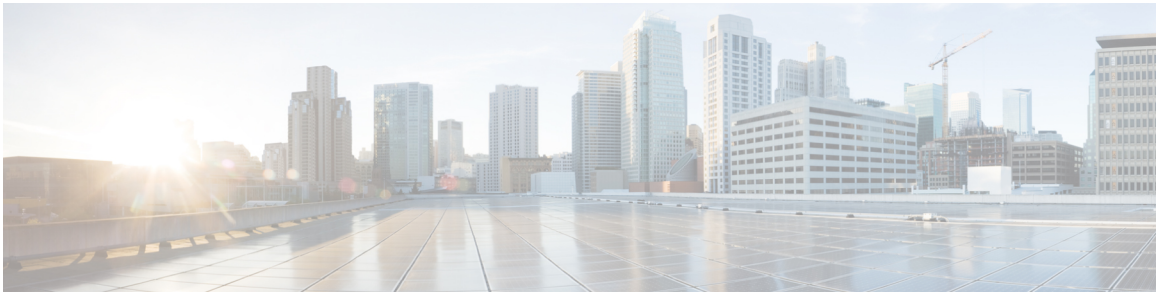
Cisco Catalyst 8000V Edge Software High Availability Configuration Guide

First Published: 2020-12-21

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



CONTENTS

Full Cisco Trademarks with Software License ?

CHAPTER 1 **Preface** 1

- Audience and Scope 1
- Feature Compatibility 1
- Document Conventions 2
- Communications, Services, and Additional Information 3
- Documentation Feedback 4
- Troubleshooting 4

CHAPTER 2 **Overview of High Availability** 5

- Reference the Chapter Map here 8
- Topologies Supported 8
- Redundancy Nodes 8
- Event Types 8

CHAPTER 3 **Configure High Availability** 11

- Configuring IOX and the Guestshell on Cisco IOS XE 11
- Configure a Tunnel Between the Cisco Catalyst 8000V Routers 12
- Configuring EIGRP over Virtual Tunnel Interfaces 13
- Verify the Tunnel Surface 14
- Configure the BFD Peer Router 14
- Install the High Availability Package 15

CHAPTER 4 **Configure High Availability for Cisco Catalyst 8000V Running on Azure** 17

Create Binding to BFD Peer 17
Configure Cloud Specific Redundancy Parameters 18
Create a Redundancy Node 18
Set Redundancy Node Parameters 19
Clear Redundancy Node Parameters 19
Authenticate the Cisco Catalyst 8000V Router 20
System Assigned Managed Identity 20
Authentication Using Azure Active Directory Service Principal 21
Obtain the Application ID and Tenant ID 23
Create an Authentication key for the Application 23
Manage Azure Active Directory Applications in Guestshell 24
Clear the Default Application 25
Clear the Application List 25
Managing all Applications 25
Configuring IAM for the Route Table 26
Route Table Entry Types 27
Configuring the Network Security Group 27

CHAPTER 5 **Configure High Availability on Cisco Catalyst 8000V Running on Amazon Web Services** 29

Create a Redundancy Node 30
Set Redundancy Node Parameters 31
Clear Redundancy Node Parameters 31
Authenticate the Cisco Catalyst 8000V Router 31
Disable Source/Destination Address Checking 32
Route Table Entry Types 32
Configure Security Group 33

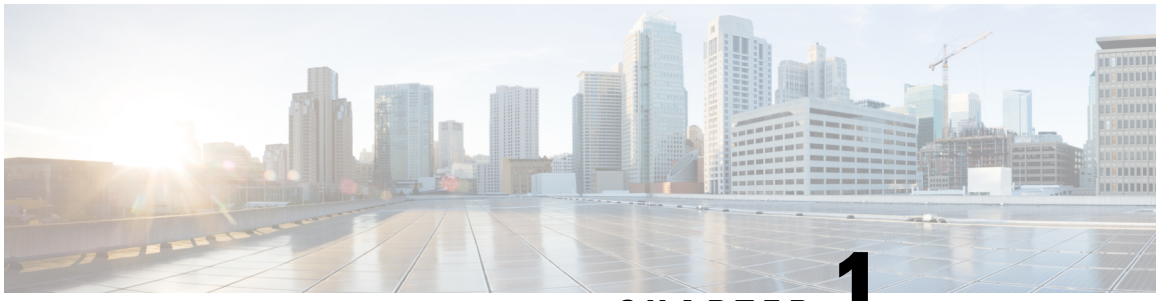
CHAPTER 6 **Configure High Availability in Cisco Catalyst 8000V Running On Google Cloud Platform** 35

Cloud Specific Configuration of Redundancy Parameters 37
Create a Redundancy Node 38
Set Redundancy Node Parameters 39
Authenticate the Cisco Catalyst 8000V Router 39

CHAPTER 7 Example Configurations 41

CHAPTER 8 Verify High Availability 43

CHAPTER 9 Troubleshoot High Availability Issues 45



CHAPTER 1

Preface

Note The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

- Audience and Scope, on page 1
- Feature Compatibility, on page 1
- Document Conventions, on page 2
- Communications, Services, and Additional Information, on page 3
- Documentation Feedback, on page 4
- Troubleshooting, on page 4

Audience and Scope

This document is designed for the person who is responsible for configuring your Cisco Enterprise router. This document is intended primarily for the following audiences:

- Customers with technical networking background and experience.
- System administrators familiar with the fundamentals of router-based internetworking but who might not be familiar with Cisco IOS software.
- System administrators who are responsible for installing and configuring internetworking equipment, and who are familiar with Cisco IOS software.

Feature Compatibility

For more information about the Cisco IOS XE software, including features available on your device as described in the configuration guides, see the respective router documentation set.

To verify support for specific features, use the Cisco Feature Navigator tool. This tool enables you to determine the Cisco IOS XE software images that support a specific software release, feature set, or a platform.

Document Conventions

This documentation uses the following conventions:

Convention	Description
^ or **Ctrl**	The ^ and **Ctrl** symbols represent the Control key. For example, the key combination **^D** or **Ctrl-D** means hold down the **Control** key while you press the **D** key. Keys are indicated in capital letters but are not case sensitive.
string	A string is a nonquoted set of characters shown in italics. For example, when setting an SNMP community string to public, do not use quotation marks around the string or the string will include the quotation marks.

The command syntax descriptions use the following conventions:

Convention	Description
bold	Bold text indicates commands and keywords that you enter exactly as shown.
italics	Italic text indicates arguments for which you supply values.
[x]	Square brackets enclose an optional element (keyword or argument).
|	A vertical line indicates a choice within an optional or required set of keywords or arguments.
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical line indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical line indicate a required choice.

Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. For example, see the following table.

Convention	Description
[x {y | z}]	Braces and a vertical line within square brackets indicate a required choice within an optional element.

Examples use the following conventions:

Convention	Description
`screen`	Examples of information displayed on the screen are set in Courier font.
`bold screen`	Examples of text that you must enter are set in Courier bold font.
< >	Angle brackets enclose text that is not printed to the screen, such as passwords.
!	An exclamation point at the beginning of a line indicates a comment line. Exclamation points are also displayed by the Cisco IOS XE software for certain processes.
[ ]	Square brackets enclose default responses to system prompts.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Note Means *reader take note*. Notes contain helpful suggestions or references to materials that may not be contained in this manual.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

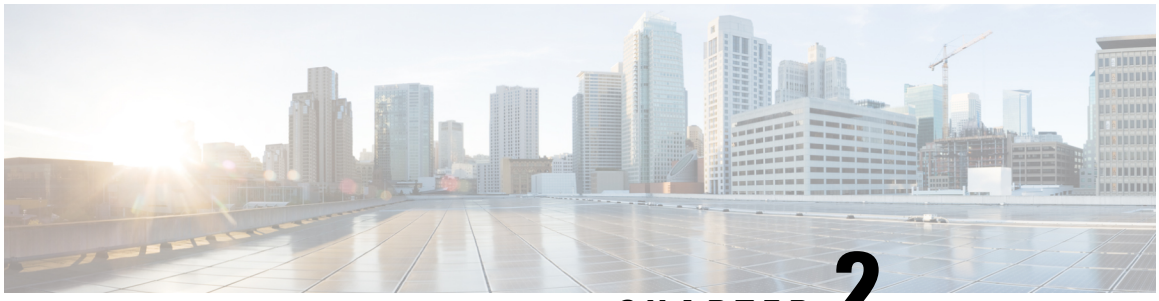
Documentation Feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.

Troubleshooting

For the most up-to-date, detailed troubleshooting information, see the Cisco TAC website at https://www.cisco.com/en/US/support/index.html.

Go to **Products by Category** and choose your product from the list, or enter the name of your product. Look under **Troubleshoot and Alerts** to find information for the issue that you are experiencing.



CHAPTER 2

Overview of High Availability

High Availability refers to the ability to establish redundancy of networking functionality and configuration data between two peer routers. This guide provides information about high availability, and how you can configure high availability on Cisco Catalyst 8000V Edge Software running on different cloud service providers.

The High Availability feature is supported for Cisco Catalyst 8000V Routers running on Microsoft Azure, Google Cloud Platform (GCP), and Amazon Web Services (AWS). A typical use case for the Cisco Catalyst 8000V is to interconnect two subnets within a virtual network. You can deploy Cisco Catalyst 8000V routers between the front-end (public) and the back-end (private) subnets. The Cisco Catalyst 8000V router represents a single point of failure for access to back-end resources. To mitigate this single point of failure, you must deploy two Cisco Catalyst 8000V routers between the two subnets.

The back-end subnet contains a routing table with entries pointing to the next hop router, which is one of the two Cisco Catalyst 8000V instances. The peer Cisco Catalyst 8000V routers communicate with one another over a tunnel using the Bi-directional Forwarding Detection (BFD) protocol. If the connection is lost between a router and a peer, BFD generates an event. This event causes the active router that is working to update the entries in the route table so that the routing table points to the default route.

The routing table controls the upstream traffic of the Cisco Catalyst 8000V router and the routing protocol configured on the router determines the path of the downstream traffic.

In cloud environments, it is common for virtual networks to implement a simplistic mechanism for routing, which is based on a centralized route table. However, you can also create multiple route tables, where each route table has a subnet assigned. This subnet acts as the source of route information, and the route table is populated automatically which includes one or more individual routes depending on the network topology. You can also configure the routes in the route table.

