

# Implementierungsleitfaden für CSR1000v HA-Redundanz auf Microsoft Azure mit AzureCLI 2.0

## Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Ziel](#)

[Topologie](#)

[Netzwerkdiagramm](#)

[Terminologie](#)

[Einschränkungen](#)

[Konfiguration](#)

[Übersicht](#)

[Schritt 1: AzureCLI 2.0 installieren.](#)

[Schritt 2: Erstellen einer Ressourcengruppe.](#)

[Schritt 3: Erstellen Sie ein Vnet.](#)

[Schritt 4: Erstellen von Routentabellen.](#)

[Schritt 5: Erstellen von Subnetzen.](#)

[Schritt 6: Erstellen Sie einen CSR1000v-Router.](#)

—

[Schritt 7: Erstellen Sie den zweiten CSR1000v-Router.](#)

[Schritt 8: Erstellen Sie eine Host-VM mit dem gleichen Verfahren wie in Schritt 6. In diesem Beispiel wird UbuntuLTS verwendet.](#)

[Schritt 9: Fügen Sie Routen zu Routing-Tabellen und virtuellen Systemen hinzu.](#)

[Schritt 10: Konfigurieren Sie die CSR1000v-Router.](#)

[Überprüfen der Hochverfügbarkeit](#)

[Fehlerbehebung](#)

[Zugehörige Informationen](#)

## Einführung

Dieses Dokument enthält eine schrittweise Anleitung zur Bereitstellung von CSR1000v-Routern für hohe Verfügbarkeit in der Microsoft Azure-Cloud mit AzureCLI 2.0. Es soll den Benutzern praktische Kenntnisse über HA und die Möglichkeit geben, ein voll funktionsfähiges Testbett bereitzustellen.

Es gibt verschiedene Methoden, um Bilder auf Azure bereitzustellen, und die bekannteste Methode für die meisten Benutzer ist das Web-Portal. AzureCLI ist jedoch ein schnelles und leistungsstarkes Tool, sobald Sie es kennen.

Ausführlichere Informationen zu Azure, zur Bereitstellung eines CSR1000v über das Webportal und zu Hochverfügbarkeit finden Sie im [Cisco CSR 1000v-Bereitstellungsleitfaden für Microsoft](#)

[Azure](#) und im zugehörigen Informationsbereich.

# Voraussetzungen

## Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

- Ein Microsoft Azure-Konto
- 2 CSR1000v und 1 virtuelles Windows/Linux-System
- AzureCLI 2.0

## Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf Cisco IOS-XE® Denali 16.7.1

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

## Ziel

Bereitstellung von 2 CSR1000v-Routern und 1 VM (Windows/Linux) Simulieren des kontinuierlichen Datenverkehrs vom privaten Rechenzentrum (VM) zum Internet (8.8.8.8) Simulieren Sie einen HA-Failover, und stellen Sie fest, dass HA erfolgreich war, indem Sie bestätigen, dass die Azure-Routing-Tabelle den Datenverkehr von CSR-A zur privaten Schnittstelle von CSR-B umgeleitet hat.

## Topologie

Um die Topologie und das Design vollständig zu verstehen, ist es vor Beginn der Konfiguration wichtig. So können potenzielle Probleme später behoben werden.

Je nach Benutzeranforderungen können verschiedene HA-Bereitstellungen möglich sein. In diesem Beispiel konfigurieren Sie die HA-Redundanz mit den folgenden Einstellungen:

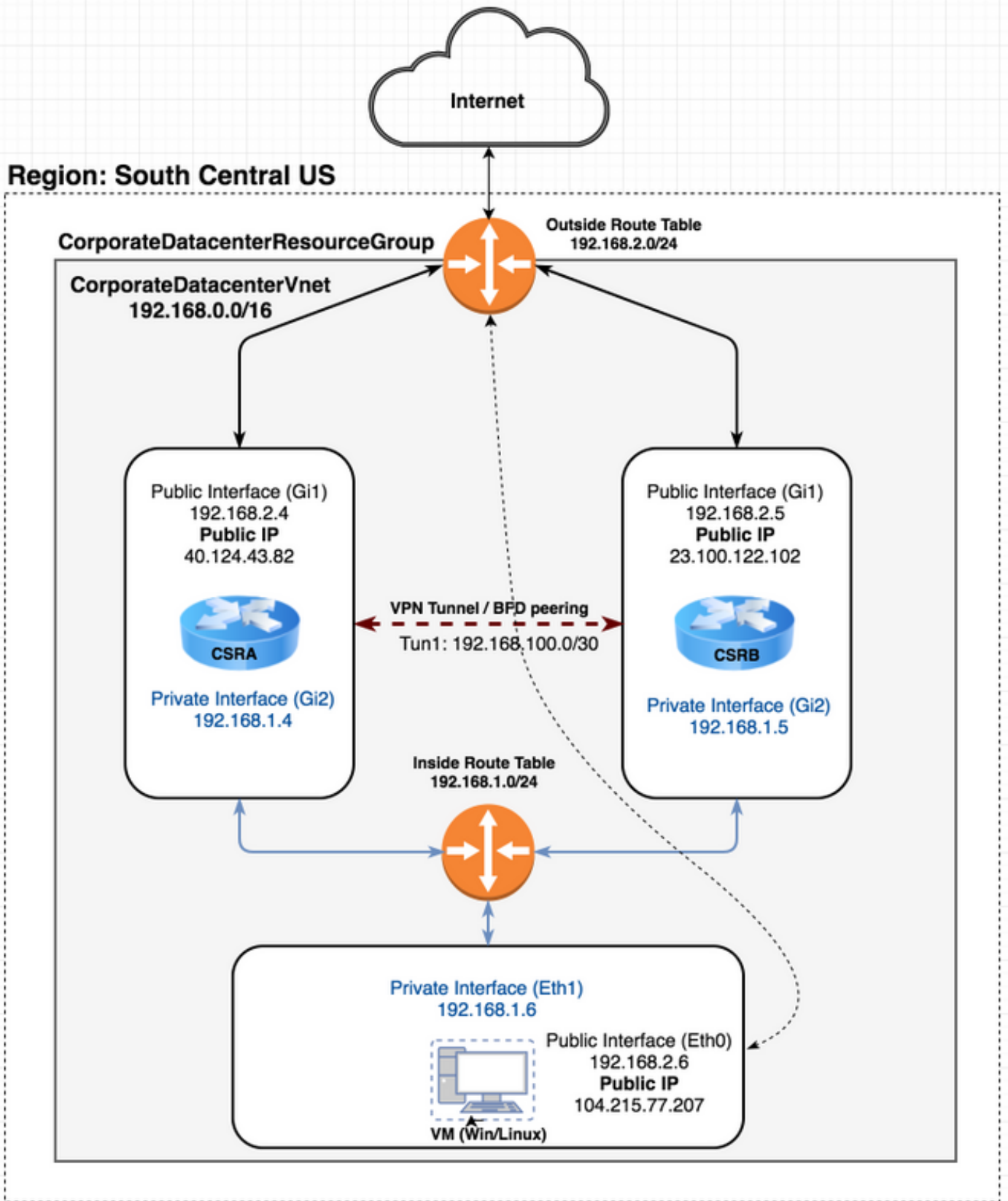
- 1x - Region (South Central US)
- 1x - Ressourcengruppe (CorporateDatacenterResourceGroup)
- 1x - Vnet (CorporateDatacenterVnet)
- 6x - Netzwerkschnittstellen (3x Inside Facing und 3x Outside Facing)
- 2x - Routentabellen (InsideRoutable und OutsideRouttable)
- 2x - CSR1000v-Router (Cisco IOS-XE® Denali 16.7.1)
- 1x - VM (Linux/Windows)

Im Moment ist der Internetzugriff über die öffentliche Schnittstelle auf dem virtuellen System aktiviert, sodass Sie darauf zugreifen und ihn konfigurieren können. Im Allgemeinen sollte der gesamte normale Datenverkehr die private Routing-Tabelle durchlaufen. Die öffentliche Schnittstelle auf der VM kann später deaktiviert werden, sodass kein Datenverkehr versehentlich

durchsickert.

