Guía de implementación de redundancia de HA CSR1000v en Amazon AWS

Contenido

Introducción

Este documento describe la guía de configuración sobre cómo implementar routers CSR1000v para alta disponibilidad en la nube de Amazon AWS. Su objetivo es proporcionar a los usuarios conocimientos prácticos de HA y la capacidad de implementar un banco de pruebas completamente funcional.

Para obtener más información sobre AWS y HA, *consulte* la sección.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Una cuenta de Amazon AWS
- 2 CSR1000v y 1 AMI de Linux/Windows en la misma región
- La versión 1 de HA es compatible con las versiones 16.5 a 16.9 de Cisco IOS-XE®. A partir de 16.11, utilice la versión 3 de HA.

Componentes Utilizados

La información de este documento se basa en Denali 16.7.1 de Cisco IOS-XE®.

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. Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Objetivo

En un entorno de zona de disponibilidad múltiple, simule el tráfico continuo desde el Data Center privado (VM) a Internet. Simule una conmutación por fallo de HA y observe que HA funciona correctamente cuando se confirma el tráfico de switches de la tabla de routing de CSRHA a la interfaz privada de CSRHA1.

Topología

Antes de que se inicie la configuración, es importante comprender completamente la topología y el diseño. Esto ayuda a resolver cualquier problema potencial más adelante.

Hay varias situaciones de implementación de HA basadas en los requisitos de la red. Para este ejemplo, la redundancia de HA se configura con estas configuraciones:

- 1x - Región
- 1x - VPC
- 3x - Zonas de disponibilidad
- 6x - Interfaces/subredes de red (3x de orientación pública/3x de orientación privada)
- 2x - Tablas de ruta (pública y privada)
- 2 routers CSR1000v (Cisco IOS-XE® Denali 16.3.1a o posterior)
- 1x - VM (Linux/Windows)

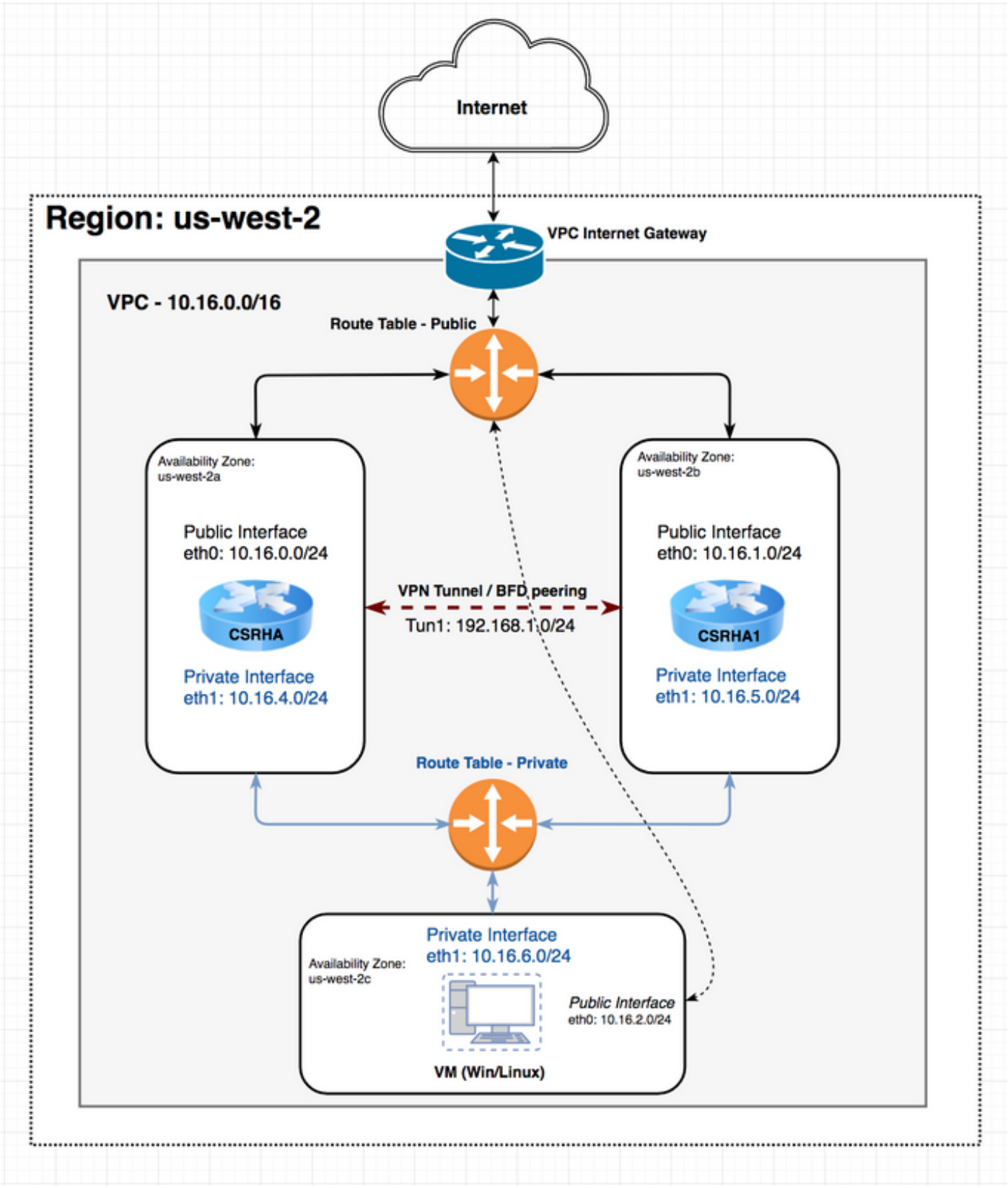
Hay dos routers CSR1000v en un par HA, en dos zonas de disponibilidad diferentes. Considere cada zona de disponibilidad como un Data Center independiente para disfrutar de una resistencia de hardware adicional.

La tercera zona es una VM, que simula un dispositivo en un Data Center privado. Por ahora, el acceso a Internet está habilitado a través de la interfaz pública en para que pueda acceder y configurar la VM. Generalmente, todo el tráfico normal debe fluir a través de la tabla de rutas

privadas.

Haga ping en la interfaz privada de la máquina virtual → tabla de rutas privadas → CSRHA → 8.8.8.8 para la simulación de tráfico. En un escenario de failover, observe que la tabla de ruta privada ha conmutado la ruta para señalar a la interfaz privada de CSRHA1.

Diagrama de la red



Terminology

RTB: ID de la tabla de rutas.

CIDR: dirección de destino para la ruta que se actualizará en la tabla de rutas.

ENI: ID de la interfaz de red de la interfaz gigabit CSR 1000v a la que se enruta el tráfico.
Por ejemplo, si CSRHA falla, CSRHA1 toma el control y actualiza la ruta en la tabla de rutas AWS para que apunte a su propio ENI.

REGIÓN: la región AWS de CSR 1000v.

Restricciones

- Para las subredes privadas, no utilice la dirección IP 10.0.3.0/24, que se utiliza internamente en Cisco CSR 1000v para alta disponibilidad. Cisco CSR 1000v necesita tener accesibilidad a Internet pública para realizar llamadas API REST que cambien la tabla de rutas AWS.

- No coloque la interfaz gig1 del CSR1000v dentro de un VRF. La HA no funciona de otro modo.

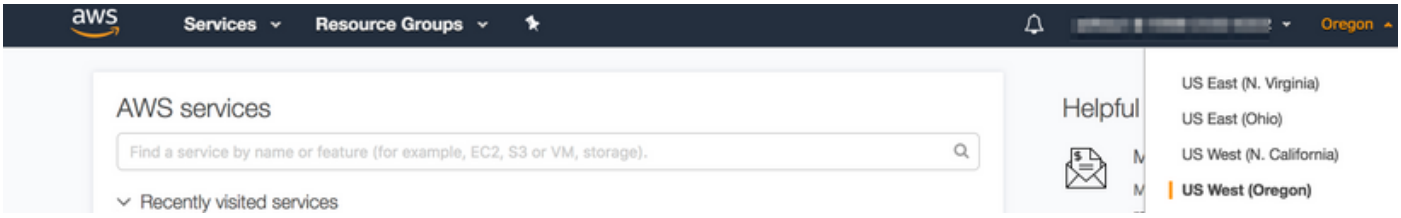
Configuración

El flujo general de configuración consiste en comenzar por la función más completa (Región/VPC) y descender hasta la más específica (Interfaz/subred). Sin embargo, no existe un orden específico de configuración. Antes de comenzar, es importante entender la topología primero .

Consejo: Dé nombres a todos sus parámetros (VPC, interfaz, subred, tablas de rutas, etc.).

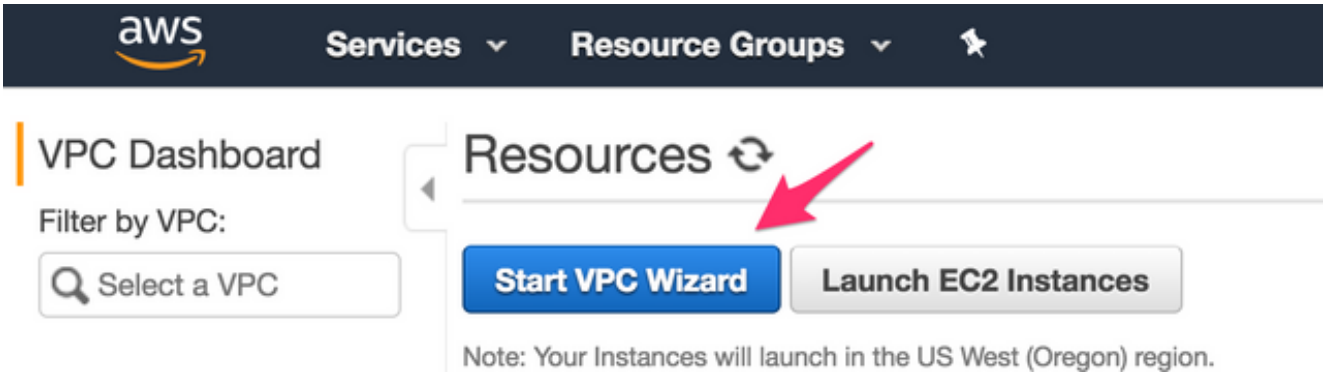
Paso 1. Seleccione una región.

Este ejemplo utiliza US West (Oregón).

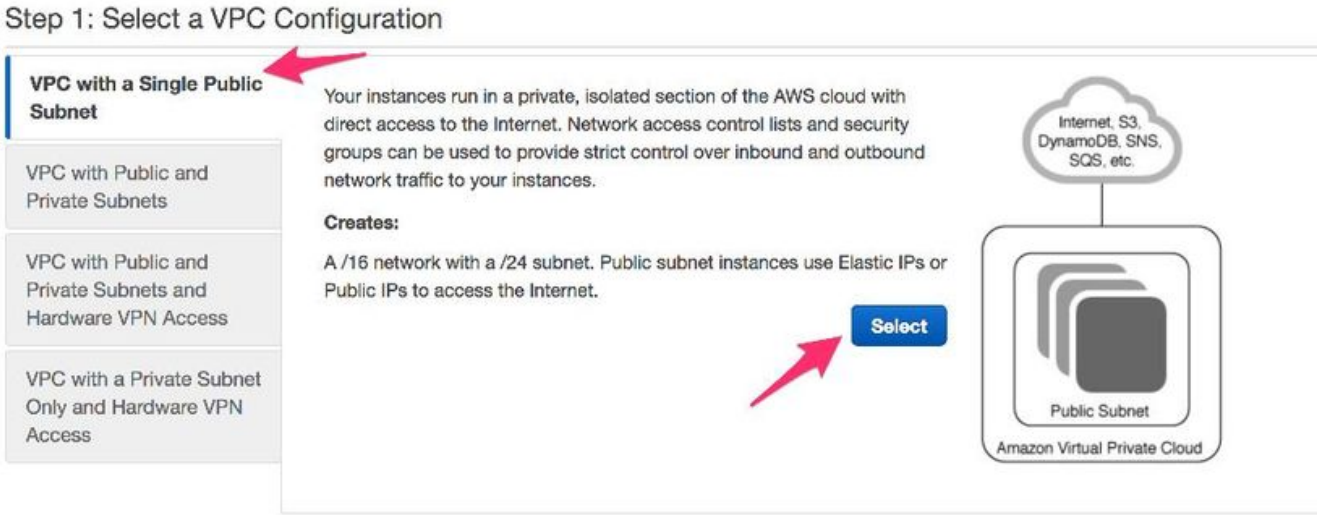


Paso 2. Crear un VPC.