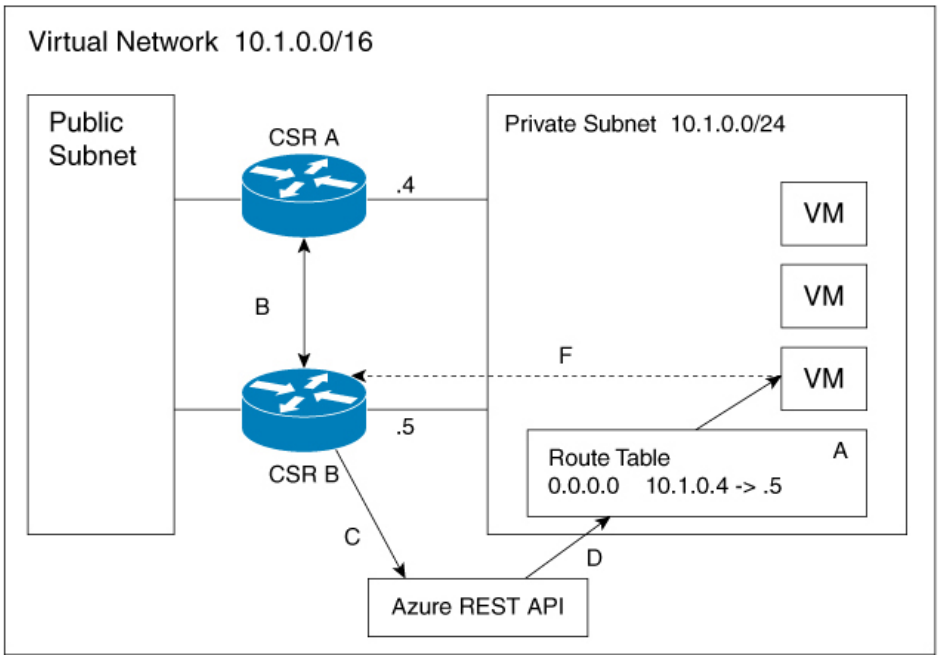
A subnet has a centralized route table, which allows two Cisco Catalyst 8000V routers to operate in a redundant mode. You can deploy two Cisco Catalyst 8000V routers in the same virtual network with their interfaces directly connected to subnets in the virtual network. You can add routes to the route table to point to one of the two redundant Cisco Catalyst 8000V routers. At any given time, one of the two Cisco Catalyst 8000V routers serves as the next-hop router for a subnet. This router is the active router for the subnet. The peer router is referred to as the passive router. The active router is the next hop for a given route destination.

The Cisco Catalyst 8000V router uses the Bi-directional Failure Detection (BFD) protocol to detect whether a peer router is operating properly. An IP tunnel is created between the two peer routers and each router periodically sends a BFD protocol message to the other router. If one router fails to receive a BFD message from the peer for a specific period, the active router concludes that the peer router has failed.

If the active router fails, the route table for the subnet can be dynamically updated to change the next hop address for one or more routes so that they refer to the passive router. If the peer router detects the failure of the active router, the peer router uses the programmatic API to update the route table entries.

For a route table entry, configure which of the two Cisco Catalyst 8000V routers is the “primary” router. The other router is the passive router if it is configured as a “secondary” router. By default, all routes are configured as secondary.

Figure 1: High Availability - Topology



The subnet on the right has an address block of 12.1.0.0/24. The two Cisco Catalyst 8000V routers that are connected to this subnet provide a redundant path for traffic leaving this leaf subnet. The subnet is associated with a route table which provides the route information to the virtual machines attached to the subnet.

Consider this scenario: Initially the default route in the route table has the IP address of the next hop router - 12.1.0.4 (Cisco Catalyst 8000V A). All the traffic leaving the subnet goes through Cisco Catalyst 8000V A. Cisco Catalyst 8000V A is currently the active router for the default route. When Cisco Catalyst 8000V A fails, Cisco Catalyst 8000V B detects the failure as this router stops receiving BFD protocol messages from Cisco Catalyst 8000V A. Cisco Catalyst 8000V B writes to the route table via a RESTAPI to change the default route to the interface of Cisco Catalyst 8000V B on the 12.1.0.0/24 subnet, which is IP address 12.1.0.5. Cisco Catalyst 8000V B then becomes the active router for the route to the 15.0.0.0 network.

Step	Description
A	Cisco Catalyst 8000V A with address 12.1.0.4 is the active router for the 15.0.0.0 network.
B	Cisco Catalyst 8000V A fails. Cisco Catalyst 8000V B detects the failure using the BFD protocol.
C	Cisco Catalyst 8000V B uses an HTTP request to the Azure REST API.
D	Azure updates the 15.0.0.0 route in the user-defined route table to the IP address of Cisco Catalyst 8000V B.

Step	Description
E	Virtual machines see the route table update.
F	Packets from the virtual machines are now directed to Cisco Catalyst 8000V B.

High Availability Features

High Availability version supports several features. Here's an overview of high availability in Cisco Catalyst 8000V.

- **Cloud Agnostic**: This version of high availability is functional on Cisco Catalyst 8000V routers running on any cloud service provider. While there are some differences in the cloud terminology and parameters, the set of functions and scripts used to configure, control, and show the high availability features are common across the different cloud service providers. High Availability is supported in Cisco Catalyst 8000V routers running on AWS, Azure, and GCP. Check with Cisco for current support of high availability in the individual provider's clouds.
- **Active/active operation**: You can configure both Cisco Catalyst 8000V routers to be active simultaneously, which allows for load sharing. In this mode of operation, each route in a route table has one of the two routers serve as the primary router and the other router as the secondary router. To enable load sharing, take all the routes and split them between the two Cisco Catalyst 8000V routers.
- **Reversion to Primary Cisco Catalyst 8000V After Fault Recovery**: You can designate a Cisco Catalyst 8000V as the primary router for a given route. While this Cisco Catalyst 8000V is up and running, it is the next hop for the route. If this Cisco Catalyst 8000V fails, the peer Cisco Catalyst 8000V takes over as the next hop for the route, maintaining network connectivity. When the original router recovers from the failure, it reclaims ownership of the route and is the next hop router.
- **User-supplied Scripts**: The guestshell is a container in which you can deploy your own scripts. HA exposes a programming interface to user-supplied scripts. This implies that you can now write scripts that can trigger both failover and reversion events. You can also develop your own algorithms and triggers to control which Cisco Catalyst 8000V provides the forwarding services for a given route.
- **New Configuration and Deployment Mechanism**: The implementation of HA has been moved out of the Cisco IOS XE code. High availability code now runs in the guestshell container. For further information on guestshell, see the *Guest Shell* section in the Programmability Configuration Guide. The configuration of redundancy nodes is performed in the guestshell using a set of Python scripts.

- Reference the Chapter Map here, on page 8
- Topologies Supported, on page 8
- Redundancy Nodes, on page 8
- Event Types, on page 8

Reference the Chapter Map here

Topologies Supported

1-for-1 redundancy topology: If both the Cisco Catalyst 8000V routers have a direct connection to the same subnet, the routers provide a 1-for-1 redundancy. An example of 1-for-1 redundancy is shown in the preceding figure. All the traffic that is intended for a Cisco Catalyst 8000V only goes to one of the routers - the Cisco Catalyst 8000V that is currently active. The active Cisco Catalyst 8000V router is the next-hop router for a subnet. The other Cisco Catalyst 8000V router is the passive router for all the routes.

Load sharing topology: In this topology, both the Cisco Catalyst 8000V routers have direct connections to different subnets within the same virtual network. Traffic from subnet A goes to router A and traffic from subnet B goes to router B. Each of these subnets is bound to different route tables. If router A fails, the route table for subnet A is updated. Instead of router A being the next hop, the route entry is changed to router B as the next hop. If router B fails, the route table for subnet B is updated. Instead of router B being the next hop, the route entry is changed to router A as the next hop.

Redundancy Nodes

A redundancy node is a set of configuration parameters that specifies an entry in a route table. The next hop of a route is updated when an active router fails. To configure a redundancy node, you require the following information:

- Route Table – The identity of the route table in the cloud. Route table includes a region or group in which the table was created, an identifier for the creator or the owner of the table, and a name or identifier for the specific table. Optionally, you can specify an individual route within the table. If you do not specify an individual route, the redundancy node represents all the routes in the table.
- Credentials - Authentication of the identity of the Cisco Catalyst 8000V router. Each cloud provider handles the process of obtaining and specifying the credentials differently.
- Next Hop - The next hop address that is written to the route entry when a trigger event occurs. Next Hop is usually the interface of the Cisco Catalyst 8000V routers on the subnet that is protected.
- Peer Router - Identifies the redundant router that will forward traffic for this route after a failure occurs on this router.
- Router Role—Identifies whether the redundancy node serves in a primary or secondary role. This is an optional parameter. If you do not specify this value, the router role defaults to a secondary role.

Event Types

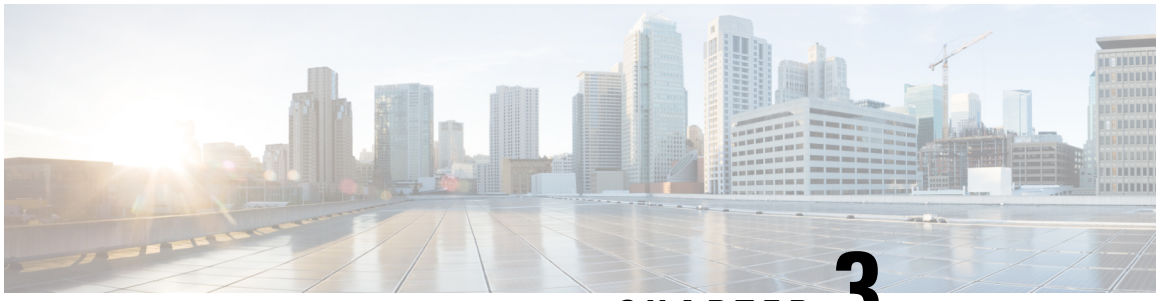
The high availability feature recognizes and responds to three types of events:

- **Peer Router Failure**: When the peer route fails, it is detected as a Peer Router Failure event. In response to this event, the event handler writes the route entry with the next hop address that is defined in the redundancy node. To enable this event to be generated, configure the BFD protocol to a peer router and associate the BFD peer under redundancy for cloud high availability.

- **Revert to Primary Router**: After a router recovers from a failure, the *Revert to Primary Router* event occurs. The purpose of this event is to ensure that the primary router for the route is re-established as the active router. This event is triggered by a timer and you need not configure this event. In the route table entry, the event handler changes the next hop address that is defined in the redundancy node only if it is different from the next hop address that is currently set for the route.

