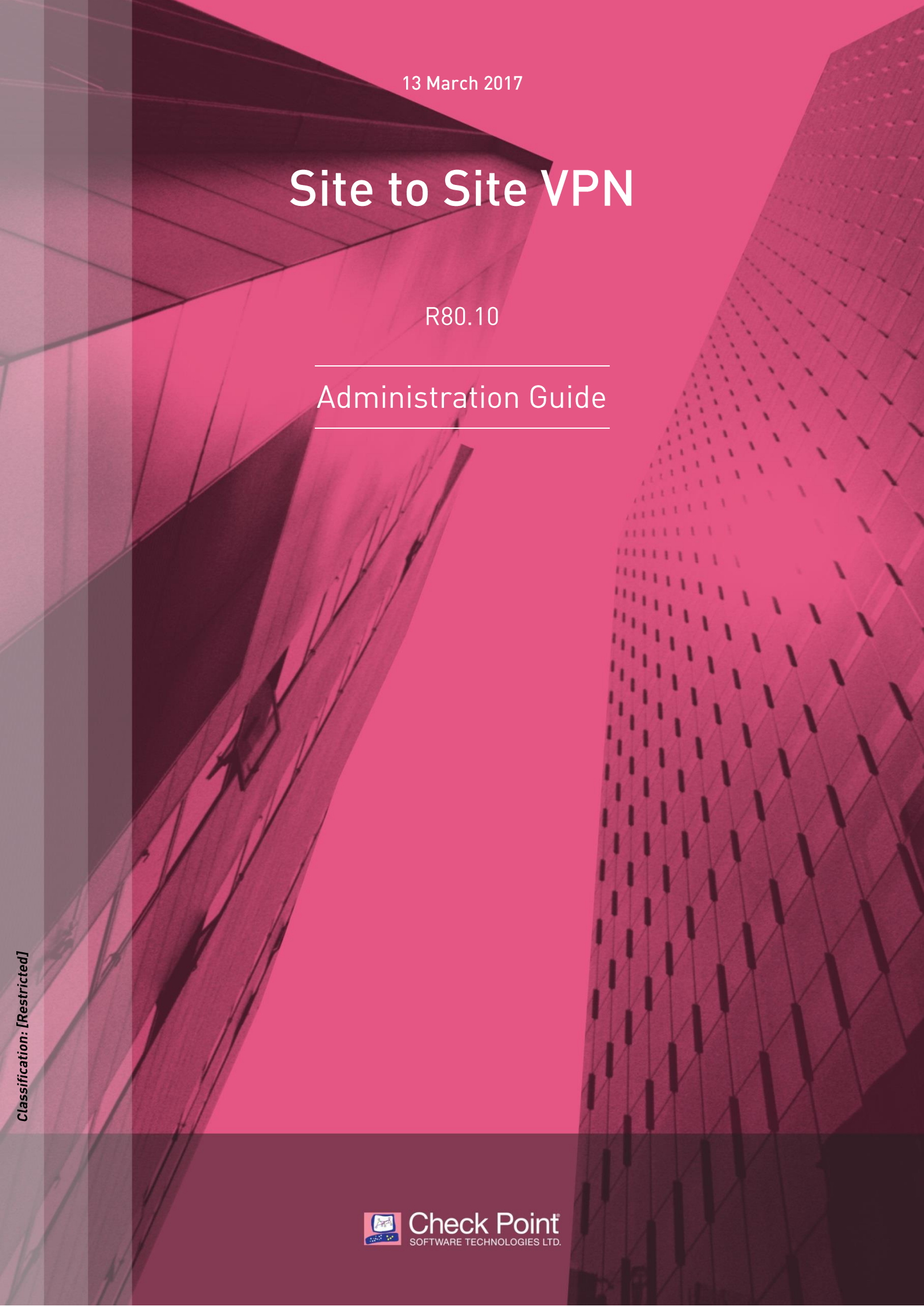
13 March 2017

Site to Site VPN

R80.10

Administration Guide



Check Point
SOFTWARE TECHNOLOGIES LTD.

Important Information

Latest Software

We recommend that you install the most recent software release to stay up-to-date with the latest functional improvements, stability fixes, security enhancements and protection against new and evolving attacks.

Check Point R80.10

For more about this release, see the R80.10 home page http://supportcontent.checkpoint.com/solutions?id=sk111841.

Latest Version of this Document

Download the latest version of this document http://supportcontent.checkpoint.com/documentation_download?ID=53104.

To learn more, visit the Check Point Support Center http://supportcenter.checkpoint.com.

Feedback

Check Point is engaged in a continuous effort to improve its documentation.

Please help us by sending your comments mailto:cp_techpub_feedback@checkpoint.com?subject=Feedback on Site to Site VPN R80.10 Administration Guide.

Searching in Multiple PDFs

To search for text in all the R80.10 PDF documents, download and extract the complete R80.10 documentation package http://supportcontent.checkpoint.com/documentation_download?ID=TBD.
Use **Shift-Control-F** in Adobe Reader or Foxit reader.

Revision History

Date	Description
13 March 2017	First release of this document

Contents

Check Point VPN

In This Section:

IPsec VPN

The IPsec VPN solution lets the Security Gateway encrypt and decrypt traffic to and from other gateways and clients. Use SmartConsole to easily configure VPN connections between Security Gateways and remote devices. You can configure Star and Mesh topologies for VPN networks, and include third-party gateways. The VPN tunnel guarantees:

- Authenticity - Uses standard authentication methods
- Privacy - All VPN data is encrypted
- Integrity - Uses industry-standard integrity assurance methods

IKE and IPsec

The Check Point VPN solution uses these secure VPN protocols to manage encryption keys, and send encrypted packets. IKE (Internet Key Exchange) is a standard key management protocol that is used to create the VPN tunnels. IPsec is protocol that supports secure IP communications that are authenticated and encrypted on private or public networks.

VPN Components

VPN is composed of:

- VPN endpoints, such as Security Gateways, Security Gateway clusters, or remote clients (such as laptop computers or mobile phones) that communicate using a VPN.
- VPN trust entities, such as a Check Point Internal Certificate Authority (ICA). The ICA is part of the Check Point suite used for creating SIC trusted connection between Security Gateways, authenticating administrators and third party servers. The ICA provides certificates for internal Security Gateways and remote access clients which negotiate the VPN link.
- VPN Management tools, such as Security Management Server and SmartConsole. SmartConsole is the SmartConsole used to access the Security Management Server. The VPN Manager is part of SmartConsole. SmartConsole enables organizations to define and deploy Intranet, and remote Access VPNs.

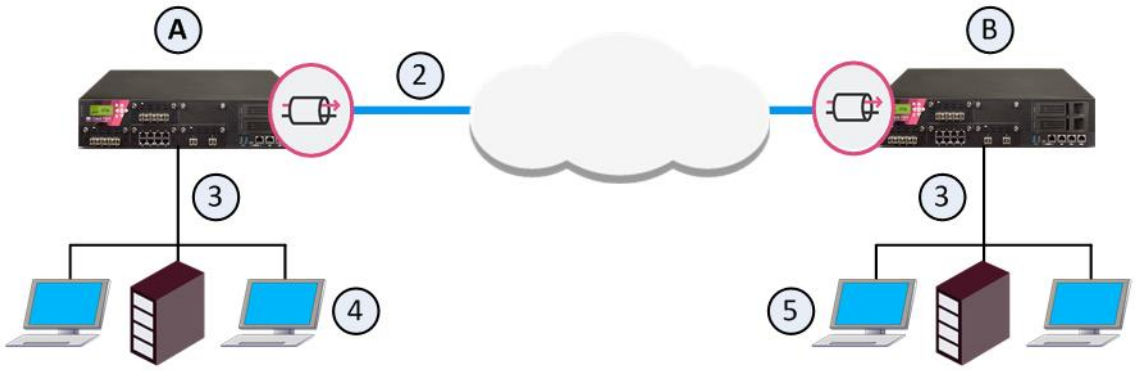
Understanding the Terminology

- **VPN -** Virtual Private Network. A secure, encrypted connection between networks and remote clients on a public infrastructure, to give authenticated remote users and sites secured access to an organization's network and resources.
- **Virtual Tunnel Interface** - Virtual Tunnel Interface. A virtual interface that is a member of an existing, Route Based, VPN tunnel.
- **VPN Peer** - A gateway that connects to a different gateway using a Virtual Tunnel Interface.

- **VPN Domain** - A group of computers and networks connected to a VPN tunnel by one VPN gateway that handles encryption and protects the VPN Domain members.
- **VPN Community** - A named collection of VPN domains, each protected by a VPN gateway.
- **VPN Security Gateway** - The gateway that manages encryption and decryption of traffic between members of a VPN Domain, typically located at one (Remote Access VPN) or both (Site to Site VPN) ends of a VPN tunnel.
- **Site to Site VPN -** An encrypted tunnel between two gateways, typically of different geographical sites.
- **Remote Access VPN** - An encryption tunnel between a Security Gateway and remote access clients, such as Endpoint Security VPN, and communities.
- **Remote Access Community** - A group of computers, appliances, and devices that access, with authentication and encryption, the internal protected network from physically remote sites.
- **Star Topology** - A "hub and spoke" virtual private network community, with gateways defined as Satellites (spokes) that create tunnels only with the central gateway ("hub").
- **Meshed topology** - A VPN community with a VPN Domain that creates a tunnel to other VPN Domains.
- **Domain-based VPN** - A method to route encrypted traffic with parameters defined by Security Gateways.
- **Route-Based VPN** - A routing method for participants in a VPN community, defined by the Virtual Tunnel Interfaces (VTI).
- **IKE (Internet Key Exchange) -** An Encryption key management protocol that enhances IPSec by providing additional features, flexibility, and ease of configuration.
- **IPSec** - A set of secure VPN protocols that manage encryption keys and encrypted packet traffic, to create a standard for authentication and encryption services.

Site to Site VPN

The basis of Site to Site VPN is the encrypted VPN tunnel. Two Security Gateways negotiate a link and create a VPN tunnel and each tunnel can contain more than one VPN connection. One Security Gateway can maintain more than one VPN tunnel at the same time.

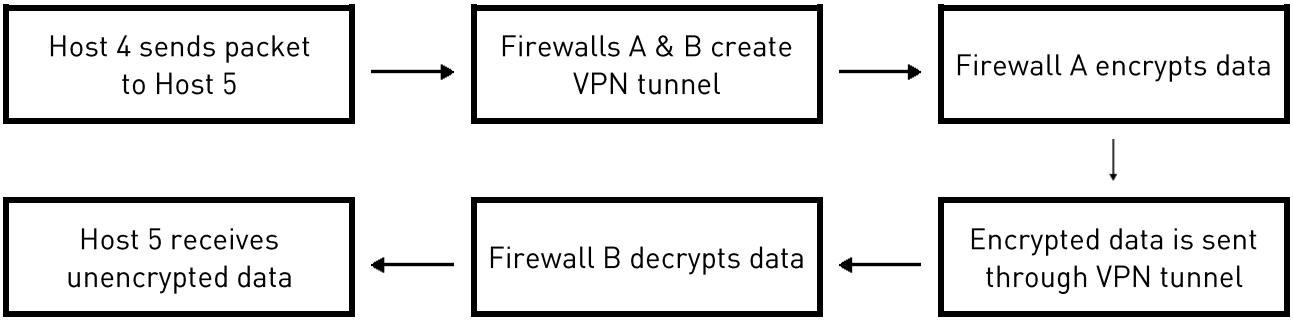


Item	Description
A, B	Security Gateways

Item	Description
2	VPN tunnel
3	Internal network in VPN domain
4	Host 4
5	Host 5

In this sample VPN deployment, Host 4 and Host 5 securely send data to each other. The Security Gateways do IKE negotiation and create a VPN tunnel. They use the IPsec protocol to encrypt and decrypt data that is sent between Host 4 and Host 5.