Die Datenverkehrssimulation wird durch Ping-Befehle von der privaten Schnittstelle des virtuellen Systems → innerhalb der Routing-Tabelle → CSRA → 8.8.8.8 durchgeführt. In einem Failover-Szenario beobachten Sie, dass die private Routing-Tabelle die Route auf die private CSRB-Schnittstelle umgeschaltet hat.

## **Netzwerkdiagramm**



## Terminologie

- Resource Group - Azure kann auf diese Weise alle Ressourcen wie virtuelle Systeme und Vnets überwachen. Dieser dient in der Regel dazu, alle Artikel zu verwalten und die Kosten nachzuverfolgen.
- Vnet - Ein virtuelles Netzwerk. (Ähnlich wie VPC in früherer Terminologie)

- Routentabelle - Enthält die Regeln für ein Subnetz und kann bestimmten Datenverkehr an eine IP-Adresse weiterleiten oder wie ein VPN-Endpunkt agieren.

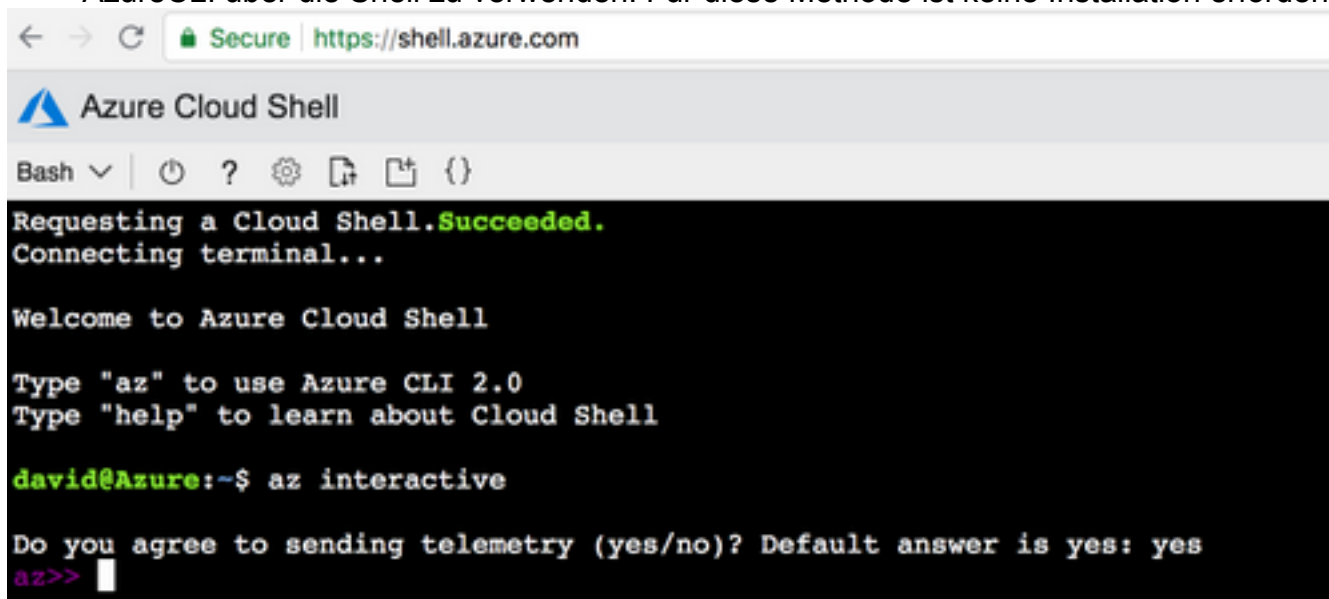
## Einschränkungen

- Azure selbst kann eine Verzögerung zwischen 40 und 50 Sekunden in einem HA-Failover verursachen.

