

Guia de distribuição da Redundância CSR1000v HA no Microsoft Azure com AzureCLI 2.0

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Objetivo](#)

[Topologia](#)

[Diagrama de Rede](#)

[Terminology](#)

[Restrições](#)

[Configuração](#)

[Overview](#)

[Etapa 1. Instale AzureCLI 2.0.](#)

[Etapa 2. Crie um grupo de recursos.](#)

[Etapa 3. Crie um Vnet.](#)

[Etapa 4. Crie tabelas de rota.](#)

[Etapa 5. Crie sub-redes.](#)

[Etapa 6. Crie um roteador CSR1000v.](#)

—

[Etapa 7. Crie o segundo roteador CSR1000v.](#)

[Etapa 8. Crie um host VM com o mesmo procedimento na etapa 6. Este exemplo usa UbuntuLTS.](#)

[Etapa 9. Adicionar rotas às tabelas de roteamento e aos VM.](#)

[Etapa 10. Configurar o Roteadores CSR1000v.](#)

[Verifique a Alta disponibilidade](#)

[Troubleshooting](#)

[Informações Relacionadas](#)

Introdução

Este original fornece um manual de configuração passo a passo em como distribuir o Roteadores CSR1000v para a Alta disponibilidade na nuvem do Microsoft Azure com o AzureCLI 2.0. É apontado dar a usuários o conhecimento prático do HA e a capacidade distribuir inteiramente o testbed funcional a -.

Há uns vários métodos para distribuir imagens em azuis celestes e o método o mais familiar para a maioria de usuários é através do portal da web. Contudo, AzureCLI é uma ferramenta rápida e poderosa uma vez que você é familiar com ela.

Para um fundo mais detalhado sobre azuis celestes, como distribuir um CSR1000v através do portal da web, e o HA, refira o [guia de distribuição de Cisco CSR 1000v para o Microsoft Azure](#) e

a seção Informação Relacionada.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Uma conta do Microsoft Azure
- 2 CSR1000v e 1 Windows/máquina virtual de Linux
- AzureCLI 2.0

Componentes Utilizados

A informação neste documento é baseada em Cisco IOS-XE® Denali 16.7.1

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se sua rede está viva, assegure-se de que você compreenda o impacto potencial do comando any.

Objetivo

Distribua 2 Roteadores CSR1000v e 1 VM (indicadores/linux). Simule o tráfego contínuo do datacenter privado (VM) ao Internet (8.8.8.8). Simule um Failover HA e observe que o HA sucedeu confirmando que a tabela de roteamento dos azuis celestes tem o tráfego comutado de CSR-A à interface confidencial do CSR-b.

Topologia

A fim compreender inteiramente a topologia e o projeto é importante antes do início da configuração. Isto ajuda a pesquisar defeitos mais tarde todos os problemas potenciais.

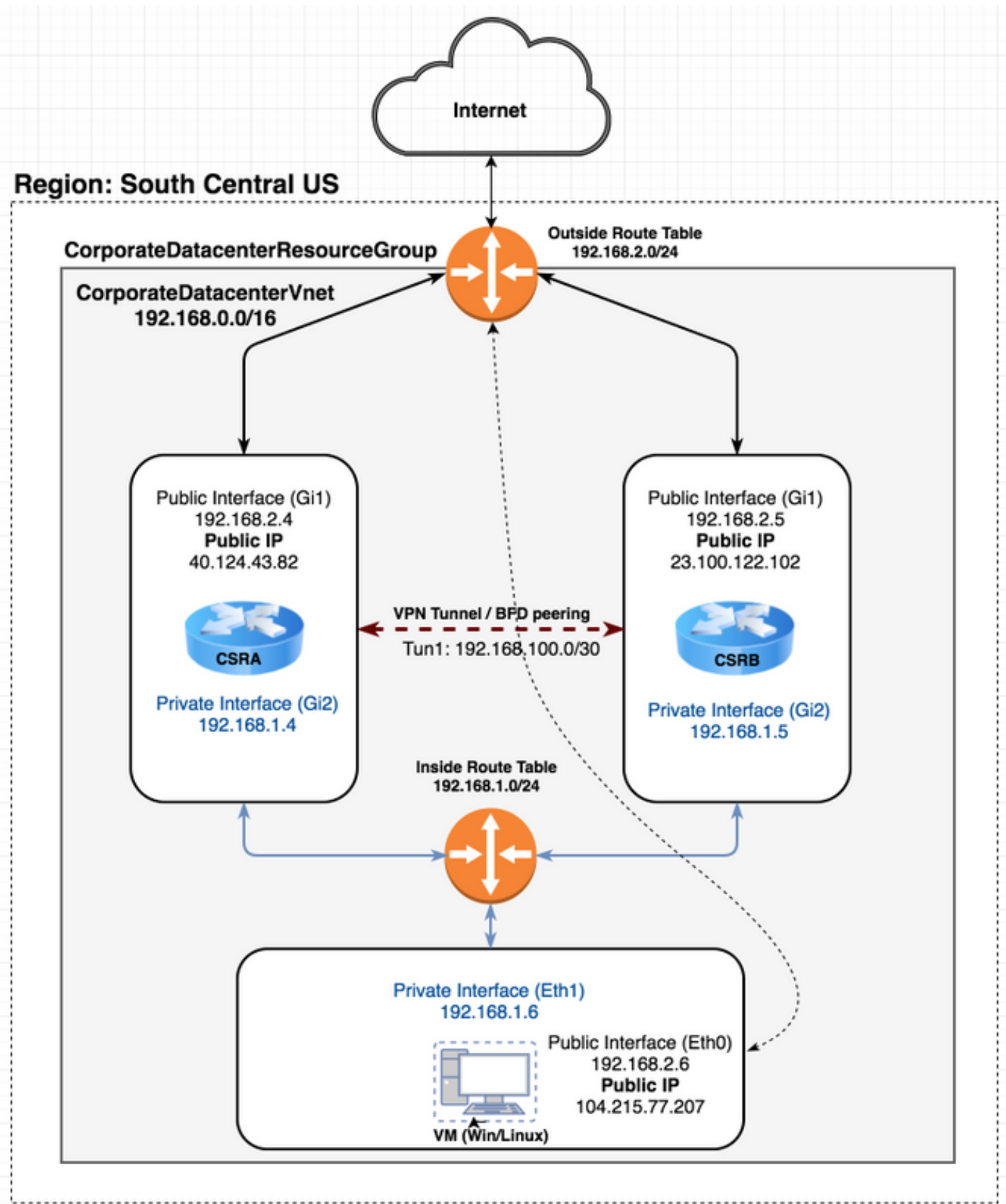
Pode haver umas várias encenações das disposições HA baseadas nas exigências de usuário. Para este exemplo, configurar a Redundância HA com estes ajustes:

- 1x - Região (central sul E.U.)
- 1x - Grupo de recursos (CorporateDatacenterResourceGroup)
- 1x - Vnet (CorporateDatacenterVnet)
- 6x - Interfaces de rede (revestimento 3x interno e revestimento 3x exterior)
- 2x - Tabelas de rota (InsideRoutetable e OutsideRoutetable)
- 2x - Roteadores CSR1000v (® Denali 16.7.1 do Cisco IOS XE)
- 1x - VM (Linux/Windows)

Por agora, o acesso à internet através da interface pública é deixado permitido no VM de modo que você possa o alcançar e configurar. Geralmente, todo o tráfego normal deve correr através da tabela de rota privada. A interface pública no VM pode ser uma deficiente mais atrasado de modo que o sem tráfego seja escapado acidentalmente.

