# Amazon Virtual Private Cloud

## Network Administrator Guide

## API Version 2011-01-01

# Amazon Virtual Private Cloud: Network Administrator Guide

# Table of Contents

# Welcome

Welcome to the *Amazon Virtual Private Cloud Network Administrator Guide*. This guide is only for Amazon VPC customers who plan to use an IPsec hardware VPN gateway with their VPC. The guide helps you configure your Amazon VPC customer gateway, which is the VPN device on your side of the VPN connection.

The VPN connection lets you bridge your VPC and IT infrastructure, and extend your existing security and management policies to your VPC instances as if they were running within your infrastructure.

# The Amazon VPC Customer Gateway

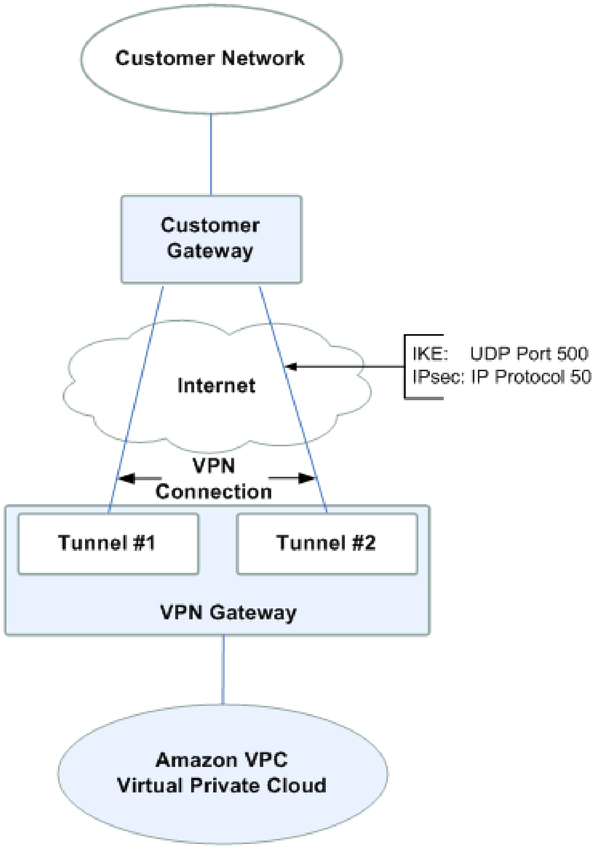
**Topics**

## Your Role

Throughout this guide, we refer to your company's *integration team*, which is the person (or persons) at your company working to integrate your infrastructure with Amazon VPC. This team might consist of just you, or might not include you at all, depending on how your company allocates network engineering resources. The important thing to know is that someone at your company must use must use the AWS Management Console to get the information you need to configure your customer gateway, and someone must actually configure the customer gateway. Your company might have a separate team for each task (an *integration team* that uses the AWS Management Console, and a separate network engineering group that has access to network devices and who configures the customer gateway). Or your company might have a single person who does both tasks, or some other arrangement entirely. This guide assumes the first scenario, and that you're someone in the network engineering group who will receive information from your company's integration team so you can then configure the customer gateway device.

## What Is a Customer Gateway?

Your company has decided to use an optional Amazon VPC *VPN connection* that links your data center (or network, etc.) to your Amazon VPC *virtual private cloud* (VPC). A *customer gateway* is the anchor on your side of that connection. It can be a physical or software appliance. The anchor on the AWS side of the VPN connection is called a *VPN gateway*. The following diagram shows your network, the customer gateway, the VPN connection that goes to the VPN gateway, and the VPC. There are two lines between the customer gateway and VPN gateway because the VPN connection consists of *two tunnels*. We chose this design to provide increased availability for the Amazon VPC service. If there's a device failure within AWS, your VPN connection will automatically fail over to the second tunnel so your access isn't interrupted. When you configure your customer gateway, it's important you configure *both* tunnels.



# Summary of What You Need to Do

The overall process of setting up the VPN connection is covered in the Amazon Virtual Private Cloud User Guide. One task in the process is to configure the customer gateway. The following table summarizes what you need to do to configure the customer gateway.

### Process for Configuring the Customer Gateway

| | |
|---|---|
| 1 | Designate an appliance to act as your Amazon VPC customer gateway (for more information, see Customer Gateway Devices We've Tested (p. 4) and Requirements for Your Customer Gateway (p. 5)). |

| | |
|---|---|
| 2 | Determine the following information about the customer gateway:<br>• The vendor (e.g., Cisco Systems), the platform (e.g., ISR Series Routers), and the software version (e.g., IOS 12.4)<br>• The Internet-routable IP address for the external interface (the address must be static and can't be behind a device performing network address translation (NAT))<br>**Note**<br>We assume the BGP ASN for the customer gateway is 65000. |
| 3 | Give the preceding information to your integration team, who will create your VPN connection using the AWS Management Console.The team will obtain information you need in order to configure the customer gateway. |
| 4 | Get this configuration information from the team. |
| 5 | Configure your customer gateway using the information from the team. |
| 6 | Notify your integration team when you're done configuring the customer gateway so they can move forward with their work. |

# Four Main Parts to Configuration

There are four main parts to the configuration of your customer gateway. Throughout this guide, we use a special symbol for each of these parts to help you understand what you need to do. The following table shows the four parts and the corresponding symbols.

| | |
|---|---|
| IKE | IKE Security Association (required to exchange keys used to establish the IPsec security association) |
| IPsec | IPsec Security Association (which handles the tunnel's encryption, authentication, etc.) |
| Tunnel | Tunnel interface (which receives traffic going to and from the tunnel) |
| BGP | BGP peering (which exchanges routes between the customer gateway and the VPN gateway) |

# Customer Gateway Devices We've Tested

Your customer gateway can be a physical or software appliance.

Following are the devices we've confirmed for customer gateways with Amazon VPC:

- Cisco Integrated Services routers running Cisco IOS 12.4 (or later) software
- Juniper J-Series routers running JunOS 9.5 (or later) software
- Juniper SSG running ScreenOS 6.1, or 6.2 (or later) software
- Juniper ISG running ScreenOS 6.1, or 6.2 (or later) software

This guide presents information about how to configure these two types of devices. If you have one of the preceding devices, but configure it for IPsec in a different way than presented in this guide, feel free to alter our suggested configuration to match your particular standards.

# Requirements for Your Customer Gateway

If you have a device that isn't in the preceding list of tested devices, this section describes the requirements the device must meet for you to use it with Amazon VPC. The following table lists the requirement the customer gateway must adhere to, the related RFC (for reference), and comments about the requirement. For an example of the configuration information if your device isn't one of the tested Cisco or Juniper devices, see Example: Generic Device (p. 40).

To provide context for the following requirements, think of each VPN connection as consisting of two separate tunnels. Each tunnel contains an IKE Security Association, an IPsec Security Association, and a BGP Peering.

| Requirement | RFC | Comments |
|---|---|---|
| Establish IKE Security Association using Pre-Shared Keys<br>IKE | RFC 2409 | The IKE Security Association is established first between the VPN gateway and customer gateway using the Pre-Shared Key as the authenticator. Upon establishment, IKE negotiates an ephemeral key to secure future IKE messages. Proper establishment of an IKE Security Association requires complete agreement among the parameters, including encryption and authentication parameters. |
| Establish IPsec Security Associations in Tunnel mode<br>IPsec | RFC 4301 | Using the IKE ephemeral key, keys are established between the VPN gateway and customer gateway to form an IPsec Security Association (SA). Traffic between gateways is encrypted and decrypted using this SA. The ephemeral keys used to encrypt traffic within the IPsec SA are automatically rotated by IKE on a regular basis to ensure confidentiality of communications. |
| Utilize the AES 128-bit encryption function | RFC 3602 | This encryption function is used to ensure privacy among both IKE and IPsec Security Associations. |
| Utilize the SHA-1 hashing function | RFC 2404 | This hashing function is used to authenticate both IKE and IPsec Security Associations. |
| Utilize Diffie-Hellman Perfect Forward Secrecy in "Group 2" mode | RFC 2409 | IKE uses Diffie-Hellman to establish ephemeral keys to secure all communication between customer gateways and VPN gateways. |
| Utilize IPsec Dead Peer Detection | RFC 3706 | The use of Dead Peer Detection enables the VPN devices to rapidly identify when a network condition prevents delivery of packets across the Internet. When this occurs, the gateways delete the Security Associations and attempt to create new associations. During this process, the alternate IPsec tunnel is utilized if possible. |

| Requirement | RFC | Comments |
|---|---|---|
| Bind tunnel to logical interface (route-based VPN)<br>Tunnel | None | Your gateway must support the ability to bind the IPsec tunnel to a logical interface. The logical interface contains an IP address used to establish BGP peering to the VPN gateway. This logical interface should perform no additional encapsulation (e.g., GRE, IP in IP). |
| Fragment IP packets before encryption | RFC 4459 | When packets are too large to be transmitted, they must be fragmented. We will not reassemble fragmented encrypted packets. Therefore, your VPN device must fragment packets *before* encapsulating with the VPN headers. The fragments are individually transmitted to the remote host, which reassembles them. For more information about fragmentation, go to the Wikipedia article on IP fragmentation. |
| Establish Border Gateway Protocol (BGP) peerings<br>BGP | RFC 4271 | BGP is used to exchange routes between the customer gateway and VPN gateway. All BGP traffic is encrypted and transmitted via the IPsec Security Association. BGP is required for both gateways to exchange the IP prefixes reachable via the IPsec SA. |

We recommend you use the techniques listed in the following table to minimize problems related to the amount of data that can be transmitted through the IPsec tunnel. Because the connection encapsulates packets with additional network headers (including IPsec), the amount of data that can be transmitted in a single packet is reduced.

| Technique | RFC | Comments |
|---|---|---|
| Adjust the maximum segment size of TCP packets entering the VPN tunnel | RFC 4459 | TCP packets are often the most prevalent type of packet across IPsec tunnels. Some gateways have the ability to change the TCP Maximum Segment Size parameter. This causes the TCP endpoints (clients, servers) to reduce the amount of data sent with each packet. This is an ideal approach, as the packets arriving at the VPN devices are small enough to be encapsulated and transmitted. |
| Reset the "Don't Fragment" flag on packets | RFC 791 | Some packets carry a flag, known as the *Don't Fragment (DF) flag*, that indicates that the packet should not be fragmented. If the packets carry the flag, the gateways generate an ICMP Path MTU Exceeded message. In some cases, applications do not contain adequate mechanisms for processing these ICMP messages and reducing the amount of data transmitted in each packet. Some VPN devices have the ability to override the DF flag and fragment packets unconditionally as required. If your customer gateway has this ability, we recommend you use it as appropriate. |

# If You Have a Firewall Between the Internet and Gateway

To use this service, you must have an Internet-routable IP address to use as the endpoint for the IPsec tunnels connecting your customer gateway to the VPN gateway. If a firewall is in place between the Internet and your gateway, the rules in the following tables must be in place to establish the IPsec tunnels. The VPN gateway addresses are in the configuration information that you'll get from the integration team.

### Inbound (from the Internet)

| | |
|---|---|
| Input Rule I1 | Source IP: VPN Gateway 1, Dest IP: Customer Gateway, Protocol: UDP, Source Port: 500, Destination Port: 500 |
| Input Rule I2 | Source IP: VPN Gateway 2, Dest IP: Customer Gateway, Protocol: UDP, Source Port: 500, Destination Port: 500 |
| Input Rule I3 | Source IP: VPN Gateway 1, Dest IP: Customer Gateway, Protocol: IP 50 (ESP) |
| Input Rule I4 | Source IP: VPN Gateway 2, Dest IP: Customer Gateway, Protocol: IP 50 (ESP) |

### Outbound (to the Internet)

| | |
|---|---|
| Output Rule O1 | Source IP: Customer Gateway, Dest IP: VPN Gateway 1, Protocol: UDP, Source Port: 500, Destination Port: 500 |
| Output Rule O2 | Source IP: Customer Gateway, Dest IP: VPN Gateway 2, Protocol: UDP, Source Port: 500, Destination Port: 500 |
| Output Rule O3 | Source IP: Customer Gateway, Dest IP: VPN Gateway 1, Protocol: IP 50 (ESP) |
| Output Rule O4 | Source IP: Customer Gateway, Dest IP: VPN Gateway 2, Protocol: IP 50 (ESP) |

Rules I1, I2, O1, and O2 enable the transmission of IKE packets. Rules I3, I4, O3, and O4 enable the transmission of IPsec packets containing the encrypted network traffic.

# Example: Cisco IOS Device

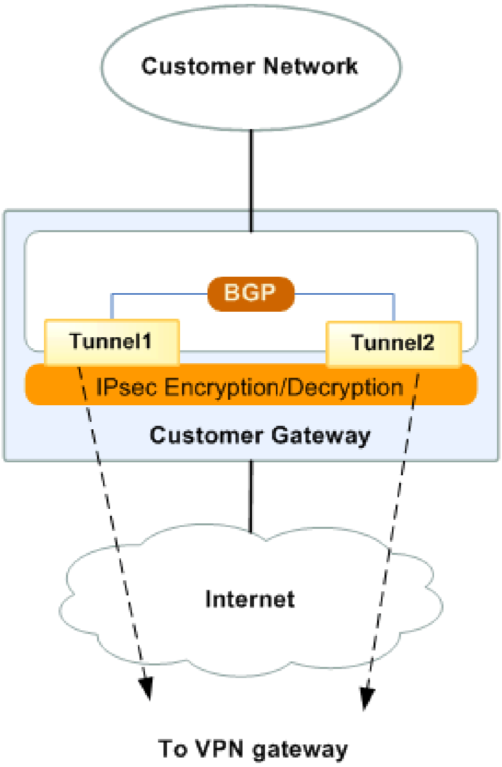
**Topics**

This section shows an example of the configuration information your integration team gives you if your customer gateway is a Cisco Integrated Services router running Cisco IOS 12.4 (or later) software.

Along with the example configuration are two diagrams: one showing the high-level layout, and another giving details that match the example configuration. You should take the real configuration information that your integration team gives you and apply it to your customer gateway.

# Cisco IOS High-Level Diagram

Following is the diagram showing the general details of your customer gateway. Note that the VPN connection consists of two separate tunnels: *Tunnel1* and *Tunnel2*. Two redundant tunnels provide an increased availability in the case of a device failure.