VPN Workflow

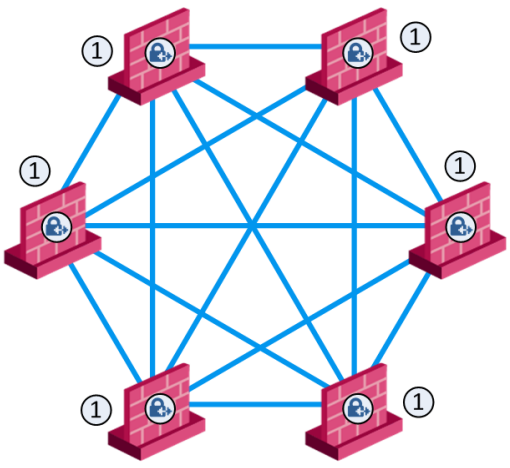


VPN Communities

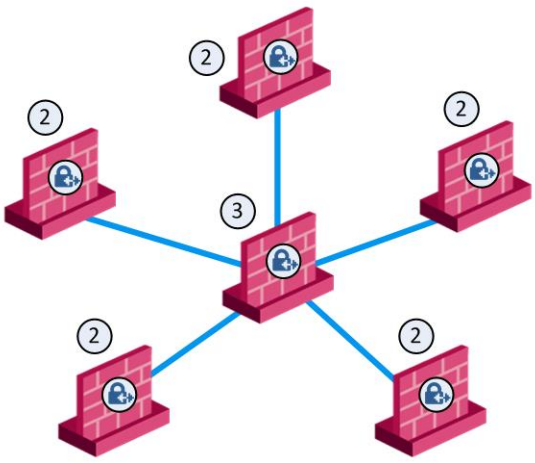
A VPN Domain is a collection of internal networks that use Security Gateways to send and receive VPN traffic. Define the resources that are included in the VPN Domain for each Security Gateway. Then join the Security Gateways into a VPN community - collection of VPN tunnels and their attributes. Network resources of different VPN Domains can securely communicate with each other through VPN tunnels that terminate at the Security Gateways in the VPN communities.

VPN communities are based on Star and Mesh topologies. In a Mesh community, there are VPN tunnels between each pair of Security Gateway. In a Star community, each satellite Security Gateway has a VPN tunnel to the central Security Gateway, but not to other Security Gateways in the community.

Note - Global VPN Communities are not supported in supported in this release.



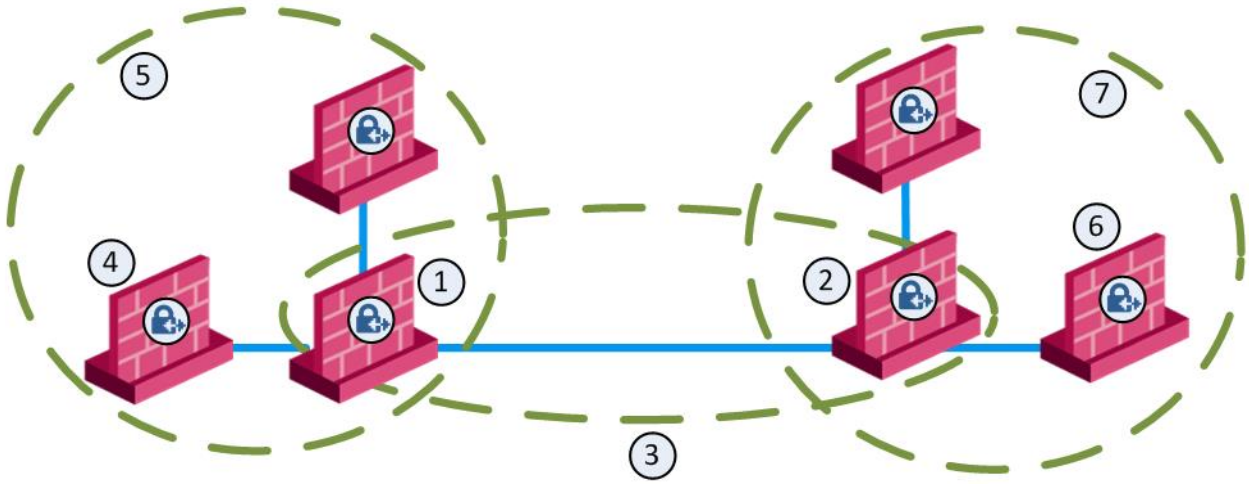
Mesh Topology



Star Topology

Item	Description
1	Security Gateway
2	Satellite Security Gateways
3	Central Security Gateway

Sample Combination VPN Community



Item	Description
1	London Security Gateway
2	New York Security Gateway
3	London - New York Mesh community
4	London company partner (external network)
5	London Star community
6	New York company partner (external network)
7	New York Star community

This deployment is composed of a Mesh community for London and New York Security Gateways that share internal networks. The Security Gateways for external networks of company partners do not have access to the London and New York internal networks. The Star VPN communities let the company partners access the internal networks.

Routing VPN Traffic

Configure the Security Gateway to route VPN traffic based on VPN Domains or based on the routing settings of the operating system.

Note - For each VPN gateway, you must configure an existing gateway as a default gateway.

Domain Based VPN

The VPN traffic is routed according to the VPN Domains that are defined in SmartConsole. Use domain based routing to let satellite Security Gateways in a star-based topology send VPN traffic to each other. The central Security Gateway creates a VPN tunnel to each satellite gateway and the traffic is routed to the correct VPN domain.

Route Based VPN

VPN traffic is routed according to the routing settings (static or dynamic) of the Security Gateway operating system. The Security Gateway uses a VTI (VPN Tunnel Interface) to send the VPN traffic as if it were a physical interface. The VTIs of Security Gateways in a VPN community connect and can support dynamic routing protocols.

Granular Routing Control

The Link Selection feature gives you granular control of the VPN traffic in the network. Use this feature to enable the Security Gateway to:

- Find the best possible route for VPN traffic
- Select the interfaces that are used for VPN traffic to internal and external networks
- Configure the IP addresses that are used for VPN traffic
- Use route probing to select available VPN tunnels
- Use Load Sharing for Link Selection to equally distribute VPN traffic to VPN tunnels

IPv6 Support and Limitations

This release includes limited IPv6 support for IPsec VPN communities:

- IPv6 is supported for Site to Site VPN only (Main IP to Main IP). The Main IP address for both Security Gateways must be defined as an IPv6 Address. You can define other IP addresses that are IPv4 or IPv6.
- IPv6 supports IKEv2 encryption only. IKEv2 is automatically always used for IPv6 traffic. The encryption method configuration applies to IPv4 traffic only.
- VPN tunneling only supports IPv4 inside an IPv4 tunnel, and IPv6 inside an IPv6 tunnel. IPv4 traffic inside an IPv6 tunnel is not supported.

These VPN features are not supported for IPv6:

- VSX
- Remote Access VPN
- CRL fetch for the internal Certificate Authority
- Multiple Entry Points (MEP)
- Route-based VPN (VTI)
- Wire Mode VPN
- Gateways with a dynamic IP address.
- Route Injection Mechanism (RIM)
- Traditional mode Firewall Policies
- IKE Denial of Service protection

- IKE Aggressive Mode
- Gateways with Dynamic IP addresses
- Traditional Mode VPN
- Migration from Traditional mode to Simplified mode
- Tunnel Management (permanent tunnels)
- Directional VPN Enforcement
- Link Selection
- GRE Tunnels
- Tunnel View in SmartView Monitor
- VPN Overview page
- vpn_route.conf configuration file

Getting Started with Site-to-Site VPN

In This Section:

Setting up Site-to-Site VPN between Gateways

Scenario: Two Check Point gateways are managed by the same Security Management Server. How do you create a site-to-site VPN between the two gateways so that they can communicate securely?

Overview of the Workflow:

1. Create the gateway objects in SmartConsole and make sure that IPsec VPN is enabled on each one
2. Generate internal CA certificates for each gateway (done automatically)
3. Create the VPN Community
4. Define the VPN Domain
5. Make sure that the VPN will work with your configured routing, or change the routing as necessary.
6. Create rules for the traffic
7. Install the Access Control Policy

Enabling IPsec VPN on a Gateway

Site to Site VPN requires two or more gateways with the IPsec VPN Software Blade enabled. Other Software Blades can be enabled on the same gateway.

Make sure that Trusted Communication is established between all gateways and the Security Management Server.

To enable the IPsec VPN Software Blade on a gateway:

1. In SmartConsole, open a gateway object.
2. On the **General Properties** page, in the **Network Security** tab, select **IPsec VPN**.
3. Click **OK**.

   An internal CA certificate for the gateway is created automatically.

Creating a VPN Community

You can create a Meshed or Star VPN Community. The procedure below shows an example of a Star Community.

To create a new VPN community:

1. In SmartConsole > **Security Policies** tab, in the **Access Tools** area, click **VPN Communities**.

2. Click the **New** icon and select **Star Community**.
   A **New Star Community** window opens.
3. Enter a name for the VPN Community.
4. In the **Center Gateways** area, click the plus icon to add one or more gateways to be in the center of the community.
5. In the **Satellite Gateways** area, click the plus icon to add one or more gateways to be around the center gateway.
6. Click **OK**.
   The Community uses the default encryption and VPN Routing settings.
7. Optional: Edit more settings for the VPN Community in the community object.

More VPN Community Settings

In addition to the gateway members, you can edit these settings for the VPN Community in the community object:

- **Encrypted Traffic** - Select **Accept all encrypted traffic** to encrypt all traffic between the Security Gateways. If this is not selected, create rules in the Security Policy Rule Base to allow encrypted traffic between community members
- **Encryption** - Select encryption settings that include the **Encryption Method** and **Encryption Suite**. See VPN Community Object - Encryption Settings (on page 37).
- **Tunnel Management** - Select settings VPN tunnels that include **Permanent Tunnels** and Tunnel Sharing. See Configuring Tunnel Features (on page 75).
- **VPN Routing** -For Star Communities, select how VPN traffic is routed between the center and satellite gateways. By default this is always set to **To center only**. See Configuring Domain Based VPN (on page 54).
- **MEP (Multiple Entry Points)** - For Star Communities, select how the entry gateway for VPN traffic is chosen. This only applies when you have multiple center gateways in the community. See Configuring MEP (on page 125).
- **Excluded Services** - Add services that are *not* to be encrypted, for example Firewall control connections. VPN tunnels are not created for the Services included here.
- **Shared Secret** - Configure shared secret authentication to use for communication with external gateways that are part of a VPN community. See Configuring a VPN with External Security Gateways Using Pre-Shared Secret (on page 23).
- **Wire Mode** - Select to define internal interfaces and communities as trusted and bypass the firewall for some communication. See Configuring Wire Mode (on page 109).
- **Advanced** - Configure advanced settings related to IKE, IPsec, and NAT. You can also **Reset All VPN Properties** to revert all VPN Community settings to their default values. See Configuring Advanced IKE Properties (on page 36).

Defining the VPN Domain for a Gateway

The VPN Domain defines the networks and IP addresses that are included in the VPN community. It is also called the Encryption Domain. When you create a Check Point gateway object, the VPN Domain is automatically defined as all IP Addresses behind the gateway, based on the topology information.

