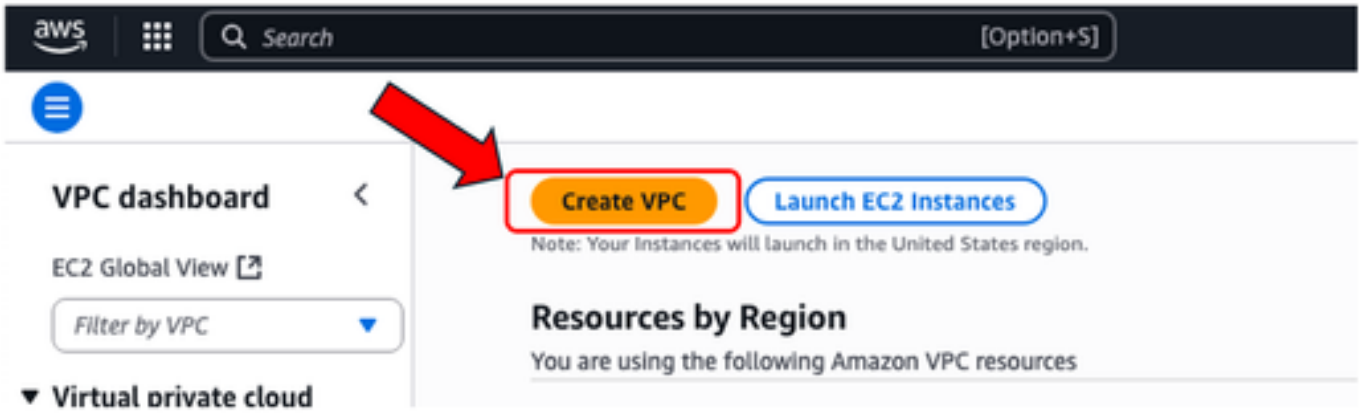
ステップ 1：地域の選択

この導入ガイドでは、VPCリージョンとして米国西部（バージニア北部）– us-east-1リージョンが選択されています。



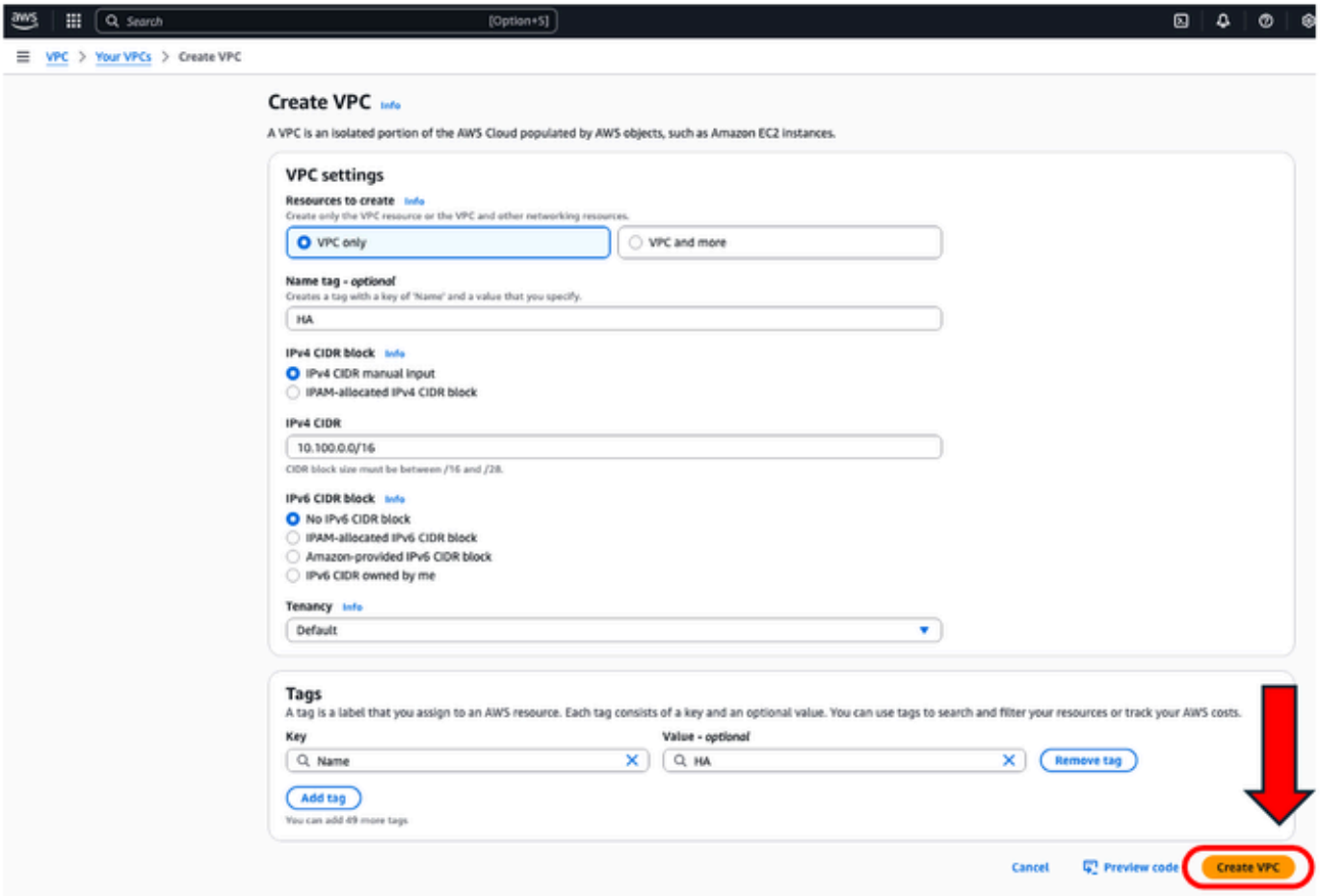
ステップ 2：VPCの作成

AWSコンソールで、[VPC] > [VPCダッシュボード] > [VPCの作成]に移動します。

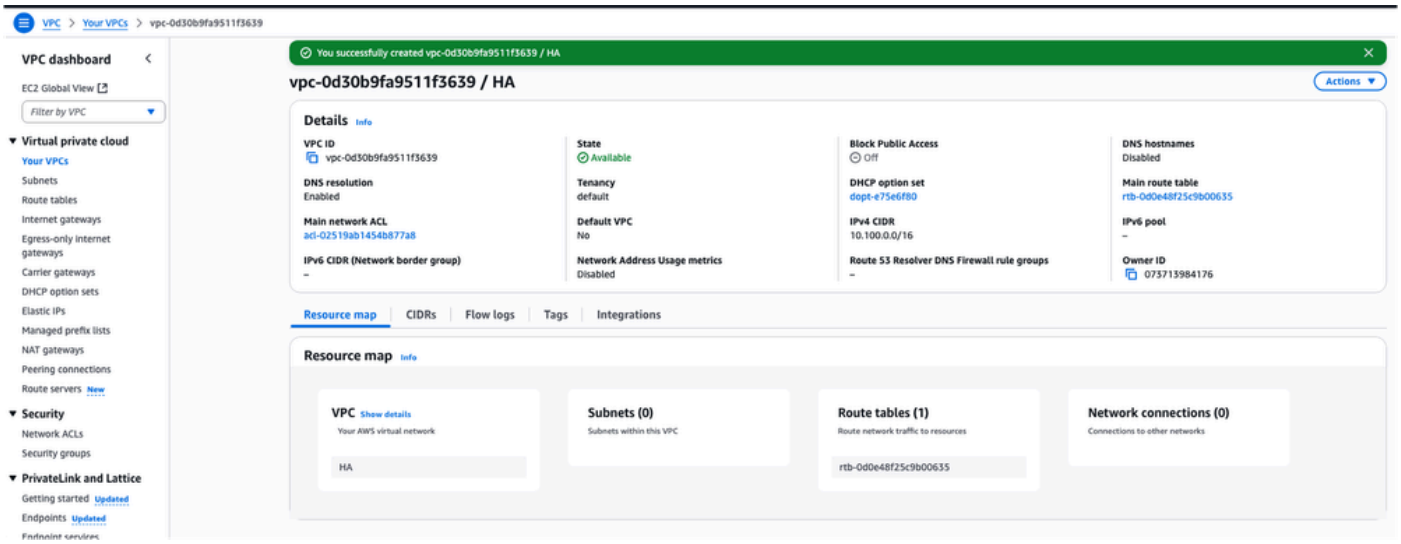


VPCを作成する際には、VPC onlyオプションを選択します。 必要に応じて、/16ネットワークを割り当てて使用できます。

この導入ガイドでは、10.100.0.0/16ネットワークを選択します。

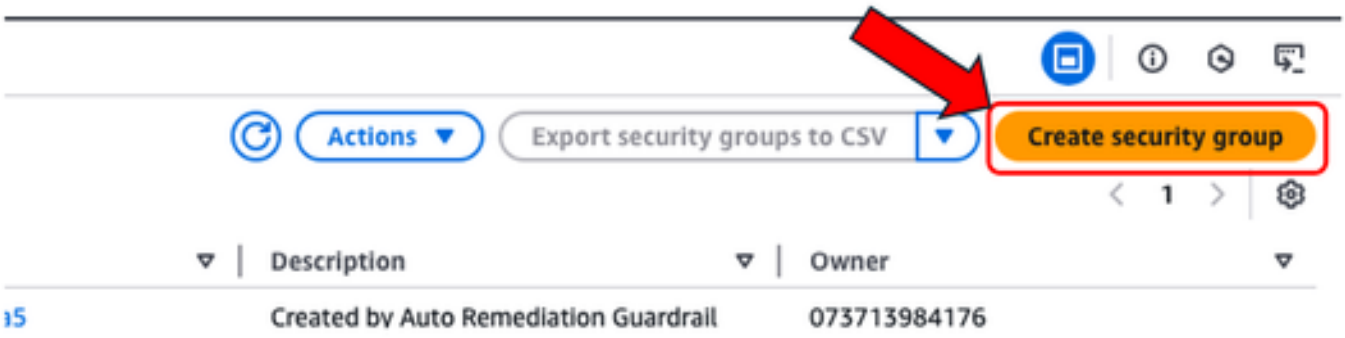


Create VPCをクリックすると、HAタグが付いたVPC-0d30b9fa9511f3639が作成されます。

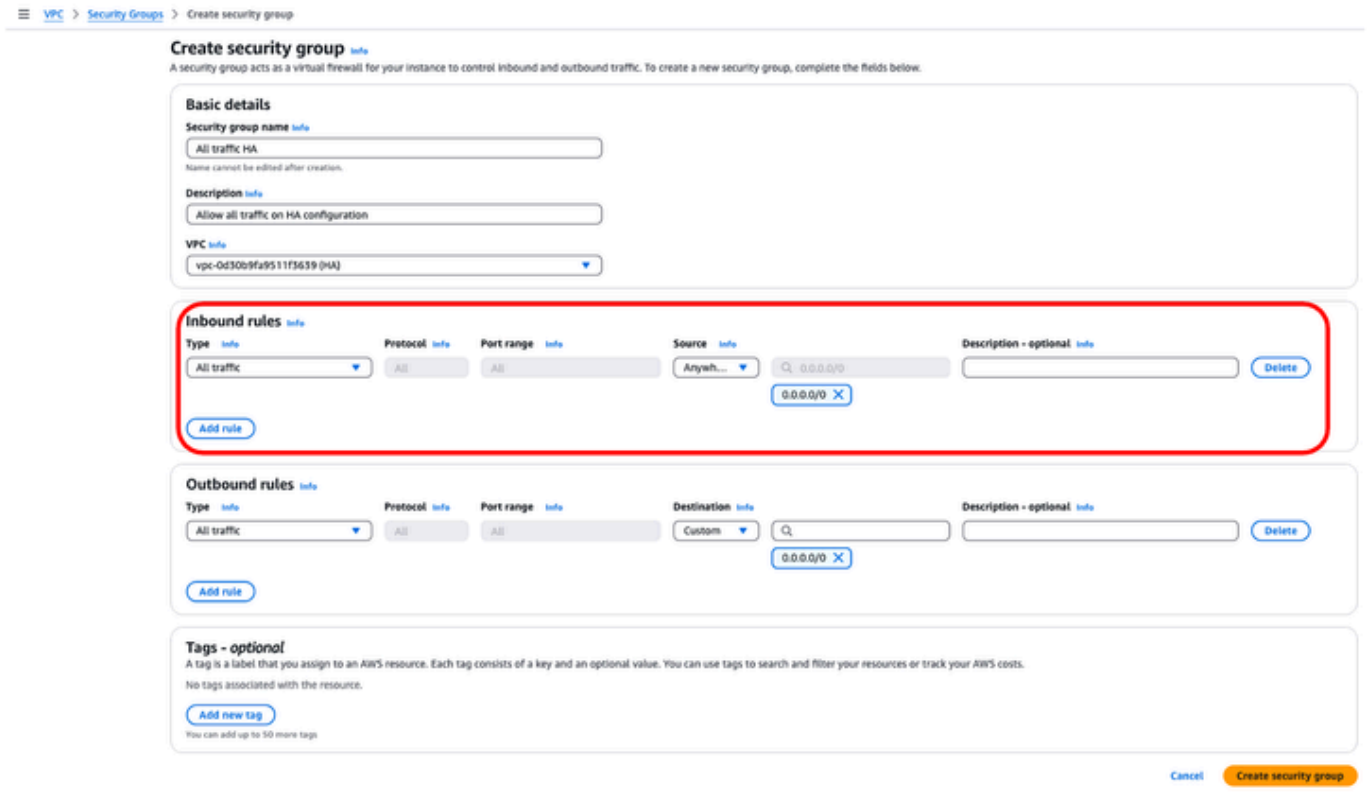


ステップ 3：VPCのセキュリティグループの作成

AWSでは、セキュリティグループはACLと同様に機能し、VPC内の構成済みVMへのトラフィックを許可または拒否します。 AWSコンソールで、[VPC] > [VPCダッシュボード] > [セキュリティ] > [セキュリティグループ] セクションに移動し、[セキュリティグループの作成] をクリックします。



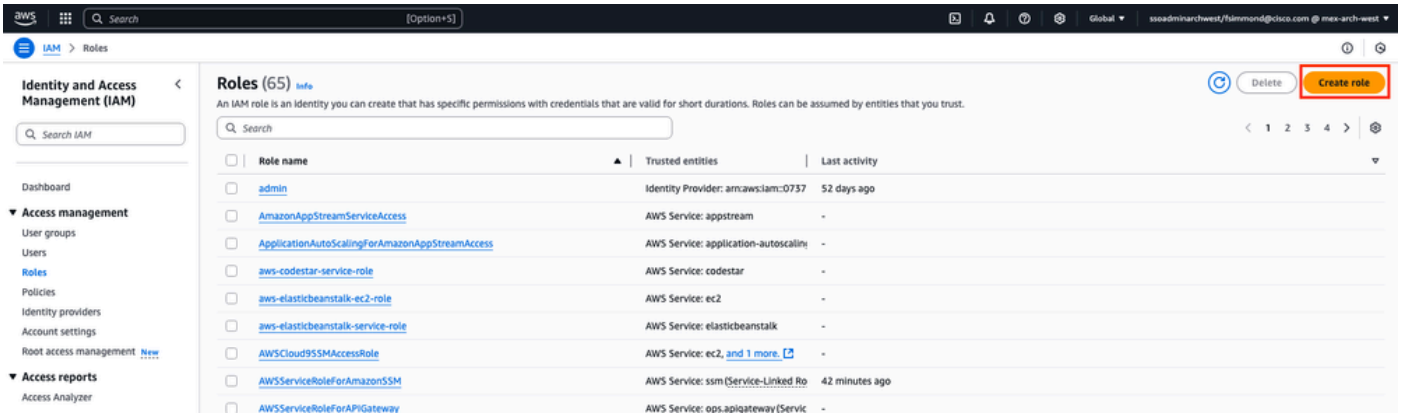
Inbound Rulesで、許可するトラフィックを定義します。この例では、0.0.0.0/0ネットワークを使用してAll Trafficを選択しています。