# Detailed Diagram and Example Configuration

Following is the diagram showing an example Cisco IOS customer gateway. After the diagram is a corresponding example of the configuration information your integration team should give you. The configuration contains a set of information for each of the two tunnels you must configure.

The example configuration refers to these items that you must provide:

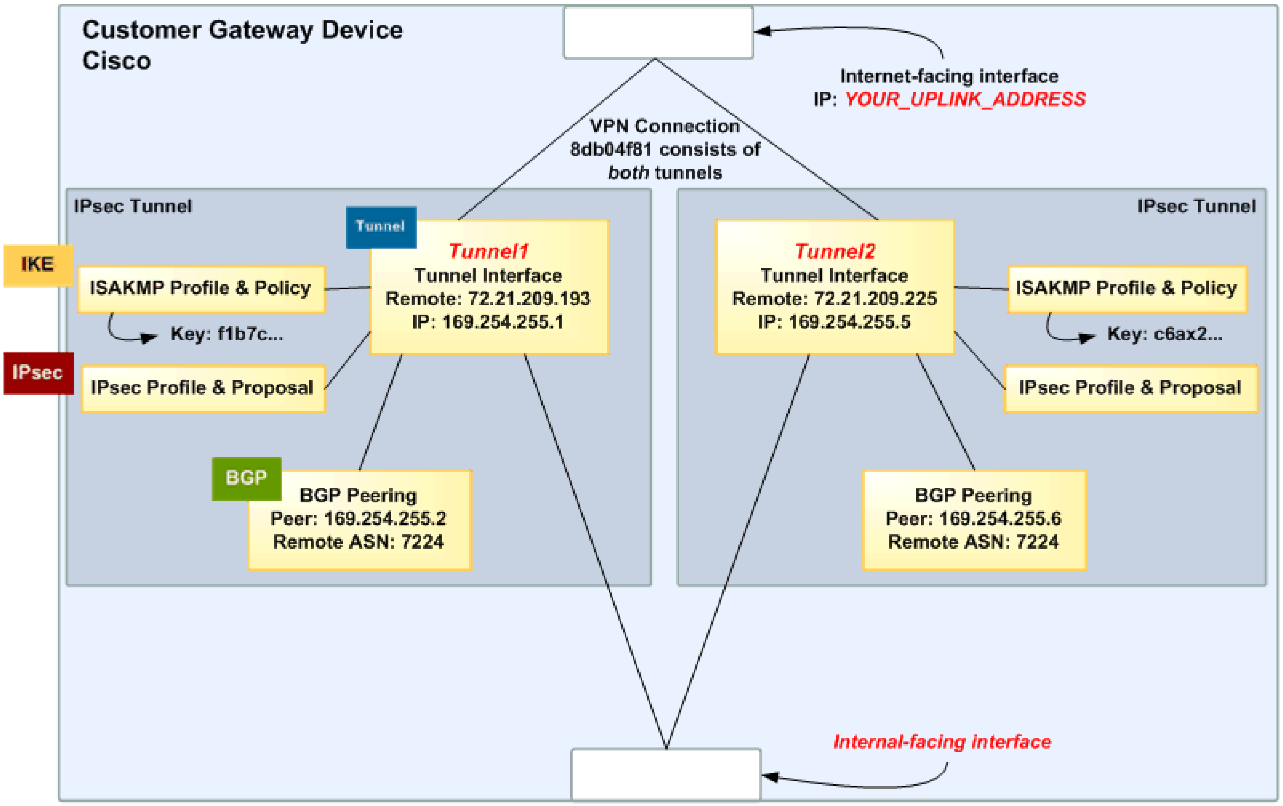
- **YOUR_UPLINK_ADDRESS—**The IP address for the Internet-routable external interface on the customer gateway (which must be static and can't be behind a device performing NAT)
- **YOUR_BGP_ASN—**The customer gateway's BGP ASN (we use 65000 by default)

The example configuration includes several dummy values to help you understand how configuration works. For example, we give dummy values for the VPN connection ID (44a8938f), VPN gateway ID (8db04f81), etc.; the IP addresses (e.g., 72.21.209.*, 169.254.255.*, etc.); and the remote ASN (7224). The actual configuration information you get will replace the dummy values with real values.

In addition to the configuration changes, you must:

- Configure the outside interface
- Configure the tunnel interface IDs (referred to as Tunnel1 and Tunnel2 in the example configuration)
- Ensure that the Crypto ISAKMP Policy Sequence number is unique
- Ensure that the Crypto IPsec Transform Set and the Crypto ISAKMP Policy Sequence are harmonious with any other IPsec tunnels configured on the device
- Configure all internal routing (getting traffic between the customer gateway and your local network)

The following diagram and example configuration highlight the items (in red italics) that you need to replace with values that apply to your own particular situation.



### Important

The following configuration information is an example of what your integration team provides you. Many of the values in the following example will be different in the actual configuration information that you receive. You must use the actual values and not the example values shown here, or your implementation will fail.

```
! Amazon Web Services
! Virtual Private Cloud

! AWS utilizes unique identifiers to manipulate the configuration of
! a VPN Connection. Each VPN Connection is assigned an identifier
! and is associated with two other identifiers, namely the
! Customer Gateway Identifier and VPN Gateway Identifier.
!
! Your VPN Connection ID   : vpn-44a8938f
! Your VPN Gateway ID      : vgw-8db04f81
! Your Customer Gateway ID : cgw-b4dc3961
!
!
! This configuration consists of two tunnels. Both tunnels must be
! configured on your Customer Gateway.
!
! -------------------------------------------------------------------------
! IPSec Tunnel #1
! -------------------------------------------------------------------------
```

IKE

```
! #1: Internet Key Exchange (IKE) Configuration
!
! A policy is established for the supported ISAKMP encryption,
! authentication, Diffie-Hellman, lifetime, and key parameters.
!
! Note that there are a global list of ISAKMP policies, each identified by
! sequence number. This policy is defined as #200, which may conflict with
! an existing policy using the same number. If so, we recommend changing
! the sequence number to avoid conflicts.
!
crypto isakmp policy 200
   encryption aes 128
   authentication pre-share
   group 2
   lifetime 28800
   hash sha
exit

! The ISAKMP keyring stores the Pre Shared Key used to authenticate the
! tunnel endpoints.
!
crypto keyring keyring-vpn-44a8938f-0
   pre-shared-key address 72.21.209.225 key plain-text-password1
exit

! An ISAKMP profile is used to associate the keyring with the particular
! endpoint.
!
crypto isakmp profile isakmp-vpn-44a8938f-0
   match identity address 72.21.209.225
   keyring keyring-vpn-44a8938f-0
exit
```

IPsec

```
! #2: IPSec Configuration
!
! The IPSec transform set defines the encryption, authentication, and IPSec
! mode parameters.
!
crypto ipsec transform-set ipsec-prop-vpn-44a8938f-0 esp-aes 128 esp-sha-hmac

   mode tunnel
exit

! The IPSec profile references the IPSec transform set and further defines
! the Diffie-Hellman group and security association lifetime.
!
crypto ipsec profile ipsec-vpn-44a8938f-0
   set pfs group2
   set security-association lifetime seconds 3600
   set transform-set ipsec-prop-vpn-44a8938f-0
exit

! Additional parameters of the IPSec configuration are set here. Note that
! these parameters are global and therefore impact other IPSec
! associations.
! This option instructs the router to clear the "Don't Fragment"
! bit from packets that carry this bit and yet must be fragmented, enabling
! them to be fragmented.
!
crypto ipsec df-bit clear

! This option enables IPSec Dead Peer Detection, which causes periodic
! messages to be sent to ensure a Security Association remains operational.
!
crypto isakmp keepalive 10 10 on-demand

! This configures the gateway's window for accepting out of order
! IPSec packets. A larger window can be helpful if too many packets
! are dropped due to reordering while in transit between gateways.
!
crypto ipsec security-association replay window-size 128

! This option instructs the router to fragment the unencrypted packets
! (prior to encryption).
!
crypto ipsec fragmentation before-encryption
```

Tunnel

```
! #3: Tunnel Interface Configuration
!
! A tunnel interface is configured to be the logical interface associated
! with the tunnel. All traffic routed to the tunnel interface will be
! encrypted and transmitted to the VPC. Similarly, traffic from the VPC
! will be logically received on this interface.
!
! Association with the IPSec security association is done through the
! "tunnel protection" command.
!
! The address of the interface is configured with the setup for your
! Customer Gateway.  If the address changes, the Customer Gateway and VPN
! Connection must be recreated with Amazon VPC.
!
interface Tunnel1
   ip address 169.254.255.2 255.255.255.252
   ip virtual-reassembly
   tunnel source YOUR_UPLINK_ADDRESS
   tunnel destination 72.21.209.225
   tunnel mode ipsec ipv4
   tunnel protection ipsec profile ipsec-vpn-44a8938f-0
   ! This option causes the router to reduce the Maximum Segment Size of
   ! TCP packets to prevent packet fragmentation.
   ip tcp adjust-mss 1396
   no shutdown
exit
```

BGP

```
! #4: Border Gateway Protocol (BGP) Configuration
!

! BGP is used within the tunnel to exchange prefixes between the
! VPN Gateway and your Customer Gateway. The VPN Gateway
! will announce the prefix corresponding to your Cloud.
!
! Your Customer Gateway must announce a default route (0.0.0.0/0),
! which can be done with the 'network' statement and
! 'default-originate' statements. Only one prefix is
! accepted by the VPN Gateway.
!
! The BGP timers are adjusted to provide more rapid detection of outages.
!
! The local BGP Autonomous System Number (ASN) (YOUR_BGP_ASN) is configured
! as part of your Customer Gateway. If the ASN must be changed, the
! Customer Gateway and VPN Connection will need to be recreated with AWS.
!
router bgp YOUR_BGP_ASN
   neighbor 169.254.255.1 remote-as 7224
   neighbor 169.254.255.1 activate
   neighbor 169.254.255.1 timers 10 30 30
   address-family ipv4 unicast
      neighbor 169.254.255.1 remote-as 7224
      neighbor 169.254.255.1 timers 10 30 30
      neighbor 169.254.255.1 default-originate
      neighbor 169.254.255.1 activate
      neighbor 169.254.255.1 soft-reconfiguration inbound
      network 0.0.0.0
   exit
exit


! --------------------------------------------------------------------------
! IPSec Tunnel #2
! --------------------------------------------------------------------------
```

IKE

```
! #1: Internet Key Exchange (IKE) Configuration
!
! A policy is established for the supported ISAKMP encryption,
! authentication, Diffie-Hellman, lifetime, and key parameters.
!
! Note that there are a global list of ISAKMP policies, each identified by
! sequence number. This policy is defined as #201, which may conflict with
! an existing policy using the same number. If so, we recommend changing
! the sequence number to avoid conflicts.
!
crypto isakmp policy 201
   encryption aes 128
   authentication pre-share
   group 2
   lifetime 28800
   hash sha
exit

! The ISAKMP keyring stores the Pre Shared Key used to authenticate the
! tunnel endpoints.
!
crypto keyring keyring-vpn-44a8938f-1
   pre-shared-key address 72.21.209.193 key plain-text-password2
exit
```

```
! An ISAKMP profile is used to associate the keyring with the particular
! endpoint.
!
crypto isakmp profile isakmp-vpn-44a8938f-1
   match identity address 72.21.209.193
   keyring keyring-vpn-44a8938f-1
exit
```

IPsec

```
! #2: IPSec Configuration
!
! The IPSec transform set defines the encryption, authentication, and IPSec
! mode parameters.
!
crypto ipsec transform-set ipsec-prop-vpn-44a8938f-1 esp-aes 128 esp-sha-hmac

   mode tunnel
exit

! The IPSec profile references the IPSec transform set and further defines
! the Diffie-Hellman group and security association lifetime.
!
crypto ipsec profile ipsec-vpn-44a8938f-1
   set pfs group2
   set security-association lifetime seconds 3600
   set transform-set ipsec-prop-vpn-44a8938f-1
exit

! Additional parameters of the IPSec configuration are set here. Note that
! these parameters are global and therefore impact other IPSec
! associations.
! This option instructs the router to clear the "Don't Fragment"
! bit from packets that carry this bit and yet must be fragmented, enabling
! them to be fragmented.
!
crypto ipsec df-bit clear

! This option enables IPSec Dead Peer Detection, which causes periodic
! messages to be sent to ensure a Security Association remains operational.
!
crypto isakmp keepalive 10 10 on-demand

! This configures the gateway's window for accepting out of order
! IPSec packets. A larger window can be helpful if too many packets
! are dropped due to reordering while in transit between gateways.
!
crypto ipsec security-association replay window-size 128

! This option instructs the router to fragment the unencrypted packets
! (prior to encryption).
!
crypto ipsec fragmentation before-encryption
```

Tunnel

```
! #3: Tunnel Interface Configuration
```

```
!
! A tunnel interface is configured to be the logical interface associated
! with the tunnel. All traffic routed to the tunnel interface will be
! encrypted and transmitted to the VPC. Similarly, traffic from the VPC
! will be logically received on this interface.
!
! Association with the IPSec security association is done through the
! "tunnel protection" command.
!
! The address of the interface is configured with the setup for your
! Customer Gateway.  If the address changes, the Customer Gateway and VPN
! Connection must be recreated with Amazon VPC.
!
interface Tunnel2
   ip address 169.254.255.6 255.255.255.252
   ip virtual-reassembly
   tunnel source YOUR_UPLINK_ADDRESS
   tunnel destination 72.21.209.193
   tunnel mode ipsec ipv4
   tunnel protection ipsec profile ipsec-vpn-44a8938f-1
   ! This option causes the router to reduce the Maximum Segment Size of
   ! TCP packets to prevent packet fragmentation.
   ip tcp adjust-mss 1396
   no shutdown
exit
```

BGP

```
! #4: Border Gateway Protocol (BGP) Configuration
!

! BGP is used within the tunnel to exchange prefixes between the
! VPN Gateway and your Customer Gateway. The VPN Gateway
! will announce the prefix corresponding to your Cloud.
!
! Your Customer Gateway must announce a default route (0.0.0.0/0),
! which can be done with the 'network' statement and
! 'default-originate' statements. Only one prefix is
! accepted by the VPN Gateway.
!
! The BGP timers are adjusted to provide more rapid detection of outages.
!
! The local BGP Autonomous System Number (ASN) (YOUR_BGP_ASN) is configured
! as part of your Customer Gateway. If the ASN must be changed, the
! Customer Gateway and VPN Connection will need to be recreated with AWS.
!
router bgp YOUR_BGP_ASN
   neighbor 169.254.255.5 remote-as 7224
   neighbor 169.254.255.5 activate
   neighbor 169.254.255.5 timers 10 30 30
   address-family ipv4 unicast
      neighbor 169.254.255.5 remote-as 7224
      neighbor 169.254.255.5 timers 10 30 30
      neighbor 169.254.255.5 default-originate
      neighbor 169.254.255.5 activate
      neighbor 169.254.255.5 soft-reconfiguration inbound
      network 0.0.0.0
   exit
```

```
exit


! Additional Notes and Questions
!  - Amazon Virtual Private Cloud Getting Started Guide:
!       http://docs.amazonwebservices.com/AWSVPC/latest/GettingStartedGuide
!  - Amazon Virtual Private Cloud Network Administrator Guide:
!       http://docs.amazonwebservices.com/AWSVPC/latest/NetworkAdminGuide
```

# How to Test the Configuration

You must first test the gateway configuration for each tunnel.