1. En la consola de AWS, navegue hasta **VPC > VPC Dashboard > Start VPC Wizard**.



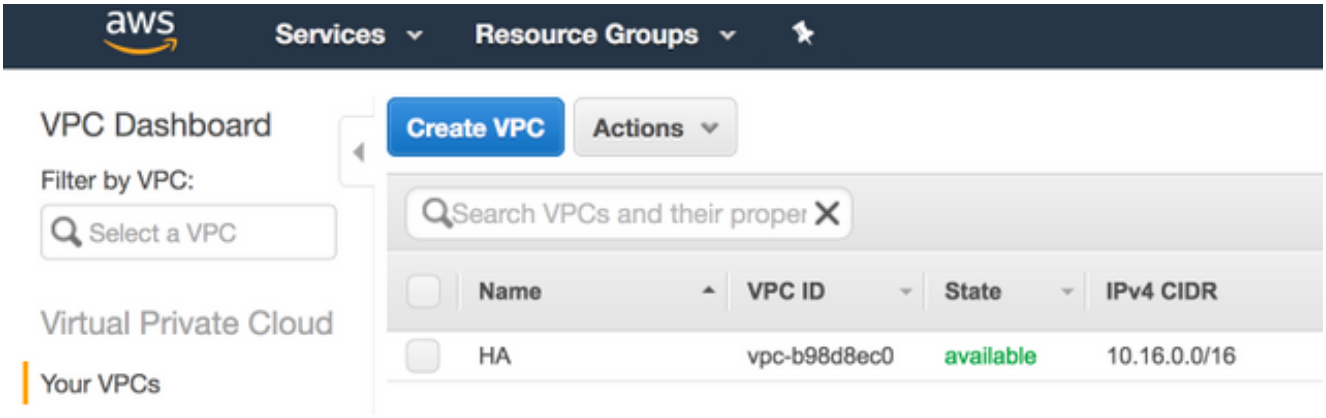
2. Elija VPC con una única subred pública.



3. Al crear una VPC, se le asigna una red /16 para que la utilice como desee.

4. También se le asigna una subred pública /24. Las instancias de subred pública utilizan IP elásticas o IP públicas para que los dispositivos accedan a Internet.

5. se crea vpc-b98d8ec0.



Paso 3. Crear un grupo de seguridad para VPC.

Los grupos de seguridad son como las ACL para permitir o denegar el tráfico.

1. En Seguridad, haga clic en **Grupos de seguridad** y en **Crear su grupo de seguridad** asociado con el VPC creado anteriormente llamado HA.

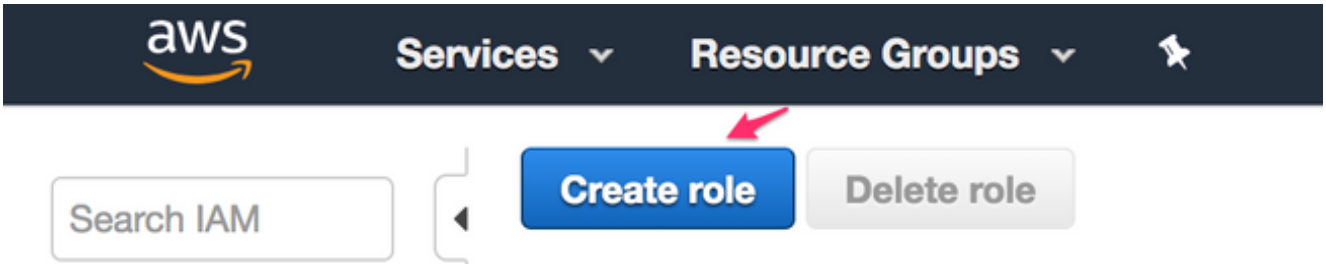
2. En Reglas de entrada, defina qué tráfico desea permitir para sg-1cf47d6d. En este ejemplo, se permite todo el tráfico.

Paso 4. Crear un rol IAM con una política y asociarlo a VPC.

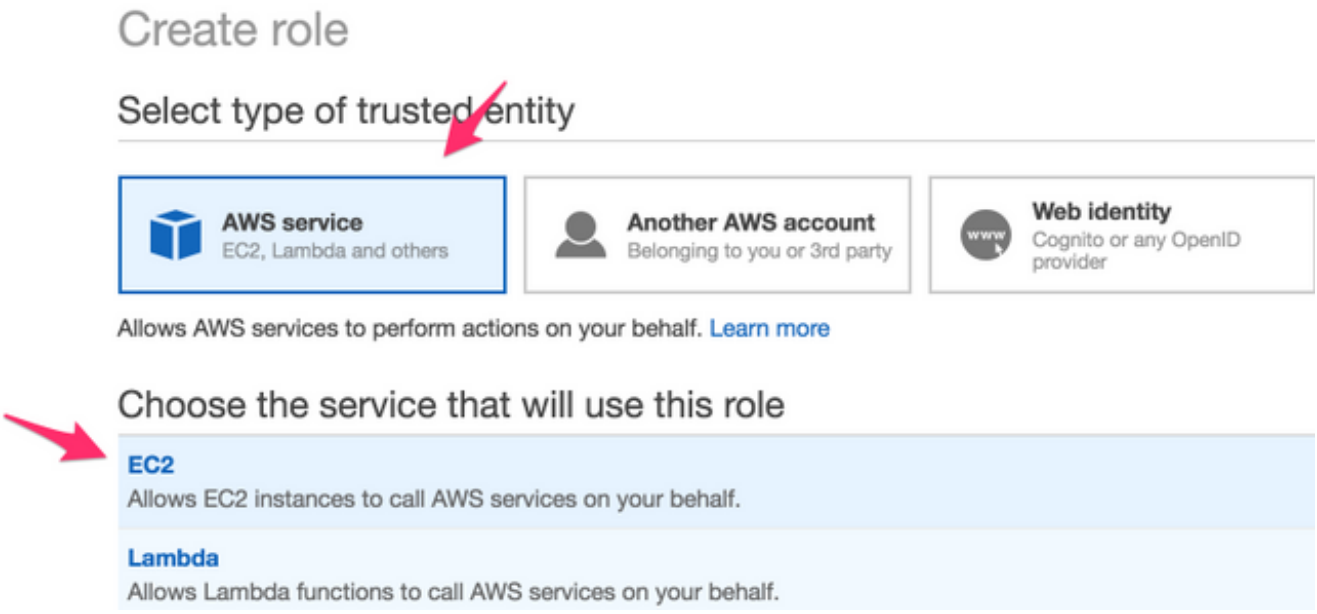
IAM le otorga a su CSR acceso a las API de Amazon.

El CSR1000v se utiliza como proxy para llamar a los comandos API de AWS para modificar la tabla de rutas. De forma predeterminada, los AMI no pueden acceder a las API. Este procedimiento crea un rol IAM y este rol se utiliza durante el inicio de una instancia CSR. IAM proporciona las credenciales de acceso para que los CSR utilicen y modifiquen las API de AWS.

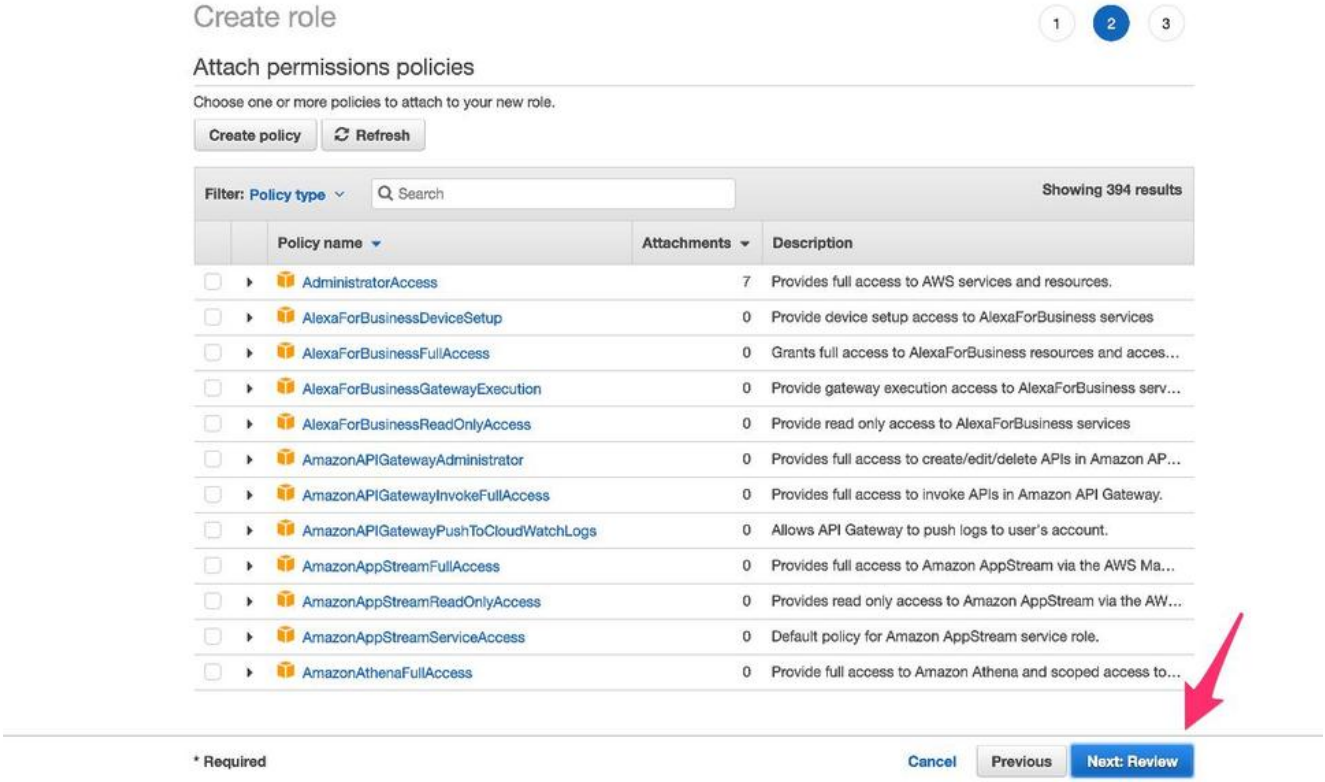
1. Cree el rol IAM. Vaya al panel de IAM y vaya a **Roles > Create Role**, como se muestra en la imagen.



2. Como se muestra en la imagen, permita que la instancia EC2 llame a AWS en su nombre.

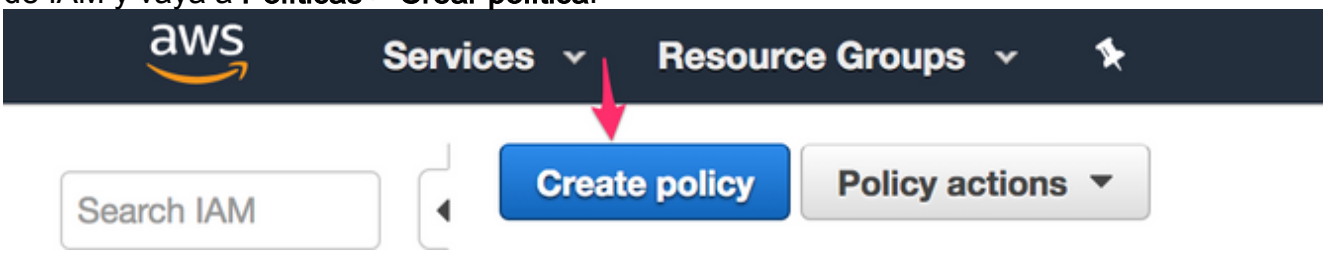


3. Cree un rol y haga clic en **Next (Siguiente): Revise**, como se muestra en la imagen.

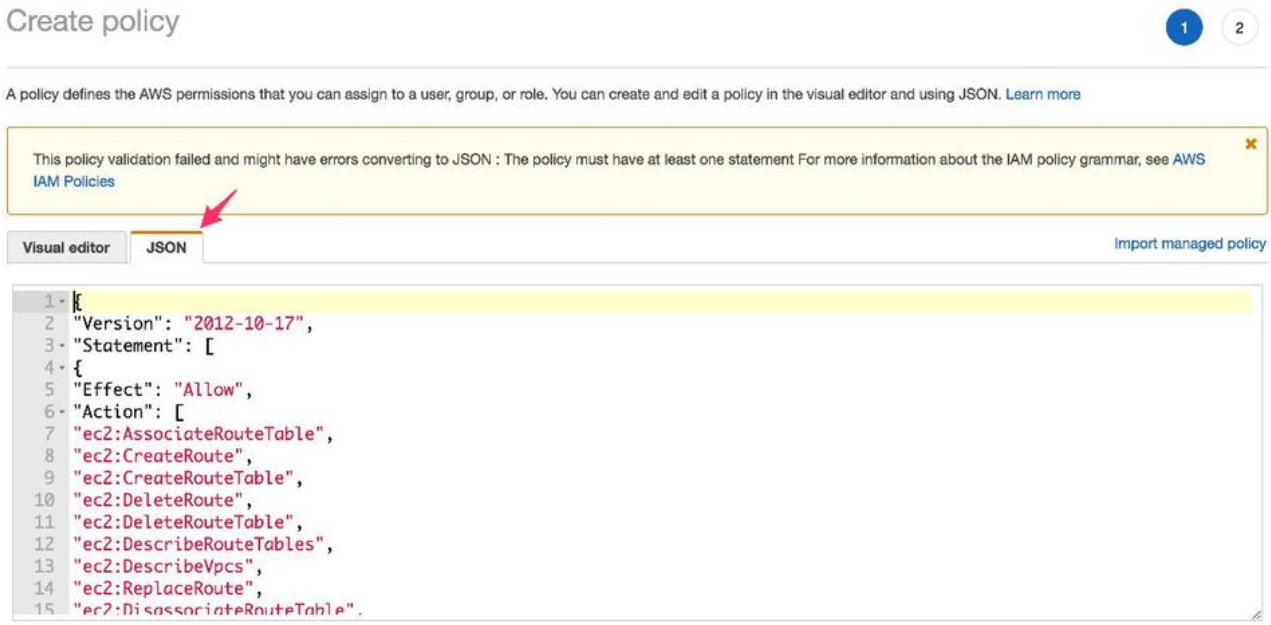


4. Asigne un nombre a la función. Para este ejemplo, como se muestra en la imagen, el nombre del rol es **routetablechange**.