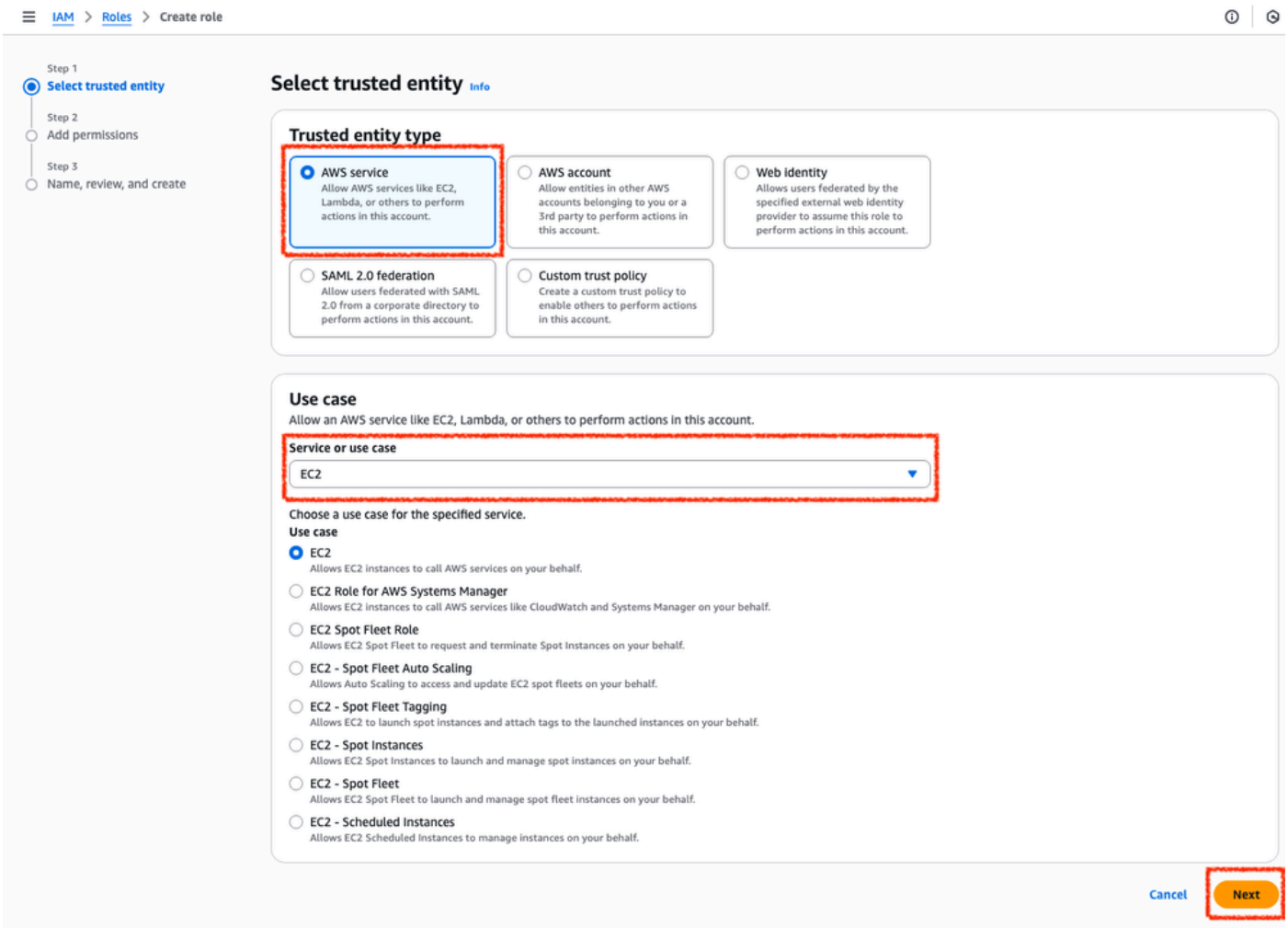
ステップ 4：ポリシーを使用してIAMロールを作成し、VPCに関連付ける

IAMはAMIにAmazon APIへの必要なアクセス権を付与します。C8000vは、AWS APIコマンドを呼び出してAWSのルートテーブルを変更するためのプロキシとして使用されます。デフォルトでは、EC2インスタンスはAPIへのアクセスを許可されていません。このため、AMIの作成時に適用される新しいIAMロールを作成する必要があります。

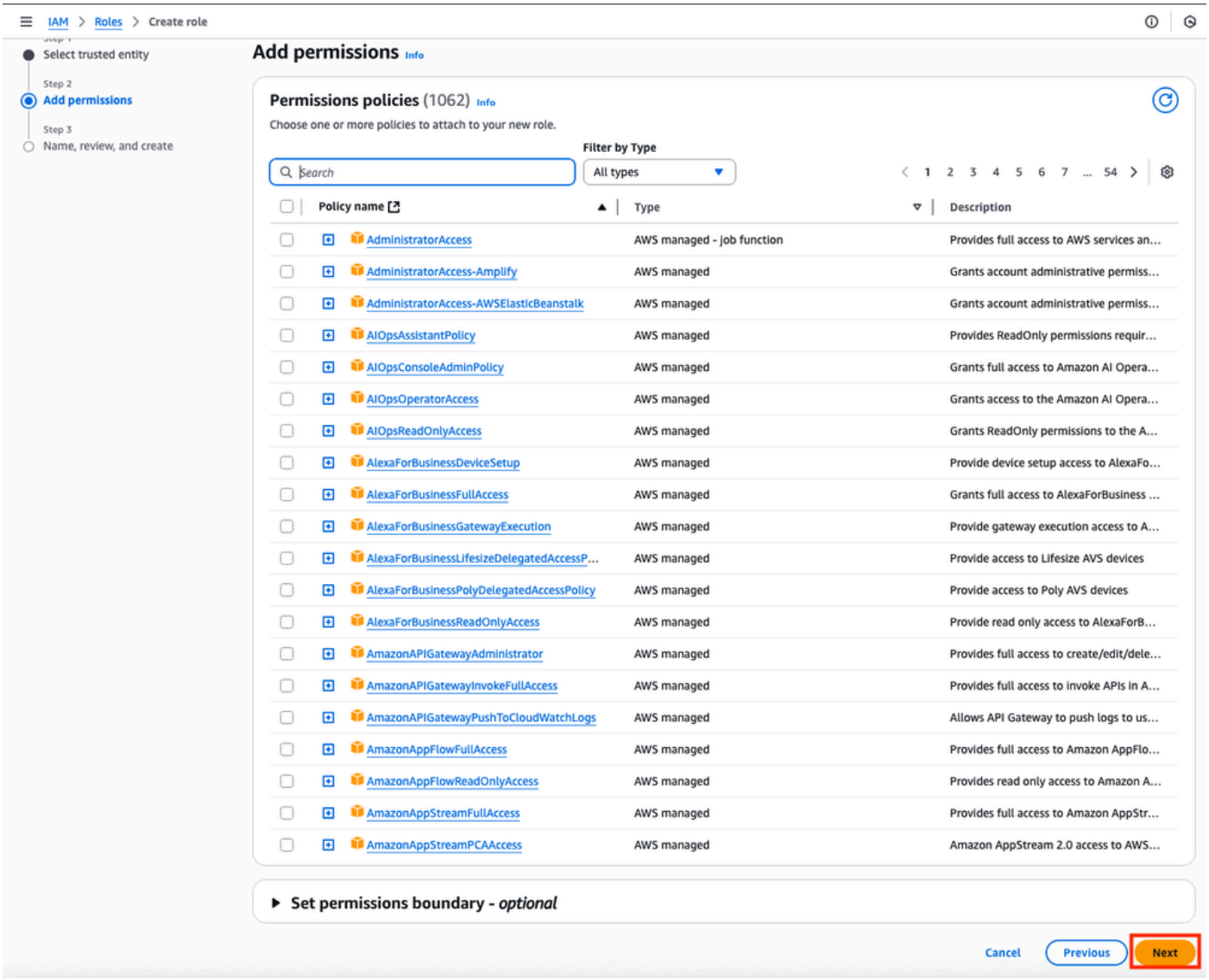
IAMダッシュボードを参照し、Access Management > Roles > Create Roleに移動します。このプロセスは、次の3つのステップで構成されます。



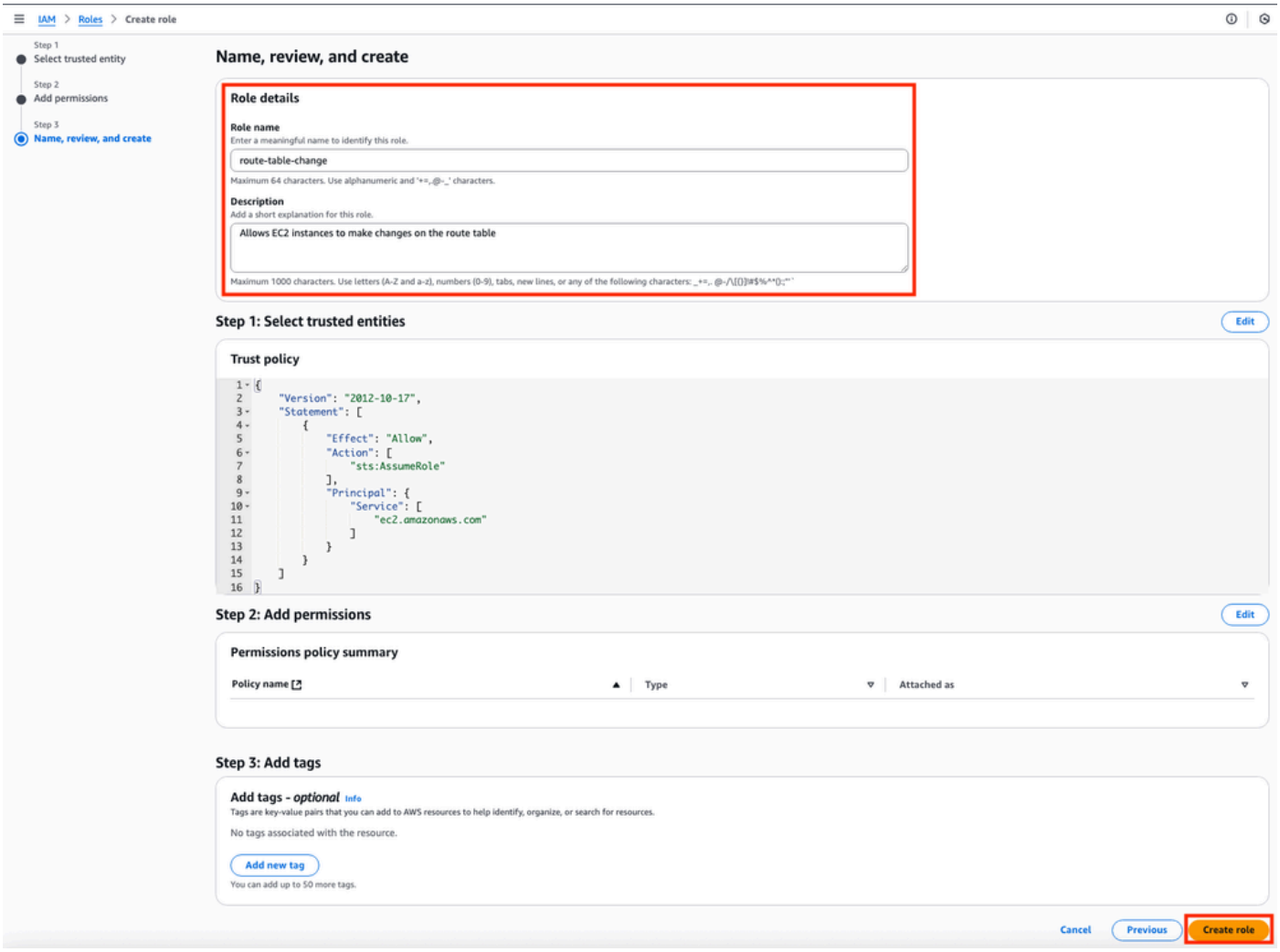
最初に、Trusted entity typeセクションでAWS Serviceオプションを選択し、このポリシーに割り当てられたサービスとしてEC2 を選択します。



終了したら、Nextをクリックします。

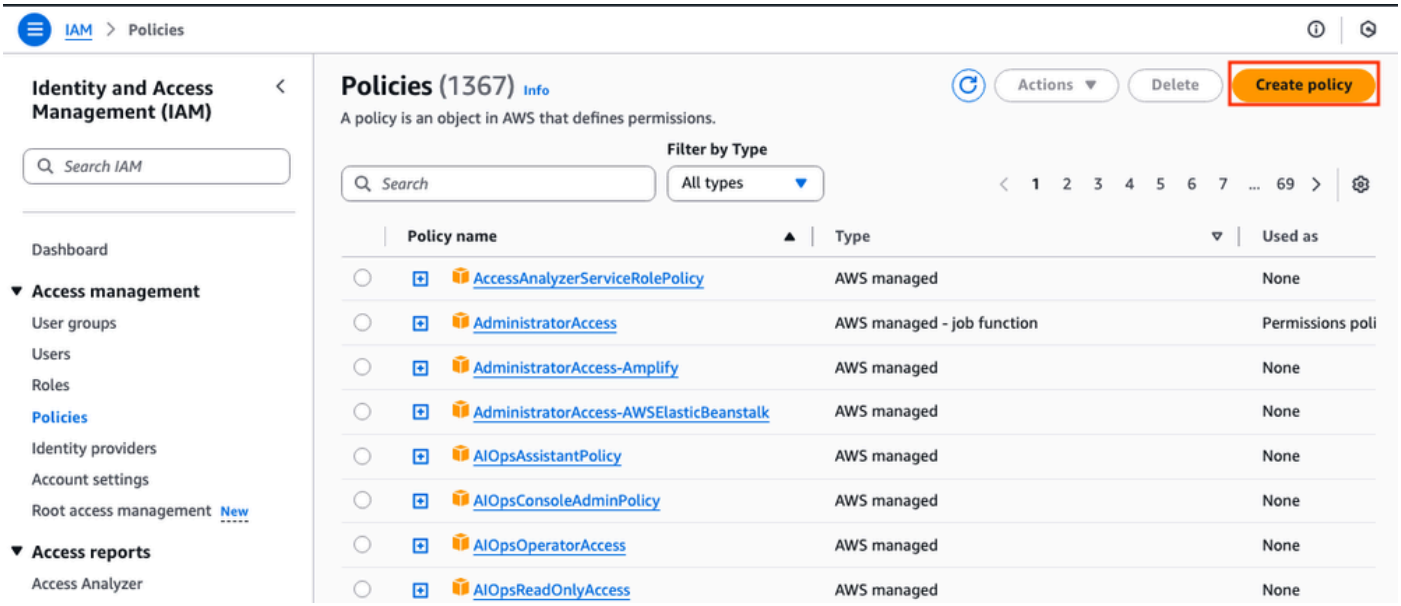


最後に、ロール名を設定して、Create Roleボタンをクリックします。

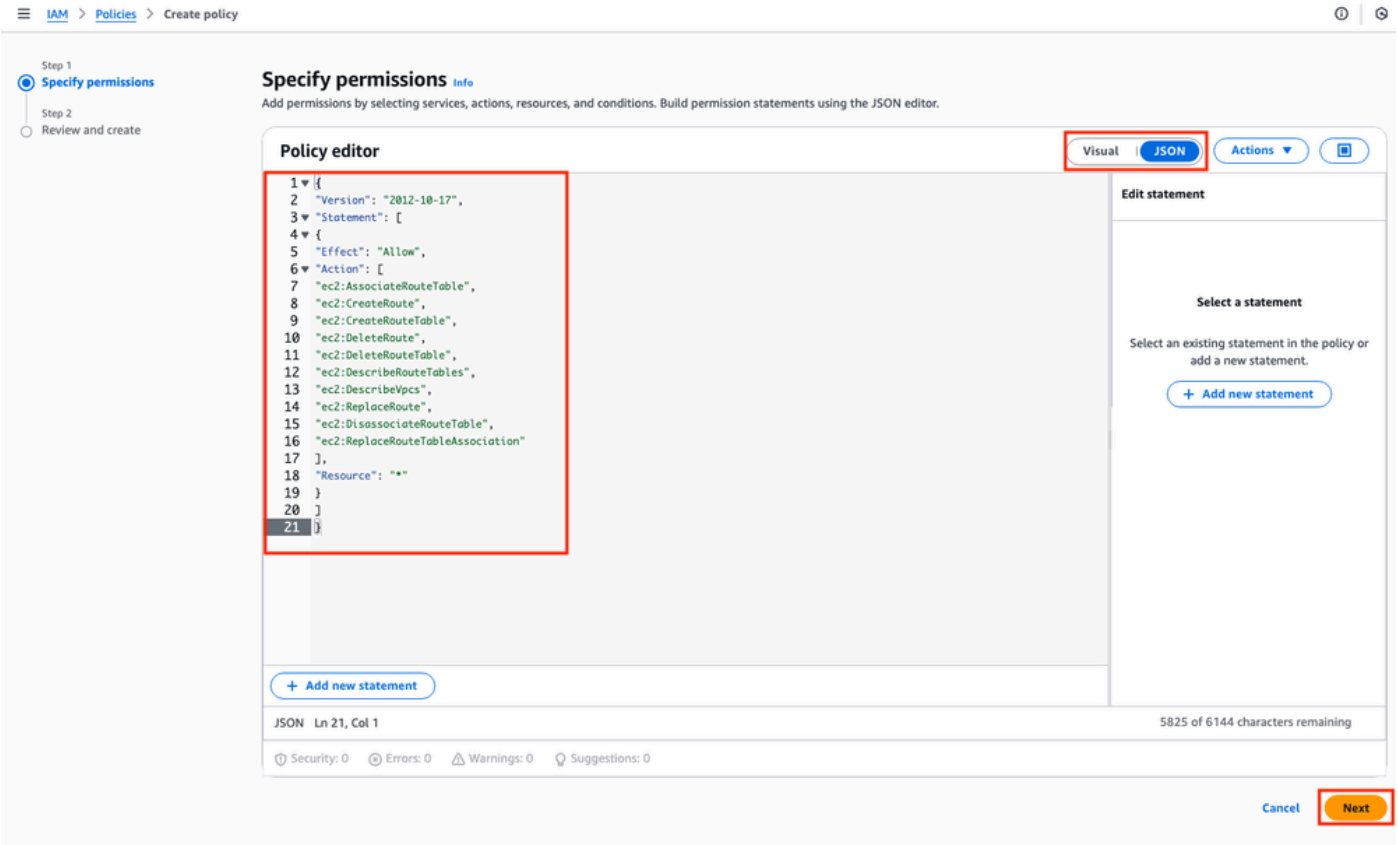


ステップ 5：IAMロールに対する信頼ポリシーの作成と適用

ロールを作成したら、必要に応じてAWSルーティングテーブルを変更するスキルを習得するために、信頼ポリシーを作成する必要があります。IAMダッシュボードのポリシーセクションに移動します。Create Policyボタンをクリックします。このプロセスは、次の2つの手順で構成されます。