You can manually define the VPN domain to include one or more networks. You must have a Network object or Network Group object that represents the domain.

To manually define the VPN Domain:

1. In SmartConsole, open a gateway object.
2. Open the **Network Management** > **VPN Domain** page.
3. Select **Manually defined** and:
   - Browse to the object list and select an object that represents the domain.
   - Browse to the object list and click **New** > **Group** or **Network** to define a new group of machines or network.
4. Click **OK**.

Confirming VPN Routing

By default, IPsec VPN uses the main **IPv4 Address**, defined in the **General Properties** page of the Gateway, for the VPN tunnel connection.

If you want to use this IP address for the VPN communication, and it is an external interface, you do not need additional routing.

If the main IP address is an internal interface, or if you want VPN communication on a different interface, make sure that:

- The **Link Selection** settings for the gateway are configured correctly ("Configuring IP Selection by Remote Peer" on page 77).
- VPN Routing is configured to allow the connections. See Network Management in the *Gaia Administration Guide* http://supportcontent.checkpoint.com/documentation_download?ID=TBD for how to configure routing in Gaia.

Configuring Site to Site VPN Rules in the Access Policy

You must configure rules to allow traffic to and from VPN Communities. Configure rules in SmartConsole > **Security Policies** > **Access Control**. All layers of the Access Control Policy can contain VPN rules.

To make a rule apply to a VPN Community, the **VPN** column of the Rule Base must contain one of these:

- **Any** - The rules applies to all VPN Communities. If you configure a new VPN Community after the rule was created, the rule also applies to the new VPN Community.
- **One or more specified VPN communities -** For example, **MyIntranet.** Right-click in the VPN column of a rule and select **Specific VPN Communities**. The rule applies to the communities shown in the VPN column.

Examples:

- This rule allows encrypted traffic between domains of member gateways of "community_X."

Name	Source	Destination	VPN	Services & Applications
Allow traffic within community	*** Any**	***Any**	**MyCommunity**	*** Any**

- This rule allows traffic from all VPN Communities to the internal network on all services.

Name	Source	Destination	VPN	Services & Applications
Allow all VPN	*** Any**	**Internal_Network**	*** Any**	*** Any**

- This rule allows traffic between two VPN domains with all services.

Name	Source	Destination	VPN	Services & Applications
Site2site VPN	**Local_VPN_Domain** **Peer_VPN_Domain**	**Local_VPN_Domain** **Peer_VPN_Domain**	**Site2Site**	*** Any**

Confirming that a VPN Tunnel Opens Successfully

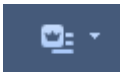
To make sure that a VPN tunnel has successfully opened:

1. Edit the VPN rule and Select **Log** as the **Track** option.
2. Open **Logs & Monitor** and click a new tab.
3. From the bottom of the window, click **Tunnel and User Monitoring.**
   Check Point SmartView Monitor opens.
4. Click the gateway to see IPsec VPN traffic and tunnels opened. A successful connection shows encrypt, decrypt and key install logs.
   Alternatively, search for VPN in SmartConsole to see the relevant logs:
5. Open SmartView Monitor and see that VPN tunnels are up.

SmartConsole Toolbars

For a guided tour of SmartConsole, click **What's New** in the left bottom corner of SmartConsole.

Global Toolbar (top left of SmartConsole)

	Description and Keyboard Shortcut
	The main SmartConsole Menu
	The **Objects** menu. Also leads to the Object Explorer **Ctrl+E**

	Description and Keyboard Shortcut
	Install policy on managed gateways **Ctrl+Shift+Enter**

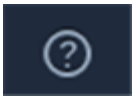
Navigation Toolbar (left side of SmartConsole)

	Description and Keyboard Shortcut
	Gateway configuration view **Ctrl+1**
	Security Policies Access Control view Security Policies Threat Prevention view **Ctrl+2**
	Logs & Monitor view **Ctrl+3**
	Manage & Settings view - review and configure the Security Management Server settings **Ctrl+4**

Command Line Interface Button (left bottom corner of SmartConsole)

	Description and Keyboard Shortcut
	Open a command line interface for management scripting and API **F9**

What's New Button (left bottom corner of SmartConsole)

	Description and Keyboard Shortcut
	Open a tour of the SmartConsole

Objects and Validations Tabs (right side of SmartConsole)

	Description
Objects	Manage security and network objects
Validations	Validation warnings and errors

System Information Area (bottom of SmartConsole)

	Description
Task List	Management activities, such as policy installation tasks

	Description
Server Details	The IP address of the Security Management Server
Connected Users	The administrators that are connected to the Security Management Server

Basic Site to Site VPN Configuration

In This Section:

Configuring a Meshed Community Between Internally Managed Gateways

To configure an internally managed VPN meshed community:

1. Install and configure the Security Gateways as described in the *R80.10 Installation & Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD.
2. In SmartConsole, double click on the Security Gateway object.
3. In the **General Properties** page:
   a) Enter the gateway **Name**.
   b) Enter the **IPv4 Address** and **IPv6 Address**.
   c) In the **Network Security** tab, Select **IPsec VPN**.
   d) Click **Communication** and establish trusted communication with the Gateway.
4. In the **Network Management** page, click **Get Interfaces**.
   a) After the interfaces show in the table, click **Edit** to open the **Interface** window.
   b) In the **Interface** window, define the general properties of the interface and the topology of the network behind it.
5. In the **Network Management** > **VPN Domain** page, define the VPN domain one of:
   - **All IP Addresses behind the Gateway based on Topology information**
   - **Manually defined** as an address range, a network, or a group that can be a combination of address ranges, networks, and even other groups.

     (There are instances where the VPN domain is a group which contains only the Security Gateway itself, for example where the Security Gateway is acting as a backup to a primary Security Gateway in an MEP environment.)

   The network Security Gateway objects are now configured, and need to be added to a VPN community.

   Note - There is nothing to configure on the **IPsec VPN** page, regarding certificates, because internally managed Security Gateways automatically receive a certificate from the internal CA.
6. Open the **Object Explorer** (Ctrl+E), and select **VPN Communities**.
   a) Click **New** > **VPN Communities** > **Meshed Community**.

The **New Meshed Community** window opens.

b) In the **Encrypted Traffic** page, select **Accept all encrypted traffic** if you need all traffic between the Security Gateways to be encrypted. If not, then create appropriate rules in the Security Policy Rule Base that allows encrypted traffic between community members (step 7).

c) On the **Gateways** page, add the Security Gateways created in step 1.

A VPN tunnel is now configured.

For information on other options, such as **Encryption**, **Shared Secret**, and **Advanced**, see: IPsec & IKE (on page 27)

7. If you did not select **Accept all encrypted traffic** in the **Encrypted Traffic** page of the Community, build an access control policy, for example:

Source	Destination	VPN	Service	Action
Any	Any	Meshed community	Any	Accept

Where "Meshed community" is the VPN community you have just defined.

Configuring a Star Community Between Internally Managed Gateways

A star VPN community is configured in much the same way as a meshed community, the difference being the options on the **Star Community** window:

- On the **VPN Routing** page - Select **To center only**.
- On the Gateways page:
  - **Central Gateways** - Add the central Security Gateways.
  - **Central Gateways** - Select **Mesh central Security Gateways** if you want the central Security Gateways to communicate.
  - **Satellite Gateways** - Add the satellite Security Gateways.

Configuring a VPN with External Security Gateways Using Certificates

Configuring a VPN with external Security Gateways (those managed by a different Security Management Server) is more involved than configuring a VPN with internal Security Gateways (managed by the same Security Management Server). This is because:

- Configuration is performed separately in two distinct systems.
- All details must be agreed and coordinated between the administrators. Details such as the IP address or the VPN domain topology cannot be detected automatically but have to be supplied manually by the administrator of the peer VPN Security Gateways.
- The gateways are likely to be using different Certificate Authorities (CAs). Even if the peer VPN Security Gateways use the Internal CA (ICA), it is still a different CA.

There are various scenarios when dealing with externally managed Security Gateways. The following description tries to address typical cases and assumes that the peers work with certificates. If this is not the case refer to Configuring a VPN with External Security Gateways

Using a Pre-Shared Secret (see "Configuring a VPN with External Security Gateways Using Pre-Shared Secret" on page 23).

Note - Configuring a VPN using PKI and certificates is more secure than using pre-shared secrets.

To configure VPN using certificates, with the external Security Gateways as satellites in a star VPN Community:

1. Obtain the certificate of the CA that issued the certificate for the peer VPN Security Gateways, from the peer administrator. If the peer Security Gateway is using the ICA, you can obtain the CA certificate using a web browser from:

   http://<IP address of peer Security Gateway or Management Server>:18264
2. In SmartConsole, define the CA object for the CA that issued the certificate for the peer (see "Enrolling with a Certificate Authority" on page 44).
3. Define the CA that will issue certificates for your side if the Certificate issued by ICA is not appropriate for the required VPN tunnel.

   You may have to export the CA certificate and supply it to the peer administrator.
4. Define the Network Object(s) of the Security Gateway(s) that are internally managed. In particular, be sure to do the following:
   - In the **General Properties** page of the Security Gateway object, select **IPsec VPN**.
   - In the **Network Management** page, define the **Topology**.
   - In the **VPN Domain** page, define the VPN Domain. If it does not contain all the IP addresses behind the Security Gateway, define the VPN domain manually by defining a group or network of machines and setting them as the VPN Domain.
5. If the ICA certificate is not appropriate for this VPN tunnel, then in the **IPsec VPN** page, generate a certificate from the relevant CA (see "Enrolling with a Certificate Authority" on page 44).
6. Define the Network Object(s) of the externally managed Security Gateway(s).
   - If it is not a Check Point Security Gateway, define an Interoperable Device: In Object Explorer, click **New > Network Object > More > Interoperable Device**.
   - If it is a Check Point Security Gateway, define an Externally Managed VPN Gateway: In Object Explorer, click **New > Network Object > Gateways and Servers > More > Externally Managed VPN Gateway**.
7. Set the various attributes of the peer Security Gateway. In particular, be sure to do the following:
   - For an Externally Managed Check Point Security Gateway: In the **General Properties** page of the Security Gateway object, select **IPsec VPN**.
   - Define the **Topology**.
   - Define the **VPN Domain** using the VPN Domain information obtained from the peer administrator. If the VPN Domain does not contain all the IP addresses behind the Security Gateway, define the VPN domain manually by defining a group or network of machines and setting them as the VPN Domain.
   - For an Externally Managed Check Point Security Gateway: In the **IPsec VPN** page, define the **Matching Criteria**. Specify that the peer must present a certificate signed by its own CA. If feasible, enforce details that appear in the certificate as well.
8. Define the Community.

These details assume that a Star Community is used, but you can also use a Meshed Community. If you are working with a Meshed community, ignore the difference between the Central Security Gateways and the Satellite Security Gateways.

- Agree with the peer administrator about the various IKE properties and set them in the **Encryption** page and the **Advanced** page of the community object.
- Define the Central Security Gateways. These are usually the internally managed ones. If no other Community is defined for them, decide whether or not to mesh the central Security Gateways. If they are already in a Community, do not mesh the central Security Gateways.
- Define the Satellite Security Gateways. These are usually the external ones.

9. Click **OK** and publish the changes.
10. Define the relevant access rules in the Security Policy.
11. Add the Community in the **VPN** column, the services in the **Service & Applications** column, the desired **Action**, and the appropriate **Track** option.
12. Install the Access Control Policy.