A simulação do tráfego é executada simulando do → da interface confidencial do VM dentro do → 8.8.8.8 do → CSRA da tabela de rota. Em um cenário de failover, observe que a tabela de rota privada comutou a rota para apontar à interface confidencial do CSRB.

Diagrama de Rede



Terminology

- Grupo de recursos - Esta é uma maneira para que os azuis celestes mantenham-se a par de todos seus recursos como máquinas virtuais e vnets. Isto é usado geralmente para controlar todos os artigos e para manter-se a par de cargas.
- Vnet - Uma rede virtual. (similar ao VPC na terminologia dos aws)
- Tabela de rota - Isto contém as regras para uma sub-rede e pode enviar o tráfego específico a um endereço IP de Um ou Mais Servidores Cisco ICM NT ou atuar como um ponto final de VPN.

Restrições

- Os azuis celestes próprios podem introduzir aproximadamente um segundo atraso 40-50 em um Failover HA.

Configuração

Há alguns métodos para distribuir VM em azuis celestes:

1. [Portal da web](#) - Documentação HA no cisco.com
2. Powershell - Modelo baseado da linha de comando para controlando recursos dos azuis celestes.
3. [AzureCLI 2.0](#) - Iguamente linha de comando baseada. É aberta e escrito no pitão e precisa de ser instalado em seu sistema local. A fim redigir este original, AzureCLI 2.0 é a versão a mais atrasada.
4. [Shell da nuvem dos azuis celestes](#) - Escolha a opção do **shell da festança** em vez da opção de **Powershell** usar AzureCLI através do shell. Nenhuma instalação é necessária para este método.

```
← → ↻ 🔒 Secure | https://shell.azure.com
Azure Cloud Shell
Bash ▾ | ⏻ ? ⚙️ 📁 {}
Requesting a Cloud Shell. Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI 2.0
Type "help" to learn about Cloud Shell

david@Azure:~$ az interactive

Do you agree to sending telemetry (yes/no)? Default answer is yes: yes
az>>
```

Powershell e AzureCLI são similares mas os comandos para AzureCLI são mais diretos. Ambos podem ser executado em Windows, MacOS, Linux. Refira a [escolha do trabalho feito com](#)

[ferramentas direito para comandos dos azuis celestes e de lado a lado dos azuis celestes CLI e do PowerShell](#) para uma comparação.

Para este exemplo, distribua todos os recursos com o AzureCLI ou o shell da nuvem. AzureCLI pode ser instalado no MacOS, no Windows ou no Linux com etapas levemente diferentes. Não há nenhuma diferença na configuração com o resto do procedimento entre AzureCLI e shell da nuvem dos azuis celestes.

```
redundancy
cloud provider azure 100
  bfd peer
  route-table
  default-gateway ip
  cidr ip
  app-key
  subscription-id
  app-id
  tenant-id
  resource-group
```

Note: Este molde é útil manter-se a par dos todos os ID e configuração que é usada mais tarde para configurar o HA nos CSR.

Visão geral

Etapa 1. Instale AzureCLI 2.0.

1. Siga as etapas de instalação para Windows, o MacOS, ou o Linux na documentação de [AzureCLI 2.0](#).
2. Para o MacOS:

```
$ brew update && brew install azure-cli
```
3. Entre aos azuis celestes e siga as instruções para autenticar sua sessão.

```
$ az login
```
4. Uma vez que a autenticação de navegador é terminada, sua informação da assinatura dos azuis celestes está retornada no formato JSON:

```
[
  {
    "cloudName": "AzureCloud",
    "id": "09e13fd4-def2-46aa-xxxx-xxxxxxxxxxxx",
    "isDefault": true,
    "name": "Microsoft Azure Enterprise",
    "state": "Enabled",
    "tenantId": "ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxx",
    "user": {
      "name": "cisco@cisco.com",
      "type": "user"
    }
  }
]
```

5. Antes que você obtenha começado com o resto das etapas de configuração, estão aqui alguns comandos úteis e pontas em AzureCLI.
- Para a ajuda com secundário-comandos disponíveis e o que fazem, use - a opção h.

```
$ az account -h
```

- Todas as saídas são retornadas no formato JSON à revelia. Para uma legibilidade mais fácil, você pode usar --opção da **tabela de emissor** a indicar em uma tabela.

```
$ az account list-locations --output table
```

- Obtenha uma lista de todos os vm disponíveis ou substitua --**toda a** opção com uma das outras opções abaixo filtrar a tabela.

```
$ az vm image list --all --output table
```

You are retrieving all the images from server which could take more than a minute. To shorten the wait, provide '--publisher', '--offer' or '--sku'. Partial name search is supported.

- Refira a documentação dos [azuis celestes CLI 2.0 de Microsoft](#) para informações detalhadas sobre de todos os comandos configuration.

Etapa 2. Crie um grupo de recursos.

- Um grupo de recursos é um recipiente que guarde recursos relacionados para uma solução dos azuis celestes. Dê um nome a seu grupo de recursos e escolha um lugar para distribuir o recipiente. Este exemplo usa a central sul E.U.

```
$ az account list-locations --output table
```

DisplayName	Latitude	Longitude	Name
East Asia	22.267	114.188	eastasia
Southeast Asia	1.283	103.833	southeastasia
Central US	41.5908	-93.6208	centralus
East US	37.3719	-79.8164	eastus
East US 2	36.6681	-78.3889	eastus2
West US	37.783	-122.417	westus
North Central US	41.8819	-87.6278	northcentralus
South Central US	29.4167	-98.5	southcentralus

```
$ az group create --name CorporateDatacenterResourceGroup --location "South Central US"
```

```
{
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup",
  "location": "southcentralus",
  "managedBy": null,
  "name": "CorporateDatacenterResourceGroup",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null
}
```

- Molde (que adiciona o grupo de recursos)

```
redundancy
cloud provider azure 100
bfd peer
route-table
default-gateway ip
```

```
cidr ip
app-key
subscription-id
app-id
tenant-id
resource-group CorporateDatacenterResourceGroup
```

Etapa 3. Crie um Vnet.

- Um Vnet é um espaço dos endereços IP de Um ou Mais Servidores Cisco ICM NT onde nossa rede é distribuída. Esta escala então é rachada em sub-redes menores e atribuída às relações. Dê um nome a seu vnet, atribua-o no grupo de recursos criado em etapa 2 e atribua-o uma escala do prefixo. Se você não especifica um prefixo, o azul celeste atribui-lhe geralmente 10.0.0.0/16.

```
$ az network vnet create --name CorporateDatacenterVnet --resource-group
CorporateDatacenterResourceGroup --address-prefix 192.168.0.0/16
{
  "newVNet": {
    "addressSpace": {
      "addressPrefixes": [
        "192.168.0.0/16"
      ]
    },
    "ddosProtectionPlan": null,
    "dhcpOptions": {
      "dnsServers": []
    },
    "enableDdosProtection": false,
    "enableVmProtection": false,
    "etag": "W/\\"7c39a7a9-46e5-4082-a016-xxxxxxxxxxxxx\\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/virtual
Networks/CorporateDatacenterVnet",
    "location": "southcentralus",
    "name": "CorporateDatacenterVnet",
    "provisioningState": "Succeeded",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "3d95d732-e46a-4fae-a34b-xxxxxxxxxxxxx",
    "subnets": [],
    "tags": {},
    "type": "Microsoft.Network/virtualNetworks",
    "virtualNetworkPeerings": []
  }
}
```

Etapa 4. Crie tabelas de rota.

