Configurazione di CSR1000v HA versione 3 su AWS, Azure e GCP

Sommario

Introduzione

In questo documento viene descritto come configurare i router CSR1000v per l'alta disponibilità versione 3 (HAv3) su Amazon Web Services (AWS), Microsoft Azure e Google Cloud Platform (GCP).

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Cloud AWS, Azure o GCP.
- Router CSR1000v.
- Cisco IOS®-XE

In questo articolo si presume che la configurazione di rete sottostante sia già stata completata e si concentra sulla configurazione HAv3.

Per informazioni dettagliate sulla configurazione, consultare la guida alla configurazione di Cisco CSR 1000v e del software Cisco ISRv.

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Un account AWS, Azure o GCP.
- 2 router CSR1000v.
- Almeno Cisco IOS®-XE Polaris 16.11.1s

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

Cisco raccomanda la conoscenza delle diverse versioni HA disponibili:

- HAv1: La configurazione HA viene eseguita come comandi IOS e si basa sul BFD come meccanismo per rilevare i guasti.
- HAv2/HAv3: L'implementazione è stata spostata nel contenitore di shell come script Python. Il BFD è opzionale e è possibile scrivere script personalizzati per rilevare errori e attivare il failover. La configurazione di Azure HAv2 è simile a HAv3, con differenze minime nei pacchetti di installazione pip e nella configurazione di ridondanza IOS.
- HAv3: L'implementazione di HA è stata in gran parte spostata dal codice Cisco IOS®-XE ed eseguita nel contenitore della shell del guest.

HAv3 è disponibile da Cisco IOS®-XE Polaris 16.11.1s e aggiunge diverse nuove funzioni:

- **Indipendente dal cloud:** Questa versione di alta disponibilità funziona su router CSR 1000v su qualsiasi provider di servizi cloud. Sebbene esistano alcune differenze nella terminologia e nei parametri del cloud, l'insieme di funzioni e script utilizzati per configurare, controllare e mostrare le funzioni di alta disponibilità sono comuni ai diversi provider di servizi cloud. L'alta disponibilità versione 3 (HAv3) è supportata nei router CSR 1000v su AWS, Azure e GCP. Il supporto per il provider GCP è stato aggiunto nella versione 16.11.1. Verificare con Cisco se attualmente è supportata l'alta disponibilità nei cloud del singolo provider.
- **Operazione attiva/attiva:** È possibile configurare entrambi i router Cisco CSR 1000v in modo che siano attivi contemporaneamente, il che consente la condivisione del carico. In questa modalità operativa, ogni router di una tabella di routing ha uno dei due router che fungono da router primario e l'altro da router secondario. Per abilitare la condivisione del carico, prendere tutti i percorsi e dividerli tra i due router Cisco CSR 1000v. Questa funzionalità è stata introdotta per i cloud basati su AWS.
- **Ripristino del CSR principale dopo il ripristino dei guasti:** È possibile designare un Cisco CSR 1000v come router principale per un determinato percorso. Questo Cisco CSR 1000v è attivo. È l'hop successivo del percorso. Se il router Cisco CSR 1000v ha esito negativo, il router peer CSR 1000v diventa l'hop successivo per il percorso, mantenendo la connettività di rete. Quando il router originale si riprende dall'errore, recupera la proprietà del router e diventa il router dell'hop successivo. Questa funzionalità è nuova anche per i cloud basati su AWS.
- **Script forniti dall'utente:** La shell dei guest è un contenitore in cui è possibile distribuire script personalizzati. HAv3 espone un'interfaccia di programmazione agli script forniti dall'utente. Ciò significa che è ora possibile scrivere script che possano attivare eventi di failover e di ripristino. È inoltre possibile sviluppare algoritmi e trigger personalizzati per controllare quale

Cisco CSR 1000v fornisce i servizi di inoltro per un determinato percorso. Questa funzionalità è stata introdotta per i cloud basati su AWS.

- **Nuovo meccanismo di configurazione e installazione:** L'implementazione di HA è stata spostata dal codice Cisco IOS®-XE. Il codice a disponibilità elevata viene ora eseguito nel contenitore della shell dei guest. Per ulteriori informazioni su guestshell, vedere la sezione "Guest Shell" nel Manuale Programmability Configuration Guide. In HAv3, la configurazione dei nodi di ridondanza viene eseguita nella shell guest che utilizza un set di script Python. Questa funzione è stata introdotta per i cloud basati su AWS.