### To test the customer gateway configuration for each tunnel

1. On your customer gateway, determine if the BGP status is *Active*.
   It takes approximately 30 seconds for a BGP peering to become active.
2. Ensure that the customer gateway is advertising the default route (0.0.0.0/0) to the VPN gateway.

When properly established, your BGP peering should be receiving one route from the VPN gateway corresponding to the prefix that your VPC integration team specified for the VPC (e.g., 10.0.0.0/24). If the BGP peering is established, you are receiving a prefix, and you are advertising a prefix, your tunnel is configured correctly. Make sure both tunnels are in this state.

Next you must test the connectivity for each tunnel.

**Important**

For the connectivity test to work, you must configure any security group or network ACL in your VPC that filters traffic to the instance to allow inbound and outbound ICMP traffic.

### To test the end-to-end connectivity

1. Launch an instance of one of the Amazon Linux AMIs into your VPC. They're available in the Quick Start menu when you use the instance launch wizard in the AWS Management Console (for more information, refer to the Amazon Virtual Private Cloud Getting Started Guide).
2. After the instance is running, get its private IP address (e.g., 10.0.0.4). The AWS Management Console displays the address as part of the instance's details.
3. On a system in your home network, use the ping command with the instance's IP address. Make sure the computer you ping from is behind the customer gateway. A successful response should be similar to the following:

```
PROMPT> ping 10.0.0.4
Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.4:
Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

**Note**

If you ping an instance from your customer gateway router, ensure you are sourcing ping messages from an internal IP address, not a tunnel IP address. Some AMIs will not respond to ping messages from the tunnel IP addresses.

If your tunnels do not test successfully, see Troubleshooting (p. 48).

# Example: Juniper JunOS Device

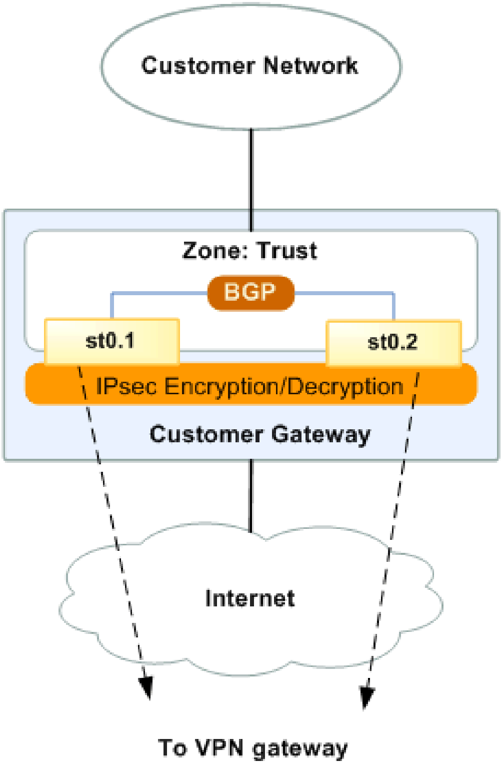
**Topics**

This section shows an example of the configuration information your integration team gives you if your customer gateway is a Juniper J-Series router running JunOS 9.5 (or later) software.

Along with the example configuration are two diagrams: one showing the high-level layout, and another giving details that match the example configuration. You should take the real configuration information that your integration team gives you and apply it to your customer gateway.

# Juniper JunOS High-Level Diagram

Following is the diagram showing the general details of your customer gateway. Note that the VPN connection consists of two separate tunnels: *st0.1* and *st0.2*. Two redundant tunnels provide an increased availability in the case of a device failure.



# Detailed Diagram and Example Configuration

Following is the diagram showing an example Juniper JunOS customer gateway. After the diagram is a corresponding example of the configuration information your integration team will should give you. The configuration contains a set of information for each of the two tunnels you must configure.

The example configuration refers to these items that you must provide:

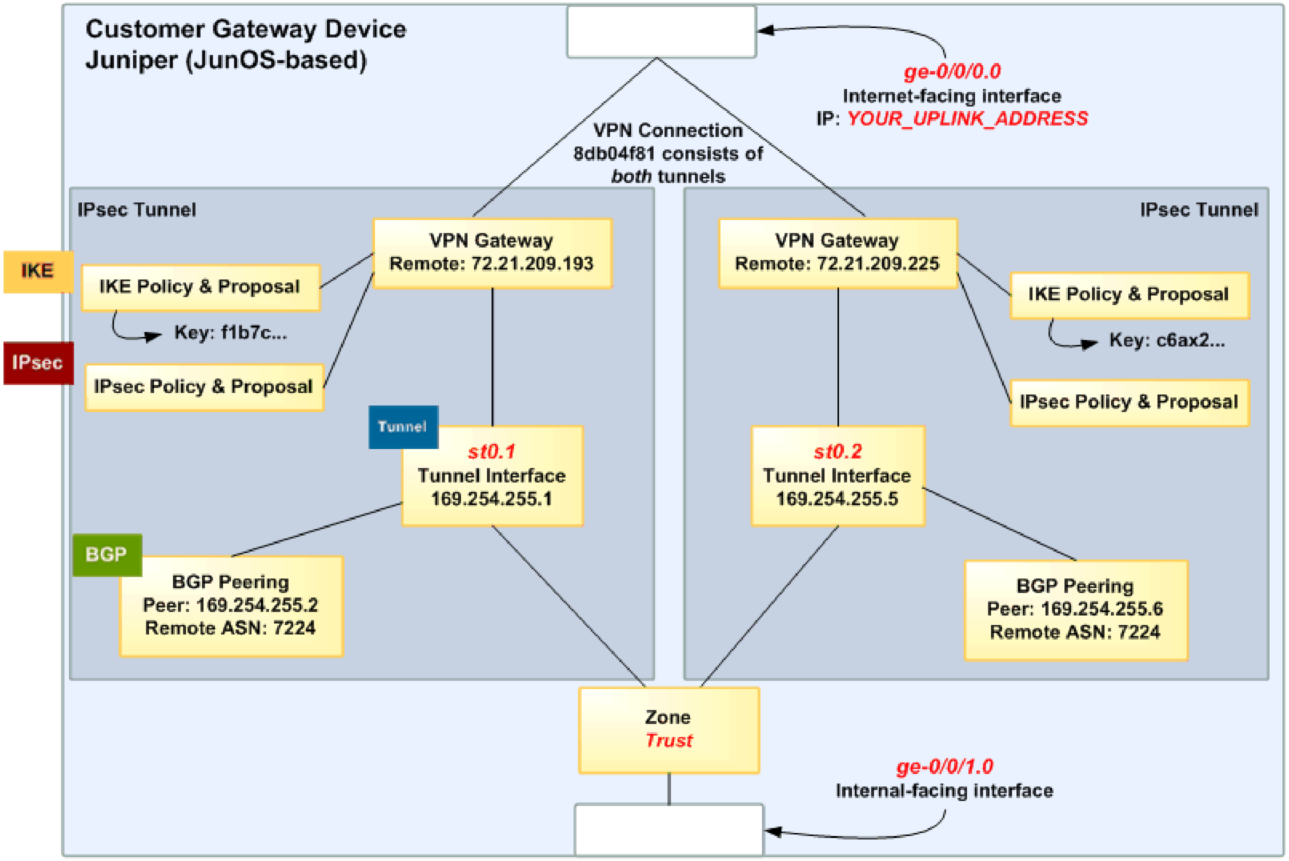
- **YOUR_UPLINK_ADDRESS—**The IP address for the Internet-routable external interface on the customer gateway (which must be static and can't be behind a device performing NAT)
- **YOUR_BGP_ASN—**The customer gateway's BGP ASN (we use 65000 by default)

The example configuration includes several dummy values to help you understand how configuration works. For example, we give dummy values for the VPN connection ID (44a8938f), VPN gateway ID (8db04f81), etc.; the IP addresses (e.g., 72.21.209.*, 169.254.255.*, etc.); and the remote ASN (7224). The actual configuration information you get will replace the dummy values to real values.

In addition to the configuration changes, you must:

- Configure the outside interface (referred to as ge-0/0/0.0 in the example configuration)
- Configure the tunnel interface IDs (referred to as st0.1 and st0.2 in the example configuration)
- Configure all internal routing (getting traffic between the customer gateway and your local network)
- Identify the security zone for the uplink interface (the following configuration information uses the default "untrust" zone)
- Identify the security zone for the inside interface (the following configuration information uses the default "trust" zone)

The following diagram and example configuration highlight the items (in red italics) that you need to replace with values that apply to your own particular situation.



**Important**

The following configuration information is an example of what the Amazon VPC API provides you. Many of the values in the following example will be different in the actual configuration information that you receive from the API. You must use the values from the API and not the example values shown here or your implementation will fail.

```
# Amazon Web Services
# Virtual Private Cloud
#
# AWS utilizes unique identifiers to manipulate the configuration of
# a VPN Connection. Each VPN Connection is assigned a VPN Connection
# Identifier and is associated with two other identifiers, namely the
# Customer Gateway Identifier and the VPN Gateway Identifier.
#
# Your VPN Connection ID   : vpn-44a8938f
# Your VPN Gateway ID      : vgw-8db04f81
# Your Customer Gateway ID : cgw-b4dc3961
#
# This configuration consists of two tunnels. Both tunnels must be
# configured on your Customer Gateway.
#
# -------------------------------------------------------------------------
# IPSec Tunnel #1
# --------------------------------------------------------------------------
IKE
# #1: Internet Key Exchange (IKE) Configuration
#
# A proposal is established for the supported IKE encryption,
# authentication, Diffie-Hellman, and lifetime parameters.
#
set security ike proposal ike-prop-vpn-44a8938f-1 authentication-method pre-
shared-keys
set security ike proposal ike-prop-vpn-44a8938f-1 authentication-algorithm sha1
set security ike proposal ike-prop-vpn-44a8938f-1 encryption-algorithm aes-128-
cbc
set security ike proposal ike-prop-vpn-44a8938f-1 lifetime-seconds 28800
set security ike proposal ike-prop-vpn-44a8938f-1 dh-group group2

# An IKE policy is established to associate a Pre Shared Key with the
# defined proposal.
#
set security ike policy ike-pol-vpn-44a8938f-1 mode main
set security ike policy ike-pol-vpn-44a8938f-1 proposals ike-prop-vpn-44a8938f-
0
set security ike policy ike-pol-vpn-44a8938f-1 pre-shared-key ascii-text plain-
text-password1

# The IKE gateway is defined to be the VPN Gateway. The gateway
# configuration associates a local interface, remote IP address, and
# IKE policy.
#
# This example shows the outside of the tunnel as interface ge-0/0/0.0.
# This should be set to the interface that IP address YOUR_UPLINK_ADDRESS is
# associated with.
# This address is configured with the setup for your Customer Gateway.
#
# If the address changes, the Customer Gateway and VPN Connection must
# be recreated.
set security ike gateway gw-vpn-44a8938f-1 ike-policy ike-pol-vpn-44a8938f-0
set security ike gateway gw-vpn-44a8938f-1 external-interface ge-0/0/0.0
set security ike gateway gw-vpn-44a8938f-1 address 72.21.209.225

# Troubleshooting IKE connectivity can be aided by enabling IKE tracing.
# The configuration below will cause the router to log IKE messages to
# the 'kmd' log. Run 'show messages kmd' to retrieve these logs.
# set security ike traceoptions file kmd
# set security ike traceoptions file size 1024768
# set security ike traceoptions file files 10
# set security ike traceoptions flag all

IPsec
# #2: IPSec Configuration
#
# The IPSec proposal defines the protocol, authentication, encryption, and
# lifetime parameters for our IPSec security association.
#
set security ipsec proposal ipsec-prop-vpn-44a8938f-1 protocol esp
set security ipsec proposal ipsec-prop-vpn-44a8938f-1 authentication-algorithm
 hmac-sha1-96
```

```
set security ipsec proposal ipsec-prop-vpn-44a8938f-1 encryption-algorithm aes-
128-cbc
set security ipsec proposal ipsec-prop-vpn-44a8938f-1 lifetime-seconds 3600

# The IPSec policy incorporates the Diffie-Hellman group and the IPSec
# proposal.
#
set security ipsec policy ipsec-pol-vpn-44a8938f-1 perfect-forward-secrecy keys
 group2
set security ipsec policy ipsec-pol-vpn-44a8938f-1 proposals ipsec-prop-vpn-
44a8938f-0

# A security association is defined here. The IPSec Policy and IKE gateways
# are associated with a tunnel interface (st0.1).
# The tunnel interface ID is assumed; if other tunnels are defined on
# your router, you will need to specify a unique interface name
# (for example, st0.10).
#
set security ipsec vpn vpn-44a8938f-1 bind-interface st0.1
set security ipsec vpn vpn-44a8938f-1 ike gateway gw-vpn-44a8938f-0
set security ipsec vpn vpn-44a8938f-1 ike ipsec-policy ipsec-pol-vpn-44a8938f-
0
set security ipsec vpn vpn-44a8938f-1 df-bit clear

# This option enables IPSec Dead Peer Detection, which causes periodic
# messages to be sent to ensure a Security Association remains operational.
#
set security ike gateway gw-vpn-44a8938f-1 dead-peer-detection
```

Tunnel

```
# #3: Tunnel Interface Configuration
#

# The tunnel interface is configured with the internal IP address.
#
set interfaces st0.1 family inet address 169.254.255.2/30
set interfaces st0.1 family inet mtu 1436
set security zones security-zone trust interfaces st0.1