1. Crie uma tabela de rota para as relações internas do revestimento.

```
$ az network route-table create --name InsideRoutetable --resource-group
CorporateDatacenterResourceGroup
{
  "disableBgpRoutePropagation": false,
  "etag": "W/\\"45088005-cb6f-4356-bb18-xxxxxxxxxxxxx\\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ro
uteTables/InsideRoutetable",
  "location": "southcentralus",
```

```

    "name": "InsideRoutetable",
    "provisioningState": "Succeeded",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "routes": [],
    "subnets": null,
    "tags": null,
    "type": "Microsoft.Network/routeTables"
  }

```

Molde (que adiciona a tabela de rota)

```

redundancy
cloud provider azure 100
bfd peer
route-table InsideRoutetable
default-gateway ip
cidr ip
app-key
subscription-id
app-id
tenant-id
resource-group CorporateDatacenterResourceGroup

```

2. Crie uma tabela de rota para as relações exteriores do revestimento.

```

$ az network route-table create --name OutsideRoutetable --resource-group
CorporateDatacenterResourceGroup
{
  "disableBgpRoutePropagation": false,
  "etag": "W/\a89b6230-9542-468c-b4b2-xxxxxxxxxxxx\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ro
uteTables/OutsideRoutetable",
  "location": "southcentralus",
  "name": "OutsideRoutetable",
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "routes": [],
  "subnets": null,
  "tags": null,
  "type": "Microsoft.Network/routeTables"
}

```

Etapa 5. Crie sub-redes.

1. Crie uma sub-rede de /24 do espaço que você atribuiu para o vnet em etapa 3, a seguir atribua-a à tabela de rota interna.

```

$ az network vnet subnet create --address-prefix 192.168.1.0/24 --name InsideSubnet --
resource-group CorporateDatacenterResourceGroup --vnet-name CorporateDatacenterVnet --
route-table InsideRoutetable
{
  "addressPrefix": "192.168.1.0/24",
  "etag": "W/\a0dbd178-3a45-48fb-xxxx-xxxxxxxxxxxx\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/vi
rtualNetworks/CorporateDatacenterVnet/subnets/InsideSubnet",
  "ipConfigurations": null,
  "name": "InsideSubnet",
  "networkSecurityGroup": null,
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "resourceNavigationLinks": null,

```



```

"routeTable": {
  "disableBgpRoutePropagation": null,
  "etag": null,
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/routeTables/InsideRoutetable",
  "location": null,
  "name": null,
  "provisioningState": null,
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "routes": null,
  "subnets": null,
  "tags": null,
  "type": null
},
"serviceEndpoints": null
}

```

2. Crie uma outra sub-rede de /24 do espaço que você atribuiu para o vnet e atribua-a à tabela de rota exterior.

```

$ az network vnet subnet create --address-prefix 192.168.2.0/24 --name OutsideSubnet --
resource-group CorporateDatacenterResourceGroup --vnet-name CorporateDatacenterVnet --
route-table OutsideRoutetable
{
  "addressPrefix": "192.168.2.0/24",
  "etag": "W/\\"874d1019-90a0-44fd-a09c-0aed8f2ede5b\\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/virtualNetworks/CorporateDatacenterVnet/subnets/OutsideSubnet",
  "ipConfigurations": null,
  "name": "OutsideSubnet",
  "networkSecurityGroup": null,
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "resourceNavigationLinks": null,
  "routeTable": {
    "disableBgpRoutePropagation": null,
    "etag": null,
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/routeTables/OutsideRoutetable",
    "location": null,
    "name": null,
    "provisioningState": null,
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "routes": null,
    "subnets": null,
    "tags": null,
    "type": null
  },
  "serviceEndpoints": null
}

```

Etapa 6. Crie um roteador CSR1000v.

Cada VM precisa de ter 2 relações que (internas e exteriores) significam 2 NIC pelo VM. Crie os 2 NIC e associe um IP do público ao NIC exterior.

1. Crie o endereço IP público.

```

$ az network public-ip create --name CSRAPublicIP --resource-group
CorporateDatacenterResourceGroup --idle-timeout 30 --allocation-method Static
{
  "publicIp": {
    "dnsSettings": null,
    "etag": "W/\\"38306703-153b-456b-b2e4-xxxxxxxxxxxx\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/publicIPAddresses/CSRA",
    "idleTimeoutInMinutes": 30,
    "ipAddress": "40.124.43.82",
    "ipConfiguration": null,
    "ipTags": [],
    "location": "southcentralus",
    "name": "CSRAPublicIP",
    "provisioningState": "Succeeded",
    "publicIpAddressVersion": "IPv4",
    "publicIpAllocationMethod": "Static",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "610e1631-331a-4971-8502-xxxxxxxxxxxx",
    "sku": {
      "name": "Basic",
      "tier": "Regional"
    },
    "tags": null,
    "type": "Microsoft.Network/publicIPAddresses",
    "zones": null
  }
}

```

2. Crie o NIC exterior e associe o endereço IP público a ele.

```

$ az network nic create --name CSRAOutsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet OutsideSubnet --vnet CorporateDatacenterVnet --
public-ip-address CSRAPublicIP
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "plk2sxe5i0llcksytfab.jx.internal.cloudapp.net",
      "internalFqdn": null
    },
    "enableAcceleratedNetworking": false,
    "enableIpForwarding": false,
    "etag": "W/\\"06fd60de-6547-4992-b506-xxxxxxxxxxxx\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/networkInterfaces/CSRAOutsideInterface",
    "ipConfigurations": [
      {
        "applicationGatewayBackendAddressPools": null,
        "applicationSecurityGroups": null,
        "etag": "W/\\"06fd60de-6547-4992-xxxx-xxxxxxxxxxxx\"",
        "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/networkInterfaces/CSRAOutsideInterface/ipConfigurations/ipconfig1",
        "loadBalancerBackendAddressPools": null,
        "loadBalancerInboundNatRules": null,
        "name": "ipconfig1",
        "primary": true,
        "privateIpAddress": "192.168.2.4",
        "privateIpAddressVersion": "IPv4",

