Extending Your IT Infrastructure Into Amazon Web Services Using Cisco DMVPN and the Cisco Cloud Services Router 1000V Series

Cisco Validated Design

August 2014

Table of Contents

Navigator

The Navigator helps you determine the applicability of this guide by summarizing its key elements: the scope or breadth of the technology covered, the proficiency or experience recommended, and the documents related to this guide. This section is a quick reference only. For more details, see the Introduction.

Use Cases

This guide addresses the following technology use cases:

- Deploying Cisco Cloud Services Router (CSR) 1000V Series in Amazon Web Services (AWS)

For more information, see the "Technology Use Cases" section in this guide.

Scope

This guide covers the following areas of technology and products:

- Cisco CSR 1000V Series
- Dynamic Multipoint VPN (DMVPN)
- AWS

For more information, see the "Design Overview" section in this guide.

Proficiency

This guide is for people with the following technical proficiencies—or equivalent experience:

- **CCNA Routing and Switching**—1 to 3 years installing, configuring, and maintaining routed and switched networks
- **CCNA Service Provider**—1 to 3 years installing, configuring, and maintaining routed and switched networks

Related Cisco Validated Design

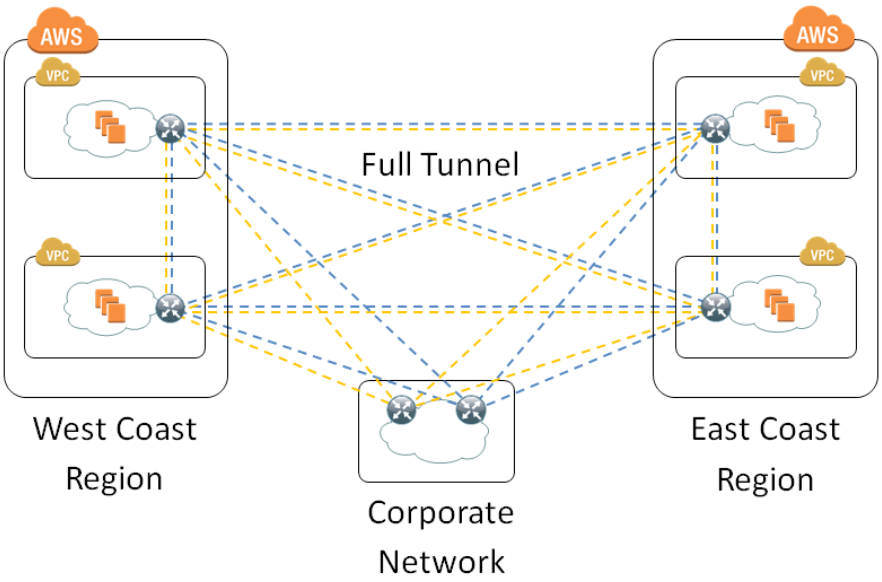
VPN WAN Technology Design Guide

Introduction

Technology Use Cases

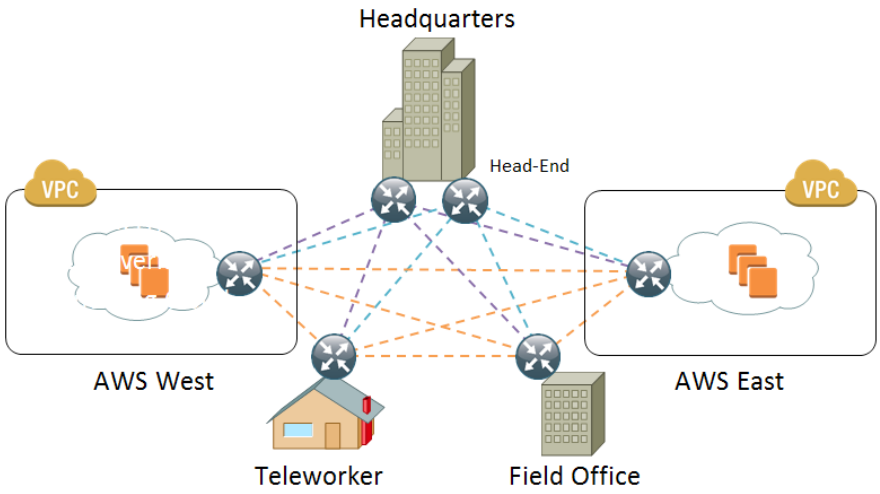
Organizations typically connect to their applications through a single VPN tunnel between their data center and AWS. With the Cisco CSR 1000V Series deployed in AWS, every enterprise- and branch-office location can now have direct VPN access into the AWS-hosted applications without back-hauling through an existing data center. This approach reduces latency, eliminates the need for expensive private WAN services, avoids per-VPN-tunnel costs that Amazon charges, and even allows AWS to participate in existing route-based VPN topologies.

Figure 1 - Multi-site hybrid cloud overlay network



AWS does not provide VPN connectivity between VPCs in discrete AWS regions, making multiregion cloud deployments complex. By deploying a Cisco CSR 1000V Series Router in each region's VPC and interconnecting Cisco CSR 1000V Series Routers through a VPN, enterprises can create a global, secure network topology within the AWS cloud.

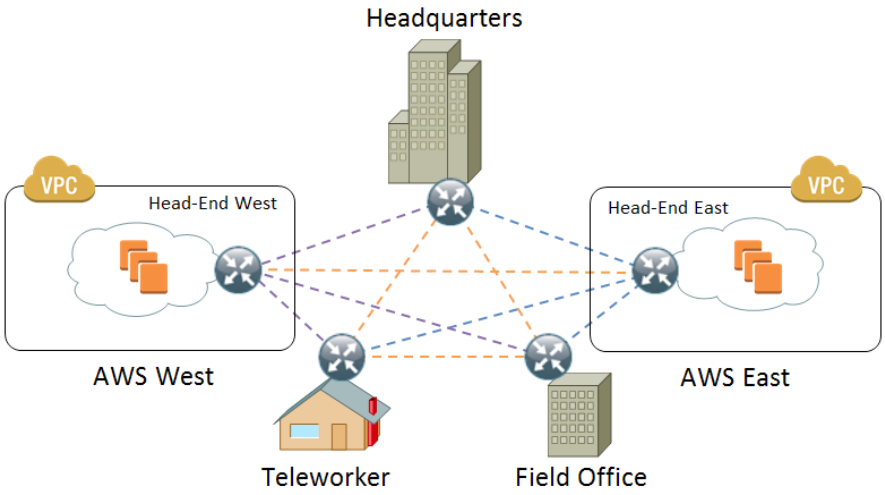
Figure 2 - Hybrid cloud overlay connecting cloud, headquarters, field offices, and teleworkers



The Cisco CSR 1000V Series is part of a family of platforms that includes the latest edge, branch-office, service, and telecommuting routers, providing the ideal platform on which to build a fully connected enterprise network. Together, these platforms provide easy multi-homing over any carrier service offering, a single routing control plane with minimal peering to the provider, automatic site-to-site IPSec tunnels, and comprehensive threat-defense.

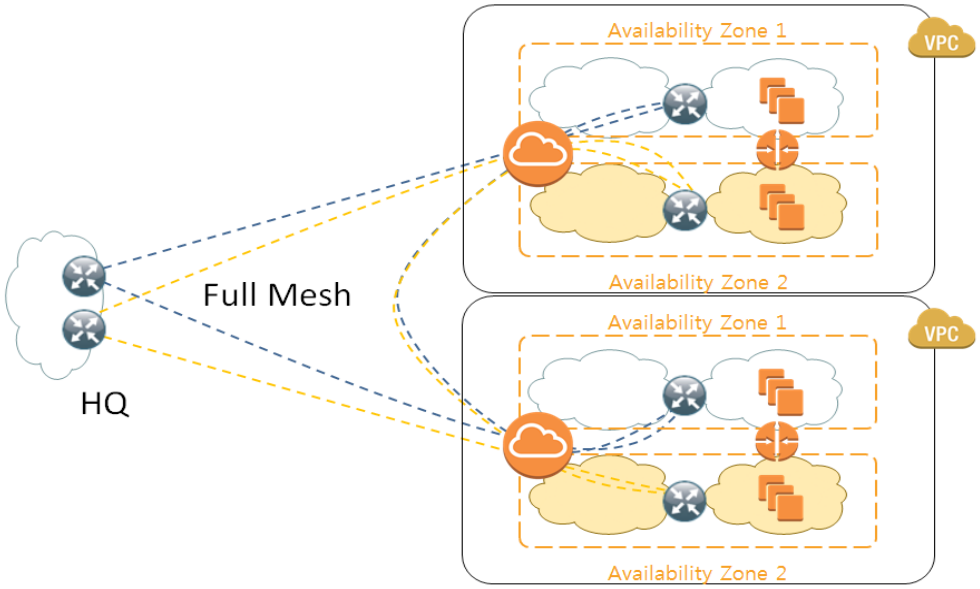
Figure 2 shows how dynamically created tunnels help avoid bottlenecks by connecting the AWS hosted, fully connected hybrid cloud.

Figure 3 - Cloud-anchored enterprise-wide overlay



If your organization wants a highly available VPN cloud with geographically disparate headend routers, you can place the headend routers in separate AWS data centers. The full mesh of dynamically created tunnels makes it possible to avoid potential bottlenecks and increased bandwidth costs associated with cloud-based headend routers by allowing spoke-to-spoke traffic (Figure 3). Only traffic destined for the application servers in the cloud flows through the headend routers.

Figure 4 - Highly available enterprise overlay with fully redundant AWS cloud router



In addition to high availability at the head-end, the Cisco CSR 1000V Series Router can provide high availability within the AWS VPC. You can place multiple Cisco CSR 1000V Series Routers in separate availability zones with a set of instances using that CSR 1000v as their default route. When maintenance is required on one of the Cisco CSR 1000V Series Routers, traffic can be routed from the CSR 1000v in one availability zone to the CSR 1000V in the other availability zone, either manually or automatically, through active monitoring. Each of the two Cisco CSR 1000V Series Routers can route to any other spoke in the Cisco DMVPN network as well as to other CSR 1000V Routers within AWS.

This design enables the following capabilities:

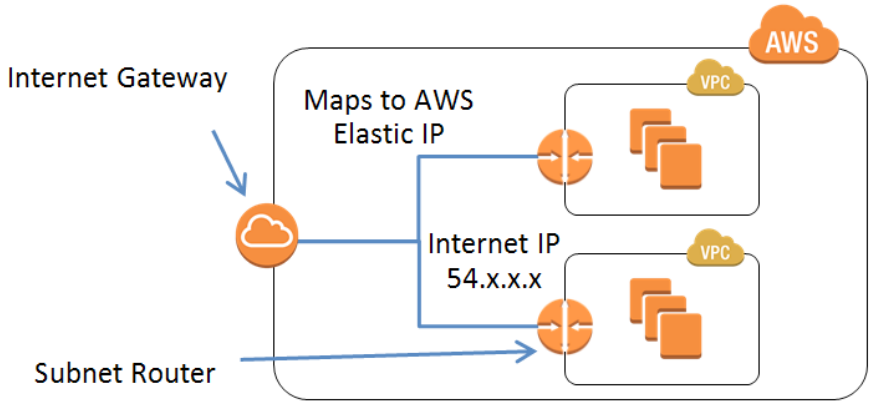
- **Single routing plane**–The Cisco CSR 1000V Series routing protocol support for Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), and Border Gateway Protocol (BGP) allows it to integrate smoothly into the rest of your enterprise network instead of creating islands in the cloud.
- **High availability**–The dual-hub Cisco DMVPN design provides a fault-tolerant overlay network with no single point of failure. This fault tolerance is increased when the hubs are geographically disparate.
- **Defense in depth**–The security provided by the overlay network through IPsec tunnels and Zone-Based Firewalls (ZBFWs) is disjointed from the underlying AWS infrastructure, providing protection for your corporate network if the AWS account is compromised.
- **Unified security policy**–Using ZBFWs, your organization can use the Cisco CSR 1000V Series in order to create a cohesive security policy across your entire network, including branch offices, mobile workers, and public clouds.