5. A continuación, debe crear una política y asociarla al rol que ha creado anteriormente. Panel de IAM y vaya a **Políticas > Crear política**.

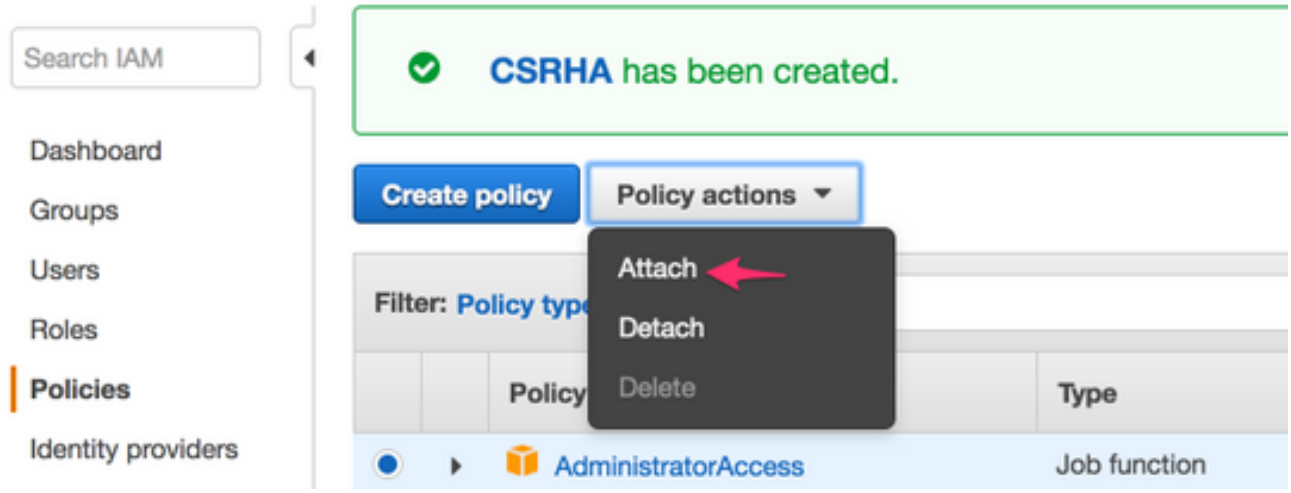



```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"ec2:AssociateRouteTable",
"ec2:CreateRoute",
"ec2:CreateRouteTable",
"ec2:DeleteRoute",
"ec2:DeleteRouteTable",
"ec2:DescribeRouteTables",
"ec2:DescribeVpcs",
"ec2:ReplaceRoute",
"ec2:DisassociateRouteTable",
"ec2:ReplaceRouteTableAssociation"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

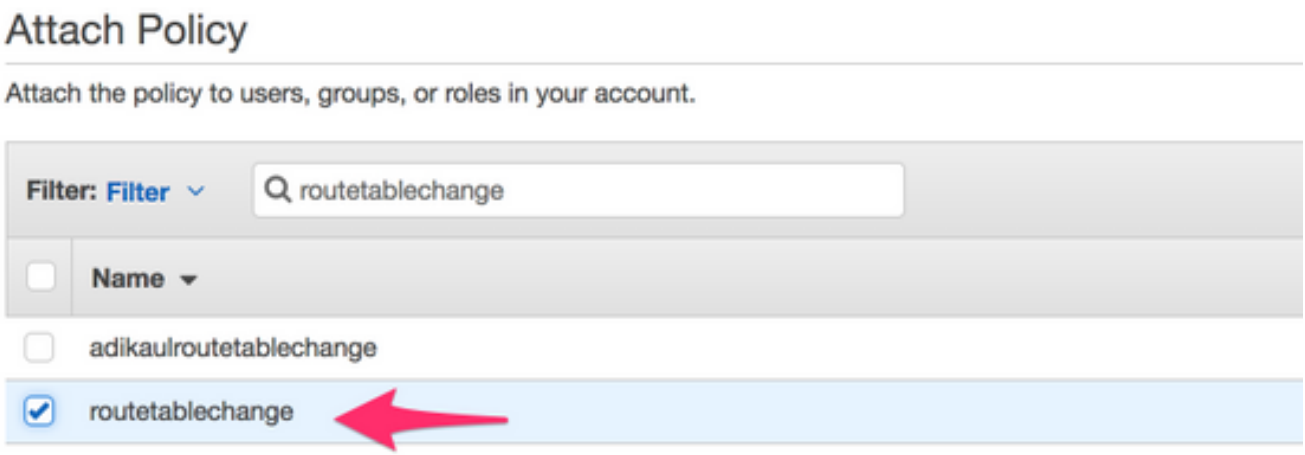


6. Asigne un nombre de directiva y adjúntelo al rol que ha creado. Para este ejemplo, el

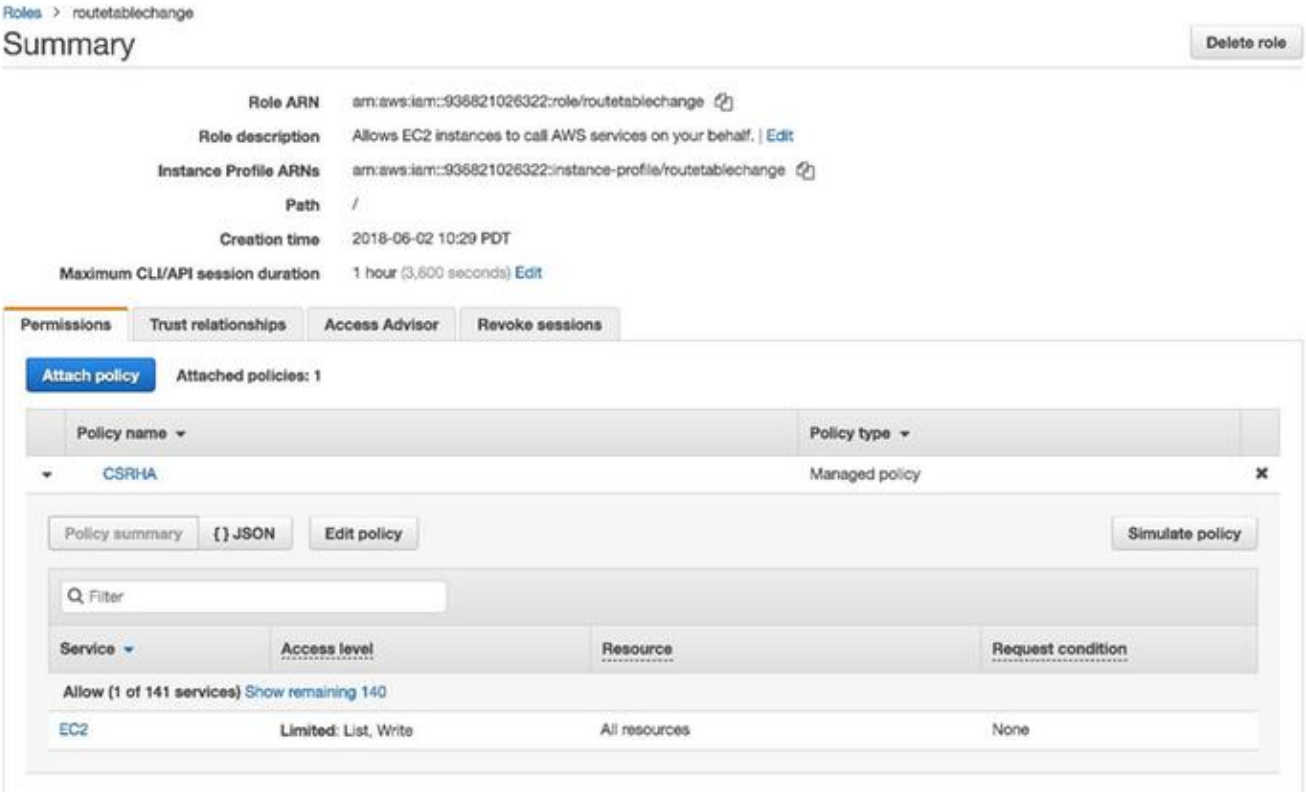
nombre de la política se llama CSRHA con acceso de administrador, como se muestra en la imagen.



7. Como se muestra en la imagen, adjunte la política al rol que creó llamado **routetablechange**.



8. Summary.



Paso 5. Inicie los CSR1000v con la función AMI que creó y asocie las subredes

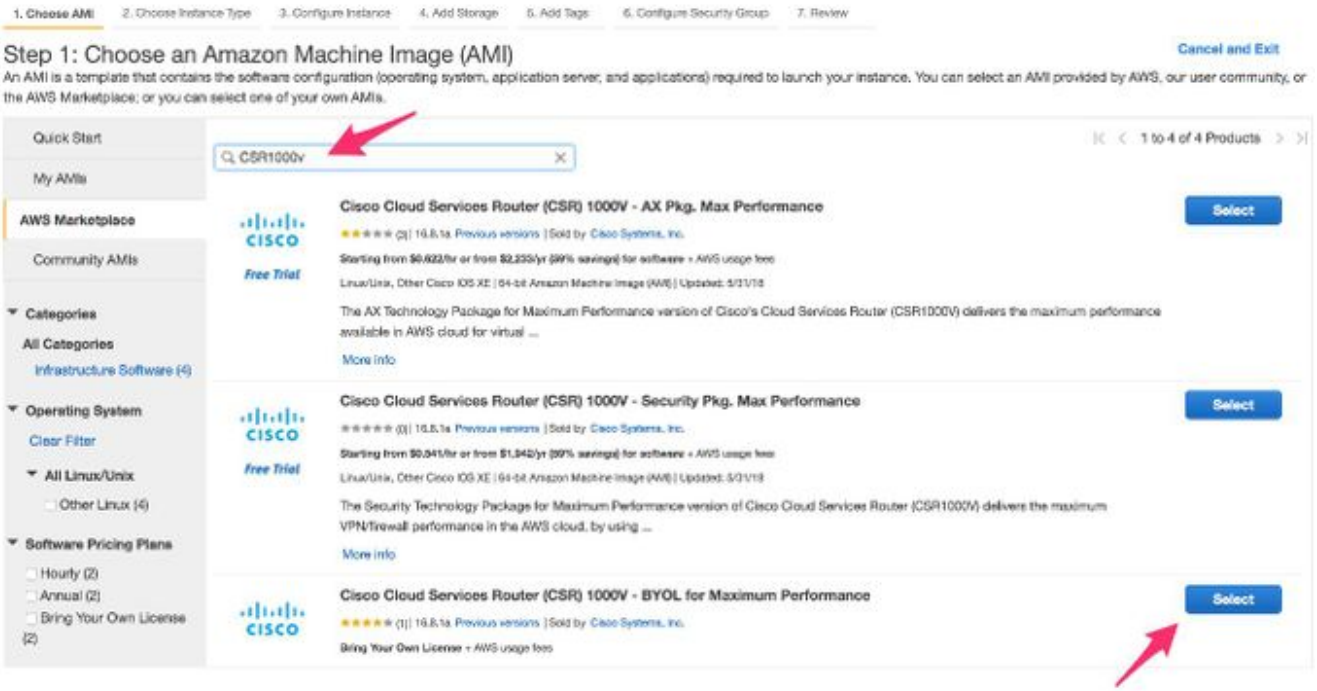
públicas/privadas.

Cada router CSR1000v tiene 2 interfaces (1 pública y 1 privada) y se encuentra en su propia zona de disponibilidad. Puede pensar en cada CSR como si se encontrara en Data Centers independientes.

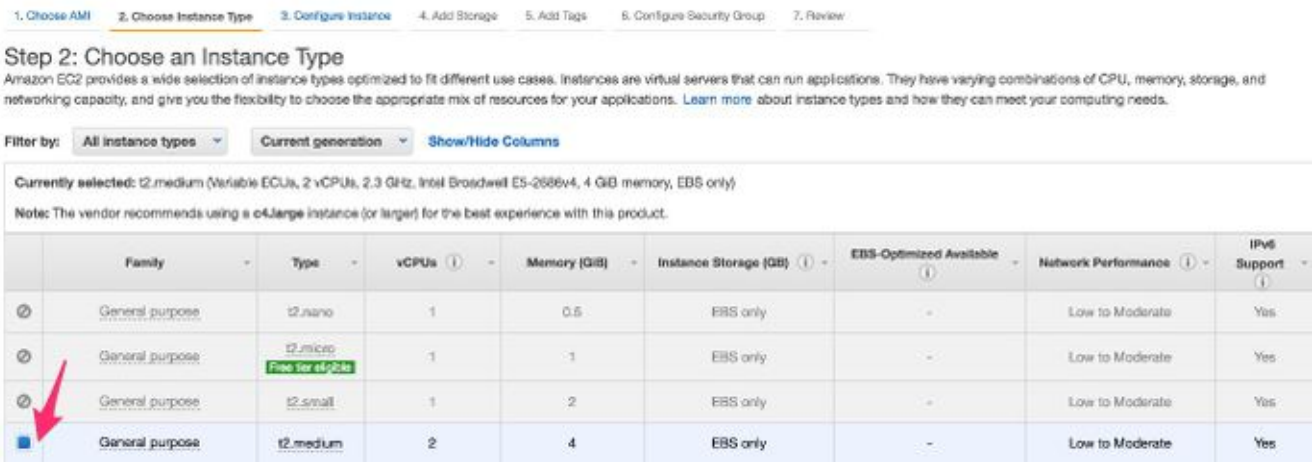
1. En la consola de AWS, seleccione **EC2** y luego haga clic en **Iniciar instancia**.

2. Seleccione AWS Marketplace.

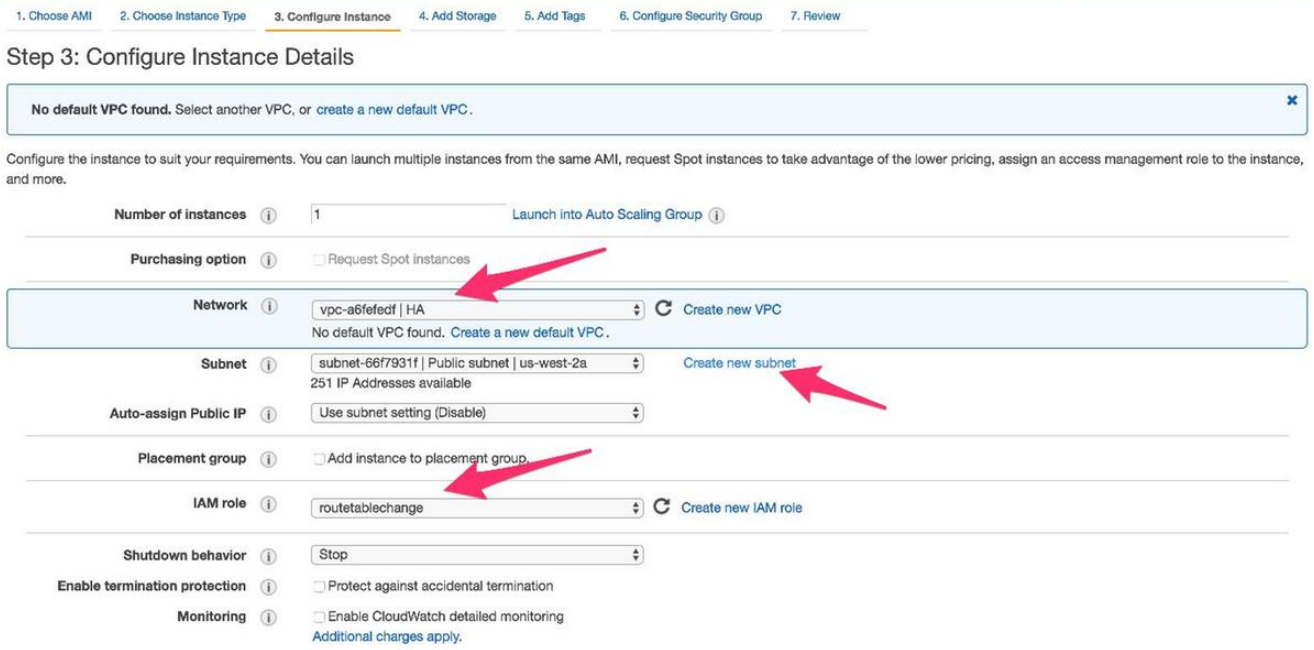
3. Introduzca CSR1000v y en este ejemplo utilizará el router para servicios basados en la nube (CSR) 1000V de Cisco - BYOL para obtener el máximo rendimiento.



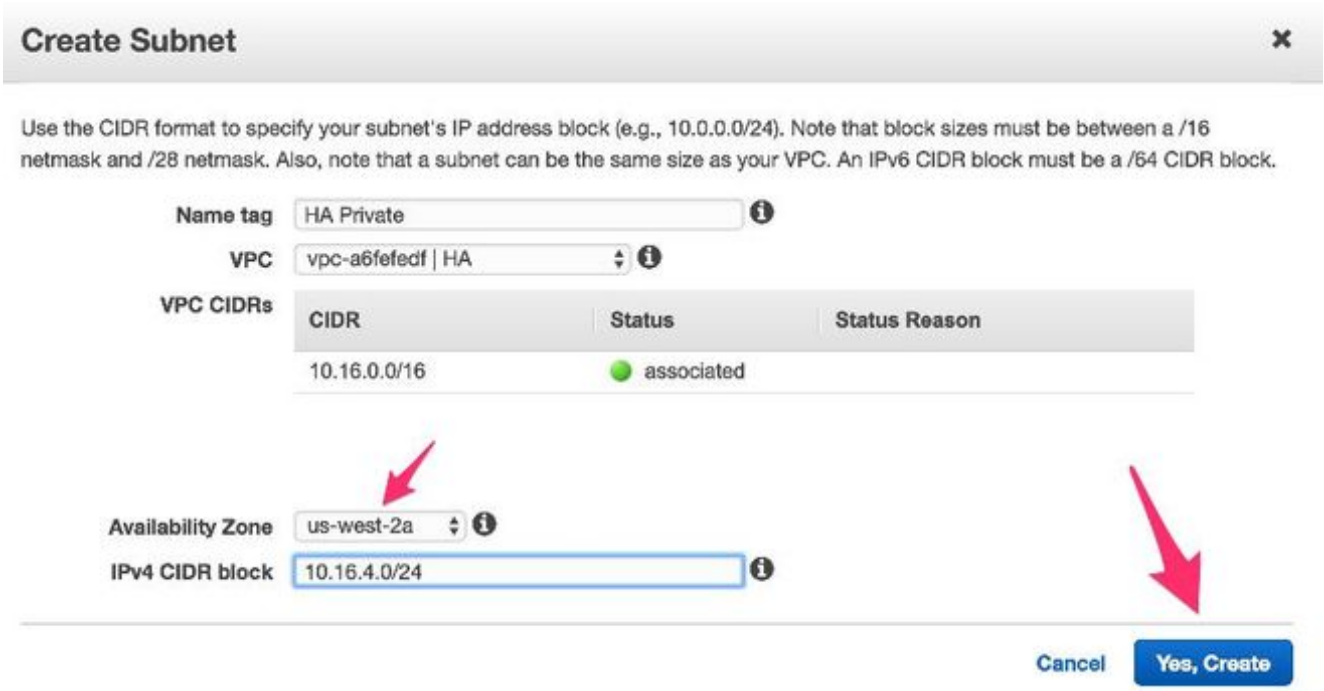
4. Seleccione un tipo de instancia. Para este ejemplo, el tipo seleccionado es **t2.medium**.



5. Mientras la instancia está configurada, debe asegurarse de seleccionar el VPC que creó anteriormente junto con el rol IAM anterior. Además, cree una subred privada que asocie a la interfaz privada.