```

```

    "privateIpAllocationMethod": "Dynamic",
    "provisioningState": "Succeeded",
    "publicIpAddress": {
      "dnsSettings": null,
      "etag": null,
      "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/publicIPAddresses/CSRAPublicIP",
      "idleTimeoutInMinutes": null,
      "ipAddress": null,
      "ipConfiguration": null,
      "ipTags": null,
      "location": null,
      "name": null,
      "provisioningState": null,
      "publicIpAddressVersion": null,
      "publicIpAllocationMethod": null,
      "resourceGroup": "CorporateDatacenterResourceGroup",
      "resourceGuid": null,
      "sku": null,
      "tags": null,
      "type": null,
      "zones": null
    },
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "subnet": {
      "addressPrefix": null,
      "etag": null,
      "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/virtualNetworks/CorporateDatacenterVnet/subnets/OutsideSubnet",
      "ipConfigurations": null,
      "name": null,
      "networkSecurityGroup": null,
      "provisioningState": null,
      "resourceGroup": "CorporateDatacenterResourceGroup",
      "resourceNavigationLinks": null,
      "routeTable": null,
      "serviceEndpoints": null
    }
  }
},
"location": "southcentralus",
"macAddress": null,
"name": "CSRAOutsideInterface",
"networkSecurityGroup": null,
"primary": null,
"provisioningState": "Succeeded",
"resourceGroup": "CorporateDatacenterResourceGroup",
"resourceGuid": "93413822-e819-4644-ac0d-xxxxxxxxxxxx",
"tags": null,
"type": "Microsoft.Network/networkInterfaces",
"virtualMachine": null
}
}

```

3. Crie o NIC interno.

```

$ az network nic create --name CSRAInsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet InsideSubnet --vnet CorporateDatacenterVnet
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],

```

```

    "dnsServers": [],
    "internalDnsNameLabel": null,
    "internalDomainNameSuffix": "gllzkplk2sxe5i011ccksytfab.jx.internal.cloudapp.net",
    "internalFqdn": null
  },
  "enableAcceleratedNetworking": false,
  "enableIpForwarding": false,
  "etag": "W/\\"bebe539f-b5ff-40fa-a122-5c27951afeb1\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/networkInterfaces/CSRAInsideInterface",
  "ipConfigurations": [
    {
      "applicationGatewayBackendAddressPools": null,
      "applicationSecurityGroups": null,
      "etag": "W/\\"bebe539f-b5ff-40fa-a122-5c27951afeb1\"",
      "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/networkInterfaces/CSRAInsideInterface/ipConfigurations/ipconfig1",
      "loadBalancerBackendAddressPools": null,
      "loadBalancerInboundNatRules": null,
      "name": "ipconfig1",
      "primary": true,
      "privateIpAddress": "192.168.1.4",
      "privateIpAddressVersion": "IPv4",
      "privateIpAllocationMethod": "Dynamic",
      "provisioningState": "Succeeded",
      "publicIpAddress": null,
      "resourceGroup": "CorporateDatacenterResourceGroup",
      "subnet": {
        "addressPrefix": null,
        "etag": null,
        "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/virtualNetworks/CorporateDatacenterVnet/subnets/InsideSubnet",
        "ipConfigurations": null,
        "name": null,
        "networkSecurityGroup": null,
        "provisioningState": null,
        "resourceGroup": "CorporateDatacenterResourceGroup",
        "resourceNavigationLinks": null,
        "routeTable": null,
        "serviceEndpoints": null
      }
    }
  ],
  "location": "southcentralus",
  "macAddress": null,
  "name": "CSRAInsideInterface",
  "networkSecurityGroup": null,
  "primary": null,
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "resourceGuid": "0f7ae52a-47c3-4563-9fe0-b1484e88296e",
  "tags": null,
  "type": "Microsoft.Network/networkInterfaces",
  "virtualMachine": null
}
}

```

4. Aliste as imagens disponíveis CSR1000v em azuis celestes. Este exemplo usa o nome da urna de **cisco:cisco-csr-1000v:16_7:16.7.120171201**.

```

az vm image list --all --publisher Cisco --offer cisco-csr-1000v
[

```

```
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "16_5",
  "urn": "cisco:cisco-csr-1000v:16_5:16.5.120170418",
  "version": "16.5.120170418"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "16_5",
  "urn": "cisco:cisco-csr-1000v:16_5:16.5.220171128",
  "version": "16.5.220171128"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "16_6",
  "urn": "cisco:cisco-csr-1000v:16_6:16.6.120170804",
  "version": "16.6.120170804"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "16_6",
  "urn": "cisco:cisco-csr-1000v:16_6:16.6.220171219",
  "version": "16.6.220171219"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "16_7",
  "urn": "cisco:cisco-csr-1000v:16_7:16.7.120171201",
  "version": "16.7.120171201"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "3_16",
  "urn": "cisco:cisco-csr-1000v:3_16:3.16.420170208",
  "version": "3.16.420170208"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "3_16",
  "urn": "cisco:cisco-csr-1000v:3_16:3.16.520170215",
  "version": "3.16.520170215"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "csr-azure-byol",
  "urn": "cisco:cisco-csr-1000v:csr-azure-byol:16.40.120170206",
  "version": "16.40.120170206"
},
{
  "offer": "cisco-csr-1000v",
  "publisher": "cisco",
  "sku": "csr-azure-byol",
  "urn": "cisco:cisco-csr-1000v:csr-azure-byol:3.16.0",
  "version": "3.16.0"
},
{
```

```

    "offer": "cisco-csr-1000v",
    "publisher": "cisco",
    "sku": "csr-azure-byol",
    "urn": "cisco:cisco-csr-1000v:csr-azure-byol:3.16.2",
    "version": "3.16.2"
  }
]

```

5. Distribua o CSR1000v com o nome da urna da imagem.

```

$ az vm create --resource-group CorporateDatacenterResourceGroup --name CSRA --location
southcentralus --image cisco:cisco-csr-1000v:16_7:16.7.120171201 --nics
CSRAOutsideInterface CSRAInsideInterface --admin-username cisco --admin-password
"Cisco1234567" --authentication-type password
Running ..
{
  "fqdns": "",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Compute/vi
rtualMachines/CSRA",
  "location": "southcentralus",
  "macAddress": "00-0D-3A-5D-83-58,00-0D-3A-5D-89-27",
  "powerState": "VM running",
  "privateIpAddress": "192.168.2.4,192.168.1.4",
  "publicIpAddress": "40.124.43.82",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "zones": ""
}

```

Após alguns minutos, as botas novas CSR1000v acima.

```

$ az vm list --resource-group CorporateDatacenterResourceGroup --show-details --output
table

```

Name	ResourceGroup	PowerState	PublicIps	Fqdns	Location	Zones
CSRA	CorporateDatacenterResourceGroup	VM running	40.124.43.82		southcentralus	

6. Entre ao CSR1000v e verifique a funcionalidade.

```

$ ssh cisco@40.124.43.82
The authenticity of host '40.124.43.82 (40.124.43.82)' can't be established.
RSA key fingerprint is SHA256:q33FHw7RlkDn
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '40.124.43.82' (RSA) to the list of known hosts.
Password:

```

```

CSRA#
CSRA#show ip interface brief
Interface IP-Address OK? Method Status Protocol
GigabitEthernet1 192.168.2.4 YES DHCP up up
GigabitEthernet2 192.168.1.4 YES DHCP up up

```

Etapa 7. Crie o segundo roteador CSR1000v.