Deploying Cisco CSR in AWS

AWS Virtual Private Cloud Details

AWS virtual private clouds (VPCs) are logically isolated networks with their own IP range, routes, security groups, and network access control lists (ACLs). IP ranges can overlap between VPCs, but NAT is required for the instances within a VPC in order to either access or be accessed by resources outside the VPC. The Internet gateway routes outside and between VPCs and the subnet router routes within the VPC. The subnet router can route between any two subnets via the "local" route.

Figure 5 - AWS VPC overview



For design purposes, Cisco CSR is thought of as sitting between the subnet outer and the instance (Figure 6). However, because of the network mechanics of an AWS VPC, the CSR sits parallel to the actual instances (Figure 7). For this reason, for the traffic to flow through the CSR, a subnet route pointing to the CSR must be added or the default gateway of each of the instances must be changed to the CSR.

Figure 6 - Logical placement of CSR in VPC

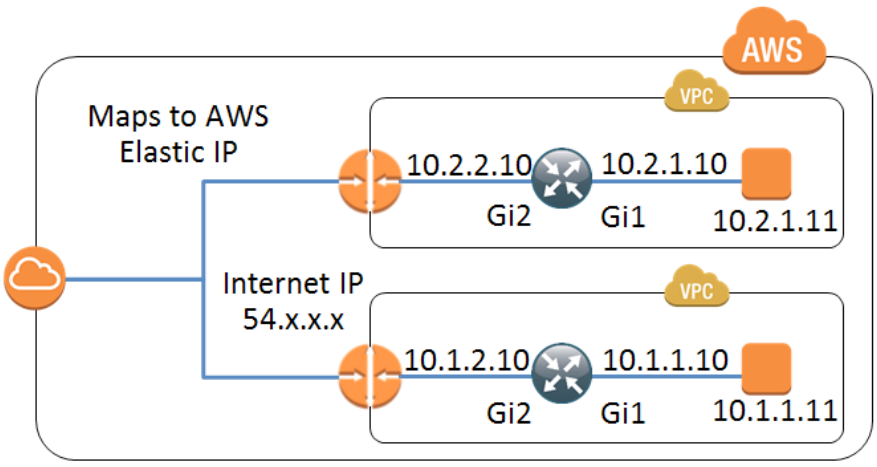
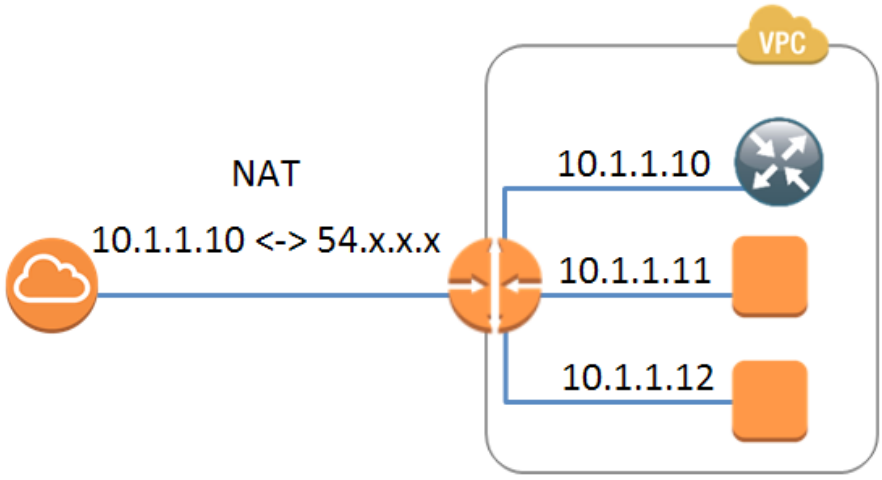


Figure 7 - Actual placement of CSR in VPC



You can deploy the CSR in either a single- or dual-subnet configuration. The dual-subnet configuration (Figure 8) is recommended because it is most like traditional router deployments and allows full-functionality of all features supported in AWS. In some circumstances, however, a single subnet deployment (Figure 9) can be simpler to integrate into existing networks.

Figure 8 - Dual subnet deployment

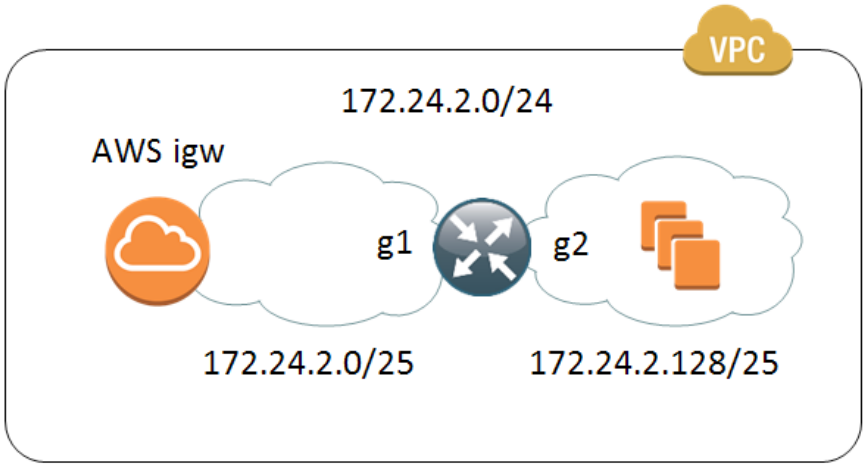
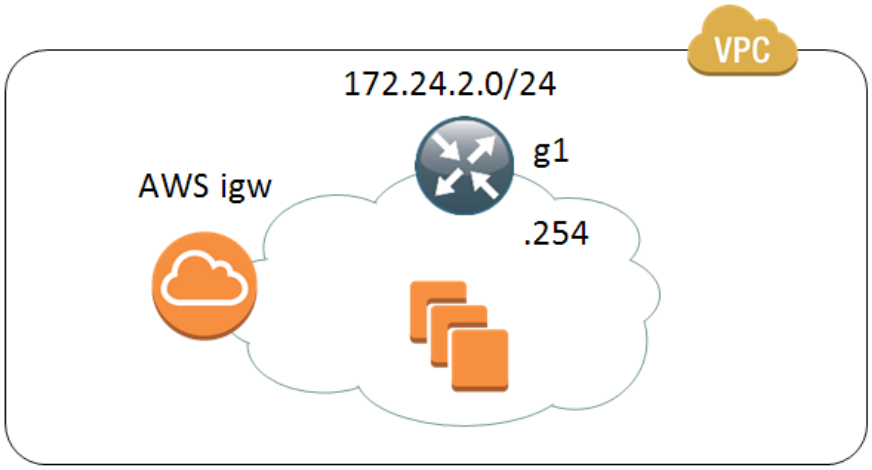


Figure 9 - Single subnet deployment



Supported Instance Types

Reader Tip

For more information, see the following topic:
http://www.cisco.com/c/en/us/td/docs/routers/CSR1000/software/aws/CSRaws/awsinstall.html

Table 1 - AMI instance specifications (bring your own license)

Instance type	EC2 compute units	Virtual cores	Memory required	Platform	I/O	Maximum number of network interfaces supported per instance (EC2-VPC only)
Standard Medium (m1.medium)	2	1	3.75 GB	32-bit 64-bit	Moderate	2
Standard Large (m1.large)	8	2 (with 2 ECUs each)	7.5 GB	64-bit	Moderate	4
Standard XL (m1. xlarge)	8	4 (with 2 ECUs each)	15 GB	64-bit	High	4
M3 Extra Large (m3.xlarge)	13	4 (with 3.25 ECUs each)	15 GB	64 bit	High	4

Table 2 - AMI instance specifications (hourly billed)

Instance type	EC2 compute units	Virtual cores	Memory required	Platform	I/O	Maximum number of network interfaces supported per instance (EC2-VPC only)
M3 Medium (m3.medium)	3	1 (with 3 ECUs each)	3.75 GB	64 bit	Moderate	2
M3 Large (m3.large)	6.5	2 (with 3.25 ECUs each)	7.5 GB	64 bit	Moderate	3
M3 Extra Large (m3.xlarge)	13	4 (with 3.25 ECUs each)	15 GB	64 bit	High	4
C3 Large (c3.large)	7	2 (with 3.5 ECUs each)	3.75 GB	64 bit	Moderate	3
C3 Extra Large (c3.xlarge)	14	4 (with 3.5 ECUs each)	7.5 GB	64 bit	Moderate	4

Deployment Details

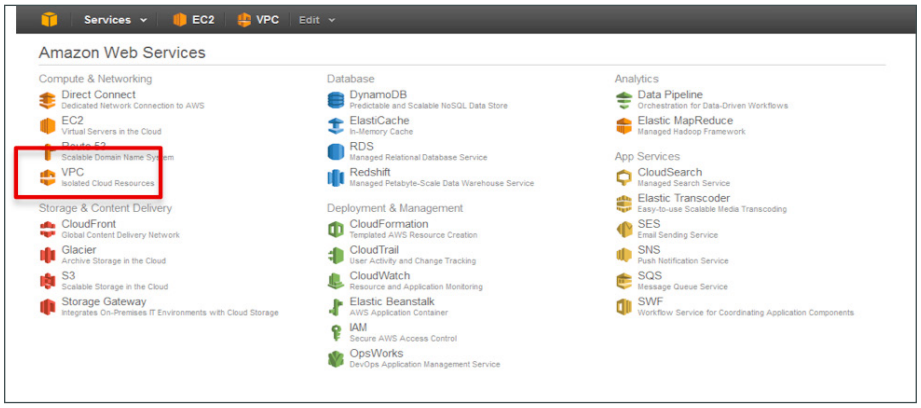
PROCESS

Deploying CSR in AWS

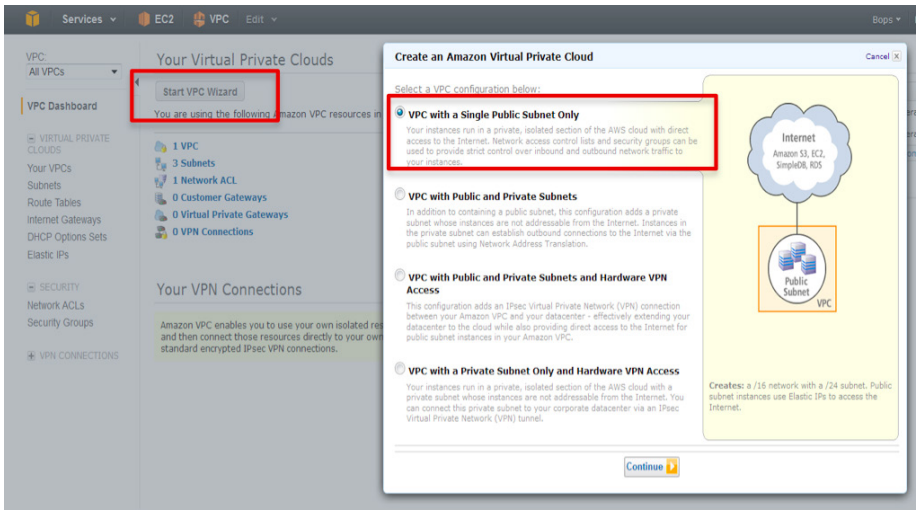
1. Create virtual private cloud

Procedure 1: Create virtual private cloud

Step 1: Log in to AWS and in the left pane, click **VPC**.

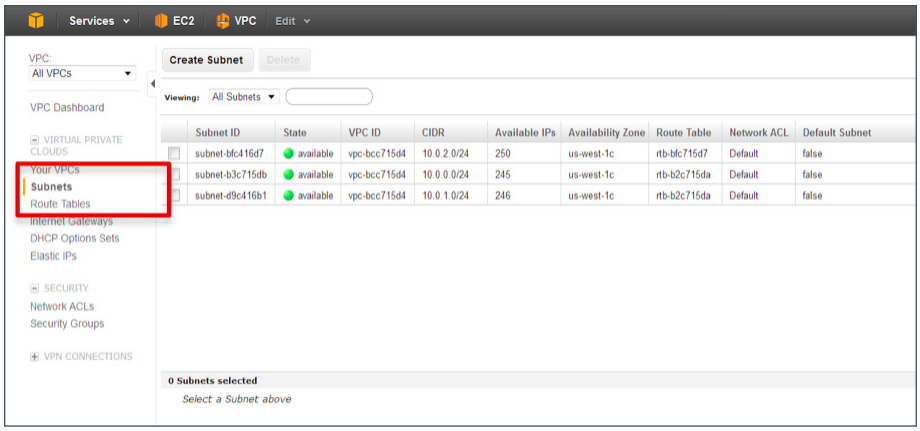


Step 2: Click **Start VPC Wizard**, and then select **VPC with Single Public Subnet Only**.