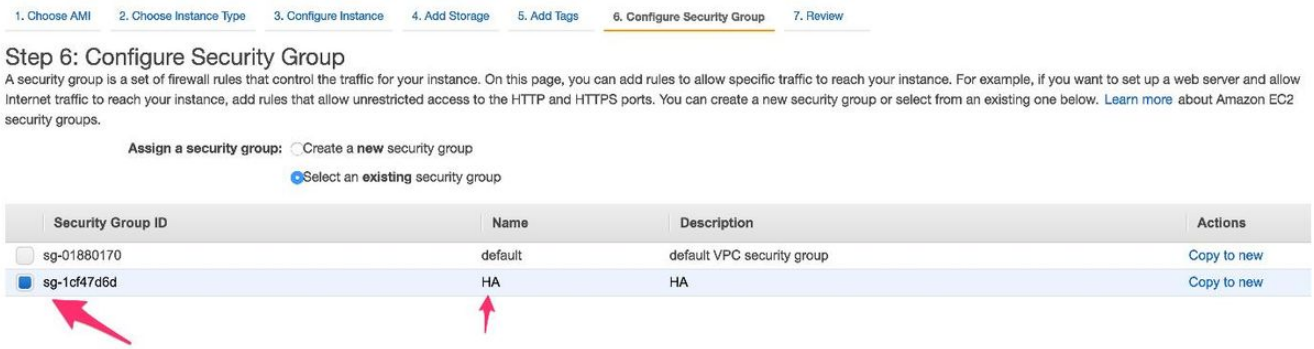
6. Haga clic en Create new Subnet for Private Subnet (Crear nueva subred para subred privada). En este ejemplo, la etiqueta Name es HA Private. Asegúrese de que se encuentra en la misma zona de disponibilidad que la subred pública.



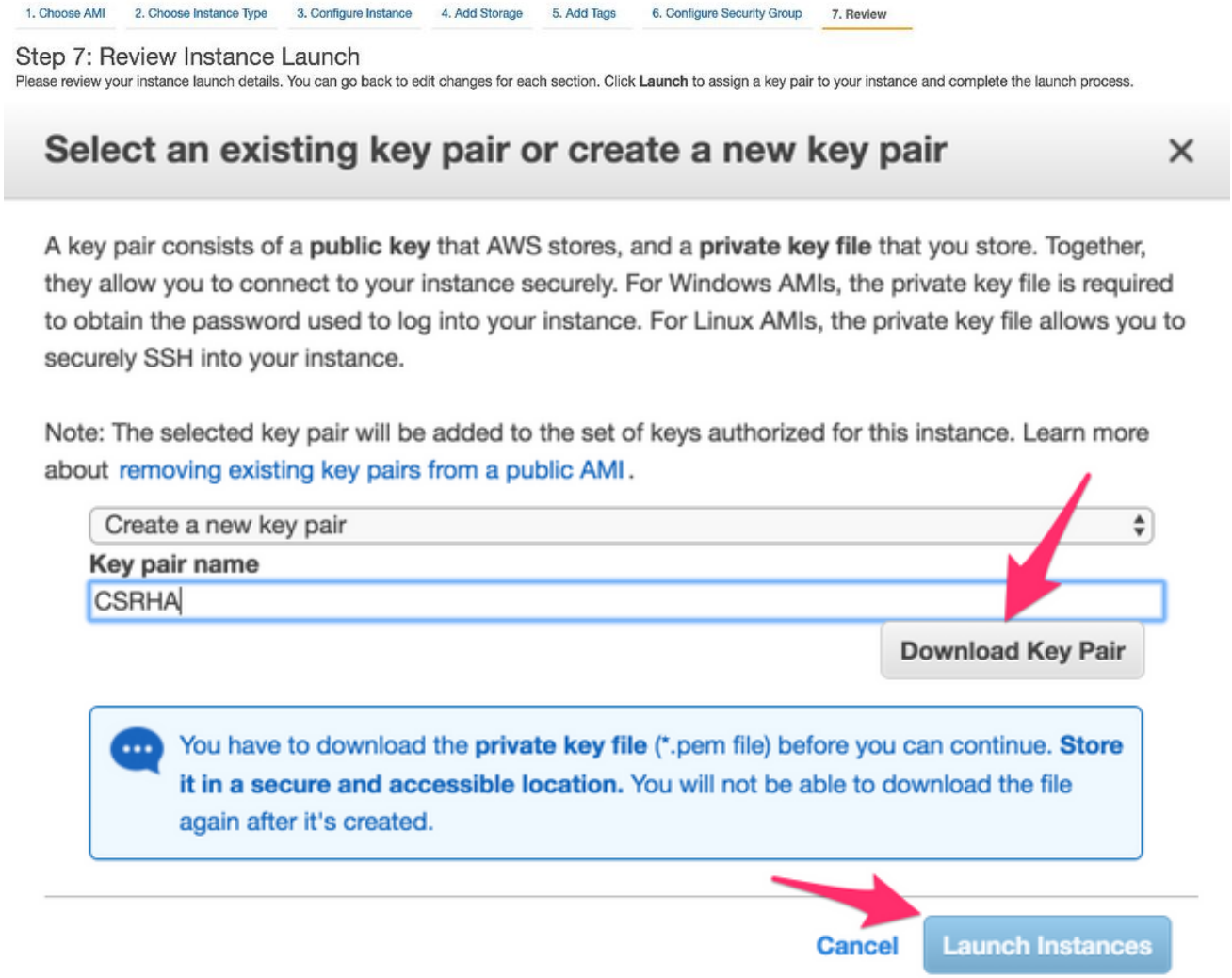
7. Desplácese hacia abajo y en Configurar detalles de instancia, haga clic en **Agregar dispositivo**, como se muestra en la imagen.

8. Una vez agregada la interfaz secundaria, asocie la subred privada que ha creado denominada HA Private. Eth0 es la interfaz pública y Eth1 es la interfaz privada. **Nota:** Es posible que la subred creada en el paso anterior no aparezca en esta lista desplegable. Es posible que tenga que actualizar o cancelar la página y volver a empezar para que aparezca la
subred.

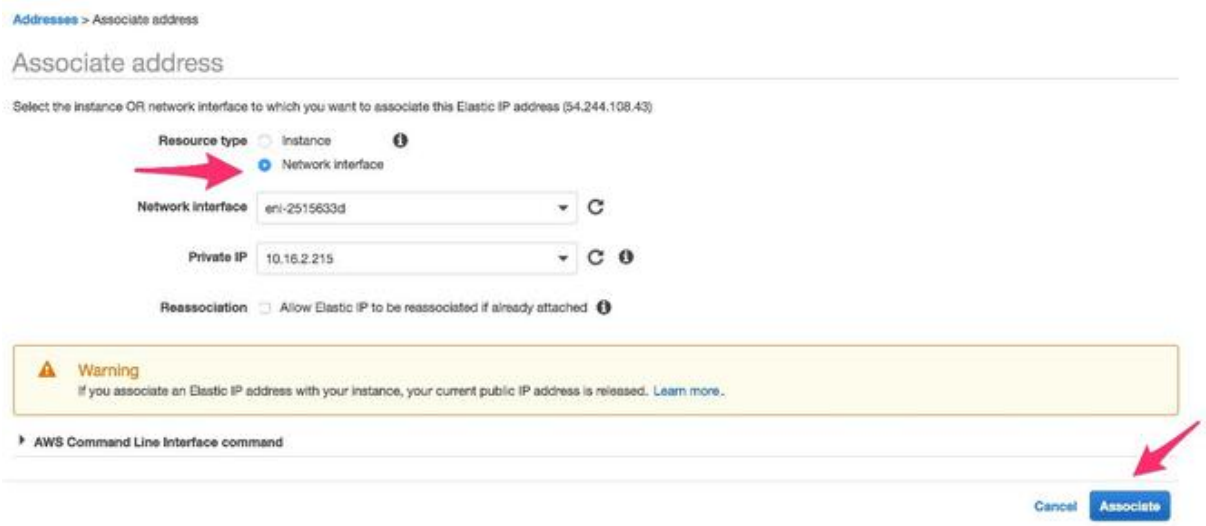
9. Seleccione el grupo de seguridad que ha creado en VPC y asegúrese de que las reglas están definidas correctamente.



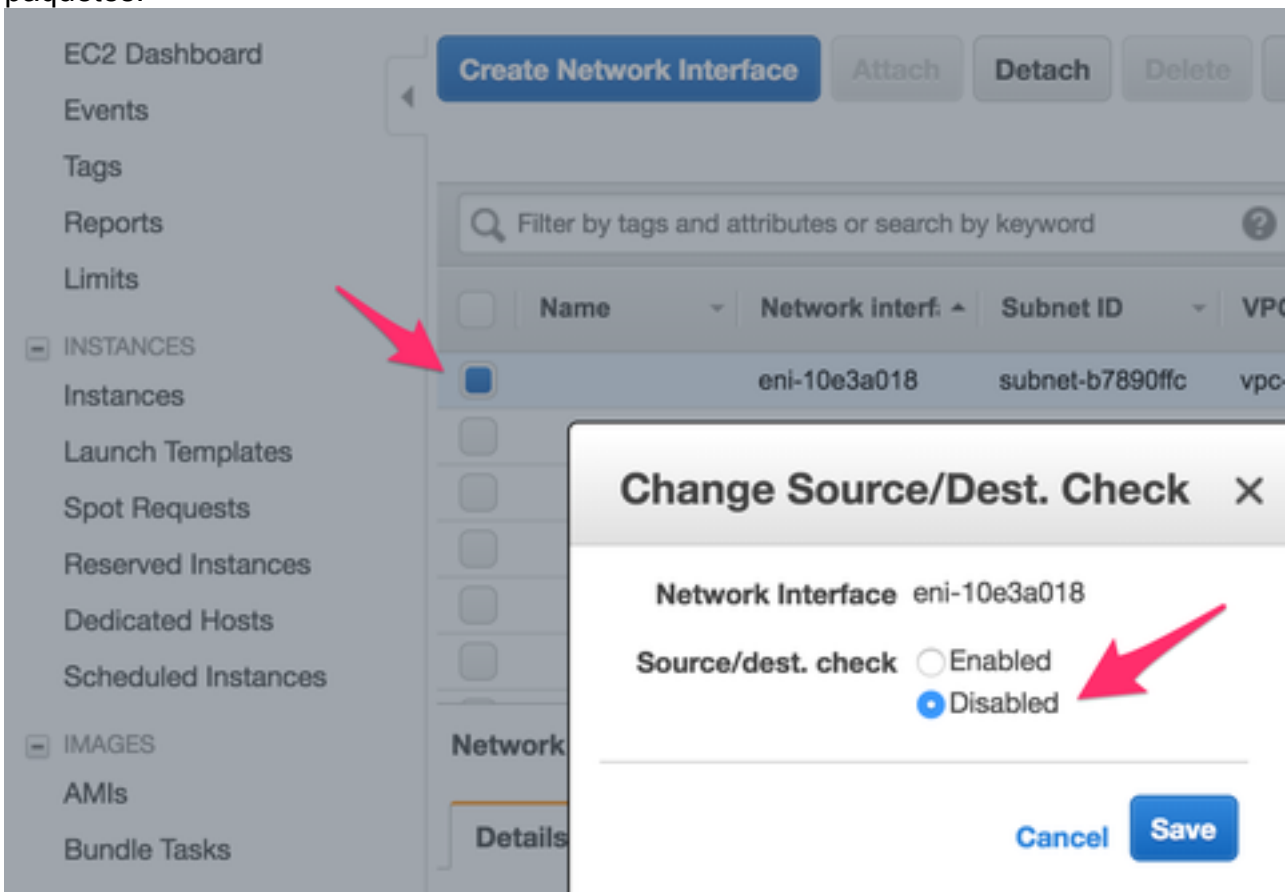
10. Cree un nuevo par de claves y asegúrese de descargar la clave privada. Puede reutilizar una clave para cada dispositivo. **Nota:** Si pierde su clave privada, no podrá volver a iniciar sesión en su CSR. No hay ningún método para recuperar claves.



11. Asocie la IP elástica con la ENI de la interfaz pública para la instancia que creó y navegue hasta la **consola de AWS > EC2 Management > Network Security > Elastic IP's**. **Nota:** La terminología pública/privada puede confundirle aquí. A efectos de este ejemplo, la definición de una interfaz pública es Eth0, que es la interfaz de cara a Internet. Desde el punto de vista de AWS, nuestra interfaz pública es su IP privada.

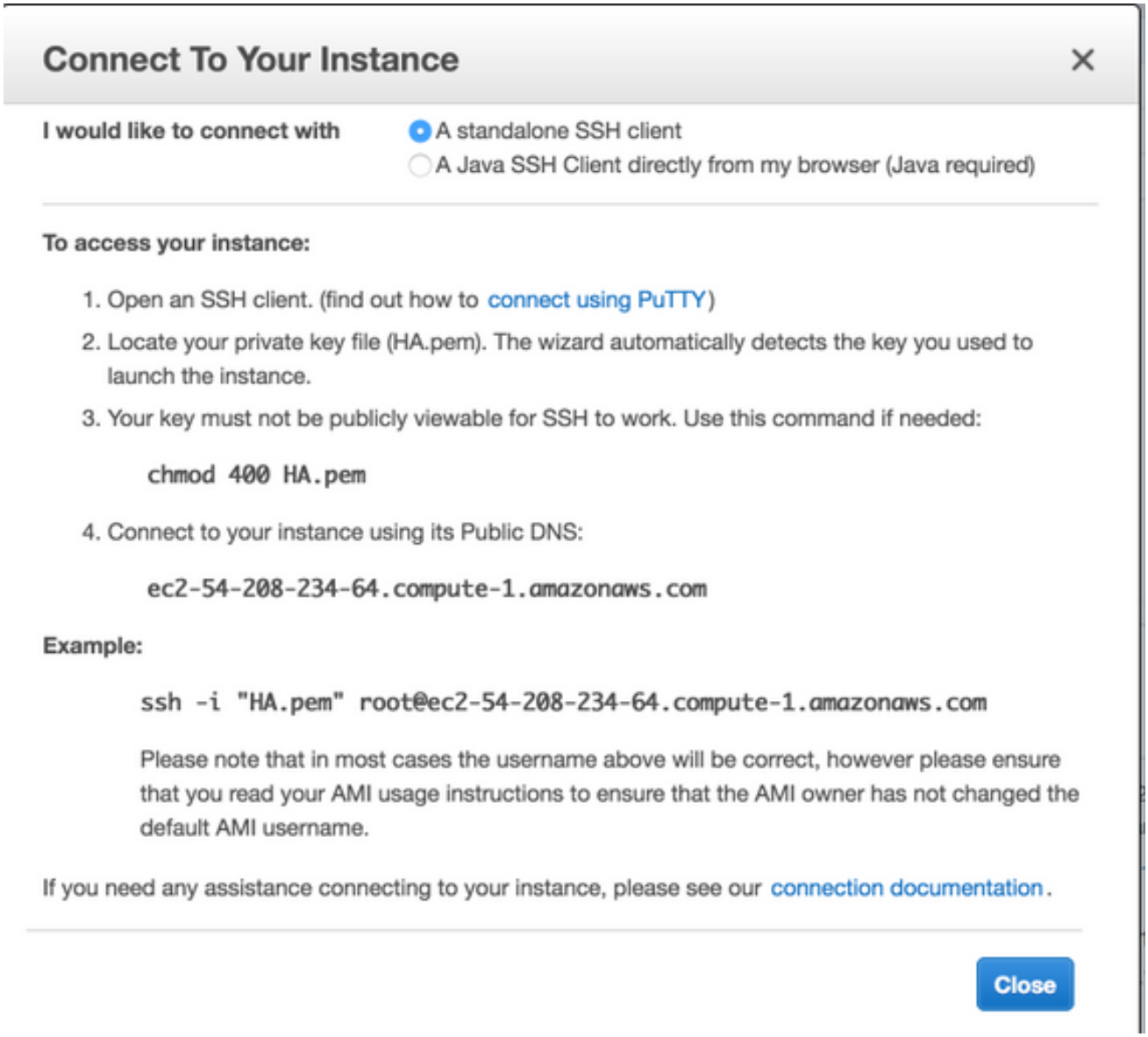