1. Crie o endereço IP público.

```

$ az network public-ip create --name CSRBPUBLICIP --resource-group
CorporateDatacenterResourceGroup --idle-timeout 30 --allocation-method Static
{
  "publicIp": {
    "dnsSettings": null,

```

```

    "etag": "W/\\"f0f98dac-ea56-4efe-8da6-81a221ac3474\\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/publicIPAddresses/CSRB",
    "idleTimeoutInMinutes": 30,
    "ipAddress": "23.100.122.102",
    "ipConfiguration": null,
    "ipTags": [],
    "location": "southcentralus",
    "name": "CSRBPublicIP",
    "provisioningState": "Succeeded",
    "publicIpAddressVersion": "IPv4",
    "publicIpAllocationMethod": "Static",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "aa03bc26-22df-4696-bd77-ca29df029d7d",
    "sku": {
      "name": "Basic",
      "tier": "Regional"
    },
    "tags": null,
    "type": "Microsoft.Network/publicIPAddresses",
    "zones": null
  }
}

```

2. Crie o NIC exterior e associe o endereço IP público a ele.

```

$ az network nic create --name CSRBOutsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet OutsideSubnet --vnet CorporateDatacenterVnet --
public-ip-address CSRBPublicIP
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "gllzkplk2sxe5i011ccksytfab.jx.internal.cloudapp.net",
      "internalFqdn": null
    },
    "enableAcceleratedNetworking": false,
    "enableIpForwarding": false,
    "etag": "W/\\"ee0a0b41-42f6-4ac2-91c2-xxxxxxxxxxxx\\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/CSRBOutsideInterface",
    "ipConfigurations": [
      {
        "applicationGatewayBackendAddressPools": null,
        "applicationSecurityGroups": null,
        "etag": "W/\\"ee0a0b41-42f6-4ac2-91c2-xxxxxxxxxxxx\\"",
        "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/CSRBOutsideInterface/ipConfigurations/ipconfig1",
        "loadBalancerBackendAddressPools": null,
        "loadBalancerInboundNatRules": null,
        "name": "ipconfig1",
        "primary": true,
        "privateIpAddress": "192.168.2.5",
        "privateIpAddressVersion": "IPv4",
        "privateIpAllocationMethod": "Dynamic",
        "provisioningState": "Succeeded",
        "publicIpAddress": {
          "dnsSettings": null,
          "etag": null,
          "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-

```

```

xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/publicIPAddresses/CSRBPUBLICIP",
    "idleTimeoutInMinutes": null,
    "ipAddress": null,
    "ipConfiguration": null,
    "ipTags": null,
    "location": null,
    "name": null,
    "provisioningState": null,
    "publicIpAddressVersion": null,
    "publicIpAllocationMethod": null,
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": null,
    "sku": null,
    "tags": null,
    "type": null,
    "zones": null
  },
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "subnet": {
    "addressPrefix": null,
    "etag": null,
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/virtualNetworks/CorporateDatacenterVnet/subnets/OutsideSubnet",
    "ipConfigurations": null,
    "name": null,
    "networkSecurityGroup": null,
    "provisioningState": null,
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceNavigationLinks": null,
    "routeTable": null,
    "serviceEndpoints": null
  }
}
],
"location": "southcentralus",
"macAddress": null,
"name": "CSRBOURSIDEINTERFACE",
"networkSecurityGroup": null,
"primary": null,
"provisioningState": "Succeeded",
"resourceGroup": "CorporateDatacenterResourceGroup",
"resourceGuid": "c3f05156-ad07-4abd-a006-xxxxxxxxxxxx",
"tags": null,
"type": "Microsoft.Network/networkInterfaces",
"virtualMachine": null
}
}

```

3. Crie o NIC interno.

```

$ az network nic create --name CSRBINsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet InsideSubnet --vnet CorporateDatacenterVnet
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "zkplk2sxe5i01lckcsytfab.jx.internal.cloudapp.net",
      "internalFqdn": null
    },
    "enableAcceleratedNetworking": false,
    "enableIpForwarding": false,

```



```

    "etag": "W/\\"15edf738-fc77-431c-80f3-xxxxxxxxxxxx\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/CSRBIInsideInterface",
    "ipConfigurations": [
      {
        "applicationGatewayBackendAddressPools": null,
        "applicationSecurityGroups": null,
        "etag": "W/\\"15edf738-fc77-431c-80f3-xxxxxxxxxxxx\"",
        "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/CSRBIInsideInterface/ipConfigurations/ipconfig1",
        "loadBalancerBackendAddressPools": null,
        "loadBalancerInboundNatRules": null,
        "name": "ipconfig1",
        "primary": true,
        "privateIpAddress": "192.168.1.5",
        "privateIpAddressVersion": "IPv4",
        "privateIpAllocationMethod": "Dynamic",
        "provisioningState": "Succeeded",
        "publicIpAddress": null,
        "resourceGroup": "CorporateDatacenterResourceGroup",
        "subnet": {
          "addressPrefix": null,
          "etag": null,
          "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/vi
rtualNetworks/CorporateDatacenterVnet/subnets/InsideSubnet",
          "ipConfigurations": null,
          "name": null,
          "networkSecurityGroup": null,
          "provisioningState": null,
          "resourceGroup": "CorporateDatacenterResourceGroup",
          "resourceNavigationLinks": null,
          "routeTable": null,
          "serviceEndpoints": null
        }
      }
    ],
    "location": "southcentralus",
    "macAddress": null,
    "name": "CSRBIInsideInterface",
    "networkSecurityGroup": null,
    "primary": null,
    "provisioningState": "Succeeded",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "085c88fc-9e78-49be-a5a7-xxxxxxxxxxxx",
    "tags": null,
    "type": "Microsoft.Network/networkInterfaces",
    "virtualMachine": null
  }
}

```

4. Distribua o segundo CSR1000v com a mesma imagem **cisco:cisco-csr-1000v:16_7:16.7.120171201**.

```

$ az vm create --resource-group CorporateDatacenterResourceGroup --name CSRBI --location
southcentralus --image cisco:cisco-csr-1000v:16_7:16.7.120171201 --nics
CSRBOOutsideInterface CSRBIInsideInterface --admin-username cisco --admin-password
"Cisc0l234567" --authentication-type password
{
  "fqdns": "",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Compute/vi
rtualMachines/CSRBI",

```

```

"location": "southcentralus",
"macAddress": "00-0D-3A-5D-8C-51,00-0D-3A-5D-85-2A",
"powerState": "VM running",
"privateIpAddress": "192.168.2.5,192.168.1.5",
"publicIpAddress": "23.100.122.102",
"resourceGroup": "CorporateDatacenterResourceGroup",
"zones": ""
}

```

Etapa 8. Crie um host VM com o mesmo procedimento na etapa 6. Este exemplo usa UbuntuLTS.