# The security zone protecting external interfaces of the router must be
# configured to allow IKE traffic inbound.
#
set security zones security-zone untrust host-inbound-traffic system-services
ike

# The security zone protecting internal interfaces (including the logical
# tunnel interfaces) must be configured to allow BGP traffic inbound.
#
set security zones security-zone trust host-inbound-traffic protocols bgp

# This option causes the router to reduce the Maximum Segment Size of
# TCP packets to prevent packet fragmentation.
#
set security flow tcp-mss ipsec-vpn mss 1396
```

BGP

```
# #4: Border Gateway Protocol (BGP) Configuration
#

# BGP is used within the tunnel to exchange prefixes between the
# VPN Gateway and your Customer Gateway. The VPN Gateway
# will announce the prefix corresponding to your VPC.
#
# Your Customer Gateway must announce a default route (0.0.0.0/0),
# which can be done with the EXPORT-DEFAULT policy. Only one prefix is
# accepted by the VPN Gateway.
#

# The BGP timers are adjusted to provide more rapid detection of outages.

#
# The local BGP Autonomous System Number (ASN) (YOUR_BGP_ASN) is configured
# as part of your Customer Gateway. If the ASN must be changed, the
# Customer Gateway and VPN Connection will need to be recreated with AWS.
#
# We establish a basic route policy to export a default route to the
# VPN Gateway.
#
set policy-options policy-statement EXPORT-DEFAULT term default from route-filter 0.0.0.0/0 exact

set policy-options policy-statement EXPORT-DEFAULT term default then accept

set policy-options policy-statement EXPORT-DEFAULT term reject then reject

set protocols bgp group ebgp type external

set protocols bgp group ebgp neighbor 169.254.255.1 export EXPORT-DEFAULT
set protocols bgp group ebgp neighbor 169.254.255.1 peer-as 7224
set protocols bgp group ebgp neighbor 169.254.255.1 hold-time 30
set protocols bgp group ebgp neighbor 169.254.255.1 local-as YOUR_BGP_ASN

# --------------------------------------------------------------------------
# IPSec Tunnel #2
# --------------------------------------------------------------------------
```

**IKE**

```
# #1: Internet Key Exchange (IKE) Configuration
#
# A proposal is established for the supported IKE encryption,
# authentication, Diffie-Hellman, and lifetime parameters.
#
set security ike proposal ike-prop-vpn-44a8938f-2 authentication-method pre-shared-keys
set security ike proposal ike-prop-vpn-44a8938f-2 authentication-algorithm sha1
set security ike proposal ike-prop-vpn-44a8938f-2 encryption-algorithm aes-128-cbc
set security ike proposal ike-prop-vpn-44a8938f-2 lifetime-seconds 28800
set security ike proposal ike-prop-vpn-44a8938f-2 dh-group group2

# An IKE policy is established to associate a Pre Shared Key with the
# defined proposal.
#
set security ike policy ike-pol-vpn-44a8938f-2 mode main
set security ike policy ike-pol-vpn-44a8938f-2 proposals ike-prop-vpn-44a8938f-2
set security ike policy ike-pol-vpn-44a8938f-2 pre-shared-key ascii-text plain-text-password2

# The IKE gateway is defined to be the VPN Gateway. The gateway
# configuration associates a local interface, remote IP address, and
# IKE policy.
#
# This example shows the outside of the tunnel as interface ge-0/0/0.0.
# This should be set to the interface that IP address YOUR_UPLINK_ADDRESS is
# associated with.
# This address is configured with the setup for your Customer Gateway.
#
# If the address changes, the Customer Gateway and VPN Connection must be recreated.
#
set security ike gateway gw-vpn-44a8938f-2 ike-policy ike-pol-vpn-44a8938f-1
set security ike gateway gw-vpn-44a8938f-2 external-interface ge-0/0/0.0
set security ike gateway gw-vpn-44a8938f-2 address 72.21.209.193

# Troubleshooting IKE connectivity can be aided by enabling IKE tracing.
# The configuration below will cause the router to log IKE messages to
# the 'kmd' log. Run 'show messages kmd' to retrieve these logs.
# set security ike traceoptions file kmd
# set security ike traceoptions file size 1024768
# set security ike traceoptions file files 10
# set security ike traceoptions flag all
```

IPsec

```
# #2: IPSec Configuration
#
# The IPSec proposal defines the protocol, authentication, encryption, and
# lifetime parameters for our IPSec security association.
#
set security ipsec proposal ipsec-prop-vpn-44a8938f-2 protocol esp
set security ipsec proposal ipsec-prop-vpn-44a8938f-2 authentication-algorithm hmac-sha1-96
set security ipsec proposal ipsec-prop-vpn-44a8938f-2 encryption-algorithm aes-128-cbc
set security ipsec proposal ipsec-prop-vpn-44a8938f-2 lifetime-seconds 3600

# The IPSec policy incorporates the Diffie-Hellman group and the IPSec
# proposal.
#
set security ipsec policy ipsec-pol-vpn-44a8938f-2 perfect-forward-secrecy keys group2
set security ipsec policy ipsec-pol-vpn-44a8938f-2 proposals ipsec-prop-vpn-44a8938f-2

# A security association is defined here. The IPSec Policy and IKE gateways
# are associated with a tunnel interface (st0.2).
# The tunnel interface ID is assumed; if other tunnels are defined on
# your router, you will need to specify a unique interface name
# (for example, st0.20).
#
set security ipsec vpn vpn-44a8938f-2 bind-interface st0.2
set security ipsec vpn vpn-44a8938f-2 ike gateway gw-vpn-44a8938f-2
set security ipsec vpn vpn-44a8938f-2 ike ipsec-policy ipsec-pol-vpn-44a8938f-
2
set security ipsec vpn vpn-44a8938f-2 df-bit clear

# This option enables IPSec Dead Peer Detection, which causes periodic
# messages to be sent to ensure a Security Association remains operational.
#
set security ike gateway gw-vpn-44a8938f-2 dead-peer-detection
```

Tunnel

```
# #3: Tunnel Interface Configuration
#

# The tunnel interface is configured with the internal IP address.
#
set interfaces st0.2 family inet address 169.254.255.6/30
set interfaces st0.2 family inet mtu 1436
set security zones security-zone trust interfaces st0.2

# The security zone protecting external interfaces of the router must be
# configured to allow IKE traffic inbound.
#
set security zones security-zone untrust host-inbound-traffic system-services
ike

# The security zone protecting internal interfaces (including the logical
# tunnel interfaces) must be configured to allow BGP traffic inbound.
#
set security zones security-zone trust host-inbound-traffic protocols bgp

# This option causes the router to reduce the Maximum Segment Size of
# TCP packets to prevent packet fragmentation.
#
set security flow tcp-mss ipsec-vpn mss 1396
```

BGP

```
# #4: Border Gateway Protocol (BGP) Configuration
#

# BGP is used within the tunnel to exchange prefixes between the
# VPN Gateway and your Customer Gateway. The VPN Gateway
# will announce the prefix corresponding to your VPC.
#
# Your Customer Gateway must announce a default route (0.0.0.0/0),
# which can be done with the EXPORT-DEFAULT policy. Only one prefix is
# accepted by the VPN Gateway.
#

# The BGP timers are adjusted to provide more rapid detection of outages.

#
# The local BGP Autonomous System Number (ASN) (YOUR_BGP_ASN) is configured
# as part of your Customer Gateway. If the ASN must be changed, the
# Customer Gateway and VPN Connection will need to be recreated with AWS.
#
```

```
# We establish a basic route policy to export a default route to the
# VPN Gateway.
#
set policy-options policy-statement EXPORT-DEFAULT term default from route-
filter 0.0.0.0/0 exact

set policy-options policy-statement EXPORT-DEFAULT term default then accept

set policy-options policy-statement EXPORT-DEFAULT term reject then reject

set protocols bgp group ebgp type external

set protocols bgp group ebgp neighbor 169.254.255.5 export EXPORT-DEFAULT
set protocols bgp group ebgp neighbor 169.254.255.5 peer-as 7224
set protocols bgp group ebgp neighbor 169.254.255.5 hold-time 30
set protocols bgp group ebgp neighbor 169.254.255.5 local-as YOUR_BGP_ASN

# ---------------------------------------------------------------------------

# Additional Notes and Questions
#  - Amazon Virtual Private Cloud Getting Started Guide:
#       http://docs.amazonwebservices.com/AWSVPC/latest/GettingStartedGuide
#  - Amazon Virtual Private Cloud Network Administrator Guide:
#       http://docs.amazonwebservices.com/AWSVPC/latest/NetworkAdminGuide
```

# How to Test the Configuration

You must first test the gateway configuration for each tunnel.

### To test the customer gateway configuration for each tunnel

1. On your customer gateway, determine if the BGP status is *Active*.
   It takes approximately 30 seconds for a BGP peering to become active.
2. Ensure that the customer gateway is advertising the default route (0.0.0.0/0) to the VPN gateway.

When properly established, your BGP peering should be receiving one route from the VPN gateway corresponding to the prefix that your VPC integration team specified for the VPC (e.g., 10.0.0.0/24). If the BGP peering is established, you are receiving a prefix, and you are advertising a prefix, your tunnel is configured correctly. Make sure both tunnels are in this state.

Next you must test the connectivity for each tunnel.

**Important**

For the connectivity test to work, you must configure any security group or network ACL in your VPC that filters traffic to the instance to allow inbound and outbound ICMP traffic.

### To test the end-to-end connectivity

1. Launch an instance of one of the Amazon Linux AMIs into your VPC. They're available in the Quick Start menu when you use the instance launch wizard in the AWS Management Console (for more information, refer to the Amazon Virtual Private Cloud Getting Started Guide).

2. After the instance is running, get its private IP address (e.g., 10.0.0.4). The AWS Management Console displays the address as part of the instance's details.
3. On a system in your home network, use the ping command with the instance's IP address. Make sure the computer you ping from is behind the customer gateway. A successful response should be similar to the following:

```
PROMPT> ping 10.0.0.4
Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.4:
Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),

Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

### Note

If you ping an instance from your customer gateway router, ensure you are sourcing ping messages from an internal IP address, not a tunnel IP address. Some AMIs will not respond to ping messages from the tunnel IP addresses.

If your tunnels do not test successfully, see Troubleshooting (p. 48).

# Example: Juniper ScreenOS Device

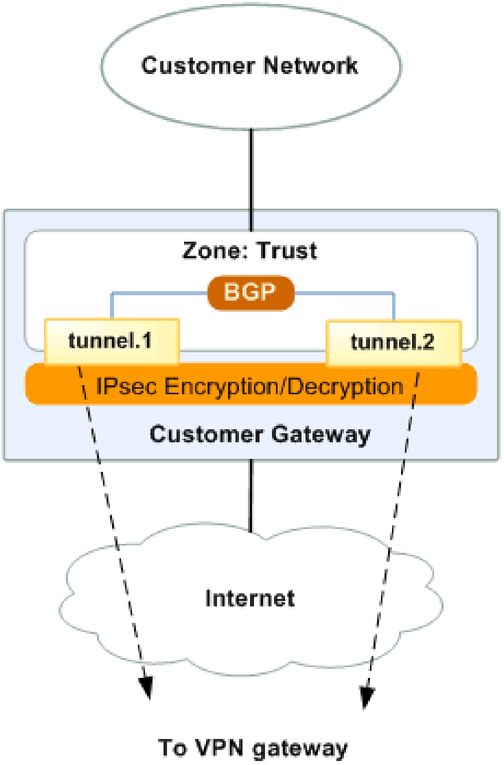
**Topics**

This section shows an example of the configuration information your integration team gives you if your customer gateway is a Juniper SSG or Netscreen series device running Juniper ScreenOS software. You can use this example as a template to apply the actual configuration your integration team give you to your customer gateway.

Along with the example configuration are two diagrams: one showing the high-level layout and another giving details that match the example configuration.

# Juniper ScreenOS High-Level Diagram

The following diagram shows the general details of your customer gateway. Note that the VPN connection consists of two separate tunnels: *tunnel.1* and *tunnel.2*. Two redundant tunnels provide an increased availability in the case of a device failure.



# Detailed Diagram and Example Configuration

The diagram in this section shows an example Juniper ScreenOS customer gateway. After the diagram is a corresponding example of the configuration information your integration team should give you. The configuration contains information for each of the two tunnels you must configure.