Step 3: Create the required subnets in the VPC, with the following properties:

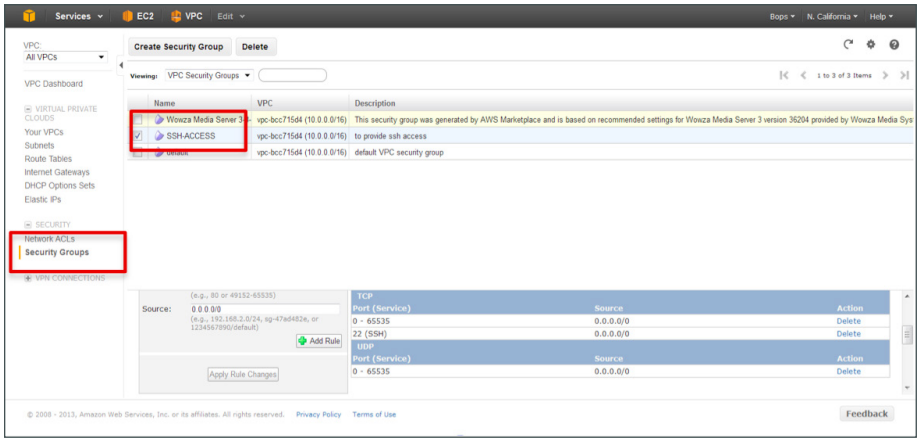
- **Default subnet**–10.0.0.0/24 (mapped to public IP)
- **Additional subnets**–0.0.1.0/24 and 10.0.2.0/24. These are private IPs and could be the "inside facing" interfaces for the CSR.



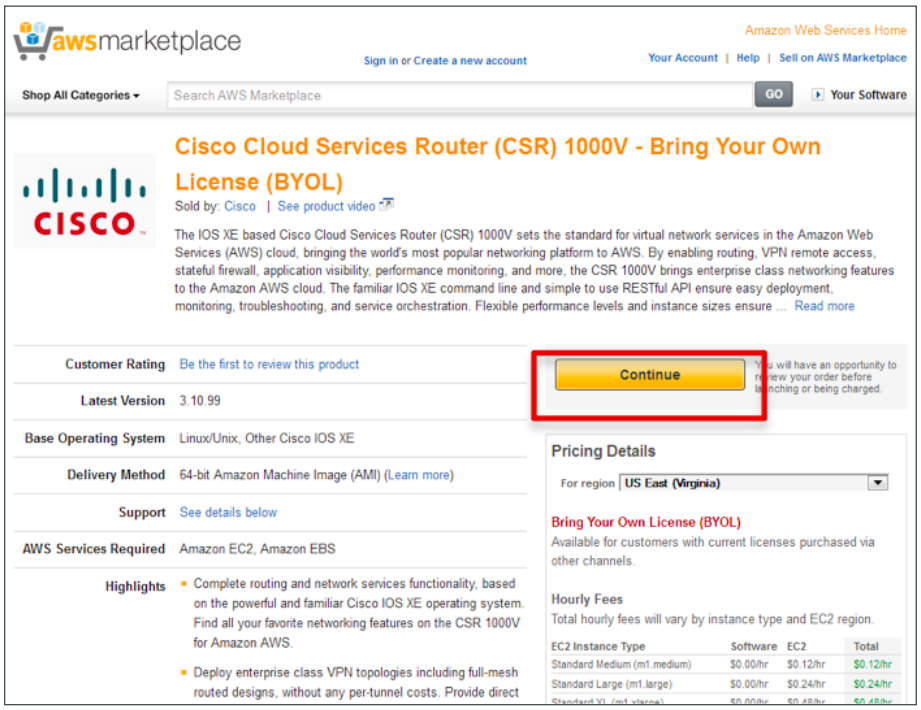
Step 4: Create a security group for the CSR, with the following properties:

- **Name**–SSH-ACCESS
- **TCP port 22 traffic**–Permitted inbound
- **SSH access to CSR for management**–Enabled

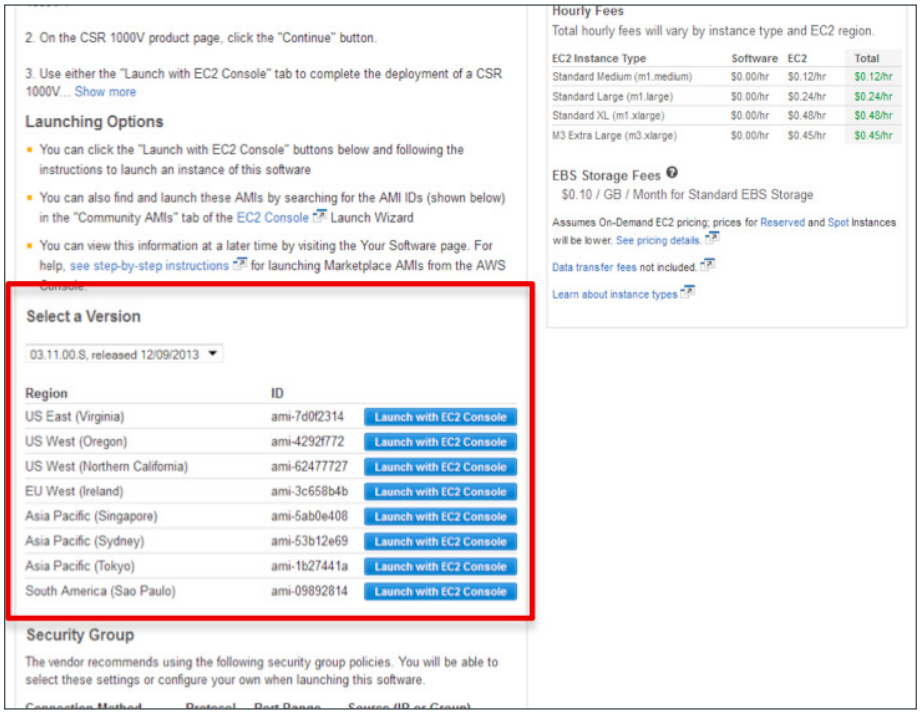
You can create additional security groups based on your use-case.



Step 5: Locate the CSR product page.



Step 6: Launch the CSR into a region.



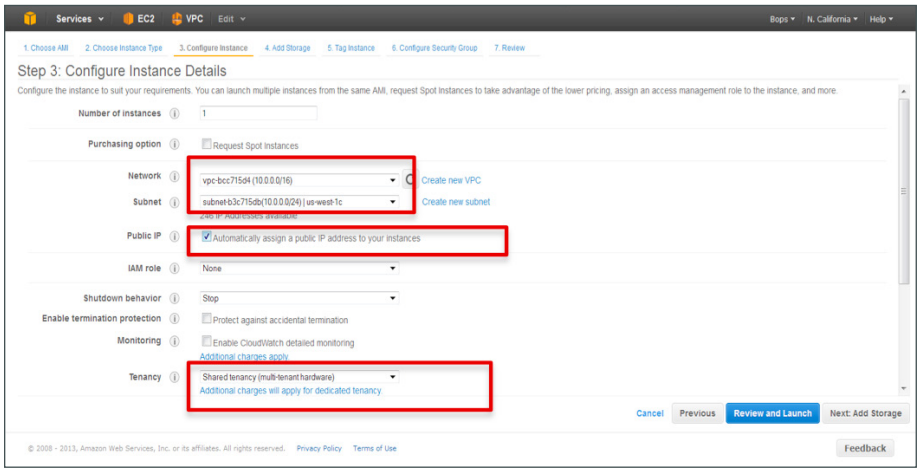
Step 7: Choose instance type. Notes:

- See Tables 1 and 2 for supported instance types.
- The minimum requirements are m1.medium for 10Mbps and m1.large for 50 Mbps.
- *ECU* stands for Elastic Compute Unit, Amazon's proprietary way of measuring CPU capacity.
- Almost all EC2 instances are hyperthreaded

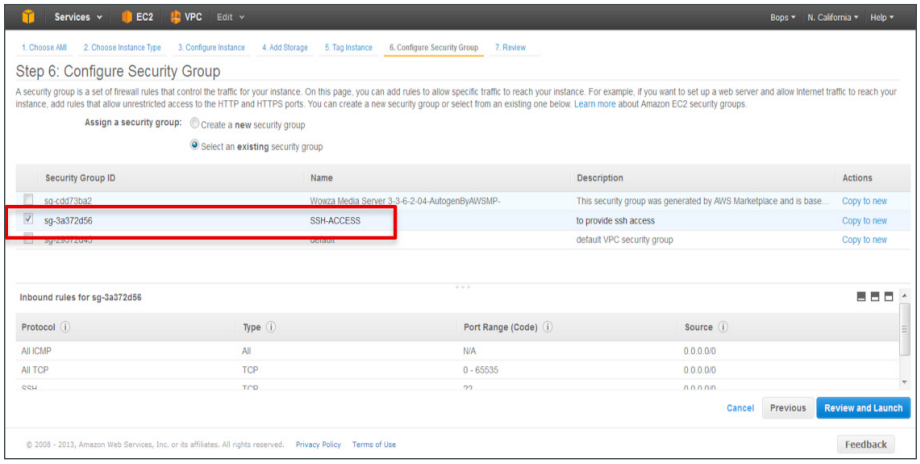
Step 8: Launch CSR into the previously created VPC, and use the following properties:

- **Automatically assign a public IP to your instances**–Enabled
- **Shared tenancy**–Default

  Dedicated tenancy (dedicated hardware) offers predictable performance for an increased price.

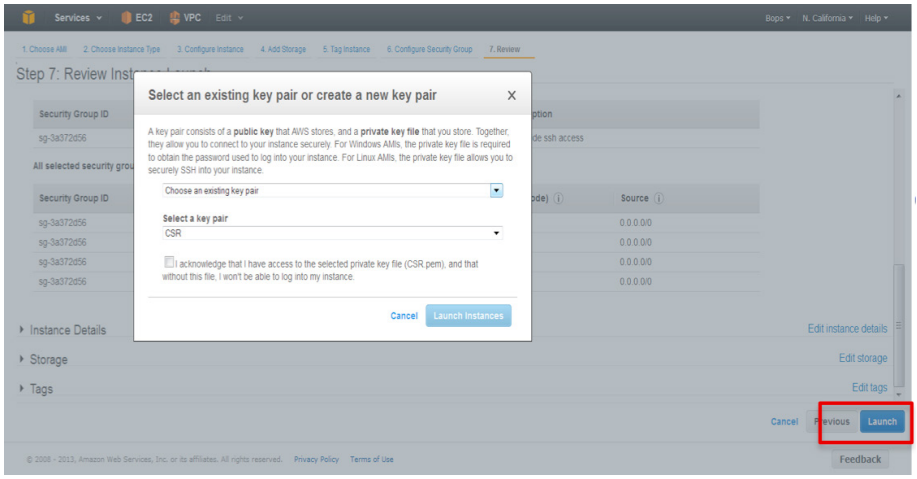


Step 9: Associate with security groups (SSH-ACCESS).



Step 10: Associate a private key with the CSR. Notes:

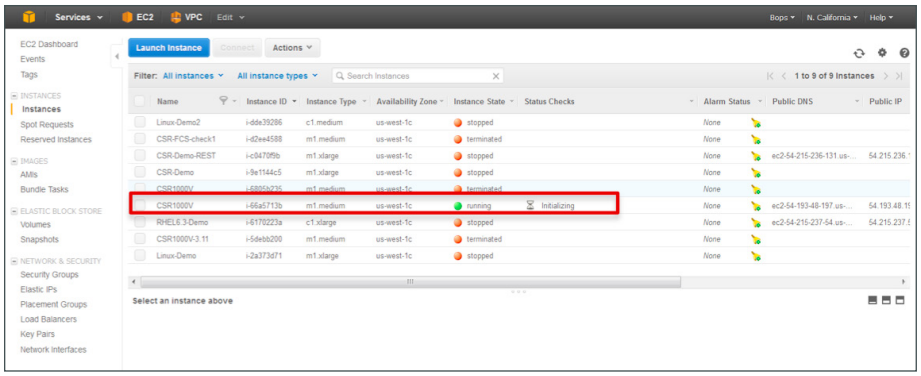
- You can create a new pair if desired.
- *Key pair* is a private key and a public key.
- You must provide the private key in order to authenticate and connect to the instance. The public key is stored on AWS



Step 11: Monitor the instance. The instance typically takes 5-10 minutes to deploy. The status will change to "2/2/ checks passed".