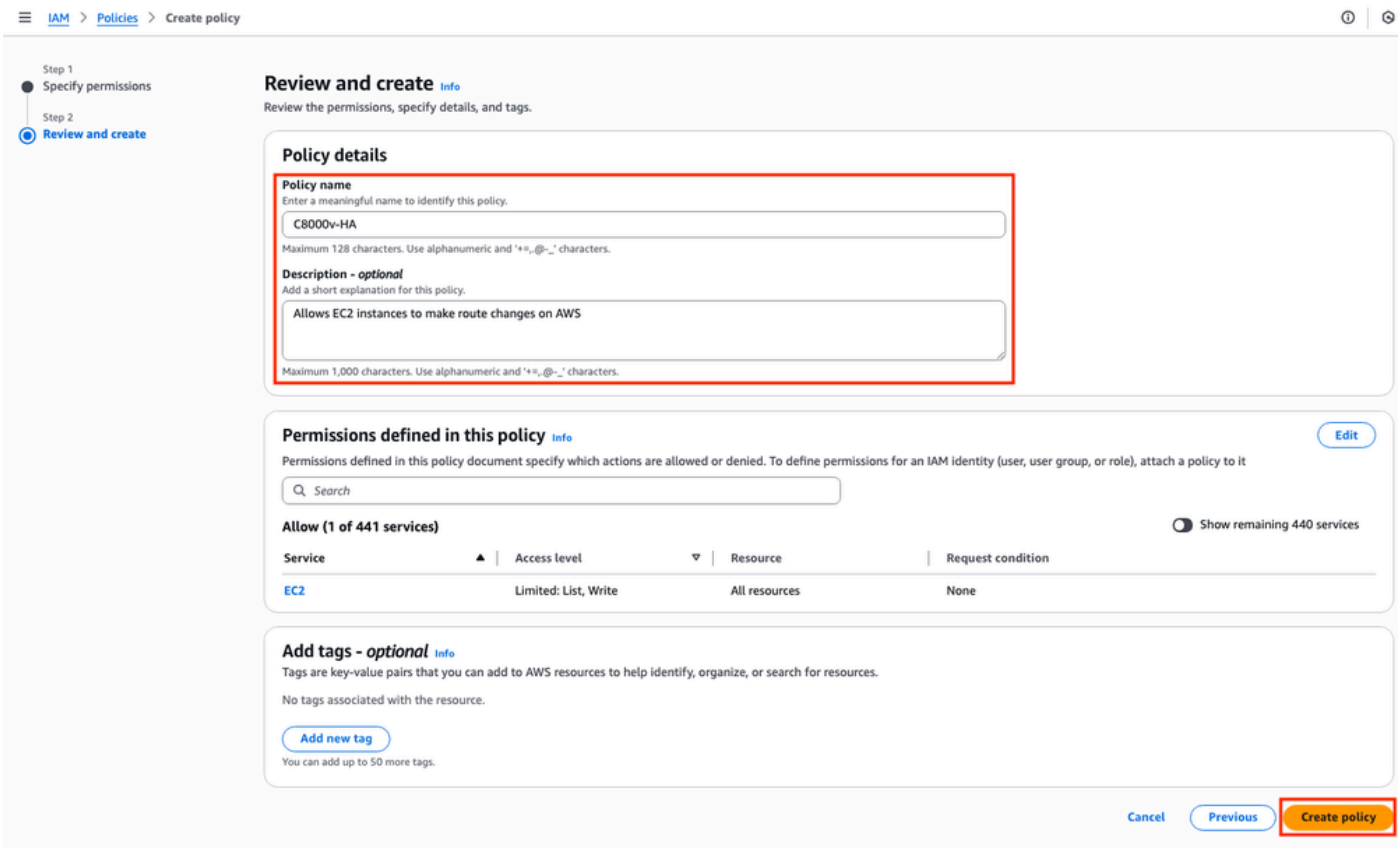
まず、ポリシーエディタでJSONが使用されていることを確認し、次に示すコマンドを適用します。設定が完了したら、Nextをクリックします。



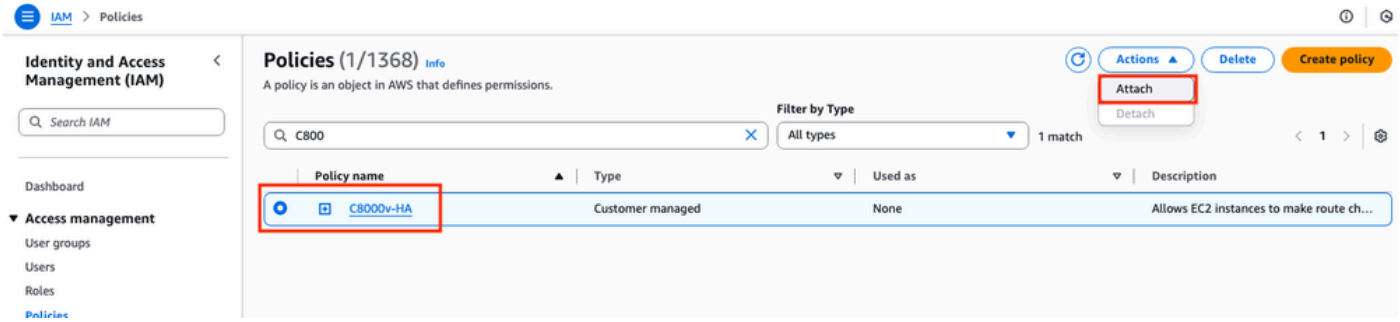
次に、このイメージで使用されているテキストコードを示します。

```
{
"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": "*"
}
]
}
```

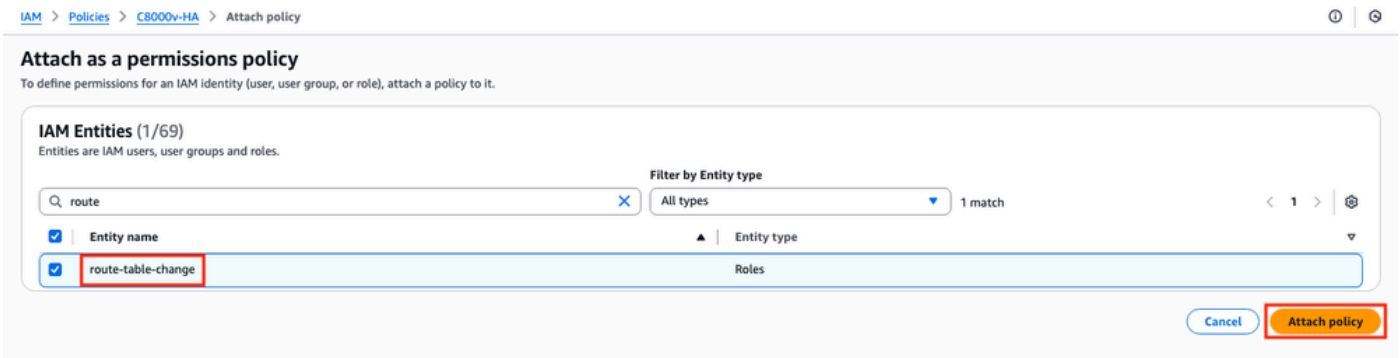
後で、Policy Nameを設定して、Create Policyをクリックします。



ポリシーを作成したら、フィルタリングを行ってポリシーを選択し、Actionsドロップダウンメニューでオプションをクリックします。



新しいウィンドウが開きます。IAM Entitiesセクションで、作成したIAM Roleをフィルタリングして選択し、Attach policyをクリックします。

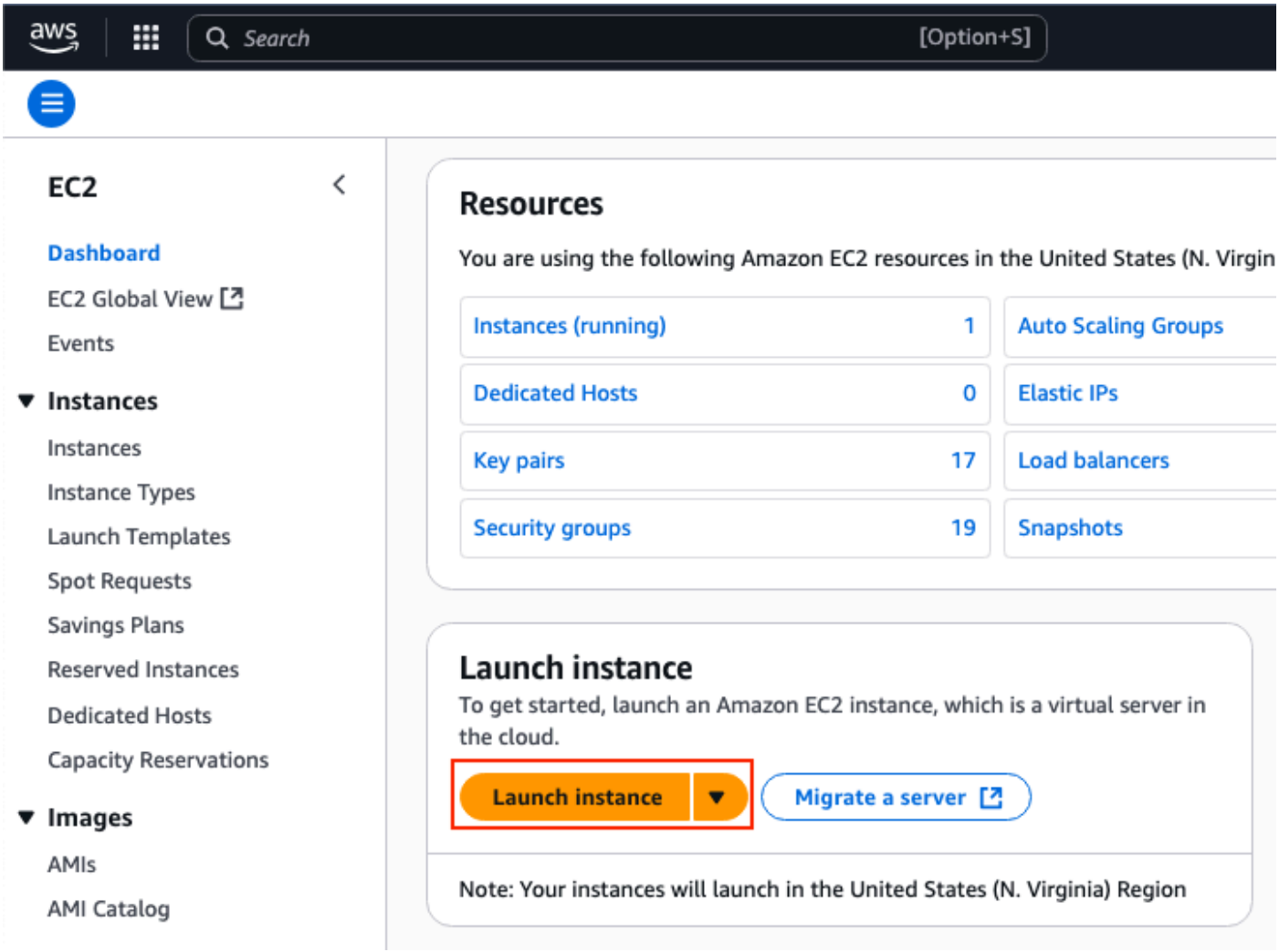


ステップ6:C8000vインスタンスの設定と起動

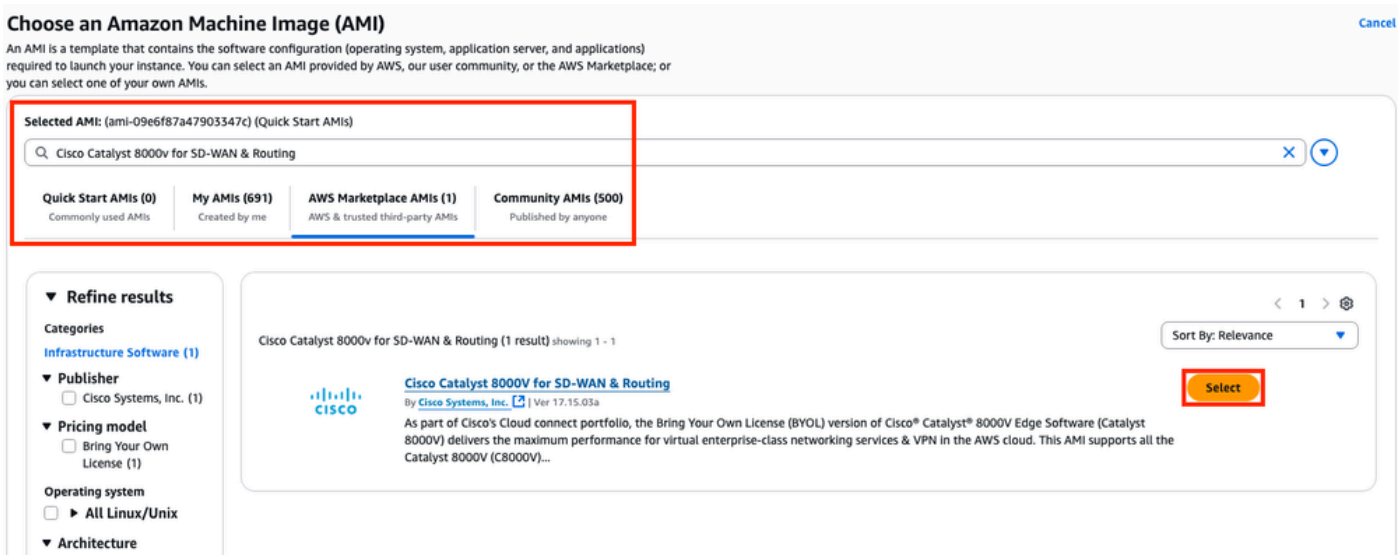
各C8000vルータには2つのインターフェイス（パブリック1つ、プライベート1つ）があり、独自

のアベイラビリティーゾーンに作成されます。

EC2ダッシュボードで、インスタンスの起動をクリックします。

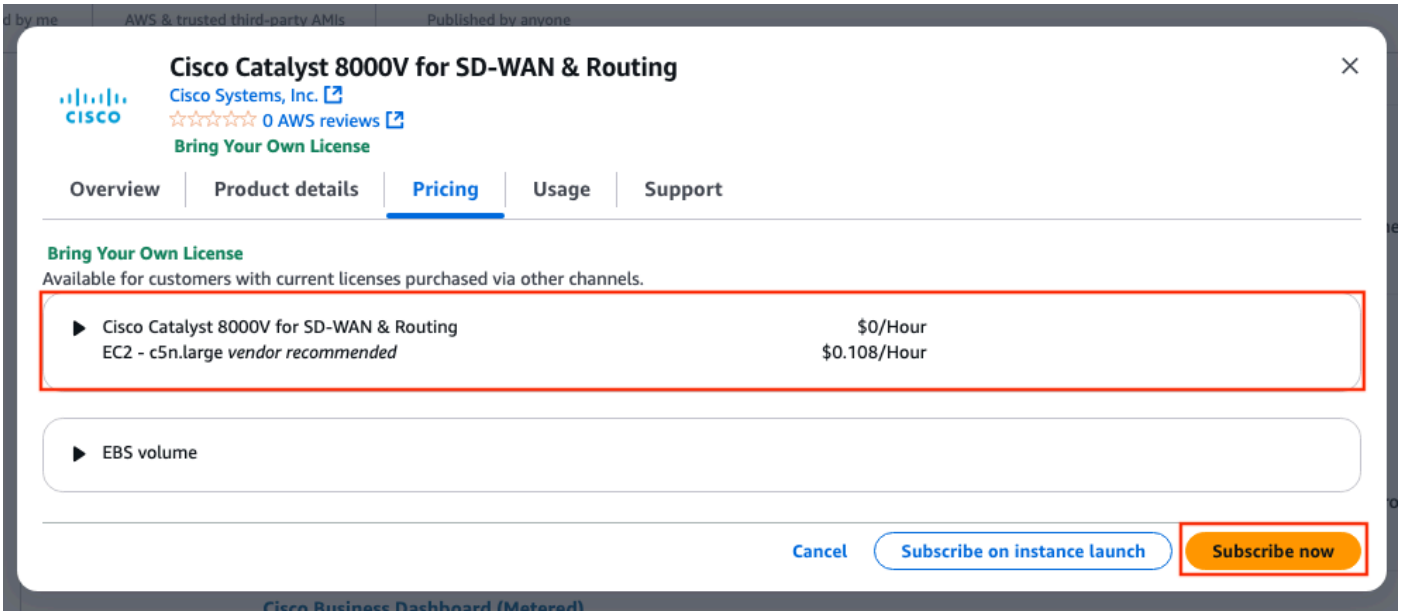


Cisco Catalyst 8000v for SD-WAN & Routingという名前でAMIデータベースをフィルタ処理します。 AWS Marketplace AMIsリストで、Selectをクリックします。