  Nota: Le risorse distribuite in AWS, Azure o GCP dalle fasi di questo documento possono sostenere un costo.

Topologia

Prima di iniziare la configurazione, è importante comprendere completamente la topologia e la progettazione. In questo modo è possibile risolvere eventuali problemi in un secondo momento.

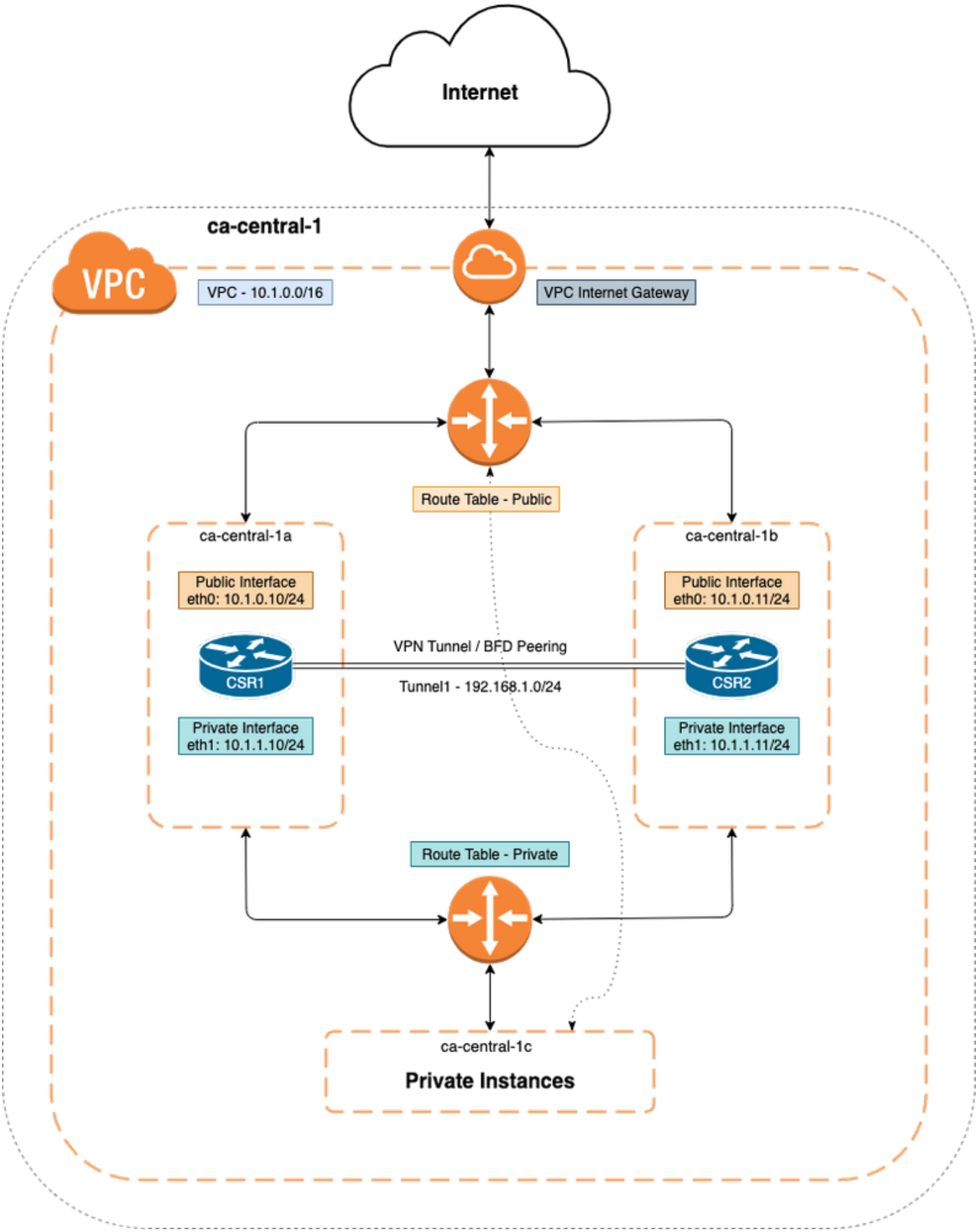
Sebbene il diagramma della topologia di rete sia basato su AWS, la distribuzione di rete sottostante tra i cloud è relativamente simile. La topologia di rete è inoltre indipendente dalla versione HA utilizzata, che si tratti di HAv1, HAv2 o HAv3.