The example configuration refers to two items that you must provide:

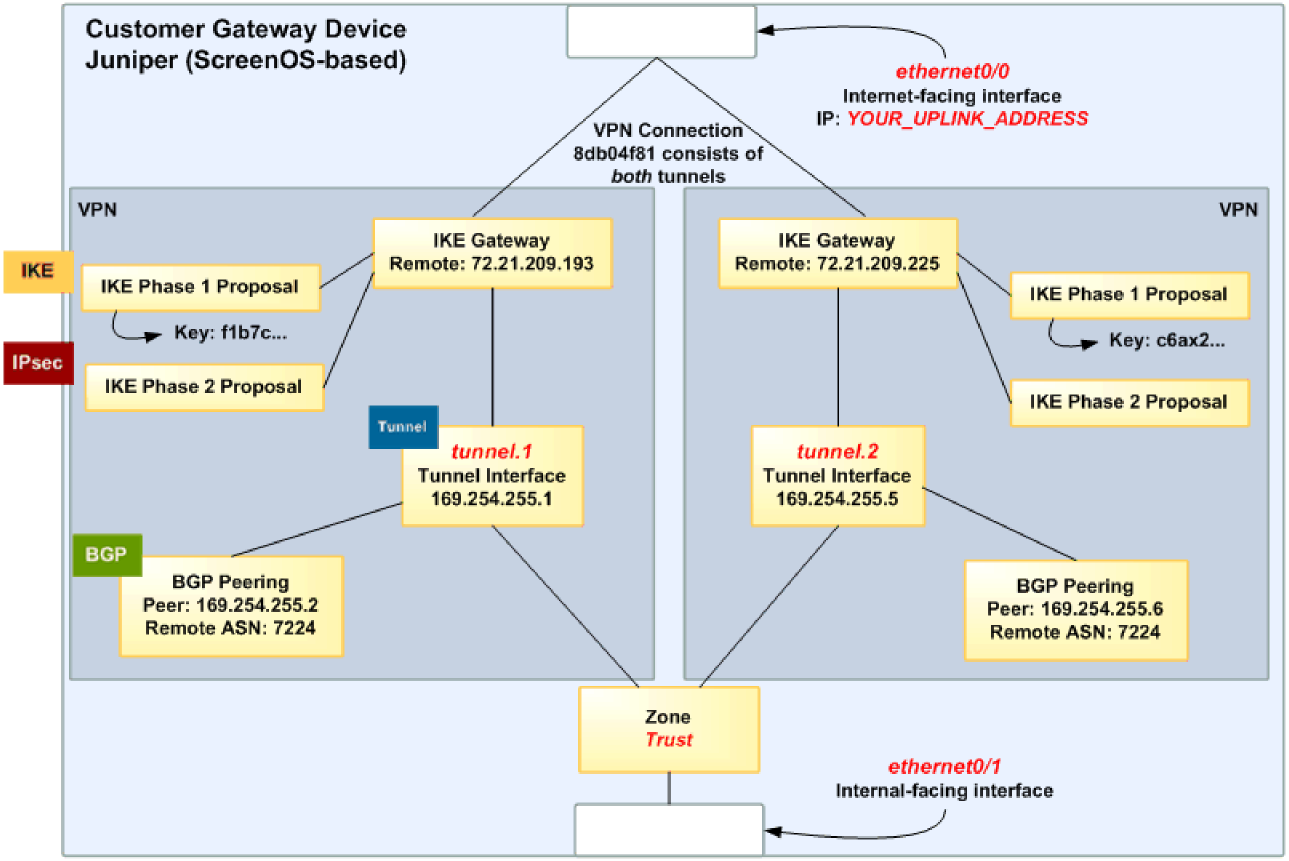
- **YOUR_UPLINK_ADDRESS—**The IP address for the Internet-routable external interface on the customer gateway (the address must be static and can't be behind a device performing NAT)
- **YOUR_BGP_ASN—**The customer gateway's BGP ASN (we use 65000 by default)

The example configuration includes several dummy values to help you understand how configuration works. For example, we give dummy values for the VPN connection ID (44a8938f), VPN gateway ID (8db04f81), etc.; the IP addresses (e.g., 72.21.209.*, 169.254.255.*, etc.); and the remote ASN (7224). The actual configuration information you get will replace the dummy values with real values.

In addition to the dummy value changes, you must:

- Configure the outside interface (referred to as ethernet0/0 in the example configuration)
- Configure the tunnel interface IDs (referred to as tunnel.1 and tunnel.2 in the example configuration)
- Configure all internal routing (getting traffic between the customer gateway and your local network)

The following diagram and example configuration highlight the items (in red italics) that you need to replace with values that apply to your own particular situation.



### Important

The following configuration information is an example of what your integration team provides you. Many of the values in the following example will be different in the configuration information that you actually receive. Of course, you must use the actual values and not the example values shown here or your implementation will fail.

### Important

The configuration below is appropriate for ScreenOS versions 6.2 and newer. A separate configuration is available for ScreenOS version 6.1.

```
# Amazon Web Services
# Virtual Private Cloud
#
# AWS utilizes unique identifiers to manipulate the configuration of a VPN
# Connection. Each VPN Connection is assigned a VPN Connection Identifier
# and is associated with two other identifiers, namely the Customer Gateway
# Identifier and the VPN Gateway Identifier.
#
# Your VPN Connection ID   : vpn-44a8938f
# Your VPN Gateway ID      : vgw-8db04f81
# Your Customer Gateway ID : cgw-b4dc3961
```

```
#
# This configuration consists of two tunnels. Both tunnels must be configured
# on your Customer Gateway.
#
# This configuration was tested on a Juniper SSG-5 running ScreenOS 6.3R2.
#
# ----------------------------------------------------------------------------
----
# IPSec Tunnel #1
# ----------------------------------------------------------------------------
----
```

IKE

```
# #1: Internet Key Exchange (IKE) Configuration
#
# A proposal is established for the supported IKE encryption, authentication,
# Diffie-Hellman, and lifetime parameters.
#

set ike p1-proposal ike-prop-vpn-44a8938f-1 preshare group2 esp aes128 sha-1
second 28800

# The IKE gateway is defined to be the VPN Gateway. The gateway configuration
# associates a local interface, remote IP address, and IKE policy.
#
# This example shows the outside of the tunnel as interface ethernet0/0. This
# should be set to the interface that IP address YOUR_UPLINK_ADDRESS is
# associated with.
#
# This address is configured with the setup for your Customer Gateway. If the
# address changes, the Customer Gateway and VPN Connection must be recreated.
#

set ike gateway gw-vpn-44a8938f-1 address 72.21.209.225 id 72.21.209.225 main
outgoing-interface ethernet0/0 preshare "plain-text-password1" proposal ike-
prop-vpn-44a8938f-1

# Troubleshooting IKE connectivity can be aided by enabling IKE debugging.
# To do so, run the following commands:
# clear dbuf          -- Clear debug buffer
# debug ike all       -- Enable IKE debugging
# get dbuf stream     -- View debug messages
# undebug all         -- Turn off debugging
```

IPsec

```
# #2: IPSec Configuration
#
# The IPSec (Phase 2) proposal defines the protocol, authentication,
# encryption, and lifetime parameters for our IPSec security association.
#

set ike p2-proposal ipsec-prop-vpn-44a8938f-1 group2 esp aes128 sha-1 second
3600
set ike gateway gw-vpn-44a8938f-1 dpd-liveness interval 10
set vpn IPSEC-vpn-44a8938f-1 gateway gw-vpn-44a8938f-1 replay tunnel proposal
ipsec-prop-vpn-44a8938f-1
```

Tunnel

```
# #3: Tunnel Interface Configuration
#
# The tunnel interface is configured with the internal IP address.
#
# To establish connectivity between your internal network and the VPC, you
# must have an interface facing your internal network in the "Trust" zone.
#

set interface tunnel.1 zone Trust
set interface tunnel.1 ip 169.254.255.2/30
set interface tunnel.1 mtu 1436
set vpn IPSEC-vpn-44a8938f-1 bind interface tunnel.1

# By default, the router will block asymmetric VPN traffic, which may occur
# with this VPN Connection. This occurs, for example, when routing policies
# cause traffic to sent from your router to VPC through one IPSec tunnel
# while traffic returns from VPC through the other.
#
# This command allows this traffic to be received by your device.

set zone Trust asymmetric-vpn


# This option causes the router to reduce the Maximum Segment Size of TCP
# packets to prevent packet fragmentation.
#

set flow vpn-tcp-mss 1396
```

BGP

```
# #4: Border Gateway Protocol (BGP) Configuration
#
# BGP is used within the tunnel to exchange prefixes between the VPN Gateway
# and your Customer Gateway. The VPN Gateway will announce the prefix
# corresponding to your VPC.
#
# Your Customer Gateway must announce a default route (0.0.0.0/0). Only one
# prefix is accepted by the VPN Gateway.
#
# The BGP timers are adjusted to provide more rapid detection of outages.
#
# The local BGP Autonomous System Number (ASN) (YOUR_BGP_ASN) is configured
# as part of your Customer Gateway. If the ASN must be changed, the
# Customer Gateway and VPN Connection will need to be recreated with AWS.
#

set vrouter trust-vr
set max-ecmp-routes 2
set protocol bgp YOUR_BGP_ASN
set hold-time 30
set ipv4 network 0.0.0.0/0
set ipv4 advertise-def-route
set enable
set neighbor 169.254.255.1 remote-as 7224
set neighbor 169.254.255.1 enable
set ipv4 neighbor 169.254.255.1 activate
exit
exit
set interface tunnel.1 protocol bgp

# --------------------------------------------------------------------------
# IPSec Tunnel #2
# --------------------------------------------------------------------------
```

IKE

```
# #1: Internet Key Exchange (IKE) Configuration
#
# A proposal is established for the supported IKE encryption, authentication,
# Diffie-Hellman, and lifetime parameters.
#

set ike p1-proposal ike-prop-vpn-44a8938f-2 preshare group2 esp aes128 sha-1
second 28800

# The IKE gateway is defined to be the VPN Gateway. The gateway configuration
# associates a local interface, remote IP address, and IKE policy.
#
# This example shows the outside of the tunnel as interface ethernet0/0. This
# should be set to the interface that IP address YOUR_UPLINK_ADDRESS is
# associated with.
#
# This address is configured with the setup for your Customer Gateway. If the
# address changes, the Customer Gateway and VPN Connection must be recreated.
#
set ike gateway gw-vpn-44a8938f-2 address 72.21.209.193 id 72.21.209.193 main
outgoing-interface ethernet0/0 preshare "plain-text-password2" proposal ike-
prop-vpn-44a8938f-2

# Troubleshooting IKE connectivity can be aided by enabling IKE debugging.
# To do so, run the following commands:
# clear dbuf          -- Clear debug buffer
# debug ike all       -- Enable IKE debugging
# get dbuf stream     -- View debug messages
# undebug all         -- Turn off debugging
```

IPsec

```
# #2: IPSec Configuration
#
# The IPSec (Phase 2) proposal defines the protocol, authentication,
# encryption, and lifetime parameters for our IPSec security association.
#

set ike p2-proposal ipsec-prop-vpn-44a8938f-2 group2 esp aes128 sha-1 second
3600
set ike gateway gw-vpn-44a8938f-2 dpd-liveness interval 10
set vpn IPSEC-vpn-44a8938f-2 gateway gw-vpn-44a8938f-2 replay tunnel proposal
ipsec-prop-vpn-44a8938f-2
```

Tunnel

```
# #3: Tunnel Interface Configuration
#
# The tunnel interface is configured with the internal IP address.
#
# To establish connectivity between your internal network and the VPC, you
# must have an interface facing your internal network in the "Trust" zone.

set interface tunnel.2 zone Trust
set interface tunnel.2 ip 169.254.255.6/30
set interface tunnel.2 mtu 1436
set vpn IPSEC-vpn-44a8938f-2 bind interface tunnel.2

# By default, the router will block asymmetric VPN traffic, which may occur
# with this VPN Connection. This occurs, for example, when routing policies
# cause traffic to sent from your router to VPC through one IPSec tunnel
# while traffic returns from VPC through the other.
#
# This command allows this traffic to be received by your device.

set zone Trust asymmetric-vpn

# This option causes the router to reduce the Maximum Segment Size of TCP
# packets to prevent packet fragmentation.

set flow vpn-tcp-mss 1396
```

BGP

```
# #4: Border Gateway Protocol (BGP) Configuration
#
# BGP is used within the tunnel to exchange prefixes between the VPN Gateway
# and your Customer Gateway. The VPN Gateway will announce the prefix
# corresponding to your VPC.
#
# Your Customer Gateway must announce a default route (0.0.0.0/0). Only one
# prefix is accepted by the VPN Gateway.
#
# The BGP timers are adjusted to provide more rapid detection of outages.
#
# The local BGP Autonomous System Number (ASN) (YOUR_BGP_ASN) is configured
# as part of your Customer Gateway. If the ASN must be changed, the
# Customer Gateway and VPN Connection will need to be recreated with AWS.
#

set vrouter trust-vr
set max-ecmp-routes 2
set protocol bgp YOUR_BGP_ASN
set hold-time 30
set ipv4 network 0.0.0.0/0
set ipv4 advertise-def-route
set enable
set neighbor 169.254.255.5 remote-as 7224
set neighbor 169.254.255.5 enable
set ipv4 neighbor 169.254.255.5 activate
exit
exit
set interface tunnel.2 protocol bgp
```

```
# Additional Notes and Questions
#  - Amazon Virtual Private Cloud Getting Started Guide:
#       http://docs.amazonwebservices.com/AWSVPC/latest/GettingStartedGuide
#  - Amazon Virtual Private Cloud Network Administrator Guide:
#       http://docs.amazonwebservices.com/AWSVPC/latest/NetworkAdminGuide
```

# How to Test the Configuration

You must first test the gateway configuration for each tunnel.

### To test the customer gateway configuration for each tunnel

1. On your customer gateway, determine if the BGP status is *Active*.
   It takes approximately 30 seconds for a BGP peering to become active.
2. Ensure that the customer gateway is advertising the default route (0.0.0.0/0) to the VPN gateway.

When properly established, your BGP peering should be receiving one route from the VPN gateway corresponding to the prefix that your VPC integration team specified for the VPC (e.g., 10.0.0.0/24). If the BGP peering is established, you are receiving a prefix, and you are advertising a prefix, your tunnel is configured correctly. Make sure both tunnels are in this state.

Next you must test the connectivity for each tunnel.

**Important**

For the connectivity test to work, you must configure any security group or network ACL in your VPC that filters traffic to the instance to allow inbound and outbound ICMP traffic.