  This Revert to Primary Router event is generated periodically using a CRON job in the guestshell environment. The job is scheduled to run every 5 minutes and checks if each redundancy node that is configured in the primary mode has this router's next hop interface set in the route table. If the route table entry already points to this router's next hop interface, then an update is not required. If a redundancy node configuration of the mode parameter is secondary, then the *Revert to Primary Router* event is ignored.

- **Redundancy Node Verification**: The event handler detects a Redundancy Node Verification event and reads the route entry that is specified by the redundancy node. The event handler writes the same data back to the route entry. This event is not generated automatically or algorithmically. This event verifies the ability of the event handler to execute its functions. Execute a script, manually or programmatically, to trigger the Redundancy Node verification event. For further information about the verification event, see *User-Defined Triggers*, in the *Advanced Programming for High Availability on Microsoft Azure* section.



CHAPTER 3

Configure High Availability

The following sections specify the common configuration steps to configure High Availability for a Cisco Catalyst 8000V running on any cloud service provider.

- Configuring IOX and the Guestshell on Cisco IOS XE, on page 11
- Configure a Tunnel Between the Cisco Catalyst 8000V Routers, on page 12
- Configuring EIGRP over Virtual Tunnel Interfaces, on page 13
- Verify the Tunnel Surface, on page 14
- Configure the BFD Peer Router, on page 14
- Install the High Availability Package, on page 15

Configuring IOX and the Guestshell on Cisco IOS XE

The following Cisco IOS XE configuration shows the commands that are required to access the guestshell. You do not need to configure these prerequisites as they are included automatically in the startup-config file.

SUMMARY STEPS

1. Perform the following configuration:
2. To configure High Availability, you must verify whether IOX is configured and running:
3. Enter the following command to verify that the guest application is defined and running:

DETAILED STEPS

	Command or Action	Purpose

Step 1	Perform the following configuration:
	Example:
	`iox`
	`ip nat inside source list GS_NAT_ACL interface GigabitEthernet1 vrf GS overload ip route vrf GS 0.0.0.0 0.0.0.0 GigabitEthernet1 192.168.35.1 global`
	`interface VirtualPortGroup0 vrf forwarding GS`
	`ip address 192.168.35.101 255.255.255.0`
	`ip nat inside no mop enabled no mop sysid`
	`ip access-list standard GS_NAT_ACL permit 192.168.35.0 0.0.0.255`
	`app-hosting appid guestshell`	

	Command or Action	Purpose
	app-vnic gateway1 virtualportgroup 0 guest-interface 0 guest-ipaddress 192.168.35.102 netmask 255.255.255.0 app-default-gateway 192.168.35.101 guest-interface 0 name-server0 8.8.8.8	
Step 2	To configure High Availability, you must verify whether IOX is configured and running: **Example:** show iox Virtual Service Global State and Virtualization Limits: Infrastructure version : 1.7 Total virtual services installed : 0 Total virtual services activated : 0 Machine types supported : LXC Machine types disabled : KVM Maximum VCPUs per virtual service : 1 Resource virtualization limits: Name Quota Committed Available --- system CPU (%) 75 0 75 memory (MB) 3072 0 3072 bootflash (MB) 20000 0 5745 IOx Infrastructure Summary: --- IOx service (CAF) : Running IOx service (HA) : Not Running IOx service (IOxman) : Running Libvirtd : Running	
Step 3	Enter the following command to verify that the guest application is defined and running: **Example:** show app-hosting list show app-hosting list App id State --- guestshell RUNNING	If the state of the guestshell displays DEPLOYED in the output of the preceding command, you must enable the guestshell by using the following command: guestshell enable Interface will be selected if configured in app-hosting Please wait for completion guestshell activated successfully Current state is: ACTIVATED guestshell started successfully Current state is: RUNNING Guestshell enabled successfully

Configure a Tunnel Between the Cisco Catalyst 8000V Routers

You must configure a tunnel between the Cisco Catalyst 8000V routers and enable Bi-directional Forwarding Detection (BFD) and a routing protocol (EIGRP or BGP) on the tunnel for peer failure detection. To authenticate and encrypt IP traffic as it traverses a network, either use an IPsec tunnel or VxLAN GPE tunnel.

Step 1 To configure an IPsec tunnel, enter the configuration mode commands to give the following configuration. The command crypto isakmp policy 1 defines an IKE policy, with a high priority (1), and enters config-isakmp configuration mode.

Example:

```
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 Tunnel1
ip address 192.168.101.1 255.255.255.252
load-interval 30
tunnel source GigabitEthernet1 tunnel mode ipsec ipv4
tunnel destination 23.96.91.169 tunnel protection ipsec profile vti-1
bfd interval 100 min_rx 100 multiplier 3
```

Step 2 To create a VxLAN GPE tunnel, enter the following configuration

```
interface Tunnel100
ip address 192.168.101.1 255.255.255.0
bfd interval 100 min_rx 100 multiplier 3 tunnel source GigabitEthernet1
tunnel mode vxlan-gpe ipv4 tunnel destination 40.114.93.164
tunnel vxlan vni 10000
```

For further information on configuring a VxLAN GPE tunnel, see the Carrier Ethernet Configuration Guide.

The tunnel destination address must be the public IP address of the corresponding Cisco Catalyst 8000V. For the tunnel IP address, use any unique IP address. However, the tunnel endpoints of each redundantCisco Catalyst 8000V must be in the same subnet.

Note To allow VxLAN to pass traffic through the tunnel, you must ensure that UDP ports 4789 and 4790 are allowed in the cloud's network security group. See the cloud provider's documentation for configuring network security filters.

Configuring EIGRP over Virtual Tunnel Interfaces

Configure EIGRP over the virtual tunnel interfaces using the following steps.

Note Other than using EIGRP, which is the protocol that is used in the following steps, you also have the option of using either BGP, or OSPF.

Before you begin

Configure either a VxLAN or IPsec tunnel between the Cisco Catalyst 8000V routers.

Step 1 **router eigrp** *as-number*

Example:

```
Device(config)# router eigrp 1
```

Enables the EIGRP routing process and enters the router configuration mode.

Step 2 **network** *ip-address subnet-mask*

Share the network of the tunnel using EIGRP.

Example:

```
network 192.168.101.0 0.0.0.255
```

Step 3 **bfd all-interfaces**

Enables BFD globally on all the interfaces that are associated with the EIGRP routing process.

Example:

```
Device(config-router)# bfd all-interfaces
```

Step 4 **end**

Exits the router configuration mode and returns the router to the privileged EXEC mode.

Example:

```
Device(config-router)# end
```

Step 5 **show bfd neighbors**

Verifies that the BFD neighbor is active and displays the routing protocols that BFD has registered.

Example:

```
Device# show bfd neighbors

IPv4 Sessions
NeighAddr      LD/RD        RH/RS    State  Int
192.168.101.2  4097/4097    Up       Up     Tu100
```

Verify the Tunnel Surface

To verify that the tunnel interface is configured and enabled, run the `show ip interface brief` command.

Example:

```
# show ip interface brief
 IP-Address OK? Method Status Protocol
GigabitEthernet1 192.168.35.20 YES DHCP up up
GigabitEthernet2 192.168.36.12 YES DHCP up up
Tunnel1                172.17.1.1         YES NVRAM      up up
VirtualPortGroup0 192.168.35.101     YES NVRAM      up up
```

Configure the BFD Peer Router

Run the following command:

Example:

```
redundancy
cloud-ha bfd peer <peer_router_ip_address>
```

This configuration command identifies the peer router. The IP address is that of the peer Cisco Catalyst 8000V within the tunnel carrying the BFD protocol between the two Cisco Catalyst 8000V routers.

Install the High Availability Package

Step 1 Execute the `#Router> guestshell` command to enter the guestshell.

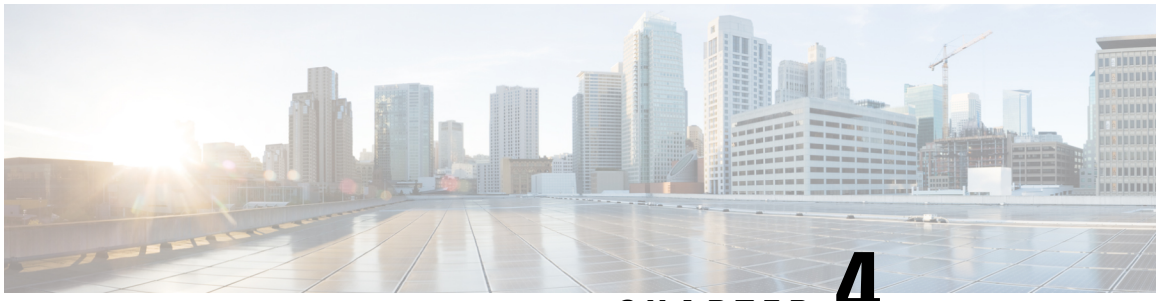
Step 2 Install the appropriate Python package based on the cloud provider on which the Cisco Catalyst 8000V instance is running:

Cloud Provider	Package Name
Microsoft Azure	csr_azure_ha
Amazon Web Services	csr_aws_ha
Google Cloud Platform	csr_gcp_ha

Note The package name for Microsoft Azure is the same for both HAv2 and HAv3. If you perform an install by executing the `pip install csr_azure_ha --user` command, the latest HA V3 is downloaded.

Step 3 Install the package that is appropriate for your cloud service provider by using the `[guestshell@guestshell]$ pip install <package_name> --user` command.

Step 4 From the home directory, navigate to the subdirectory named `cloud:[guestshell@guestshell]$ cd cloud`.



CHAPTER 4

Configure High Availability for Cisco Catalyst 8000V Running on Azure

High Availability is supported for Cisco Catalyst 8000V on Cisco IOS XE 17.4 release and later.