1. Crie o endereço IP público.

```

$ az network public-ip create --name VMHostPublicIP --resource-group
CorporateDatacenterResourceGroup --idle-timeout 30 --allocation-method Static
{
  "publicIp": {
    "dnsSettings": null,
    "etag": "W/\ "5943a230-1eeb-4cf0-b856-xxxxxxxxxxxx\" ",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/publicIPAddresses/VMHostPublicIP",
    "idleTimeoutInMinutes": 30,
    "ipAddress": "104.215.77.207",
    "ipConfiguration": null,
    "ipTags": [],
    "location": "southcentralus",
    "name": "VMHostPublicIP",
    "provisioningState": "Succeeded",
    "publicIpAddressVersion": "IPv4",
    "publicIpAllocationMethod": "Static",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "ea19c10a-2fd3-498f-b984-xxxxxxxxxxxx",
    "sku": {
      "name": "Basic",
      "tier": "Regional"
    },
    "tags": null,
    "type": "Microsoft.Network/publicIPAddresses",
    "zones": null
  }
}

```

2. Crie o NIC exterior e associe o OutsideSubnet e o endereço IP público a ele. Quando as sub-redes são associadas com os NIC, um endereço IP de Um ou Mais Servidores Cisco ICM NT está atribuído automaticamente ao NIC. Neste exemplo, o OutsideSubnet é 192.168.2.0/24 e o endereço IP de Um ou Mais Servidores Cisco ICM NT atribuído automaticamente ao NIC é 192.168.2.6.

```

$ az network nic create --name VMHostOutsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet OutsideSubnet --vnet CorporateDatacenterVnet --
public-ip-address VMHostPublicIP
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "gzkplk2sxe5i011ccksytfab.jx.internal.cloudapp.net",
      "internalFqdn": null
    },
    "enableAcceleratedNetworking": false,

```

```
"enableIpForwarding": false,
"etag": "W/\\"2c70c97b-6470-42c8-b481-xxxxxxxxxxxx\"",
"id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/VMHostOutsideInterface",
"ipConfigurations": [
  {
    "applicationGatewayBackendAddressPools": null,
    "applicationSecurityGroups": null,
    "etag": "W/\\"2c70c97b-6470-42c8-b481-xxxxxxxxxxxx\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/VMHostOutsideInterface/ipConfigurations/ipconfig1",
    "loadBalancerBackendAddressPools": null,
    "loadBalancerInboundNatRules": null,
    "name": "ipconfig1",
    "primary": true,
    "privateIpAddress": "192.168.2.6",
    "privateIpAddressVersion": "IPv4",
    "privateIpAllocationMethod": "Dynamic",
    "provisioningState": "Succeeded",
    "publicIpAddress": {
      "dnsSettings": null,
      "etag": null,
      "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/pu
blicIPAddresses/VMHostPublicIP",
      "idleTimeoutInMinutes": null,
      "ipAddress": null,
      "ipConfiguration": null,
      "ipTags": null,
      "location": null,
      "name": null,
      "provisioningState": null,
      "publicIpAddressVersion": null,
      "publicIpAllocationMethod": null,
      "resourceGroup": "CorporateDatacenterResourceGroup",
      "resourceGuid": null,
      "sku": null,
      "tags": null,
      "type": null,
      "zones": null
    },
  },
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "subnet": {
    "addressPrefix": null,
    "etag": null,
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/vi
rtualNetworks/CorporateDatacenterVnet/subnets/OutsideSubnet",
    "ipConfigurations": null,
    "name": null,
    "networkSecurityGroup": null,
    "provisioningState": null,
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceNavigationLinks": null,
    "routeTable": null,
    "serviceEndpoints": null
  }
},
"location": "southcentralus",
"macAddress": null,
"name": "VMHostOutsideInterface",
```

```

    "networkSecurityGroup": null,
    "primary": null,
    "provisioningState": "Succeeded",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "89588a04-6ba6-467d-a86f-xxxxxxxxxxxx",
    "tags": null,
    "type": "Microsoft.Network/networkInterfaces",
    "virtualMachine": null
  }
}

```

3. Crie o NIC interno.

```

$ az network nic create --name VMHostInsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet InsideSubnet --vnet CorporateDatacenterVnet
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "zkplk2sxe5i01lcksytfab.jx.internal.cloudapp.net",
      "internalFqdn": null
    },
    "enableAcceleratedNetworking": false,
    "enableIpForwarding": false,
    "etag": "W/\\"dda7eacf-4670-40c2-999c-xxxxxxxxxxxx\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/VMHostInsideInterface",
    "ipConfigurations": [
      {
        "applicationGatewayBackendAddressPools": null,
        "applicationSecurityGroups": null,
        "etag": "W/\\"dda7eacf-4670-40c2-999c-xxxxxxxxxxxx\"",
        "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/VMHostInsideInterface/ipConfigurations/ipconfig1",
        "loadBalancerBackendAddressPools": null,
        "loadBalancerInboundNatRules": null,
        "name": "ipconfig1",
        "primary": true,
        "privateIpAddress": "192.168.1.6",
        "privateIpAddressVersion": "IPv4",
        "privateIpAllocationMethod": "Dynamic",
        "provisioningState": "Succeeded",
        "publicIpAddress": null,
        "resourceGroup": "CorporateDatacenterResourceGroup",
        "subnet": {
          "addressPrefix": null,
          "etag": null,
          "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/vi
rtualNetworks/CorporateDatacenterVnet/subnets/InsideSubnet",
          "ipConfigurations": null,
          "name": null,
          "networkSecurityGroup": null,
          "provisioningState": null,
          "resourceGroup": "CorporateDatacenterResourceGroup",
          "resourceNavigationLinks": null,
          "routeTable": null,
          "serviceEndpoints": null
        }
      }
    ]
  }
}

```

```

    ],
    "location": "southcentralus",
    "macAddress": null,
    "name": "VMHostInsideInterface",
    "networkSecurityGroup": null,
    "primary": null,
    "provisioningState": "Succeeded",
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "resourceGuid": "8ef12cdd-cc31-432e-99cf-xxxxxxxxxxxx",
    "tags": null,
    "type": "Microsoft.Network/networkInterfaces",
    "virtualMachine": null
  }
}

```

4. Distribua Ubuntu VM. Este exemplo usa UbuntuLTS.

```
az vm image list --output table
```

You are viewing an offline list of images, use --all to retrieve an up-to-date list

Offer	Publisher	Skus	Urn
UrnAlias	Version		
CentOS	OpenLogic	7.3	OpenLogic:CentOS:7.3:latest
CentOS	latest		
CoreOS	CoreOS	Stable	CoreOS:CoreOS:Stable:latest
CoreOS	latest		
Debian	credativ	8	credativ:Debian:8:latest
Debian	latest		
openSUSE-Leap	SUSE	42.3	SUSE:openSUSE-Leap:42.3:latest
openSUSE-Leap	latest		
RHEL	RedHat	7.3	RedHat:RHEL:7.3:latest
RHEL	latest		
SLES	SUSE	12-SP2	SUSE:SLES:12-SP2:latest
SLES	latest		
UbuntuServer	Canonical	16.04-LTS	Canonical:UbuntuServer:16.04-
LTS:latest		UbuntuLTS	latest
WindowsServer	MicrosoftWindowsServer	2016-Datacenter	
MicrosoftWindowsServer:WindowsServer:2016-Datacenter:latest			Win2016Datacenter latest
WindowsServer	MicrosoftWindowsServer	2012-R2-Datacenter	
MicrosoftWindowsServer:WindowsServer:2012-R2-Datacenter:latest			Win2012R2Datacenter latest
WindowsServer	MicrosoftWindowsServer	2012-Datacenter	
MicrosoftWindowsServer:WindowsServer:2012-Datacenter:latest			Win2012Datacenter latest
WindowsServer	MicrosoftWindowsServer	2008-R2-SP1	
MicrosoftWindowsServer:WindowsServer:2008-R2-SP1:latest			Win2008R2SP1 latest

```

$ az vm create --resource-group CorporateDatacenterResourceGroup --name VmHost --location southcentralus --image UbuntuLTS --admin-user cisco --admin-password Cisc0l234567 --nics VMHostOutsideInterface VMHostInsideInterface --authentication-type password

```

```

{
  "fqdns": "",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Compute/virtualMachines/VmHost",
  "location": "southcentralus",
  "macAddress": "00-0D-3A-5D-B7-CB,00-0D-3A-5D-B8-9B",
  "powerState": "VM running",
  "privateIpAddress": "192.168.2.6,192.168.1.6",
  "publicIpAddress": "104.215.77.207",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "zones": ""
}

```

Etapa 9. Adicionar rotas às tabelas de roteamento e aos VM.