Per questo esempio di topologia, la ridondanza HA è configurata con queste impostazioni in AWS:

- 1x - Regione
- 1x - VPC
- 3x - Zone di disponibilità
- 4x - Interfacce/subnet di rete (2x pubbliche/2x private)
- 2 tabelle di routing (pubbliche e private)
- 2 router - CSR1000v (Cisco IOS®-XE 17.01.01)

In una coppia HA sono presenti due router CSR1000v, in due diverse zone di disponibilità. La terza zona è un'istanza privata, che simula un dispositivo in un centro dati privato. In genere, tutto il traffico normale deve passare attraverso la tabella di route privata (o interna).

Esempio di rete



Esempio di rete

Configurazione dei router CSR1000v

Configurazione indipendente dal cloud

Passaggio 1. Configurare l'hosting dell'app IOX e la shell dei guest, in modo da fornire la raggiungibilità IP nella shell dei guest. Questo passaggio può essere configurato automaticamente per impostazione predefinita quando viene installato CSR1000v.

vrf definition GS ! iox app-hosting appid guestshell app-vnic gateway1 virtualportgroup 0 guest-interface 0 guest-ipaddress 192.168.35.102 netmask 255.255.255.0 app-default-gateway 192.168.35.101 guest-interface 0 name-server0 8.8.8.8 ! interface VirtualPortGroup0 vrf forwarding GS ip address 192.168.35.101 255.255.255.0 ip nat inside ! interface GigabitEthernet1 ip nat outside ! ip access-list standard GS_NAT_ACL permit 192.168.35.0 0.0.0.255 ! ip nat inside source list GS_NAT_ACL interface GigabitEthernet1 vrf GS overload ! ! The static route points to the G1 ip address's gateway ip route vrf GS 0.0.0.0 0.0.0.0 GigabitEthernet1 10.1.0.1 global

Passaggio 2. Abilitare e accedere a Guestshell.

```
Device#guestshell enable
Interface will be selected if configured in app-hosting
Please wait for completion
guestshell installed successfully
Current state is: DEPLOYED
guestshell activated successfully
Current state is: ACTIVATED
guestshell started successfully
Current state is: RUNNING
Guestshell enabled successfully

Device#guestshell
[guestshell@guestshell ~]$
```

> **Nota:** Per ulteriori informazioni su guestshell, vedere - Guida alla configurazione della programmabilità

Passaggio 3. Verificare che Guestshell sia in grado di comunicare con Internet.

```
[guestshell@guestshell ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=109 time=1.74 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=109 time=2.19 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=109 time=2.49 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=109 time=1.41 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=109 time=3.04 ms
```

Passaggio 4. (Facoltativo) Abilitare il rilevamento BFD (Bi-Directional Forwarding Detection) e un protocollo di routing come EIGRP (Enhanced Interior Gateway Routing Protocol) o BGP (Border Gateway Protocol) sul tunnel per il rilevamento degli errori dei peer. Configurare un tunnel VxLAN o IPsec tra i router Cisco CSR 1000v.

- Tunnel IPsec tra i router Cisco CSR 1000v.

crypto isakmp policy 1 encr aes 256 authentication pre-share crypto isakmp key cisco address crypto ipsec transform-set uni-perf esp-aes 256 esp-sha-hmac mode tunnel crypto ipsec profile vti-1 set security-association lifetime kilobytes disable set security-association lifetime seconds 86400 set transform-set uni-perf set pfs group2 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 redundancy cloud-ha bfd peer Example - #CSR1 ! 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 10.1.0.11 ! redundancy cloud-ha bfd peer 192.168.1.2 #CSR2 ! 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 10.1.0.10 ! redundancy cloud-ha bfd peer 192.168.1.1

- Tunnel VxLAN tra i router Cisco CSR 1000v.