## Konfiguration

Es gibt einige Methoden zur Bereitstellung von VMs auf Azure:

1. [Webportal](#) - HA-Dokumentation auf cisco.com
2. Powershell - Befehlszeilenbasiertes Modell zur Verwaltung von Azure-Ressourcen.
3. [AzureCLI 2.0](#) - Auch Befehlszeilenbasiert. Es ist Open Source und in python geschrieben und muss auf Ihrem lokalen System installiert werden. Zur Erstellung dieses Dokuments ist AzureCLI 2.0 die neueste Version.
4. [Azure Cloud Shell](#): Wählen Sie die **Bash-Shell**-Option anstelle der **Powershell**-Option, AzureCLI über die Shell zu verwenden. Für diese Methode ist keine Installation erforderlich.



```
← → ↻ 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 und AzureCLI sind ähnlich, aber die Befehle für AzureCLI sind einfacher. Beide können unter Windows, MacOS und Linux ausgeführt werden. Einen Vergleich [finden Sie unter Auswählen der richtigen Werkzeuge für Azure und neben Azure CLI sowie PowerShell-Befehle.](#)

In diesem Beispiel können Sie alle Ressourcen entweder mit AzureCLI oder Cloud Shell bereitstellen. AzureCLI kann mit leicht unterschiedlichen Schritten auf MacOS, Windows oder Linux installiert werden. Zwischen AzureCLI und Azure Cloud Shell besteht bei der Konfiguration während des restlichen Verfahrens kein Unterschied.

```
redundancy
cloud provider azure 100
bfd peer
```

```
route-table
default-gateway ip
cidr ip
app-key
subscription-id
app-id
tenant-id
resource-group
```

**Hinweis:** Diese Vorlage ist hilfreich, um alle IDs und Konfigurationen nachzuverfolgen, die später zur Konfiguration der HA auf den CSRs verwendet werden.

## Übersicht

### Schritt 1: AzureCLI 2.0 installieren.

1. Befolgen Sie die Installationsschritte für Windows, MacOS oder Linux in der [AzureCLI 2.0-Dokumentation](#).
2. Für MacOS:  

```
$ brew update && brew install azure-cli
```
3. Melden Sie sich bei Azure an, und befolgen Sie die Anweisungen zur Authentifizierung der Sitzung.  

```
$ az login
```
4. Sobald die Browser-Authentifizierung abgeschlossen ist, werden Ihre Azure-Abonnementinformationen im JSON-Format zurückgegeben:

```
[
  {
    "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. Bevor Sie mit den übrigen Konfigurationsschritten beginnen, finden Sie hier einige nützliche Befehle und Tipps zu AzureCLI.
- Verwenden Sie die **-h**-Option, um Hilfe zu verfügbaren Unterbefehlen und ihren Aufgaben zu erhalten.

```
$ az account -h
```

- Alle Ausgaben werden standardmäßig im JSON-Format zurückgegeben. Um die Lesbarkeit zu vereinfachen, können Sie die Option **—Output table** (Ausgabetable) verwenden, um sie in einer Tabelle anzuzeigen.

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

- Rufen Sie eine Liste aller verfügbaren VMs ab, oder ersetzen Sie die **—all**-Option durch eine der anderen Optionen unten, um die Tabelle zu filtern.

```
$ 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.

- Ausführliche Informationen zu allen Konfigurationsbefehlen finden Sie in der Microsoft [Azure CLI 2.0-Dokumentation](#).

## Schritt 2: Erstellen einer Ressourcengruppe.

- Eine Ressourcengruppe ist ein Container, der die zugehörigen Ressourcen für eine Azure-Lösung enthält. Geben Sie der Ressourcengruppe einen Namen, und wählen Sie einen Speicherort für die Bereitstellung des Containers aus. In diesem Beispiel wird South Central US verwendet.