### To test the end-to-end connectivity

1. Launch an instance of one of the Amazon Linux AMIs into your VPC. They're available in the Quick Start menu when you use the instance launch wizard in the AWS Management Console (for more information, refer to the Amazon Virtual Private Cloud Getting Started Guide).
2. After the instance is running, get its private IP address (e.g., 10.0.0.4). The AWS Management Console displays the address as part of the instance's details.
3. On a system in your home network, use the ping command with the instance's IP address. Make sure the computer you ping from is behind the customer gateway. A successful response should be similar to the following:

```
PROMPT> ping 10.0.0.4
Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.4:
Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),

Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

**Note**

If you ping an instance from your customer gateway router, ensure you are sourcing ping messages from an internal IP address, not a tunnel IP address. Some AMIs will not respond to ping messages from the tunnel IP addresses.

If your tunnels do not test successfully, see Troubleshooting (p. 48).

# Example: Generic Device

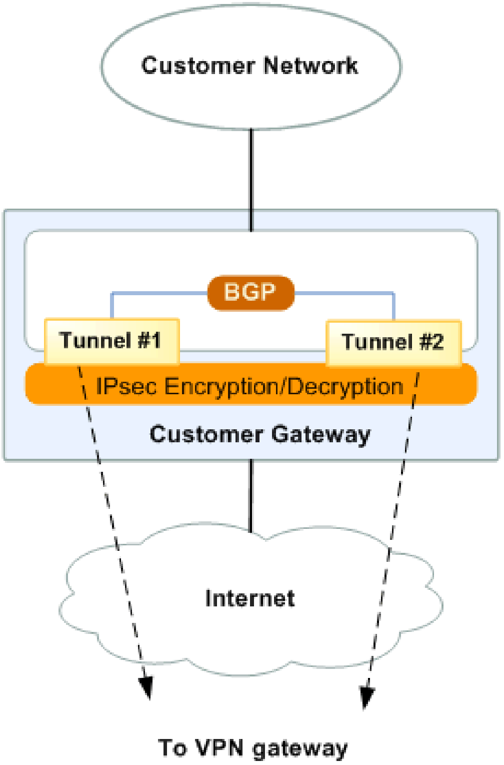
**Topics**

If your customer gateway isn't one of the types discussed earlier in this guide, your integration team will provide you with generic information that you can use to configure your customer gateway. This section contains an example of that information.

Along with the example configuration are two diagrams: one showing the high-level layout, and another giving details that match the example configuration. You should take the real configuration information that your integration team gives you and apply it to your customer gateway.

# High-Level Diagram

Following is the diagram showing the general details of your customer gateway. Note that the VPN connection consists of two separate tunnels: *Tunnel #1* and *Tunnel #2*. Two redundant tunnels provide an increased availability in the case of a device failure.



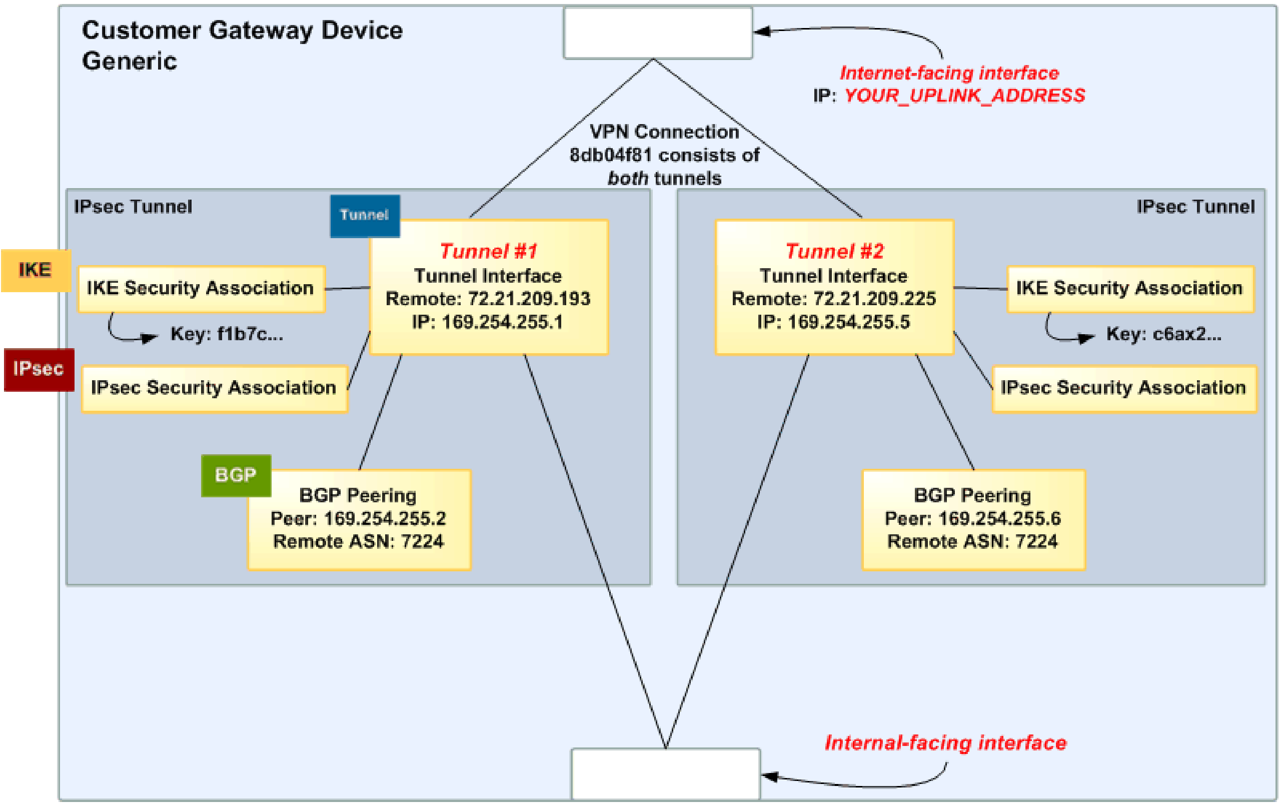
# Detailed Diagram and Example Configuration

Following is the diagram showing an example generic customer gateway. After the diagram is a corresponding example of the configuration information you should get from your integration team. It contains a set of information for each of the two tunnels you must configure.

The example configuration refers to two items that you must provide:

- **YOUR_UPLINK_ADDRESS—**The IP address for the Internet-routable external interface on the customer gateway (which must be static and can't be behind a device performing NAT)
- **YOUR_BGP_ASN—**The customer gateway's BGP ASN

The example configuration includes several dummy values we're using to help you understand how configuration works. For example, we're using dummy values for the VPN connection ID (44a8938f), VPN gateway ID (8db04f81), etc.; the IP addresses (e.g., 72.21.209.*, 169.254.255.*, etc.); and the remote ASN (7224). The actual configuration information you get will have real values in place of those dummy values.



```
Amazon Web Services
Virtual Private Cloud

VPN Connection Configuration
===============================================
AWS utilizes unique identifiers to manipulate the configuration of
a VPN Connection. Each VPN Connection is assigned a VPN identifier
and is associated with two other identifiers, namely the
Customer Gateway Identifier and the VPN Gateway Identifier.

Your VPN Connection ID       : vpn-44a8938f
Your VPN Gateway ID          : vgw-8db04f81
Your Customer Gateway ID     : cgw-b4dc3961

A VPN Connection consists of a pair of IPSec tunnel security associations (SAs).

It is important that both tunnel security associations be configured.


IPSec Tunnel #1
================================================
IKE
#1: Internet Key Exchange Configuration

Configure the IKE SA as follows
- Authentication Method    : Pre-Shared Key
- Pre-Shared Key           : plain-text-password1
- Authentication Algorithm : sha1
- Encryption Algorithm     : aes-128-cbc
- Lifetime                 : 28800 seconds
- Phase 1 Negotiation Mode : main
- Perfect Forward Secrecy  : Diffie-Hellman Group 2

IPsec
#2: IPSec Configuration

Configure the IPSec SA as follows:
- Protocol                 : esp
- Authentication Algorithm : hmac-sha1-96
- Encryption Algorithm     : aes-128-cbc
- Lifetime                 : 3600 seconds
- Mode                     : tunnel
- Perfect Forward Secrecy  : Diffie-Hellman Group 2

IPSec Dead Peer Detection (DPD) will be enabled on the AWS Endpoint. We
recommend configuring DPD on your endpoint as follows:
- DPD Interval             : 10
- DPD Retries              : 3

IPSec ESP (Encapsulating Security Payload) inserts additional
headers to transmit packets. These headers require additional space,
which reduces the amount of space available to transmit application data.
To limit the impact of this behavior, we recommend the following
configuration on your Customer Gateway:
- TCP MSS Adjustment       : 1396 bytes
- Clear Don't Fragment Bit : enabled
- Fragmentation            : Before encryption

Tunnel
#3: Tunnel Interface Configuration

Your Customer Gateway must be configured with a tunnel interface that is
associated with the IPSec tunnel. All traffic transmitted to the tunnel
interface is encrypted and transmitted to the VPN Gateway.

Additionally, the VPN Gateway and Customer Gateway establish the BGP
peering from your tunnel interface.

The Customer Gateway and VPN Gateway each have two addresses that relate
to this IPSec tunnel. Each contains an outside address, upon which encrypted
traffic is exchanged. Each also contain an inside address associated with
the tunnel interface.

The Customer Gateway outside IP address was provided when the Customer Gateway
was created. Changing the IP address requires the creation of a new
Customer Gateway.

The Customer Gateway inside IP address should be configured on your tunnel
interface.

Outside IP Addresses:
- Customer Gateway:        : YOUR_UPLINK_ADDRESS
- VPN Gateway              : 72.21.209.225

Inside IP Addresses
- Customer Gateway         : 169.254.255.2/30
- VPN Gateway              : 169.254.255.1/30

Configure your tunnel to fragment at the optimal size:
- Tunnel interface MTU     : 1436 bytes

BGP
#4: Border Gateway Protocol (BGP) Configuration:

The Border Gateway Protocol (BGPv4) is used within the tunnel, between the inside
IP addresses, to exchange routes from the VPC to your home network. Each
BGP router has an Autonomous System Number (ASN). Your ASN was provided
to AWS when the Customer Gateway was created.

BGP Configuration Options:
- Customer Gateway ASN     : YOUR_BGP_ASN
- VPN Gateway ASN          : 7224
- Neighbor IP Address      : 169.254.255.1
- Neighbor Hold Time       : 30

Configure BGP to announce the default route (0.0.0.0/0) to the VPN Connection
Gateway. The VPN Gateway will announce prefixes to your Customer
Gateway based upon the prefixes assigned in the creation of the VPC.

IPSec Tunnel #2
=====================================================
IKE
#1: Internet Key Exchange Configuration

Configure the IKE SA as follows
- Authentication Method    : Pre-Shared Key
- Pre-Shared Key           : plain-text-password2
- Authentication Algorithm : sha1
- Encryption Algorithm     : aes-128-cbc
- Lifetime                 : 28800 seconds
- Phase 1 Negotiation Mode : main
- Perfect Forward Secrecy  : Diffie-Hellman Group 2

IPsec
#2: IPSec Configuration

Configure the IPSec SA as follows:
- Protocol                 : esp
- Authentication Algorithm : hmac-sha1-96
- Encryption Algorithm     : aes-128-cbc
- Lifetime                 : 3600 seconds
- Mode                     : tunnel
- Perfect Forward Secrecy  : Diffie-Hellman Group 2

IPSec Dead Peer Detection (DPD) will be enabled on the AWS Endpoint. We
recommend configuring DPD on your endpoint as follows:
- DPD Interval             : 10
- DPD Retries              : 3

IPSec ESP (Encapsulating Security Payload) inserts additional
headers to transmit packets. These headers require additional space,
which reduces the amount of space available to transmit application data.
To limit the impact of this behavior, we recommend the following
configuration on your Customer Gateway:
- TCP MSS Adjustment       : 1396 bytes
- Clear Don't Fragment Bit : enabled
- Fragmentation            : Before encryption

Tunnel
#3: Tunnel Interface Configuration

Your Customer Gateway must be configured with a tunnel interface that is
associated with the IPSec tunnel. All traffic transmitted to the tunnel
interface is encrypted and transmitted to the VPN Gateway.

Additionally, the VPN Gateway and Customer Gateway establish the BGP
peering from your tunnel interface.

The Customer Gateway and VPN Gateway each have two addresses that relate
to this IPSec tunnel. Each contains an outside address, upon which encrypted
traffic is exchanged. Each also contain an inside address associated with
the tunnel interface.

The Customer Gateway outside IP address was provided when the Customer Gateway
was created. Changing the IP address requires the creation of a new
Customer Gateway.

The Customer Gateway inside IP address should be configured on your tunnel
interface.

Outside IP Addresses:
- Customer Gateway:        : YOUR_UPLINK_ADDRESS
- VPN Gateway              : 72.21.209.193

Inside IP Addresses
- Customer Gateway         : 169.254.255.6/30
- VPN Gateway              : 169.254.255.5/30

Configure your tunnel to fragment at the optimal size:
- Tunnel interface MTU     : 1436 bytes

BGP
#4: Border Gateway Protocol (BGP) Configuration:

The Border Gateway Protocol (BGPv4) is used within the tunnel, between the inside
IP addresses, to exchange routes from the VPC to your home network. Each
BGP router has an Autonomous System Number (ASN). Your ASN was provided
to AWS when the Customer Gateway was created.

BGP Configuration Options:
- Customer Gateway ASN     : YOUR_BGP_ASN
- VPN Gateway ASN          : 7224
- Neighbor IP Address      : 169.254.255.5
- Neighbor Hold Time       : 30

Configure BGP to announce the default route (0.0.0.0/0) to the VPN Connection
Gateway. The VPN Gateway will announce prefixes to your Customer
Gateway based upon the prefixes assigned in the creation of the VPC.

Additional Notes and Questions
========================================================

- Amazon Virtual Private Cloud Getting Started Guide:
http://docs.amazonwebservices.com/AWSVPC/latest/GettingStartedGuide
- Amazon Virtual Private Cloud Network Administrator Guide:
http://docs.amazonwebservices.com/AWSVPC/latest/NetworkAdminGuide
```

# How to Test the Configuration

You must first test the gateway configuration for each tunnel.

### To test the customer gateway configuration for each tunnel

1. On your customer gateway, determine if the BGP status is *Active*.
   It takes approximately 30 seconds for a BGP peering to become active.
2. Ensure that the customer gateway is advertising the default route (0.0.0.0/0) to the VPN gateway.

When properly established, your BGP peering should be receiving one route from the VPN gateway corresponding to the prefix that your VPC integration team specified for the VPC (e.g., 10.0.0.0/24). If the BGP peering is established, you are receiving a prefix, and you are advertising a prefix, your tunnel is configured correctly. Make sure both tunnels are in this state.

Next you must test the connectivity for each tunnel.

**Important**

For the connectivity test to work, you must configure any security group or network ACL in your VPC that filters traffic to the instance to allow inbound and outbound ICMP traffic.

### To test the end-to-end connectivity

1. Launch an instance of one of the Amazon Linux AMIs into your VPC. They're available in the Quick Start menu when you use the instance launch wizard in the AWS Management Console (for more information, refer to the Amazon Virtual Private Cloud Getting Started Guide).
2. After the instance is running, get its private IP address (e.g., 10.0.0.4). The AWS Management Console displays the address as part of the instance's details.
3. On a system in your home network, use the ping command with the instance's IP address. Make sure the computer you ping from is behind the customer gateway. A successful response should be similar to the following:

```
PROMPT> ping 10.0.0.4
Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.4:
Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),

Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

### Note

If you ping an instance from your customer gateway router, ensure you are sourcing ping messages from an internal IP address, not a tunnel IP address. Some AMIs will not respond to ping messages from the tunnel IP addresses.

If your tunnels do not test successfully, see Troubleshooting (p. 48).

# Troubleshooting

**Topics**

This section contains troubleshooting information to use if your tunnels aren't in the correct state when you test your customer gateway (for general testing instructions applicable to all customer gateways, see How to Test the Configuration (p. 46)).

