JUNIPER NETWORKS

Junos® OS

Group VPNs Feature Guide for Routing Devices

Release
14.1

Published: 2014-06-12

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Junos® OS Group VPNs Feature Guide for Routing Devices
14.1

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at http://www.juniper.net/techpubs/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at http://www.juniper.net/books.

Supported Platforms

For the features described in this document, the following platforms are supported:

- MX Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   system {
     scripts {
       commit {
         file ex-script.xsl;
       }
     }
   }
   interfaces {
     fxp0 {
       disable;
       unit 0 {
         family inet {
           address 10.0.0.1/24;
         }
       }
     }
   }
   ```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

   ```
   [edit]
   user@host# load merge /var/tmp/ex-script.conf
   load complete
   ```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   commit {
     file ex-script-snippet.xsl; }
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the *CLI User Guide*.

Documentation Conventions

Table 1 on page xi defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xi defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**

Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Encloses optional keywords or variables.	**stub <default-metric** ***metric*****>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

Indention and braces ({ })	Identifies a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.

Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page at the Juniper Networks Technical Documentation site at http://www.juniper.net/techpubs/index.html, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at https://www.juniper.net/cgi-bin/docbugreport/.
- E-mail—Send your comments to **techpubs-comments@juniper.net**. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit http://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/

- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: http://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://tools.juniper.net/SerialNumberEntitlementSearch/

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at http://www.juniper.net/cm/.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see http://www.juniper.net/support/requesting-support.html.

PART 1

Overview

CHAPTER 1

Introduction to Group VPNs

Group VPN Overview

Group VPN Technology Overview

This section explains the technological concepts of group VPN.

Understanding Group VPN

Group virtual private network (VPN) is a new category of VPN that eliminates the need for point-to-point VPN tunnels in a mesh architecture. It is a set of features that are necessary to secure IP multicast group traffic or unicast traffic over a private WAN that originates on or flows through a router.

Group VPN introduces the concept of a trusted group to eliminate point-to-point tunnels and their associated overlay routing. All group members share a common security association (SA), also known as a group SA (GSA). This enables group members to decrypt traffic that was encrypted by any other group member.

Group VPN provides the following advantages:

- Provides data security and transport authentication, helping to meet security compliance and internal regulation by encrypting all WAN traffic.
- Enables high-scale network meshes and eliminates complex peer-to-peer key management with group encryption keys.
- Maintains network intelligence such as full-mesh connectivity, natural routing path, and quality of service (QoS) in MPLS networks.
- Grants authenticated membership control with a centralized key server.
- Helps to ensure low latency and jitter by enabling full-time, direct communications between sites, without requiring transport through a central hub.
- Reduces traffic loads on customer premises equipment (CPE) and provider-edge (PE) encryption devices by using the core network for replication of multicast traffic, avoiding packet replication at each individual peer site.

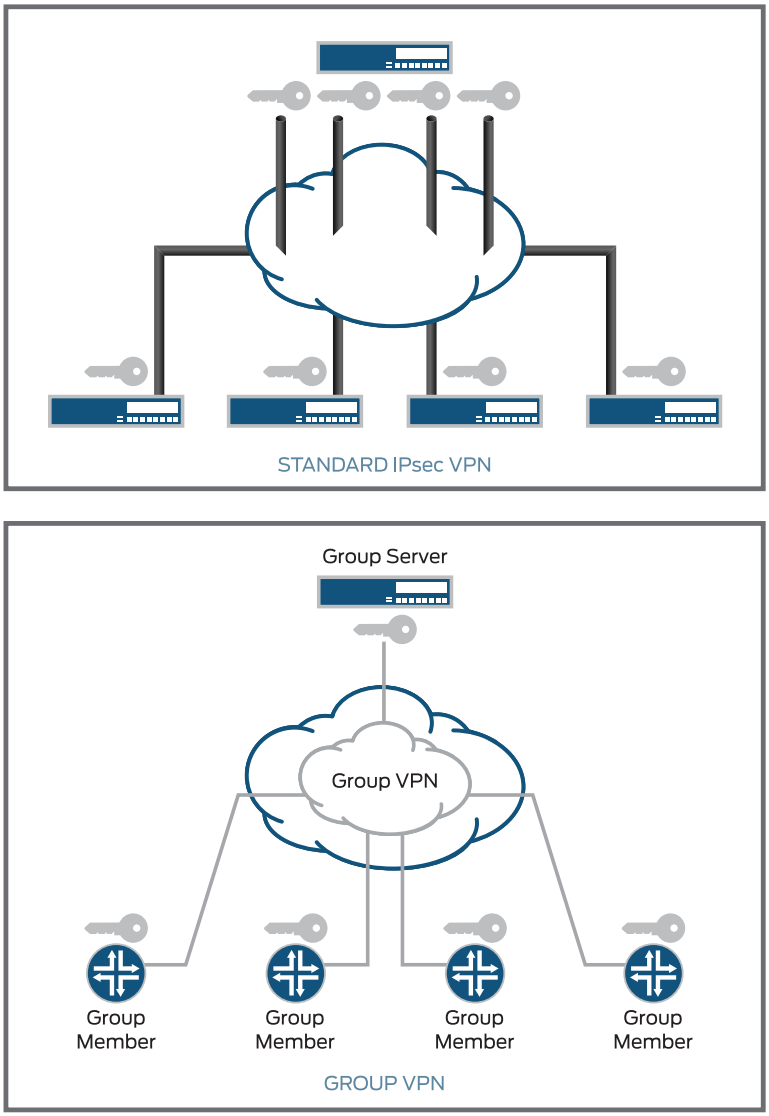
Group VPN and Standard IPsec VPN

A group VPN is built on standards-based technologies that integrate routing and encryption together in the network. An IPsec security SA is a unidirectional agreement between VPN participants that defines the rules to use for authentication and encryption algorithms, key exchange mechanisms, and secure communications.

Traditional IPsec VPN deployments tackle the problem of securing traffic between gateways in the network by creating an overlay network based on the use of point-to-point tunnels. Traffic carried over these tunnels is normally encrypted and authenticated in order to provide data integrity and confidentiality. Secure group members are managed through the Group Domain of Interpretation protocol (GDOI). The GDOI solution takes a different approach by disassociating the encryption and authentication problem from the transport. By doing this, GDOI-based solutions provide a way to encrypt branch-to-branch communications without the need to configure branch-to-branch tunnels.

With current VPN implementations, the SA is a point-to-point tunnel between two routers. A group VPN extends the IPsec architecture to support SAs that are shared by a group of routers (see Figure 1 on page 5). A key server distributes keys and policies to all registered and authenticated member routers. By distributing policies from a centralized point and by sharing the same group security association (the entire group has a single Phase 2 IPsec SA) with authenticated group members, key distribution and management are greatly simplified.

Figure 1: Standard IPsec VPN and Group VPN



Group VPN is a client/server architecture. All members have a unique Phase 1 IKE SA with the key server. Hence, if there are **n** members, there is a total of **n** Phase 1 IKE SAs. However, the entire group shares a single Phase 2 SA.

In traditional IPsec, tunnel endpoint addresses are used as a new packet source and destination. The packet is then routed over the IP infrastructure, using the encrypting gateway source IP address and the decrypting gateway destination IP address. In the case of group VPN, IPsec-protected data packets encapsulate the original source and destination packet addresses of the host in the outer IP header to preserve the IP address. The biggest advantage of tunnel header preservation is the ability to route encrypted packets using the underlying network routing infrastructure.

Thus, with group VPNs, any-to-any connectivity is achieved by preserving the original source and destination IP addresses in the outer header. Secure multicast packets are

replicated in the same way as cleartext multicast packets in the core network. The external IP header in the group VPN is an exact copy of the IP header of the original packet within the ESP header versus the external IP header in a traditional VPN that contains the IP addresses of the VPN gateways.

Table 3: Group VPN vs Traditional Point-to-Point IPsec

Feature	Traditional Point-to-Point IPsec Tunnels	Group VPN
Scalability	IKE/IPsec tunnels between each pair of peers.	Scalable architecture. Single SA and key pair used for entire any-to-any group.
Any-to-any instant connectivity	Can't be done to scale.	Can be done to high-scale.
Overlay routing	Supports overlay routing.	No overlays-native routing.
IP Header Preservation	New IP Header added to original packed results in Limited advanced quality-of-service (QoS).	Keeps original IP header on IPsec packet, and preserves advanced QoS.

Understanding the GDOI Protocol

The Group Domain of Interpretation (GDOI) protocol described in RFC 6407 is used to distribute a set of cryptographic keys and policies to a group of devices. GDOI is defined as the Internet Security Association Key Management Protocol (ISAKMP) Domain of Interpretation (DOI) for group key management. In a group management model, the GDOI protocol operates between a group member and a group controller or key server (GC/KS) and manages group security associations (GSAs) and group-keys for a set of security participants. The ISAKMP defines two phases of negotiation. GDOI is a Phase 2 protocol protected by a Phase 1 ISAKMP security association. IKEv1 is specified in RFC 6407 as a Phase 1 protocol.

GDOI introduces two different encryption keys:

- Key encryption key (KEK)—Used to secure the control plane. KEK is the name of the key used by the group members to decrypt rekey messages from the GC/KS. This key is part of the Security Association Key Encryption Key (SAK).
- Traffic encryption key (TEK)—Used to secure the data plane. TEK is the name of the key used by the group members to encrypt or decrypt communication between other group members. This key is part of the Security Association Transport Encryption Key (SAT).

As with standard IPsec, all keys have a lifetime and have to be rekeyed. The keys distributed through GDOI are group keys and are used by the entire group.

The GSAs and key management are handled through two types of GDOI exchanges:

- **groupkey-pull**—This exchange allows a member to request SAs and keys shared by the group from the server.

In the pull method, the group member requests the group SA and policy from the key server. This request is protected over the IKE SA.

The **groupkey-pull** is the first exchange in the GDOI protocol and is used for group member registration with the GC/KS. The group member specifies the group with which it wants to register, and the GC/KS sends all necessary GSAs and keys to the group member if the member is authorized to join the group. The complete exchange is secured by a Phase 1 SA (IKEv1 SA), which is established with IKEv1 before the **groupkey-pull** exchange begins. The **groupkey-pull** is part of Phase 2 of the GDOI protocol.

- **groupkey-push**—This exchange is a single rekey message that allows the server to send group SAs and keys to members before existing group SAs expire. Rekey messages are unsolicited messages sent from the server to members.

  The **groupkey-push** is the second exchange in the GDOI protocol and is used to send new GSAs and keys from the GC/KS to the group members of a group (rekeying). This exchange is secured by a SA KEK (SAK), which is installed during the **groupkey-pull** exchange. The **groupkey-push** is part of Phase 2 of the GDOI protocol.

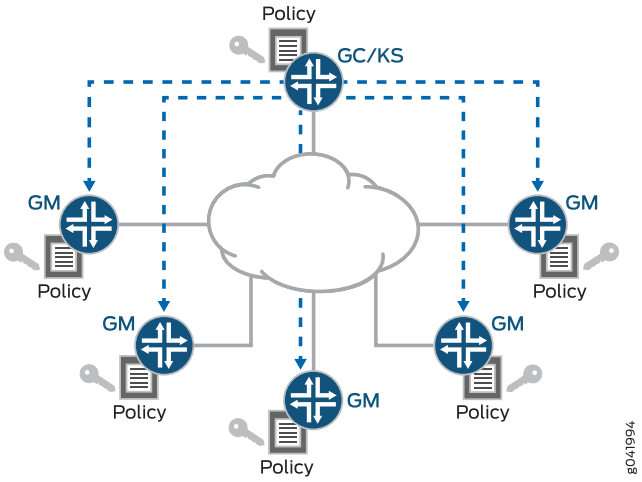
NOTE: Junos OS uses only the groupkey-pull exchange for group member rekeying. The groupkey-push exchange is not supported. For information about group VPN rekeying, see "Rekeying a Group Member" on page 12.

GDOI Protocol and Group VPN

Group VPN is the name of the security technology from Juniper Networks. Group VPN uses the GDOI protocol (RFC 6407) as a base, in addition to other functionalities.

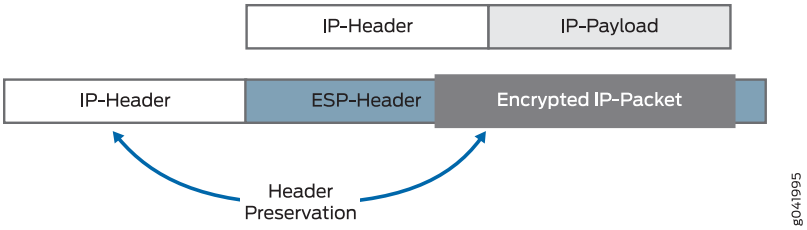
The group VPN technology is based on the GDOI protocol to handle the most important functionality. This protocol is specified in RFC 6407 and defines an ISAKMP Domain of Interpretation (DOI) to manage GSAs and keys for a group of security participants. Thus all members of the group share identical information to encrypt and decrypt traffic among each other. The creation, management, and distribution of GSAs and group keys are centralized and performed by the GC/KS. Figure 2 on page 8 provides a brief overview of the group VPN functionality using GDOI.

Figure 2: Group VPN Using GDOI



The group members use the Encapsulating Security Payload (ESP) protocol in tunnel mode to secure the traffic. However, in group VPN the tunnel mode is modified. Because there is no direct association between the group members, it is not necessary to use special IP addresses in the outer IP header (that is, IP addresses of IPsec gateways). Every group member can decrypt the traffic of every other group member. Thus, the inner IP-Header is copied to the outer IP-Header, and the underlying routing infrastructure and QoS infrastructure can be used. This feature is called Header Preservation and is shown in Figure 3 on page 8.

Figure 3: Header Preservation



To get GSAs and group keys, the group member must register with the GC/KS for a specific group. The result is an IKEv1 SA, which is only needed to secure the registration process. After the registration, the group member has all the information to communicate with the other group members (SAT), as well as the information to successfully decrypt the rekeying messages (SAK). The GC/KS send out rekeying messages before either the SAT or SAK lifetime expires. It is also possible to send a SAT update as well as a SAK update in the same rekey message. The IKEv1 SA is no longer needed and is deleted after the lifetime expires (no IKEv1-rekeying).

Group VPN Traffic

The group VPN traffic includes:

- Control-plane-traffic—Traffic from the group members to the GC/KS in the group VPN deployment with the GDOI protocol only.

- Data-plane-traffic—Traffic between the group members in the group VPN deployment with the ESP protocol only, that is already known from IPsec.

Group Security Association

Unlike traditional IPsec encryption solutions, group VPN uses the concept of group security association (GSA). GSA is similar to an IPsec SA in terms of functionality. GSAs are shared among all group members of a common GDOI group. All members in the group VPN group can communicate with each other using a common encryption policy and a shared GSA. With a common encryption policy and a shared GSA, there is no need to negotiate IPsec between group members. This reduces the resource load on the IPsec routers. Traditional group member scalability (number of tunnels and associated SA) does not apply to group VPN group members.

Group Controller/Key Server

A group controller or key server (GC/KS) is a device used for creating and maintaining the group VPN control plane. It is responsible for creation and distribution of GSAs and group keys. All information the group members need to communicate with other group members is provided by the GC/KS. All encryption policies, such as interesting traffic, encryption protocols, security association, rekey timers, and so on, are centrally defined on the GC/KS and are pushed down to all group members at registration time. Group members authenticate with the GC/KS using IKE Phase 1 and then download the encryption policies and keys required for group VPN operation. The GC/KS is also responsible for refreshing and distributing the keys. Unlike traditional IPsec, interesting traffic defined on the GC/KS is downloaded to every group member, whether or not the group member owns that network.

NOTE: **The GC/KS functionality is not supported on MX Series routers. The MX Series routers that are configured as group members can connect with Cisco GC/KS only.**

Group Member

A group member is a device used for the traffic encryption process and is responsible for the actual encryption and decryption of data traffic. A group member is configured with IKE Phase 1 parameters and GC/KS information. Encryption policies are defined centrally on the GC/KS and downloaded to the group member at the time of registration. Based on these downloaded policies, the group member determines whether traffic needs to be encrypted or decrypted and what keys to use. The group members use the ESP protocol for securing transport data sent to other group members.

From a functionality point of view, a group member is similar to an IPsec gateway. However, the SAs in normal IPsec are handled per connection between both IPsec gateways and in GDOI, whereas in group VPN the SAs are provided by the GC/KS on a per group basis.

The group member registers with the key server to get the IPsec SA or SAs that are necessary to communicate with the group. The group member provides the group ID to

the key server to get the respective policy and keys for this group. The group members re-register with the server before the current IPsec SAs expire as part of the rekeying process, so that there is no loss of traffic.

NOTE: Only **groupkey-pull** exchange is used for rekeying. The **groupkey-push** from the GC/KS is not supported. For information about group VPN rekeying, see "Rekeying a Group Member" on page 12.

Group VPN Implementation Overview

This section explains the Juniper Networks solution for implementing group VPN.

Enabling Group VPN

Service set is used to enable group VPN on a particular interface.

Configuring the Service Set

The group VPN is configured inside a service set using the **ipsec-group-vpn** statement at the **[edit services service-set service-set-name]** hierarchy level.

Sample Service Set Configuration