```
$ 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
}
```

- Vorlage (Hinzufügen der Ressourcengruppe)

```
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
```

## Schritt 3: Erstellen Sie ein Vnet.

- Ein Vnet ist ein Bereich mit IP-Adressen, in dem unser Netzwerk bereitgestellt wird. Dieser

Bereich wird dann in kleinere Subnetze aufgeteilt und Schnittstellen zugewiesen. Geben Sie Ihrem Vnet einen Namen, weisen Sie es der in Schritt 2 erstellten Ressourcengruppe zu und weisen Sie einen Präfixbereich zu. Wenn Sie kein Präfix angeben, weist Azure Ihnen in der Regel 10.0.0.0/16 zu.

```
$ 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": []
  }
}
```

## Schritt 4: Erstellen von Routentabellen.

### 1. Erstellen Sie eine Routentabelle für die Schnittstellen mit Innenbereich.

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

**Vorlage (Hinzufügen einer Routing-Tabelle)**

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. Erstellen Sie eine Routentabelle für die nach außen gerichteten Schnittstellen.

```

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

```

## Schritt 5: Erstellen von Subnetzen.

### 1. Erstellen Sie ein /24-Subnetz aus dem in Schritt 3 für das Vnet zugewiesenen Speicherplatz, und weisen Sie es dann der Tabelle für die interne Route zu.

```

$ 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/virtualNetworks/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. Erstellen Sie ein weiteres /24-Subnetz aus dem für das Vnet reservierten Bereich, und weisen Sie es der Tabelle für die externe Route zu.

```

$ 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/vi
rtualNetworks/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/ro
uteTables/OutsideRoutetable",
    "location": null,
    "name": null,
    "provisioningState": null,
    "resourceGroup": "CorporateDatacenterResourceGroup",
    "routes": null,
    "subnets": null,
    "tags": null,
    "type": null
  },
  "serviceEndpoints": null
}

```

## Schritt 6: Erstellen Sie einen CSR1000v-Router.

Jede VM muss über zwei Schnittstellen (innen und außen) verfügen, was jeweils 2 NICs pro VM bedeutet. Erstellen Sie die zwei NICs, und ordnen Sie eine öffentliche IP der externen Netzwerkkarte zu.

### 1. Erstellen Sie die öffentliche IP-Adresse.

```

$ 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/pu
blicIPAddresses/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. Erstellen Sie die externe Netzwerkkarte, und ordnen Sie die öffentliche IP-Adresse ihr zu.

```

$ az network nic create --name CSRAOutsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet OutsideSubnet --vnet CorporateDatacenterVnet --
public-ip-address CSRAPublicIP
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "plk2sxe5i01lcksytfab.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/ne
tworkInterfaces/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/ne
tworkInterfaces/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/pu
blicIPAddresses/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/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": "CSRAOutsideInterface",
"networkSecurityGroup": null,
"primary": null,
"provisioningState": "Succeeded",
"resourceGroup": "CorporateDatacenterResourceGroup",
"resourceGuid": "93413822-e819-4644-ac0d-xxxxxxxxxxxx",
"tags": null,
"type": "Microsoft.Network/networkInterfaces",
"virtualMachine": null
}
}

```

### 3. Erstellen Sie die interne Netzwerkkarte.