12. Inhabilite Source/Dest Check mientras navega a **EC2 > Network Interfaces**. Verifique cada ENI para la verificación de origen/destino. De forma predeterminada, todos los ENI vienen con esta verificación de origen/destino habilitada. Una función antisimulación diseñada para evitar dejar que un ENI se vea desbordado por tráfico que no está realmente destinado a él, verificando que el ENI es el destino del tráfico antes de reenviarlo. El router rara vez es el destino real de un paquete. Esta función debe estar inhabilitada en todos los ENI de tránsito CSR o no puede reenviar
paquetes.



13. Conéctese a su CSR1000v. **Nota:** Es posible que el nombre de usuario proporcionado por AWS a SSH en el CSR1000v se incluya incorrectamente como raíz. Cambie esto a ec2-user si es necesario.**Nota:** Debe poder hacer ping a la dirección DNS para SSH en. Aquí está ec2-54-208-234-64.compute-1.amazonaws.com. Verifique que la subred/eni pública del router esté asociada con la Tabla de rutas públicas. Vaya brevemente al paso 8 para saber cómo asociar la subred a la tabla de

rutas.



Paso 6. Repita el paso 5 y cree la segunda instancia CSR1000v para HA.

Subred pública: 10.16.1.0/24

Subred privada: 10.16.5.0/24

Si no puede hacer ping a la dirección IP elástica de este nuevo AMI, vaya brevemente al paso 8 y asegúrese de que la subred pública esté asociada a la tabla de rutas públicas.

Paso 7. Repita el Paso 5 y Cree una VM (Linux/Windows) desde AMI Marketplace.

Para este ejemplo, utilice Ubuntu Server 14.04 LTS en el mercado.

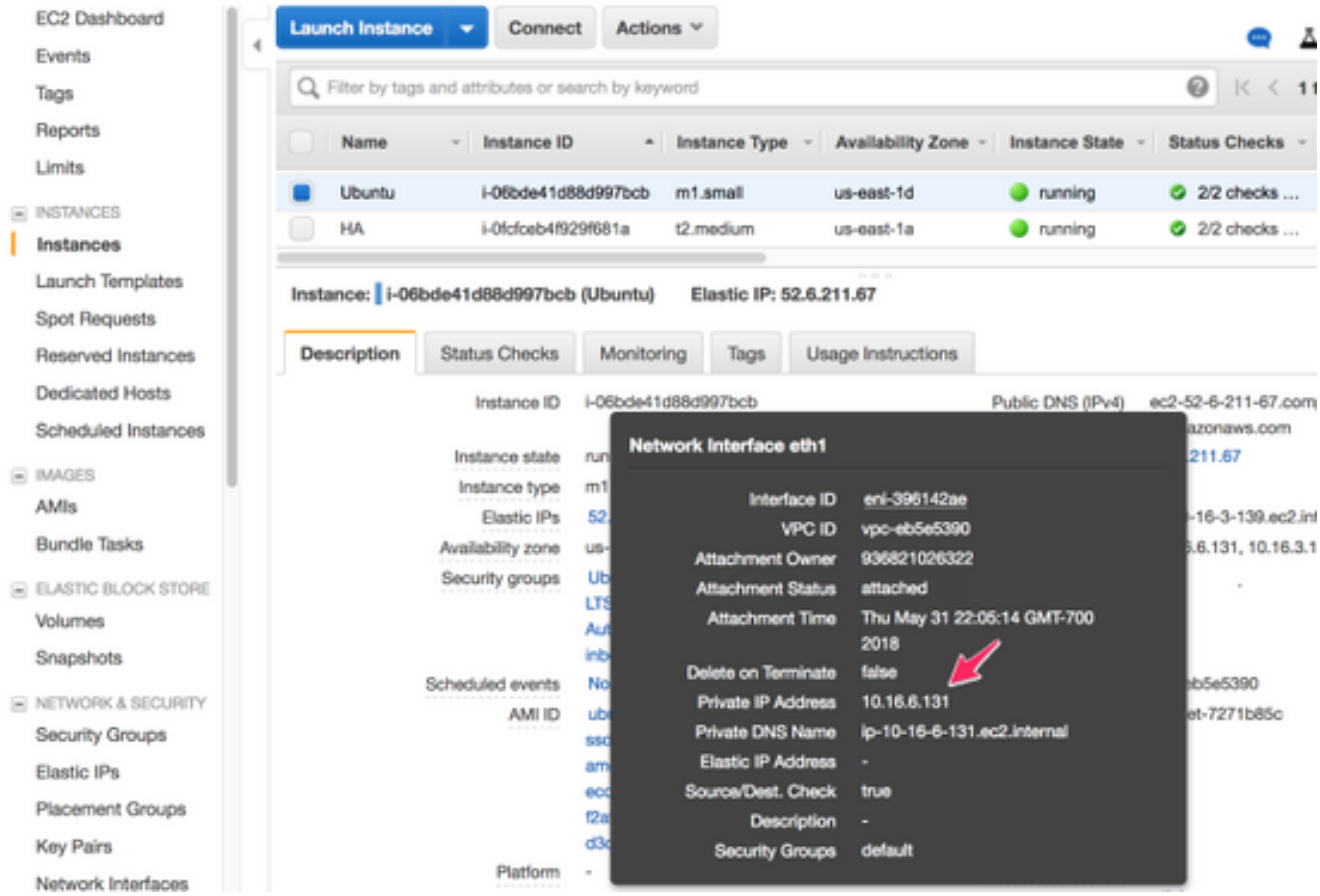
Subred pública: 10.16.2.0/24

Subred privada: 10.16.6.0/24

Si no puede hacer ping a la dirección IP elástica de este nuevo AMI, vaya brevemente al paso

8 y asegúrese de que la subred pública esté asociada a la tabla de rutas públicas.

1. Eth0 se crea de forma predeterminada para la interfaz pública. Cree una segunda interfaz llamada eth1 para la subred privada.



2. La dirección IP que configure en Ubuntu es la interfaz privada eth1 asignada por AWS.