```
[edit services]
service-set service-set-name {
   interface-service {
      service-interface pic-interface;
   }
}
ipsec-group-vpn vpn-name;
```

NOTE:

- Only one group member can be configured per service set.
- Next-hop style service set is not supported with Group VPN.

Applying the Service Set

A service set is applied at the interface level.

Sample Applying Service Set Configuration

```
[edit interfaces]
interface-name {
   unit 0 {
      family inet {
         service {
            input {
               service-set service-set-name;
            }
            output {
               service-set service-set-name;
            }
         }
         address 10.0.30.2/30;
      }
   }
}
```

Packet Steering

The interface-style service set configuration is used to steer traffic from the Packet Forwarding Engine to the PIC. Packets received on an interface with a service set pointing to the group VPN object are forwarded to the PIC by being injected into the corresponding service interface.

Registering a Group Member

The group member registration to the server starts when the **ipsec-group-vpn** statement is configured for a service set and the service interface is up. When the service interface

goes down, all GSAs associated with this PIC are cleared, and no registration is triggered for these group VPNs until the PIC comes up.

Group member registration involves establishing IKE SA with the GC/KS followed by a **groupkey-pull** exchange to download the SAs and the traffic keys for the specified group identifier.

NOTE: Junos OS does not support traffic-based SA negotiation triggering for group VPNs.

Rekeying a Group Member

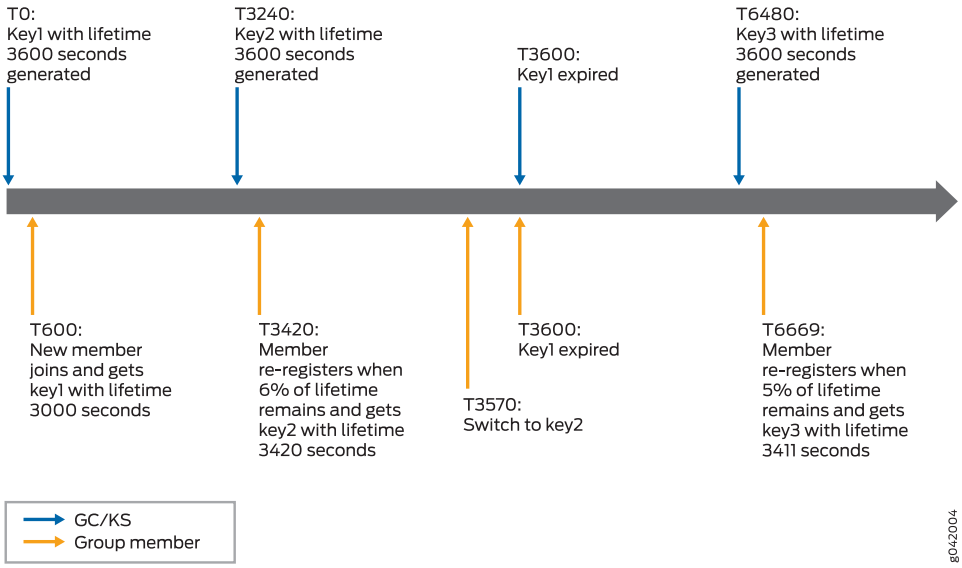
Junos OS group VPN does not support server-to-member rekeying. If rekeying is done from the GC/KS, the key is ignored by the group member.

For group member rekeying, the group members re-register with the GC/KS at the end of the soft lifetime expiry. After rekeying, both the old key and the new key can be used for decryption. However, the new key is not used for encryption until 30 seconds of lifetime of the old key is remaining.

Taking as an example, the GC/KS is configured with a lifetime of 3600 seconds and is connected to one group member without retransmit. Based on the server configuration, the GC/KS generates a new key when 10 percent of the lifetime is remaining. The group member, however, re-registers with the GC/KS when 5 percent to 7 percent of the lifetime is remaining.

Figure 4 on page 12 represents the rekeying process between the GC/KS and the group member.

Figure 4: Group Member Rekeying



Authenticating a Group Member

Junos OS does not provide Public Key Infrastructure (PKI) support for group VPNs, and as a result, pre-shared keys are used for group member authentication.

Fragmenting Group VPN Traffic

Because of the header preservation functionality and the usage of the underlying routing infrastructure, it is necessary to fragment the packets before encryption occurs (if it cannot be prevented).

Hence, pre-fragmentation is supported and is recommended for all deployments.

To avoid post-fragmentation, set the **clear**, **set**, and **copy** options for the DF bit in the group VPN configuration.

Based on this flag setting, the IPsec header has either the **df-bit** set to **clear**, **set**, or **copy** from the inner packet.

NOTE: The DF bit has the clear option set as default.

Sample DF Bit Configuration

```
[edit]
security {
   group-vpn {
      member {
         ipsec {
            vpn vpn-group-name {
               df-bit clear;
            }
         }
      }
   }
}
```

Encrypting Group VPN Traffic

Group members encrypt traffic based on the GSAs and keys provided by the GC/KS. The group VPN encryption path is as follows:

1. Packet received by the Packet Forwarding Engine is checked against a flow match. If a match is found, the packet is further processed and transmitted.
2. If a match is not found, a rule lookup is performed. If a match is found, a flow is created, and the packet is further processed and transmitted.
3. If the rule lookup fails, the packet is dropped.

NOTE: GSA is not triggered during packet processing.

Decrypting Group VPN Traffic

After registration is successful and group VPN SAs are installed, an ESP session is created. Unlike IPsec VPN which creates an ESP session with the source and destination IP to be the gateway address, group VPN creates the ESP session with a zero source and destination IP. Because the ESP session is already created at SA installation, packets are expected to match the existing ESP session.

The group VPN decryption path is as follows:

1. Packet received by the Packet Forwarding Engine undergoes a fragmentation check. If the packet is fragmented, it is assembled for further processing.
2. After packet assembling or if the packet is not fragmented, a zero source and destination IP is used in the 5-tuple decrypt flow lookup. If a match is found, the packet is further processed and transmitted.
3. If the decrypt flow lookup fails, the packet is checked against an SPI flow with zero source and destination IP.
4. If the SPI flow lookup fails, the packet is dropped.
5. If an SPI flow match is found, a decrypt flow is created to avoid the SPI flow lookup for subsequent packets.

Configuring a Routing Instance for Group VPN

Routing instances are supported for both control and data traffic. To enable routing instance support on control plan traffic for a group member to reach the GC/KS in a given VRF routing instance, add the **routing-instance** statement at the **[edit security group-vpn member ike gateway *gateway-name* local-address *address*]** hierarchy.

No additional CLI is required to support a routing instance for data plane packets, as it is determined based on the media interface on which the service set is applied.

Establishing Multiple Groups, Policies, and SAs

Junos OS provides support for one group VPN per service set. However, multiple service sets can be created to support multiple groups in a routing instance. Multiple SAs can be configured per group. However, multiple policies for the same traffic key/SPI is not supported. If the server sends two policies for the same TEK, then they must be paired to be accepted, for instance, A-B and B-A where A and B are IP addresses or subnets. If multiple unpaired policies for a given TEK are received, registration fails and a system log message is generated.

Connecting with Multiple Cooperative GC/KS

For a group member to work with a GC/KS in the cooperative mode, the configuration is extended to allow a maximum of four servers in the server list.

During rekeying, the group member tries to connect to the GC/KS. When the connection to the GC/KS fails, the group member tries to reconnect to the GC/KS. After three retries with an interval of 10 seconds, if the connection to the GC/KS is not restored, the group

member tries to establish a connection with the next available server on the server list. This process is repeated until the group member connects to a GC/KS. During this time, the unexpired GDOI IPsec SAs on the group members are not cleaned up, so group VPN traffic is not affected. The time gap between rekeying and hard lifetime expiry provides sufficient time for the group members to connect to the next available server, in such cases.

Changing Group VPN Configuration

Most group VPN configuration changes result in deleting existing IKE SA and GDOI SA and re-registration, which triggers both phase 1 and SA download with traffic keys.

Bypassing Group VPN Configuration

The service filter needs to be configured on the interface on which the service set is applied if certain traffic like a routing protocol needs to bypass a group VPN. The packet matching service filter will not come to the PIC for service processing and is directly forwarded to the Routing Engine.

Sample Service Set Filter Configuration