```

$ az network nic create --name CSRAInsideInterface --resource-group
CorporateDatacenterResourceGroup --subnet InsideSubnet --vnet CorporateDatacenterVnet
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "g1lzklk2sxe5i011ccksytfab.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/ne

```

```

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/ne
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/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": "CSRAInsideInterface",
  "networkSecurityGroup": null,
  "primary": null,
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup",
  "resourceGuid": "0f7ae52a-47c3-4563-9fe0-b1484e88296e",
  "tags": null,
  "type": "Microsoft.Network/networkInterfaces",
  "virtualMachine": null
}
}

```

4. Listen Sie die verfügbaren CSR1000v-Bilder auf Azure auf. In diesem Beispiel wird der Abfragenname von **cisco:cisco-csr-1000v:16\_7:16.7.120171201** verwendet.

```

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. Stellen Sie CSR1000v mit dem Namen des Abbilds bereit.

```
$ 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": ""
}

```

**Nach einigen Minuten wird der neue CSR1000v gestartet.**

```

$ 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. Melden Sie sich beim CSR1000v an, und überprüfen Sie die Funktionalität.

```

$ 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

```

## Schritt 7: Erstellen Sie den zweiten CSR1000v-Router.

### 1. Erstellen Sie die öffentliche IP-Adresse.

```

$ 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/pu
blicIPAddresses/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. Erstellen Sie die externe Netzwerkkarte, und ordnen Sie die öffentliche IP-Adresse ihr zu.

```

$ az network nic create --name CSRBOURSIDEINTERFACE --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/CSRBOURSIDEINTERFACE",
    "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/CSRBOURSIDEINTERFACE/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/pu
blicIPAddresses/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/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": "CSRBOutsideInterface",
"networkSecurityGroup": null,
"primary": null,
"provisioningState": "Succeeded",
"resourceGroup": "CorporateDatacenterResourceGroup",
"resourceGuid": "c3f05156-ad07-4abd-a006-xxxxxxxxxxxx",
"tags": null,
"type": "Microsoft.Network/networkInterfaces",
"virtualMachine": null
}
}

```

### 3. Erstellen Sie die interne Netzwerkkarte.

```

$ az network nic create --name CSRBIInsideInterface --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/\"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. Bereitstellen des zweiten CSR1000v mit dem gleichen Bild **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
CSRBOUtsideInterface CSRBIInsideInterface --admin-username cisco --admin-password
"Cisco1234567" --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": ""
}

```

## Schritt 8: Erstellen Sie eine Host-VM mit dem gleichen Verfahren wie in Schritt 6. In diesem Beispiel wird UbuntuLTS verwendet.

### 1. Erstellen Sie die öffentliche IP-Adresse.

```
$ 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. Erstellen Sie die externe Netzwerkkarte, und ordnen Sie das OutsideSubnet und die öffentliche IP-Adresse ihr zu. Wenn Subnetze NICs zugeordnet sind, wird der Netzwerkkarte automatisch eine IP-Adresse zugewiesen. In diesem Beispiel ist das OutsideSubnet 192.168.2.0/24 und die IP-Adresse, die der Netzwerkkarte automatisch zugewiesen wird, 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/networkInterfaces/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. Erstellen Sie die interne Netzwerkkarte.

```
$ az network nic create --name VMHostInsideInterface --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/\\"dda7eacf-4670-40c2-999c-xxxxxxxxxxxxx\\"",
    "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxxx/resourceGroups/CorporateDatacenterResourceGroup/providers/Microsoft.Network/ne
tworkInterfaces/VMHostInsideInterface",
    "ipConfigurations": [
      {
        "applicationGatewayBackendAddressPools": null,
        "applicationSecurityGroups": null,
        "etag": "W/\\"dda7eacf-4670-40c2-999c-xxxxxxxxxxxxx\\"",
        "id": "/subscriptions/09e13fd4-def2-46aa-xxxx-
xxxxxxxxxxxxx/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-
xxxxxxxxxxxxx/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-xxxxxxxxxxxxx",
}
```

```

    "tags": null,
    "type": "Microsoft.Network/networkInterfaces",
    "virtualMachine": null
  }
}

```

#### 4. Stellen Sie die Ubuntu VM bereit. In diesem Beispiel wird UbuntuLTS verwendet.