```
ubuntu@ip-10-16-2-139:~$ cd /etc/network/interfaces.d/

ubuntu@ip-10-16-2-139:/etc/network/interfaces.d$ sudo vi eth1.cfg

auto eth1
iface eth1 inet static
  address 10.16.6.131
  netmask 255.255.255.0
  network 10.16.6.0
  up route add -host 8.8.8.8 gw 10.16.6.1 dev eth1
```

3. Desactive la interfaz o reinicie la máquina virtual.

```
ubuntu@ip-10-16-2-139:/etc/network/interfaces.d$ sudo ifdown eth1 && sudo ifup eth1
ubuntu@ip-10-16-2-139:/etc/network/interfaces.d$ sudo reboot
```

4. Ping 8.8.8.8 para el ensayo. Asegúrese de que la ruta 8.8.8.8 haya sido agregada por el paso 7.

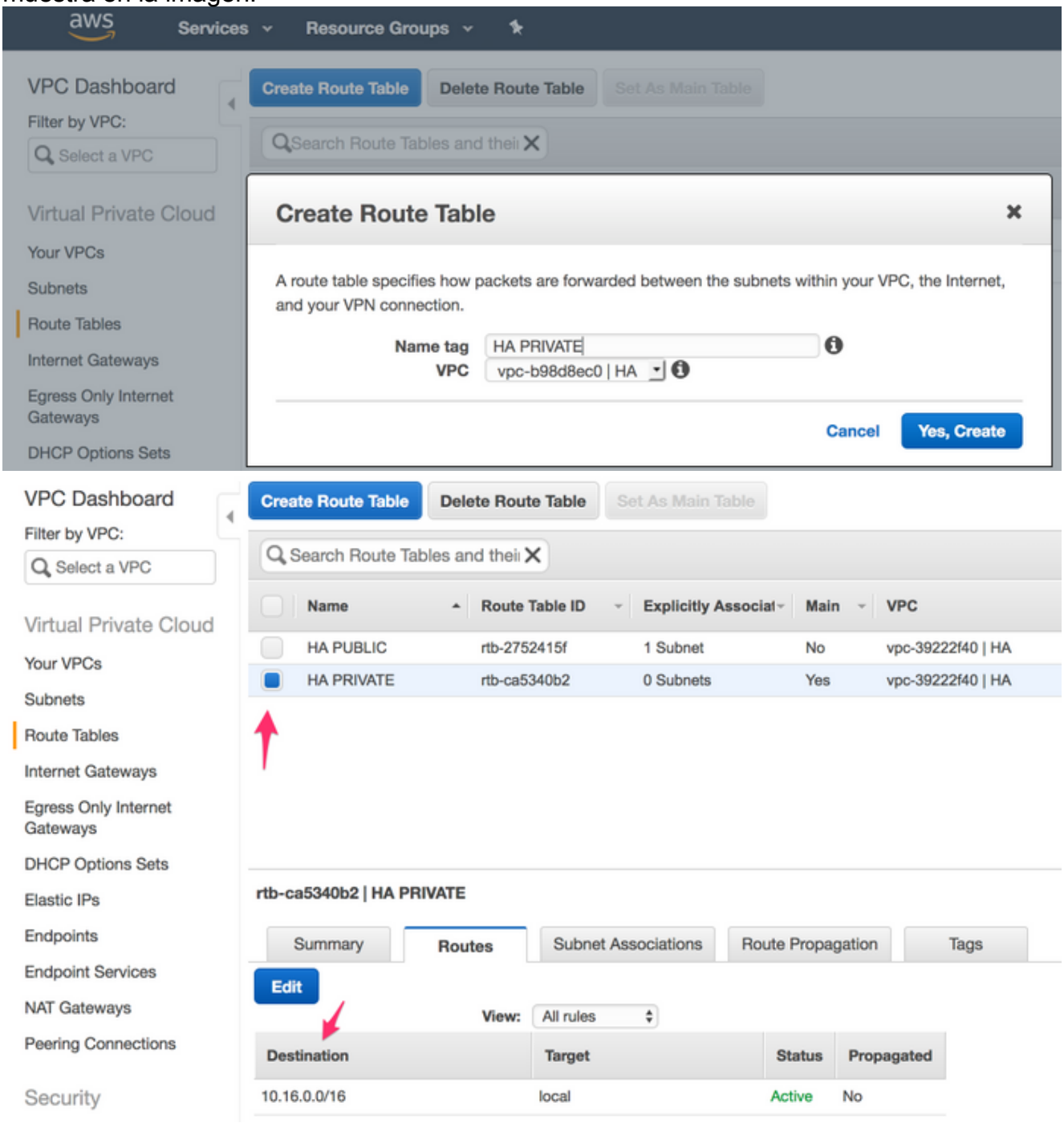
```
ubuntu@ip-10-16-2-139:~$ route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 10.16.2.1 0.0.0.0 UG 0 0 0 eth0
8.8.8.8 10.16.6.1 255.255.255.255 UGH 0 0 0 eth1     <--------------
10.16.3.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
10.16.6.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1
```

Si 8.8.8.8 no aparece en la tabla, agréguela manualmente:

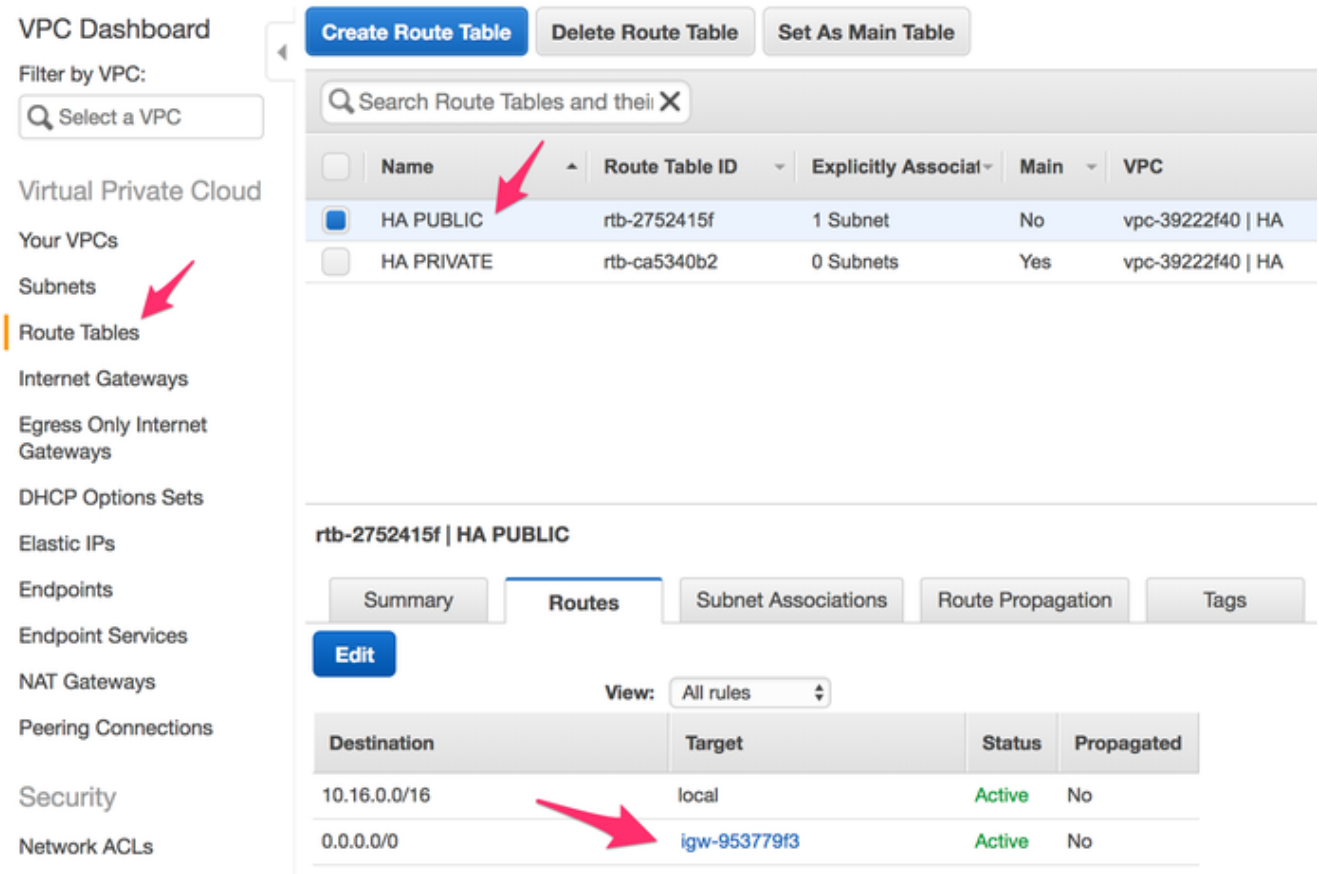
```
ubuntu@ip-10-16-2-139:~$ sudo route add -host 8.8.8.8 gw 10.16.6.1 dev eth1
```

Paso 8. Configure las Tablas de Ruta Privada y Pública.

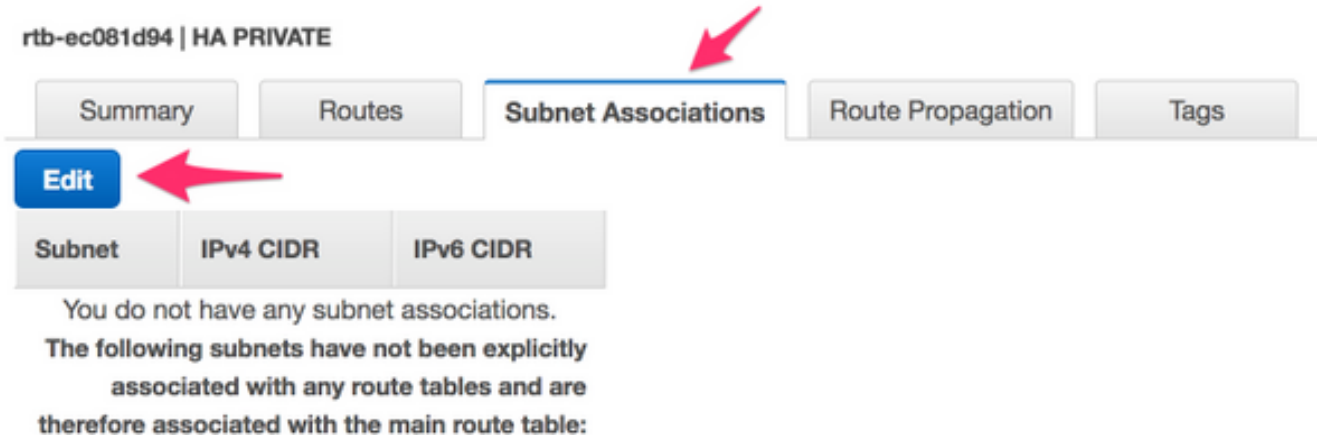
1. Cuando se crea un VPC a través del asistente en el paso 2, se crean automáticamente dos tablas de ruta. Si sólo hay una tabla de rutas, cree otra para sus subredes privadas, como se muestra en la imagen.



2. Esta es una vista de las dos tablas de rutas. La tabla de rutas PÚBLICAS tiene la puerta de enlace a Internet (igw-95377973) conectada automáticamente. Etiquete estas dos tablas como corresponda. La tabla PRIVATE NO debe tener esta ruta.



3. Asociar las 6 subredes a la tabla de rutas adecuada 3 Las interfaces públicas están asociadas con la tabla de rutas públicas:Subredes públicas: 10.16.0.0/24, 10.16.1.0/24 y 10.16.2.0/24 3 Las interfaces privadas están asociadas con la tabla de rutas privadas:Subredes privadas: 10.16.4.0/24, 10.16.5.0/24 y 10.16.6.0/24



Paso 9. Configure la traducción de direcciones de red (NAT) y el túnel GRE con BFD y cualquier protocolo de enrutamiento.

Configure el túnel de encapsulación de routing genérico (GRE) a través de las IP elásticas de los CSR 1000v (se recomienda evitar los problemas de renovación de concesión de DHCP, que detectan fallos falsos). Los valores de detección de reenvío de bidireccionamiento (BFD) se pueden configurar para que sean más agresivos que los que se muestran en este ejemplo, si se requiere una convergencia más rápida. Sin embargo, esto puede llevar a eventos de peer down BFD durante la conectividad intermitente. Los valores de este ejemplo detectan fallas de peer en 1,5 segundos. Hay un retraso variable de unos pocos segundos entre el momento en que se ejecuta el comando API de AWS y cuando los cambios en la tabla de ruteo VPC entran en efecto.

- Configuración en CSRHA

  GRE y BFD: se utilizan para observar las condiciones de conmutación por fallo de HA