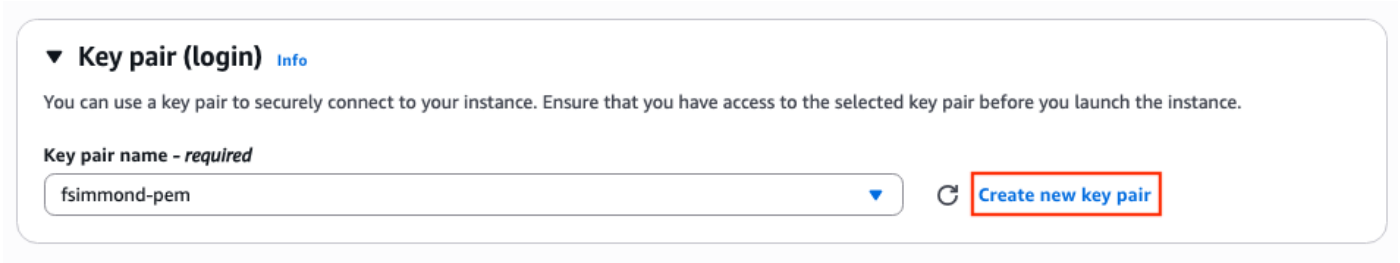
AMIに対応するサイズを選択します。この例では、c5n.largeサイズが選択されています。これは

、ネットワークに必要な容量によって異なります。選択したら、Subscribe nowをクリックします。

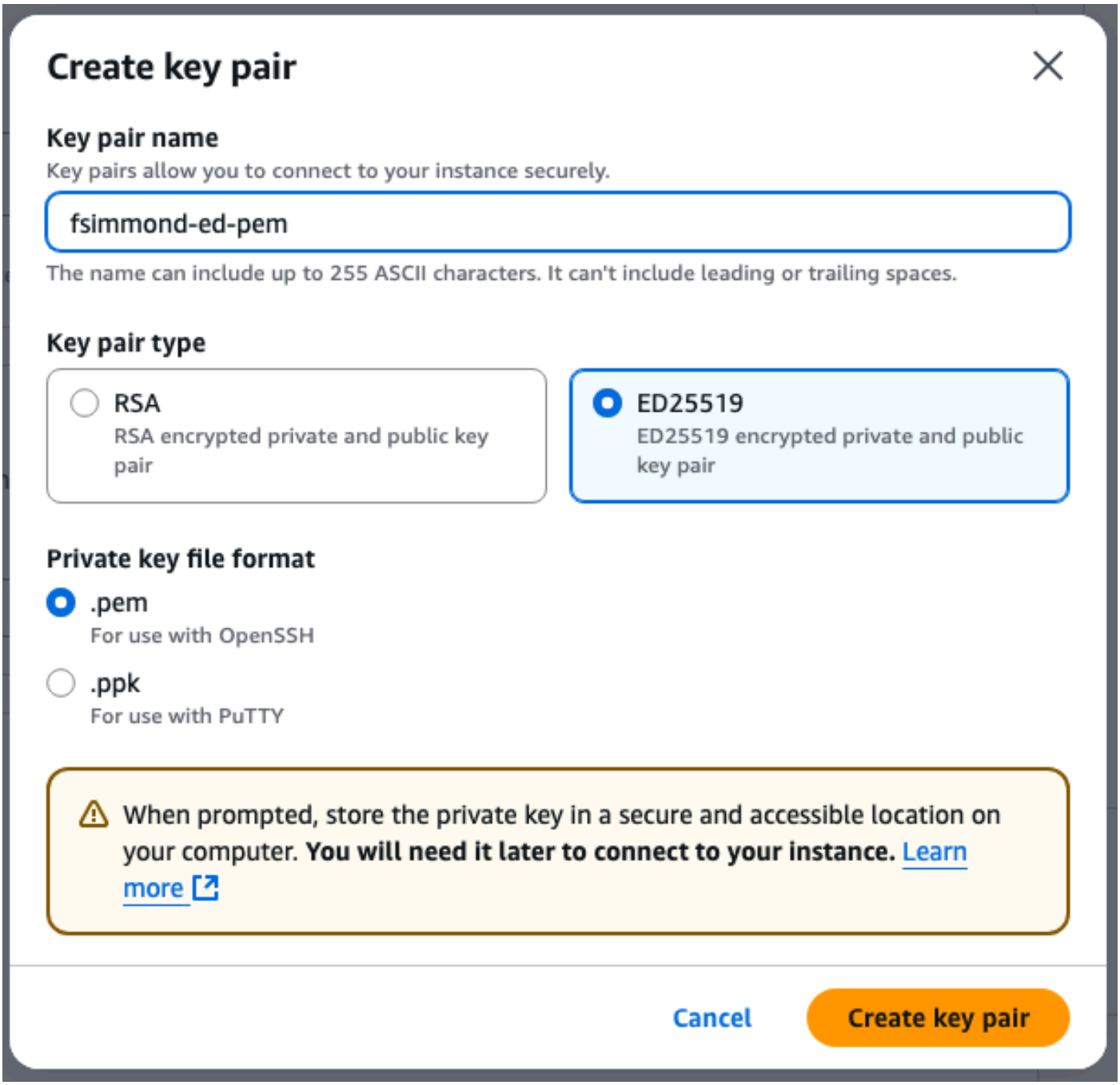


ステップ 6.1：リモートアクセス用のキーペアの設定

AMIに登録すると、複数のオプションを含む新しいウィンドウが表示されます。Key pair (login)セクションで、キーペアが存在しない場合は、Create new key pairをクリックします。作成したデバイスごとに1つのキーを再利用できます。



新しいポップアップウィンドウが表示されます。この例では、ED25519暗号化を使用した.pemキーファイルが作成されます。すべてが設定されたら、Create key pairをクリックします。



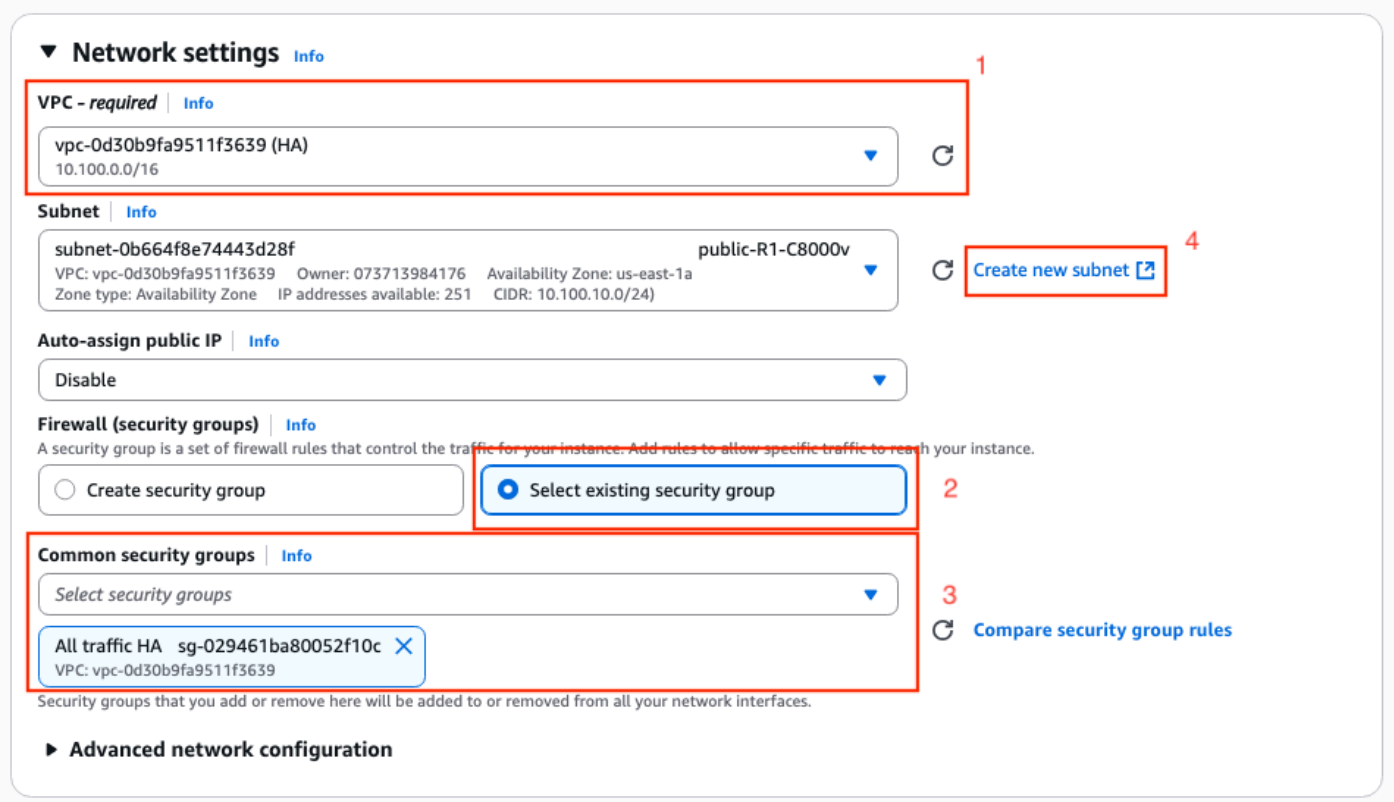
ステップ 6.2：AMIのサブネットの作成および設定

Network Settingsセクションで、Editをクリックします。セクション内の新しいオプションが利用可能になりました。
1. この作業で使用するVPCを選択します。この例では、HAという名前のVPCが選択されます。

2. Firewall (security groups)セクションで、Select existing security groupを選択します。

3. オプション2を選択すると、「共通セキュリティグループ」オプションが使用可能になります。フィルタリングを行い、目的のセキュリティグループを選択します。この例では、All traffic HAセキュリティグループが選択されています。

4.（オプション）これらのデバイスのサブネットが作成されていない場合は、Create new subnetをクリックします。



Webブラウザの新しいタブが開き、「サブネットの作成」セクションが表示されます。

1. ドロップダウンリストから、この設定に対応するVPCを選択します。

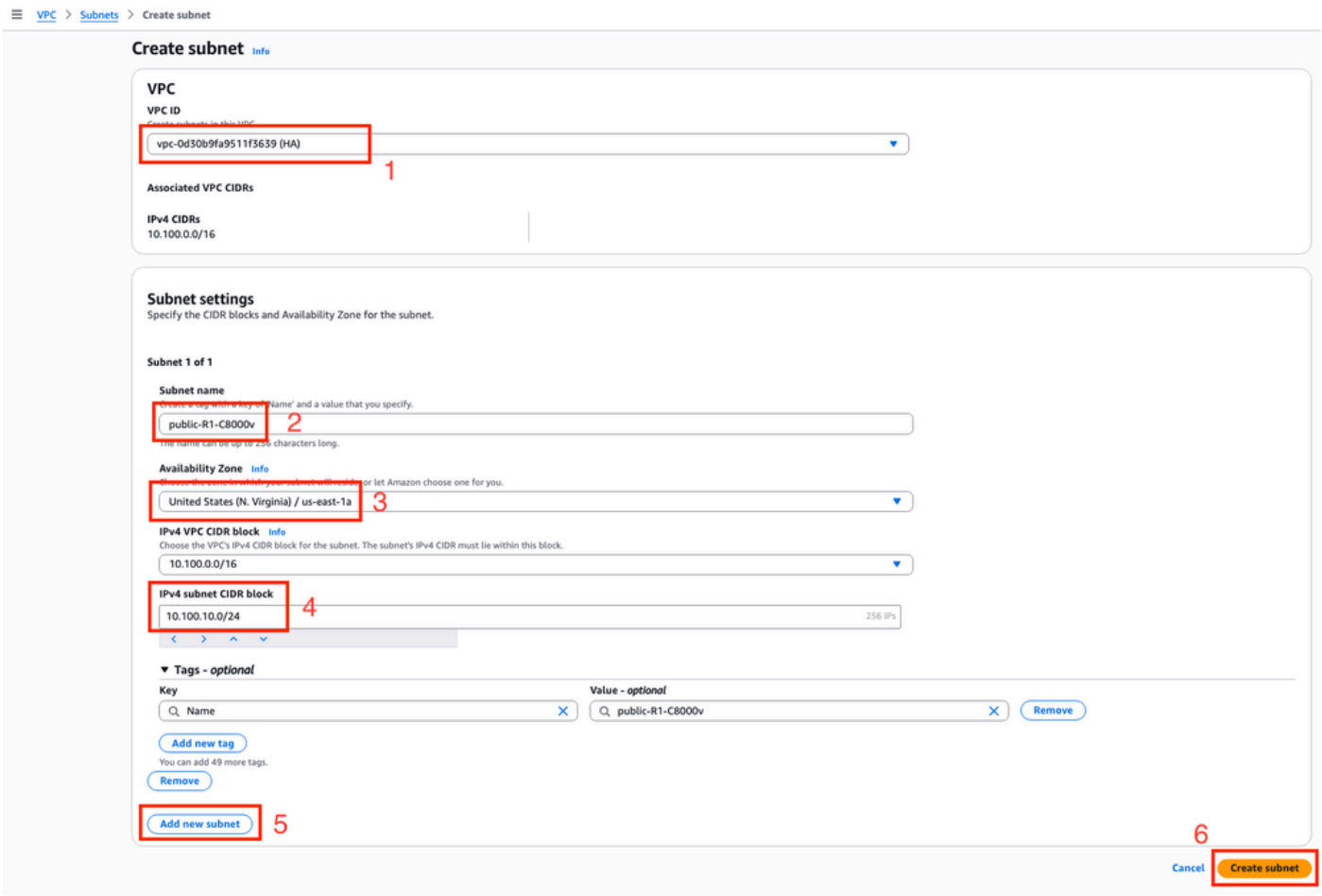
2. 新しいサブネットの名前を設定します。

3. このサブネットの可用性ゾーンを定義します。（設定の詳細については、このドキュメントの「トポロジ」セクションを参照してください）。

4. VPC CIDRブロックに属するサブネットブロックを設定します。

5. さらに、使用されるすべてのサブネットは、「新しいサブネットの追加」セクションをクリックして各サブネットについて2 ~ 4の手順を繰り返すことで作成できます。

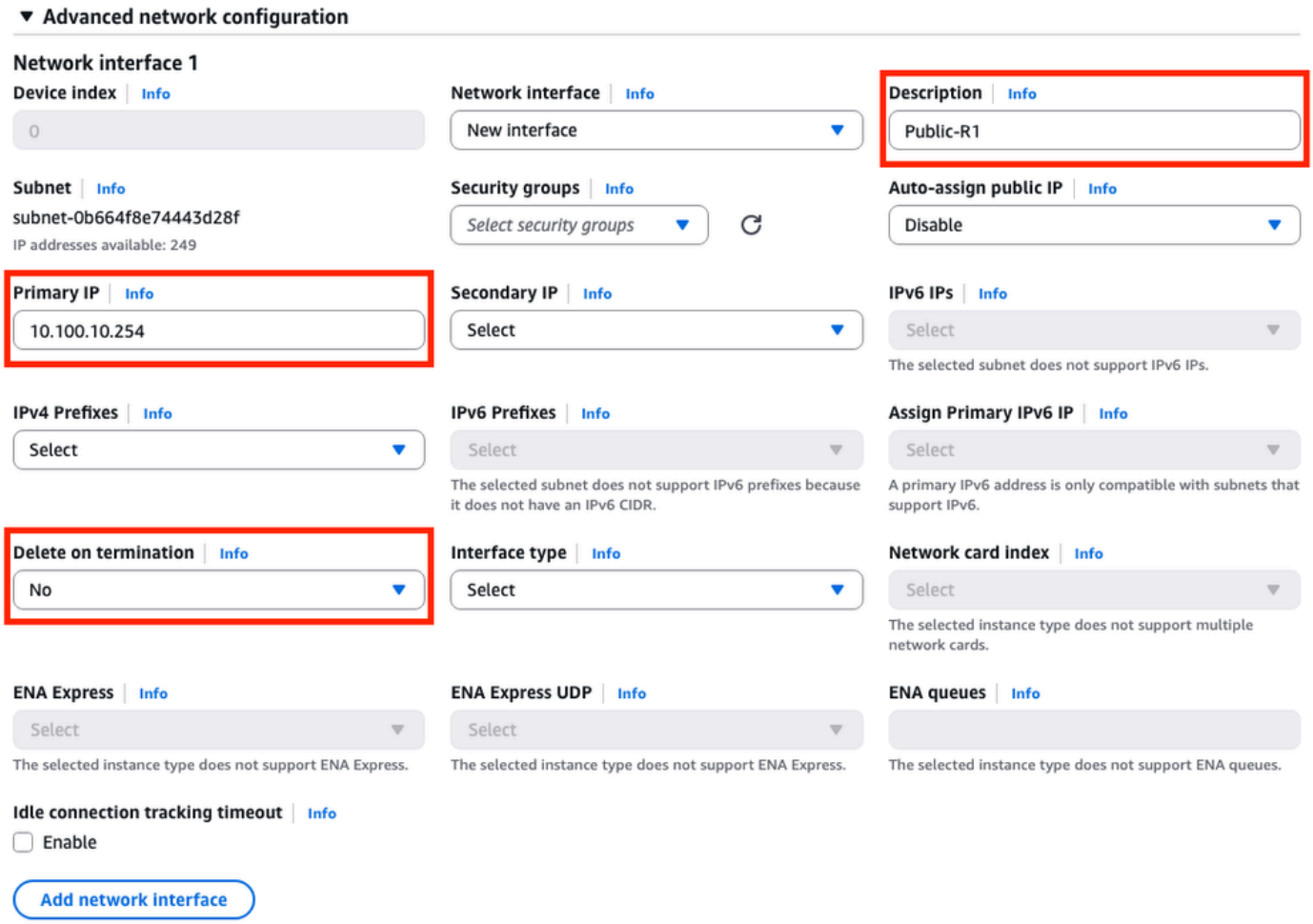
6. 終了したら、Create subnetをクリックします。前のページに移動して、設定を続行します。