```
[edit interfaces]
interface-name {
  unit 0 {
    family inet {
      service {
        input {
          service-set service-set-name service-filter filter-name;
        }
        output {
          service-set service-set-name service-filter filter-name;
        }
      }
    }
  }
}
```

Supported GDOI IPsec Parameters

Every GDOI group has a unique ID. It is used as a common base between GC/KS and the group member to communicate about GSAs and group keys.

During the registration process, the GC/KS sends Security Association Transport Encryption Keys (SATs) to the group members. All parameters regarding the whole group security policy are configured on the GC/KS. The SAT is used by the group members to protect the traffic exchanged among each other. Table 4 on page 16 shows the parameters of the SAT.

Table 4: SAT Parameters

Parameters	Supported Values
Encryption	• DES-CBC • 3DES-CBC • AES-CBC 128 • AES-CBC 192 • AES-CBC 256
Integrity	• HMAC-MD5-96 • HMAC-SHA1-96
Lifetime	Any supported value

Besides the cryptographic algorithms, the traffic, which should be encrypted by the group members, is part of the SAT policy (traffic selector).

The following statements can be used on a Juniper Networks group member. Thus, the addresses have to be specified under the IKE hierarchy level. The enumeration is also prioritized. Thus, in the following example configuration, KS1 is contacted before KS2.

Sample GDOI IPsec Parameters Configuration

```
[edit security]
group-vpn {
   member {
      ike {
         gateway gateway-name {
            ike-policy policy-name;
            server-address <IP_KS1> <IP_KS2> <IP_KS2> <IP_KS4>;
            local-address <IP_GM> routing-instance routing-instance-name;
         }
      }
      ipsec {
         vpn vpn-group-name {
            ike-gateway gateway-name;
            group group-ID;
            match-direction output;
         }
      }
   }
}
```

Supported GDOI IKEv1 Parameters

The group members use only IKEv1 during the registration process in the group VPN environment. Table 5 on page 17 provides an overview of the defined parameters of the IKEv1 SA.

Table 5: IKEv1 SA Parameters of Group Member

Parameter	Supported Values
Encryption	• DES-CBC • 3DES-CBC • AES-CBC 128 • AES-CBC 192 • AES-CBC 256
Authentication	Pre-shared key (minimum 20 signs)
Integrity	• MD5 • SHA1
Diffie-Hellman Group	• Group 1 • Group 2 • Group 5 • Group 14
Lifetime	Any supported value

The above-mentioned IKEv1 standards are configured as follows:

Sample IKEv1 Configuration