Configuring a VPN with External Security Gateways Using Pre-Shared Secret

Configuring VPN with external Security Gateways (those managed by a different Security Management Server is more involved than configuring VPN with internal Security Gateways (managed by the same Security Management Server) because:

- Configuration is done separately in two distinct systems.
- All details must be agreed and coordinated between the administrators. Details such as the IP address or the VPN domain topology cannot be detected automatically but have to be supplied manually by the administrator of the peer VPN Security Gateways.

There are various scenarios when dealing with externally managed Security Gateways. The following description tries to address typical cases but assumes that the peers work with pre-shared secrets. If this is not the case refer to Configuring a VPN with External Security Gateways Using PKI (see "Configuring a VPN with External Security Gateways Using Certificates" on page 21).

Note - Configuring a VPN using PKI and certificates is considered more secure than using pre-shared secrets.

To configure a VPN using pre-shared secrets, with the external Security Gateways as satellites in a star VPN Community:

1. Define the Network Object(s) of the Security Gateways that are internally managed. In particular, be sure to:
   - In the **General Properties** page of the Security Gateway object, in the Network Security tab, select **IPsec VPN**.
   - In the **Network Management** page, define the **Topology**.
   - In the **Network Management** > **VPN Domain** page, define the VPN Domain. If it does not contain all the IP addresses behind the Security Gateway, define the VPN domain manually by defining a group or network of machines and setting them as the VPN Domain.
2. Define the Network Object(s) of the externally managed Security Gateway(s).
   - If it is not a Check Point Security Gateway, define an Interoperable Device: In Object Explorer click **New > Network Object > More > Interoperable Device**.
   - If it is a Check Point Security Gateway, define an Externally Managed VPN Gateway: In Object Explorer, click **New > Network Object > Gateways and Servers > More > Externally Managed VPN Gateway**.
3. Set the various attributes of the peer Security Gateway. In particular, make sure to configure:
   - In the **Topology** page, define the **Topology** and the **VPN Domain** using the VPN Domain information obtained from the peer administrator.
   - If the VPN Domain does not contain all the IP addresses behind the Security Gateway, define the VPN domain manually by defining a group or network of machines and setting them as the VPN Domain.
4. Define the Community.

   The following details assume that a Star Community was chosen, but a Meshed Community is an option as well. If you are working with a Mesh community, ignore the difference between the Central Security Gateways and the Satellite Security Gateways.
   - Agree with the peer administrator about the various IKE properties and set them in the **Encryption** page and the **Advanced** page of the community object.
   - Define the Central Security Gateways. These will usually be the internally managed ones. If there is no another Community defined for them, decide whether or not to mesh the central Security Gateways. If they are already in a Community, do not mesh the central Security Gateways.
   - Define the Satellite Security Gateways. These will usually be the external ones.
5. Publish the changes.
6. Agree on a pre-shared secret with the administrator of the external Community members. Then, in the **Shared Secret** page of the community, select **Use only Shared Secret for all external members**. For each external peer, enter the pre-shared secret.
7. Define the relevant access rules in the Access Control Policy. Add the Community in the **VPN** column, the services in the **Services & Applications** column, the desired **Action**, and the appropriate **Track** option.
8. Install the Security Policy.

Firewall Control Connections in VPN Communities

Check Point Nodes communicate with other Check Point Nodes by means of control connections. For example, a control connection is used when the Security Policy is installed from the Security

Management Server to a Security Gateway. Also, logs are sent from Security Gateways to the Security Management Server across control connections. Control connections use Secure Internal Communication (SIC).

Control connections are allowed using Implied Rules in the Access Control Rule Base. Implied Rules are added to or removed from the Access Control Rule Base, by selecting or clearing options in the **Firewall** page of the SmartConsole **Global Properties**.

Some administrators prefer not to rely on implied rules, and instead prefer to define explicit rules in the Access Control Rule Base.

Why Turning off Firewall Implied Rules Blocks Control Connections

If you turn off implicit rules, you may not be able to install a Policy on a remote Security Gateway. Even if you define explicit rules in place of the implied rules, you may still not be able to install the policy:

The administrator wishes to configure a VPN between Security Gateways A and B by configuring SmartConsole. To do this, the administrator must install a Policy from the Security Management Server to the Security Gateways.

1. The Security Management Server successfully installs the Policy on Security Gateway A. As far as gateway A is concerned, Security Gateways A and B now belong to the same VPN Community. However, B does not yet have this Policy.
2. The Security Management Server tries to open a connection to Security Gateway B in order to install the Policy.
3. Security Gateway A allows the connection because of the explicit rules allowing the control connections, and starts IKE negotiation with Security Gateway B to build a VPN tunnel for the control connection.
4. Security Gateway B does not know how to negotiate with A because it does not yet have the Policy. Therefore Policy installation on Security Gateway B fails.

The solution for this is to make sure that control connections do not have to pass through a VPN tunnel.

Allowing Firewall Control Connections Inside a VPN

If you turn off implied rules, you must make sure that control connections are not changed by the Security Gateways. To do this, add the services that are used for control connections to the **Excluded Services** page of the Community object.

Note - Although control connections between the Security Management Server and the Security Gateway are not encrypted by the community, they are nevertheless encrypted and authenticated using Secure Internal Communication (SIC).

Discovering Which Services are Used for Control Connections

1. In the main menu, select **View > Implied Rules**.
2. In the Global Properties **Firewall** page, verify that control connections are accepted.
3. Examine the Access Control Rule Base to see what Implied Rules are visible. Note the services used in the Implied Rules.

Simplified and Traditional Modes

By default, VPN configuration works with *Simplified* mode. *Simplified* mode uses VPN Communities for Site to Site VPN configuration, as described throughout this guide.

Traditional mode is a different, legacy way to configure Site to Site VPN where one of the actions available in the Security Policy Rule Base is **Encrypt**. When encrypt is selected, all traffic between the Security Gateways is encrypted. For details about Traditional Mode, see the *R77 VPN Administration Guide*.

In a policy package, all layers must use the same VPN mode.

Moving from Traditional Mode to Simplified Mode

To switch from Traditional mode to Simplified mode:

1. On the **Global Properties** > **VPN** page, select either **Simplified mode to all new Firewall Policies**, or **Traditional or Simplified per new Firewall Policy**.
2. Click **OK**.
3. From the SmartConsole **Menu**, select **Manage policies**.

   The **Manage Policies** window opens.
4. Click **New.**

   The **New Policy** window opens.
5. Give a name to the new policy and select **Access Control**.

In the Security Policy Rule Base, a new column marked **VPN** shows and the **Encrypt** option is *no longer available* in the **Action** column. You are now working in Simplified Mode.

IPsec & IKE

In This Section:

Overview

In symmetric cryptographic systems, both communicating parties use the same key for encryption and decryption. The material used to build these keys must be exchanged in a secure fashion. Information can be securely exchanged only if the key belongs exclusively to the communicating parties.

The goal of the *Internet Key Exchange* (IKE) is for both sides to independently produce the same symmetrical key. This key then encrypts and decrypts the regular IP packets used in the bulk transfer of data between VPN peers. IKE builds the VPN tunnel by authenticating both sides and reaching an agreement on methods of encryption and integrity. The outcome of an IKE negotiation is a *Security Association* (SA).

This agreement upon keys and methods of encryption must also be performed securely. For this reason, IKE is composed of two phases. The first phase lays the foundations for the second. Both IKEv1 and IKEv2 are supported in Security Gateways of version R71 and higher.

Diffie-Hellman (DH) is that part of the IKE protocol used for exchanging the material from which the symmetrical keys are built. The Diffie-Hellman algorithm builds an encryption key known as a "shared secret" from the private key of one party and the public key of the other. Since the IPsec symmetrical keys are derived from this DH key shared between the peers, at no point are symmetric keys actually exchanged.

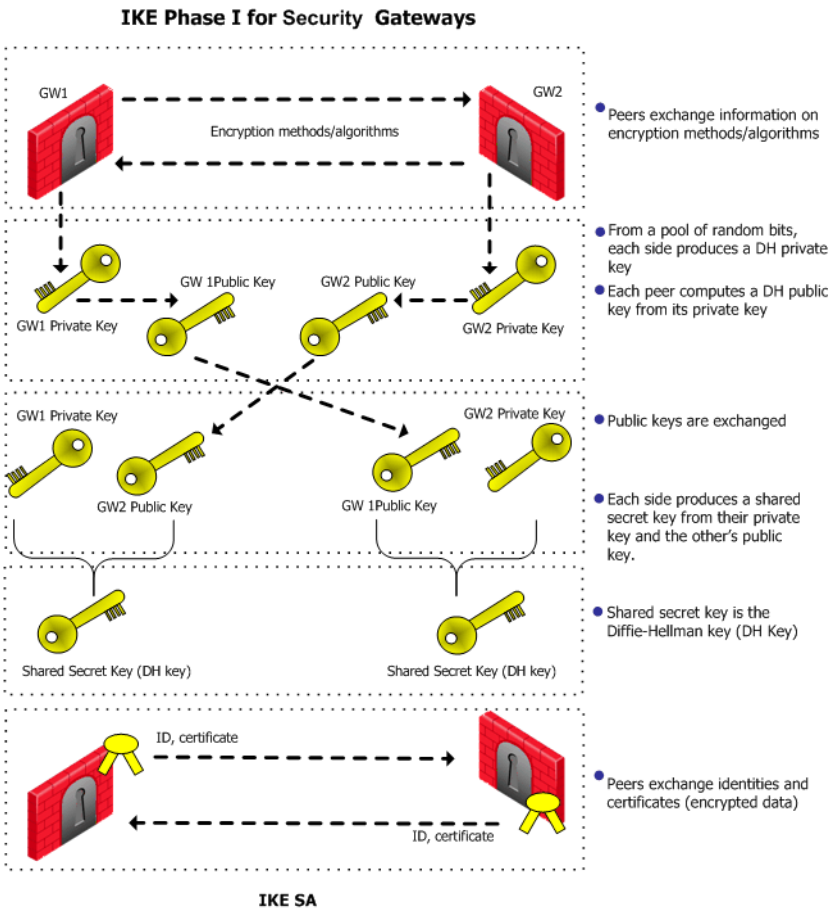
IKE Phase I

During IKE Phase I:

- The peers authenticate, either by certificates or via a pre-shared secret. (More authentication methods are available when one of the peers is a remote access client.)
- A Diffie-Hellman key is created. The nature of the Diffie-Hellman protocol means that both sides can independently create the shared secret, a key which is known only to the peers.
- Key material (random bits and other mathematical data) as well as an agreement on methods for IKE phase II are exchanged between the peers.