Network SettingsセクションのSubnetサブセクションで、Refreshアイコンをクリックして、作成されたサブネットをドロップダウンリストに表示します。

ステップ 6.3：AMIインターフェイスの設定

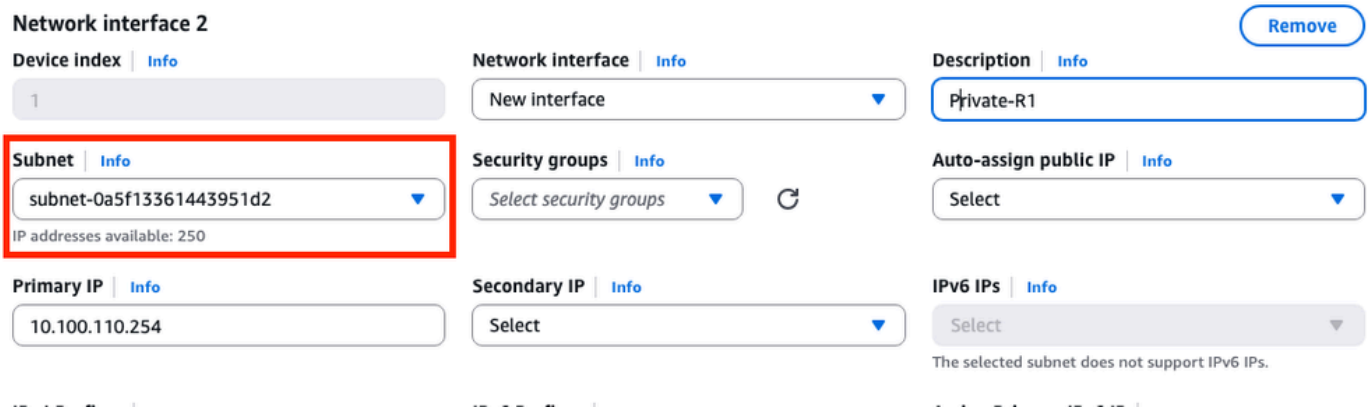
Network Settingsセクションで、Advanced Network configuration サブセクションを展開します。次のオプションが表示されます。



このメニューで、Description、Primary IP、Delete on terminationの各パラメータを設定します。Primary IPパラメータには、サブネットの最初に使用可能なアドレスを除くすべてのIPアドレスを使用します。これはAWSによって内部的に使用されます。

この例では、Delete on terminationパラメータはNoに設定されています。ただし、環境によってはyesに設定することもできます。

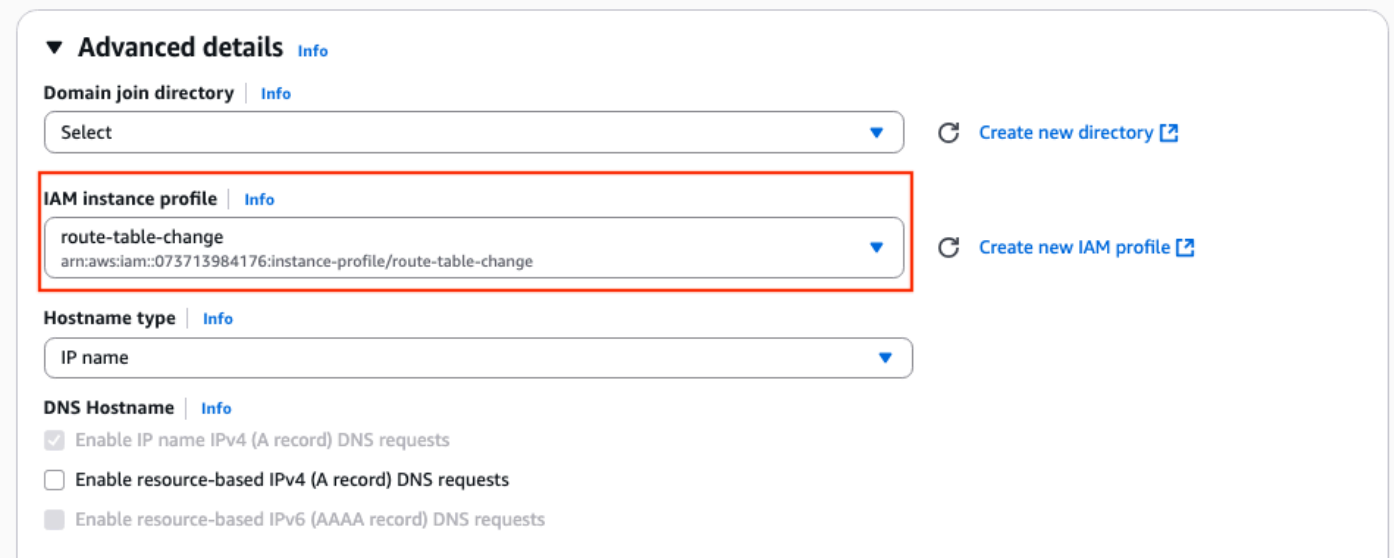
このトポロジのため、プライベートサブネットには2番目のインターフェイスが必要です。Add network interfaceをクリックすると、次のプロンプトが表示されます。ただし、インターフェイスは今回サブネットを選択するオプションを提供します。



すべてのパラメータをネットワークインターフェイス1で行ったように設定したら、次の手順に進みます。

ステップ 6.4：IAMインスタンスプロファイルをAMIに設定する

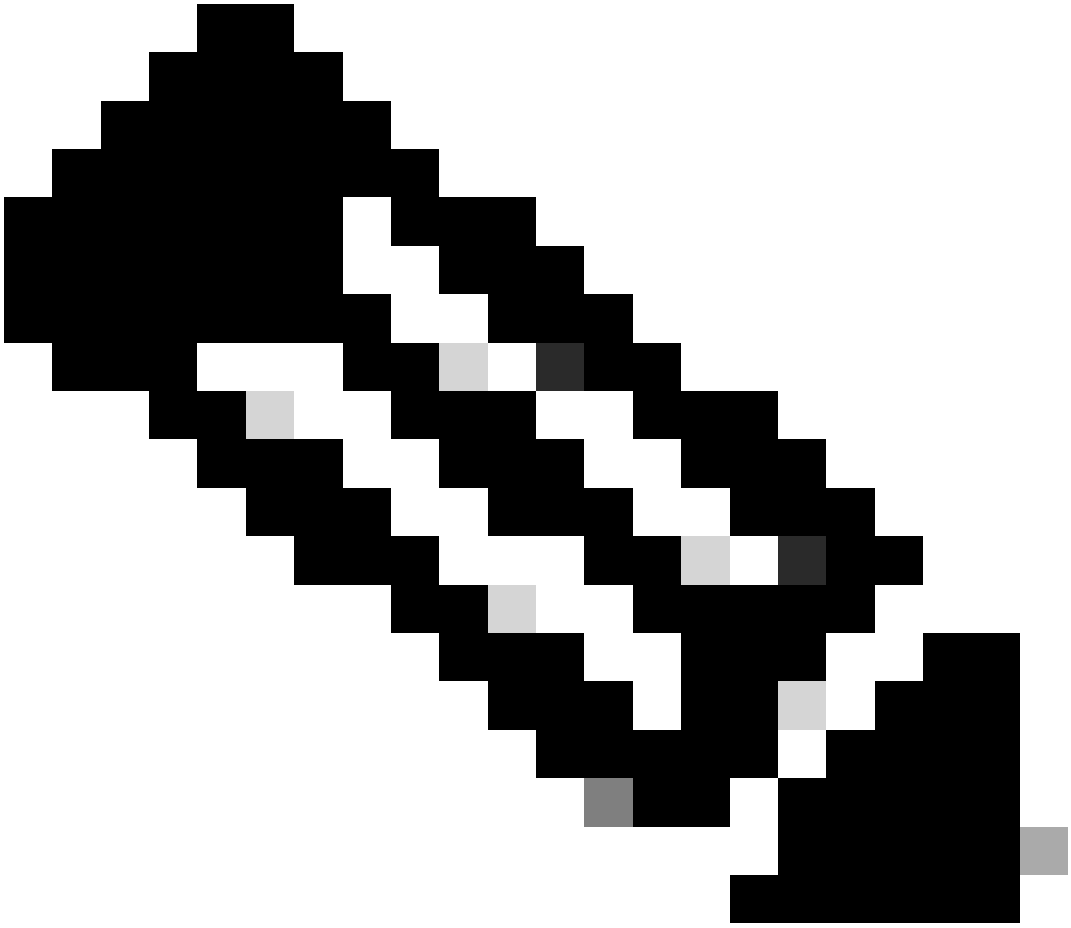
Advanced detailsセクションで、IAM instance profileパラメータで作成したIAMロールを選択します。



ステップ6.5: (オプション) AMIでのクレデンシャルの設定

Advanced detailsセクションで、User data - optional セクションに移動し、この設定を適用してインスタンスの作成中にユーザ名とパスワードを設定します。

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



注:AWSがSSHを使用してC8000vに入力したユーザ名が、ルートとして誤ってリストされる可能性があります。必要に応じて、これをec2-userに変更します。

ステップ 6.6：インスタンス設定の完了

すべてが設定されたら、Launch Instanceをクリックします。

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.

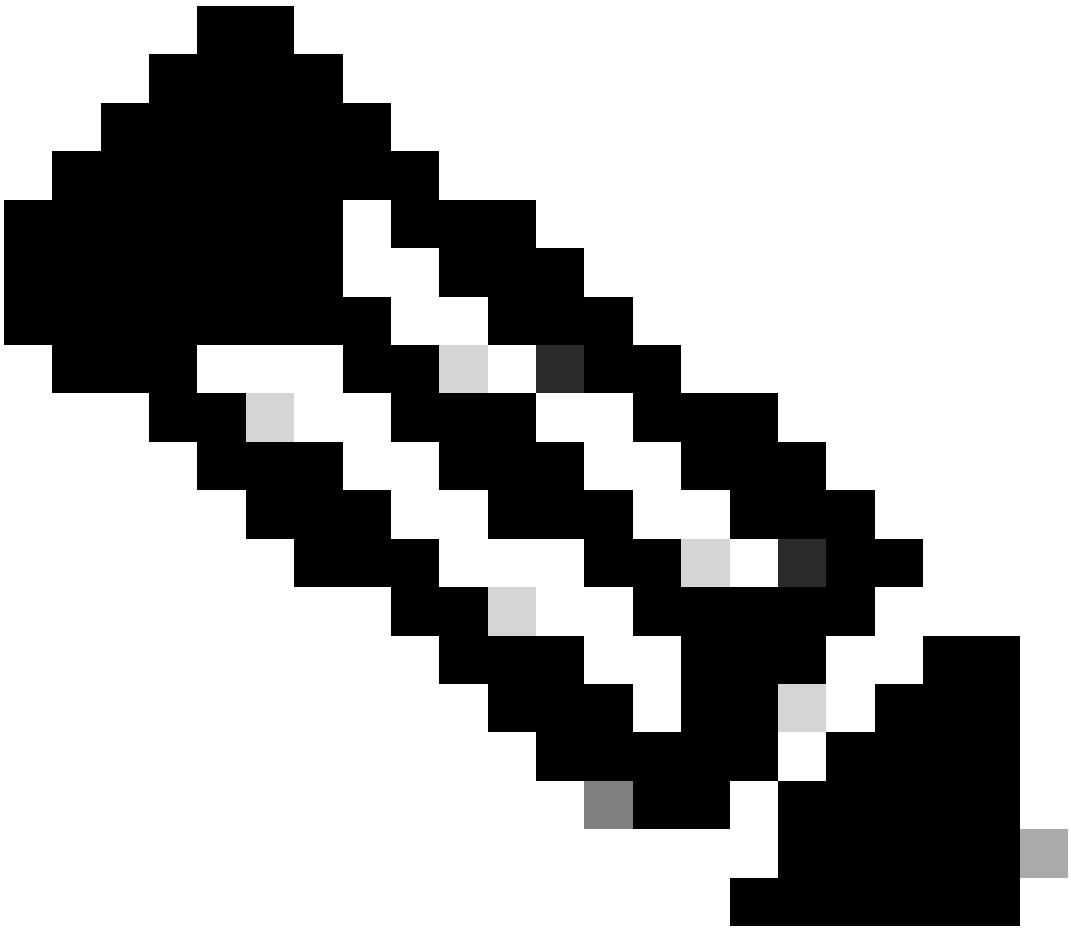
Cancel

Launch instance

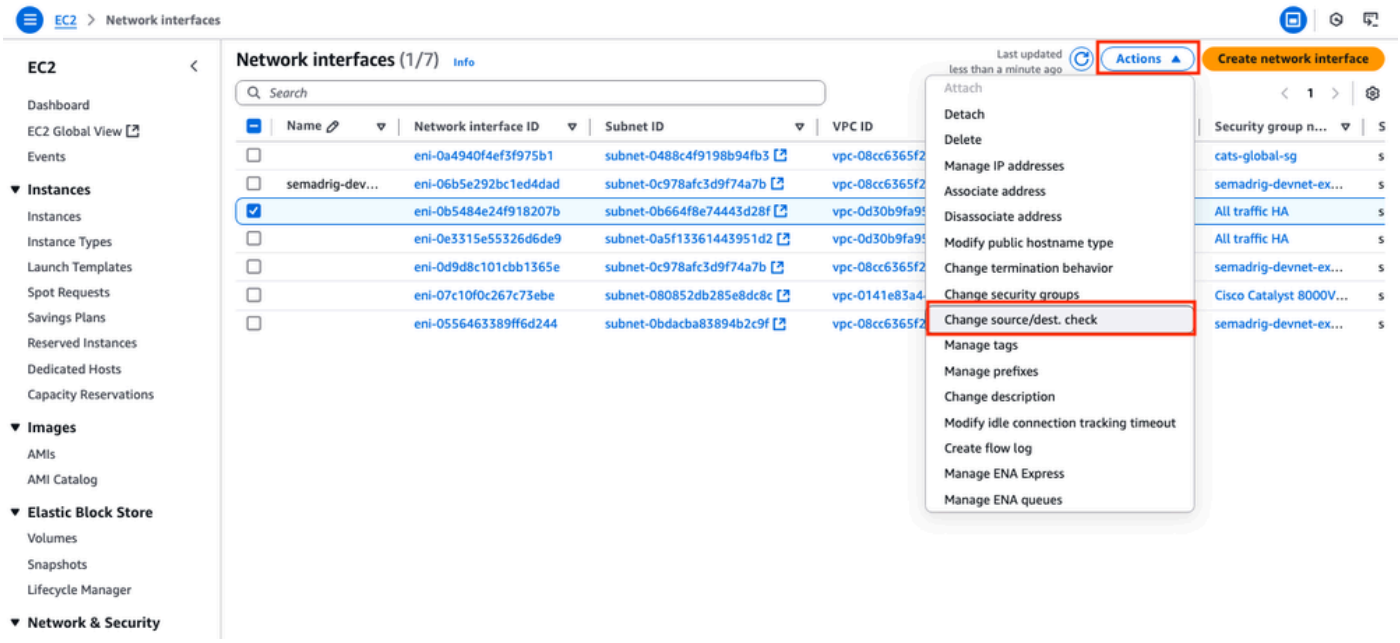
Preview code

ステップ 6.7：ENIでの送信元/宛先チェックの無効化

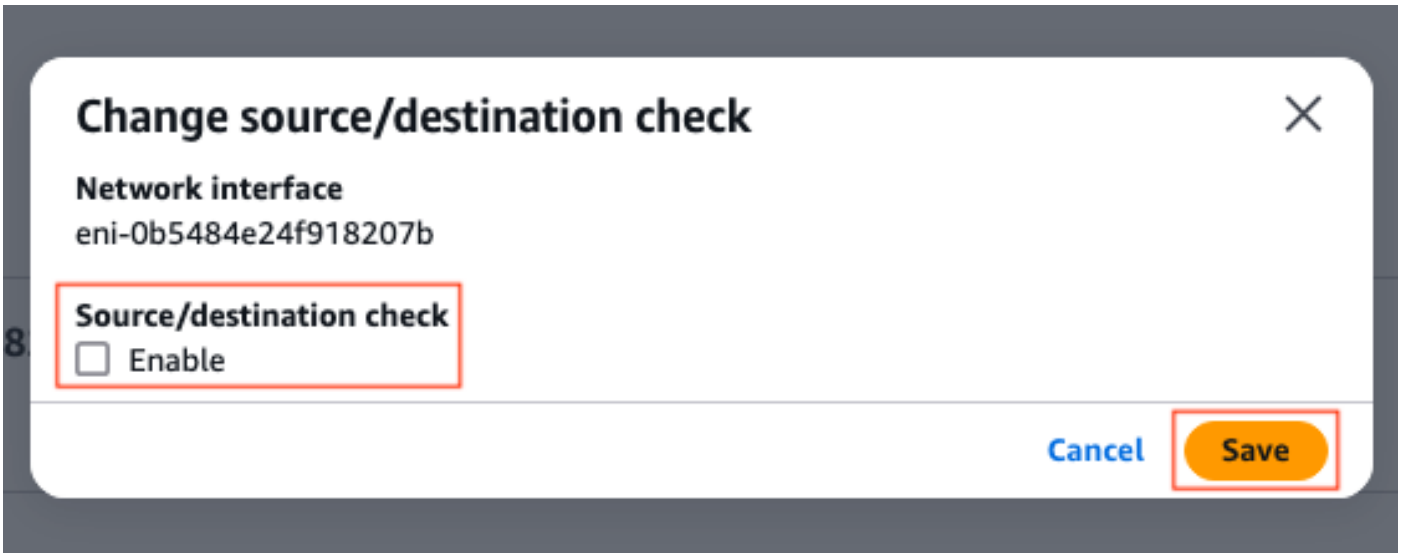
インスタンスが作成されたら、AWSでsrc/dstチェック機能を無効にして、同じサブネットのインターフェイス間の接続を取得します。EC2 Dashboard > Network & Security > Network interfacesセクションで、ENIsを選択して、Actionsをクリックします > Change source/dest. checkを選択します。