Tech Tip

The AWS System Log for the CSR instance will be incomplete; however, this does not imply an unsuccessful boot.



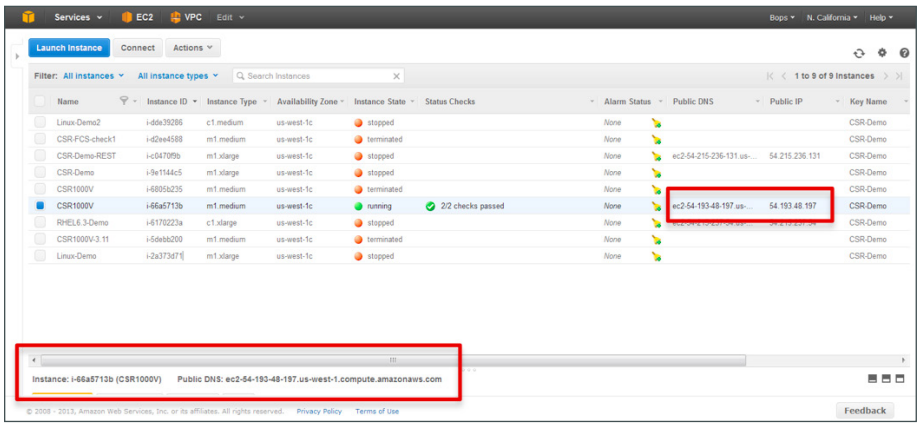
Step 12: Verify the CSR in the VPC. SSH into the CSR instance using this IP. Login as "ec2-user". No password is required.

```
ssh -i <key file> ec2-user@<ip address>
```

Tech Tip

If you are using Windows Putty SSH client, follow instructions here:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html



Step 13: Verify the public interface.

This public IP is a 1:1 NAT to the private IP performed by the Internet Gateway for the VPC and is transparent to user.

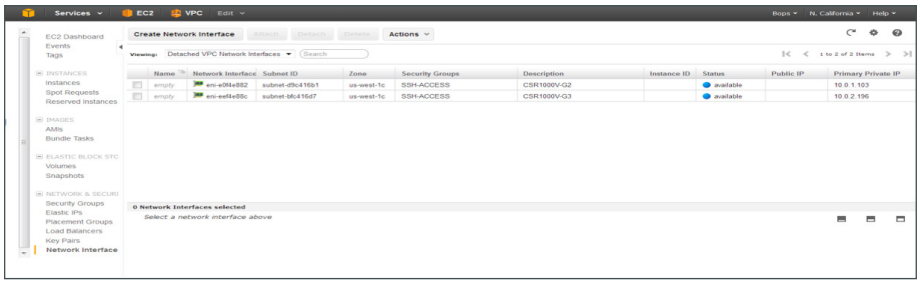
```
ip-10-0-0-24#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       10.0.0.24       YES DHCP   up                    up
VirtualPortGroup0      10.0.0.24       YES unset  down                  down
ip-10-0-0-24#
```

Caution

Automatic public IP assignment is only available during launch. Rebooting CSR will disassociate the public IP and the instance will become inaccessible.

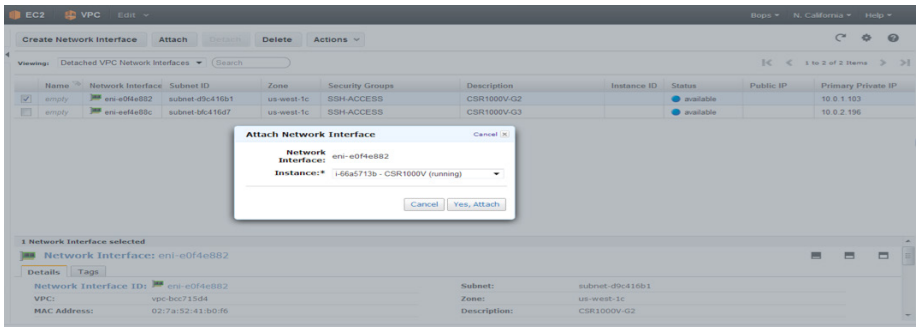
To ensure that the IP is persistent across reboots, associate the CSR instance with an Elastic IP.

Step 14: Configure and manage the CSR.



If you're using a single subnet deployment as shown in Figure 9, skip steps 15-17.

Step 15: Attach the network interface.



Step 16: Configure IP address.

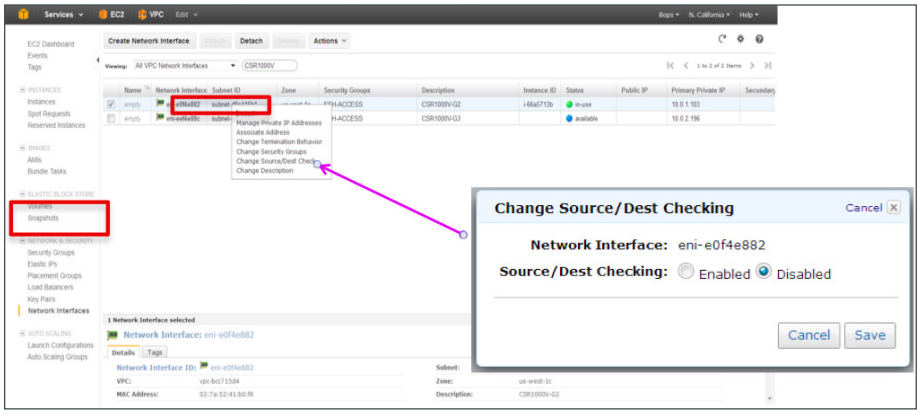
```
ip-10-0-0-24#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       10.0.0.24       YES DHCP   up                    up
GigabitEthernet2       unassigned      YES unset  administratively down down

ip-10-0-0-24(config)#int g2
ip-10-0-0-24(config-if)#ip address 10.0.1.103 255.255.255.0
ip-10-0-0-24(config-if)#no shut
ip-10-0-0-24(config-if)#end

ip-10-0-0-24#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       10.0.0.24       YES DHCP   up                    up
GigabitEthernet2       10.0.1.103      YES manual up                    up
```

IP address defined during interface creation

Step 17: Disable Source/Dest checking.



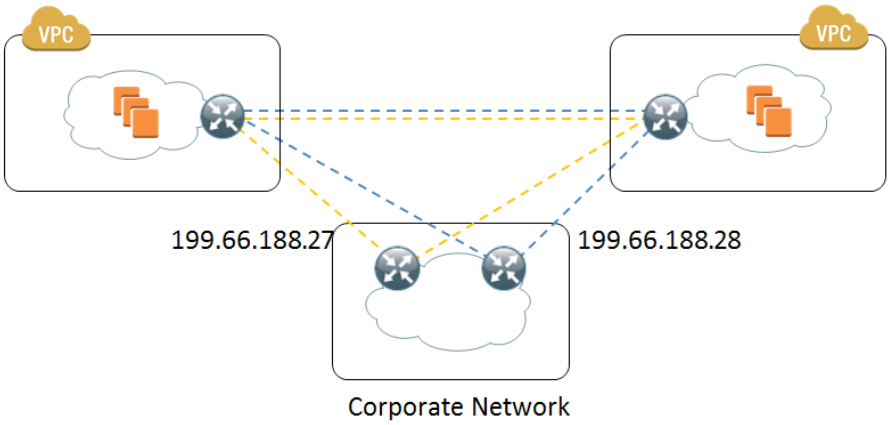
DMVPN Design: Disallowing Direct Internet Access from Spokes

Design Overview

You can configure Cisco DMVPN to either allow or disallow routers in the AWS network spokes from directly accessing Internet. This example disallows direct Internet access by placing the outside interface of the AWS Cisco CSR 1000V Series Router in a VRF and then sending a default route from the Cisco DMVPN hub routers. You could use this scenario for private enterprise applications that are hosted on AWS and therefore do not need direct Internet connectivity or for public applications that should be accessed through the enterprise Internet connections.

The next two DMVPN design example use Figure 10 as a reference.

Figure 10 – DMVPN Example Reference Diagram



Deployment Details

This design uses a single DMVPN, dual-hub configuration, EIGRP as the Cisco DMVPN routing protocol, and OSPF as the enterprise routing protocol. The AWS Cisco CSR 1000V Series Routers are configured as DMVPN spokes and EIGRP stub routers. The DMVPN hub routers, typically located in the enterprise headquarters locations, advertise a default route to the Cisco DMVPN spokes and advertise the AWS subnets to the rest of the enterprise. Cisco DMVPN Phase 3 with Next Hop Resolution Protocol (NHRP) redirection is configured to provide spoke-to-spoke tunnel support. This configuration allows AWS application in different Amazon VPCs to communicate directly with each other. Additionally, enterprise branch-office sites can be part of the same Cisco DMVPN, allowing path optimization where the branch office can use secure, direct access to the AWS-hosted applications without having to transit the headquarters' network.

PROCESS

Configuring Hub

1. Configure ISAKMP and IPsec
2. Configure the mGRE tunnel
3. Configure EIGRP
4. Configure OSPF

Procedure 1: Configure ISAKMP and IPsec

Step 1: Configure the Internet security association and key management protocol (ISAKMP) policy.

The ISAKMP policy for DMVPN uses the following:

- 256-bit advanced encryption standard (AES)
- 256-bit secure hash standard (SHA)
- Authentication by pre-shared key (PSK)

```
crypto isakmp policy 10
 encr aes 256
 hash sha256
 authentication pre-share
```

Step 2: Configure the ISAKMP pre-shared key.

```
crypto isakmp key Cisco123 address 0.0.0.0
```

Step 3: Define the IPsec transform set.

A *transform set* is an acceptable combination of security protocols, algorithms, and other settings to apply to IPsec-protected traffic. Peers agree to use a particular transform set when protecting a particular data flow.

The IPsec transform set for DMVPN uses the following:

- Cisco Encapsulating Security Payload (ESP) with the 256-bit AES encryption algorithm
- Cisco ESP with the 256-bit SHA authentication algorithm

```
crypto ipsec transform-set xform esp-aes 256 esp-sha256-hmac
 mode transport
```

Step 4: Create the IPsec profile.

Step 5: The IPsec profile creates an association between an ISAKMP profile and an IPsec transform-set.

```
crypto ipsec profile ipsec-prof
 set transform-set xform
```

Procedure 2: Configure the mGRE tunnel

Step 1: Configure basic interface settings

```
interface Tunnel0
 ip address 172.24.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip tcp adjust-mss 1360
```

Tech Tip

Configure the IP MTU to 1400 and the ip tcp adjust-mss to 1360. There is a 40 byte difference that corresponds to the combined iP and TCP header length.

Step 2: Configure the tunnel.

Step 3: DMVPN uses multipoint GRE (mGRE) tunnels. This type of tunnel requires a source interface only. Use the same source interface that you use to connect to the Internet. Enabling encryption on this interface requires the application of the IPsec profile configured in the previous procedure.

```
interface Tunnel0
 tunnel source GigabitEthernet1
 tunnel mode gre multipoint
 tunnel key 1
 tunnel protection ipsec profile ipsec-prof
```

Step 4: Configure NHRP.

Step 5: The DMVPN hub router acts in the role of NHRP server for all of the spokes. NHRP is used by remote routers to determine the tunnel destinations for peers attached to the mGRE tunnel.

```
interface Tunnel0
 ip nhrp map multicast dynamic
 ip nhrp map 172.24.0.2 199.66.188.28
 ip nhrp network-id 1
 ip nhrp redirect
```

Step 6: Configure EIGRP.

Step 7: Summarize all routes as a default route towards the spokes.

```
interface Tunnel0
 ip summary-address eigrp 1 0.0.0.0 0.0.0.0
```

Procedure 3: Configure EIGRP

Step 1: Enable the EIGRP process for DMVPN

The tunnel interface is the only EIGRP interface, and you need to explicitly list its network range.

```
router eigrp 1
 network 172.24.0.0
 passive-interface default
 no passive-interface Tunnel1
```

Procedure 4: Configure OSPF

Step 1: Configure OSPF.