Example: interface Tunnel100 ip address 192.168.1.1 255.255.255.0 bfd interval 500 min_rx 500 multiplier 3 tunnel source GigabitEthernet1 tunnel mode vxlan-gpe ipv4 tunnel destination tunnel vxlan vni 10000 redundancy cloud-ha bfd peer

Passaggio 4.1. (Facoltativo) Configurare le interfacce EIGRP over Tunnel.

router eigrp 1 bfd interface Tunnel1 network 192.168.1.0 0.0.0.255

- Gli script personalizzati possono essere utilizzati per attivare il failover, ad esempio:

event manager applet Interface_GigabitEthernet2 event syslog pattern “Interface GigabitEthernet2, changed state to administratively down” action 1 cli command “enable” action 2 cli command “guestshell run node_event.py -i 10 -e peerFail” exit exit

Configurazione specifica AWS

- Parametri AWS HA

Parameter	Switch	Description
Node Index	-i	Index that is used to uniquely identify this node. Valid values: 1-1023.
Region Name	-rg	Name of the region that contains the route table. For example, us-west-2.
Route Table Name	-t	Name of the route table to be updated. The name of the route table must begin with the substring rtb-. For example, rtb-001333c29ef2aec5f
Route	-r	If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table. The CSR cannot change routes which are of type local or gateway.
Next Hop Interface	-n	Name of the interface to which packets should be forwarded in order to reach the destination route. The name of the interface must begin with the substring eni-. For example, eni-07160c7e740ac8ef4.
Mode	-m	Indicates whether this router is the primary or secondary router for servicing this route. Valid values are primary or secondary. This is an optional parameter. The default value is secondary.

Passaggio 1. Configurare l'autenticazione con IAM.

Affinché il router CSR1000v possa aggiornare una tabella di routing nella rete AWS, è necessario autenticare il router. In AWS, è necessario creare un criterio che consenta al router CSR 1000v di accedere alla tabella di routing. Viene quindi creato un ruolo IAM che utilizza questo criterio e viene applicato alla risorsa EC2.

Dopo aver creato le istanze di CSR 1000v EC2, il ruolo IAM creato deve essere collegato a ciascun router

Il criterio utilizzato nel nuovo ruolo IAM è:

{ "Version": "2012-10-17", "Statement": [{ "Sid": "VisualEditor0", "Effect": "Allow", "Action": ["logs:CreateLogStream", "cloudwatch:", "s3:", "ec2:AssociateRouteTable", "ec2:CreateRoute", "ec2:CreateRouteTable", "ec2:DeleteRoute", "ec2:DeleteRouteTable", "ec2:DescribeRouteTables", "ec2:DescribeVpcs", "ec2:ReplaceRoute", "ec2:DescribeRegions", "ec2:DescribeNetworkInterfaces", "ec2:DisassociateRouteTable", "ec2:ReplaceRouteTableAssociation", "logs:CreateLogGroup", "logs:PutLogEvents"], "Resource": "*" }] }

Nota: Per ulteriori informazioni, fare riferimento al ruolo IAM con un criterio e associarlo al VPC.

Passaggio 2. Installare il pacchetto Python HA.

```
[guestshell@guestshell ~]$ pip install csr_aws_ha --user
[guestshell@guestshell ~]$ source ~/.bashrc
```

Passaggio 3. Configurare i parametri HA sul router primario.

```
[guestshell@guestshell ~]$ create_node.py -i 10 -t rtb-01c5b0633a3422575 -rg ca-central-1 -n eni-0bc1912748614df2a -r 0.0.0.0/0 -m primary
```

Passaggio 4. Configurare i parametri HA sul router secondario.

```
[guestshell@guestshell ~]$ create_node.py -i 10 -t rtb-01c5b0633a3422575 -rg ca-central-1 -n eni-0e351ab1b8f416728 -r 0.0.0.0/0 -m secondary
```

- Formato nodo:

```
create_node.py -i n -t rtb-private-route-table-id -rg region-id -n eni-CSR-id -r route(x.x.x.x/x) -m
```

Configurazione specifica di Azure

- Parametri di Azure HA

The following table specifies the redundancy parameters that are specific to Microsoft Azure:

Parameter Switch	Switch	Description
Node Index	-i	The index that is used to uniquely identify this node. Valid values: 1-255.
Cloud Provider	-p	Specifies the type of Azure cloud: azure, azusgov, or azchina.
Subscription ID	-s	The Azure subscription id.
Resource Group Name	-g	The name of the route table to be updated.
Route Table Name	-t	The name of the route table to be updated.
Route	-r	IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table of type "virtual appliance".
Next Hop Address	-n	The IP address of the next hop router. Use the IP address that is assigned to this CSR 1000v on the subnet which utilizes this route table. Can be an IPv4 or IPv6 address.
Mode	-m	Indicates whether this router is the primary or secondary router for servicing this route. Default value is secondary.