```
[edit security]
group-vpn {
   member {
      ike {
         proposal proposal-name {
            authentication-algorithm sha1;
            authentication-method pre-shared-keys;
            dh-group group5;
            encryption-algorithm aes-128-cbc;
            lifetime-seconds 3600;
         }
         policy policy-name {
            mode main;
            proposals proposal-name;
            pre-shared-key ascii-text "SECRET DATA";
         }
      }
   }
}
```

Applying Dynamic Policies

The **input** and **output** options under the **ipsec-group-vpn** statement specify if the dynamic policies received from the server are used when the interface on which the service set is applied is the incoming or outgoing interface. This provides flexibility to specify different rules in the incoming and outgoing directions.

Supporting TOS and DSCP

Type of service (TOS) and DiffServ Code Points (DSCP) bit are copied from the inner packet to the ESP packet.

Interoperability of Group Members

Cisco's implementation of GDOI is called Group Encryption Transport (GET) VPN. While group VPN in Junos OS and Cisco's GET VPN are both based on RFC 6407, *The Group Domain of Interpretation*, there are some implementation differences that you need to be aware of when deploying GDOI in a networking environment that includes both Juniper Networks security and routing devices and Cisco routers. For more information, see the current Junos OS release notes.

Group VPN interoperability is as follows:

- Junos OS provides minimal interoperability support with Cisco IOS GC/KS support.
- The MX Series group member is able to communicate with the SRX Series group member and Cisco group member.
- Junos OS does not provide support for group VPN interoperability with the SRX Series group VPN server.

Table 6: Group VPN Interoperability

Group Member (GM)	SRX GM	MX GM	Cisco GM	SRX GS	Cisco GS
MX GM	Yes	Yes	Yes	No	Yes

Following are some of the known issues with Cisco IOS GC/KS:

- The unicast **groupkey-push** message does not work because the proprietary ACK is expected by the Cisco GC/KS following unicast push.

  As a workaround, the **groupkey-pull** is used for rekey.

- The deny policy is used on a Cisco server to add an exception to the group policy.

  As a workaround, this can be done by configuring firewall rules on an MX Series group member. Also, Junos OS group members can work with the deny policy by not failing the negotiation and simply ignoring the contents. This allows system administrators to easily manage networks where both Cisco group members and Junos OS group members co-exist.

Group VPN Limitations

Junos OS group VPN does not provide support for the following:

- GDOI **groupkey-push** exchange. Hence, both unicast and multicast push are not supported.
- Multicast traffic

- Post-fragmentation of packets
- GDOI SNMP MIBs
- Anti-replay
- GAP payload
- Protocol and port in the policies sent by the server. The group member honors only the IP address/subnet specified in the policy.
- Multiple unpaired policies for the same traffic key/SPI
- Overlapping of both local and remote IP across routing instances in an IKE gateway configuration
- Overlapping group VPN policies that can result in mismatched SAs
- IPv6 for control and data traffic
- Co-existence of IPsec and group VPN on the same service set
- Co-existence of services like NAT and ALG on the same service set. NAT and group VPN can co-exist on different service sets. However, they cannot co-exist on the same service set.
- Site To Site (S2S) VPN and Dynamic End Point (DEP) VPN can co-exist with group VPN on different service sets. However, they cannot co-exist on the same service set.
- Multiple groups on same service set
- Group member support with SRX GC/KS
- Logical Key Hierarchy (LKH)
- Graceful restart
- High availability
- Unified ISSU
- PKI support for authentication
- AMS interface and load balancing support
- Multiple groups per service set
- Same gateway for multiple groups, wherein the same local and remote address pair cannot be used for multiple groups.

Related Documentation

- Example: Configuring Group VPNs on Routing Devices on page 39

PART 2

Configuration

CHAPTER 2

Configuring Group VPNs

Example: Configuring Group VPN on Routing Devices

Group VPN Overview

Group VPN Technology Overview

This section explains the technological concepts of group VPN.

Understanding Group VPN

Group virtual private network (VPN) is a new category of VPN that eliminates the need for point-to-point VPN tunnels in a mesh architecture. It is a set of features that are necessary to secure IP multicast group traffic or unicast traffic over a private WAN that originates on or flows through a router.

Group VPN introduces the concept of a trusted group to eliminate point-to-point tunnels and their associated overlay routing. All group members share a common security

association (SA), also known as a group SA (GSA). This enables group members to decrypt traffic that was encrypted by any other group member.

Group VPN provides the following advantages:

- Provides data security and transport authentication, helping to meet security compliance and internal regulation by encrypting all WAN traffic.
- Enables high-scale network meshes and eliminates complex peer-to-peer key management with group encryption keys.
- Maintains network intelligence such as full-mesh connectivity, natural routing path, and quality of service (QoS) in MPLS networks.
- Grants authenticated membership control with a centralized key server.
- Helps to ensure low latency and jitter by enabling full-time, direct communications between sites, without requiring transport through a central hub.
- Reduces traffic loads on customer premises equipment (CPE) and provider-edge (PE) encryption devices by using the core network for replication of multicast traffic, avoiding packet replication at each individual peer site.

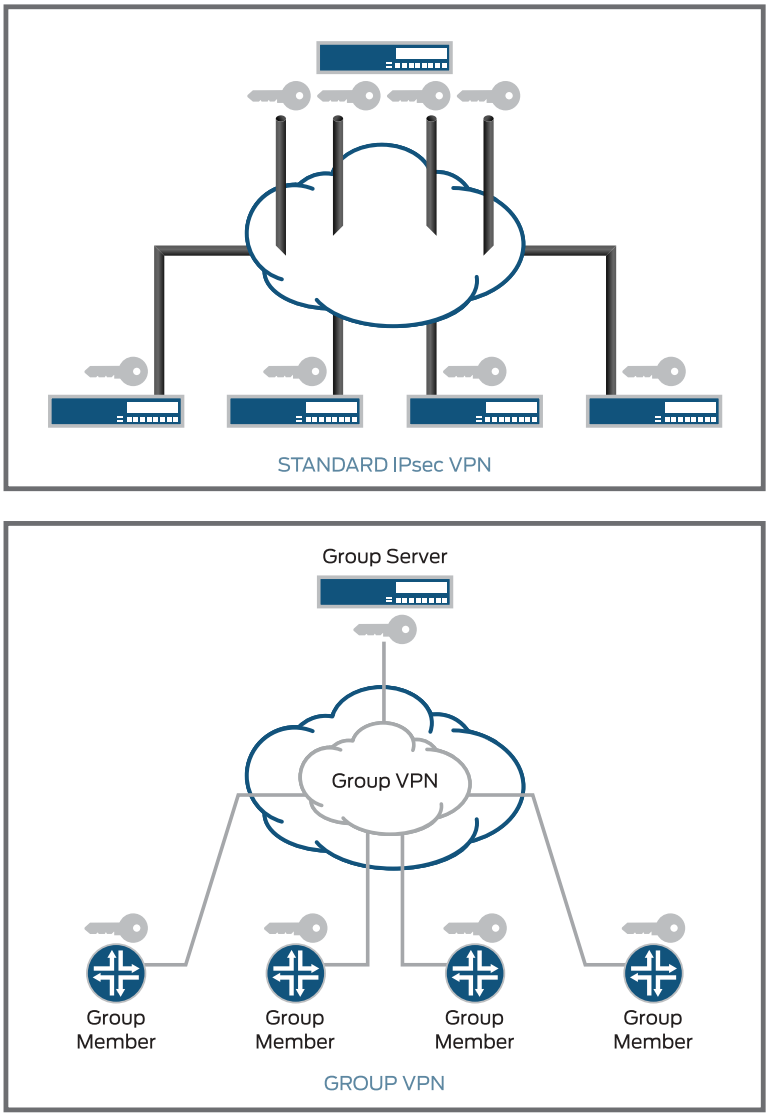
Group VPN and Standard IPsec VPN

A group VPN is built on standards-based technologies that integrate routing and encryption together in the network. An IPsec security SA is a unidirectional agreement between VPN participants that defines the rules to use for authentication and encryption algorithms, key exchange mechanisms, and secure communications.

Traditional IPsec VPN deployments tackle the problem of securing traffic between gateways in the network by creating an overlay network based on the use of point-to-point tunnels. Traffic carried over these tunnels is normally encrypted and authenticated in order to provide data integrity and confidentiality. Secure group members are managed through the Group Domain of Interpretation protocol (GDOI). The GDOI solution takes a different approach by disassociating the encryption and authentication problem from the transport. By doing this, GDOI-based solutions provide a way to encrypt branch-to-branch communications without the need to configure branch-to-branch tunnels.

With current VPN implementations, the SA is a point-to-point tunnel between two routers. A group VPN extends the IPsec architecture to support SAs that are shared by a group of routers (see Figure 1 on page 5). A key server distributes keys and policies to all registered and authenticated member routers. By distributing policies from a centralized point and by sharing the same group security association (the entire group has a single Phase 2 IPsec SA) with authenticated group members, key distribution and management are greatly simplified.

Figure 5: Standard IPsec VPN and Group VPN



Group VPN is a client/server architecture. All members have a unique Phase 1 IKE SA with the key server. Hence, if there are **n** members, there is a total of **n** Phase 1 IKE SAs. However, the entire group shares a single Phase 2 SA.

In traditional IPsec, tunnel endpoint addresses are used as a new packet source and destination. The packet is then routed over the IP infrastructure, using the encrypting gateway source IP address and the decrypting gateway destination IP address. In the case of group VPN, IPsec-protected data packets encapsulate the original source and destination packet addresses of the host in the outer IP header to preserve the IP address. The biggest advantage of tunnel header preservation is the ability to route encrypted packets using the underlying network routing infrastructure.

Thus, with group VPNs, any-to-any connectivity is achieved by preserving the original source and destination IP addresses in the outer header. Secure multicast packets are

replicated in the same way as cleartext multicast packets in the core network. The external IP header in the group VPN is an exact copy of the IP header of the original packet within the ESP header versus the external IP header in a traditional VPN that contains the IP addresses of the VPN gateways.

Table 7: Group VPN vs Traditional Point-to-Point IPsec

Feature	Traditional Point-to-Point IPsec Tunnels	Group VPN
Scalability	IKE/IPsec tunnels between each pair of peers.	Scalable architecture. Single SA and key pair used for entire any-to-any group.
Any-to-any instant connectivity	Can't be done to scale.	Can be done to high-scale.
Overlay routing	Supports overlay routing.	No overlays-native routing.
IP Header Preservation	New IP Header added to original packed results in Limited advanced quality-of-service (QoS).	Keeps original IP header on IPsec packet, and preserves advanced QoS.

Understanding the GDOI Protocol

The Group Domain of Interpretation (GDOI) protocol described in RFC 6407 is used to distribute a set of cryptographic keys and policies to a group of devices. GDOI is defined as the Internet Security Association Key Management Protocol (ISAKMP) Domain of Interpretation (DOI) for group key management. In a group management model, the GDOI protocol operates between a group member and a group controller or key server (GC/KS) and manages group security associations (GSAs) and group-keys for a set of security participants. The ISAKMP defines two phases of negotiation. GDOI is a Phase 2 protocol protected by a Phase 1 ISAKMP security association. IKEv1 is specified in RFC 6407 as a Phase 1 protocol.

GDOI introduces two different encryption keys:

- Key encryption key (KEK)—Used to secure the control plane. KEK is the name of the key used by the group members to decrypt rekey messages from the GC/KS. This key is part of the Security Association Key Encryption Key (SAK).
- Traffic encryption key (TEK)—Used to secure the data plane. TEK is the name of the key used by the group members to encrypt or decrypt communication between other group members. This key is part of the Security Association Transport Encryption Key (SAT).

As with standard IPsec, all keys have a lifetime and have to be rekeyed. The keys distributed through GDOI are group keys and are used by the entire group.

The GSAs and key management are handled through two types of GDOI exchanges:

- **groupkey-pull**—This exchange allows a member to request SAs and keys shared by the group from the server.

In the pull method, the group member requests the group SA and policy from the key server. This request is protected over the IKE SA.

The **groupkey-pull** is the first exchange in the GDOI protocol and is used for group member registration with the GC/KS. The group member specifies the group with which it wants to register, and the GC/KS sends all necessary GSAs and keys to the group member if the member is authorized to join the group. The complete exchange is secured by a Phase 1 SA (IKEv1 SA), which is established with IKEv1 before the **groupkey-pull** exchange begins. The **groupkey-pull** is part of Phase 2 of the GDOI protocol.

- **groupkey-push**—This exchange is a single rekey message that allows the server to send group SAs and keys to members before existing group SAs expire. Rekey messages are unsolicited messages sent from the server to members.

  The **groupkey-push** is the second exchange in the GDOI protocol and is used to send new GSAs and keys from the GC/KS to the group members of a group (rekeying). This exchange is secured by a SA KEK (SAK), which is installed during the **groupkey-pull** exchange. The **groupkey-push** is part of Phase 2 of the GDOI protocol.

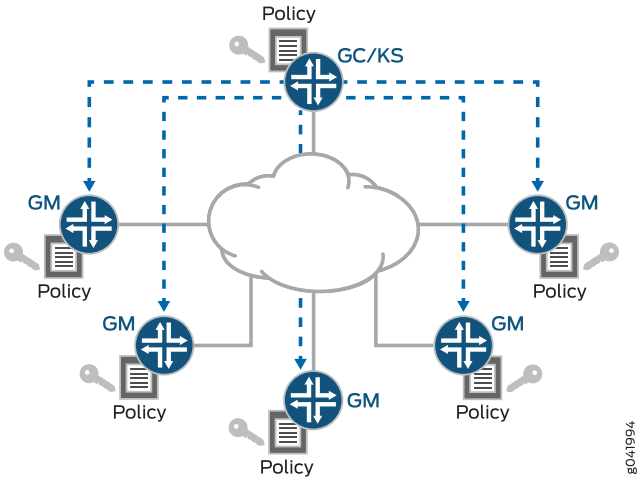
NOTE: Junos OS uses only the groupkey-pull exchange for group member rekeying. The groupkey-push exchange is not supported. For information about group VPN rekeying, see "Rekeying a Group Member" on page 12.

GDOI Protocol and Group VPN

Group VPN is the name of the security technology from Juniper Networks. Group VPN uses the GDOI protocol (RFC 6407) as a base, in addition to other functionalities.

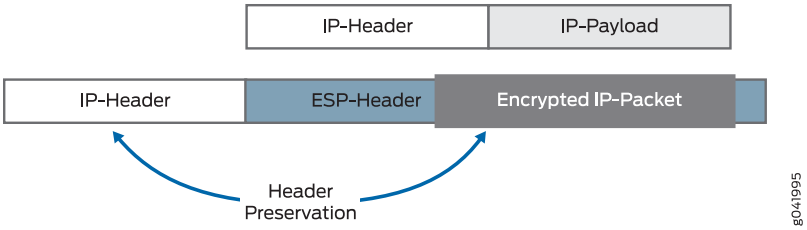
The group VPN technology is based on the GDOI protocol to handle the most important functionality. This protocol is specified in RFC 6407 and defines an ISAKMP Domain of Interpretation (DOI) to manage GSAs and keys for a group of security participants. Thus all members of the group share identical information to encrypt and decrypt traffic among each other. The creation, management, and distribution of GSAs and group keys are centralized and performed by the GC/KS. Figure 2 on page 8 provides a brief overview of the group VPN functionality using GDOI.

Figure 6: Group VPN Using GDOI



The group members use the Encapsulating Security Payload (ESP) protocol in tunnel mode to secure the traffic. However, in group VPN the tunnel mode is modified. Because there is no direct association between the group members, it is not necessary to use special IP addresses in the outer IP header (that is, IP addresses of IPsec gateways). Every group member can decrypt the traffic of every other group member. Thus, the inner IP-Header is copied to the outer IP-Header, and the underlying routing infrastructure and QoS infrastructure can be used. This feature is called Header Preservation and is shown in Figure 3 on page 8.

Figure 7: Header Preservation



To get GSAs and group keys, the group member must register with the GC/KS for a specific group. The result is an IKEv1 SA, which is only needed to secure the registration process. After the registration, the group member has all the information to communicate with the other group members (SAT), as well as the information to successfully decrypt the rekeying messages (SAK). The GC/KS send out rekeying messages before either the SAT or SAK lifetime expires. It is also possible to send a SAT update as well as a SAK update in the same rekey message. The IKEv1 SA is no longer needed and is deleted after the lifetime expires (no IKEv1-rekeying).

Group VPN Traffic

The group VPN traffic includes:

- Control-plane-traffic—Traffic from the group members to the GC/KS in the group VPN deployment with the GDOI protocol only.

- Data-plane-traffic—Traffic between the group members in the group VPN deployment with the ESP protocol only, that is already known from IPsec.

Group Security Association

Unlike traditional IPsec encryption solutions, group VPN uses the concept of group security association (GSA). GSA is similar to an IPsec SA in terms of functionality. GSAs are shared among all group members of a common GDOI group. All members in the group VPN group can communicate with each other using a common encryption policy and a shared GSA. With a common encryption policy and a shared GSA, there is no need to negotiate IPsec between group members. This reduces the resource load on the IPsec routers. Traditional group member scalability (number of tunnels and associated SA) does not apply to group VPN group members.

Group Controller/Key Server

A group controller or key server (GC/KS) is a device used for creating and maintaining the group VPN control plane. It is responsible for creation and distribution of GSAs and group keys. All information the group members need to communicate with other group members is provided by the GC/KS. All encryption policies, such as interesting traffic, encryption protocols, security association, rekey timers, and so on, are centrally defined on the GC/KS and are pushed down to all group members at registration time. Group members authenticate with the GC/KS using IKE Phase 1 and then download the encryption policies and keys required for group VPN operation. The GC/KS is also responsible for refreshing and distributing the keys. Unlike traditional IPsec, interesting traffic defined on the GC/KS is downloaded to every group member, whether or not the group member owns that network.

NOTE: The GC/KS functionality is not supported on MX Series routers. The MX Series routers that are configured as group members can connect with Cisco GC/KS only.

Group Member

A group member is a device used for the traffic encryption process and is responsible for the actual encryption and decryption of data traffic. A group member is configured with IKE Phase 1 parameters and GC/KS information. Encryption policies are defined centrally on the GC/KS and downloaded to the group member at the time of registration. Based on these downloaded policies, the group member determines whether traffic needs to be encrypted or decrypted and what keys to use. The group members use the ESP protocol for securing transport data sent to other group members.

From a functionality point of view, a group member is similar to an IPsec gateway. However, the SAs in normal IPsec are handled per connection between both IPsec gateways and in GDOI, whereas in group VPN the SAs are provided by the GC/KS on a per group basis.

The group member registers with the key server to get the IPsec SA or SAs that are necessary to communicate with the group. The group member provides the group ID to the key server to get the respective policy and keys for this group. The group members

re-register with the server before the current IPsec SAs expire as part of the rekeying process, so that there is no loss of traffic.

NOTE: Only groupkey-pull exchange is used for rekeying. The groupkey-push from the GC/KS is not supported. For information about group VPN rekeying, see "Rekeying a Group Member" on page 12.

Group VPN Implementation Overview

This section explains the Juniper Networks solution for implementing group VPN.

Enabling Group VPN

Service set is used to enable group VPN on a particular interface.

Configuring the Service Set

The group VPN is configured inside a service set using the **ipsec-group-vpn** statement at the **[edit services service-set service-set-name]** hierarchy level.

Sample Service Set Configuration

```
[edit services]
service-set service-set-name {
   interface-service {
      service-interface pic-interface;
   }
}
ipsec-group-vpn vpn-name;
```

NOTE:

- Only one group member can be configured per service set.
- Next-hop style service set is not supported with Group VPN.

Applying the Service Set

A service set is applied at the interface level.

Sample Applying Service Set Configuration