```
interface Tunnel1
  ip address 192.168.1.1 255.255.255.0
  bfd interval 500 min_rx 500 multiplier 3
  tunnel source GigabitEthernet1
  tunnel destination 52.10.183.185 /* Elastic IP of the peer CSR */
!
router eigrp 1
  bfd interface Tunnel1
  network 192.168.1.0
  passive-interface GigabitEthernet1
```

  NAT y routing: se utiliza para la accesibilidad a Internet de VM a través de la interfaz privada

```
interface GigabitEthernet1
  ip address dhcp
  ip nat outside
  no shutdown
!
interface GigabitEthernet2
  ip address dhcp
  ip nat inside
  no shutdown
!
ip nat inside source list 10 interface GigabitEthernet1 overload
!
access-list 10 permit 10.16.6.0 0.0.0.255
!
ip route 10.16.6.0 255.255.255.0 GigabitEthernet2 10.16.4.1
```

- Configuración en CSRHA1

  GRE y BFD: se utilizan para observar las condiciones de conmutación por fallo de HA

```
interface Tunnel1
  ip address 192.168.1.2 255.255.255.0
  bfd interval 500 min_rx 500 multiplier 3
  tunnel source GigabitEthernet1
  tunnel destination 50.112.227.77 /* Elastic IP of the peer CSR */
!
router eigrp 1
  bfd interface Tunnel1
  network 192.168.1.0
  passive-interface GigabitEthernet1
```

  NAT y routing: se utiliza para la accesibilidad a Internet de VM a través de la interfaz privada

```
interface GigabitEthernet1
  ip address dhcp
  ip nat outside
  no shutdown
!
interface GigabitEthernet2
  ip address dhcp
  ip nat inside
```

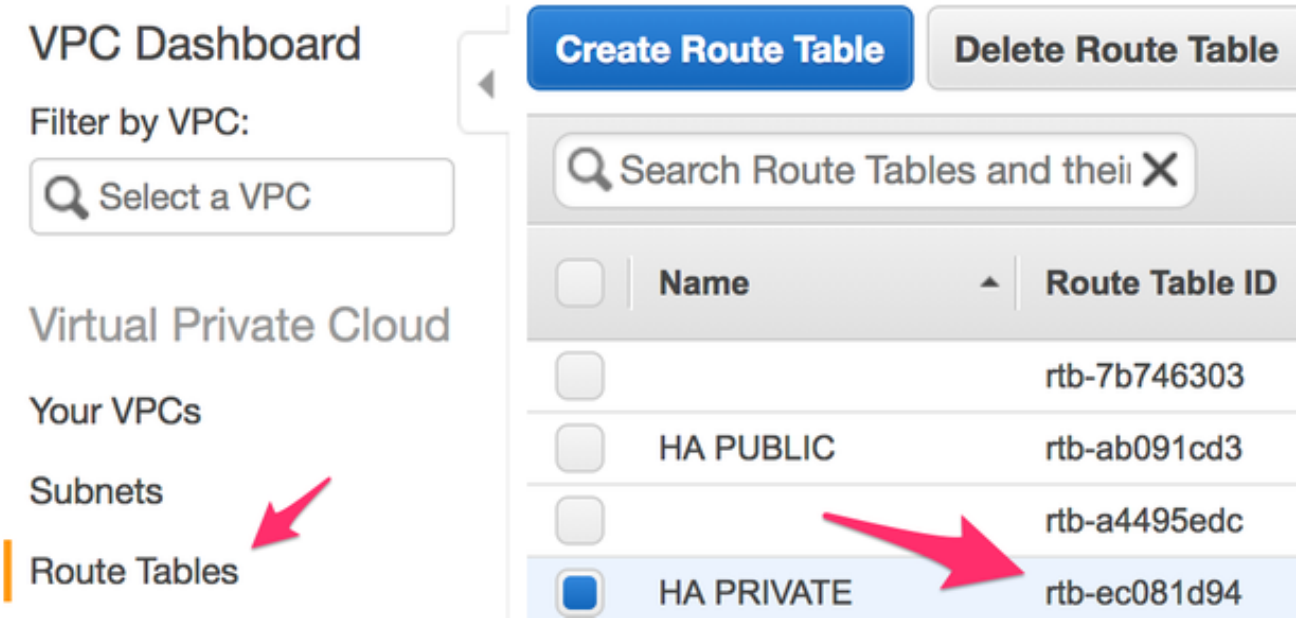
```
  no shutdown
!
ip nat inside source list 10 interface GigabitEthernet1 overload
!
access-list 10 permit 10.16.6.0 0.0.0.255
!
ip route 10.16.6.0 255.255.255.0 GigabitEthernet2 10.16.5.1
```

Paso 10. Configure High Availability (Denali 16.3.1a o posterior de Cisco IOS XE).

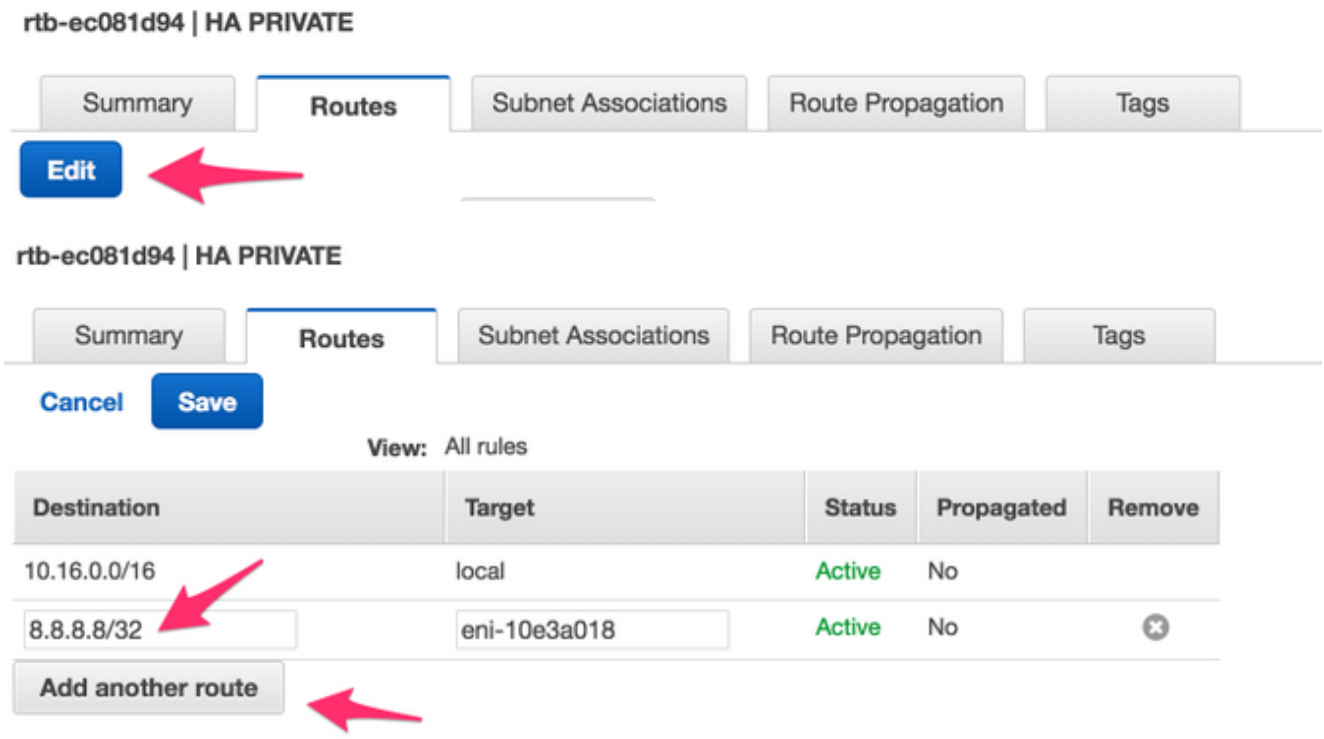
Supervise los eventos de peer down BFD configurando cada CSR 1000v mediante el comando del proveedor de nube aws especificado a continuación. Utilice este comando para definir los cambios de ruteo a (VPC) Route-table-id, Network-interface-id y CIDR después de que se detecte un error AWS HA como un peer down BFD.

```
CSR(config)# redundancy
CSR(config-red)# cloud provider [aws | azure] node-id
# bfd peer ipaddr
# route-table table-name
# cidr ip ipaddr/prefix
# eni elastic-network-intf-name
# region region-name
```

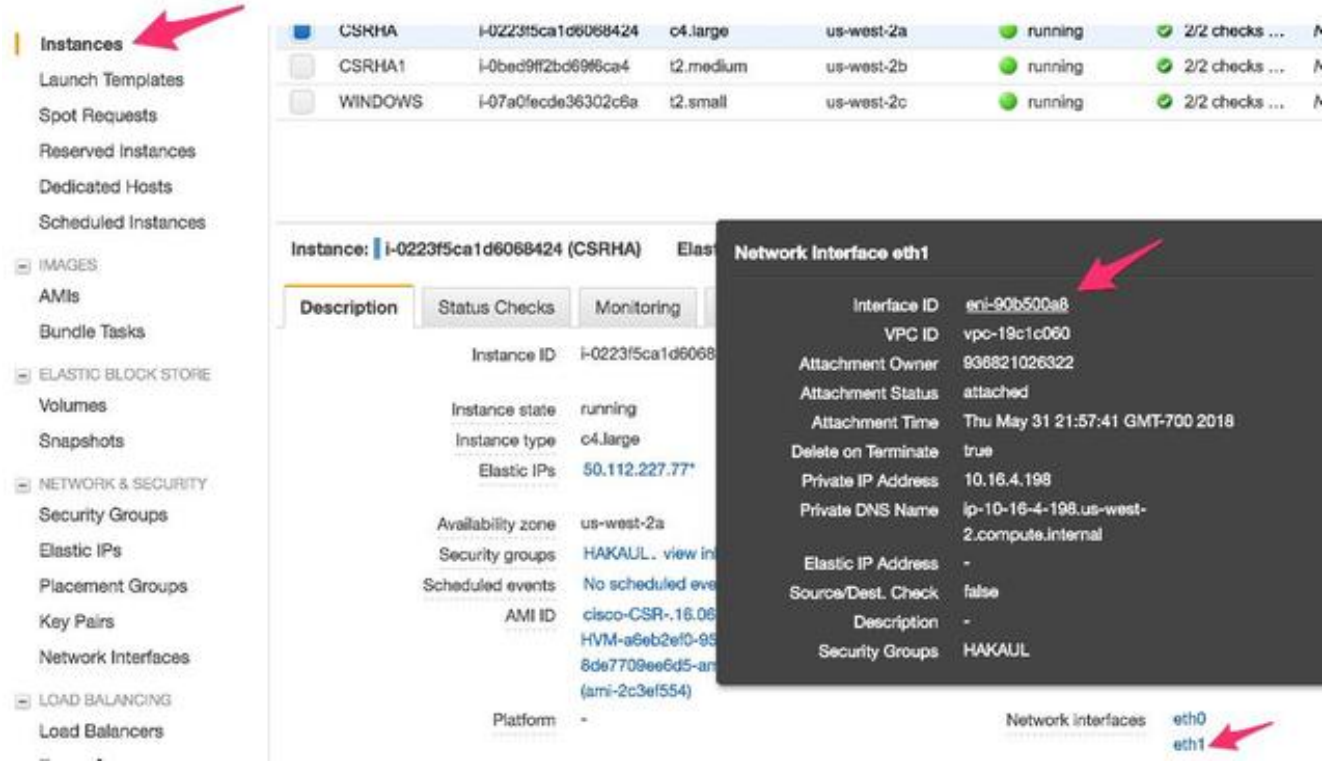
1. La dirección IP del par #bfd es la dirección IP del túnel del par.
   ```
   CSRHA#show bfd neighbors