```
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 Cisc01234567 --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": ""
}

```

### Schritt 9: Fügen Sie Routen zu Routing-Tabellen und virtuellen Systemen hinzu.

1. Fügen Sie eine Standardroute für das interne Subnetz hinzu, um Datenverkehr über CSR A weiterzuleiten, indem Sie die Next-Hop-IP-Adresse auf 192.168.1.4 festlegen. Dies erfolgt auf der 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/ro
uteTables/InsideRouteTable/routes/default_route",
  "name": "default_route",
  "nextHopIpAddress": "192.168.1.4",
  "nextHopType": "VirtualAppliance",
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup"
}

```

## 2. Fügen Sie eine Route für den Datenverkehr im Netzwerk hinzu, um das Internet auf der OutsideRouteTable zu erreichen.

```

$ 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/ro
uteTables/OutsideRouteTable/routes/internet",
  "name": "internet",
  "nextHopIpAddress": null,
  "nextHopType": "Internet",
  "provisioningState": "Succeeded",
  "resourceGroup": "CorporateDatacenterResourceGroup"
}

```

## 3. Melden Sie sich bei der Ubuntu VM an, und fügen Sie eine Route hinzu, um den Datenverkehr durch die interne Schnittstelle zu 8.8.8.8 zu erzwingen. Azure-Routing-Tabelle verwendet automatisch die erste IP-Adresse in einem Subnetz als Gateway. Das Subnetz der Inside-Schnittstelle (eth1) ist 192.168.1.0/24, d. h. 192.168.1.1 ist die Standard-GW-Adresse für die 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

```

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

## Vorlage (Hinzufügen von cir 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

```

**Hinweis:** NAT muss in Schritt 10 auf den CSR1000v-Routern konfiguriert werden, um einen Ping an das Internet zu senden (8.8.8.8). **Hinweis:** Die Schritte 10-14 behandeln die Konfiguration der CSR1000v-Router für HA. Abkürzungen aus dem [Cisco CSR 1000v-Bereitstellungsleitfaden für Microsoft Azure](#) werden ab der Konfiguration eines Trustpools bereitgestellt. Ausführliche Informationen finden Sie im Leitfaden.

## Schritt 10: Konfigurieren Sie die CSR1000v-Router.

### 1. Konfigurieren eines Trustpools auf beiden CSR1000v-Routern

```

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. Konfigurieren Sie einen IPSec-Tunnel zwischen den Cisco CSR 1000v-Routern, und aktivieren Sie BFD (Bi-directional Forwarding Detection) und EIGRP (BGP) im Tunnel zwischen den Routern, um Peer-Fehlererkennung zu ermöglichen. **Hinweis:** Die Tunnelzieladresse in der Konfiguration ist die öffentliche IP-Adresse des Peer-CSR.CSRA-Konfiguration

```

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
CSRB-Konfiguration
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. Auf beiden CSR1000v-Routern werden dieselbe Konfiguration für NAT und Routing verwendet. Dies ist für die Erreichbarkeit des VM-Internets über die interne Schnittstelle vorgesehen.

```

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. Hinzufügen von Zugriffskontrollen (IAM) für eine Routentabelle. In AzureCLI kann die Anwendung (CSRA und CSRB) die InsideRouteTable in Azure während eines Failovers ändern. Beachten Sie die ID der InsideRouteTable, die als **—scopes**-Option im nächsten Abschnitt verwendet werden soll.

```

$ 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/ro

```

```
uteTables/InsideRoutetable",
  "location": "southcentralus",
  "name": "InsideRoutetable",
  ...
```

### Vorlage (Hinzufügen der Abonnement-ID)

```
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-xxxxxxxxxxxxx
app-id
tenant-id
resource-group CorporateDatacenterResourceGroup
```

5. Erstellen Sie die IAM-Rolle für die InsideRouteTable. Die Option **—Scopes** wird aus dem **ID-Feld** der vorherigen Ausgabe übernommen. Notieren Sie sich die **App-ID**, das **Kennwort** (der App-Schlüssel) und die **Tenant-ID**.