```
[edit interfaces]
interface-name {
   unit 0 {
      family inet {
         service {
            input {
               service-set service-set-name;
            }
            output {
               service-set service-set-name;
            }
         }
         address 10.0.30.2/30;
      }
   }
}
```

Packet Steering

The interface-style service set configuration is used to steer traffic from the Packet Forwarding Engine to the PIC. Packets received on an interface with a service set pointing to the group VPN object are forwarded to the PIC by being injected into the corresponding service interface.

Registering a Group Member

The group member registration to the server starts when the **ipsec-group-vpn** statement is configured for a service set and the service interface is up. When the service interface

goes down, all GSAs associated with this PIC are cleared, and no registration is triggered for these group VPNs until the PIC comes up.

Group member registration involves establishing IKE SA with the GC/KS followed by a **groupkey-pull** exchange to download the SAs and the traffic keys for the specified group identifier.

NOTE: Junos OS does not support traffic-based SA negotiation triggering for group VPNs.

Rekeying a Group Member

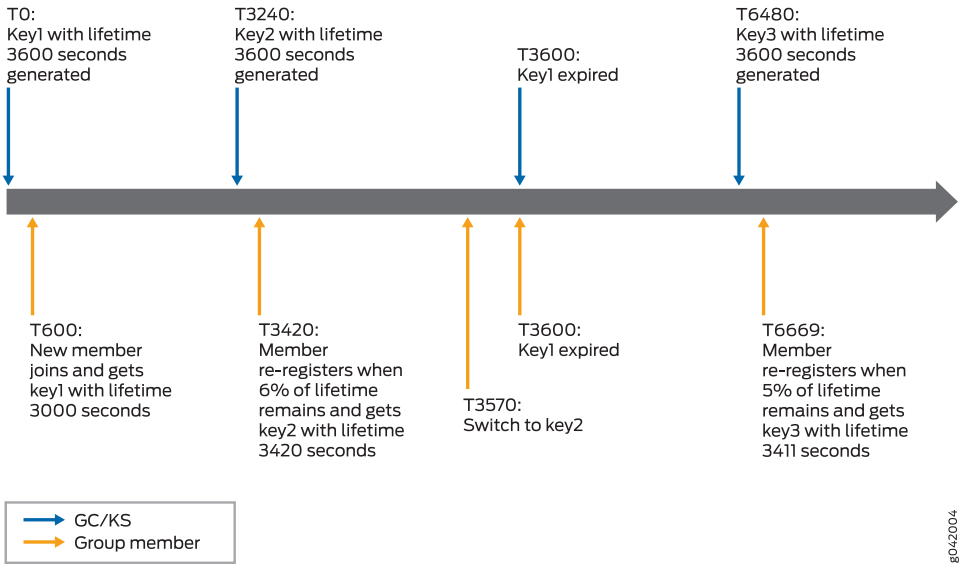
Junos OS group VPN does not support server-to-member rekeying. If rekeying is done from the GC/KS, the key is ignored by the group member.

For group member rekeying, the group members re-register with the GC/KS at the end of the soft lifetime expiry. After rekeying, both the old key and the new key can be used for decryption. However, the new key is not used for encryption until 30 seconds of lifetime of the old key is remaining.

Taking as an example, the GC/KS is configured with a lifetime of 3600 seconds and is connected to one group member without retransmit. Based on the server configuration, the GC/KS generates a new key when 10 percent of the lifetime is remaining. The group member, however, re-registers with the GC/KS when 5 percent to 7 percent of the lifetime is remaining.

Figure 4 on page 12 represents the rekeying process between the GC/KS and the group member.

Figure 8: Group Member Rekeying



Authenticating a Group Member

Junos OS does not provide Public Key Infrastructure (PKI) support for group VPNs, and as a result, pre-shared keys are used for group member authentication.

Fragmenting Group VPN Traffic

Because of the header preservation functionality and the usage of the underlying routing infrastructure, it is necessary to fragment the packets before encryption occurs (if it cannot be prevented).

Hence, pre-fragmentation is supported and is recommended for all deployments.

To avoid post-fragmentation, set the **clear**, **set**, and **copy** options for the DF bit in the group VPN configuration.

Based on this flag setting, the IPsec header has either the **df-bit** set to **clear**, **set**, or **copy** from the inner packet.

NOTE: The DF bit has the clear option set as default.

Sample DF Bit Configuration

```
[edit]
security {
   group-vpn {
      member {
         ipsec {
            vpn vpn-group-name {
               df-bit clear;
            }
         }
      }
   }
}
```

Encrypting Group VPN Traffic

Group members encrypt traffic based on the GSAs and keys provided by the GC/KS. The group VPN encryption path is as follows:

1. Packet received by the Packet Forwarding Engine is checked against a flow match. If a match is found, the packet is further processed and transmitted.
2. If a match is not found, a rule lookup is performed. If a match is found, a flow is created, and the packet is further processed and transmitted.
3. If the rule lookup fails, the packet is dropped.

NOTE: GSA is not triggered during packet processing.

Decrypting Group VPN Traffic

After registration is successful and group VPN SAs are installed, an ESP session is created. Unlike IPsec VPN which creates an ESP session with the source and destination IP to be the gateway address, group VPN creates the ESP session with a zero source and destination IP. Because the ESP session is already created at SA installation, packets are expected to match the existing ESP session.

The group VPN decryption path is as follows:

1. Packet received by the Packet Forwarding Engine undergoes a fragmentation check. If the packet is fragmented, it is assembled for further processing.
2. After packet assembling or if the packet is not fragmented, a zero source and destination IP is used in the 5-tuple decrypt flow lookup. If a match is found, the packet is further processed and transmitted.
3. If the decrypt flow lookup fails, the packet is checked against an SPI flow with zero source and destination IP.
4. If the SPI flow lookup fails, the packet is dropped.
5. If an SPI flow match is found, a decrypt flow is created to avoid the SPI flow lookup for subsequent packets.

Configuring a Routing Instance for Group VPN

Routing instances are supported for both control and data traffic. To enable routing instance support on control plan traffic for a group member to reach the GC/KS in a given VRF routing instance, add the **routing-instance** statement at the **[edit security group-vpn member ike gateway *gateway-name* local-address *address*]** hierarchy.

No additional CLI is required to support a routing instance for data plane packets, as it is determined based on the media interface on which the service set is applied.

Establishing Multiple Groups, Policies, and SAs

Junos OS provides support for one group VPN per service set. However, multiple service sets can be created to support multiple groups in a routing instance. Multiple SAs can be configured per group. However, multiple policies for the same traffic key/SPI is not supported. If the server sends two policies for the same TEK, then they must be paired to be accepted, for instance, A-B and B-A where A and B are IP addresses or subnets. If multiple unpaired policies for a given TEK are received, registration fails and a system log message is generated.

Connecting with Multiple Cooperative GC/KS

For a group member to work with a GC/KS in the cooperative mode, the configuration is extended to allow a maximum of four servers in the server list.

During rekeying, the group member tries to connect to the GC/KS. When the connection to the GC/KS fails, the group member tries to reconnect to the GC/KS. After three retries with an interval of 10 seconds, if the connection to the GC/KS is not restored, the group member tries to establish a connection with the next available server on the server list. This process is repeated until the group member connects to a GC/KS. During this time,

the unexpired GDOI IPsec SAs on the group members are not cleaned up, so group VPN traffic is not affected. The time gap between rekeying and hard lifetime expiry provides sufficient time for the group members to connect to the next available server, in such cases.

Changing Group VPN Configuration

Most group VPN configuration changes result in deleting existing IKE SA and GDOI SA and re-registration, which triggers both phase 1 and SA download with traffic keys.

Bypassing Group VPN Configuration

The service filter needs to be configured on the interface on which the service set is applied if certain traffic like a routing protocol needs to bypass a group VPN. The packet matching service filter will not come to the PIC for service processing and is directly forwarded to the Routing Engine.

Sample Service Set Filter Configuration

```
[edit interfaces]
interface-name {
  unit 0 {
    family inet {
      service {
        input {
          service-set service-set-name service-filter filter-name;
        }
        output {
          service-set service-set-name service-filter filter-name;
        }
      }
    }
  }
}
```

Supported GDOI IPsec Parameters

Every GDOI group has a unique ID. It is used as a common base between GC/KS and the group member to communicate about GSAs and group keys.

During the registration process, the GC/KS sends Security Association Transport Encryption Keys (SATs) to the group members. All parameters regarding the whole group security policy are configured on the GC/KS. The SAT is used by the group members to protect the traffic exchanged among each other. Table 4 on page 16 shows the parameters of the SAT.

Table 8: SAT Parameters

Parameters	Supported Values
Encryption	• DES-CBC • 3DES-CBC • AES-CBC 128 • AES-CBC 192 • AES-CBC 256

Table 8: SAT Parameters *(continued)*

Parameters	Supported Values
Integrity	• HMAC-MD5-96 • HMAC-SHA1-96
Lifetime	Any supported value

Besides the cryptographic algorithms, the traffic, which should be encrypted by the group members, is part of the SAT policy (traffic selector).

The following statements can be used on a Juniper Networks group member. Thus, the addresses have to be specified under the IKE hierarchy level. The enumeration is also prioritized. Thus, in the following example configuration, KS1 is contacted before KS2.

Sample GDOI IPsec Parameters Configuration

```
[edit security]
group-vpn {
   member {
      ike {
         gateway gateway-name {
            ike-policy policy-name;
            server-address <IP_KS1> <IP_KS2> <IP_KS2> <IP_KS4>;
            local-address <IP_GM> routing-instance routing-instance-name;
         }
      }
      ipsec {
         vpn vpn-group-name {
            ike-gateway gateway-name;
            group group-ID;
            match-direction output;
         }
      }
   }
}
```

Supported GDOI IKEv1 Parameters

The group members use only IKEv1 during the registration process in the group VPN environment. Table 5 on page 17 provides an overview of the defined parameters of the IKEv1 SA.

Table 9: IKEv1 SA Parameters of Group Member

Parameter	Supported Values
Encryption	• DES-CBC • 3DES-CBC • AES-CBC 128 • AES-CBC 192 • AES-CBC 256
Authentication	Pre-shared key (minimum 20 signs)

Table 9: IKEv1 SA Parameters of Group Member *(continued)*

Parameter	Supported Values
Integrity	• MD5 • SHA1
Diffie-Hellman Group	• Group 1 • Group 2 • Group 5 • Group 14
Lifetime	Any supported value

The above-mentioned IKEv1 standards are configured as follows:

Sample IKEv1 Configuration