   IPv4 Sessions
   NeighAddr LD/RD RH/RS State Int
   192.168.1.2 4097/4097 Up Up Tu1
   ```
2. El nombre de la tabla #route-table se encuentra en la consola AWS, navegue hasta **VPC > Tablas de ruta**. Esta acción altera la tabla de rutas privadas.



3. El prefijo/dirección IP de #cidr es la dirección de destino para la ruta que se actualizará en la tabla de rutas. En la consola de AWS, navegue hasta **VPC > Tablas de rutas**. Desplácese hacia abajo, haga clic en **Edit** y luego en **Add another route**. Añada nuestra dirección de destino de prueba 8.8.8.8 y el ENI privado de CSRHA.



4. #eni elastic-network-intf-name se encuentra en su instancia EC2. Haga clic en su interfaz de cara privada eth1 para cada CSR correspondiente y utilice el ID de interfaz.



5. El nombre #region es el nombre de código que se encuentra en el documento AWS. Esta lista puede cambiar o aumentar. Para encontrar las últimas actualizaciones, visite el documento Región y Zonas de Disponibilidad de Amazon.

Code	Name
us-east-1	US East (N. Virginia)
us-east-2	US East (Ohio)
us-west-1	US West (N. California)
us-west-2	US West (Oregon)
ca-central-1	Canada (Central)
eu-central-1	EU (Frankfurt)
eu-west-1	EU (Ireland)
eu-west-2	EU (London)
eu-west-3	EU (Paris)
ap-northeast-1	Asia Pacific (Tokyo)
ap-northeast-2	Asia Pacific (Seoul)
ap-northeast-3	Asia Pacific (Osaka-Local)
ap-southeast-1	Asia Pacific (Singapore)
ap-southeast-2	Asia Pacific (Sydney)
ap-south-1	Asia Pacific (Mumbai)
sa-east-1	South America (São Paulo)

Ejemplo de Configuración de Redundancia en CSRHA

```
redundancy
cloud provider aws 1
  bfd peer 192.168.1.2
  route-table rtb-ec081d94
  cidr ip 8.8.8.8/32
  eni eni-90b500a8
  region us-west-2
```

Ejemplo de Configuración de Redundancia en CSRHA1

```
redundancy
cloud provider aws 1
  bfd peer 192.168.1.1
  route-table rtb-ec081d94
  cidr ip 8.8.8.8/32
  eni eni-10e3a018
  region us-west-2
```

Verificar alta disponibilidad

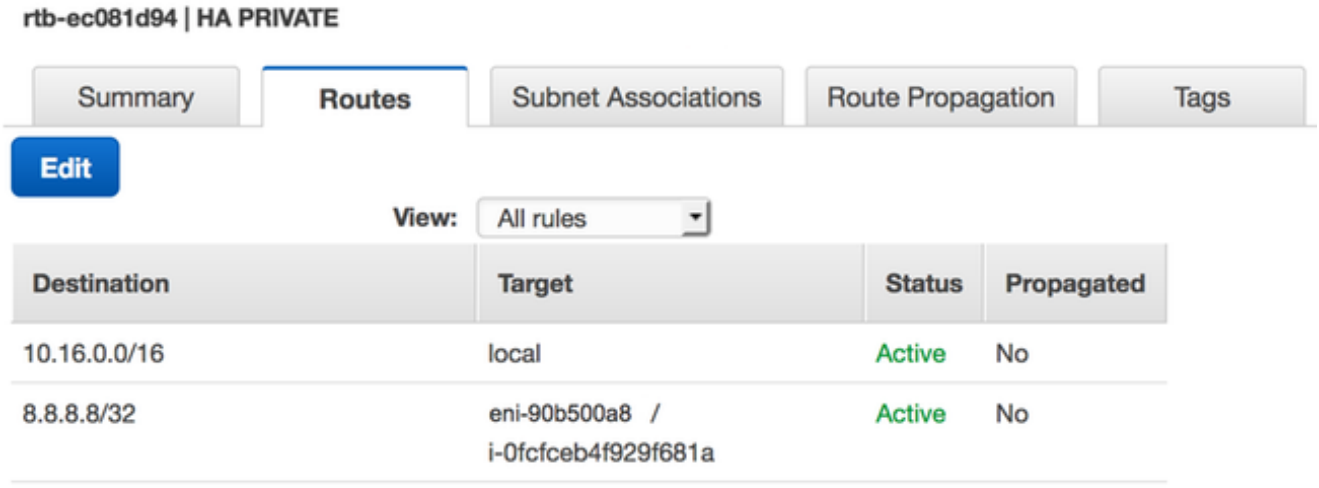
1. Compruebe las configuraciones de nube y BFD.

```
CSRHA#show bfd nei

IPv4 Sessions
NeighAddr LD/RD RH/RS State Int
192.168.1.2 4097/4097 Up Up Tu1


CSRHA#show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(1)
H Address Interface Hold Uptime SRTT RTO Q Seq
(sec) (ms) Cnt Num
0 192.168.1.2 Tu1 12 00:11:57 1 1470 0 2


CSRHA#show redundancy cloud provider aws 1

Cloud HA: work_in_progress=FALSE
Provider : AWS node 1
State : idle
BFD peer     = 192.168.1.2
BFD intf     = Tunnel1
route-table  = rtb-ec081d94
cidr         = 8.8.8.8/32
eni          = eni-90b500a8
region       = us-west-2
```

2. Ejecute un ping continuo desde la VM al destino. Asegúrese de que el ping se realiza a través de la interfaz eth1 privada.

```
ubuntu@ip-10-16-3-139:~$ ping -I eth1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) from 10.16.6.131 eth1: 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=50 time=1.60 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=50 time=1.62 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=50 time=1.57 ms
```

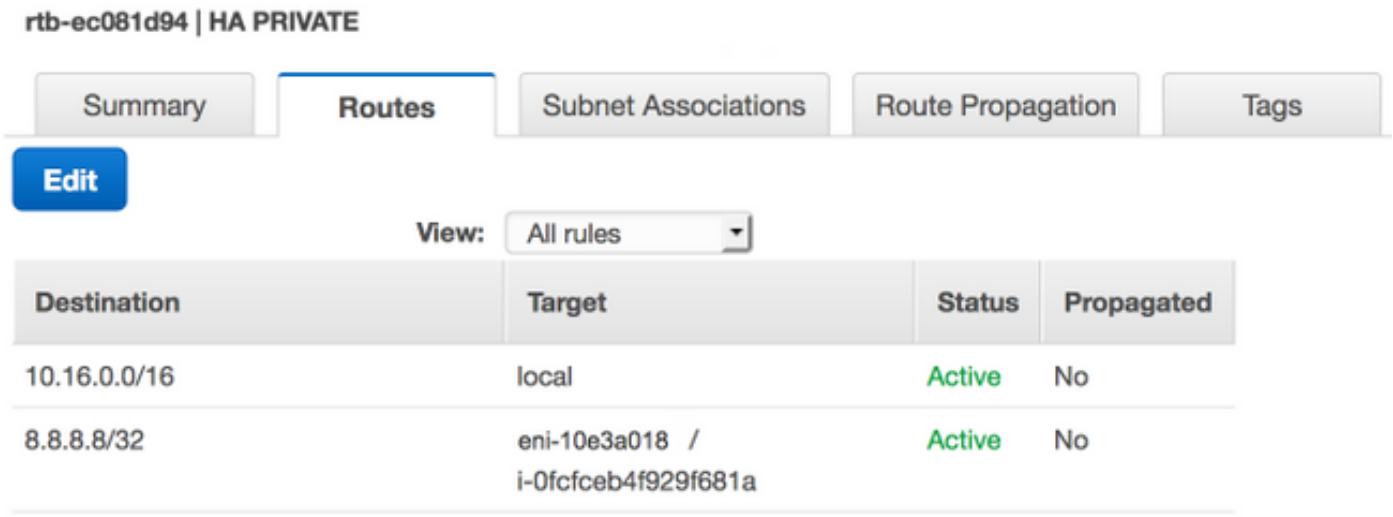
3. Verifique la tabla de rutas privadas. El eni es actualmente la interfaz privada de CSRHA donde este es el tráfico.



4. Cierre el túnel 1 de CSRHA para simular una falla de HA.

```
CSRHA(config)#int Tun1
CSRHA(config-if)#shut
```

5. Observe que la tabla de rutas apunta al nuevo ENI que es la interfaz privada de CSRHA1.



Troubleshoot

- Asegúrese de que los recursos están asociados. Al crear VPC, subredes, interfaces, tablas de rutas, etc., muchas de estas no se asocian entre sí automáticamente. No tienen conocimiento el uno del otro.
- Asegúrese de que la IP elástica y cualquier IP privada estén asociadas con las interfaces correctas, con las subredes adecuadas, agregadas a la tabla de rutas correcta, conectadas al router correcto y a la VPC y zona correctas, vinculadas con el rol IAM y los grupos de seguridad.
- Desactive la comprobación de origen/destino por ENI.
- Para Cisco IOS XE 16.3.1a o posterior, estos son los comandos de verificación adicionales disponibles.

```
show redundancy cloud provider [aws | azure] node-id
debug redundancy cloud [all | trace | detail | error]
debug ip http all
```

- A continuación se indican los errores más comunes observados en las depuraciones:

Problema: error de httpc_send_request

Resolución: Http se utiliza para enviar la llamada API desde el CSR a AWS. Asegúrese de que DNS puede resolver el nombre DNS que aparece en la instancia. Asegúrese de que el tráfico http no esté bloqueado.

```
*May 30 20:08:06.922: %VXE_CLOUD_HA-3-FAILED: VXE Cloud HA BFD state transitioned, AWS node 1
event httpc_send_request failed
*May 30 20:08:06.922: CLOUD-HA : AWS node 1 httpc_send_request failed (0x12)
URL=http://ec2.us-east-2b.amazonaws.com
```

Problema: la tabla de rutas rtb-9c0000f4 y la interfaz eni-32791318 pertenecen a redes diferentes

Resolución: El nombre de región y ENI no están configurados correctamente en redes diferentes. La región y ENI deben estar en la misma zona que el router.

```
*May 30 23:38:09.141: CLOUD-HA : res content iov_len=284 iov_base=<?xml version="1.0"
encoding="UTF-8"?>
<Response><Errors><Error><Code>InvalidParameterValue</Code><Message>route table rtb-9c0000f4 and
interface eni-32791318 belong to different
networks</Message></Error></Errors><RequestID>af3f228c-d5d8-4b23-b22c-
f6ad999e70bd</RequestID></Response>
```

Problema: No está autorizado para realizar esta operación. Mensaje de error de autorización codificado.

Resolución: Función/política IAM JSON creada incorrectamente o no aplicada a CSR. El rol IAM autoriza al CSR a realizar llamadas API.

```
*May 30 22:22:46.437: CLOUD-HA : res content iov_len=895 iov_base=<?xml version="1.0"
encoding="UTF-8"?>
<Response><Errors><Error><Code>UnauthorizedOperation</Code><Message>You are not authorized to
perform this operation. Encoded
authorization failure message: qYvEB4MUdOB8m2itSteRgnOuslAaxhAbDph5qGRJkjJbrESajbmF5HWUR-
MmHYeRAlpKZ3Jg_y-
_tMlYel5l_ws8Jd9q2W8YDXBl3uXQqfW_cjjrgy9jhnGY0nOaNu65aLpfqui8kS_4RPOpm5grRFfo99-
8uv_N3mYaBqKFPn3vUcSYKBmxFIIkJKcjY9esOeLIOWDcnYGGu6AGGMoMxWDtk0K8nwk4IjLDcnd2cDXeENS45w1PqzKGPsH
v3wD28TS5xRjIrPXYrT18UpV6lLA_09Oh4737VncQKfzbz4tPpnAkoW0mJLQ1vDpPmNvHUpEng8KrGWYNfbfemoDtWqIdABf
aLLLmh4saNtnQ_OMBoTi4toBLEb2BNdMkl1UVBIxqTqdFUVRS**MSG 00041 TRUNCATED** **MSG 00041
CONTINUATION
#01**qLosAb5Yx0DrOsLSQwzS95VGvQM_n87LBHYbAWWhqWj3UfP_zmiak7dlm9P41mFCucEB3Cs4FRsFtb-
9q44VtyQJaS2sU2nhGe3x4uGEsl7F1pNv5vhVeYOZB3tbOfbV1_Y4trZwYPFgLKgBShZp-WNmUKUJsKc1-
6KGqmp7519imvh66JgwgmU9DT_qAZ-jEjkqWjBrxg6krw</Message></Error></Errors><RequestID>4cf31249-
2a6e-4414-ae8d-6fb825b0f398</RequestID></Response>
```

Información Relacionada

- Redundancia de gateway VPC: Cisco
- Guía de implementación del router de servicios en la nube Cisco CSR serie 1000v para servicios web de Amazon
- Desglose de tipos de instancia
- EC2 y VPC
- Elastic Network Interfaces, de EC2 User Guide, incluye el nº de ENI por tipo de instancia
- Cómo mejorar las redes en Linux, información de fondo útil
- Instancias/explicación de arrendatario y procedimientos específicos
- Documentación general de EC2
- Documentación general de VPC
- Regiones y zonas de disponibilidad
- CSR1000v High Availability versión 3

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).