OSPF is used to distribute the routes from the DMVPN network into the local enterprise routing tables.

```
router ospf 1
 redistribute static subnets route-map static2ospf
```

Step 2: Add the static route for the DMVPN Network.

Step 3: This should be a summary route of the DMVPN network and all of the networks that exist at the spokes.

```
ip route 172.24.0.0 255.255.0.0 Null0
```

Step 4: Create the access list for the route map.

```
access-list 1 permit 172.24.0.0 0.0.255.255
```

Step 5: Configure route maps.

```
route-map static2ospf permit 10
 match ip address 1
```

PROCESS

Spoke Configuration

1. Configure a front door VRF
2. Configure ISAKMP and IPsec
3. Configure the mGRE tunnel
4. Configure EIGRP

Procedure 1: Configure a front door VRF

To improve security, you can deploy a front-door VRF to separate the externally facing interface required to terminate the tunnel from the internal network. However, deploying a Front-Door VRF in AWS presents particular challenges because of the lack of console access. Specifically, IOS removes the address from an interface when it is added to a VRF. You can use the Embedded Event Manager to programmatically add the interface to the VRF, and then reconfigure the interface even though communication to the CSR is lost.

Step 1: Create a VRF.

```
vrf definition internet-vrf
 rd 1:1
!
address-family ipv4
exit-address-family
```

Step 2: Create a Cisco EEM applet.

```
event manager applet fvrf
 event none
 action 1.0 cli command "enable"
 action 1.1 cli command "conf t"
 action 1.2 cli command "interface gig1"
 action 1.3 cli command "vrf forwarding internet-vrf"
 action 1.4 cli command "ip address dhcp"
 action 2.0 cli command "end"
```

Step 3: Run the EEM Applet

```
event manager run fvrf
```

Connectivity will be lost when the command is run

Step 4: Reconnect to the CSR and verify interface configuration.

```
interface GigabitEthernet1
 vrf forwarding internet-vrf
 ip address dhcp
 negotiation auto
```

Procedure 2: Configure ISAKMP and IPsec

Step 1: Configure ISAKMP Policy.

```
crypto isakmp policy 10
 encr aes 256
 hash sha256
 authentication pre-share
```

Step 2: Configure the ISAKMP pre-shared key.

```
crypto isakmp key Cisco123 address 0.0.0.0
```

Step 3: Define the IPsec transform set.

```
crypto ipsec transform-set xform esp-aes 256 esp-sha256-hmac
 mode transport
```

Step 4: Create the IPsec profile

```
crypto ipsec profile ipsec-prof
 set transform-set xform
```

Procedure 3: Configure the mGRE tunnel

Step 1: Configure basic interface settings.

```
interface Tunnel0
 ip address 172.24.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip tcp adjust-mss 1360
```

Step 2: Configure the tunnel.

```
interface Tunnel0
 tunnel source GigabitEthernet1
 tunnel mode gre multipoint
 tunnel key 1
 tunnel protection ipsec profile ipsec-prof
```

Step 3: Configure NHRP. Specify the IP addresses of the hub routers below.

```
interface Tunnel0
 ip nhrp network-id 1
 ip nhrp nhs 172.24.0.1 nbma 199.66.188.27 multicast
 ip nhrp nhs 172.24.0.2 nbma 199.66.188.28 multicast
 ip nhrp shortcut
```

Procedure 4: Configure EIGRP

Step 1: Enable the EIGRP process for DMVPN.

All interfaces on the router are EIGRP interfaces, but only the DMVPN tunnel interface is non-passive. The network range must include all interface IP addresses either in a single network statement or in multiple network statements. All DMVPN spoke routers should run EIGRP stub routing to improve network stability and reduce resource utilization.

```
router eigrp 1
 network 172.24.0.0
 passive-interface default
 no passive-interface Tunnel1
 eigrp stub connected
```

DMVPN Design: Direct Internet Access from AWS Spokes

Design Overview

This example is similar to the previous Cisco DMVPN design. The main difference is that the outside interface of the AWS Cisco CSR 1000V is not placed in a separate VRF, but it is instead kept in the global table. Instead of receiving a default route from the Cisco DMVPN hub router, the AWS Cisco CSR 1000V uses the default route that the AWS DHCP server provides to send traffic directly to the Internet. At the Cisco DMVPN hub routers, specific OSPF routes are redistributed into the Cisco DMVPN EIGRP process to control which networks are reached through the Cisco DMVPN network. Finally, NAT is used to translate the inside address to the elastic IP address assigned to the Cisco CSR 1000V Series.

Deployment Details

PROCESS

Configuring the Hub

1. Configure outside interface
2. Configure ISAKMP and IPsec
3. Configure the mGRE tunnel
4. Configure EIGRP
5. Configure OSPF

Procedure 1: Configure outside interface

Step 1: Configure the outside interface to DHCP.

```
interface GigabitEthernet1
 ip address dhcp
 negotiation auto
```

Procedure 2: Configure ISAKMP and IPsec

Step 1: Configure the ISAKMP policy.

The ISAKMP policy for DMVPN uses the following:

- 256-bit AES
- 256-bit SHA
- Authentication by PSK

```
crypto isakmp policy 10
 encr aes 256
 hash sha256
 authentication pre-share
```

Step 2: Configure the ISAKMP pre-shared key.

```
crypto isakmp key Cisco123 address 0.0.0.0
```

Step 3: Define the IPsec transform set.

A transform set is an acceptable combination of security protocols, algorithms, and other settings to apply to IPsec-protected traffic. Peers agree to use a particular transform set when protecting a particular data flow.

The IPsec transform set for DMVPN uses the following:

- Cisco ESP with the 256-bit AES encryption algorithm
- Cisco ESP with the 256-bit SHA authentication algorithm

```
crypto ipsec transform-set xform esp-aes 256 esp-sha256-hmac
 mode transport
```

Step 4: Create the IPsec profile.

The IPsec profile creates an association between an ISAKMP profile and an IPsec transform-set.

```
crypto ipsec profile ipsec-prof
 set transform-set xform
```

Procedure 3: Configure the mGRE tunnel

Step 1: Configure basic interface settings

```
interface Tunnel0
 ip address 172.24.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip tcp adjust-mss 1360
```

Tech Tip

Configure the IP MTU to 1400 and the ip tcp adjust-mss to 1360. There is a 40 byte difference that corresponds to the combined iP and tcP header length.

Step 2: Configure the tunnel.

DMVPN uses mGRE tunnels. This type of tunnel requires a source interface only. Use the same source interface that you use to connect to the Internet. Enabling encryption on this interface requires the application of the IPsec profile configured in the previous procedure.

```
interface Tunnel0
 tunnel source GigabitEthernet1
 tunnel mode gre multipoint
 tunnel key 1
 tunnel protection ipsec profile ipsec-prof
```

Step 3: Configure NHRP.

The DMVPN hub router acts in the role of NHRP server for all of the spokes. NHRP is used by remote routers to determine the tunnel destinations for peers attached to the mGRE tunnel.

```
interface Tunnel0
 ip nhrp map multicast dynamic
 ip nhrp map 172.24.0.2 199.66.188.28
 ip nhrp network-id 1
 ip nhrp redirect
```

Procedure 4: Configure EIGRP

Step 1: Enable the EIGRP process for DMVPN.

The tunnel interface is the only EIGRP interface, and you need to explicitly list its network range. The routes for the enterprise network are redistributed into OSPF to be sent to the spokes.

```
router eigrp 1
 network 172.24.0.0
 passive-interface default
 no passive-interface Tunnel1
 redistribute ospf 1 metric 10000 10 255 1 1500 route-map ospf2eigrp
```

Procedure 5: Configure OSPF

Step 1: Configure OSPF.

OSPF is used to distribute the routes from the DMVPN network into the local enterprise routing tables.

```
router ospf 1
 redistribute static subnets route-map static2ospf
```

Step 2: Add a static route for the DMVPN network.

This should be a summary route of the DMVPN network and all of the networks that exist at the spokes.

```
ip route 172.24.0.0 255.255.0.0 Null0
```

Step 3: Define access lists for the route maps.

The route for the DMVPN network is redistributed into OSPF and the routes for the enterprise networks are distributed into EIGRP.

```
access-list 1 permit 172.24.0.0 0.0.255.255
access-list 2 permit 199.66.188.0 0.0.0.255
access-list 2 permit 172.18.0.0 0.0.0.255
```

Step 4: Configure route maps

```
route-map static2ospf permit 10
 match ip address 1
!
route-map ospf2eigrp permit 10
 match ip address 2
```

PROCESS

Configuring Spoke

1. Configure ISAKMP and IPsec
2. Configure the mGRE tunnel
3. Configure EIGRP

Procedure 1: Configure ISAKMP and IPsec

Step 1: Configure the ISAKMP policy.

```
crypto isakmp policy 10
 encr aes 256
 hash sha256
 authentication pre-share
```

Step 2: Configure the ISAKMP pre-shared key.

```
crypto isakmp key Cisco123 address 0.0.0.0
crypto isakmp keepalive 30
```

Step 3: Define the IPsec transform set.

```
crypto ipsec transform-set xform esp-aes 256 esp-sha256-hmac
 mode transport
```

Step 4: Create the IPsec profile.

```
crypto ipsec profile ipsec-prof
 set transform-set xform
```

Procedure 2: Configure the mGRE tunnel

Step 1: Configure basic interface settings.

```
interface Tunnel0
 ip address 172.24.0.5 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip tcp adjust-mss 1360
```

Step 2: Configure the tunnel.

```
interface Tunnel0
 tunnel source GigabitEthernet1
 tunnel mode gre multipoint
 tunnel key 1
 tunnel protection ipsec profile ipsec-prof
```

Step 3: Configure NHRP.

Specify the IP addresses of the hub routers below.

```
interface Tunnel0
 ip nhrp network-id 1
 ip nhrp nhs 172.24.0.1 nbma 199.66.188.27 multicast
 ip nhrp nhs 172.24.0.2 nbma 199.66.188.28 multicast
 ip nhrp shortcut
```

Procedure 3: Configure EIGRP

Step 1: Enable an EIGRP process for DMVPN.

All interfaces on the router are EIGRP interfaces, but only the DMVPN tunnel interface is non-passive. The network range must include all interface IP addresses either in a single network statement or in multiple network statements. All DMVPN spoke routers should run EIGRP stub routing to improve network stability and reduce resource utilization.

```
router eigrp 1
 network 172.24.0.0
 passive-interface default
 no passive-interface Tunnel1
 eigrp stub connected
```

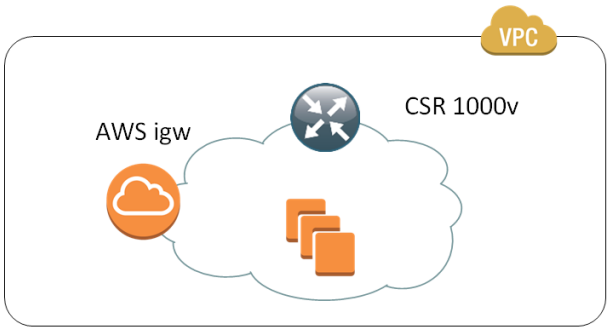
VPC Gateway Redundancy

Design Overview

Cisco CSR 1000V is software appliance version of the Cisco Aggregation Services Routers (ASR) 1000 Series. It can be used to extend advanced enterprise network and security services such as IPSec VPNs, NAT, FW, application visibility, and SLA monitoring into an AWS VPC environment.

Cisco CSR 1000v runs as an EC2 instance and is launched from the AWS marketplace. Figure 11 shows a notional view of the CSR 1000v in an AWS VPC. By using the VPC routing table, traffic from the EC2 instances will be forwarded through the CSR 1000v so that services can be applied.

Figure 11 - CSR 1000v in one-armed mode



Since the CSR 1000v runs as an EC2 instance, it can rely on native EC2 high availability mechanisms in the event of underlying compute hardware issues. In this case, the CSR would be restarted and recovery times would be on the order of minutes. For designs that require fast convergence, the CSR 1000v can be deployed in a redundant pair with failover between them.