```
[edit security]
group-vpn {
  member {
    ike {
      proposal proposal-name {
        authentication-algorithm sha1;
        authentication-method pre-shared-keys;
        dh-group group5;
        encryption-algorithm aes-128-cbc;
        lifetime-seconds 3600;
      }
      policy policy-name {
        mode main;
        proposals proposal-name;
        pre-shared-key ascii-text "SECRET DATA";
      }
    }
  }
}
```

Applying Dynamic Policies

The **input** and **output** options under the **ipsec-group-vpn** statement specify if the dynamic policies received from the server are used when the interface on which the service set is applied is the incoming or outgoing interface. This provides flexibility to specify different rules in the incoming and outgoing directions.

Supporting TOS and DSCP

Type of service (TOS) and DiffServ Code Points (DSCP) bit are copied from the inner packet to the ESP packet.

Interoperability of Group Members

Cisco's implementation of GDOI is called Group Encryption Transport (GET) VPN. While group VPN in Junos OS and Cisco's GET VPN are both based on RFC 6407, *The Group Domain of Interpretation*, there are some implementation differences that you need to

be aware of when deploying GDOI in a networking environment that includes both Juniper Networks security and routing devices and Cisco routers. For more information, see the current Junos OS release notes.

Group VPN interoperability is as follows:

- Junos OS provides minimal interoperability support with Cisco IOS GC/KS support.
- The MX Series group member is able to communicate with the SRX Series group member and Cisco group member.
- Junos OS does not provide support for group VPN interoperability with the SRX Series group VPN server.

Table 10: Group VPN Interoperability

Group Member (GM)	SRX GM	MX GM	Cisco GM	SRX GS	Cisco GS
MX GM	Yes	Yes	Yes	No	Yes

Following are some of the known issues with Cisco IOS GC/KS:

- The unicast **groupkey-push** message does not work because the proprietary ACK is expected by the Cisco GC/KS following unicast push.

  As a workaround, the **groupkey-pull** is used for rekey.

- The deny policy is used on a Cisco server to add an exception to the group policy.

  As a workaround, this can be done by configuring firewall rules on an MX Series group member. Also, Junos OS group members can work with the deny policy by not failing the negotiation and simply ignoring the contents. This allows system administrators to easily manage networks where both Cisco group members and Junos OS group members co-exist.

Group VPN Limitations

Junos OS group VPN does not provide support for the following:

- GDOI **groupkey-push** exchange. Hence, both unicast and multicast push are not supported.
- Multicast traffic
- Post-fragmentation of packets
- GDOI SNMP MIBs
- Anti-replay
- GAP payload
- Protocol and port in the policies sent by the server. The group member honors only the IP address/subnet specified in the policy.
- Multiple unpaired policies for the same traffic key/SPI

- Overlapping of both local and remote IP across routing instances in an IKE gateway configuration
- Overlapping group VPN policies that can result in mismatched SAs
- IPv6 for control and data traffic
- Co-existence of IPsec and group VPN on the same service set
- Co-existence of services like NAT and ALG on the same service set. NAT and group VPN can co-exist on different service sets. However, they cannot co-exist on the same service set.
- Site To Site (S2S) VPN and Dynamic End Point (DEP) VPN can co-exist with group VPN on different service sets. However, they cannot co-exist on the same service set.
- Multiple groups on same service set
- Group member support with SRX GC/KS
- Logical Key Hierarchy (LKH)
- Graceful restart
- High availability
- Unified ISSU
- PKI support for authentication
- AMS interface and load balancing support
- Multiple groups per service set
- Same gateway for multiple groups, wherein the same local and remote address pair cannot be used for multiple groups.

Example: Configuring Group VPNs on Routing Devices

This example shows how to configure group VPNs to extend the IP Security (IPsec) architecture to support group security associations (GSAs) that are shared by a group of routers.

Requirements

This example uses the following hardware and software components:

- Two MX Series 3D Universal Edge Routers with MS-MIC-16G or MS-MPC-PIC line cards
- Reachability to one or more Cisco Group Controllers or Key Servers (GC/KS)
- Junos OS Release 14.1 or later running on the MX Series routers

Before you begin:

1. Configure the routers for network communication.
2. Configure the Cisco GC/KS.
3. Configure the group member device interfaces.

Overview

Starting with Junos OS Release 14.1, MX Series routers with MS-MIC-16G and MS-MPC-PIC line cards provide the group VPN member functionality support with one or more Cisco group controllers or key servers (GC/KS). The group members can connect to a maximum of four Cisco GC/KSs with minimum interoperability with the cooperative servers.

This feature also provides system logging support for group VPN functionality, and routing instance support for both control and data traffic.

Topology

In Figure 9 on page 40, a group VPN is configured between a Cisco group server, GC/KS – and two group members, GM1 and GM2. The group members are connected to host devices.

In Figure 10 on page 41, a group VPN is configured between GM1 and GM2, and GC/KS1 and GC/KS2 are the primary and secondary group servers, respectively.

Figure 9: Group VPN with Single GC/KS

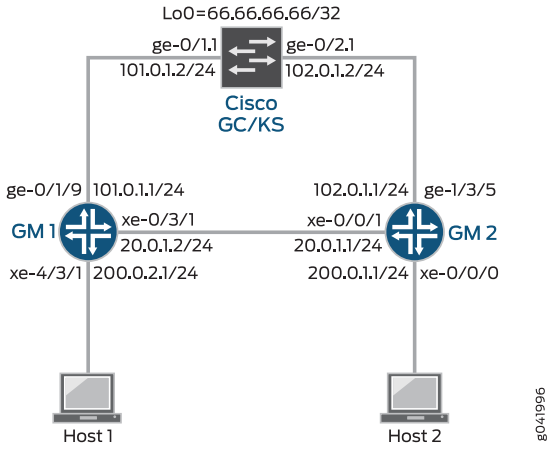
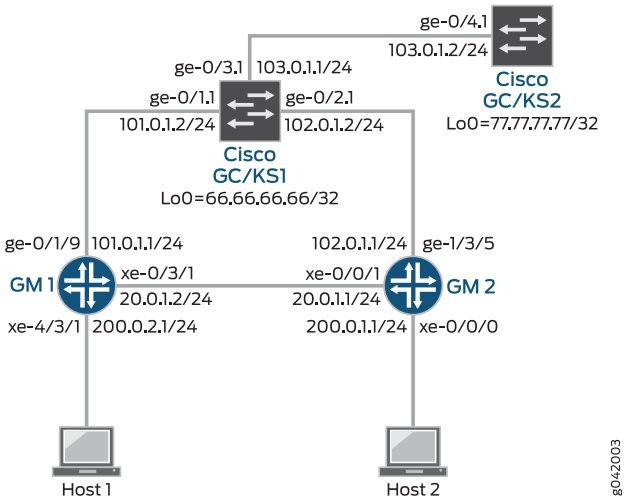


Figure 10: Group VPN with Multiple GC/KS



Configuration

- Configuring Group VPN with a Single GC/KS on page 41
- Configuring Group VPN with Multiple GC/KS on page 45

Configuring Group VPN with a Single GC/KS

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

GM1

```
set interfaces ms-4/0/0 unit 1 family inet
set interfaces ge-0/1/9 vlan-tagging
set interfaces ge-0/1/9 unit 1 vlan-id 11
set interfaces ge-0/1/9 unit 1 family inet address 101.0.1.1/24
set interfaces xe-0/3/1 vlan-tagging
set interfaces xe-0/3/1 unit 1 vlan-id 1
set interfaces xe-0/3/1 unit 1 family inet service input service-set gvpn-service-set
set interfaces xe-0/3/1 unit 1 family inet service output service-set gvpn-service-set
set interfaces xe-0/3/1 unit 1 family inet address 20.0.1.2/24
set interfaces xe-4/3/1 unit 0 family inet address 200.0.2.1/24
set routing-options static route 66.66.66.66/32 next-hop 101.0.1.2
set routing-options static route 200.0.1.0/24 next-hop 20.0.1.1
set security group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
set security group-vpn member ike proposal ike-proposal dh-group group2
set security group-vpn member ike proposal ike-proposal authentication-algorithm sha1
set security group-vpn member ike proposal ike-proposal encryption-algorithm 3des-cbc
set security group-vpn member ike policy ike-policy mode main
set security group-vpn member ike policy ike-policy proposals ike-proposal
set security group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
set security group-vpn member ike gateway gw-group1 ike-policy ike-policy
set security group-vpn member ike gateway gw-group1 server-address 66.66.66.66
set security group-vpn member ike gateway gw-group1 local-address 101.0.1.1
```

```
set security group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
set security group-vpn member ipsec vpn vpn-group1 group 1
set security group-vpn member ipsec vpn vpn-group1 match-direction output
set services service-set gvpn-service-set interface-service service-interface ms-4/0/0.1
set services service-set gvpn-service-set ipsec-group-vpn vpn-group1
```

GM2

```
set interfaces ms-0/2/0 unit 1 family inet
set interfaces xe-0/0/0 unit 0 family inet address 200.0.1.1/24
set interfaces xe-0/1/1 vlan-tagging
set interfaces xe-0/1/1 unit 1 vlan-id 1
set interfaces xe-0/1/1 unit 1 family inet service input service-set gvpn-service-set
set interfaces xe-0/1/1 unit 1 family inet service output service-set gvpn-service-set
set interfaces xe-0/1/1 unit 1 family inet address 20.0.1.1/24
set interfaces ge-1/3/5 vlan-tagging
set interfaces ge-1/3/5 unit 1 vlan-id 11
set interfaces ge-1/3/5 unit 1 family inet address 102.0.1.1/24
set routing-options static route 66.66.66.66/32 next-hop 102.0.1.2
set routing-options static route 200.0.2.0/24 next-hop 20.0.1.2
set security group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
set security group-vpn member ike proposal ike-proposal dh-group group2
set security group-vpn member ike proposal ike-proposal authentication-algorithm sha1
set security group-vpn member ike proposal ike-proposal encryption-algorithm 3des-cbc
set security group-vpn member ike policy ike-policy mode main
set security group-vpn member ike policy ike-policy proposals ike-proposal
set security group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
set security group-vpn member ike gateway gw-group1 ike-policy ike-policy
set security group-vpn member ike gateway gw-group1 server-address 66.66.66.66
set security group-vpn member ike gateway gw-group1 local-address 102.0.1.1
set security group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
set security group-vpn member ipsec vpn vpn-group1 group 1
set security group-vpn member ipsec vpn vpn-group1 match-direction output
set services service-set gvpn-service-set interface-service service-interface ms-0/2/0.1
set services service-set gvpn-service-set ipsec-group-vpn vpn-group1
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*.

To configure GM1:

1. Configure Router GM1 interfaces.

```
[edit interfaces]
user@GM1# set ms-4/0/0 unit 1 family inet

user@GM1# set ge-0/1/9 vlan-tagging
user@GM1# set ge-0/1/9 unit 1 vlan-id 11
user@GM1# set ge-0/1/9 unit 1 family inet address 101.0.1.1/24

user@GM1# set xe-0/3/1 vlan-tagging
user@GM1# set xe-0/3/1 unit 1 vlan-id 1
user@GM1# set xe-0/3/1 unit 1 family inet service input service-set gvpn-service-set
user@GM1# set xe-0/3/1 unit 1 family inet service output service-set gvpn-service-set
```

```
user@GM1# set xe-0/3/1 unit 1 family inet address 20.0.1.2/24

user@GM1# set interfaces xe-4/3/1 unit 0 family inet address 200.0.2.1/24
```

2. Configure static routes to reach the group server and member 2.

```
[edit routing-options]
user@GM1# set static route 66.66.66.66/32 next-hop 101.0.1.2
user@GM1# set static route 200.0.1.0/24 next-hop 20.0.1.1
```

3. Define the IKE proposal.

```
[edit security]
user@GM1# set group-vpn member ike proposal ike-proposal
```

4. Configure the Phase 1 SA for ike-proposal.

```
[edit security]
user@GM1# set group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
user@GM1# set group-vpn member ike proposal ike-proposal dh-group group2
user@GM1# set group-vpn member ike proposal ike-proposal
  authentication-algorithm sha1
user@GM1# set group-vpn member ike proposal ike-proposal encryption-algorithm
  3des-cbc
```

5. Define the IKE policy.

```
[edit security]
user@GM1# set group-vpn member ike policy ike-policy mode main
user@GM1# set group-vpn member ike policy ike-policy proposals ike-proposal
user@GM1# set group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
```

6. Set the remote gateways for gw-group1.

```
[edit security]
user@GM1# set group-vpn member ike gateway gw-group1 ike-policy ike-policy
user@GM1# set group-vpn member ike gateway gw-group1 server-address
  66.66.66.66
user@GM1# set group-vpn member ike gateway gw-group1 local-address 101.0.1.1
```

7. Configure the group identifier and IKE gateway for gw-group1.

```
[edit security]
user@GM1# set group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
user@GM1# set group-vpn member ipsec vpn vpn-group1 group 1
user@GM1# set group-vpn member ipsec vpn vpn-group1 match-direction output
```

8. Configure the service set for gw-group1.

```
[edit services]
user@GM1# set service-set gvpn-service-set interface-service service-interface
  ms-4/0/0.1
user@GM1# set service-set gvpn-service-set ipsec-group-vpn vpn-group1
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show routing-options**, **show security**, and **show services** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

GM1

```
user@GM1# show interfaces
ge-0/1/9 {
  vlan-tagging;
  unit 1 {
    vlan-id 11;
    family inet {
      address 101.0.1.1/24;
    }
  }
}
xe-0/3/1 {
  vlan-tagging;
  unit 1 {
    vlan-id 1;
    family inet {
      service {
        input {
          service-set gvpn-service-set;
        }
        output {
          service-set gvpn-service-set;
        }
      }
      address 20.0.1.2/24;
    }
  }
}
ms-4/0/0 {
  unit 1 {
    family inet;
  }
}
xe-4/3/1 {
  unit 0 {
    family inet {
      address 200.0.2.1/24;
    }
  }
}

user@GM1# show routing-options
static {
  route 66.66.66.66/32 next-hop 101.0.1.2;
  route 200.0.1.0/24 next-hop 20.0.1.1;
}

user@GM1# show security
group-vpn {
  member {
    ike {
      proposal ike-proposal {
        authentication-method pre-shared-keys;
        dh-group group2;
        authentication-algorithm sha1;
        encryption-algorithm 3des-cbc;
      }
```