- Create Binding to BFD Peer, on page 17
- Configure Cloud Specific Redundancy Parameters, on page 18
- Create a Redundancy Node, on page 18
- Set Redundancy Node Parameters, on page 19
- Clear Redundancy Node Parameters, on page 19
- Authenticate the Cisco Catalyst 8000V Router, on page 20
- System Assigned Managed Identity, on page 20
- Authentication Using Azure Active Directory Service Principal, on page 21
- Obtain the Application ID and Tenant ID, on page 23
- Create an Authentication key for the Application, on page 23
- Manage Azure Active Directory Applications in Guestshell, on page 24
- Clear the Default Application, on page 25
- Clear the Application List, on page 25
- Managing all Applications, on page 25
- Configuring IAM for the Route Table, on page 26
- Route Table Entry Types, on page 27
- Configuring the Network Security Group, on page 27

Create Binding to BFD Peer

When you configure High Availability with IOS XE releases 17.4 and later, you can create a binding to a BFD peer by executing the following command:

Example:

```
redundancy
cloud-ha bfd peer <peerIpAddress>
```

Configure Cloud Specific Redundancy Parameters

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

Parameter Switch	Switch	Description
Node Index	-i	The index that is used to uniquely identify this node. Valid values: 1–255.
Cloud Provider	-p	Specifies the type of Azure cloud: azure, azusgov, or azchina.
Subscription ID	-s	The Azure subscription id.
Resource Group Name	-g	The name of the route table to be updated.
Route Table Name	-t	The name of the route table to be updated.
Route	-r	IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table of type “virtual appliance”.
Next Hop Address	-n	The IP address of the next hop router. Use the IP address that is assigned to this Cisco Catalyst 8000V on the subnet which utilizes this route table. Can be an IPv4 or IPv6 address.
Mode	-m	Indicates whether this router is the primary or secondary router for servicing this route. Default value is secondary.

Create a Redundancy Node

Run the following script to create a redundancy node and add it to the database: `create_node { switch value } […[{ switch value }].`

You must configure the following parameters for a valid redundancy node:

- Node Index

- Cloud Provider
- Subscription ID
- Resource Group Name
- Route Table Name

```
create_node -i 10 -p azure -s b0b1a9e2-4444-4ca5-acd9-bebd1e6873eb -g ds-rg -t ds-sub2-RouteTable -r
 15.0.0.0/8 -n 192.168.7.4
```

If the configuration is successful, the script returns a value of zero.

Set Redundancy Node Parameters

Procedure

	Command or Action	Purpose
Step 1	To change the value of parameters in an existing redundancy node, run the following script: `set_params{ switch value } […[{ switch value }].` **Example:** `set_params.py -i 10 -r 15.0.0.0/16 -n 192.168.7.5`	The index parameter (-i) is mandatory. This command sets the values of the specified parameters. If the specified parameter is already defined for the redundancy node, the value of the parameter is updated. `set_params -i 10 -n 192.168.7.5 -m primary` In this example, the next hop address and mode will be updated for the redundancy node with index 10. If this configuration is successful, the script returns a value of zero.

Clear Redundancy Node Parameters

If you want to clear the value of specified parameters for an existing redundancy node, run the following script:
`clear_params -i value { switch } […[{ switch }].`

Example:

```
clear_params -i 10 -r -n
```

In this example, the clear_params script clears both the route and next hop address parameters.

Specify only the switch parameter when you clear an associated value. Do not include the current value of the parameter.

Note Only the index parameter is required. The values of any additional specified parameters are cleared.

If the clearing is successful, the script returns a value of zero.

Authenticate the Cisco Catalyst 8000V Router

To update a routing table in the Azure network, you must first authenticate the Cisco Catalyst 8000V router. This is accomplished by creating an application which represents the Cisco Catalyst 8000V router in the Azure Active Directory. You can use the application that is granted permissions, to access the Azure network resources.

You can create the application by using the following two mechanisms:

- System-assigned managed identity - Azure automatically creates an application and binds it to the router. This mechanism was previously called as Managed Service Identity by Azure.
- Manual application registration in Azure Active Directory - Here, the user creates an application in the Azure Active Directory, which represents the Cisco Catalyst 8000V router.

You can manually create a managed identity in Azure Active Directory by creating an application which represents the router. The application is assigned a set of identifiers; tenant ID, application ID, and application key. These application identifiers must be configured in the high availability feature either as the default AAD application or within an individual redundancy node.

Alternatively, when you create the Cisco Catalyst 8000V, you can configure Azure to create a system-assigned managed identity for the Cisco Catalyst 8000V instance. In this case, you need not configure any application identifiers in the high availability feature. That is, in the absence of the configuration of an application's tenant ID, application ID, and application key, the high availability feature assumes that the Cisco Catalyst 8000V router is using a system-assigned managed identity.

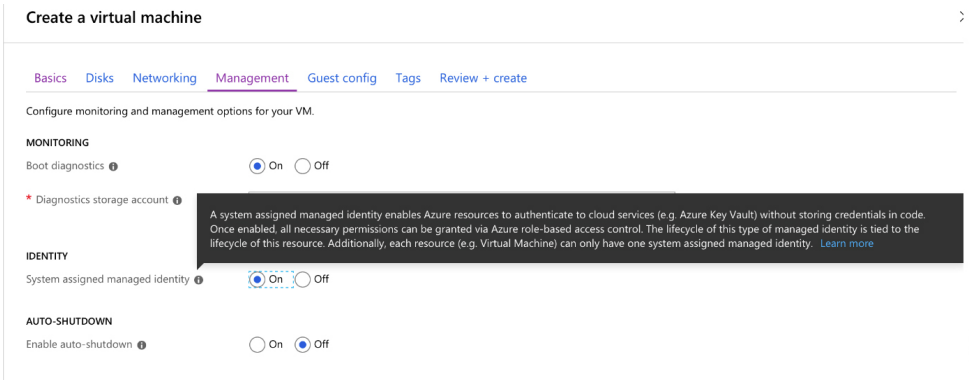
System Assigned Managed Identity

When you create the Cisco Catalyst 8000V router, you can enable for it to be assigned a system managed identity by Azure. There are two ways in which you can create a Cisco Catalyst 8000V router from the Azure marketplace:

- Solution template – A Cisco Catalyst 8000V router is created along with other Azure resources to create a networking solution in a single step.
- Standalone – A standalone Cisco Catalyst 8000V is created, usually within an existing virtual network, with the base Cisco Catalyst 8000V image.

If you create a Cisco Catalyst 8000V router by using one of the solution template offerings in the Azure marketplace, a system-assigned managed identify for the Cisco Catalyst 8000V is enabled by default. If you create a standalone Cisco Catalyst 8000V by using a base Cisco Catalyst 8000V image, a system-managed identity is enabled as shown in the following image:

Figure 2: Enable System Managed Identity



Authentication Using Azure Active Directory Service Principal

This section explains how to create an application in a Microsoft Azure Active Directory with permissions to access Microsoft Azure Resource Manager APIs.

SUMMARY STEPS

1. See the latest instructions on registering an application with Azure Active Directory in Microsoft Azure documentation. See also: https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-v1-add-azure-ad-app.
2. Go to the portal for Microsoft Azure by visiting https://portal.azure.com.
3. Choose your account name and sign in using your Microsoft Azure password.
4. In the left navigation, click **Azure Active Directory** and select an Active Directory in the main pane. Click **Switch Directory** at the top of the pane to select the active directory.
5. Verify whether you are authorized to create a new application. See the following Microsoft Azure documentation for creating an application in the Azure Active Directory: Use portal to create an Azure Active Directory application and service principal that can access resources.
6. Navigate to the Active Directory that you want to use.
7. To create a new application, select **Create** > **New Application Registration**.
8. Specify the name of the application and ensure that **Web App / API** is selected as the Application type
9. Specify the Sign-on URL. Use a name for the sign-on URL which is in the URI format, but it does not have to be reachable. You can use a string in the following format: `http://<your_directory_domain_name>/<app_name>`. For example, if your application name is myapp, and the domain name of your directory is `\mydir.onmicrosoft.com`, use the following is the sign-on URL: `http://mydir.onmicrosoft.com/myapp`.
10. Click **Create**.
11. Navigate to the Azure Active Directory page. Search for the application that you created. Make a note of the assigned Application ID.

DETAILED STEPS

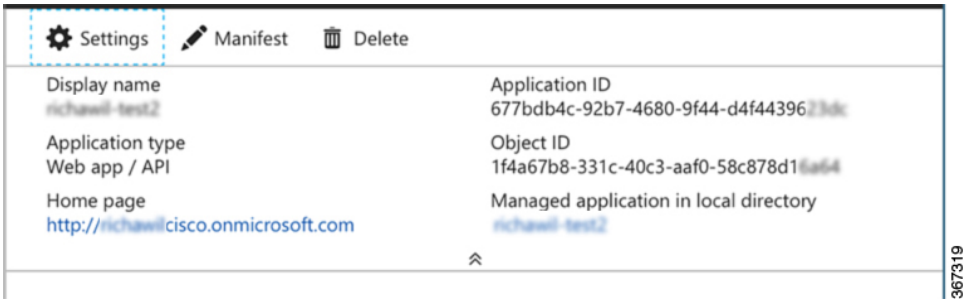
	Command or Action	Purpose
Step 1	See the latest instructions on registering an application with Azure Active Directory in Microsoft Azure documentation. See also: https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-v1-add-azure-ad-app.	
Step 2	Go to the portal for Microsoft Azure by visiting https://portal.azure.com.	
Step 3	Choose your account name and sign in using your Microsoft Azure password.	
Step 4	In the left navigation, click **Azure Active Directory** and select an Active Directory in the main pane. Click **Switch Directory** at the top of the pane to select the active directory.	
Step 5	Verify whether you are authorized to create a new application. See the following Microsoft Azure documentation for creating an application in the Azure Active Directory: Use portal to create an Azure Active Directory application and service principal that can access resources.	
Step 6	Navigate to the Active Directory that you want to use.	
Step 7	To create a new application, select **Create** > **New Application Registration**.	
Step 8	Specify the name of the application and ensure that **Web App / API** is selected as the Application type	
Step 9	Specify the Sign-on URL. Use a name for the sign-on URL which is in the URI format, but it does not have to be reachable. You can use a string in the following format: `http://<your_directory_domain_name>/<app_name>`. For example, if your application name is myapp, and the domain name of your directory is `\mydir.onmicrosoft.com`, use the following is the sign-on URL: `http://mydir.onmicrosoft.com/myapp`.	
Step 10	Click **Create**.	
Step 11	Navigate to the Azure Active Directory page. Search for the application that you created. Make a note of the assigned Application ID.	

Obtain the Application ID and Tenant ID

Before you begin

Create an application in the Microsoft Azure Active Directory.

Step 1 After you create the application, the registered app should appear on the screen as shown in the following image:



Step 2 Use the portal to create an Azure Active Directory application and service principal that can access resources. Make a note of the Application ID. See step 2 in the *Get application ID and authentication key* section in the Microsoft Documentation.

Step 3 Select **Azure Active Directory**.

Step 4 Select **Properties**. Make a note of the value in the **Directory ID** field. This is your tenant ID.

Create an Authentication key for the Application

Step 1 From the Microsoft Azure portal, select the **Azure Active Directory**.

Step 2 Select **App Registrations**.

Step 3 Select the application that you previously created in the *Obtain the Application ID and Tenant ID* section.