## Cisco IOS Troubleshooting

This section presents information to use when troubleshooting the connectivity of a Cisco customer gateway. The information is presented in four parts: IKE, IPsec, tunnel, and BGP. You can troubleshoot the four parts in any order you'd like, but we suggest you start with IKE (at the bottom of the network stack) and move up.

### IKE

Use the following command. The response shown is for a customer gateway with IKE configured correctly.

```
router# show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
192.168.37.160  72.21.209.193   QM_IDLE           2001    0 ACTIVE
192.168.37.160  72.21.209.225   QM_IDLE           2002    0 ACTIVE
```

You should see one or more lines containing a *src* of the Remote Gateway specified in the tunnels. The *state* should be `QM_IDLE` and *status* should be `ACTIVE`. The absence of an entry, or any entry in another indicate that IKE is not configured properly.

For further troubleshooting, run the following commands to enable log messages that provide diagnostic information.

```
router# term mon
router# debug crypto isakmp
```

To disable debugging, use the following command.

```
router# no debug crypto isakmp
```

# IPsec

Use the following command. The response shown is for a customer gateway with IPsec configured correctly.

```
router# show crypto ipsec sa
interface: Tunnel1
    Crypto map tag: Tunnel1-head-0, local addr 192.168.37.160

    protected vrf: (none)
    local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
    remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
    current_peer 72.21.209.225 port 500
     PERMIT, flags={origin_is_acl,}
     #pkts encaps: 149, #pkts encrypt: 149, #pkts digest: 149
     #pkts decaps: 146, #pkts decrypt: 146, #pkts verify: 146
     #pkts compressed: 0, #pkts decompressed: 0
     #pkts not compressed: 0, #pkts compr. failed: 0
     #pkts not decompressed: 0, #pkts decompress failed: 0
     #send errors 0, #recv errors 0

     local crypto endpt.: 174.78.144.73, remote crypto endpt.: 72.21.209.225
     path mtu 1500, ip mtu 1500, ip mtu idb FastEthernet0
     current outbound spi: 0xB8357C22(3090512930)

     inbound esp sas:
      spi: 0x6ADB173(112046451)
       transform: esp-aes esp-sha-hmac ,
       in use settings ={Tunnel, }
       conn id: 1, flow_id: Motorola SEC 2.0:1, crypto map: Tunnel1-head-0
       sa timing: remaining key lifetime (k/sec): (4467148/3189)
       IV size: 16 bytes
       replay detection support: Y  replay window size: 128
       Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xB8357C22(3090512930)
       transform: esp-aes esp-sha-hmac ,
       in use settings ={Tunnel, }
       conn id: 2, flow_id: Motorola SEC 2.0:2, crypto map: Tunnel1-head-0
       sa timing: remaining key lifetime (k/sec): (4467148/3189)
       IV size: 16 bytes
       replay detection support: Y  replay window size: 128
       Status: ACTIVE
     outbound ah sas:

     outbound pcp sas:

interface: Tunnel2
     Crypto map tag: Tunnel2-head-0, local addr 174.78.144.73

     protected vrf: (none)
     local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
     remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
     current_peer 72.21.209.193 port 500
      PERMIT, flags={origin_is_acl,}
     #pkts encaps: 26, #pkts encrypt: 26, #pkts digest: 26
     #pkts decaps: 24, #pkts decrypt: 24, #pkts verify: 24
     #pkts compressed: 0, #pkts decompressed: 0
     #pkts not compressed: 0, #pkts compr. failed: 0
     #pkts not decompressed: 0, #pkts decompress failed: 0
     #send errors 0, #recv errors 0

     local crypto endpt.: 174.78.144.73, remote crypto endpt.: 72.21.209.193
     path mtu 1500, ip mtu 1500, ip mtu idb FastEthernet0
     current outbound spi: 0xF59A3FF6(4120526838)

     inbound esp sas:
      spi: 0xB6720137(3060924727)
       transform: esp-aes esp-sha-hmac ,
       in use settings ={Tunnel, }
       conn id: 3, flow_id: Motorola SEC 2.0:3, crypto map: Tunnel2-head-0
       sa timing: remaining key lifetime (k/sec): (4387273/3492)
       IV size: 16 bytes
       replay detection support: Y  replay window size: 128
       Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xF59A3FF6(4120526838)
       transform: esp-aes esp-sha-hmac ,
       in use settings ={Tunnel, }
       conn id: 4, flow_id: Motorola SEC 2.0:4, crypto map: Tunnel2-head-0
       sa timing: remaining key lifetime (k/sec): (4387273/3492)
       IV size: 16 bytes
       replay detection support: Y  replay window size: 128
       Status: ACTIVE

     outbound ah sas:

     outbound pcp sas:
```

For each tunnel interface, you should see both an *inbound esp sas* and *outbound esp sas*. Assuming an SA is listed ("spi: 0xF95D2F3C", for example) and the *Status* is `ACTIVE`, IPsec is configured correctly.

For further troubleshooting, use the following command to enable debugging.

```
router# debug crypto ipsec
```

Use the following command to disable debugging.

```
router# no debug crypto ipsec
```

# Tunnel

First, check that you have the necessary firewall rules in place. For a list of the rules, see If You Have a Firewall Between the Internet and Gateway (p. 7).

If your firewall rules are set up correctly, then continue troubleshooting with the following command.

```
router# show interfaces tun1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 169.254.255.2/30
  MTU 17867 bytes, BW 100 Kbit/sec, DLY 50000 usec,
    reliability 255/255, txload 2/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 174.78.144.73, destination 72.21.209.225
  Tunnel protocol/transport IPSEC/IP
  Tunnel TTL 255
  Tunnel transport MTU 1427 bytes
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "ipsec-vpn-92df3bfb-0")
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 1 packets/sec
  5 minute output rate 1000 bits/sec, 1 packets/sec
    407 packets input, 30010 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

Ensure the *line protocol* is up. Check that the tunnel source IP address, source interface and destination respectively match the tunnel configuration for the customer gateway outside IP address, interface, and VPN gateway outside IP address. Ensure that *Tunnel protection via IPSec* is present. Make sure to run the command on both tunnel interfaces. To resolve any problems here, review the configuration.

Also use the following command, replacing `169.254.255.1` with the inside IP address of your VPN gateway.

```
router# ping 169.254.255.1 df-bit size 1410

Type escape sequence to abort.
Sending 5, 1410-byte ICMP Echos to 169.254.255.1, timeout is 2 seconds:
Packet sent with the DF bit set
!!!!!
```

You should see 5 exclamation points.

For further troubleshooting, review the configuration.

# BGP

Use the following command.

```
router# show ip bgp summary
BGP router identifier 192.168.37.160, local AS number 65000
BGP table version is 8, main routing table version 8
2 network entries using 312 bytes of memory
2 path entries using 136 bytes of memory
3/1 BGP path/bestpath attribute entries using 444 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
Bitfield cache entries: current 1 (at peak 2) using 32 bytes of memory
BGP using 948 total bytes of memory
BGP activity 4/1 prefixes, 4/1 paths, scan interval 15 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
169.254.255.1   4  7224     363     323        8    0    0 00:54:21        1
169.254.255.5   4  7224     364     323        8    0    0 00:00:24        1
```

Here, both neighbors should be listed. For each, you should see a *State/PfxRcd* value of 1.

If the BGP peering is up, verify that your customer gateway router is advertising the default route (0.0.0.0/0) to the VPC.

```
router# show bgp all neighbors 169.254.255.1 advertised-routes
For address family: IPv4 Unicast
BGP table version is 3, local router ID is 174.78.144.73
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,

     r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Originating default network 0.0.0.0

Network             Next Hop            Metric   LocPrf Weight Path
*> 10.120.0.0/16    169.254.255.1          100        0   7224    i

Total number of prefixes 1
```

Additionally, ensure that you're receiving the prefix corresponding to your VPC from the VPN gateway.

```
router# show ip route bgp
     10.0.0.0/16 is subnetted, 1 subnets
B       10.255.0.0 [20/0] via 169.254.255.1, 00:00:20
```

For further troubleshooting, review the configuration.

## VPN Gateway Attachment

Make sure your VPN gateway is attached to your VPC. Your integration team does this with the AWS Management Console.

If you have questions or need further assistance, please use the Amazon VPC Discussion Forums.

# Juniper JunOS Troubleshooting

This section presents information to use when troubleshooting the connectivity of a Juniper customer gateway. The information is presented in four parts: IKE, IPsec, tunnel, and BGP. You can troubleshoot the four parts in any order you'd like, but we suggest you start with IKE (at the bottom of the network stack) and move up.

## IKE

Use the following command. The response shown is for a customer gateway with IKE configured correctly.

```
user@router> show security ike security-associations
Index   Remote Address  State  Initiator cookie  Responder cookie  Mode
4       72.21.209.225   UP     c4cd953602568b74  0d6d194993328b02  Main
3       72.21.209.193   UP     b8c8fb7dc68d9173  ca7cb0abaedeb4bb  Main
```

You should see one or more lines containing a Remote Address of the Remote Gateway specified in the tunnels. The *State* should be `UP`. The absence of an entry, or any entry in another state (such as `DOWN`) is an indication that IKE is not configured properly.

For further troubleshooting, enable the IKE trace options (as recommended in the example configuration information (see Example: Juniper JunOS Device (p. 19)). Then run the following command to print a variety of debugging messages to the screen.

```
user@router> monitor start kmd
```

From an external host, you can retrieve the entire log file with the following command.

```
scp username@router.hostname:/var/log/kmd
```

## IPsec

Use the following command. The response shown is for a customer gateway with IPsec configured correctly.

```
user@router> show security ipsec security-associations
Total active tunnels: 2
ID      Gateway         Port  Algorithm        SPI      Life:sec/kb Mon vsys
<131073 72.21.209.225   500   ESP:aes-128/sha1 df27aae4 326/ unlim   -   0
>131073 72.21.209.225   500   ESP:aes-128/sha1 5de29aa1 326/ unlim   -   0
<131074 72.21.209.193   500   ESP:aes-128/sha1 dd16c453 300/ unlim   -   0
>131074 72.21.209.193   500   ESP:aes-128/sha1 c1e0eb29 300/ unlim   -   0
```

Specifically, you should see at least two lines per Gateway address (corresponding to the Remote Gateway). Note the carets at the beginning of each line (< >) which indicate the direction of traffic for the particular entry. The output has separate lines for inbound traffic ("<", traffic from the VPN gateway to this customer gateway) and outbound traffic (">").

For further troubleshooting, enable the IKE traceoptions (for more information, see the preceding section about IKE).

# Tunnel

First, double-check that you have the necessary firewall rules in place. For a list of the rules, see If You Have a Firewall Between the Internet and Gateway (p. 7).

If your firewall rules are set up correctly, then continue troubleshooting with the following command.

```
user@router> show interfaces st0.1
  Logical interface st0.1 (Index 70) (SNMP ifIndex 126)
    Flags: Point-To-Point SNMP-Traps Encapsulation: Secure-Tunnel
    Input packets : 8719
    Output packets: 41841
    Security: Zone: Trust
    Allowed host-inbound traffic : bgp ping ssh traceroute
    Protocol inet, MTU: 9192
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
      Destination: 169.254.255.0/30, Local: 169.254.255.2
```

Make sure that the *Security: Zone* is correct, and that the *Local* address matches the customer gateway tunnel inside address.

Next, use the following command, replacing `169.254.255.1` with the inside IP address of your VPN gateway. Your results should look like the response shown here.

```
user@router> ping 169.254.255.1 size 1410 do-not-fragment
PING 169.254.255.1 (169.254.255.1): 1410 data bytes
64 bytes from 169.254.255.1: icmp_seq=0 ttl=64 time=71.080 ms
64 bytes from 169.254.255.1: icmp_seq=1 ttl=64 time=70.585 ms
```

For further troubleshooting, review the configuration.

# BGP

Use the following command.

```
user@router> show bgp summary
Groups: 1 Peers: 2 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0                 2          1          0          0          0          
0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn 
State|#Active/Received/Accepted/Damped...
169.254.255.1          7224          9         10       0       0        1:00 
1/1/1/0              0/0/0/0
169.254.255.5          7224          8          9       0       0          56 
0/1/1/0              0/0/0/0
```

For further troubleshooting, use the following command, replacing `169.254.255.1` with the inside IP address of your VPN gateway.

```
user@router> show bgp neighbor 169.254.255.1
Peer: 169.254.255.1+179 AS 7224 Local: 169.254.255.2+57175 AS 65000
  Type: External    State: Established    Flags: <ImportEval Sync>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Export: [ EXPORT-DEFAULT ]
  Options: <Preference HoldTime PeerAS LocalAS Refresh>
  Holdtime: 30 Preference: 170 Local AS: 65000 Local System AS: 0
  Number of flaps: 0
  Peer ID: 169.254.255.1    Local ID: 10.50.0.10       Active Holdtime: 30
  Keepalive Interval: 10         Peer index: 0
  BFD: disabled, down
  Local Interface: st0.1
  NLRI for restart configured on peer: inet-unicast
  NLRI advertised by peer: inet-unicast
  NLRI for this session: inet-unicast
  Peer supports Refresh capability (2)
  Restart time configured on the peer: 120
  Stale routes from peer are kept for: 300
  Restart time requested by this peer: 120
  NLRI that peer supports restart for: inet-unicast
  NLRI that restart is negotiated for: inet-unicast
  NLRI of received end-of-rib markers: inet-unicast
  NLRI of all end-of-rib markers sent: inet-unicast
  Peer supports 4 byte AS extension (peer-as 7224)
  Table inet.0 Bit: 10000
    RIB State: BGP restart is complete
    Send state: in sync
    Active prefixes:              1
    Received prefixes:            1
    Accepted prefixes:            1
    Suppressed due to damping:    0
    Advertised prefixes:          1
Last traffic (seconds): Received 4    Sent 8    Checked 4
Input messages:  Total 24     Updates 2       Refreshes 0     Octets 505
Output messages: Total 26     Updates 1       Refreshes 0     Octets 582
Output Queue[0]: 0
```

Here you should see *Received prefixes* and *Advertised prefixes* listed at 1 each. This should be within the *Table inet.0* section.

If the *State* is not `Established`, check the *Last State* and *Last Error* for details of what is required to correct the problem.

If the BGP peering is up, verify that your customer gateway router is advertising the default route (0.0.0.0/0) to the VPC.