```
        policy ike-policy {
            mode main;
            proposals ike-proposal;
        }
        gateway gw-group1 {
            ike-policy ike-policy;
            server-address 66.66.66.66;
            local-address 101.0.1.1;
        }
    }
    ipsec {
        vpn vpn-group1 {
            ike-gateway gw-group1;
            group 1;
            match-direction output;
        }
    }
  }
}
```

```
user@GM1# show services
service-set gvpn-service-set {
  interface-service {
    service-interface ms-4/0/0.1;
  }
  ipsec-group-vpn vpn-group1;
}
```

Configuring Group VPN with Multiple GC/KS

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

GM1

```
set interfaces ms-4/0/0 unit 1 family inet
set interfaces ge-0/1/9 vlan-tagging
set interfaces ge-0/1/9 unit 1 vlan-id 11
set interfaces ge-0/1/9 unit 1 family inet address 101.0.1.1/24
set interfaces xe-0/3/1 vlan-tagging
set interfaces xe-0/3/1 unit 1 vlan-id 1
set interfaces xe-0/3/1 unit 1 family inet service input service-set gvpn-service-set
set interfaces xe-0/3/1 unit 1 family inet service output service-set gvpn-service-set
set interfaces xe-0/3/1 unit 1 family inet address 20.0.1.2/24
set interfaces xe-4/3/1 unit 0 family inet address 200.0.2.1/24
set routing-options static route 66.66.66.66/32 next-hop 101.0.1.2
set routing-options static route 200.0.1.0/24 next-hop 20.0.1.1
set security group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
set security group-vpn member ike proposal ike-proposal dh-group group2
set security group-vpn member ike proposal ike-proposal authentication-algorithm sha1
set security group-vpn member ike proposal ike-proposal encryption-algorithm 3des-cbc
set security group-vpn member ike policy ike-policy mode main
set security group-vpn member ike policy ike-policy proposals ike-proposal
set security group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
```

```
set security group-vpn member ike gateway gw-group1 ike-policy ike-policy
set security group-vpn member ike gateway gw-group1 server-address 66.66.66.66
set security group-vpn member ike gateway gw-group1 server-address 77.77.77.77
set security group-vpn member ike gateway gw-group1 local-address 101.0.1.1
set security group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
set security group-vpn member ipsec vpn vpn-group1 group 1
set security group-vpn member ipsec vpn vpn-group1 match-direction output
set services service-set gvpn-service-set interface-service service-interface ms-4/0/0.1
set services service-set gvpn-service-set ipsec-group-vpn vpn-group1
```

GM2

```
set interfaces ms-0/2/0 unit 1 family inet
set interfaces xe-0/0/0 unit 0 family inet address 200.0.1.1/24
set interfaces xe-0/1/1 vlan-tagging
set interfaces xe-0/1/1 unit 1 vlan-id 1
set interfaces xe-0/1/1 unit 1 family inet service input service-set gvpn-service-set
set interfaces xe-0/1/1 unit 1 family inet service output service-set gvpn-service-set
set interfaces xe-0/1/1 unit 1 family inet address 20.0.1.1/24
set interfaces ge-1/3/5 vlan-tagging
set interfaces ge-1/3/5 unit 1 vlan-id 11
set interfaces ge-1/3/5 unit 1 family inet address 102.0.1.1/24
set routing-options static route 66.66.66.66/32 next-hop 102.0.1.2
set routing-options static route 200.0.2.0/24 next-hop 20.0.1.2
set security group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
set security group-vpn member ike proposal ike-proposal dh-group group2
set security group-vpn member ike proposal ike-proposal authentication-algorithm sha1
set security group-vpn member ike proposal ike-proposal encryption-algorithm 3des-cbc
set security group-vpn member ike policy ike-policy mode main
set security group-vpn member ike policy ike-policy proposals ike-proposal
set security group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
set security group-vpn member ike gateway gw-group1 ike-policy ike-policy
set security group-vpn member ike gateway gw-group1 server-address 66.66.66.66
set security group-vpn member ike gateway gw-group1 server-address 77.77.77.77
set security group-vpn member ike gateway gw-group1 local-address 102.0.1.1
set security group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
set security group-vpn member ipsec vpn vpn-group1 group 1
set security group-vpn member ipsec vpn vpn-group1 match-direction output
set services service-set gvpn-service-set interface-service service-interface ms-0/2/0.1
set services service-set gvpn-service-set ipsec-group-vpn vpn-group1
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode*.

To configure GM1:

1. Configure Router GM1 interfaces.

```
[edit interfaces]
user@GM1# set ms-4/0/0 unit 1 family inet

user@GM1# set ge-0/1/9 vlan-tagging
user@GM1# set ge-0/1/9 unit 1 vlan-id 11
user@GM1# set ge-0/1/9 unit 1 family inet address 101.0.1.1/24
```

```
user@GM1# set xe-0/3/1 vlan-tagging
user@GM1# set xe-0/3/1 unit 1 vlan-id 1
user@GM1# set xe-0/3/1 unit 1 family inet service input service-set gvpn-service-set
user@GM1# set xe-0/3/1 unit 1 family inet service output service-set gvpn-service-set
user@GM1# set xe-0/3/1 unit 1 family inet address 20.0.1.2/24


user@GM1# set xe-4/3/1 unit 0 family inet address 200.0.2.1/24
```

2. Configure static routes to reach the group server and member 2.

```
[edit routing-options]
user@GM1# set static route 66.66.66.66/32 next-hop 101.0.1.2
user@GM1# set static route 200.0.1.0/24 next-hop 20.0.1.1
```

3. Define the IKE proposal.

```
[edit security]
user@GM1# set group-vpn member ike proposal ike-proposal
```

4. Configure the Phase 1 SA for ike-proposal.

```
[edit security]
user@GM1# set group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
user@GM1# set group-vpn member ike proposal ike-proposal dh-group group2
user@GM1# set group-vpn member ike proposal ike-proposal
  authentication-algorithm sha1
user@GM1# set group-vpn member ike proposal ike-proposal encryption-algorithm
  3des-cbc
```

5. Define the IKE policy.

```
[edit security]
user@GM1# set group-vpn member ike policy ike-policy mode main
user@GM1# set group-vpn member ike policy ike-policy proposals ike-proposal
user@GM1# set group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
```

6. Set the remote gateways for gw-group1.

```
[edit security]
user@GM1# set group-vpn member ike gateway gw-group1 ike-policy ike-policy
user@GM1# set group-vpn member ike gateway gw-group1 server-address
  66.66.66.66
user@GM1# set group-vpn member ike gateway gw-group1 server-address 77.77.77.77
user@GM1# set group-vpn member ike gateway gw-group1 local-address 101.0.1.1
```

7. Configure the group identifier and IKE gateway for gw-group1.

```
[edit security]
user@GM1# set group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
user@GM1# set group-vpn member ipsec vpn vpn-group1 group 1
user@GM1# set group-vpn member ipsec vpn vpn-group1 match-direction output
```

8. Configure the service set for gw-group1.

```
[edit services]
user@GM1# set service-set gvpn-service-set interface-service service-interface
  ms-4/0/0.1
user@GM1# set service-set gvpn-service-set ipsec-group-vpn vpn-group1
```

Results From configuration mode, confirm your configuration by entering the **show interfaces**, **show routing-options**, **show security**, and **show services** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

GM1

```
user@GM1# show interfaces
ge-0/1/9 {
  vlan-tagging;
  unit 1 {
    vlan-id 11;
    family inet {
      address 101.0.1.1/24;
    }
  }
}
xe-0/3/1 {
  vlan-tagging;
  unit 1 {
    vlan-id 1;
    family inet {
      service {
        input {
          service-set gvpn-service-set;
        }
        output {
          service-set gvpn-service-set;
        }
      }
      address 20.0.1.2/24;
    }
  }
}
ms-4/0/0 {
  unit 1 {
    family inet;
  }
}
xe-4/3/1 {
  unit 0 {
    family inet {
      address 200.0.2.1/24;
    }
  }
}

user@GM1# show routing-options
static {
  route 66.66.66.66/32 next-hop 101.0.1.2;
  route 200.0.1.0/24 next-hop 20.0.1.1;
}

user@GM1# show security
group-vpn {
  member {
    ike {
      proposal ike-proposal {
```

```
            authentication-method pre-shared-keys;
            dh-group group2;
            authentication-algorithm sha1;
            encryption-algorithm 3des-cbc;
          }
          policy ike-policy {
            mode main;
            proposals ike-proposal;
          }
          gateway gw-group1 {
            ike-policy ike-policy;
            server-address [ 66.66.66.66 77.77.77.77 ];
            local-address 101.0.1.1;
          }
        }
        ipsec {
          vpn vpn-group1 {
            ike-gateway gw-group1;
            group 1;
            match-direction output;
          }
        }
      }
    }
```

```
user@GM1# show services
service-set gvpn-service-set {
  interface-service {
    service-interface ms-4/0/0.1;
  }
  ipsec-group-vpn vpn-group1;
}
```

Verification

Confirm that the configuration is working properly.

- Verifying the Group Member IKE SA on page 49
- Verifying the Group Member IPsec SA on page 50
- Verifying the Group Member IPsec Statistics on page 51

Verifying the Group Member IKE SA

Purpose Verify the IKE SAs on Router GM1.

Action From operational mode, run the **show security group-vpn member ike security-associations detail** command.

```
user@GM1> show security group-vpn member ike security-associations detail
IKE peer 66.66.66.66, Index 2994970, Gateway Name: gw-group1
  Role: Initiator, State: UP
  Initiator cookie: 7fad16089a123bcd, Responder cookie: 536b33ffe89799de
  Exchange type: Main, Authentication method: Pre-shared-keys
  Local: 101.0.1.1:848, Remote: 66.66.66.66:848
  Lifetime: Expires in 175 seconds
  Peer ike-id: 66.66.66.66
  Xauth user-name: not available
  Xauth assigned IP: 0.0.0.0
  Algorithms:
   Authentication        : hmac-sha1-96
   Encryption            : 3des-cbc
   Pseudo random function: hmac-sha1
   Diffie-Hellman group  : DH-group-2
  Traffic statistics:
   Input  bytes  :                  752
   Output bytes  :                  716
   Input  packets:                    5
   Output packets:                    5
  Flags: IKE SA is created
  IPSec security associations: 0 created, 0 deleted
  Phase 2 negotiations in progress: 0
```

Meaning Router GM1 has established the IKE SA with the GC/KS for the group.

Verifying the Group Member IPsec SA

Purpose Verify the IPsec SAs on Router GM1.

Action From operational mode, run the **show security group-vpn member ipsec security-associations detail** command.

```
user@GM1> show security group-vpn member ipsec security-associations detail
Virtual-system: root Group VPN Name: vpn-group1
  Local Gateway: 102.0.1.1,  GDOI Server: 66.66.66.66
  Group Id: 1
  Rule Match Direction: output,  Tunnel-MTU: 1500
  Routing Instance: default
  DF-bit: clear
  Stats:
      Pull Succeeded            :   18
      Pull Failed               :   0
      Pull Timeout              :   0
      Pull Aborted              :   0
      Server Failover           :   0
      Delete Received           :   0
      Exceed Maximum Keys(4)    :   0
      Exceed Maximum Policies(1):   0
      Unsupported Algo          :   0
  Flags:
      Rekey Needed:   no

    List of policies received from server:
    Tunnel-id: 10001
      Source IP: ipv4_subnet(any:0,[0..7]=200.0.2.0/24)
      Destination IP: ipv4_subnet(any:0,[0..7]=200.0.1.0/24)

      Direction: bi-directional, SPI: e1c117c7
      Protocol: ESP, Authentication: sha1, Encryption: 3des
      Hard lifetime: Expires in 2526 seconds
      Lifesize Remaining:  Unlimited
      Soft lifetime: Expires in 2366 seconds
      Mode: Tunnel, Type: Group VPN, State: installed
      Anti-replay service: N/A
```

Meaning Router GM1 has established the IPsec SA with the GC/KS.

Verifying the Group Member IPsec Statistics

Purpose Verify the IPsec statistics on Router GM1.