Step 4 Click **Settings**.

Step 5 To create a key for API access, select **Keys** and specify a value for **Duration**. Duration is the length of time after which the key becomes invalid.

Step 6 Make a note of the API key from the **Value** field.

Caution Store the API key carefully as it cannot be retrieved later.

Step 7 You must convert the API key to URL unencoded format. To find a suitable conversion tool, enter URL encoder into an Internet search engine. You might need the unencoded API key for procedures such as *Configure Failure Detection for the Cisco Catalyst 8000V on Microsoft Azure*.

Example:

```
URL encoded API Key:   5yOhH593dtD%2FO8gzAlWgulrkWz5dH02d2STk3LdbI4c%3D
        URL unencoded API Key:  5yOhH593dtD/O8gzAlWgulrkWz5dH02d2STk3LdbI4c=
```

Manage Azure Active Directory Applications in Guestshell

There are a set of utility scripts that can be run in the guestshell environment to manage applications in the Azure Active Directory, whether they were created manually as user-assigned identities or system-assigned identities. The following sections describe the use of these scripts and how to configure the binding between a redundancy node and the application used to authenticate the Cisco Catalyst 8000V router.

- Managing user-defined applications: If you have chosen to use a user-assigned identity for the Cisco Catalyst 8000V router, the application that was created in Azure Active Directory must be configured in the high availability feature. The application can be configured as the default application used for all the redundancy nodes, or for individual redundancy nodes.
- Set the default application: If you configure a user-assigned application as the default application using the `set_default_aad_app` script, all the redundancy nodes use the specified application for authentication, unless a redundancy node has an individual application configured.

Set the Default Application

Set the default application by running the `set_default_aad_app.py{ switch value } […[{ switch value }]` script. See the following table for the AAD Redundancy Node Parameters:

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

```
[guestshell@guestshell]$ set_default_aad_app.py -p azure -d
c4426c0b-036f-4bfb-b2d4-5c910c5389d6 -a 3d6e2ef4-8160-4092-911d-53c8f68ba808 -k
hZFvMGfzJuwFiukez27e/duyztom1bj7QL0Yix+KY9c=
```

```
[guestshell@guestshell]$ set_default_aad_app.py -h
usage: set_default_aad_app.py [-h] -p {azure,azusgov,azchina} -a A -d D -k K
AAD Application
required arguments:
  -p {azure,azusgov,azchina} <cloud_provider> {azure | azusgov | azchina}
  -a A                    to add the applicationId
  -d D                    to add the tenantId
  -k K                    to add the applicationKey
```

Clear the Default Application

You can clear the default user-assigned application configuration by using the `clear_default_aad_app` script.

```
[guestshell@guestshell]$ clear_default_aad_app.py
```

Clear the Application List

If you create a user-assigned application and associate the application with individual redundancy nodes, information about these applications is cached in memory. You can display the list of known applications by using the `show_auth_applications.py` script. Clear the cache using the clear_aad_application_list script.

```
[guestshell@guestshell]$ clear_aad_application_list.py
```

Managing all Applications

Use the following scripts to manage all the applications - user-assigned or system-assigned.

Showing Authentication Applications

Cisco Catalyst 8000V router maintains a list of configured applications. You can view this list by using the show_auth_applications script.

```
[guestshell@guestshell]$ show_auth_applications.py
```

Clearing the Authentication Token

When an event is triggered on a redundancy node, the Cisco Catalyst 8000V router uses the configured application to obtain an authentication token from the Azure network. This token is cached up to five minutes in the router. You can clear the cached token by using the `clear_token` script.

This script clears either the default user-assigned application or the system-assigned application. The script does not clear the token on any user assigned application which is explicitly configured on an individual redundancy node.

```
[guestshell@guestshell]$ clear_token.py
```

Refreshing the Authentication Token

The Cisco Catalyst 8000V router can be forced to obtain a new token for the active application by using the `refresh_token` script.

This script refreshes either the default user-assigned application or the system-assigned application. This script does not refresh the token on any user-assigned application which is explicitly configured on an individual redundancy node.

```
[guestshell@guestshell]$ refresh_token.py
```

Select the Authentication Application

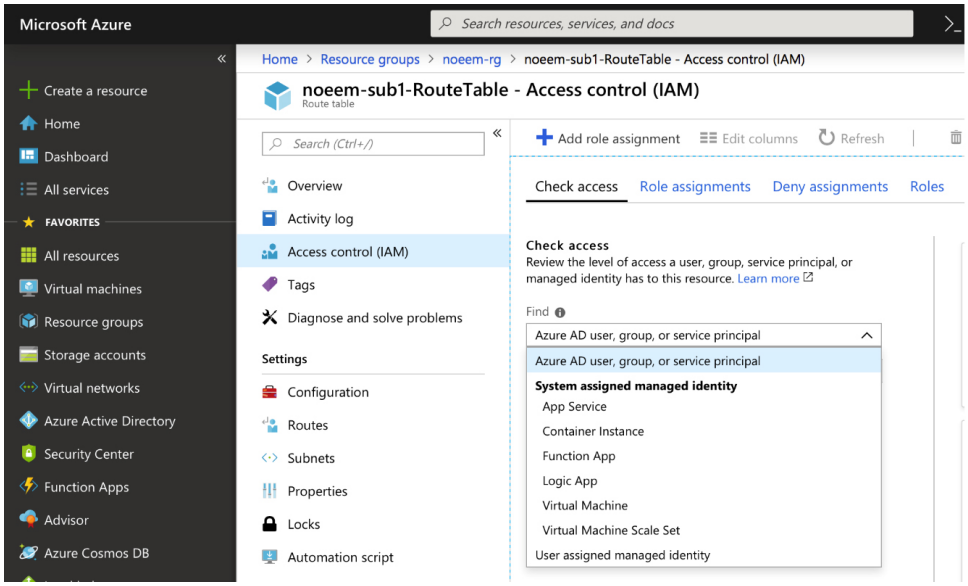
You can choose either system-assigned or user-assigned applications to identify a Cisco Catalyst 8000V router for the purpose of authentication. You can use the same mechanism for all the applications within a single Cisco Catalyst 8000V router. You can also have multiple user-assigned applications across multiple redundancy nodes.

The following table summarizes which application is used by the Cisco Catalyst 8000V router when processing a redundancy node:

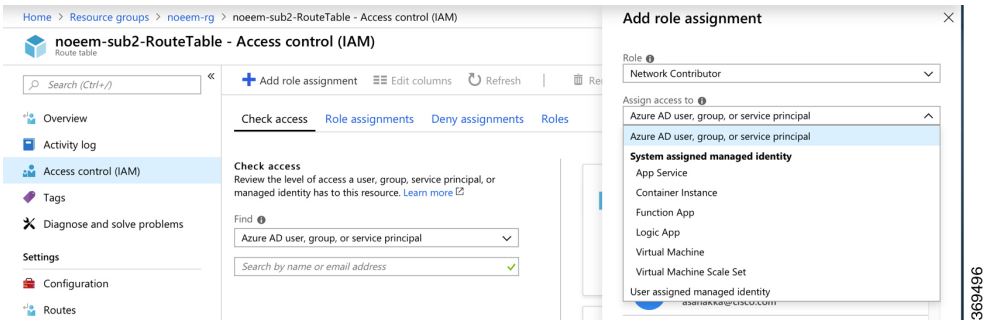
Is A Default Application Configured?	Does Node Have a User Assigned Application Configured?	Will Cisco Catalyst 8000V Use This Application?
No	No	System assigned application
No	Yes	User assigned application configured on this redundancy node
Yes	No	User assigned application configured as the default by set_default_aad_app.py
Yes	No	User assigned application configured on this redundancy node

Configuring IAM for the Route Table

Step 1 To add an application into an existing network, in the **All Resources** pane, choose a private side subnet from the left pane. For example, *noeem-sub1-RouteTable*.



Step 2 In the center pane, select **Access control (IAM)**. Select the plus icon to add a role assignment.



Step 3 In the Add Role Assignment screen, set the **Role to Network Contributor**.

Step 4 Select the **Assign Access to Pulldown** menu. If you are using system-assigned managed identity, select the **Virtual Machine** sub option and go to Step 6. If you are using user-assigned managed identity, select the option and go to step 5.

Step 5 In the **Select** field, enter the name of the user-assigned application that you created in **Azure Active Directory**. Click **Save**.

Step 6 In the **Select** field, enter the name given to the Cisco Catalyst 8000V instance. If you have configured the Cisco Catalyst 8000V instance properly for system-assigned identity, the Cisco Catalyst 8000V instance appears in the search results.

Step 7 Select the Cisco Catalyst 8000V instance by name, and click **Save**.

Route Table Entry Types

The route tables in Microsoft Azure support different entry types. The entry type for a route can be one of the following: Virtual network gateway, Internet, or Virtual Appliance. The next hop address identifies a resource in the Azure network.

Routes with an entry type of Virtual network gateway or Internet do not have an explicit IP address for the next hop and are not supported by the High Availability feature.

When you configure High Availability on a Cisco Catalyst 8000V instance, you can specify individual routes to be updated in the case of failure. Ensure that you configure each individual route as having an entry type of Virtual Appliance. If you configure a redundancy node that represents all the entries in the route table, ensure that all the routes have an entry type of Virtual Appliance.

Configuring the Network Security Group

If you have a network security group attached to NIC0 of the router, you must allow the BFD protocol to pass the interface. Configure an inbound and outbound security rule that allows ports 4789 and 4790 to be passed.