注：このオプションを使用できるようにするには、ENIを1つずつ選択する必要があります。



新しいウィンドウが表示されます。新しいメニューでEnableチェックボックスをオフにして、Saveをクリックします。

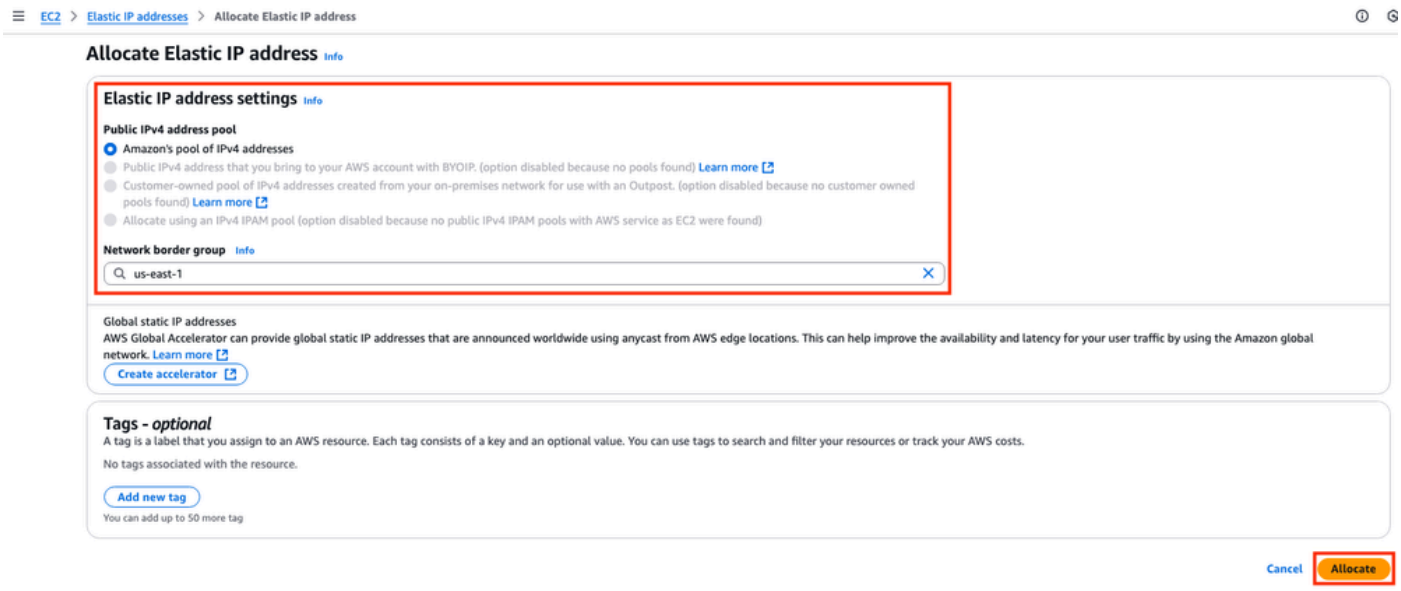


ステップ 6.8：インスタンスのパブリックENIへのElastic IPの作成と関連付け

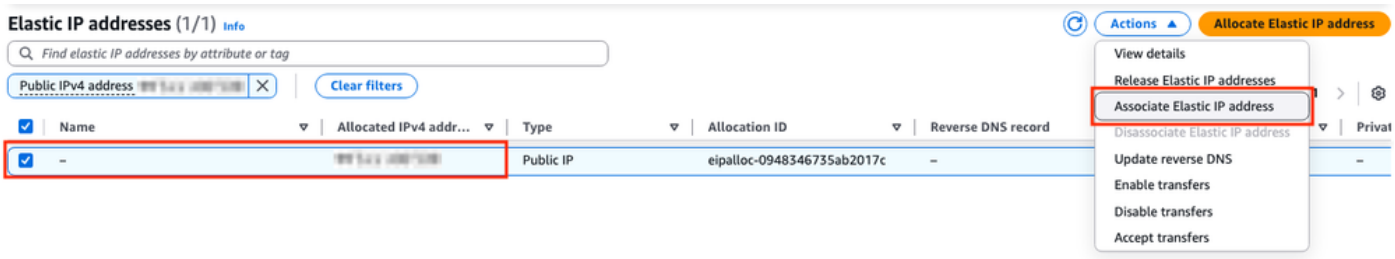
EC2ダッシュボード>ネットワークとセキュリティ> Elastic IPsセクションで、Allocate Elastic IP addressをクリックします。

別のセクションに移動します。この例では、アベイラビリティーゾーンus-east-1とともにAmazon pool of IPv4 addressesオプションが選択されています。終了したら、Allocateをクリッ

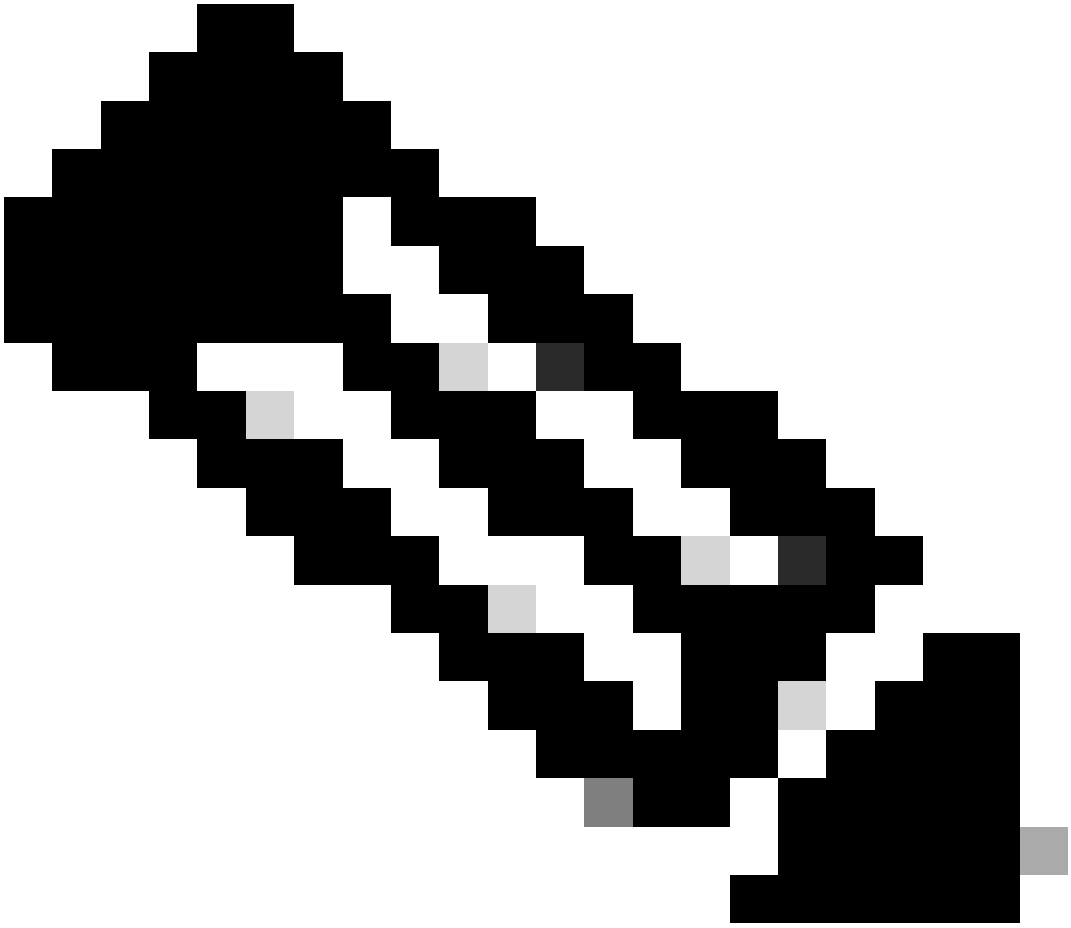
クします。



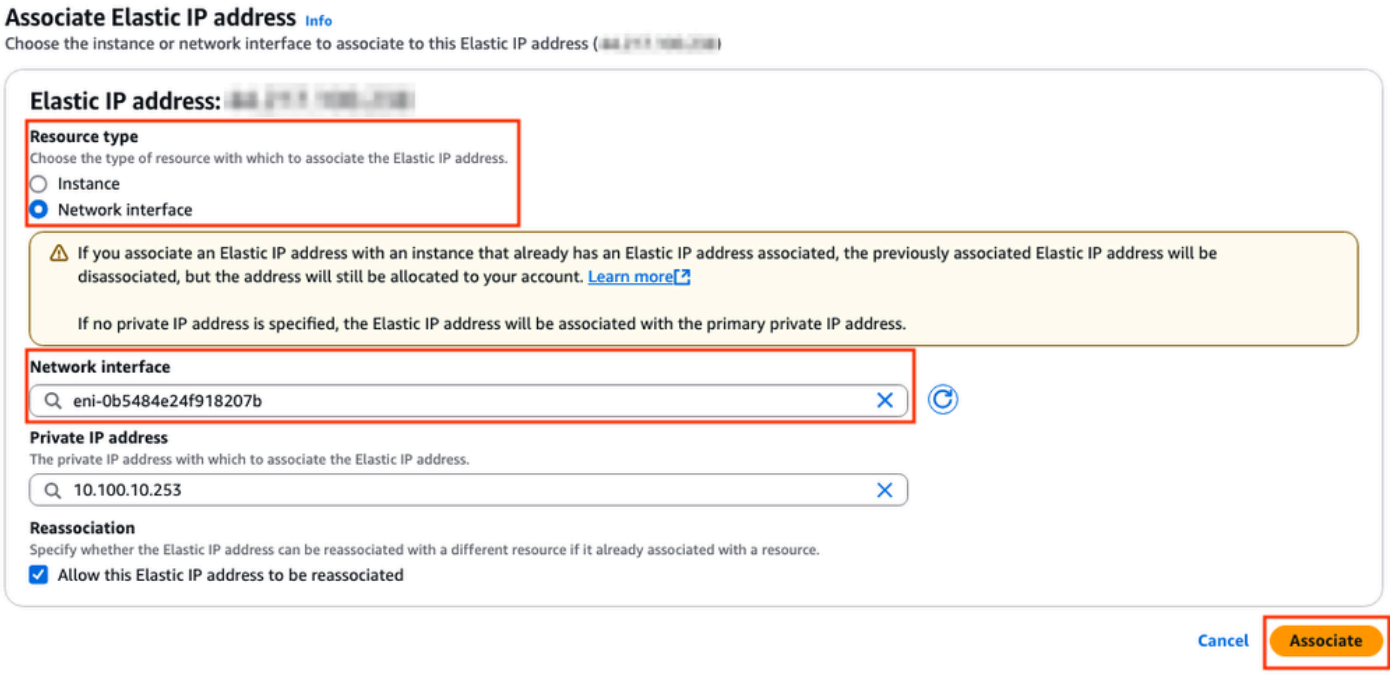
IPアドレスが作成されたら、インスタンスのパブリックインターフェイスにIPアドレスを割り当てます。EC2 Dashboard > Network & Security > Elastic IPsセクションで、Actions > Associate Elastic IP addressの順にクリックします。



この新しいセクションでは、Network interfaceオプションを選択し、対応するインターフェイスのパブリックENIを探します。対応するパブリックIPアドレスを関連付け、Associateをクリックします。



注：適切なENI IDを取得するには、EC2ダッシュボード>インスタンスセクションに移動します。次にインスタンスを選択し、Networkingセクションにチェックマークを入れます。パブリックインターフェイスのIPアドレスを探して、同じ行のENI値を取得します。



手順 7：手順6を繰り返して、HA用に2つ目のC8000vインスタンスを作成します

このドキュメントの「トポロジ」のセクションを参照して各インターフェイスに対応する情報を把握し、6.1から6.6に同じ手順を繰り返してください。

ステップ 8：手順6を繰り返して、AMI MarketplaceからVM(Linux/Windows)を作成します

この例では、Ubuntuサーバ22.04.5 LTSがAMI Marketplaceから内部ホストとして選択されています。

ens5はデフォルトでパブリックインターフェイス用に作成されます。この例では、プライベートサブネット用の2番目のインターフェイス（デバイス上のens6）を作成します。

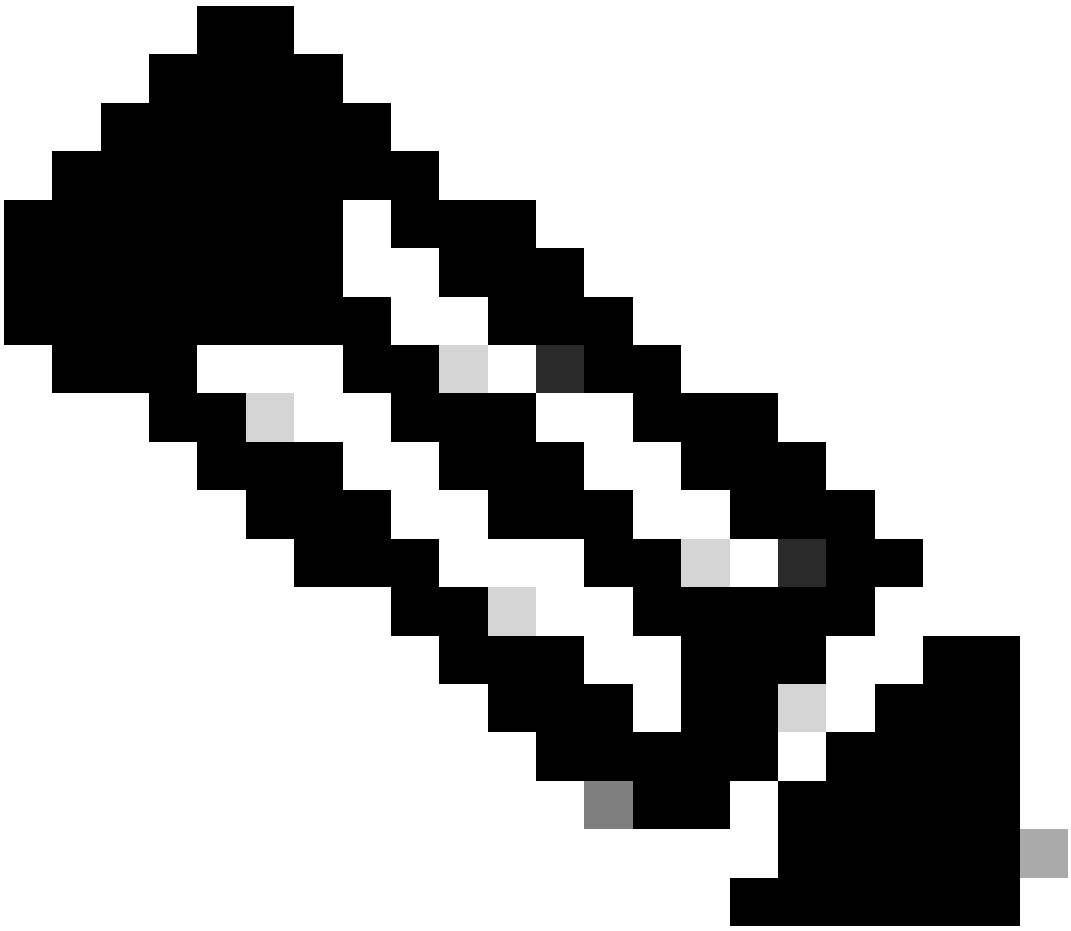
<#root>

```
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
```

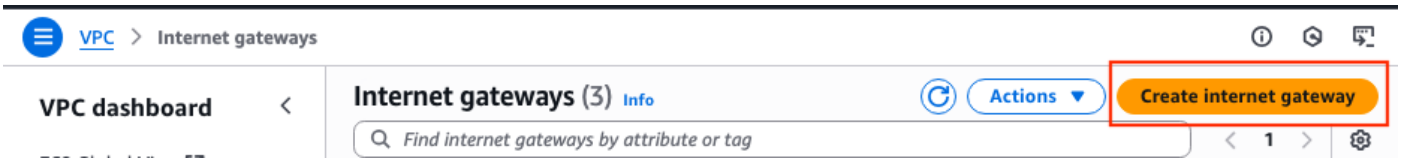
```
10.100.130.254

 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
```