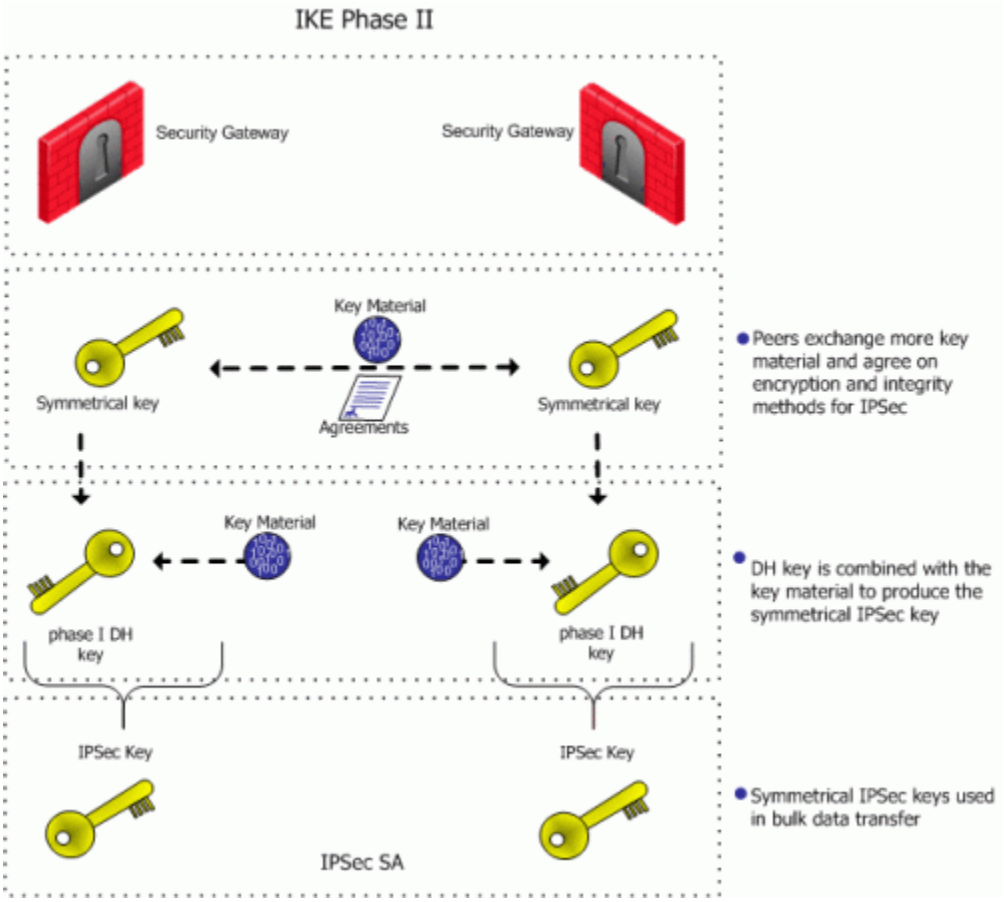
In terms of performance, the generation of the Diffie-Hellman Key is slow and heavy. The outcome of this phase is the IKE SA, an agreement on keys and methods for IKE phase II. Figure below illustrates the process that takes place during IKE phase I.

Note - The exact negotiation stages differ between IKEv1 and IKEv2.

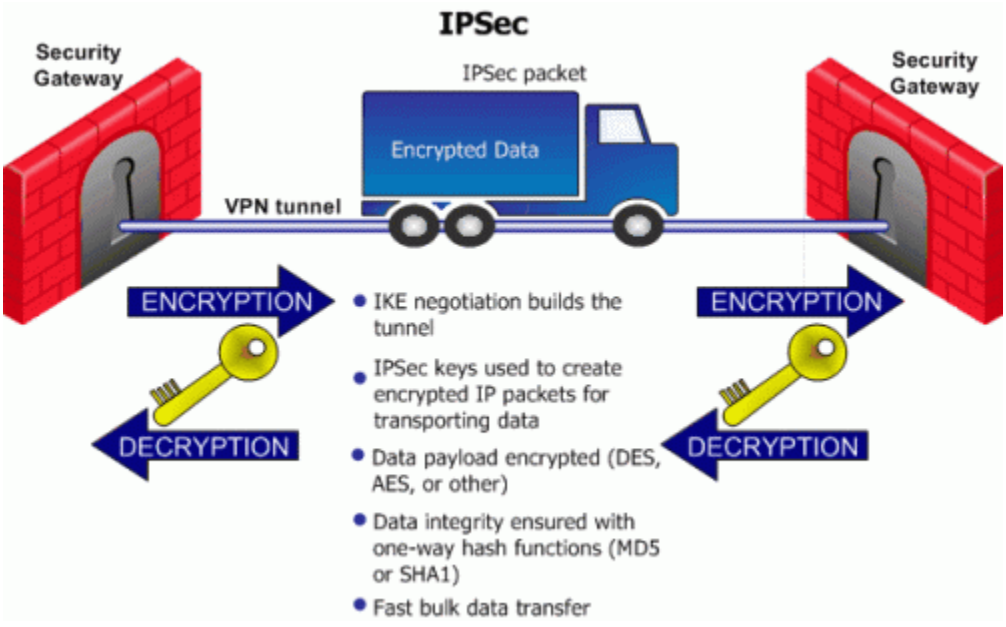

IKE Phase I for Security Gateways
GW1
GW2
Encryption methods/algorithms
Peers exchange information on encryption methods/algorithms
GW1 Private Key
GW 1Public Key
GW2 Public Key
GW2 Private Key
From a pool of random bits, each side produces a DH private key
Each peer computes a DH public key from its private key
GW1 Private Key
GW2 Private Key
GW2 Public Key
GW 1Public Key
Public keys are exchanged
Each side produces a shared secret key from their private key and the other's public key.
Shared Secret Key (DH key)
Shared Secret Key (DH key)
Shared secret key is the Diffie-Hellman key (DH Key)
ID, certificate
ID, certificate
Peers exchange identities and certificates (encrypted data)
IKE SA

IKE Phase II (Quick mode or IPSec Phase)

IKE phase II is encrypted according to the keys and methods agreed upon in IKE phase I. The key material exchanged during IKE phase II is used for building the IPsec keys. The outcome of phase II is the IPsec Security Association. The IPsec SA is an agreement on keys and methods for IPsec, thus IPsec takes place according to the keys and methods agreed upon in IKE phase II.



After the IPsec keys are created, bulk data transfer takes place:



IKEv1 and IKEv2

IKEv2 is supported inside VPN communities working in Simplified mode.

IKEv2 is configured in the **VPN Community Properties window > Encryption**. The default setting is **IKEv1 only**. IKEv2 is automatically always used for IPv6 traffic. The encryption method configuration applies to IPv4 traffic only.

For Remote users, the IKE settings are configured in **Global Properties > Remote Access > VPN Authentication and Encryption**. IKEv2 is not supported for Remote Access.

Note - IKEv2 is not supported on UTM-1 Edge devices or VSX objects before R75.40VS. If UTM-1 Edge devices or such VSX objects are included in a VPN Community, the Encryption setting should be **Support IKEv1**.

Methods of Encryption and Integrity

Two parameters are decided during the negotiation:

- Encryption algorithm
- Hash algorithm

Parameter	IKE Phase 1 (IKE SA)	IKE PHASE 2 (IPSec SA)
Encryption	• AES-128 • AES-256(default) • 3DES • DES • CAST (IKEv1 only)	• AES-128 (default) • AES-256 • 3DES • DES • DES-40CP (IKEv1 only) • CAST (IKEv1 only) • CAST-40 (IKEv1 only) • NULL • AES-GCM-128 • AES-GCM-256
Integrity	• MD5 • SHA1 (default) • SHA -256 • AES-XCBC • SHA -384	• MD5 • SHA1 (default) • SHA -256 • AES-XCBC • SHA -384

NULL means perform an integrity check only; *packets are not encrypted*.

Diffie Hellman Groups

The Diffie-Hellman key computation (also known as exponential key agreement) is based on the Diffie Hellman (DH) mathematical groups. A Security Gateway supports these DH groups during the two phases of IKE.

Parameter	IKE Phase 1 (IKE SA)	IKE Phase 2 (IPSec SA)
Diffie Hellman Groups	• Group2 (1024 bits) (default) • Group1 (768 bits) • Group5 (1536 bits) • Group14 (2048 bits) • Group19 (256-bit ECP) • Group20 (384-bit ECP)	• Group2 (1024 bits) (default) • Group1 (768 bits) • Group5 (1536 bits) • Group14 (2048 bits) • Group19 (256-bit ECP) • Group20 (384-bit ECP)

A group with more bits ensures a key that is harder to break, but carries a heavy cost in terms of performance, since the computation requires more CPU cycles.

Phase I modes

Between Security Gateways, there are two modes for IKE phase I. These modes only apply to IKEv1:

- Main Mode
- Aggressive Mode

If aggressive mode is *not* selected, the Security Gateway defaults to main mode, performing the IKE negotiation using six packets; aggressive mode performs the IKE negotiation with three packets.

Main mode is preferred because:

- Main mode is partially encrypted, from the point at which the shared DH key is known to both peers.
- Main mode is less susceptible to **Denial of Service** (DoS) attacks. In main mode, the DH computation is performed *after* authentication. In aggressive mode, the DH computation is performed parallel to authentication. A peer that is not yet authenticated can force processor intensive Diffie-Hellman computations on the other peer.

Note - Use aggressive mode when a Check Point Security Gateway needs to negotiate with third party VPN solutions that do not support main mode.

When dealing with remote access, IKE has additional modes:

- *Hybrid mode*. Hybrid mode provides an alternative to IKE phase I, where the Security Gateway is allowed to authenticate using certificates and the client via some other means, such as SecurID. For more information on Hybrid mode, see the *Remote Access VPN Administration Guide*.
- *Office mode*. Office mode is an extension to the IKE protocol. Office Mode is used to resolve routing issues between remote access clients and the VPN domain. During the IKE negotiation, a special mode called *config mode* is inserted between phases I and II. During config mode, the remote access client requests an IP address from the Security Gateway. After the Security Gateway assigns the IP address, the client creates a virtual adapter in the Operating System. The virtual adapter uses the assigned IP address. For more information, see the *Remote Access VPN Administration Guide*.

Renegotiating IKE & IPsec Lifetimes

IKE phase I is more processor intensive than IKE phase II, because the Diffie-Hellman keys have to be produced, and the peers authenticated, each time. For this reason, IKE phase I is performed less frequently. However, the IKE SA is only valid for a certain period, after which the IKE SA must be renegotiated. The IPsec SA is valid for an even shorter period, meaning many IKE phase II's take place.

The period between each renegotiation is known as the **lifetime**. Generally, the shorter the lifetime, the more secure the IPsec tunnel (at the cost of more processor intensive IKE negotiations). With longer lifetimes, future VPN connections can be set up more quickly. By default, IKE phase I occurs once a day; IKE phase II occurs every hour but the time-out for each phase is configurable.

Configure the frequency of IKE and IPsec Security Associations in SmartConsole: **VPN Community Properties** > **Advanced**.

Perfect Forward Secrecy

The keys created by peers during IKE phase II and used for IPsec are based on a sequence of random binary digits exchanged between peers, and on the DH key computed during IKE phase I.

The DH key is computed once, then used a number of times during IKE phase II. Since the keys used during IKE phase II are based on the DH key computed during IKE phase I, there exists a mathematical relationship between them. For this reason, the use of a single DH key may weaken the strength of subsequent keys. If one key is compromised, subsequent keys can be compromised with less effort.

In cryptography, **Perfect Forward Secrecy** (PFS) refers to the condition in which the compromise of a current session key or long-term private key does *not* cause the compromise of earlier or subsequent keys. Security Gateways meet this requirement with a PFS mode. When PFS is enabled, a fresh DH key is generated during IKE phase II, and renewed for each key exchange.

However, because a new DH key is generated during each IKE phase I, no dependency exists between these keys and those produced in subsequent IKE Phase I negotiations. Enable PFS in IKE phase II only in situations where extreme security is required.

The supported DH groups for PFS are: 1, 2, 5, 14, 19, and 20. The default is group 2 (1042 bits).

Configure this in **VPN Community Properties** > **Encryption** > **IKE Security Association (Phase 2)** > **Use Perfect Forward Secrecy**.

Note - PFS mode is supported only between gateways, not between Security Gateways and remote access clients.