In typical Ethernet environments, gateway redundancy is provided by protocols such as hot standby router protocol (HSRP) and virtual router redundancy protocol (VRRP). These protocols present a pair of routers as a single virtual IP address that can be used by hosts as their default gateway. HSRP and VRRP use link local multicast packets for peer status monitoring and active gateway selection.

In an AWS VPC environment, link local multicast and broadcast traffic are not supported. This section will discuss an alternate gateway redundancy option for the CSR 1000v when used in an AWS VPC.

In this design, you:

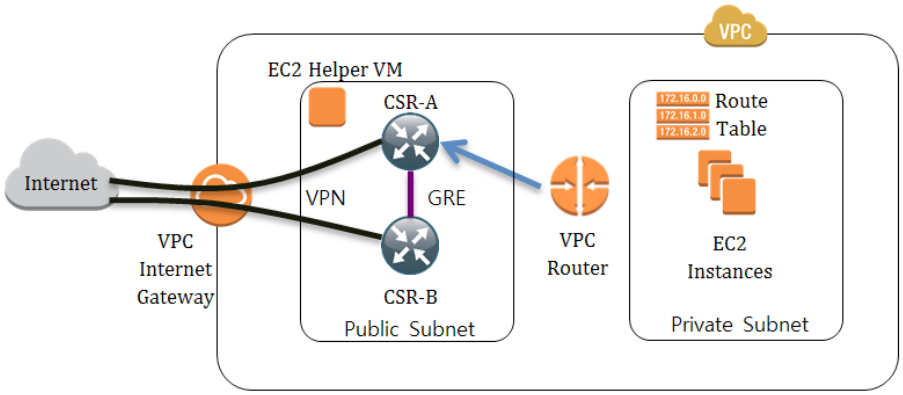
- Deploy a pair of CSR 1000vs into an AWS VPC.
- Configure a GRE tunnel between the CSRs.
- Enable bi-directional forwarding detection (BFD) and a routing protocol such as EIGRP or BGP on the GRE to do peer failure detection.
- Configure each CSR 1000v with a embedded event manager (EEM) applet that will monitor for BFD peer down events.

  When a BFD peer down event is detected, the EEM applet will use the AWS EC2 VPC API to modify the VPC route table to redirect traffic around the failure.

Deployment Details

The topology in Figure 10 is an example of a VPN gateway configuration.

Figure 12 – Example VPN gateway configuration



This topology uses a single availability zone and two VPC subnets. Each CSR has a single Ethernet interface that is connected to the public VPC subnet. This public subnet has a VPC route table with a default route target of the Internet gateway. Each CSR has a VPN tunnel to Internet. These tunnels would typically terminate at another VPN device located on the enterprise network. Finally, a GRE tunnel is configured between the local CSRs. This GRE tunnels allows the CSRs to exchange BFD control packets that are used for peer failure detection.

Tech Tip

Configuration of BFD requires a premium license.

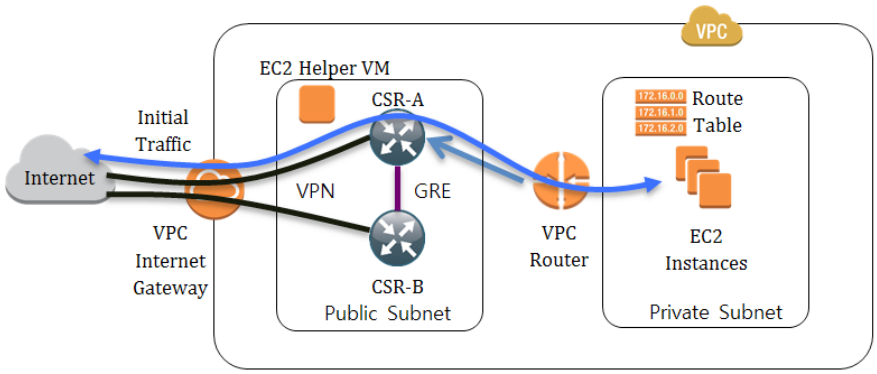
Since the CSR is not directly connected to the private subnet, a static route for the private subnet is added to each CSR. This static route points the address of the VPC router on the public subnet. This address will always be the first usable address of a subnet. For example, the VPC router address for the subnet 172.24.2.0/25 will be 172.24.2.1.

Other topologies, including multiple availability zones, single or multi subnet VPCs, multiple VPN tunnels, and multiple CSR Ethernet interfaces, are possible and would be applicable to this solution.

EIGRP is used as the routing protocol, but other routing protocol could be used. The primary purpose of the routing protocol is to register as a BFD client. BFD requires at least one client protocol before it will initiate neighbor discovery. An additional benefit of the GRE tunnel and the routing protocol is that they can be used to establish a back-up path in case of VPN tunnel failures.

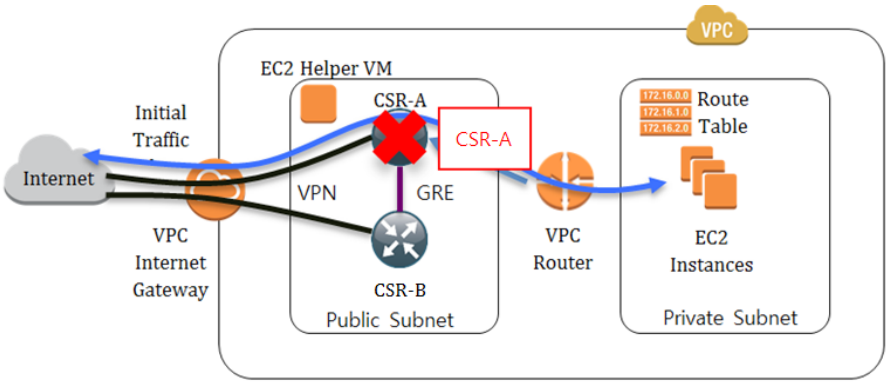
The EC2 instances reside in a private subnet with its own VPC route table. The default route for this subnet will have a target of the network interface of one of the CSRs. Because the VPC route table only allows for one active target per route, only one CSR is in the egress traffic path for this subnet. Ingress traffic flow over the VPN tunnels is determined by the remote VPN devices, so it is possible that CSR-B is the active ingress path or that load sharing is being done between CSR-A and CSR-B. In this example, ingress and egress traffic is initially being forwarded through CSR-A, as shown in Figure 13.

Figure 13 - Initial traffic flow



CSR-A then fails, as shown in Figure 14. The goal is to shift traffic so that it will egress through CSR-B and no longer ingress through CSR-A.

Figure 14 - CSR-A failure



For the ingress traffic flow, the remote VPN device will need to detect that the VPN tunnel terminated at CSR-A is no longer available. This can be done using traditional VPN tunnel high availability techniques such as routing protocols (with or with out BFD) and Internet key exchange (IKE) dead peer detection.

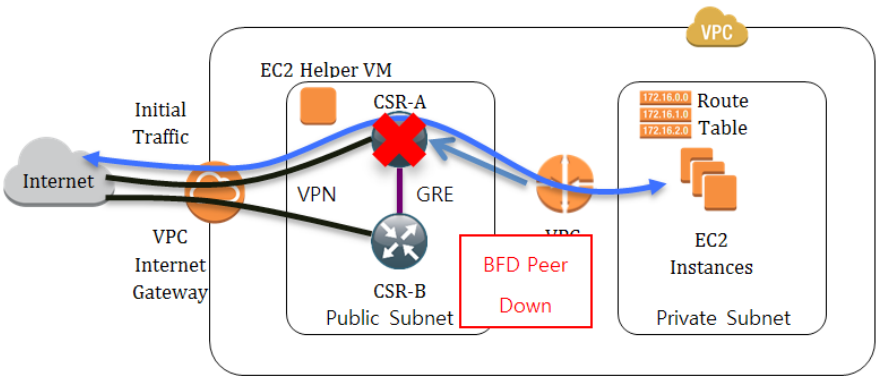
For the egress traffic direction, CSR-B will detect the failure of CSR-A and modify the VPC route table to redirect traffic to CSR-B. This is accomplished using BFD over the GRE tunnel and an EEM applet.

When CSR-A fails, BFD will timeout, generating a log message on CSR-B. An example BFD peer down message when using EIGRP is as follows:

```
%DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 172.24.33.1 (Tunnel33) is down: BFD peer down notified
```

EEM is an event detection and automation technology available the CSR. In this scenario, an EEM applet is configured to run whenever the BFD peer down log message is generated.

Figure 15 - EEM applet triggered with BFD peer down



When triggered, this EEM applet will use the AWS API EC2 replace-route command to modify the VPC route table to make itself the new target for the default route.

The CSR cannot access the AWS EC2 API directly. This requires use of a helper VM with the AWS EC2 CLI tools installed. CSR-B will SSH into the helper EC2 VM and run the ec2-replace-route command.

Tech Tip

Restrict access of the VM to the local subnet in order to avoid compromise.

Amazon Linux EC2 instances have the CLI tools installed by default. Other Linux distributions will require manual installation. Because the CSR is using SSH, it will require Linux account credentials to log into the VM.

Tech Tip

By default, Amazon Linux disables password authentication, but it can be enabled by editing the /etc/ssh/sshd_config file to include a line with **PasswordAuthentication yes**

Finally, the AWS CLI tools will need to be configured with AWS account credentials in order to provide the necessary privileges for the ec2-replace-route command. For a link to the AWS documentation on setting up the AWS EC2 CLI tools, see Appendix D.

Figure 16 - EEM Applet used in Figure 13

```
event manager environment q "

event manager environment USER eem

event manager environment PASS cisco123

event manager environment IP 172.24.2.84

event manager environment RTB rtb-c41b78a5

event manager environment CIDR 0.0.0.0/0

event manager environment ENI eni-65ef154e

event manager applet replace-route
```

To promote the reusability of this applet, local variables are separated out of the body of the EEM applet and are defined as EEM environment variables. The variables used are as follows:

- **q**–used to substitute a quotation mark into the ssh command
- **USER**–Linux user account of the helper VM
- **PASS**–Linux user password of the helper VM
- **IP**–IP address of the helper VM
- **RTB**–the route table ID for the private subnet VPC route table
- **CIDR**–destination value for the default route
- **ENI**–network interface ID of the CSR gigabit interface

The route table ID and network interface ID can be found using the AWS console.