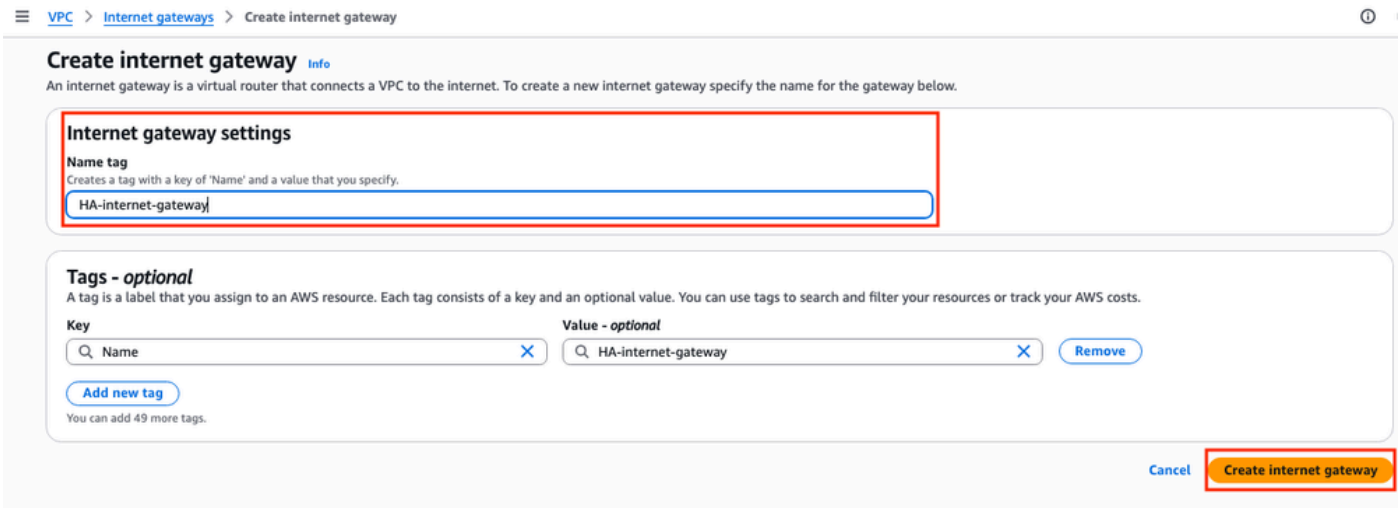
注：インターフェイスに何らかの変更が加えられた場合は、インターフェイスのフラップまたはVMのリロードを実行して、これらの変更を適用してください。

ステップ 9：VPC用のインターネットゲートウェイ(IGW)の作成と設定

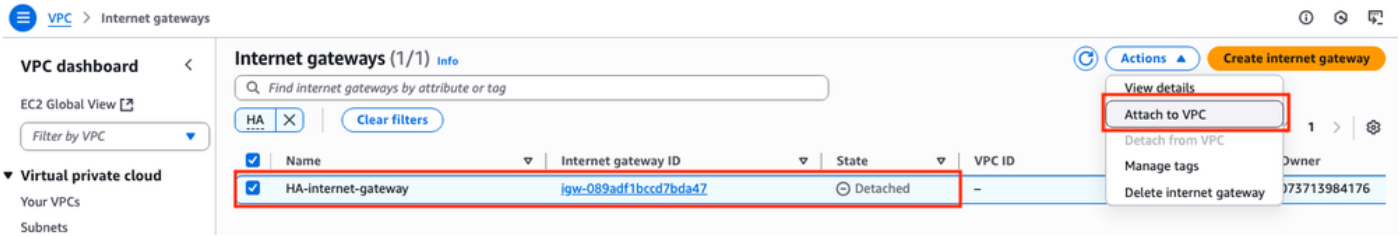
VPCダッシュボード**>**仮想プライベートクラウド**>**インターネットゲートウェイセクションで、インターネットゲートウェイの**作成**をクリックします。



この新しいセクションで、このゲートウェイの**name tag**を作成し、**Create internet gateway**をクリックします。

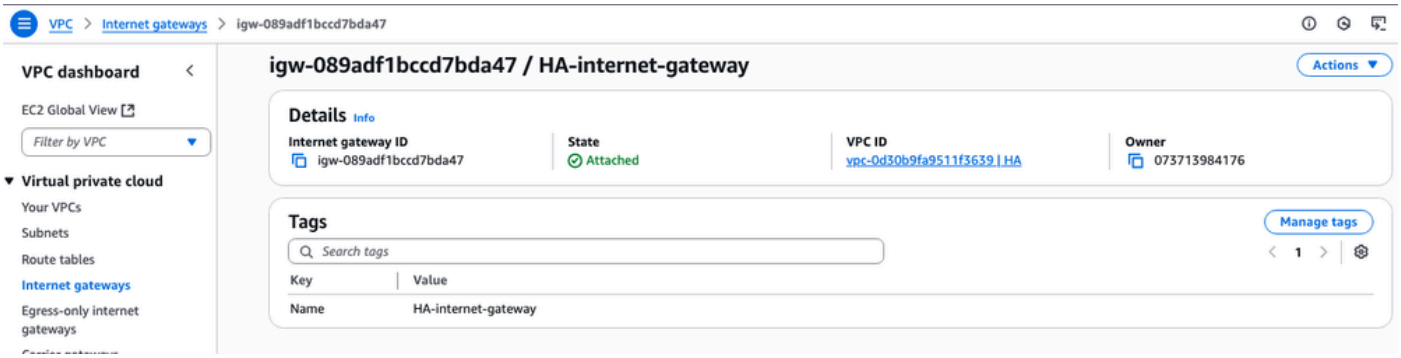


IGWを作成したら、対応するVPCに接続します。**VPC**ダッシュボード>仮想プライベートクラウド>インターネットゲートウェイセクションに移動し、対応するIGWを選択します。**Actions > Attach to VPC**の順にクリックします。



この新しいセクションでは、**HA**という名前のVPCを選択します。 この例では、**Attach internet gateway**をクリックします。

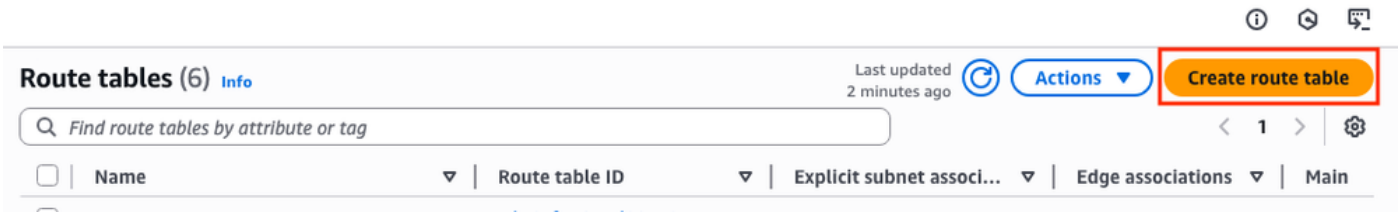
IGWでは、次に示すようにAttachedステートを示す必要があります。



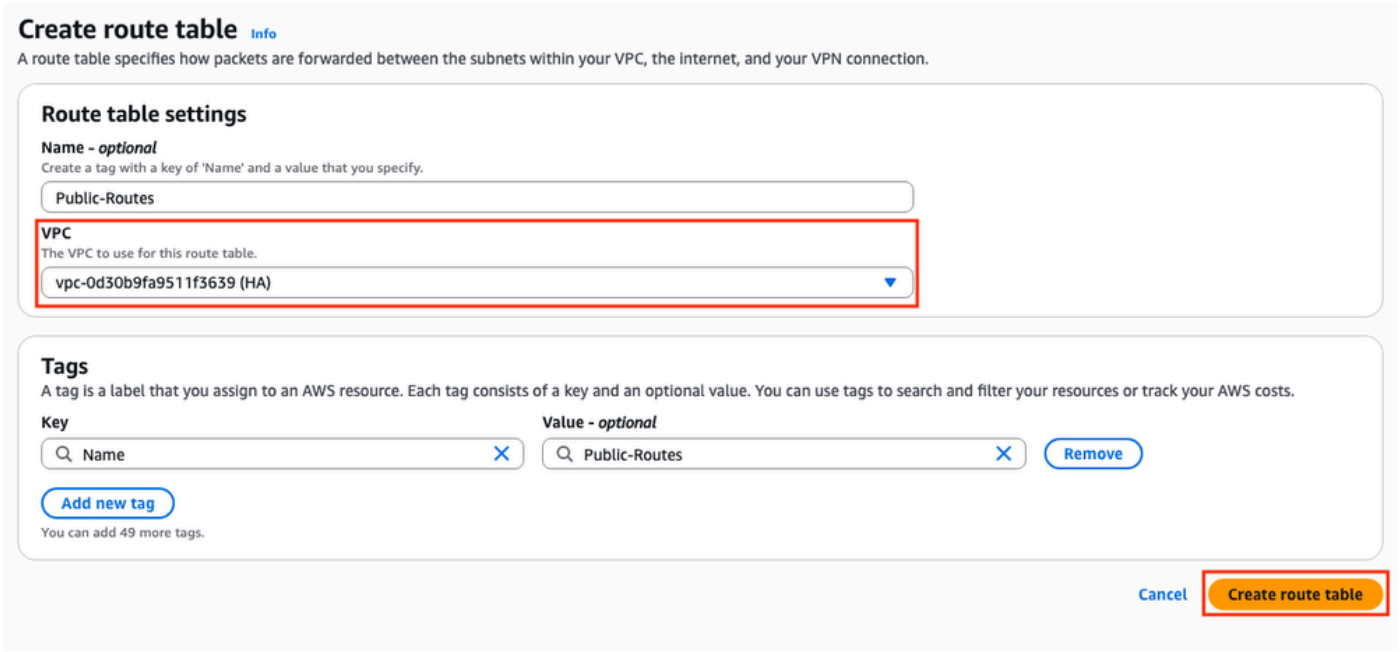
ステップ 10：パブリックおよびプライベートサブネット用のルートテーブルをAWSで作成および設定する

ステップ 10.1：パブリックルートテーブルの作成と設定

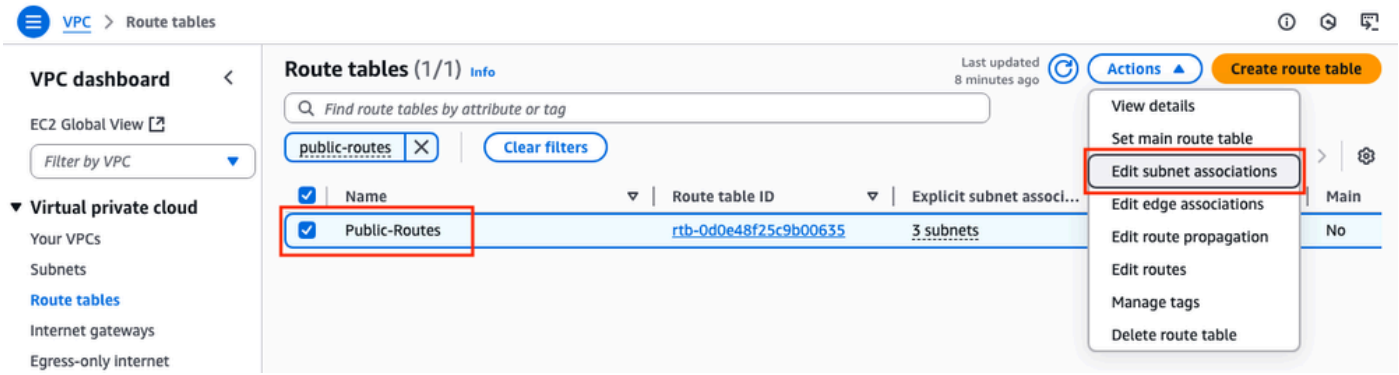
このトポロジでHAを確立するには、すべてのパブリックサブネットとプライベートサブネットを対応するルートテーブルに関連付けます。VPCダッシュボード>仮想プライベートクラウド>ルートテーブルセクションで、ルートテーブルの作成をクリックします。



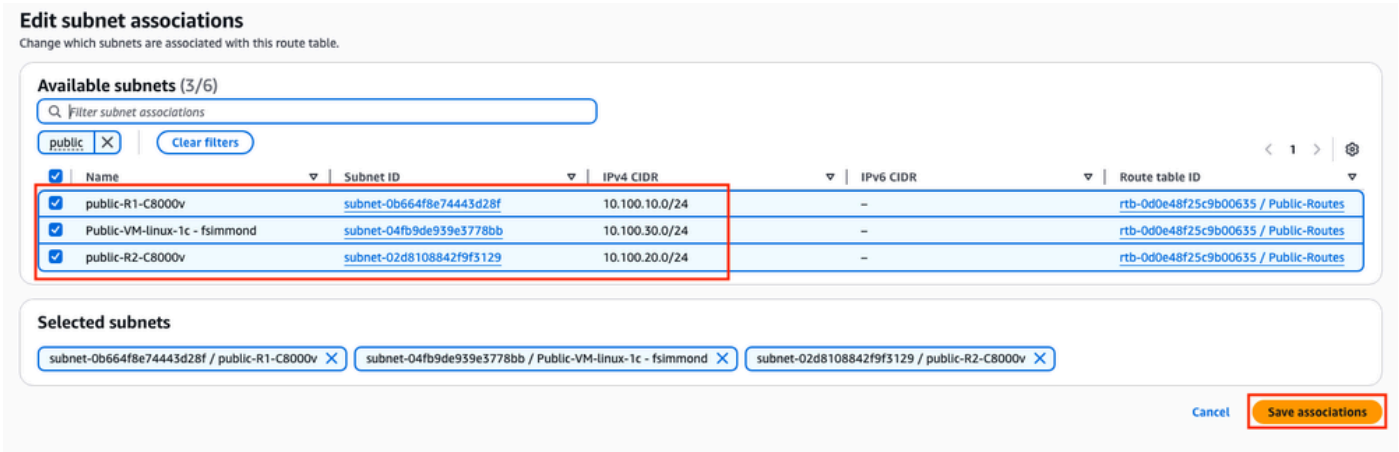
新しいセクションで、このトポロジに対応するVPCを選択します。選択したら、Create route tableをクリックします。



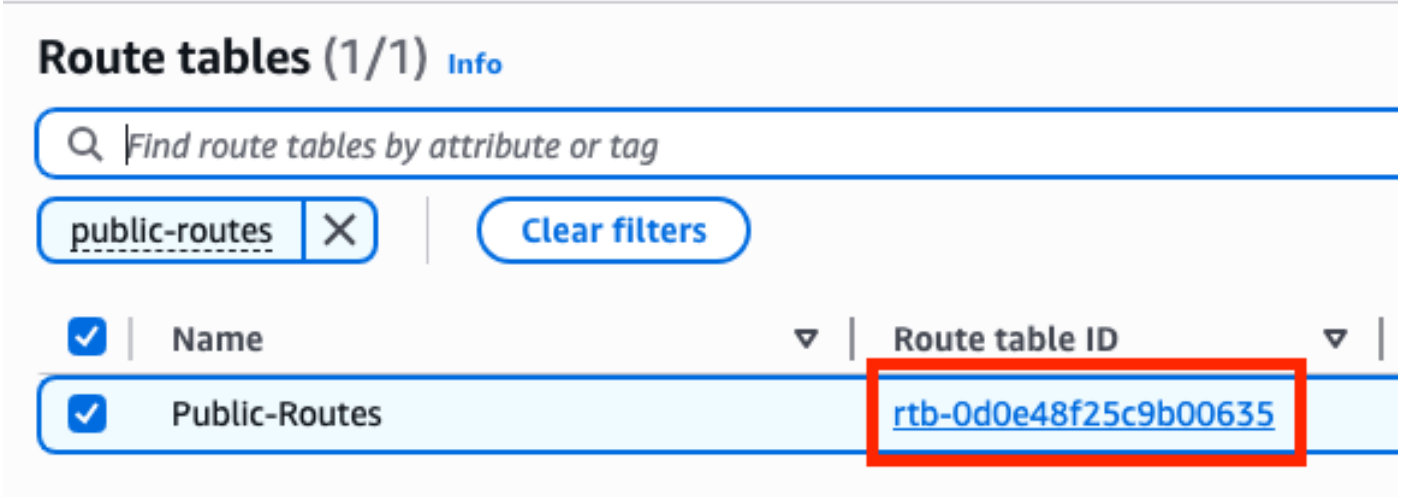
Route tablesセクションで、createdテーブルを選択し、Actions > Edit Subnet associationsの順にクリックします。



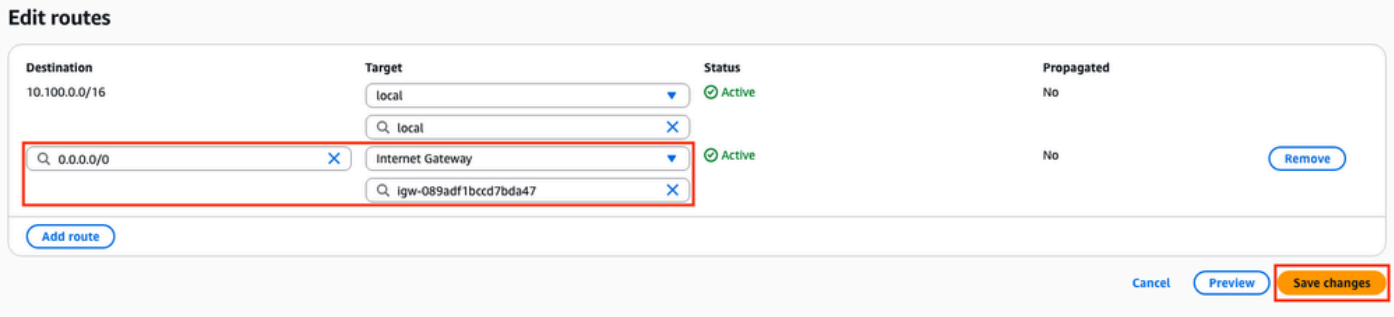
次に、対応するサブネットを選択し、Save associationsをクリックします。