Nota: L'interfaccia rivolta verso l'esterno deve essere configurata su Gigabit Ethernet1. Questa è l'interfaccia usata per raggiungere le API di Azure. HA non può funzionare correttamente in caso contrario. All'interno di guestshell, verificare che il comando curl possa recuperare i metadati da Azure.

```
[guestshell@guestshell ~]$ curl -H "Metadata:true" http://169.254.169.254/metadata/instance?api-version=2020-06-01
```

Passaggio 1. L'autenticazione per le chiamate all'API CSR1000v deve essere abilitata con Azure Active Directory (AAD) o Managed Service Identity (MSI). Per ulteriori informazioni, fare riferimento a Configurazione dell'autenticazione per le chiamate all'API CSR1000v. Senza questo

passaggio, il router CSR1000v non può essere autorizzato ad aggiornare la tabella dei percorsi.

Parametri AAD

Parameter Name	Switch	Description
Cloud Provider	-p	Specifies which Azure cloud is in use {azure | azusgov | azchina}
Tenant ID	-d	Identifies the AAD instance.
Application ID	-a	Identifies the application in AAD.
Application Key	-k	Access key that is created for the application. Key should be specified in unencoded URL format.

Passaggio 2. Installare il pacchetto Python HA.

```
[guestshell@guestshell ~]$ pip install csr_azure_ha --user
[guestshell@guestshell ~]$ source ~/.bashrc
```

Passaggio 3. Configurare i parametri HA sul router primario (per questo passaggio è possibile utilizzare MSI o AAD).

- Con autenticazione MSI.

```
[guestshell@guestshell ~]$ create_node -i 10 -p azure -s xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx -g ResourceGroup -t Private-RouteTable -r 0.0.0.0/0 -n 10.1.0.10 -m primary
```

- Con autenticazione AAD (sono necessari ulteriori flag -a, -d, -k).

```
[guestshell@guestshell ~]$ create_node -i 10 -p azure -s xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx -g ResourceGroup -t Private-RouteTable -r 0.0.0.0/0 -n 10.1.0.10 -m primary -a 1e0f69c3-b6aa-46cf-b5f9-xxxxxxxxx -d ae49849c-2622-4d45-b95e-xxxxxxxxx -k bDEN1k8batJqpeqjAuUvaUCZn5Md6rWEi=
```

Passaggio 4. Configurare i parametri HA sul router secondario.

- Con autenticazione MSI

```
[guestshell@guestshell ~]$ create_node -i 10 -p azure -s xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx -g ResourceGroup -t Private-RouteTable -r 0.0.0.0/0 -n 10.1.0.11 -m secondary
```

- Con autenticazione AAD (sono necessari ulteriori flag -a, -d, -k)

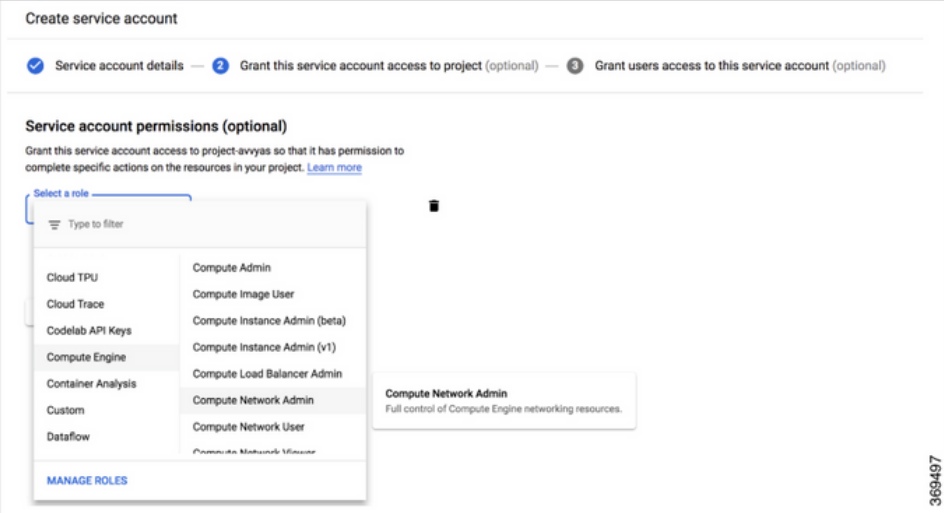
```
[guestshell@guestshell ~]$ create_node -i 10 -p azure -s xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx --g ResourceGroup -t Private-RouteTable -r 0.0.0.0/0 -n 10.0.0.11 -m secondary -a 1e0f69c3-b6aa-46cf-b5f9-xxxxxxxxx -d ae49849c-2622-4d45-b95e-xxxxxxxxx -k bDEN1k8batJqpeqjAuUvaUCZn5Md6rWEi=
```