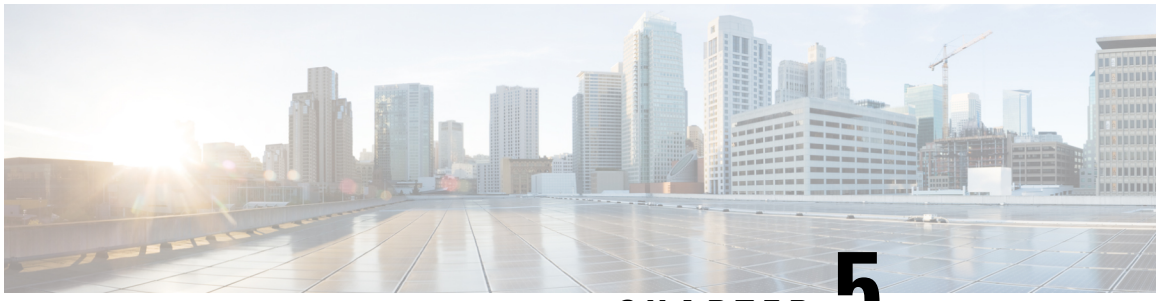
Configuring the Console Timeout

When you start an SSH session to the Cisco Catalyst 8000V router, ensure that you do not configure the terminal VTY timeout as infinite. That is, do not configure: exec-timeout 0 0. Use a non-zero value for the timeout; for example, exec-timeout 4 0. This command specifies a timeout of four minutes and zero seconds. The exec-timeout 0 0 command causes an issue as Azure enforces a timeout for the console idle period of 4 to 30 minutes. When the idle timer expires, Azure disconnects the SSH session. However, the session is not

cleared from the point of view of the Cisco Catalyst 8000V as the timeout was set to infinite (by the exec-timeout 0 0 configuration command). The disconnection causes a terminal session to be orphaned. The session in the Cisco Catalyst 8000V remains open indefinitely. If you try to establish a new SSH session, a new virtual terminal session is used. If this pattern continues, the maximum number of simultaneous terminal sessions allowed is reached and no new sessions can be established. In addition to configuring the exec-timeout command correctly, it is also a good practice to delete idle virtual terminal sessions using the commands that are shown in the following example:

```
RouterA# show users
Line User Host(s) Idle Location
2 vty 0 cisco idle 00:07:40 128.107.241.177
* 3 vty 1 cisco idle 00:00:00 128.107.241.177
RouterA# clear line 2
```

Note If the workaround in the preceding scenarios are ineffective, as a last resort, you can restart the Cisco Catalyst 8000V router in the Azure portal.



CHAPTER 5

Configure High Availability on Cisco Catalyst 8000V Running on Amazon Web Services

Table 1: Cloud Specific Configuration of Redundancy Parameters

Parameter	Switch	Description
Node Index	-i	Index that is used to uniquely identify this node. Valid values: 1–1023.
Region Name	-rg	Name of the region that contains the route table. For example, us-west-2.
Route Table Name	-t	Name of the route table to be updated. The name of the route table must begin with the substring rtb-. For example, rtb-001333c29ef2aec5f
Route	-r	If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table. The Cisco Catalyst 8000V instance cannot change routes which are of type local or gateway.
Next Hop Interface	-n	Name of the interface to which packets should be forwarded in order to reach the destination route. The name of the interface must begin with the substring eni-. For example, eni-07160c7e740ac8ef4.

Parameter	Switch	Description
Mode	-m	Indicates whether this router is the primary or secondary router for servicing this route. Valid values are primary or secondary. This is an optional parameter. The default value is secondary.

- Create a Redundancy Node, on page 30
- Set Redundancy Node Parameters, on page 31
- Clear Redundancy Node Parameters, on page 31
- Authenticate the Cisco Catalyst 8000V Router, on page 31
- Disable Source/Destination Address Checking, on page 32
- Route Table Entry Types, on page 32
- Configure Security Group, on page 33

Create a Redundancy Node

SUMMARY STEPS

1. Run the following script to create a redundancy node and add it to the database.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Run the following script to create a redundancy node and add it to the database. **Example:** `create_node { switch value } […[{ switch value }]`	A valid redundancy node must have the following parameters configured: • Node Index • Region Name • Route Table Name • Next Hop Interface Name For example, `create_node.py -i 2 -t rtb-001333c29ef2aec5e -rg us-west-2 -n eni-07160c7e740ac8ef3 -r 2600:1f14:49b:9b03::/64` If successful, the script returns a value of zero.

Set Redundancy Node Parameters

To change the value of parameters in an existing redundancy node, run the following script: `set_params -i node_index { switch value } […[{ switch value }].`

Example:

```
set_params.py -i 10 -r 15.0.0.0/16 -m primary
```

The index parameter (-i) is mandatory. This command sets the values of the specified parameters. If the specified parameter is already defined for the redundancy node, the value of the parameter is updated.

If this configuration is successful, the script returns a value of zero.

Clear Redundancy Node Parameters

If you want to clear the value of specified parameters for an existing redundancy node, run the following script: `clear_params -i node_index {switch … switch].`

Example:

```
clear_params -i 10 -r -n
```

In this example, the clear_params script clears both the route and next hop address parameters.

Specify only the switch parameter when you clear an associated value. Do not provide the existing values for the parameters to be cleared.

If the clearing is successful, the script returns a value of zero.

Authenticate the Cisco Catalyst 8000V Router

If you want the Cisco Catalyst 8000V router to update a routing table in the AWS network, you must first authenticate the router. In AWS, you must create a policy that permits the Cisco Catalyst 8000V router to access the route table. For example:

```
{
"Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "cloudwatch:",
                "s3:",
                "ec2:AssociateRouteTable",
                "ec2:CreateRoute",
```

```
                "ec2:CreateRouteTable",
                "ec2:DeleteRoute",
                "ec2:DeleteRouteTable",
                "ec2:DescribeRouteTables",
                "ec2:DescribeVpcs",
                "ec2:ReplaceRoute",
                "ec2:DescribeRegions",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DisassociateRouteTable",
                "ec2:ReplaceRouteTableAssociation",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
```

An IAM role is then created using this policy and applied to the EC2 resource.

After the Cisco Catalyst 8000V EC2 instances are created, the IAM role created above needs to be attached to each router.

Note See the AWS documentation for instructions on how to create policies, IAM roles, and how to associate a role to an EC2 instance.

Disable Source/Destination Address Checking

By default, network interfaces created in AWS have source and destination address checking enabled. The interface verifies all the traffic that passes through matches the source or destination address of the interface, otherwise it is dropped. For the Cisco Catalyst 8000V to perform routing, this setting must be disabled on each Cisco Catalyst 8000V interface.

Note See the AWS documentation for instructions on how to disable source/destination address checking on a network interface

Route Table Entry Types

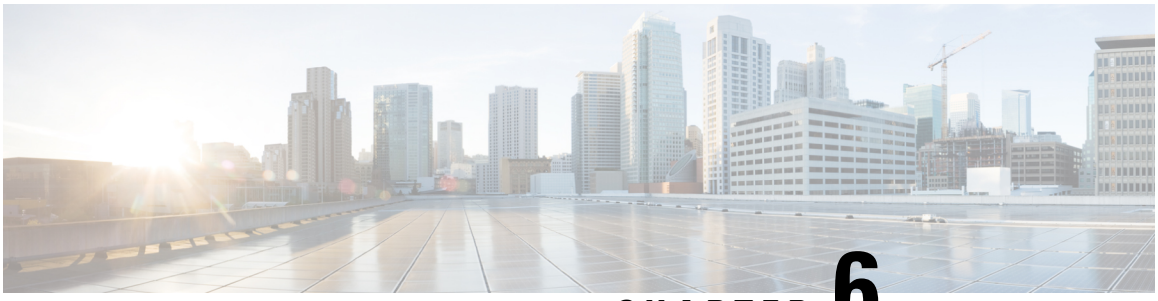
The route tables in AWS cloud support different target types. These route targets include multiple types of gateways and connections. The Cisco Catalyst 8000V router is only capable of updating routes with a network interface target. Routes with other target types are ignored for the purposes of high availability.

If you configure a redundancy node without a specific route destination, the Cisco Catalyst 8000V attempts to update all the routes within a route table with a target type of network interface. All the other routes are ignored.

Configure Security Group

If you have a security group in use by the eth0 interface of the EC2 instance of the Cisco Catalyst 8000V, you must allow the BFD protocol to pass through the interface. Configure an inbound and outbound security rule that allows ports 4789 and 4790 to be passed.

Note See the AWS documentation for instructions on configuring security groups and attaching them to subnets and network interfaces.



CHAPTER 6

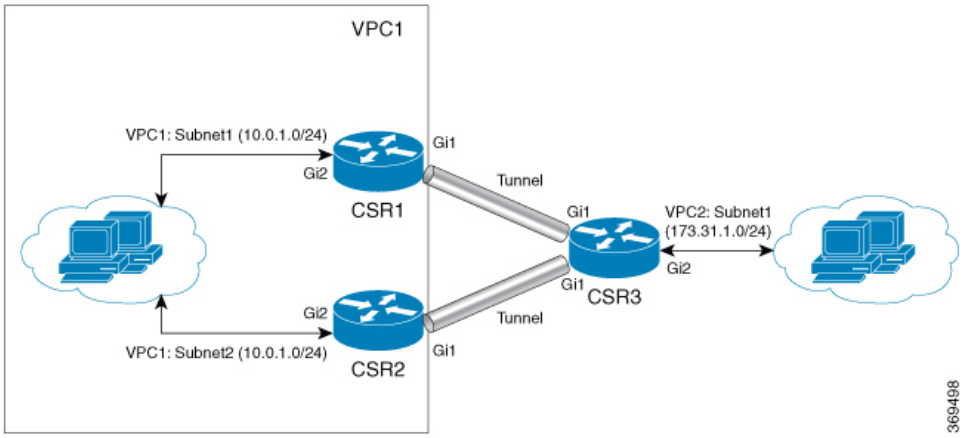
Configure High Availability in Cisco Catalyst 8000V Running On Google Cloud Platform

In the Google cloud, each static route belongs to the route table associated with a VPC and consists of following fields:

- Name and Description: These fields identify the route. A name is required, but a description is optional. Every route in your project must have a unique name.
- Network: Each route must be associated with exactly one VPC network.
- Destination range: The destination range is a single IPv4 CIDR block containing the IP addresses of systems that receive incoming packets. GCP does not support IPv6 destination ranges. Destinations must be expressed in CIDR notation, and the broadest destination possible is 0.0.0.0/0.
- Priority: Priority is used to determine which route should be used if multiple routes have identical destinations. Lower numbers indicate higher priorities; for example, a route with a priority value of 100 has a higher priority than one with a priority value of 200.
- Next hop: Static routes can have next hops that point to the default Internet gateway, a GCP instance, or a Cloud VPN tunnel. Refer to static route next hops for more information.
- Tags: You can specify a list of network tags so that the route will only apply to instances that have at least one of the listed tags. If you don't specify tags, GCP applies the route to all instances in the network.

For more information, see https://cloud.google.com/vpc/docs/routes. To configure High Availability in an active/active operation for two Cisco Catalyst 8000V routers in the Google network, you must create two routes in the route collection for each destination range, where each route points to one of the two routers as the next hop.