サブネットを関連付けたら、ルートテーブルIDハイパーリンクをクリックして、テーブルに適切なルートを追加します。次に、Edit Routesをクリックします。

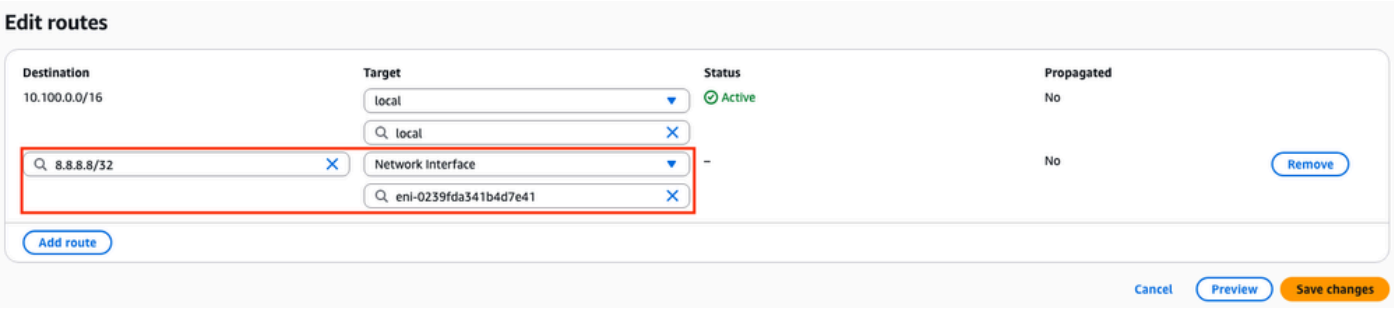


インターネットアクセスを取得するには、Add routeをクリックし、このパブリックルートテーブルと、ステップ9で作成したIGWを、上記のパラメータでリンクします。選択したら、Save changesをクリックします。



ステップ 10.2：プライベートルートテーブルの作成と設定

パブリックルートテーブルが作成されたので、ルートにインターネットゲートウェイを追加する以外は、プライベートルートとプライベートサブネットに手順10を繰り返します。この例では、8.8.8.8のトラフィックはプライベートサブネットを通過する必要があるため、ルーティングテーブルは次のようになります。



ステップ 11基本的なネットワーク設定、ネットワークアドレス変換(NAT)、BFDを使用したGREトンネル、およびルーティングプロトコルの確認と設定

AWSでインスタンスとそのルーティング設定を準備したら、デバイスを設定します。

C8000v R1の設定：

```
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の設定：

```
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
```

ステップ 12ハイアベイラビリティの設定(Cisco IOS® XE Denali 16.3.1a以降)

VM間の冗長性と接続が設定されたので、ルーティングの変更を定義するHA設定を設定します。BFDピアダウンなどのAWS HAエラーの後で設定する必要があるRoute-table-id、Network-interface-id、およびCIDR値を設定します。

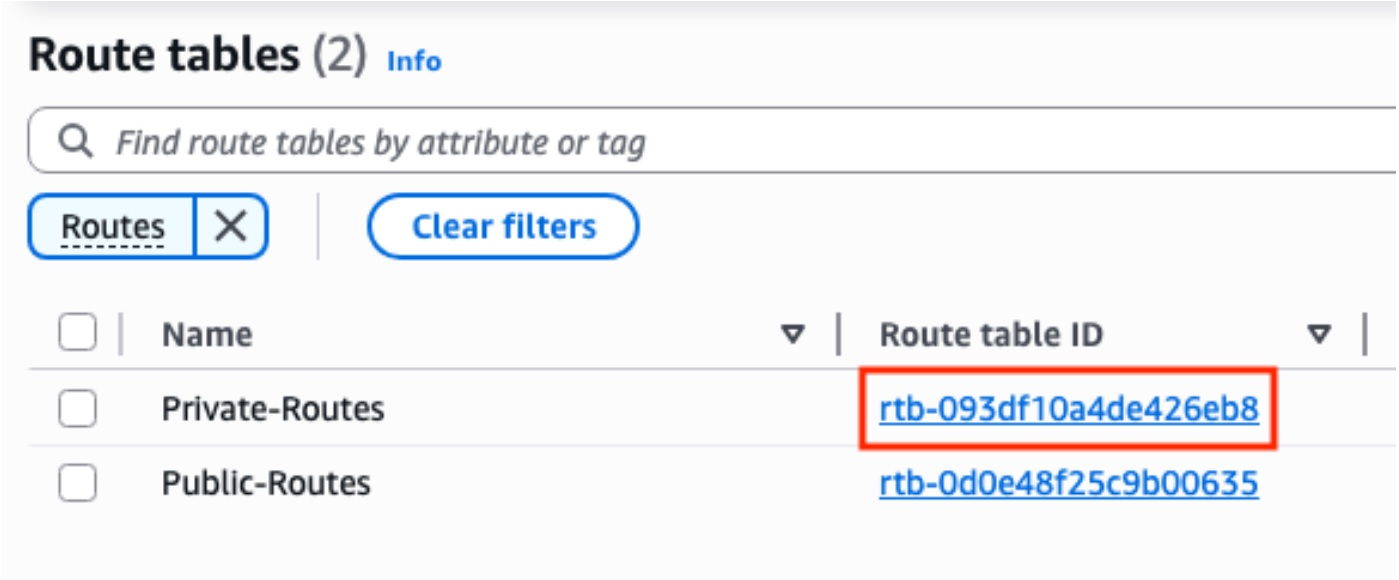
```
Router(config)# redundancy
Router(config-red)# cloud provider aws (node-id)
```

```
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>
```

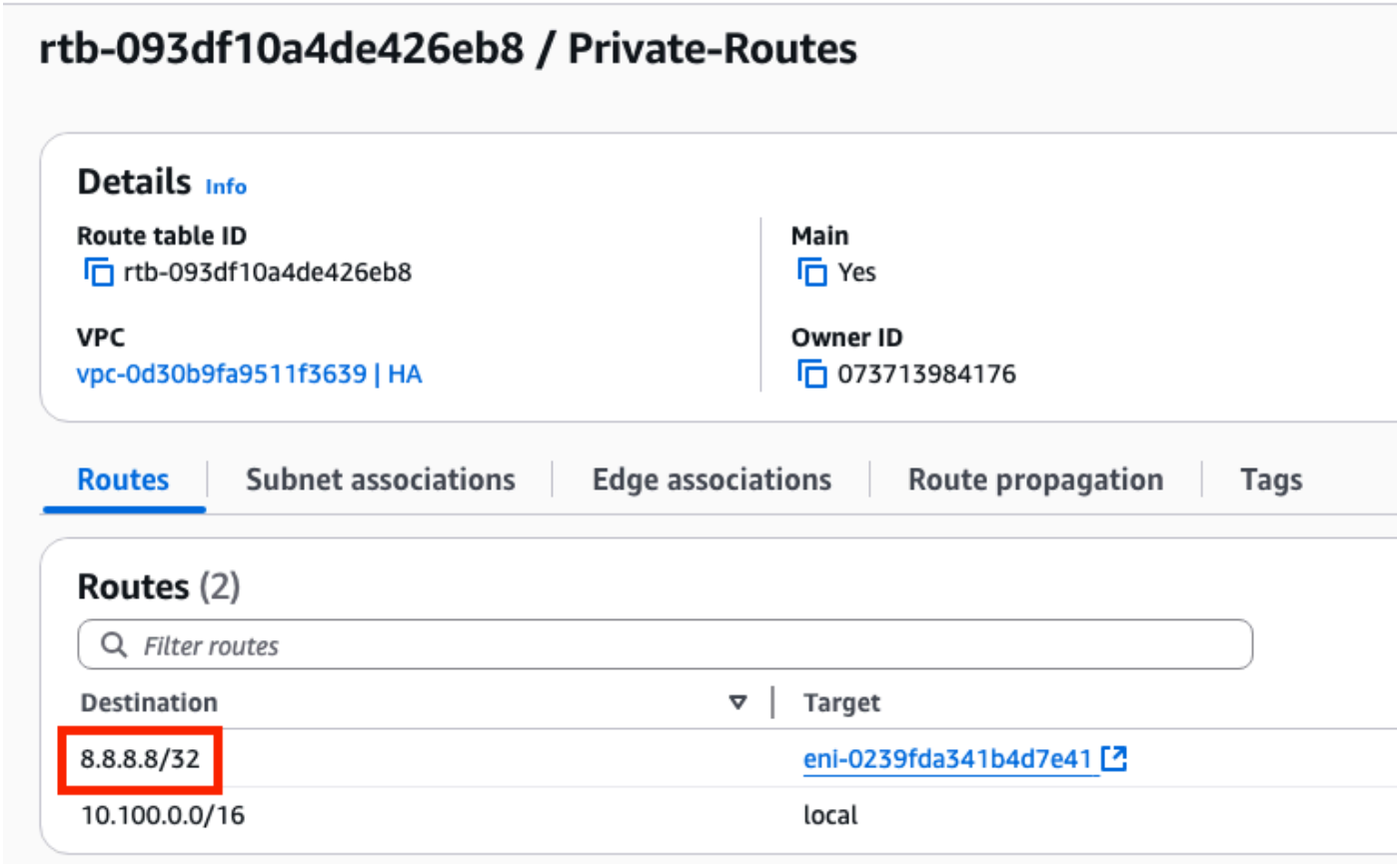
bfd peerパラメータはトンネルピアのIPアドレスに関連しています。これは、show bfd neighborの出力を使用して確認できます。

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

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

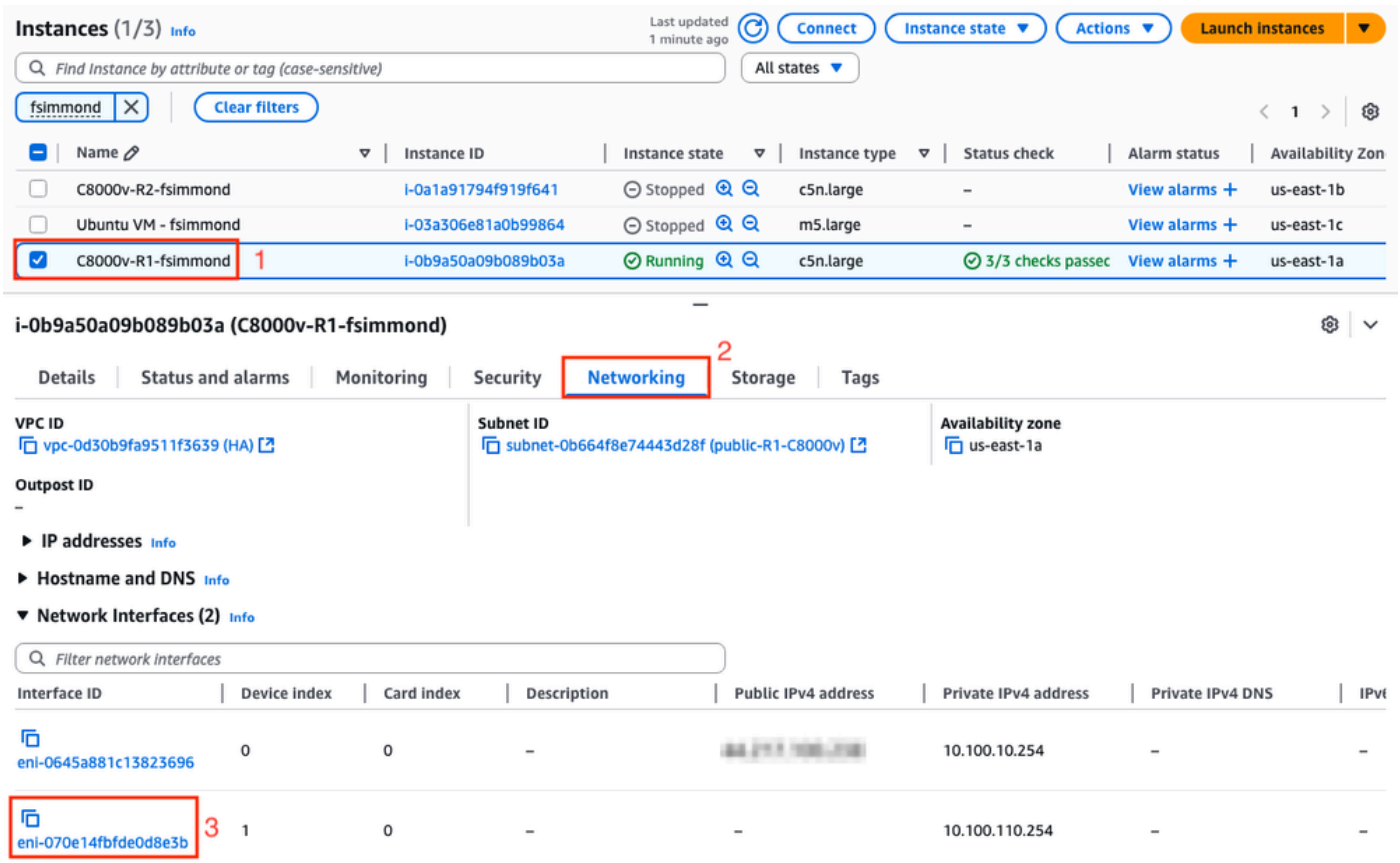
route-tableパラメータは、VPCダッシュボード>仮想プライベートクラウド>ルートテーブルセクションにあるプライベートルートテーブルIDに関連します。対応するルートテーブルIDをコピーします。



cidr ipパラメータは、プライベートルートテーブルに追加されたルートプレフィックス（手順10.2で作成されたルート）に関連します。



eniパラメータは、設定されるインスタンスの対応するプライベートインターフェイスのENI IDに関連します。この例では、インスタンスのインターフェイスGigabitEthernet2のENI IDが使用されます。



regionパラメータは、VPCが配置されているリージョンのAWSドキュメントで見つかったコード名に関連しています。この例では、us-east-1 リージョンが使用されます。

ただし、このリストは変更されたり拡大したりする可能性があります。最新の更新プログラムについては、AmazonRegion and Availabilityゾーンドキュメントを参照してください。

これらの情報をすべて考慮した、VPC内の各ルータの設定例を次に示します。

C8000v R1の設定例：

```
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
```

C8000v R2の設定例：

```
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
```

検証

1. C8000v R1インスタンスのステータスを確認します。トンネルとクラウドの冗長性が稼働していることを確認します。

```
R1#show bfd neighbors

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

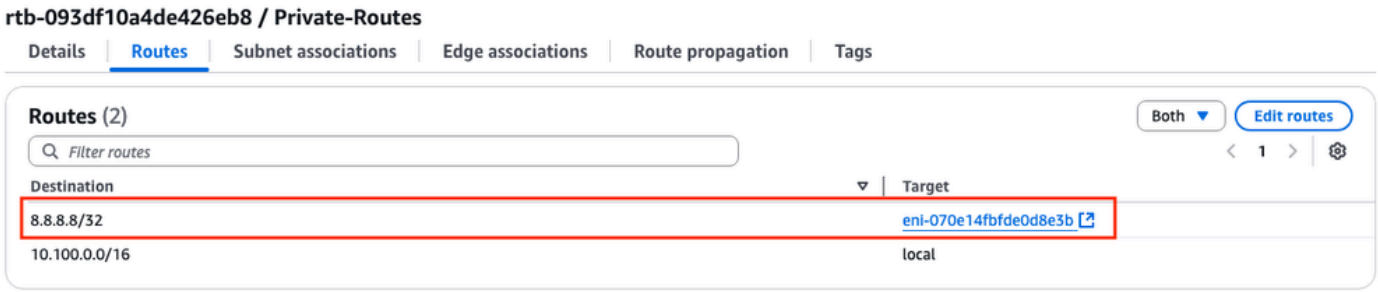
```
R1#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H Address Interface Hold Uptime SRTT RTO Q Seq
(sec) (ms) Cnt Num
0 192.168.200.2 Tu1 10 00:16:52 2 1470 0 2
```

```
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. ルータの背後にあるホストVMから8.8.8.8に対して連続pingを実行します。pingがプライベートインターフェイスを通過することを確認してください。

```
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. AWS WebGUIを開き、ルーティングテーブルのステータスを確認します。現在のENIは、R1インスタンスのプライベートインターフェイスに属しています。



4. HAフェールオーバーイベントをシミュレートするために、R1インスタンスでTunnel1インターフェイスをシャットダウンして、ルート変更をトリガーします。

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

5. AWSのルートテーブルで再度チェックしてください。ENI IDはR2プライベートインターフェイスENI IDに変更されました。

トラブルシュート

次に、展開の再作成時に忘れられたり、設定が誤っていたりすることの多い一般的なポイントの大部分を示します。

- リソースが関連付けられていることを確認します。VPC、サブネット、インターフェイス、ルートテーブルなどを作成するとき、これらの多くは自動的に相互に関連付けられません。彼らはお互いに何の知識も持っていない。
- Elastic IPと任意のプライベートIPが正しいインターフェイスに関連付けられており、正しいサブネットが正しいルートテーブルに追加され、正しいルータに接続され、正しいVPCとゾーンがIAMロールとセキュリティグループにリンクされていることを確認します。

- ENIごとのSource/Destチェックを無効にします。
  このセクションで説明したすべての点を確認しても問題が解決しない場合は、次の出力を収集し、可能であればHAフェールオーバーをテストしてから、Cisco TACでケースをオープンしてください。

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

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。