1. Adicionar uma rota padrão para que a sub-rede interna distribua o tráfego com CSR A ajustando o endereço IP de Um ou Mais Servidores Cisco ICM NT do salto seguinte como 192.168.1.4. Isto é feito no InsideRouteTable.

```
$ az network route-table route create --address-prefix 8.8.8.8/32 --name default_route --next-hop-type VirtualAppliance --resource-group CorporateDatacenterResourceGroup --route-table-name InsideRouteTable --next-hop-ip-address 192.168.1.4
{
  "addressPrefix": "8.8.8.8/32",
  "etag": "W/\"ef9e650a-5d70-455d-b958-5a0efc07e7ad\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/routeTables/InsideRouteTable/routes/default_route",
  "name": "default_route",
  "nextHopIpAddress": "192.168.1.4",
  "nextHopType": "VirtualAppliance",
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup"
}
```

2. Adicionar uma rota para o tráfego na rede para alcançar o Internet no OutsideRouteTable.

```
$ az network route-table route create --address-prefix 8.8.8.8/32 --name internet --next-hop-type Internet --resource-group CorporateDatacenterResourceGroup --route-table-name OutsideRouteTable
{
  "addressPrefix": "8.8.8.8/32",
  "etag": "W/\"d2c7e32e-8d32-4856-a3a6-xxxxxxxxxxxx\"",
  "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/routeTables/OutsideRouteTable/routes/internet",
  "name": "internet",
  "nextHopIpAddress": null,
  "nextHopType": "Internet",
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup"
}
```

3. Entre a Ubuntu VM e adicionar uma rota para forçar o tráfego através da interface interna a 8.8.8.8. A tabela de rota dos azuis celestes usa automaticamente o primeiro IP em uma sub-rede como seu gateway. A sub-rede da interface interna (eth1) é 192.168.1.0/24 que significa que 192.168.1.1 é o endereço gw do padrão para o host VM.

```
$ ifconfig
eth0 Link encap:Ethernet HWaddr 00:0d:3a:5d:b7:cb
inet addr:192.168.2.6 Bcast:192.168.2.255 Mask:255.255.255.0
inet6 addr: fe80::20d:3aff:fe5d:b7cb/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:3986 errors:0 dropped:0 overruns:0 frame:0
TX packets:2881 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:3475393 (3.4 MB) TX bytes:592740 (592.7 KB)

eth1 Link encap:Ethernet HWaddr 00:0d:3a:5d:b8:9b
inet addr:192.168.1.6 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::20d:3aff:fe5d:b89b/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:2 errors:0 dropped:0 overruns:0 frame:0
TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:762 (762.0 B) TX bytes:1620 (1.6 KB)
```

```

$ sudo route add -host 8.8.8.8 gw 192.168.1.1 dev eth1
$ route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
0.0.0.0          192.168.2.1    0.0.0.0         UG    0      0      0 eth0
8.8.8.8          192.168.1.1    255.255.255.255 UGH   0      0      0 eth1
168.63.129.16   192.168.2.1    255.255.255.255 UGH   0      0      0 eth0
169.254.169.254 192.168.2.1    255.255.255.255 UGH   0      0      0 eth0
192.168.1.0      0.0.0.0        255.255.255.0   U     0      0      0 eth1
192.168.2.0      0.0.0.0        255.255.255.0   U     0      0      0 eth0

```

Molde (que adiciona o cidr IP)

```

redundancy
cloud provider azure 100
bfd peer
route-table InsideRoutetable
default-gateway ip
cidr ip 8.8.8.8/32
app-key
subscription-id
app-id
tenant-id
resource-group CorporateDatacenterResourceGroup

```

Note: O NAT deve ser configurado no Roteadores CSR1000v na etapa 10 a fim sibilar o Internet (8.8.8.8). **Note:** As etapas 10-14 cobrem a configuração do Roteadores CSR1000v para o HA. As etapas abreviadas do [guia de distribuição de Cisco CSR 1000v para o Microsoft Azure](#) são começo fornecido de configuram um Trustpool. Visite o guia para detalhes completos.

Etapa 10. Configurar o Roteadores CSR1000v.

1. Configurar um Trustpool em ambo o Roteadores CSR1000v

```

Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.

Router(config)#crypto pki trustpool import url
http://www.cisco.com/security/pki/trs/ios.p7b
Reading file from http://www.cisco.com/security/pki/trs/ios.p7b
Loading http://www.cisco.com/security/pki/trs/ios.p7b !!!
% PEM files import succeeded.

```

2. Configurar um túnel de IPsec entre 1000v Router de Cisco CSR e permita a detecção bidirecional da transmissão (BFD) e um protocolo de roteamento (EIGRP ou BGP) no túnel entre o Roteadores para a detecção de falha do par. **Note:** O endereço de destino de túnel na configuração é o endereço IP público do par CSR. Configuração CSRA

```

crypto isakmp policy 1
  encr aes 256
  authentication pre-share
crypto isakmp key cisco address 0.0.0.0
!
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 Tunnell
ip address 192.168.101.1 255.255.255.252
bfd interval 500 min_rx 500 multiplier 3
tunnel source GigabitEthernet1
tunnel mode ipsec ipv4
tunnel destination 23.100.122.102 /* Public IP of the peer CSR */
tunnel protection ipsec profile vti-1
!
router eigrp 1
bfd all-interfaces
network 192.168.101.0

```

Configuração CSRB

```

crypto isakmp policy 1
encr aes 256
authentication pre-share
crypto isakmp key cisco address 0.0.0.0
!
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 Tunnell
ip address 192.168.101.2 255.255.255.252
bfd interval 500 min_rx 500 multiplier 3
tunnel source GigabitEthernet1
tunnel mode ipsec ipv4
tunnel destination 40.124.43.82 /* Public IP of the peer CSR */
tunnel protection ipsec profile vti-1
!
router eigrp 1
bfd all-interfaces
network 192.168.101.0

```

3. A mesma configuração para o NAT e o roteamento são usados em ambos os Roteadores CSR1000v. Isto é para a alcançabilidade do Internet VM através da interface interna.

```

interface GigabitEthernet1
ip nat outside
!
interface GigabitEthernet2
ip nat inside
!
ip nat inside source list 10 interface GigabitEthernet1 overload
access-list 10 permit 192.168.1.0 0.0.0.255 /* Translating the inside subnet of the VM */
!
ip route 0.0.0.0 0.0.0.0 192.168.2.1
ip route 192.168.1.0 255.255.255.0 GigabitEthernet2 192.168.1.1

```

4. Adicionar controles de acesso (eu estou) para uma tabela de rota. Em AzureCLI, permita o aplicativo (CSRA e CSRB) alterar o InsideRouteTable nos azuis celestes durante um Failover. Note a **identificação do InsideRouteTable** a ser usado como --opção dos **espaços na próxima seção**.

```

$ az network route-table show --resource-group CorporateDatacenterResourceGroup --name
InsideRoutetable
{
  "disableBgpRoutePropagation": false,
  "etag": "W/\"f0c85464-bba0-465a-992a-xxxxxxxxxxxx\" ",

```



```

    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/routeTables/InsideRoutetable",
    "location": "southcentralus",
    "name": "InsideRoutetable",
    ...