To understand this better, consider the following topology:



In the above topology, there are two routers configured in the HA mode. Both the routers have one interface in VPC1 and another in VPC. These two Cisco Catalyst 8000V routers have a Tunnel configured to another Cisco Catalyst 8000V instance that has an interface in VPC2. In this scenario, the following are the route entries in VPC1 for destination range of VPC 2 (172.31.0.0/16):

route-vcp2-c8000v1	172.31.0.0/16	100	None	IP:10:1:0:3	test-vpc
route-vcp2-c8000v2	172.31.0.0/16	200	None	IP:10.0.2.3	test-vpc

The active route is decided based on the route priority. Since route-vpc2-c8000v1 has a lower value, this route has a higher priority, thereby making Cisco Catalyst 8000V 1 as the active route.

Reversion to Primary Cisco Catalyst 8000V After Fault Recovery

If Cisco Catalyst 8000V 1 fails, Cisco Catalyst 8000V 2 detects a peer fail event through the BFD tunnel and deletes route-vpc2-c8000v1 from route collection making route-vpc2-c8000v2 as the active route for destination range 172.31.0.0/16.

When Cisco Catalyst 8000V 1 recovers, it adds route-vpc2-c8000v1 route back to the route collection which makes it the primary route again for all traffic to VPC 2. Please note it is possible to set equal route priority for both route entries in which case Google cloud uses both routes to send traffic to destination range.

On each Cisco Catalyst 8000V instance, you must create nodes corresponding to each route entry in route collection with next hop as the two Cisco Catalyst 8000V instances.

When you use the mode (primary or secondary) option in HA to create a new node, ensure that the route with the higher priority (lower number) is marked as primary and the route with lower priority is marked as secondary.

User-Supplied Scripts

The guestshell is a container in which you can deploy your own scripts. High Availability exposes a programming interface to user-supplied scripts, so you can write scripts that can trigger both failover and reversion events. You can develop your own algorithms and triggers to control which Cisco Catalyst 8000V provides the forwarding services for a given route.

- Cloud Specific Configuration of Redundancy Parameters, on page 37
- Create a Redundancy Node, on page 38
- Set Redundancy Node Parameters, on page 39

• Authenticate the Cisco Catalyst 8000V Router, on page 39

Cloud Specific Configuration of Redundancy Parameters

Parameter	Is this parameter required?	Switch	Description
Node Index	Yes	-i	The index that is used to uniquely identify this node. Valid values: 1–255.
Cloud Provider	Yes	-p	Specify gcp for this parameter.
Project	Yes	-g	Specify the Google Project ID.
routeName	Yes	-a	The route name for which this Cisco Catalyst 8000V is next hop. For example from Fig. 2, if we are configuring node on Cisco Catalyst 8000V 1, this would be route-vpc2-c8000v1.
peerRouteName	Yes	-b	The route name for which the BFD peer Cisco Catalyst 8000V is next hop. For example from Fig. 2, if we are configuring node on Cisco Catalyst 8000V 1, this would be route-vpc2-c8000v2.
Route	yes	-r	The IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table of type virtual appliance. Note: Currently Google cloud does not have IPv6 support in VPC.

Parameter	Is this parameter required?	Switch	Description
Next hop address	Yes	-n	The IP address of the next hop router. Use the IP address that is assigned to this Cisco Catalyst 8000V on the subnet which utilizes this route table. The value can be an IPv4 or IPv6 address. Note: Currently Google cloud does not have IPv6 support in VPC.
hopPriority	Yes	-o	The route priority for the route for which the current Cisco Catalyst 8000V is the next hop.
VPC	Yes	-v	The VPC network name where the route with the current Cisco Catalyst 8000V as the next hop exists.

Create a Redundancy Node

Run the following script to create a redundancy node and add it to the database: `create_node { switch value } […[{ switch value }]`.

You must configure the following parameters for a valid redundancy node:

- Node Index
- Cloud Provider
- Project ID
- Route Name
- Peer Route Name
- Route
- Next Hop Address
- Hop Priority
- VPC Name

```
create_node -i 1 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr -a route-name1 -b route-name2
 -p gcp -v vpc_name
```

If the configuration is successful, the script returns a value of zero.

Set Redundancy Node Parameters

To change the value of parameters in an existing redundancy node, run the following script: `set_params{ switch value } […[{ switch value }].`

Example:

```
set_params -i 10 -r 15.0.0.0/16 -n 172.168.7.5
```

The index parameter (-i) is mandatory. This command sets the values of the specified parameters. If the specified parameter is already defined for the redundancy node, the value of the parameter is updated.

When a node index value of zero is specified, the values that are provided by the command for the specified parameters are treated as the default values for these parameters.

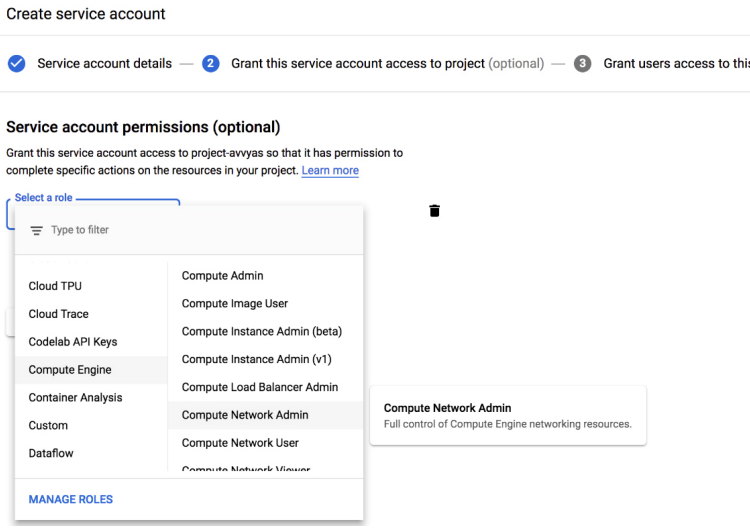
If this configuration is successful, the script returns a value of zero.

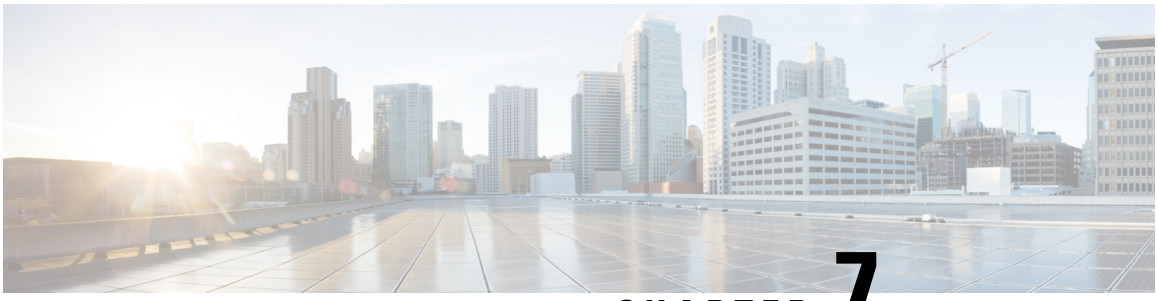
Authenticate the Cisco Catalyst 8000V Router

SUMMARY STEPS

1. Ensure that the service account associated with the Cisco Catalyst 8000V routers at least have a Compute Network Admin permission.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Ensure that the service account associated with the Cisco Catalyst 8000V routers at least have a Compute Network Admin permission.	 You can also provide the required permissions in a credentials file with name 'credentials.json' and place it under the /home/guestshell directory. The credentials file overrides the permissions supplied through the service account associated with the Cisco Catalyst 8000V instance.



CHAPTER 7

Example Configurations

Example: Redunancy Nodes with Active/Active Configuration

Consider the HA configuration where route-name1 corresponds to route entry with next hop as Cisco Catalyst 8000V 1 and route-name2 corresponds to route entry with next hop as Cisco Catalyst 8000V 2 for destination network ‘dest_network’. To configure the routers in an active/active mode, set equal route priority for route-name1 and route-name2. In this case, Google cloud distributes the traffic between the routes using a five-tuple hash for affinity, thus implementing an ECMP routing design.

The node configuration on both routers corresponding to the route entries in Google route collection for the VPC would be:

```
create_node -i 1 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr_c8000v1 -a
route-name1 -b route-name2 -p gcp -v vpc_name
```

```
create_node -i 2 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr_c8000v2 -a
route-name2 -b route-name1 -p gcp -v vpc_name
```

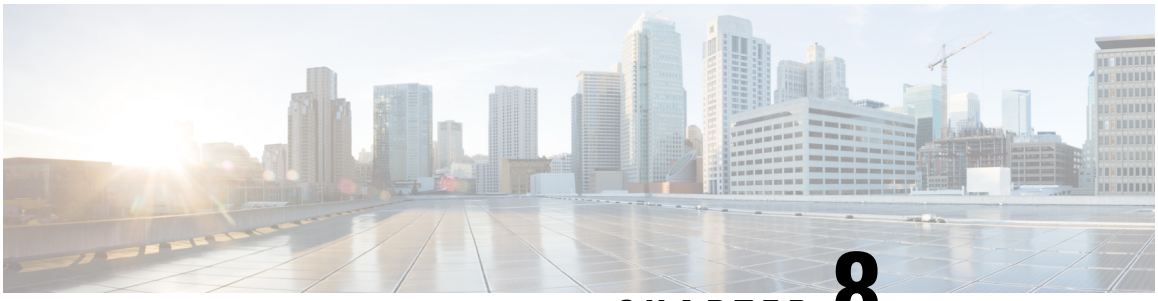
Example: Redundancy Nodes with Active-Passive Configuration

Similarly, to configure Cisco Catalyst 8000V instances in an active-passive mode, set the priority of one route higher than the other. In this case, Google cloud routes all the traffic from the VPC vpc_name to dest_network via the higher priority route (route-name1 for this example).