Action From operational mode, run the **show security group-vpn member ipsec statistics** command.

```
user@GM1> show security group-vpn member ipsec statistics
PIC: ms-0/2/0, Service set: gvpn-service-set

ESP Statistics:
  Encrypted bytes:              264
  Decrypted bytes:              264
  Encrypted packets:              3
  Decrypted packets:              3
AH Statistics:
  Input bytes:                    0
  Output bytes:                   0
  Input packets:                  0
  Output packets:                 0
Errors:
  AH authentication failures:     0
  ESP authentication failures:    0
  ESP decryption failures:        0
  Bad headers: 0, Bad trailers: 0
  Replay before window drops: 0, Replayed pkts: 0
  IP integrity errors: 0, Exceeds tunnel MTU: 0
  Rule lookup failures: 0, No SA errors: 0
  Flow errors: 0, Misc errors: 0
```

Meaning **ESP Statistics** shows that packet flows have been encrypted and decrypted between the group members. Router GM1 has encrypted 3 packets and has received 3 decrypted packets from Router GM2.

Troubleshooting

To troubleshoot the group VPN configuration, see:

- Negotiating the IKE SA on page 52
- Establishing the IKE SA on page 53
- Downloading the GDOI IPsec SA on page 53
- Traffic Encryption and Decryption on page 54
- Troubleshooting System Log Messages on page 54

Negotiating the IKE SA

Problem The IKE SA negotiation is not triggered on the group member.

The output of the **show ike** and **show security group-vpn member ike security-associations** commands does not display the IKE negotiations.

Solution To troubleshoot the IKE negotiation issue:

1. Check if the service interface status is up.

   Use **show interfaces terse | match ms** to check if the MS interface is down. An MS interface goes down when the PIC is rebooting.

2. Look for **Ignore gvpn vpn_name since it is inactive** in the log file **/var/log/gkmd**.

   Check if the group VPN is referenced by any service set in the configuration.

   a. Enable **security group-vpn member ike traceoptions**.

   b. Look for the following system log messages in the trace log file:

      - **Dec 2 16:09:54 GVPN:iked_pm_gvpn_trigger called for gvpn200**
      - **Dec 2 16:09:54 GVPN:PM NULL for gvpn gvpn200**
      - **Dec 2 16:09:54 GVPN:Ignore gvpn gvpn200 since it is inactive**

This means either the service set is inactive or the service interface is down.

Establishing the IKE SA

Problem The IKE SA is not getting established with the GC/KS.

In this scenario, the IKE SA state is down in the **show security group-vpn member ike security-associations** command output:

```
user@GM1> show security group-vpn member ike security-associations
Index   State  Initiator cookie  Responder cookie  Mode           Remote Address

       5295626 DOWN   2d47c125d2a9805e  0000000000000000  Main           2.2.2.2
```

Solution To troubleshoot the IKE SA issue:

1. Check if the server address configured under **[edit security group-vpn member ike gateway]** is the correct one and is reachable.
2. Use the **ping** command between the remote devices to check network connectivity.
3. Check if the local address in the **group-vpn** configuration is also a configured address on any of the physical interfaces in the configuration.
4. Check if the IKE proposals match between the group member and the GC/KS.

   If there is a misconfiguration on the IKE SA negotiation, then do the following:

   a. Enable **security group-vpn member ike traceoption**.

   b. Look for the following message in the trace log file:
   Dec 2 15:39:54 ikev2_fb_negotiation_done_isakmp: Entered IKE error code No proposal chosen (14), IKE SA 8dd7000 (neg 8dda800).

5. Look for a **No proposal chosen** error in the log file **/var/log/gkmd**.

Downloading the GDOI IPsec SA

Problem The GDOI IPsec SAs are not downloaded from the GC/KS.

In this scenario, the GDOI **groupkey-pull** with the configured GC/KS fails, and the **show security group-vpn member ipsec sa** command output does not display anything.

Solution To troubleshoot the GDOI IPsec SA issue:

1. Check if the IKE SA has been established with the GC/KS.
2. Check if the group ID configured on the GC/KS and the group member match.
3. Look for any group SA installation failures or other failures in the log file **/var/log/gkmd**.

   Look for the following syslog messages to confirm use of an unsupported GDOI SA algorithm:

   - **Dec 2 15:32:49 simpleman gkmd[1701]: Failed to install SA because of unsupported algo(encr: 3des-cbc, auth : (null)) for SPI 0x6645cdb5 from server 110.1.1.1**
   - **Dec 2 15:32:49 simpleman gkmd[1701]: Member registration failed with key server 110.1.1.1 for group vpn gvpn200, reason SA unusable**

   Look for the following syslog messages to confirm use of unsupported GDOI policies:

   - **Dec 2 15:34:34 simpleman gkmd[1701]: Failed to install SA because of too many(2) policies for SPI 0x6951550c from server 110.1.1.1**
   - **Dec 2 15:34:34 simpleman gkmd[1701]: Member registration failed with key server 110.1.1.1 for group vpn gvpn200, reason SA unusable**

Traffic Encryption and Decryption

Problem The CLI shows IPsec SAs as installed, but traffic does not go through the SAs.

In this scenario, traffic matching the rules received from the server fails to get encrypted or decrypted. The **show security group-vpn member ipsec statistics** command output displays a zero value for encrypt and decrypt packet count.

Solution Look for any counter in the error section of the CLI output for the **statistics** command going up.

Troubleshooting System Log Messages

Problem System log messages are generated to record the different group VPN events.

Solution To interpret the system log messages, refer to the following:

- **Dec 2 15:29:10 simpleman gkmd[1701]: Member registration succeeded with key server 110.1.1.1 for group vpn gvpn200**—GDOI pull was successful.
- **Dec 2 15:21:18 simpleman gkmd[1701]: Member registration failed with key server 110.1.1.1 for group vpn gvpn200, reason Timed out**—GDOI pull failed.
- **Dec 2 15:34:34 simpleman gkmd[1701]: Failed to install SA because of too many(2) policies for SPI 0x6951550c from server 110.1.1.1**—GDOI SA installation failed because of too many policies.
- **Dec 2 15:21:18 simpleman gkmd[1701]: Server 110.1.1.1 is unreachable for group vpn gvpn200**—Single GC/KS failed (Non-COOP)

- **Dec 2 15:51:49 simpleman gkmd[1701]: Current key server 110.1.1.1 is unreachable and will try registering with next Key Server 110.1.1.2 for group vpn gvpn200**—Particular GC/KS is not responding (COOP).
- **Dec 2 15:56:24 simpleman gkmd[1701]: All servers are unreachable for group vpn gvpn200**—None of the GC/KS are responding (COOP).
- **Dec 2 16:01:43 simpleman gkmd[1701]: Member re-registering with Key Server 110.1.1.1 for group-vpn gvpn200**—Member re-registration with the GC/KS.
- **Dec 2 16:01:43 simpleman gkmd[1701]: Creating TEK with SPI 0xb35200ac tunnel_id 10001 for group vpn gvpn200**—GDOI SA TEK creation was successful.
- **Dec 2 16:29:01 simpleman gkmd[1701]: Deleting TEK with SPI 0x6dba2a76 tunnel_id 10001 for group vpn gvpn200 and reason cleared from CLI**—GDOI SA TEK destroy was successful with reason.

  Different down reasons are as follows:

  - Cleared from CLI
  - Hard lifetime expired
  - Too many TEKs
  - Configuration change
  - SA install error
  - Stale SA
  - Interface down

Configuring Group VPNs on Routing Devices

You can configure an MX Series router with MS-MIC-16G and MS-MPC-PIC line cards to provide the group VPN member functionality support with one or more Cisco group controllers or key servers (GC/KSs). The group members can connect to a maximum of four Cisco GC/KSs with minimum interoperability with the cooperative servers.

The group VPN feature also provides system logging support for the group VPN functionality, and routing instance support for both control and data traffic.

Before you begin:

1. Configure the routers for network communication.
2. Configure the Cisco GC/KS.
3. Configure the group member device interfaces.
4. Configure a static route to reach the group server.

To configure a group VPN member, complete the following tasks:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@GM1# edit security
```

2. Define the IKE proposal.

```
[edit security]
user@GM1# set group-vpn member ike proposal proposal-name
```

3. Configure the Phase 1 SA for the IKE proposal.

```
[edit security]
user@GM1# set group-vpn member ike proposal proposal-name authentication-method
  pre-shared-keys
user@GM1# set group-vpn member ike proposal proposal-name dh-group group
user@GM1# set group-vpn member ike proposal proposal-name
  authentication-algorithm sha1
user@GM1# set group-vpn member ike proposal proposal-name encryption-algorithm
  3des-cbc
```

4. Define the IKE policy.

```
[edit security]
user@GM1# set group-vpn member ike policy policy-name mode main
user@GM1# set group-vpn member ike policy policy-name proposals proposal-name
user@GM1# set group-vpn member ike policy policy-name pre-shared-key ascii-text
  text
```

5. Set the remote gateways for the IKE gateway group.

```
[edit security]
user@GM1# set group-vpn member ike gateway gateway-group-name ike-policy
  policy-name
user@GM1# set group-vpn member ike gateway gateway-group-name server-address
  server-IP-address
user@GM1# set group-vpn member ike gateway gateway-group-name local-address
  server-facing-interface-IP-address
```

NOTE: To configure a group member to connect to multiple group servers, add the IP address of all the servers to the remote IKE gateway group configuration.

For example,

```
[edit security]
user@GM1# set group-vpn member ike gateway gw-group1 server-address
  66.66.66.66
user@GM1# set group-vpn member ike gateway gw-group1 server-address
  77.77.77.77
```

6. Configure the group identifier and IKE gateway for the remote gateway group.

```
[edit security]
user@GM1# set group-vpn member ipsec vpn vpn-name ike-gateway
  gateway-group-name
user@GM1# set group-vpn member ipsec vpn vpn-name group group-ID
user@GM1# set group-vpn member ipsec vpn vpn-name match-direction output
```

7. In configuration mode, go to the following hierarchy level:

```
[edit]
user@GM1# edit services
```

8. Configure the service set for the remote gateway group.

```
[edit services]
user@GM1# set service-set service-set-name interface-service service-interface
  service-interface
user@GM1# set service-set service-set-name ipsec-group-vpn vpn-name
```

NOTE: The service set has to be applied on the interface connecting to the other group member.

For example:

```
[edit interfaces]
user@GM1# set xe-0/3/1 unit 1 family inet service input service-set
  gvpn-service-set
user@GM1# set xe-0/3/1 unit 1 family inet service output service-set
  gvpn-service-set
```

9. Verify and commit the configuration.

For example:

```
[edit security]
user@GM1# set group-vpn member ike proposal ike-proposal authentication-method
  pre-shared-keys
user@GM1# set group-vpn member ike proposal ike-proposal dh-group group2
user@GM1# set group-vpn member ike proposal ike-proposal authentication-algorithm
  sha1
user@GM1# set group-vpn member ike proposal ike-proposal encryption-algorithm
  3des-cbc
user@GM1# set group-vpn member ike policy ike-policy mode main
user@GM1# set group-vpn member ike policy ike-policy proposals ike-proposal
user@GM1# set group-vpn member ike policy ike-policy pre-shared-key ascii-text
  ""$9$QEni3/t1RSM87uO87-V4oz36"
user@GM1# set group-vpn member ike gateway gw-group1 ike-policy ike-policy
user@GM1# set group-vpn member ike gateway gw-group1 server-address 66.66.66.66
user@GM1# set group-vpn member ike gateway gw-group1 local-address 101.0.1.1
user@GM1# set group-vpn member ipsec vpn vpn-group1 ike-gateway gw-group1
user@GM1# set group-vpn member ipsec vpn vpn-group1 group 1
user@GM1# set group-vpn member ipsec vpn vpn-group1 match-direction output

[edit services]
user@GM1# set service-set gvpn-service-set interface-service service-interface
  ms-4/0/0.1
user@GM1# set service-set gvpn-service-set ipsec-group-vpn vpn-group1

[edit]
user@GM1# commit
commit complete
```

Related Documentation

- Example: Configuring Group VPN on Routing Devices on page 23

PART 3

Index

Index