```

Molde (que adiciona a assinatura-identificação)

```

redundancy
cloud provider azure 100
bfd peer
route-table InsideRoutetable
default-gateway ip
cidr ip 8.8.8.8/32
app-key
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxx
app-id
tenant-id
resource-group CorporateDatacenterResourceGroup

```

5. Crie mim estão papel para o InsideRouteTable. --a opção dos **espaços** é tomada do campo **identificação da saída** precedente. Note a **APP-identificação**, a **senha** (que é a APP-chave), e a **identificação do inquilino**.

```

$ az ad sp create-for-rbac -n "InsideRouteTableIAM" --role "network contributor" --scopes
/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/routeTables/InsideRoutetable --years 2099
{
"appId": "576dd4f1-c08d-xxxx-xxxx-xxxxxxxxxxxx",
"displayName": "InsideRouteTableIAM",
"name": "http://InsideRouteTableIAM",
"password": "aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxx",
"tenant": "ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxx"
}

```

Molde (que adiciona a APP-chave, a APP-identificação, e a inquilino-identificação)

```

redundancy
cloud provider azure 100
bfd peer
route-table InsideRoutetable
default-gateway ip
cidr ip 8.8.8.8/32
app-key aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxx
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxx
app-id 576dd4f1-c08d-46b9-cccc-xxxxxxxxxxxx
tenant-id ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxx
resource-group CorporateDatacenterResourceGroup

```

6. Configurar a Redundância da nuvem em ambo o Roteadores. A única diferença entre a configuração em ambo o Roteadores é os pares e o gateway padrão do bfd. Configuração **CSRA**

```

redundancy
cloud provider azure 100
bfd peer 192.168.101.2
route-table InsideRoutetable
default-gateway ip 192.168.1.4
cidr ip 8.8.8.8/32
app-key aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxx
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxx
app-id 576dd4f1-c08d-46b9-cccc-xxxxxxxxxxxx
tenant-id ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxx
resource-group CorporateDatacenterResourceGroup

```

Configuração CSRB

```

redundancy

```

```

cloud provider azure 100
bfd peer 192.168.101.1
route-table InsideRoutetable
default-gateway ip 192.168.1.5
cidr ip 8.8.8.8/32
app-key aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxxxx
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx
app-id 576dd4f1-c08d-46b9-cccc-xxxxxxxxxxxxxx
tenant-id ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxxxx
resource-group CorporateDatacenterResourceGroup

```

Verifique a Alta disponibilidade

1. Verifique o BFD e as configurações da nuvem.

```

CSRA#show ip interface brief
Interface                IP-Address      OK? Method Status          Protocol
GigabitEthernet1        192.168.2.4     YES DHCP    up              up
GigabitEthernet2        192.168.1.4     YES DHCP    up              up
Tunnell                  192.168.101.1  YES manual up              up

```

```

CSRB#show ip interface brief
Interface                IP-Address      OK? Method Status          Protocol
GigabitEthernet1        192.168.2.5     YES DHCP    up              up
GigabitEthernet2        192.168.1.5     YES DHCP    up              up
Tunnell                  192.168.101.2  YES NVRAM  up              up

```

```

CSRA#show bfd neighbors

```

```

IPv4 Sessions
NeighAddr                LD/RD           RH/RS           State           Int
192.168.101.2            4097/4097      Up              Up              Tu1

```

```

CSRA#show redundancy cloud provider azure 100
Cloud HA: work_in_progress=FALSE
Provider : AZURE node 100
State : idle
BFD peer      = 192.168.101.2
BFD intf      = Tunnell
resource group = CorporateDatacenterResourceGroup
subscription id = 09e13fd4-def2-46aa-xxxx-xxxxxxxxxxxxxx
tenant id     = ae49849c-2622-4d45-b95e-xxxxxxxxxxxxxx
application id = 1e0f69c3-b6aa-46cf-b5f9-xxxxxxxxxxxxxx
application key = aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxxxx
route-table   = InsideRoutetable
cidr           = 8.8.8.8/32
Default Gateway IP = 192.168.1.4

```

2. Execute um sibilo e um traceroute do VM ao destino. Assegure-se de que o sibilo seja através da relação eth1 interna.

```

$ ping -I eth1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) from 192.168.1.6 eth1: 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=54 time=10.5 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=54 time=10.6 ms

```

```

$ traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.1.4 (192.168.1.4)  1.516 ms  1.503 ms  1.479 ms

```

```

cisco@VmHost:~$ ping -I eth1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) from 192.168.1.6 eth1: 56(84) bytes of data.

```

```
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=10.3 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=10.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=117 time=10.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=117 time=10.2 ms
```

3. Traceroute mostra que o trajeto do VM a 8.8.8.8 é através da interface interna do CSRA.

```
cisco@VmHost:~$ sudo traceroute -I 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.1.4 (192.168.1.4)  34.003 ms  34.000 ms  33.998 ms
```

4. Feche a relação do túnel 1 do CSRA para simular um Failover.

```
CSRA#config t
Enter configuration commands, one per line.  End with CNTL/Z.
CSRA(config)#int tunnell
CSRA(config-if)#sh
```

5. Observe que o tráfego corre através agora da interface confidencial do CSRB.

```
cisco@VmHost:~$ sudo traceroute -I 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.1.5 (192.168.1.5)  1.294 ms  1.291 ms  1.290 ms
```

Note: A nuvem dos azuis celestes pode introduzir um atraso ao falhar sobre. O atraso deve realizar-se já não do que 1 minuto.

Troubleshooting

- Permita debuga para observar mensagens durante o Failover HA.

```
CSRA#debug redundancy cloud all
CSRA#debug ip http all
```

- Os erros da autenticação e das credenciais são devido aos controles de acesso inválidos que permite que o CSR1000v faça atendimentos API à tabela de rota dos azuis celestes. Verifique novamente que as identificações apropriadas estão configuradas na etapa 10.

```
*Jul 13 23:29:53.365: CLOUD-HA : res content iov_len=449
iov_base={"error":"invalid_client","error_description":"AADSTS70002:
Error validating credentials. AADSTS50012: Invalid client secret is provided.\r\nTrace ID:
56873e4b-3781-4ee6-8bd9-xxxxxxxxxxxx\r\n
Correlation ID: cce94817-29eb-4ebd-833a-\r\nTimestamp: 2018-07-13
23:29:54Z","error_codes":[70002,50012],"timestamp":"2018-07-13
23:29:54Z","trace_id":"56873e4b-3781-4ee6-8bd9-xxxxxxxxxxxx","correlation_id":"cce94817-29eb-
4ebd-833a"}
```

Informações Relacionadas

- [Azuis celestes CLI 2.0](#)
- [Guia de distribuição de Cisco CSR 1000v para o Microsoft Azure](#)
- [Escolhendo o trabalho feito com ferramentas direito para comandos dos azuis celestes e de lado a lado dos azuis celestes CLI e do PowerShell](#)