Figure 17 – Route table ID in the AWS console

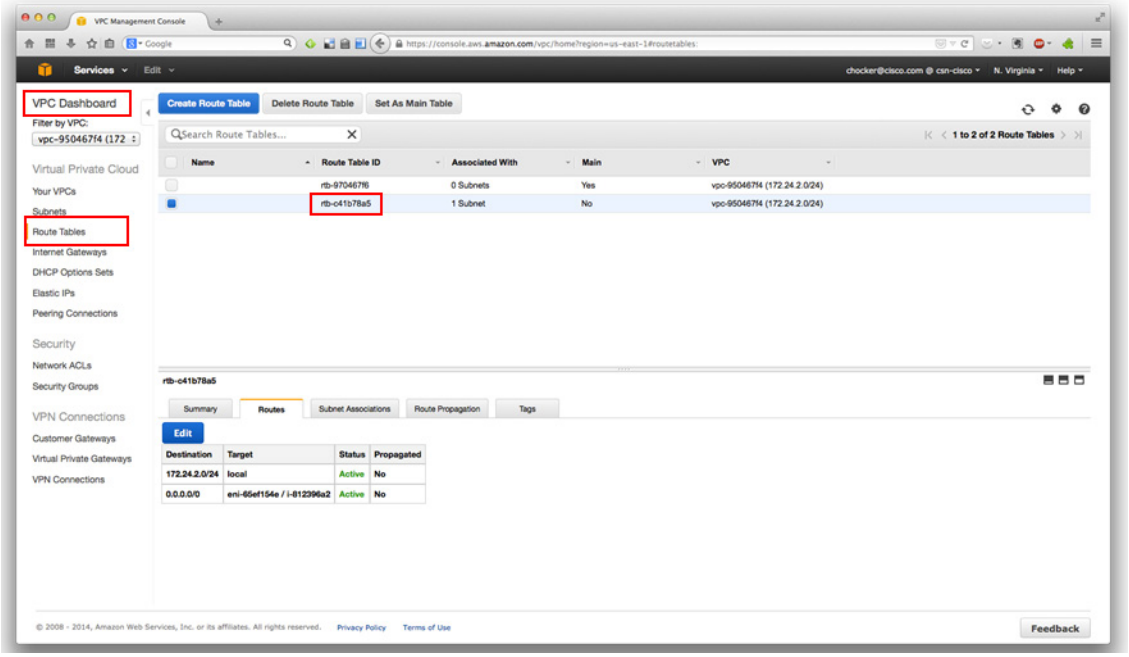
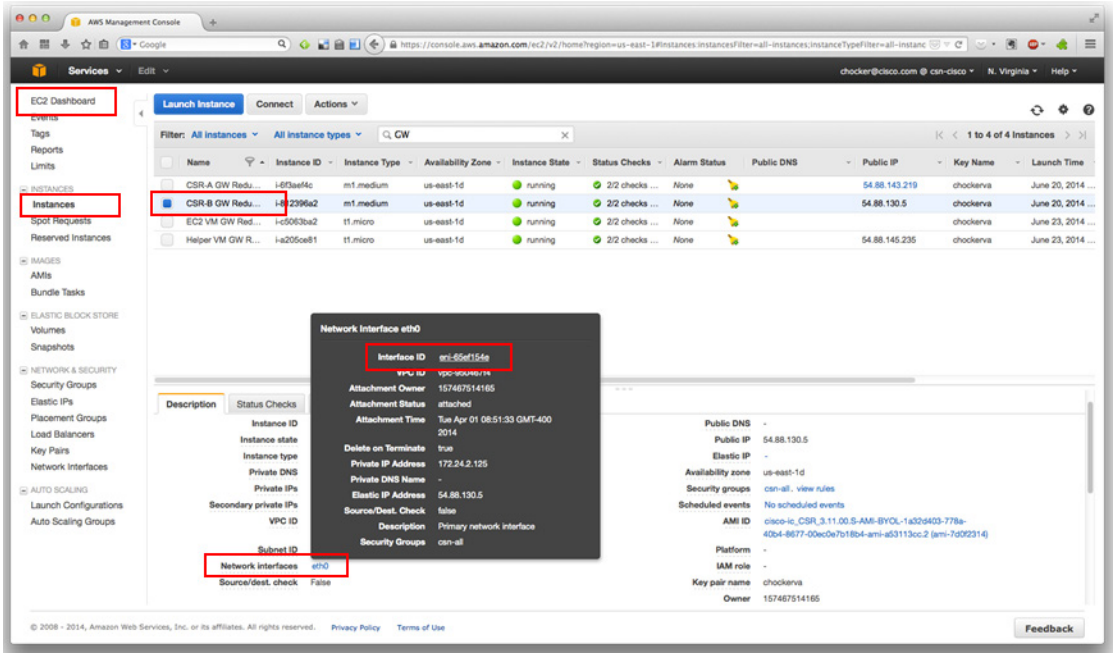


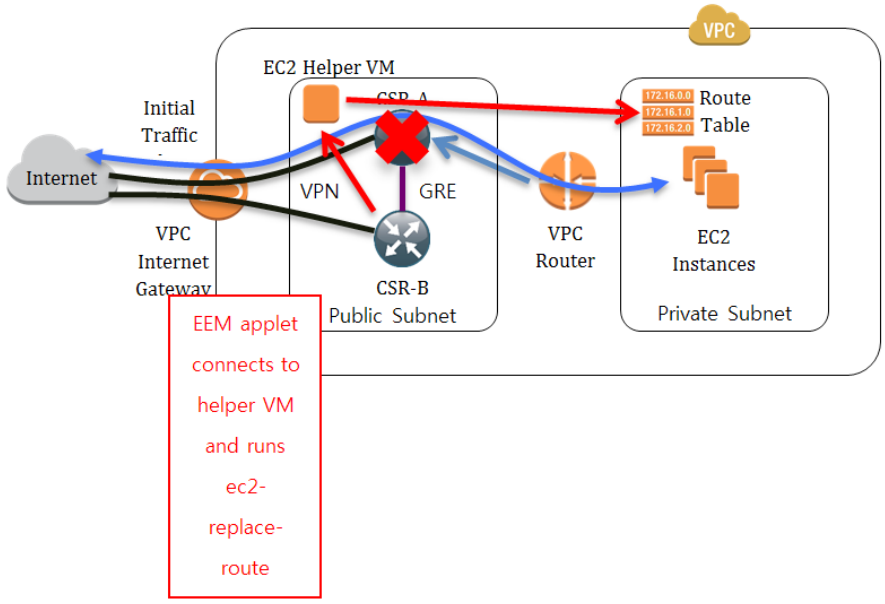
Figure 18 – Network interface ID for CSR-B in the AWS console



The EEM configuration on CSR-A and CSR-B will be nearly the same. The only required difference would the ENI environment variable, which should be set to the network interface ID of the local CSR.

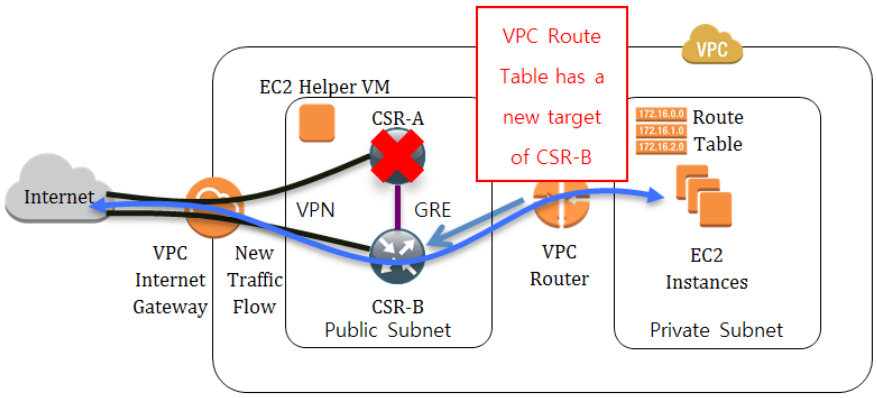
Figure 19 shows CSR-B modifying the VPC Route table for the default route.

Figure 19 - EEM applet modifying the VPC route table



Once the VPC route table is modified, the VPC will begin directing egress traffic to the CSR-B, as shown in Figure 20.

Figure 20 - New traffic flow



Appendix A: Product List

The following products and software versions have been validated.

Functional Area	Product	Part Numbers	Software Version
Cloud Router	Cisco® Cloud Services Router 1000V	N/A	Cisco IOS XE Software Release 3.11

Appendix B: Technical Feature Supplement

PROCESS

Adding Supplemental Features

1. Configure NAT
2. Configure zone-based firewall
3. Secure public interface
4. Configure IPSLA

Procedure 1: Configure NAT

NAT gives the inside AWS instances direct access to the Internet using the elastic IP address of the Cisco CSR 1000V Series Router. Because the outside interface of the CSR 1000V is not assigned the elastic IP address directly, a second NAT is done from the AWS internal address to the actual elastic IP address.

Step 1: Configure NAT on interfaces.

```
interface GigabitEthernet1
 ip nat outside
!
interface GigabitEthernet2
 ip nat inside
```

Step 2: Configure addresses to NAT.

```
ip access-list standard nat
 permit 172.24.2.0 0.0.0.255
```

Step 3: Configure NAT.

```
ip nat inside source list nat interface GigabitEthernet1 overload
```

The Cisco CSR 1000V can also perform NAT port translation to allow direct access of services through protocols such as HTTP. Providing direct access to the AWS hosted instances allows offloading of bandwidth onto the cloud service provider when central inspection is not required. In the following configuration, 172.24.2.17 is the internal AWS IP address allocated to the outside interface of the CSR 1000V and 172.24.2.200 is the internal AWS IP address of the router providing service on port 80.

Tech Tip

Remember to open the ports within the Amazon Security Group for the static NAT entries.

Tech Tip

Port 22 is used for CSR for access and should not be NATed.

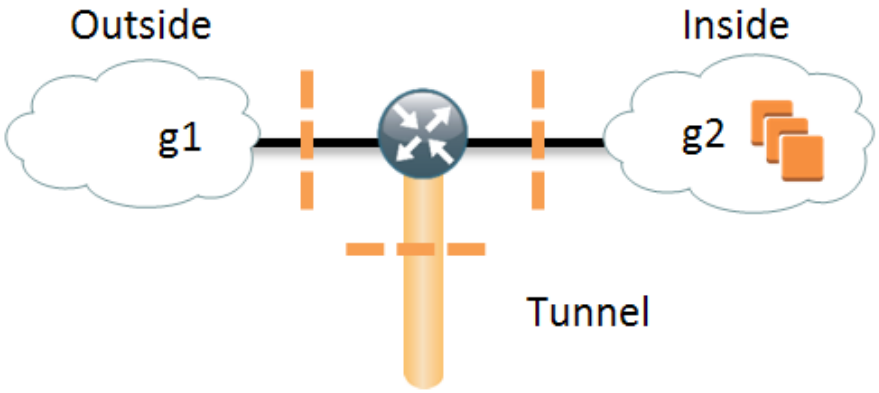
Step 4: Optionally, configure 1:1 NAT.

```
ip nat inside source static tcp 172.24.2.200 80 172.24.2.17 80 extendable
```

Procedure 2: Configure zone-based firewall

When directly accessing services in the cloud service provider or when more granular security is needed, you can configure ZBFWs on the Cisco CSR 1000v in order to extend the enterprise security policy to the AWS VPC. The following configuration defines three zones: inside, outside, and tunnel. Protocol inspection is used to inspect and allow traffic between zones. An ACL is used to define ports for which protocol inspection is not available. Because there is no need for traffic to flow below the tunnel and the outside interface, it is not allowed.

Figure 21 - Deploying multiple security zones with CSR



Step 1: Configure class maps.

```
class-map type inspect match-any inside-tunnel
 match protocol tcp
 match protocol udp
 match protocol icmp
class-map type inspect match-any tunnel-inside
 match protocol icmp
 match protocol http
 match protocol https
```

```
 match protocol ssh
 match access-group name tunnel-inside
class-map type inspect match-any inside-outside
 match protocol tcp
 match protocol udp
 match protocol icmp
class-map type inspect match-any outside-inside
 match protocol http
 match protocol https
 match access-group name outside-inside
```

Step 2: Configure policy maps.

```
policy-map type inspect inside-tunnel
 class type inspect inside-tunnel
  inspect
 class class-default
  drop log
policy-map type inspect outside-inside
 class type inspect outside-inside
  inspect
 class class-default
  drop log
policy-map type inspect inside-outside
 class type inspect inside-outside
  inspect
 class class-default
  drop log
policy-map type inspect tunnel-inside
 class type inspect tunnel-inside
  inspect
 class class-default
  drop log
```

Step 3: Configure security zones.

```
zone security outside
zone security inside
zone security tunnel
```

Step 4: Configure zone pairs.

```
zone-pair security inside-outside source inside destination outside
 service-policy type inspect inside-outside
zone-pair security inside-tunnel source inside destination tunnel
 service-policy type inspect inside-tunnel
zone-pair security outside-inside source outside destination inside
 service-policy type inspect outside-inside
zone-pair security tunnel-inside source tunnel destination inside
 service-policy type inspect tunnel-inside
```

Step 5: Configure interfaces in the security zones.

```
interface Tunnel0
zone-member security tunnel
!
interface GigabitEthernet1
zone-member security outside
!
interface GigabitEthernet2
zone-member security inside
```

Step 6: Add IP access lists.

```
ip access-list extended outside-inside
ip access-list extended tunnel-inside
 permit tcp any host 172.24.2.200 eq 3389
```

Procedure 3: Secure public interface

ACLs can be used to protect the router from outside traffic. The following ACL prevents all traffic except what is required to remotely manage the router, create the tunnel, and perform DHCP on the outside interface.

Step 1: Create an access list.

```
ip access-list extended internet
 permit esp any any
 permit udp any eq isakmp any
 permit udp any any eq isakmp
 permit udp any eq non500-isakmp any
 permit udp any any eq non500-isakmp
 permit tcp any any eq 22
 permit tcp any eq 22 any
 permit udp any eq bootps any eq bootpc
 permit udp any eq bootpc any eq bootps
```

Step 2: Apply Access List to Interface

```
interface GigabitEthernet1
 ip access-group internet in
 ip access-group internet out
```

Tech Tip

You can further limit SSH access by applying a vty access class. If the Gig1 interface is in a VRF path, be sure to use the vrf-also command option with the access-class command (access-class 34 in vrf-also).