IP Compression

IP compression is a process that reduces the size of the data portion of the TCP/IP packet. Such a reduction can cause significant improvement in performance. IPsec supports the *Flate/Deflate* IP compression algorithm. Deflate is a smart algorithm that adapts the way it compresses data to the actual data itself. Whether to use IP compression is decided during IKE phase II. IP compression is not enabled by default.

IP compression is important for Remote Access client users with slow links.

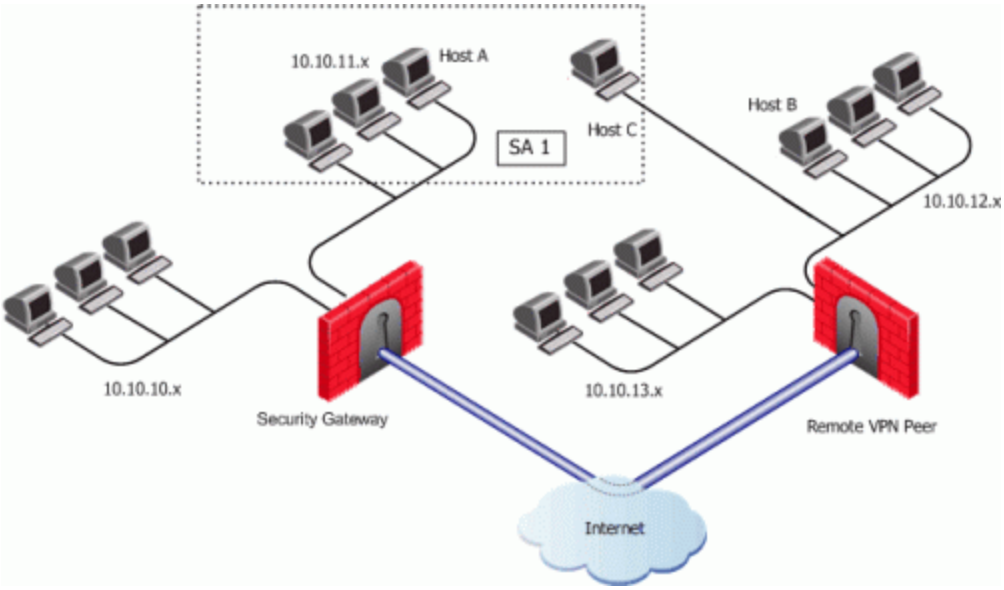
Security Gateway encryption makes TCP/IP packets appear "mixed up". This kind of data cannot be compressed and bandwidth is lost as a result. If IP compression is enabled, packets are compressed *before* encryption. This has the effect of recovering the lost bandwidth.

Subnets and Security Associations

By default, a VPN tunnel is created for the complete subnets that host computers reside on, and not just for the host computers involved in the communication.

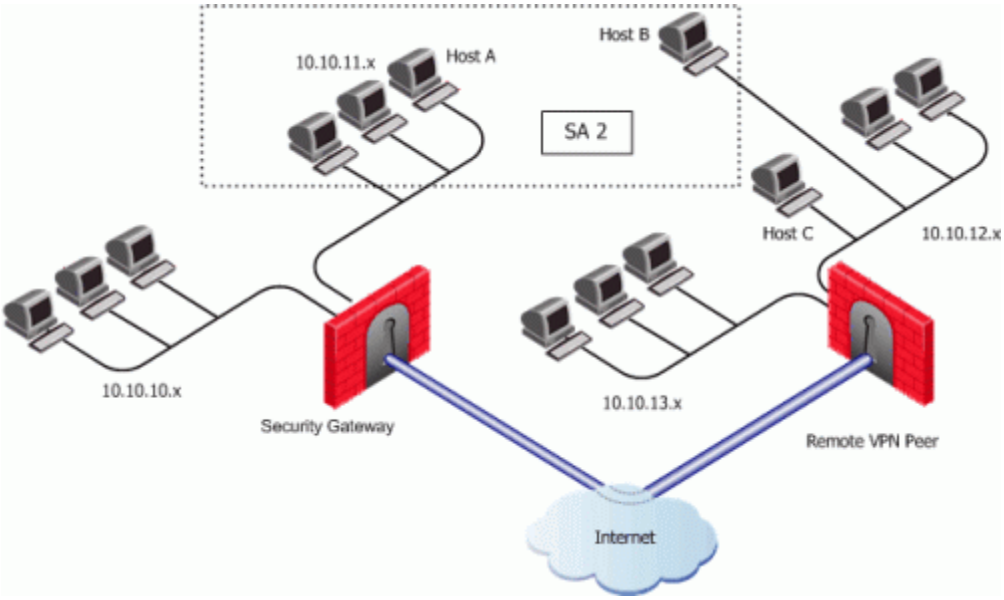
Unique SA Per Pair of Peers

If you disable the **Support Key exchange for subnets** option on each Security Gateway, you can create a *unique* Security Association for a pair of peers.



If the Security Gateway is configured to **Support key exchange for subnets** but the option is unsupported on the remote peer, when host A communicates with host C, a Security Association (SA 1) will be negotiated between host A's subnet and host C's IP address. The same SA is then used between any host on the 10.10.11.x subnet and Host C.

When host A communicates with host B, a separate Security Association (SA 2) is negotiated between host A's subnet and host B. As before, the same SA is then used between any host in 10.10.11.x subnet and Host B.



When **Support Key exchange for subnets** is not enabled on communicating Security Gateways, then a security association is negotiated between individual IP addresses; in effect, a unique SA per host.

IKE DoS Protection

Understanding DoS Attacks

Denial of Service (DoS) attacks are intended to reduce performance, block legitimate users from using a service, or even bring down a service. They are not direct security threats in the sense that no confidential data is exposed, and no user gains unauthorized privileges. However, they consume computer resources such as memory or CPU.

Generally, there are two kinds of DoS attack. One kind consists of sending malformed (garbage) packets in the hope of exploiting a bug and causing the service to fail. In the other kind of DoS attack, an attacker attempts to exploit a vulnerability of the service or protocol by sending well-formed packets. IKE DoS attack protection deals with the second kind of attack.

IKE DoS Attacks

The IKE protocol requires that the receiving Security Gateway allocates memory for the first IKE Phase 1 request packet that it receives. The Security Gateway replies, and receives another packet, which it then processes using the information gathered from the first packet.

An attacker can send many IKE first packets, while forging a different source IP address for each. The receiving Security Gateway is obliged to reply to each, and assign memory for each. This can consume all CPU resources, thereby preventing connections from legitimate users.

The attacker sending IKE packets can pretend to be a machine that is allowed to initiate IKE negotiations, such as a Check Point Security Gateway. This is known as an identified source. The attacker can also pretend to have an IP address that the receiving Security Gateway does not know about, such as a / SecureClient, or a Check Point Security Gateway with a dynamic IP address. This is known as an unidentified source.

Defense Against IKE DoS Attacks

When the number of simultaneous IKE negotiations handled exceeds the accepted threshold, it concludes that it is either under load or experiencing a Denial of Service attack. In such a case, the Security Gateway can filter out peers that are the probable source of a potential Denial of Service attack. The following sections describe different types of defenses against IKE DoS attacks.

IKE DoS protection is not supported for IPv6 addresses.

SmartConsole IKE DoS Attack Protection Settings

To protect against IKE DoS attacks, configure the SmartConsole **IKE Denial of Service Protection** settings, in the **VPN >Advanced** page of the **Global Properties**. IKE DoS protection is not supported for IPv6 addresses.

- **Support IKE DoS protection from identified source** — The default setting for identified sources is **Stateless**. If the Security Gateway is under load, this setting requires the peer to respond to an IKE notification in a way that proves that the IP address of the peer is not spoofed. If the peer cannot prove this, the Security Gateway does not begin the IKE negotiation.

  If the source is identified, protecting using **Puzzles** is over cautious, and may affect performance. A third possible setting is **None**, which means no DoS protection.
- **Support IKE DoS protection from unidentified source** — The default setting for unidentified sources is **Puzzles**. If the Security Gateway is under load, this setting requires the peer to solve

a mathematical puzzle. Solving this puzzle consumes peer CPU resources in a way that makes it difficult to initiate multiple IKE negotiations simultaneously.

For unidentified sources, **Stateless** protection may not be sufficient because an attacker may well control all the IP addresses from which the IKE requests appear to be sent. A third possible setting is **None**, which means no DoS protection.

Advanced IKE DoS Attack Protection Settings

Advanced IKE DoS attack protection can be configured on the Security Management Server with GuiDBedit, the Check Point Database Tool. Configure the protection in the Global Properties shown. IKE DoS protection is not supported for IPv6.

ike_dos_threshold

Values: 0-100. Default: 70. Determines the percentage of maximum concurrent ongoing negotiations, above which the Security Gateway will request DoS protection. If the threshold is set to 0, the Security Gateway will always request DoS protection.

ike_dos_puzzle_level_identified_initiator

Values: 0-32. Default: 19. Determines the level of the puzzles sent to known peer Security Gateways. This attribute also determines the maximum puzzle level a Security Gateway is willing to solve.

ike_dos_puzzle_level_unidentified_initiator

Values: 0-32. Default: 19. Determines the level of the puzzles sent to unknown peers (such as / SecureClient and DAIP Security Gateways). This attribute also determines the maximum puzzle level that DAIP Security Gateways and / SecureClient are willing to solve.

ike_dos_max_puzzle_time_gw

Values: 0-30000. Default: 500. Determines the maximum time in milliseconds a Security Gateway is willing to spend solving a DoS protection puzzle.

ike_dos_max_puzzle_time_daip

Values: 0-30000. Default: 500. Determines the maximum time in milliseconds a DAIP Security Gateway is willing to spend solving a DoS protection puzzle.

ike_dos_max_puzzle_time_sr

Values: 0-30000. Default: 5000. Determines the maximum time in milliseconds a is willing to spend solving a DoS protection puzzle.

ike_dos_supported_protection_sr

Values: None, Stateless, Puzzles. Default: Puzzles. When downloaded to / SecureClient, it controls the level of protection the client is willing to support.

Security Gateways use the **ike_dos_protection_unidentified_initiator** property (equivalent to the SmartConsole Global Property: **Support IKE DoS Protection from unidentified Source**) to decide

what protection to require from remote clients, but / SecureClient clients use the **ike_dos_protection**. This same client property is called **ike_dos_supported_protection_sr** on the Security Gateway.

Protection After Successful Authentication

You can configure fields in GuiDBedit to protect against IKE DoS attacks from peers who may authenticate successfully and then attack a Security Gateway. These settings are configured in the Global Properties table and not per Security Gateway. By default these protections are off. Once you enter a value, they will be activated.

To limit the amount of IKE Security Associations (SA's) that a user can open, configure the following fields:

Type of VPN	Field	Recommended Value
Site to site	number_of_ISAKMP_SAs_kept_per_peer	5
Remote user	number_of_ISAKMP_SAs_kept_per_user	5

To limit the amount of tunnels that a user can open per IKE, configure the following fields:

Type of VPN	Field	Recommended Value
Site to site	number_of_ipsec_SAs_per_IKE_SA	30
Remote user	number_of_ipsec_SAs_per_user_IKE_SA	5

Client Properties

Some Security Gateway properties change name when they are downloaded to Remote Access VPN Clients. The modified name appears in the Userc.C file, as follows:

Property Names

Property Name on Gateway	User.C Property name on Client
ike_dos_protection_unidentified_initiator (Equivalent to the SmartConsole Global Property: **Support IKE DoS Protection from unidentified Source**)	**ike_dos_protection** or **ike_support_dos_protection**
ike_dos_supported_protection_sr	ike_dos_protection
ike_dos_puzzle_level_unidentified_initiator	**ike_dos_acceptable_puzzle_level**
ike_dos_max_puzzle_time_sr	**ike_dos_max_puzzle_time**

Configuring Advanced IKE Properties

IKE is configured in two places:

- On the VPN community network object (for IKE properties).
- On the Security Gateway network object (for subnet key exchange).