```
user@router> show route advertising-protocol bgp 169.254.255.1

inet.0: 10 destinations, 11 routes (10 active, 0 holddown, 0 hidden)
  Prefix                  Nexthop              MED     Lclpref    AS path
* 0.0.0.0/0               Self                                    I
```

Additionally, ensure that you're receiving the prefix corresponding to your VPC from the VPN gateway.

```
user@router> show route receive-protocol bgp 169.254.255.1

inet.0: 10 destinations, 11 routes (10 active, 0 holddown, 0 hidden)
  Prefix                  Nexthop              MED     Lclpref    AS path
* 10.110.0.0/16           169.254.255.1        100                7224 I
```

## VPN Gateway Attachment

Make sure your VPN gateway is attached to your VPC. Your integration team does this with the AWS Management Console.

If you have questions or need further assistance, please use the Amazon VPC Discussion Forums.

# Juniper ScreenOS Troubleshooting

This section presents information to use when troubleshooting the connectivity of a Juniper ScreenOS-based customer gateway. The information is presented in four parts: IKE, IPsec, tunnel, and BGP. You can troubleshoot the four parts in any order you'd like, but we suggest you start with IKE (at the bottom of the network stack) and move up.

## IKE and IPsec

Use the following command. The response shown is for a customer gateway with IKE configured correctly.

```
ssg5-serial-> get sa
total configured sa: 2
HEX ID    Gateway         Port Algorithm     SPI      Life:sec kb Sta   PID
vsys
00000002<   72.21.209.225  500 esp:a128/sha1 80041ca4  3385 unlim A/-    -1 0
00000002>   72.21.209.225  500 esp:a128/sha1 8cdd274a  3385 unlim A/-    -1 0
00000001<   72.21.209.193  500 esp:a128/sha1 ecf0bec7  3580 unlim A/-    -1 0
00000001>   72.21.209.193  500 esp:a128/sha1 14bf7894  3580 unlim A/-    -1 0
```

You should see one or more lines containing a Remote Address of the Remote Gateway specified in the tunnels. The *Sta* should be `A/-` and the *SPI* should be a hexidecimal number other than `00000000`. Entries in other states indicate that IKE is not configured properly.

For further troubleshooting, enable the IKE trace options (as recommended in the example configuration information (see Example: Juniper ScreenOS Device (p. 30)).

## Tunnel

First, double-check that you have the necessary firewall rules in place. For a list of the rules, see If You Have a Firewall Between the Internet and Gateway (p. 7).

If your firewall rules are set up correctly, then continue troubleshooting with the following command.

```
ssg5-serial-> get interface tunnel.1
  Interface tunnel.1:
  description tunnel.1
  number 20, if_info 1768, if_index 1, mode route
  link ready
```

```
vsys Root, zone Trust, vr trust-vr
admin mtu 1500, operating mtu 1500, default mtu 1500
*ip 169.254.255.2/30
*manage ip 169.254.255.2
route-deny disable
bound vpn:
  IPSEC-1

Next-Hop Tunnel Binding table
Flag Status Next-Hop(IP)    tunnel-id  VPN

pmtu-v4 disabled
ping disabled, telnet disabled, SSH disabled, SNMP disabled
web disabled, ident-reset disabled, SSL disabled

OSPF disabled  BGP enabled  RIP disabled  RIPng disabled  mtrace disabled
PIM: not configured  IGMP not configured
NHRP disabled
bandwidth: physical 0kbps, configured egress [gbw 0kbps mbw 0kbps]
           configured ingress mbw 0kbps, current bw 0kbps
           total allocated gbw 0kbps
```

Make sure that you see *link:ready*, and that the *IP* address matches the customer gateway tunnel inside address.

Next, use the following command, replacing `169.254.255.1` with the inside IP address of your VPN gateway. Your results should look like the response shown here.

```
ssg5-serial-> ping 169.254.255.1
Type escape sequence to abort

Sending 5, 100-byte ICMP Echos to 169.254.255.1, timeout is 1 seconds
!!!!!
Success Rate is 100 percent (5/5), round-trip time min/avg/max=32/32/33 ms
```

For further troubleshooting, review the configuration.

# BGP

Use the following command.

```
ssg5-serial-> get vrouter trust-vr protocol bgp neighbor
Peer AS Remote IP       Local IP          Wt Status   State     ConnID Up/Down
-------------------------------------------------------------------------------
--
   7224 169.254.255.1   169.254.255.2    100 Enabled  ESTABLISH     10 00:01:01

   7224 169.254.255.5   169.254.255.6    100 Enabled  ESTABLISH     11 00:00:59
```

Both BGP peers should be listed as State: `ESTABLISH`, which means the BGP connection to the VPN gateway is active.

For further troubleshooting, use the following command, replacing `169.254.255.1` with the inside IP address of your VPN gateway.

```
ssg5-serial-> get vr trust-vr prot bgp neigh 169.254.255.1
peer: 169.254.255.1,  remote AS: 7224, admin status: enable
type: EBGP, multihop: 0(disable), MED: node default(0)
connection state: ESTABLISH, connection id: 18 retry interval: node de
fault(120s), cur retry time 15s
configured hold time: node default(90s), configured keepalive: node default(30s)
configured adv-interval: default(30s)
designated local IP: n/a
local IP address/port: 169.254.255.2/13946, remote IP address/port:
169.254.255.1/179
router ID of peer: 169.254.255.1, remote AS: 7224
negotiated hold time: 30s, negotiated keepalive interval: 10s
route map in name: , route map out name:
weight: 100 (default)
self as next hop: disable
send default route to peer: disable
ignore default route from peer: disable
send community path attribute: no
reflector client: no
Neighbor Capabilities:
  Route refresh: advertised and received
  Address family IPv4 Unicast:  advertised and received
force reconnect is disable
total messages to peer: 106, from peer: 106
update messages to peer: 6, from peer: 4
Tx queue length 0, Tx queue HWM: 1
route-refresh messages to peer: 0, from peer: 0
last reset 00:05:33 ago, due to BGP send Notification(Hold Timer Expired)(code
 4 : subcode 0)
number of total successful connections: 4
connected: 2 minutes 6 seconds
Elapsed time since last update: 2 minutes 6 seconds
```

If the BGP peering is up, verify that your customer gateway router is advertising the default route (0.0.0.0/0) to the VPC. Note that this command applies to ScreenOS version 6.2.0 and higher.

```
ssg5-serial-> get vr trust-vr protocol bgp  rib neighbor 169.254.255.1 advertised
i: IBGP route, e: EBGP route, >: best route, *: valid route
               Prefix         Nexthop    Wt  Pref   Med Orig    AS-Path
--------------------------------------------------------------------------------
--------
>i          0.0.0.0/0         0.0.0.0 32768   100     0  IGP
Total IPv4 routes advertised: 1
```

Additionally, ensure that you're receiving the prefix corresponding to your VPC from the VPN gateway. Note that this command applies to ScreenOS version 6.2.0 and higher.

```
ssg5-serial-> get vr trust-vr protocol bgp  rib neighbor 169.254.255.1 received
i: IBGP route, e: EBGP route, >: best route, *: valid route
               Prefix         Nexthop    Wt  Pref   Med Orig    AS-Path
--------------------------------------------------------------------------------
--------
>e*     10.0.0.0/16   169.254.255.1   100   100   100  IGP   7224
Total IPv4 routes received: 1
```

## VPN Gateway Attachment

Make sure your VPN gateway is attached to your VPC. Your integration team does this with the AWS Management Console.

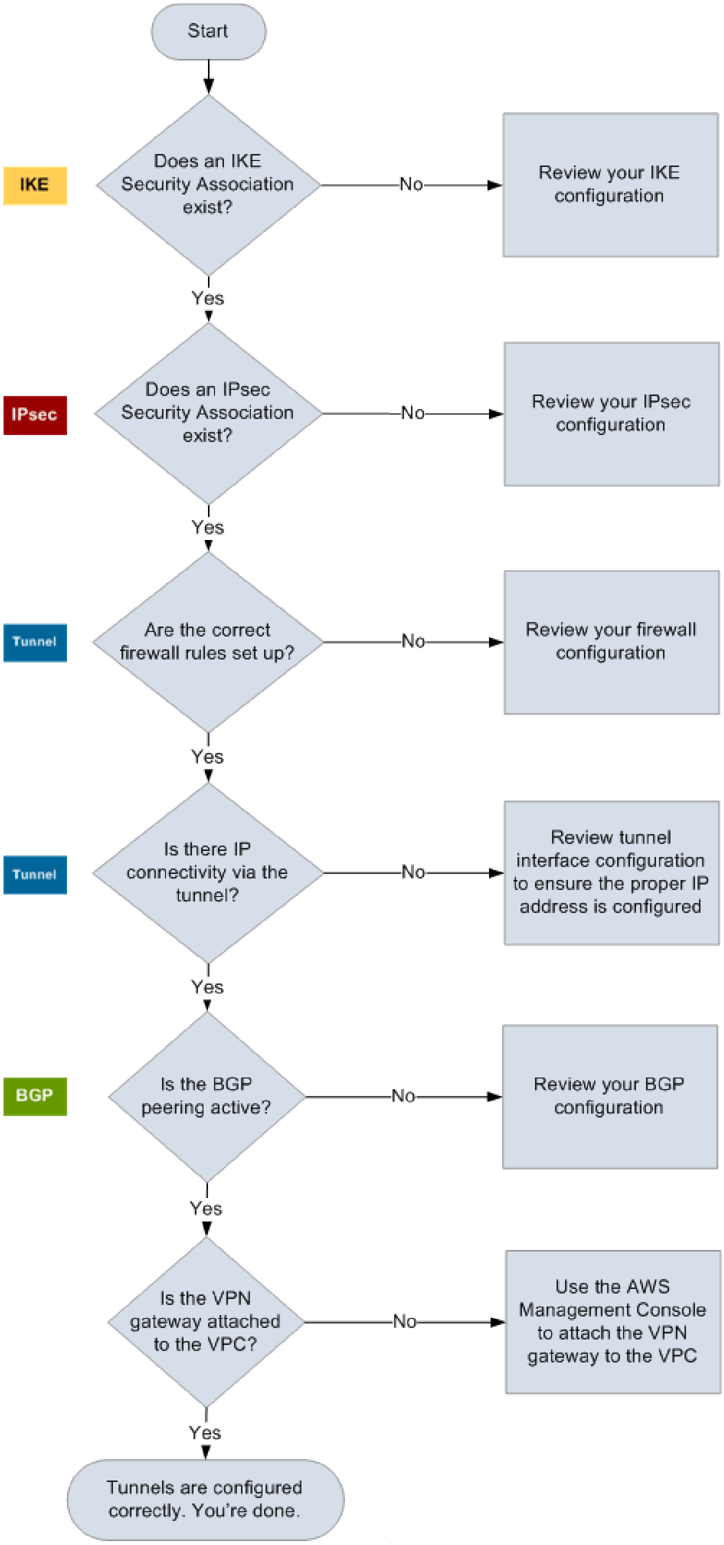
If you have questions or need further assistance, please use the Amazon VPC Discussion Forums.

# Generic Device Troubleshooting

The following diagram and table provide general instructions for troubleshooting a customer gateway other than a Cisco IOS or Juniper JunOS device.

**Tip**

In troubleshooting problems, you might find it useful to enable the debug features of your gateway device. Consult your gateway device vendor for details.


Start
IKE
Does an IKE Security Association exist?
No
Review your IKE configuration
Yes
IPsec
Does an IPsec Security Association exist?
No
Review your IPsec configuration
Yes
Tunnel
Are the correct firewall rules set up?
No
Review your firewall configuration
Yes
Tunnel
Is there IP connectivity via the tunnel?
No
Review tunnel interface configuration to ensure the proper IP address is configured
Yes
BGP
Is the BGP peering active?
No
Review your BGP configuration
Yes
Is the VPN gateway attached to the VPC?
No
Use the AWS Management Console to attach the VPN gateway to the VPC
Yes
Tunnels are configured correctly. You're done.

| | |
|---|---|
| IKE | Determine if an IKE Security Association exists.<br>An IKE security association is required to exchange keys that are used to establish the IPsec Security Association.<br>If no IKE security association exists, review your IKE configuration settings. You must configure the encryption, authentication, perfect-forward-secrecy, and mode parameters as listed in the customer gateway configuration.<br>If an IKE security association exists, move on to IPsec. |
| IPsec | Determine if an IPsec Security Association exists.<br>An IPsec security association is the tunnel itself. Query your customer gateway to determine if an IPsec Security Association is active. Proper configuration of the IPsec SA is critical. You must configure the encryption, authentication, perfect-forward-secrecy, and mode parameters as listed in the customer gateway configuration.<br>If no IPsec Security Association exists, review your IPsec configuration.<br>If an IPsec Security Association exists, move on to the tunnel. |
| Tunnel | Confirm the required firewall rules are set up (for a list of the rules, see If You Have a Firewall Between the Internet and Gateway (p. 7)). If they are, move forward.<br>Determine if there is IP connectivity via the tunnel.<br>Each side of the tunnel has an IP address as specified in the customer gateway configuration. The VPN gateway address is the address used as the BGP neighbor address. From your customer gateway, ping this address to determine if IP traffic is being properly encrypted and decrypted.<br>If the ping isn't successful, review your tunnel interface configuration to ensure the proper IP address is configured.<br>If the ping is successful, move on to BGP. |
| BGP | Determine if the BGP peering is active.<br>For each tunnel, do the following:<br>• On your customer gateway, determine if the BGP status is *Active* or *Established*. It may take approximately 30 seconds for a BGP peering to become active.<br>• Ensure that the customer gateway is advertising the default route (0.0.0.0/0) to the VPN gateway.<br>If the tunnels are not in this state, review your BGP configuration.<br>If the BGP peering is established, you are receiving a prefix, and you are advertising a prefix, your tunnel is configured correctly. Ensure both tunnels are in this state, and you're done. |
| | Make sure your VPN gateway is attached to your VPC. Your integration team does this with the AWS Management Console. |

If you have questions or need further assistance, please use the Amazon VPC Discussion Forums.

# Document History

This documentation is associated with the 2011-01-01 release of Amazon Virtual Private Cloud. This guide was last updated on 14 March 2011.

The following table describes the important changes since the last release of the Amazon VPC documentation set.

| Change | Description | Release Date |
|---|---|---|
| Updates to Configuration Templates | Updated the configuration templates to include information about encrypting packages after fragmentation. Also removed information about VRF from the Cisco configuration and removed information about the routing instance (RI) from the Juniper JunOS configuration. | In this release |