The node configuration on both routers corresponding to the route entries in Google route collection for the VPC would be:

```
create_node -i 1 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr_c8000v1 -a
route-name1 -b route-name2 -p gcp -v vpc_name
```

```
create_node -i 2 -g <project-id> -r dest_network -o 400 -n nexthop_ip_addr_c8000v2 -a
route-name2 -b route-name1 -p gcp -v vpc_name
```

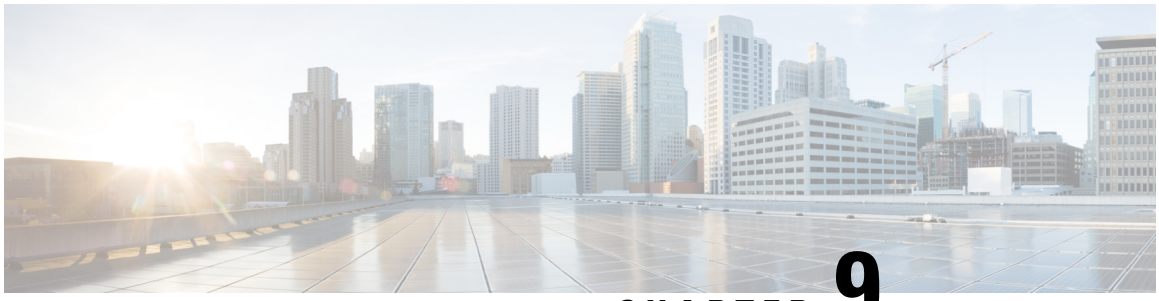


CHAPTER 8

Verify High Availability

Perform the following verification procedure by checking the log files. You can write a verbose log file to the directory ~/cloud/HA/events. Examine this log file to verify whether the operation is successful.

```
[guestshell@guestshell events]$ node_event.py -i node_index -e verify
[guestshell@guestshell events]$ cd /home/guestshell/cloud/HA/events
[guestshell@guestshell events]$ ls event.2018-06-13 20:10:21.093942
```



CHAPTER 9

Troubleshoot High Availability Issues

Open the event file that is generated. This file is a debug log of the attempt to read and update the route described by the redundancy node. If the HA setup works as expected, the configuration output displays the status *Event handling completed*. If the system does not display this status, examine the log file in detail to determine which step of the verification failed.

Some of the common causes for failure include:

- Inability to obtain authentication credentials.
- The guestshell does not have network access.
- The authentication service is not running in Guestshell.
- The credentials for Cisco Catalyst 8000V are missing or incorrect.
- The router cannot access the route table entry.
- The route table was not correctly identified in the redundancy node
- The router was not granted permission to access the route table
- The specific route specified in the redundancy node does not exist

Note Cisco recommends that you use the `node_event` script with the verify event to test the configuration and the operation of the redundancy node.

Example: Troubleshooting Issues for High Availability

Execute the following command: `router#show iox`. See the following examples that provide the possible issues and how you can check and resolve these issues:

```
Router#show iox

IOx Infrastructure Summary:
---------------------------
IOx service (CAF)    : Running
IOx service (HA)     : Not Supported
IOx service (IOxman) : Running
Libvirtd             : Running

Router#guestshell enable
```

```
Router#show app-hosting list
App id                           State
---------------------------------------------------------
guestshell                       RUNNING

Router#guestshell
[guestshell@guestshell ~]$

[guestshell@guestshell ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=38 time=25.7 ms

Possible Cause:
The configuration of IOX and the creation of the VirtualPortGroup interface to provide the
 guestshell network access is part of the "day zero" configuration of the C8000V.  If any
of the above steps did not work, check that the startup configuration of the C8000V has
been altered.

How to Fix:
A reload of the C8000V will re-apply the day zero configuration.

---------

Problem:
HA package installation failure

How to Check:
Router#guestshell
Router#guestshell
[guestshell@guestshell ~]$ ls
cloud
[guestshell@guestshell ~]$ cd cloud
[guestshell@guestshell cloud]$ ls
HA

You should see the directory ~/cloud/HA.
On an Azure provided cloud, you should also see a ~/cloud/authMgr directory.

Possible Cause:
The HA package was not installed, or was not installed using the --user option.

How to Fix:
Install the package and set up the environment:
 pip install c8000v_<provider>_ha --user
 source ~/.bashrc

---------

Problem:
HA server not running.

How to Check:
[guestshell@guestshell ~]$ systemctl status c8000v_ha
● c8000v_ha.service - C8000V High Availability service
   Loaded: loaded (/etc/systemd/user/c8000v_ha.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2019-04-08 15:01:51 UTC; 2h 1min ago
 Main PID: 286 (python)
   CGroup: /system.slice/libvirtd.service/system.slice/c8000v_ha.service
           ├─286 python /home/guestshell/.local/lib/python2.7/site-packages/c...
           └─295 python /home/guestshell/.local/lib/python2.7/site-packages/c...

On an Azure provided network, the auth-token service should also be running.
[guestshell@guestshell ~]$ systemctl status c8000v_ha
```

```
● c8000v_ha.service - C8000V High Availability service
   Loaded: loaded (/etc/systemd/user/c8000v_ha.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2019-04-08 15:01:51 UTC; 2h 1min ago
 Main PID: 286 (python)
   CGroup: /system.slice/libvirtd.service/system.slice/c8000v_ha.service
           ├─286 python /home/guestshell/.local/lib/python2.7/site-packages/c...
           └─295 python /home/guestshell/.local/lib/python2.7/site-packages/c...
[guestshell@guestshell ~]$ systemctl status auth-token
● auth-token.service - Authentication Token service
   Loaded: loaded (/etc/systemd/user/auth-token.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2019-04-08 16:08:15 UTC; 57min ago
 Main PID: 542 (python)
   CGroup: /system.slice/libvirtd.service/system.slice/auth-token.service
           └─542 /usr/bin/python /home/guestshell/.local/lib/python2.7/site-p...

Possible Cause:
If the HA server has an error and crashes, it is automatically restarted.

How to Fix:
A service can be restarted manually
[guestshell@guestshell ~]$ sudo systemctl start c8000v_ha


---------

Problem:
C8000V authentication not working on Azure.
This is an Azure specific error.

How to check:
If you perform a node_event on a redundancy node, and it fails while trying to read the
route table, it will generate a file ~/cloud/HA/events/routeTableGetRsp.
[guestshell@guestshell ~]$ cat routeTableGetRsp
{"error":{"code":"AuthenticationFailedMissingToken","message":"Authentication failed. The
'Authorization' header is missing the access token."}}

Possible Cause:
There are multiple possible causes.  And it depends upon the authentication mechanism you
are using:
 - System assigned managed identity
 - Registered application in Azure Active Directory (AAD)

Likely cause of a failure using system assigned managed identity is that it is not enabled
 on C8000V.

How to Fix:
Verify the C8000V is enabled for system assigned managed identity.
In the Azure portal, navigate to the virtual machine running the C8000V.
Under the Settings menu, select the Identity item.
Under the system assigned tab, verify the status is set to On.

When using AAD for authentication, the likely cause of the error is a mis-configuration of
 the application or a mis-match in the identifiers for the application configured in the
guestshell.

How to Fix:
The application in AAD must be given the proper permissions to read and write a route table.
In the Azure portal, navigate to the registered application you have created.
Under the API Access menu, select the Required permissions item.
Select the Windows Azure Active Directory API.  In the Enable Access pane, verify the
following permissions are set:
 - Application permission to read and write directory data
 - Delegated permission to sign in and read user profile
Select the Windows Azure Service Management API.  In the Enable Access pane, verify the
```

```
following permissions are set:
 - Delegated permission to access Azure service management as organization users

How to Fix:
In the Azure portal, navigate to the registered application you have created.
Select the Setting button for the application.
Verify the application_id, tenant_id, and application key in the portal match the values
configured in guestshell.  Verify the application key configured in guestshell is in URL
unencoded format.

-------------

Problem:
Route table entry not updated by a peer failure event.

How to Check:
For every node event a log file is generated in the directory ~/cloud/HA/events.
This file will indicate the event that was processed and its result.  Examine this file for
 possible errors.  It is likely in the case of an error that a file
~/cloud/HA/events/routeTableGetRsp is also written.  Also examine this file for additional
 insights.

Possible Causes:
A route was not correctly identified in a redundancy node.  Depending upon what parameter
in the redundancy node is in error, you may see different results.

Some examples:
[guestshell@guestshell events]$ cat routeTableGetRsp
{"error":{"code":"SubscriptionNotFound","message":"The subscription
'b0b1a9e2-444c-4ca5-acd9-bebd1e6874ef' could not be found."}}
This implies the Azure subscription ID was not entered correctly.

[guestshell@guestshell events]$ cat node*
Route GET request failed with code 403
Route table get response:
 {"error":{"code":"AuthorizationFailed","message":"The client
'b3ce41c0-bcef-41d7-9741-26bea31221c1' with object id 'b3ce41c0-bcef-41d7-9741-26bea31221c1'
 does not have authorization to perform action 'Microsoft.Network/routeTables/read' over
scope
'/subscriptions/b0b1a9e2-444c-4ca5-acd9-bebd1e6873eb/resourceGroups/gsday0-rg/providers/Microsoft.Network/routeTables/gsday0-sub4-RouteTable'."}}
Route table not found.
This implies the name of the route table was incorrect or does not exist.

[guestshell@guestshell events]$ cat node*
Did not find route 17.0.0.0/8 event type peerFail
This implies that the route does not exist.

How to Fix:
Make sure the identifiers in the redundancy node match the values in the cloud provider's
portal.

------------

Problem:
Route table entry not updated by a peer failure event.

How to Check:
For every node event a log file is generated in the directory ~/cloud/HA/events.
This file will indicate the event that was processed and its result.  Examine this file for
 possible errors.  It is likely in the case of an error that a file
~/cloud/HA/events/routeTableGetRsp is also written.  Also examine this file for additional
 insights.

Possible Causes:
```

```
The C8000V has not been given permission to access the route table.
Fetching the route table
Route table get response:
 {"error":{"code":"AuthorizationFailed","message":"The client
'b3ce41c0-bcef-41d7-9741-26bea31221c1' with object id 'b3ce41c0-bcef-41d7-9741-26bea31221c1'
 does not have authorization to perform action 'Microsoft.Network/routeTables/read' over
scope
'/subscriptions/b0b1a9e2-444c-4ca5-acd9-bebd1e6873eb/resourceGroups/gsday0-rg/providers/Microsoft.Network/routeTables/gsday0-sub2-RouteTable'."}}
Route GET request failed with code 403
Route table get response:
 {"error":{"code":"AuthorizationFailed","message":"The client
'b3ce41c0-bcef-41d7-9741-26bea31221c1' with object id 'b3ce41c0-bcef-41d7-9741-26bea31221c1'
 does not have authorization to perform action 'Microsoft.Network/routeTables/read' over
scope
'/subscriptions/b0b1a9e2-444c-4ca5-acd9-bebd1e6873eb/resourceGroups/gsday0-rg/providers/Microsoft.Network/routeTables/gsday0-sub2-RouteTable'."}}
Route table not found.
C8000V HA: Set route table for verify
Route Table not found


If none of these troubleshooting tips have resolved your problem, run this command:
[guestshell@guestshell ~]$ cd ~/cloud/HA
[guestshell@guestshell ~]$ bash debug_ha.sh
[guestshell@guestshell ~]$ ls /bootflash
You should see a file name ha_debug.tar.  Copy this file off the C8000V and provide it to
Cisco Technical Support for analysis.
```