VPN Community Object - Encryption Settings

IPv6 automatically works with IKEv2 encryption only. The option that you select here, applies to IPv4 traffic.

To configure a VPN Community object:

1. In SmartConsole, click **Open Object Explorer** (Ctrl+E).
2. From the navigation tree, click **VPN Communities**.
3. Double-click the VPN Community object.
   The Community object window opens and shows the **Gateways** page.
4. From the navigation tree, click **Encryption**.
5. Configure the settings.
6. Click **OK** and publish the changes.

Encryption Method

- **Encryption Method** - For IKE phase I and II.
  - **IKEv2 only** - Only support encryption using IKEv2. Security Gateways in this community cannot access peer gateways that support IKEv1 only.
  - **Prefer IKEv2, support IKEv1** - If a peer supports IKEv2, the Security Gateway will use IKEv2. If not, it will use IKEv1 encryption. This is recommended if you have a community of older and new Check Point Security Gateways.
  - **IKEv1 only** - IKEv2 is not supported.

Encryption Suite

- **Use this encryption suite** - Select the methods negotiated in IKE phase 2 and used in IPSec connections. Select and choose the option for best interoperability with other vendors in your environment.
  - **VPN-A** or **VPN B -** See RFC 4308 for more information.
  - **Suite-B GCM-128** or **256** - See RFC 6379 for more information.
- **Custom encryption suite -** If you require algorithms other than those specified in the other options, select the properties for IKE Phase 1, including which **Diffie-Hellman group** to use. Also, select properties for IKE Phase 2.

  Note - Suite-B GCM-128 and 256 encryption suites are supported on R71.50 gateways and from R75.40 gateways.

If there is a Dynamic IP Gateway inside the community, R77.30 (or lower) community member gateways that respond to its IKE negotiation, use the configuration defined in **Global Properties** > **Remote Access** > **VPN -Authentication and Encryption**.

More

- **Use aggressive mode** (Main mode is the default) - Select only if the peer only supports aggressive mode. This is only supported with IKEv1.
- **Use Perfect Forward Secrecy**, and the **Diffie-Hellman group** - Select if you need extremely high security.
- **Support IP compression** - Select to decrease bandwidth consumption and for interoperability with third party peers configured to use IP Compression.

VPN Community Object - Advanced Settings

Configure these options in the VPN Community object **Advanced** page:

IKE (Phase 1)

When to renegotiate the IKE Security Associations.

IKE (Phase 2)

When to renegotiate the IPsec security associations. This sets the expiration time of the IPsec encryption keys.

NAT

Disable NAT inside the VPN community - Select to not apply NAT for the traffic while it passes through IPsec tunnels in the community.

Reset

Reset all VPN properties to the default.

On the Gateway Network Object

1. On the **IPsec VPN > VPN Advanced** page, select one of the options in the **VPN Tunnel Sharing** section. There are several settings that control the number of VPN tunnels between peer gateways:

   Note - Wire Mode is not supported for IPv6 connections.

   - **Use the community settings** - Create the number of VPN tunnels as defined on the community **Tunnel Management** page.
   - **Custom settings**:
     - **One VPN tunnel per each pair of hosts** - A VPN tunnel is created for every session initiated between every pair of hosts.
     - **One VPN tunnel per subnet pair** - After a VPN tunnel has been opened between two subnets, subsequent sessions between the same subnets will share the same VPN tunnel. This is the default setting and is compliant with the IPsec industry standard.
     - **One VPN tunnel per Gateway pair -** One VPN tunnel is created between peer gateways and shared by all hosts behind each peer gateway.
2. On the **Capacity Optimization** page, you can maximize VPN throughput by limiting the following connection parameters:
   - **Maximum concurrent IKE negotiations**
   - **Maximum concurrent tunnels -** For Remote Access tunnels

   If you have many employees working remotely, you might want to raise the default values.

Public Key Infrastructure

In This Section:

Need for Integration with Different PKI Solutions

X.509-based PKI solutions provide the infrastructure that enables entities to establish trust relationships between each other based on their mutual trust of the Certificate Authority (CA). The trusted CA issues a certificate for an entity, which includes the entity's public key. Peer entities that trust the CA can trust the certificate — because they can verify the CA's signature — and rely on the information in the certificate, the most important of which is the association of the entity with the public key.

IKE standards recommend the use of PKI in VPN environments, where strong authentication is required.

A Security Gateway taking part in VPN tunnel establishment must have an RSA key pair and a certificate issued by a trusted CA. The certificate contains details about the module's identity, its public key, CRL retrieval details, and is signed by the CA.

When two entities try to establish a VPN tunnel, each side supplies its peer with random information signed by its private key and with the certificate that contains the public key. The certificate enables the establishment of a trust relationship between the Security Gateways; each gateway uses the peer Security Gateway public key to verify the source of the signed information and the CA's public key to validate the certificate's authenticity. In other words, the validated certificate is used to authenticate the peer.

Every deployment of Check Point Security Management Server includes an Internal Certificate Authority (ICA) that issues VPN certificates for the VPN modules it manages. These VPN certificates simplify the definition of VPNs between these modules.

Situations can arise when integration with other PKI solutions is required, for example:

- A VPN must be established with a Security Gateway managed by an external Security Management Server. For example, the peer Security Gateway belongs to another organization which utilizes Check Point products, and its certificate is signed by its own Security Management Server ICA.
- A VPN must be established with a non-Check Point VPN entity. In this case, the peer's certificate is signed by a third-party CA.
- An organization may decide, for whatever reason, to use a third party CA to generate certificates for its Security Gateways.

Supporting a Wide Variety of PKI Solutions

Check Point Security Gateways support many different scenarios for integrating PKI in VPN environments.

- **Multiple CA Support for Single VPN Tunnel** – Two Security Gateways present a certificate signed by different ICAs.
- **Support for non-ICA CAs** – In addition to ICA, Security Gateways support the following Certificate Authorities:
- **External ICA** - The ICA of another Security Management Server
- **Other OPSEC certified PKI solutions**
- **CA Hierarchy** – CAs are typically arranged in a hierarchical structure where multiple CAs are subordinate to a root authority CA. A subordinate CA is a Certificate Authority certified by another Certificate Authority. Subordinate CA's can issue certificates to other, more subordinate CAs, forming a certification chain or hierarchy.

PKI and Remote Access Users

The Check Point Suite supports certificates not only for Security Gateways but for users as well. For more information, see Introduction to Remote Access VPN for information about user certificates.

PKI Deployments and VPN

Following are some sample CA deployments:

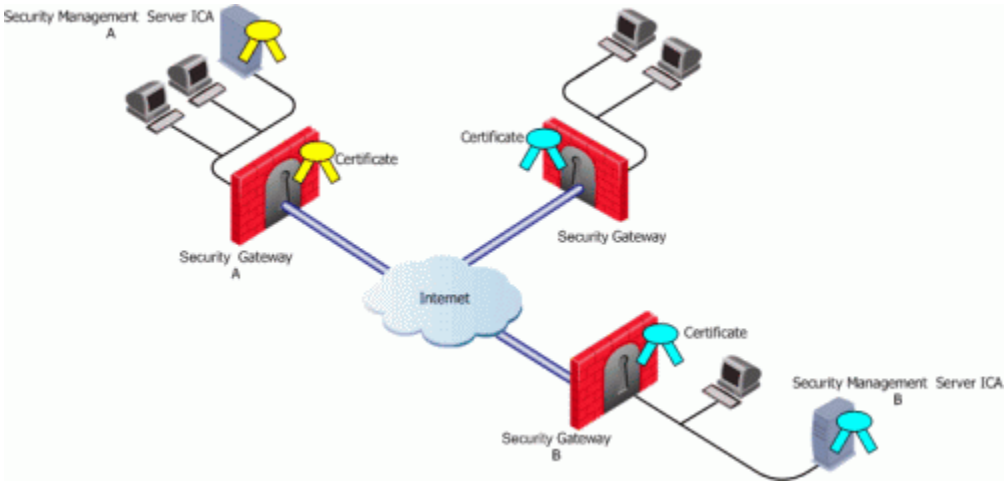
- Simple Deployment - internal CA
- CA of an external Security Management Server
- CA services provided over the Internet
- CA on the LAN

Simple Deployment – Internal CA

When the VPN tunnel is established between Security Gateways managed by the same Security Management Server, each peer has a certificate issued by the Security Management Server's ICA.

CA of An External Security Management Server

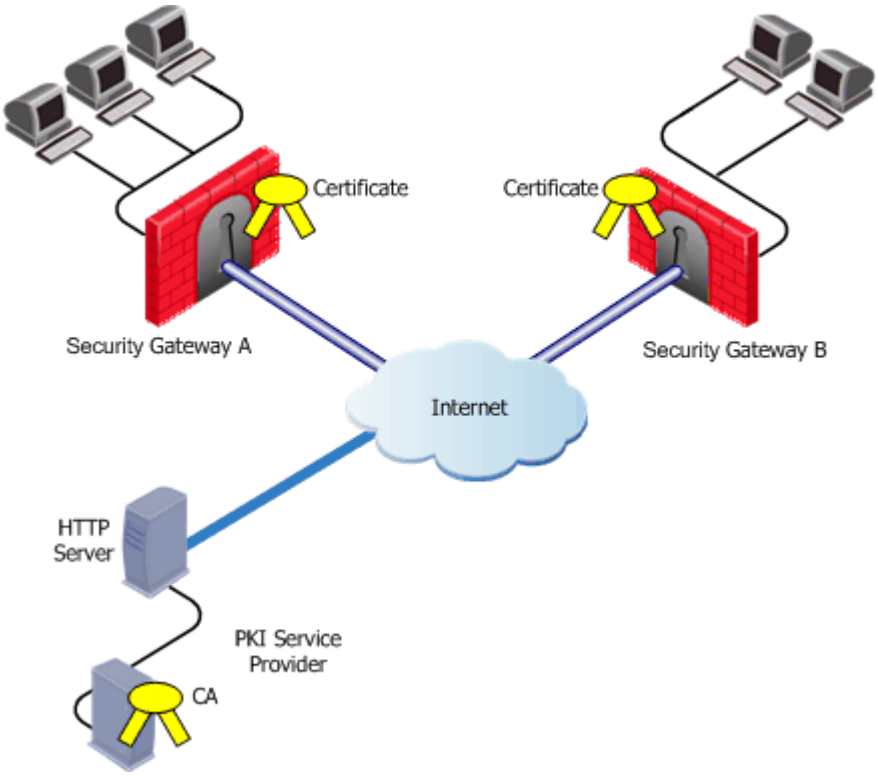
If a Check Point Security Gateway is managed by an external Security Management Server (for example, when establishing a VPN tunnel with another organization's VPN modules), each peer has a certificate signed by its own Security Management Server's ICA.



Security Management Server A issues certificates for Security Management Server B that issues certificates for Security Gateway B.