```
$ 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-xxxxxxxxxxxxx",
  "displayName": "InsideRouteTableIAM",
  "name": "http://InsideRouteTableIAM",
  "password": "aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxxx",
  "tenant": "ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxxx"
}
```

### Vorlage (Hinzufügen von App-Key, App-ID und Tenant-ID)

```
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-xxxxxxxxxxxxx
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxxx
app-id 576dd4f1-c08d-46b9-cccc-xxxxxxxxxxxxx
tenant-id ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxxx
resource-group CorporateDatacenterResourceGroup
```

6. Konfigurieren der Cloud-Redundanz auf beiden Routern Der einzige Unterschied zwischen der Konfiguration auf beiden Routern sind die bfd-Peers und das Standard-Gateway. CSRA-Konfiguration

```
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-xxxxxxxxxxxxx
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxxx
app-id 576dd4f1-c08d-46b9-cccc-xxxxxxxxxxxxx
tenant-id ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxxx
resource-group CorporateDatacenterResourceGroup
```

### CSRB-Konfiguration

```
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-xxxxxxxxxxxxx
subscription-id 09e13fd4-xxxx-xxxx-xxxx-xxxxxxxxxxxxx
app-id 576dd4f1-c08d-46b9-cccc-xxxxxxxxxxxxx
tenant-id ae49849c-2622-xxxx-xxxx-xxxxxxxxxxxxx
resource-group CorporateDatacenterResourceGroup

```

## Überprüfen der Hochverfügbarkeit

### 1. Überprüfen Sie BFD- und Cloud-Konfigurationen.

```

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-xxxxxxxxxxxxx
tenant id         = ae49849c-2622-4d45-b95e-xxxxxxxxxxxxx
application id     = 1e0f69c3-b6aa-46cf-b5f9-xxxxxxxxxxxxx
application key    = aaafc573-e84e-42ac-b4e3-xxxxxxxxxxxxx
route-table       = InsideRoutetable
cidr               = 8.8.8.8/32
Default Gateway IP = 192.168.1.4

```

### 2. Führen Sie einen Ping und eine Traceroute von der VM zum Ziel aus. Stellen Sie sicher, dass der Ping über die interne eth1-Schnittstelle gesendet wird.

```

$ 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 zeigt, dass der Pfad vom virtuellen System zu 8.8.8.8 über die interne CSRA-Schnittstelle verläuft.

```
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. Fahren Sie die Tunnel-1-Schnittstelle der CSRA herunter, um ein Failover zu simulieren.

```
CSRA#config t
Enter configuration commands, one per line.  End with CNTL/Z.
CSRA(config)#int tunnel1
CSRA(config-if)#sh
```

5. Beachten Sie, dass der Datenverkehr nun über die private CSRB-Schnittstelle fließt.

```
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
```

**Hinweis:** Azure Cloud kann bei einem Failover zu einer Verzögerung führen. Die Verzögerung darf nicht länger als 1 Minute sein.

## Fehlerbehebung

- Aktivieren Sie Debugger, um Meldungen während des HA-Failovers zu beobachten.

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

- Authentifizierungs- und Anmeldefehler sind auf ungültige Zugriffskontrollen zurückzuführen, die es dem CSR1000v ermöglichen, API-Anrufe an die Azure-Routing-Tabelle zu tätigen. Überprüfen Sie, ob die richtigen IDs in Schritt 10 konfiguriert sind.

```
*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"}
```

## Zugehörige Informationen

- [Azure CLI 2.0](#)
- [Cisco CSR 1000v - Implementierungsleitfaden für Microsoft Azure](#)
- [Auswahl des richtigen Werkzeugs für Azure und nebenseitige Azure-CLI- und PowerShell-Befehle](#)