Configurazione specifica di GCP

- Parametri GCP HA

Parameter	Is this parameter required?	Switch	Description
Node Index	Yes	-i	The index that is used to uniquely identify this node. Valid values: 1-255.
Cloud Provider	Yes	-p	Specify gcp for this parameter.
Project	Yes	-g	Specify the Google Project ID.
routeName	Yes	-a	The route name for which this CSR is next hop. For example from Fig. 2, if we are configuring node on CSR 1, this would be route-vpc2-csr1.
peerRouteName	Yes	-b	The route name for which the BFD peer CSR is next hop. For example from Fig. 2, if we are configuring node on CSR 1, this would be route-vpc2-csr2.
Route	yes	-r	The IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table of type virtual appliance. Note: Currently Google cloud does not have IPv6 support in VPC.
Next hop address	Yes	-n	The IP address of the next hop router. Use the IP address that is assigned to this CSR 1000v on the subnet which utilizes this route table. The value can be an IPv4 or IPv6 address. Note: Currently Google cloud does not have IPv6 support in VPC.
hopPriority	Yes	-o	The route priority for the route for which the current CSR is the next hop.
VPC	Yes	-v	The VPC network name where the route with the current CSR as the next hop exists.

Nota: Verificare che l'account del servizio associato ai router CSR 1000v disponga almeno di un'autorizzazione di amministratore di rete di calcolo.

Command or Action	Purpose
Ensure that the service account associated with the CSR 1000v routers at least have a Compute Network Admin permission.	 You can also provide the required permissions in a credentials file with name 'credentials.json' and place it under the /home/guestshell directory. The credentials file overrides the permissions supplied through the service account associated with the CSR 1000v instance.

Passaggio 1. Installare il pacchetto Python HA.

```
[guestshell@guestshell ~]$ pip install csr_gcp_ha --user
[guestshell@guestshell ~]$ source ~/.bashrc
```

Passaggio 2. Configurare i parametri HA sul router primario.

[guestshell@guestshell ~]$ create_node -i 1 -g -r dest_network -o 200 -n nexthop_ip_addr -a route-vpc2-csr1 -b route-vpc2-csr2 -p gcp -v vpc_name

Passaggio 3. Configurare i parametri HA sul router secondario.

[guestshell@guestshell ~]$ create_node -i 1 -g -r dest_network -o 200 -n nexthop_ip_addr -a route-vpc2-csr2 -b route-vpc2-csr1 -p gcp -v vpc_name

Verifica

Per verificare che la configurazione funzioni correttamente, consultare questa sezione.

Passaggio 1. Attivare un failover con il flag peerFail node_event.py.

[guestshell@guestshell ~]$ node_event.py -i 10 -e peerFail 200: Node_event processed successfully

Passaggio 2. Passare alla tabella delle route private del provider del cloud, verificare che la route abbia aggiornato l'hop successivo al nuovo indirizzo IP.

Risoluzione dei problemi

Al momento non sono disponibili informazioni specifiche per la risoluzione dei problemi di questa configurazione.

Informazioni correlate

- Per informazioni dettagliate sulla configurazione di HAv3, consultare la guida alla configurazione di Cisco CSR 1000v e del software Cisco ISRv
- La configurazione di Azure HAv2 è simile a HAv3, con differenze minime nei pacchetti di installazione pip e nella configurazione di ridondanza IOS. La documentazione è disponibile nella guida alla configurazione di CSR1000v HA versione 2 in Microsoft Azure
- La configurazione di Azure HAv1 con CLI è disponibile nella guida alla distribuzione della ridondanza HA CSR1000v in Microsoft Azure con Azure CLI 2.0
- La configurazione di AWS HAv1 è disponibile nella guida all'installazione della ridondanza HA CSR1000v su Amazon AWS
- Documentazione e supporto tecnico – Cisco Systems