Tech Tip

Policy must be reconciled between interface ACLs and ZBFWs when both are used simultaneously.

Procedure 4: Configure IPSLA

IP SLA is used tool to generate synthetic traffic to gather network performance metrics such as delay and loss. The following example will monitor the two spokes from the hub.

Step 1: Configure IP SLA on the hub.

```
ip sla 1
 icmp-echo 172.24.0.5 source-ip 172.24.0.5
 tag DMVPN_SLA
ip sla 2
 icmp-echo 172.24.0.1 source-ip 172.24.0.6
 tag DMVPN_SLA
```

Step 2: Create a schedule for IP SLA on the hub.

```
ip sla group schedule 1 1-2 schedule-period 60 frequency 60 start-time now
life forever
```

Step 3: Create a IP SLA responder on the spokes.

```
ip sla responder
```

Step 4: Show the IP SLA statistics collected.

```
show ip sla statistics
```

Appendix C: Sample Configuration for VPC Gateway Redundancy

CSR-A

```
version 15.4
service timestamps debug datetime msec
service timestamps log datetime msec
no platform punt-keepalive disable-kernel-core
platform console virtual
!
hostname CSR-A
!
boot-start-marker
boot-end-marker
!
aaa new-model
!
aaa authentication login default local
aaa authorization exec default local
!
aaa session-id common
!
no ip domain lookup
!
subscriber templating
!
multilink bundle-name authenticated
!
crypto pki trustpoint TP-self-signed-208042347
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-208042347
 revocation-check none
 rsakeypair TP-self-signed-208042347
!
crypto pki certificate chain TP-self-signed-208042347
 certificate self-signed 01
  30820229 30820192 A0030201 02020101 300D0609 2A864886 F70D0101 05050030
  30312E30 2C060355 04031325 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 32303830 34323334 37301E17 0D313430 33313831 34343234
  395A170D 32303031 30313030 30303030 5A303031 2E302C06 03550403 1325494F
  532D5365 6C662D53 69676E65 642D4365 72746966 69636174 652D3230 38303432
```

```
  33343730 819F300D 06092A86 4886F70D 01010105 0003818D 00308189 02818100
  BDDDA641 7A750902 23B63746 2D3DACFC 776F89A2 F4960F6B 19673309 36AEF620
  A61FFBC8 2984312C 03160B6F 887AB254 49063821 2E8FA3DD 9B9622D4 687D871F
  F03D35F3 790723E4 0892424C 441CD535 4A457E02 25EA16E2 68A9064B 0874896E
  5CD52617 6B28B26C 07EE4B5D 020F2964 5234EB55 38EB2175 02D129E0 30B17A81
  02030100 01A35330 51300F06 03551D13 0101FF04 05300301 01FF301F 0603551D
  23041830 168014E6 B8C2B043 C691E45B 287D1A0A 30DD905B C1A77F30 1D060355
  1D0E0416 0414E6B8 C2B043C6 91E45B28 7D1A0A30 DD905BC1 A77F300D 06092A86
  4886F70D 01010505 00038181 0058C096 57A0D0C9 B28C8EAA BEA74B84 53BA2062
  12B64621 60D348F8 FB152E52 565F0CCA 77430C45 55458D06 6190D7E7 6DAB65D9
  B92CD045 9119BA29 6B295BB9 5128CEE9 0EC6AD18 35C3D8AC 54563CE9 62D04947
  8B9B31CF 56308CE3 19BEFC95 795121C5 44673211 B9DE5B9A 0AFB687F 1D33979F
  4191CE4B 3E9CB684 272BCD98 F5
   quit
license udi pid CSR1000V sn 9F4TK27JDDU
license boot level premium
spanning-tree extend system-id
!
username ec2-user privilege 15 secret 5 $1$MIDl$ZZmKyk5rWdQ/UdrGS0v/N.
!
redundancy
 mode none
!
crypto ikev2 profile default
 match identity remote fqdn domain cisco.com
 identity local fqdn csn-aws-va-CSR5.cisco.com
 authentication remote pre-share key cisco123
 authentication local pre-share key cisco123
!
crypto ikev2 dpd 10 2 on-demand
!
ip ssh rsa keypair-name ssh-key
ip ssh version 2
ip ssh pubkey-chain
  username ec2-user
   key-hash ssh-rsa 1CE65034F2481508E0466998CE6C8AB2 chockerva
!
interface Tunnel33
 ip address 172.24.33.1 255.255.255.252
 bfd interval 500 min_rx 500 multiplier 3
 tunnel source GigabitEthernet1
 tunnel destination 172.24.2.125
!
interface Tunnel98
 ip address 172.24.98.2 255.255.255.252
 ip summary-address eigrp 1 172.24.2.0 255.255.255.0
 bfd interval 500 min_rx 500 multiplier 3
```

```
 tunnel source GigabitEthernet1
 tunnel destination 54.200.135.205
 tunnel protection ipsec profile default
!
interface VirtualPortGroup0
 ip unnumbered GigabitEthernet1
 no mop enabled
 no mop sysid
!
interface GigabitEthernet1
 ip address dhcp
 negotiation auto
!
router eigrp 1
 bfd interface Tunnel98
 bfd interface Tunnel33
 network 172.24.0.0
 passive-interface GigabitEthernet1
!
!
virtual-service CSR_mgmt
 activate
!
ip forward-protocol nd
!
no ip http server
ip http secure-server
ip route 172.24.2.128 255.255.255.128 172.24.2.1
!
control-plane
!
line con 0
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 transport input ssh
!
event manager environment q "
event manager environment USER chocker
event manager environment PASS IamD1B4U
event manager environment IP 172.24.2.84
event manager environment RTB rtb-c41b78a5
event manager environment CIDR 0.0.0.0/0
event manager environment ENI eni-060ce72d
event manager applet replace-route
 event syslog pattern "\(Tunnel33\) is down: BFD peer down notified"
```

```
 action 1.0 cli command "enable"
 action 2.0 cli command "ssh -l $USER $IP $q ec2-replace-route $RTB -r $CIDR -n
$ENI$q" pattern "word:"
 action 2.1 cli command "$PASS"
!
end
```

CSR-B

```
version 15.4
service timestamps debug datetime msec
service timestamps log datetime msec
no platform punt-keepalive disable-kernel-core
platform console virtual
!
hostname CSR-B
!
boot-start-marker
boot-end-marker
!
aaa new-model
!
aaa authentication login default local
aaa authorization exec default local
!
aaa session-id common
!
subscriber templating
!
multilink bundle-name authenticated
!
crypto pki trustpoint TP-self-signed-3088625601
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-3088625601
 revocation-check none
 rsakeypair TP-self-signed-3088625601
!
crypto pki certificate chain TP-self-signed-3088625601
 certificate self-signed 01
  3082022B 30820194 A0030201 02020101 300D0609 2A864886 F70D0101 05050030
  31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303838 36323536 3031301E 170D3134 30343031 31333033
  34375A17 0D323030 31303130 30303030 305A3031 312F302D 06035504 03132649
  4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D33 30383836
  32353630 3130819F 300D0609 2A864886 F70D0101 01050003 818D0030 81890281
  8100C354 8092363B B6FAEDA3 C86D3E6D 098BE68E 816A817B 91E11086 284F01EE
  71252DBC 6EBC5498 ACDB7CD2 7EA49F68 7FFCDEC1 5E3B0C7B 1802431F CD0EC583
```

```
  1D8A1636 DA0DBB46 3D57587A FCA519AE 75054641 96AB1491 EE23A624 E95D442D
  BCCC7890 7B2AA21B 1CFD9195 A3787271 A2BBDA0F 316C1497 9D889531 58FDABE4
  FEDD0203 010001A3 53305130 0F060355 1D130101 FF040530 030101FF 301F0603
  551D2304 18301680 14BC4913 F3712915 87A396C3 752F2472 FF5CA2DF BB301D06
  03551D0E 04160414 BC4913F3 71291587 A396C375 2F2472FF 5CA2DFBB 300D0609
  2A864886 F70D0101 05050003 81810061 F3FD3584 A5BA99FD 51C0689E EBF557F6
  D5AC4BD6 D6975B79 DEB139E3 2E182087 C1C9839A DBF7AEA3 4CBA3632 41D8CFE2
  BEFDBE98 8292814D C322A153 150C8787 FD40BAB8 8E4BBF9D 642733B4 B1EEB0CD
  50A6EBFE D3A91922 494CB001 F34BFE6F BE906F82 ED2BED87 AA6B41E6 444943F5
  1A824738 610DF594 61EF842C 0D3C9D
   quit
license udi pid CSR1000V sn 9MZ1BE4UHG2
license boot level premium
spanning-tree extend system-id
!
username ec2-user privilege 15 secret 5 $1$sg9o$.4qkVnSQJSB4V/Onto.Si0
username csn-admin privilege 15 password 7 0230590F44551F287E1D
!
redundancy
 mode none
!
crypto ikev2 profile default
 match identity remote fqdn domain cisco.com
 identity local fqdn csn-aws-va-CSR8.cisco.com
 authentication remote pre-share key cisco123
 authentication local pre-share key cisco123
!
crypto ikev2 dpd 10 2 on-demand
!
ip ssh rsa keypair-name ssh-key
ip ssh version 2
ip ssh pubkey-chain
  username ec2-user
   key-hash ssh-rsa 1CE65034F2481508E0466998CE6C8AB2 chockerva
!
interface Tunnel33
 ip address 172.24.33.2 255.255.255.252
 bfd interval 500 min_rx 500 multiplier 3
 tunnel source GigabitEthernet1
 tunnel destination 172.24.2.126
!
interface Tunnel96
 ip address 172.24.96.1 255.255.255.252
 ip summary-address eigrp 1 172.24.2.0 255.255.255.0
 bfd interval 500 min_rx 500 multiplier 3
 tunnel source GigabitEthernet1
 tunnel destination 54.200.135.205
```

```
 tunnel protection ipsec profile default
!
interface VirtualPortGroup0
 ip unnumbered GigabitEthernet1
 no mop enabled
 no mop sysid
!
interface GigabitEthernet1
 ip address dhcp
 negotiation auto
!
router eigrp 1
 bfd interface Tunnel96
 bfd interface Tunnel33
 network 172.24.0.0
 passive-interface GigabitEthernet1
!
virtual-service CSR_mgmt
 activate
!
ip forward-protocol nd
!
no ip http server
ip http secure-server
ip route 172.24.2.128 255.255.255.128 172.24.2.1
!
control-plane
!
line con 0
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 transport input ssh
!
event manager environment q "
event manager environment USER chocker
event manager environment PASS IamD1B4U
event manager environment IP 172.24.2.84
event manager environment RTB rtb-c41b78a5
event manager environment CIDR 0.0.0.0/0
event manager environment ENI eni-65ef154e
event manager applet replace-route
 event syslog pattern "\(Tunnel33\) is down: BFD peer down notified"
 action 1.0 cli command "enable"
 action 2.0 cli command "ssh -l $USER $IP $q ec2-replace-route $RTB -r $CIDR -n
$ENI$q" pattern "word:"
```

```
 action 2.1 cli command "$PASS"
!
end
```

Appendix D: References

External Cisco CSR 1000v Product Page:

http://www.cisco.com/go/cloudrouter/

Cisco CSR 1000v for AWS Deployment Guide:

https://supportforums.cisco.com/sites/default/files/CSRaws_final_0.pdf

Cisco CSR 1000v for AWS Documentation:

http://www.cisco.com/c/en/us/td/docs/routers/CSR1000/software/aws/CSRaws.html

Cisco CSR 1000v for AWS Community Forum:

https://supportforums.cisco.com/community/CSR-amazon

Embedded Event Manager Configuration Guide:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/eem/configuration/xe-3s/asr1000/eem-xe-3s-asr1000-book.html

Bidirectional Forwarding Detection Configuration Guide:

http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_bfd/configuration/xe-3s/asr1000/irb-xe-3s-asr1000-book.html

AWS EC2 CLI Documentation:

http://docs.aws.amazon.com/AWSEC2/latest/CommandLineReference/Welcome.html

Feedback

Please use the feedback form to send comments and suggestions about this guide.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

B-0000AWS14-1 08/14