CA Services Over the Internet

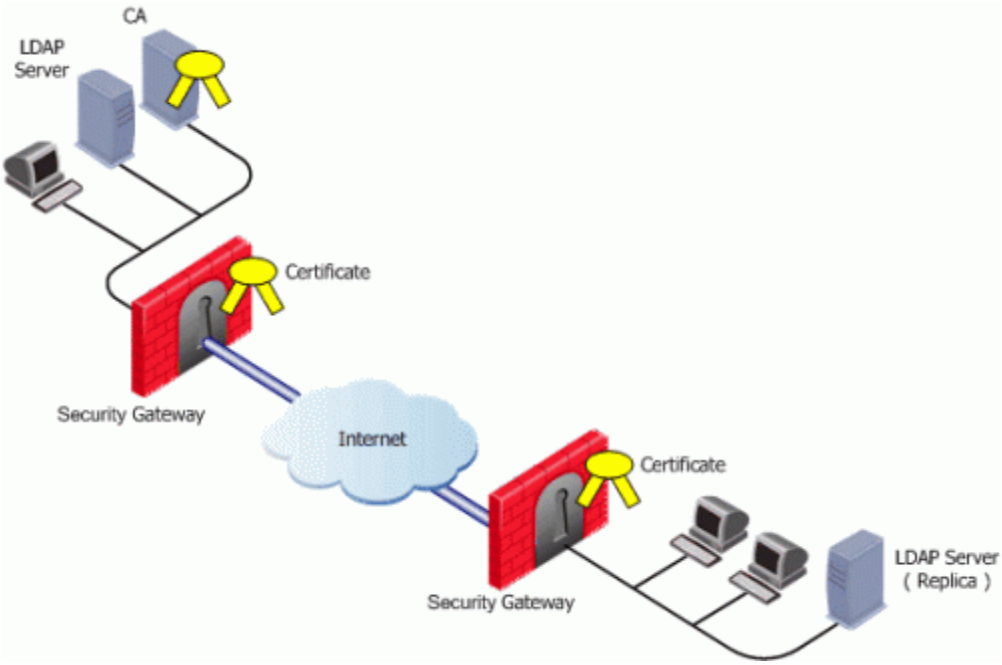
If the certificate of a Security Gateway is issued by a third party CA accessible over the Internet, CA operations such as registration or revocation are usually performed through HTTP forms. CRLs are retrieved from an HTTP server functioning as a CRL repository.



Security Gateways A and B receive their certificates from a PKI service provider accessible via the web. Certificates issued by external CA's may be used by Security Gateways managed by the same Security Management Server to verification.

CA Located on the LAN

If the peer VPN Security Gateway certificate is issued by a third party CA on the LAN, the CRL is usually retrieved from an internal LDAP server, as shown:



Trusting An External CA

A trust relationship is a crucial prerequisite for establishing a VPN tunnel. However, a trust relationship is possible only if the CA that signs the peer's certificate is "trusted." Trusting a CA means obtaining and validating the CA's own certificate. Once the CA's Certificate has been validated, the details on the CA's certificate and its public key can be used to both obtain and validate other certificates issued by the CA.

The Internal CA (ICA) is automatically trusted by all modules managed by the Security Management Server that employs it. External CAs (even the ICA of another Check Point Security Management Server) are not automatically trusted, so a module must first obtain and validate an external CA's certificate. The external CA must provide a way for its certificate to be imported into the Security Management Server.

If the external CA is:

- The ICA of an external Security Management Server, see the *R80.10 Security Management Server Administration Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD for further information
- An OPSEC Certified CA, use the CA options on the **Servers and OSPEC Applications** tab to define the CA and obtain its certificate

Subordinate Certificate Authorities

A subordinate CA is a Certificate Authority certified by another Certificate Authority. Subordinate CAs can issue certificates to other, more subordinate CAs, in this way forming a certification chain or hierarchy. The CA at the top of the hierarchy is the root authority or root CA. Child Certificate Authorities of the root CA are referred to as Subordinate Certificate Authorities.

With the CA options on the **Servers and OSPEC Applications** tab, you can define either a Certificate Authority as either Trusted or Subordinate. Subordinate CAs are of the type OPSEC, and not trusted.

Enrolling a Managed Entity

Enrollment means requesting and obtaining a certificate from a CA, for an entity.

The process of enrollment begins with the generation of a key pair. A certificate request is then created out of the public key and additional information about the module. The type of the certificate request and the rest of the enrollment process depends on the CA type.

The case of an internally managed Security Gateway is the simplest, because the ICA is located on the Security Management Server machine. The enrollment process is completed automatically.

To obtain a certificate from an OPSEC Certified CA, Security Management Server takes the module details and the public key and encodes a PKCS#10 request. The request (which can include *SubjectAltName* for OPSEC certificates and Extended Key Usage extensions) is delivered to the CA manually by the administrator. Once the CA issues the certificate the administrator can complete the process by importing the certificate to the Security Management Server.

A certificate can also be obtained for the Security Gateway using Automatic Enrollment. With Automatic Enrollment, you can automatically issue a request for a certificate from a trusted CA for any Security Gateway in the community. Automatic Enrollment supports the following protocols:

- **SCEP**
- **CMPV1**
- **CMPV2**

Note - During SCEP enrollment, some HTTP requests may be larger than 2K, and may be dropped by the HTTP protocol inspection mechanism if enabled (**Web Intelligence > HTTP Protocol Inspection > HTTP Format Sizes**). A change of the default value will be required to enable these HTTP requests. If enrollment still fails, enrollment must be done manually. For more information, see the *R80.10 IPS Administration Guide* http://supportcontent.checkpoint.com/documentation_download?ID=24806.

Validation of a Certificate

When an entity receives a certificate from another entity, it must:

1. Verify the certificate signature, i.e. verify that the certificate was signed by a trusted CA. If the certificate is not signed directly by a trusted CA, but rather by a subsidiary of a trusted CA, the path of CA certificates is verified up to the trusted CA.
2. Verify that the certificate chain has not expired.
3. Verify that the certificate chain is not revoked. A CRL is retrieved to confirm that the serial number of the validated certificate is not included among the revoked certificates.

In addition, VPN verifies the validity of the certificate's use in the given situation, confirming that:

- The certificate is authorized to perform the required action. For example, if the private key is needed to sign data (e.g., for authentication) the **KeyUsage** extension on the certificate – if present – is checked to see if this action is permitted.
- The peer used the correct certificate in the negotiation. When creating a VPN tunnel with an externally managed module, the administrator may decide that only a certificate signed by a

specific CA from among the trusted CAs can be accepted. (Acceptance of certificates with specific details such as a *Distinguished Name* is possible as well).

Revocation Checking

There are two available methods useful in determining the status of a certificate:

1. CRL
2. Online Certificate Status Protocol (OCSP)

Enrolling with a Certificate Authority

A certificate is automatically issued by the ICA for all internally managed entities that are VPN capable. That is, after the administrator has checked selected **IPsec VPN** in the **Network Security** tab of the **General Properties** page for network objects.

The process for obtaining a certificate from an OPSEC PKI or External Check Point CA is identical.

Manual Enrollment with OPSEC Certified PKI

To create a PKCS#10 Certificate Request:

1. Create the CA object (see "Trusting an OPSEC Certified CA" on page 48).
2. Open the **IPsec VPN** tab of the relevant Network Object.
3. In the **Certificate List** field click **Add**.

   The **Certificate Properties** window is displayed.
4. Enter the **Certificate Nickname**

   The nickname is only an identifier and has no bearing on the content of the certificate.
5. From the **CA to enroll from** drop-down box, select the direct OPSEC CA/External Check Point CA that will issue the certificate.

   Note - The list displays only those subordinate CA's that lead directly to a trusted CA and the trusted CAs themselves. If the CA that issues the certificate is a subordinate CA that does not lead directly to a trusted CA, the subordinate CA will not appear in the list.
6. Choose the appropriate method for Key Pair creation and storage (see "Distributed Key Management and Storage" on page 47).
7. Click **Generate**.

   The **Generate Certificate Properties** window is displayed.
8. Enter the appropriate DN.

   The final DN that appears in the certificate is decided by the CA administrator.

   If a **Subject Alternate Name** extension is required in the certificate, check the **Define Alternate Name** check box.

   Adding the object IP as Alternate name extension can be configured as a default setting by selecting in **Menu** > **Global Properties** > **Advanced** > **Configure > Certificates and PKI properties**, the options:

   add_ip_alt_name_for_opsec_certs

   add_ip_alt_name_for_ICA_certs

   The configuration in this step is also applicable for Internal CAs.
9. Click **OK**.

   The public key and the DN are then used to DER-encode a PKCS#10 Certificate Request.
10. After the Certificate Request is ready, click **View**.

The **Certificate Request View** window appears with the encoding.

11. Copy the whole text in the window and deliver it to the CA.

    The CA administrator must now complete the task of issuing the certificate. Different CAs provide different ways of doing this, such as an advanced enrollment form (as opposed to the regular form for users). The issued certificate may be delivered in various ways, for example email. After the certificate has arrived, it needs to be stored:

    a) In **Object Explorer** (Ctrl+E), go to the **Servers** category and select the CA object.

    b) Open the OPEC PKI tab, click **Get** and browse to the location in which the certificate was saved.

    c) Select the appropriate file and verify the certificate details.

    d) Close object and save.

Automatic Enrollment with the Certificate Authority

On the OPSEC PKI tab of the CA object, make sure **Automatically enroll certificate** is selected and SCEP or CMP are chosen as the connecting protocol. Then:

1. On the relevant network object, open the **VPN** tab.
2. In the **Certificates List** section, click **Add...**

   The **Certificate Properties** window opens.
3. Enter a **Certificate Nickname** (any string used as an identifier)
4. From the drop-down menu, select the CA that issues the certificate.

   Note - The menu shows only trusted CAs and subordinate CAs that lead directly to a trusted CA. If the CA that issues the certificate is a subordinate CA that does not lead directly to a trusted CA, it is not in the menu.
5. Select a method for key pair generation and storage.
6. Click **Generate**, and select **Automatic enrollment**.

   The **Generate Keys and Get Automatic Enrollment Certificate** window opens.
   - Supply the **Key Identifier** and your secret **authorization code**.
   - Click **OK**.
7. When the certificate appears in the **Certificates List** on the network objects VPN page, click **View** and either **Copy to Clipboard** or **Save to File** the text in the **Certificate Request View** window.
8. Send the request to CA administrator.

   Different Certificate Authorities provide different means for doing this, for example an advanced enrollment form on their website. The issued certificate can be delivered in various ways, such as email. Once you have received the certificate, save it to disk.
9. On the **VPN** tab of the network object, select the appropriate certificate in the **Certificates List**, and click **Complete...**
10. Browse to the folder where you stored the issued certificate, select the certificate and verify the certificate details.
11. Close the network object and **Save**.

Enrolling through a Subordinate CA

When enrolling through a subordinate CA:

- Supply the password of the subordinate CA which issues the certificate, not the CA at the top of the hierarchy

- The subordinate CA must lead directly to a trusted CA

CRL

VPN can retrieve the CRL from either an HTTP server or an LDAP server. If the CRL repository is an HTTP server, the module uses the URL published in the CRL **Distribution Point** extension on the certificate and opens an HTTP connection to the CRL repository to retrieve the CRL.

If the CRL repository is an LDAP server, VPN attempts to locate the CRL in one of the defined LDAP account units. In this scenario, an LDAP account unit must be defined. If the CRL **Distribution Point** extension exists, it publishes the DN of the CRL, namely, the entry in the Directory under which the CRL is published or the LDAP URI. If the extension does not exist, VPN attempts to locate the CRL in the entry of the CA itself in the LDAP server.

OCSP

Online Certificate Status Protocol (OCSP) enables applications to identify the state of a certificate. OCSP may be used for more timely revocation information than is possible with CRLs and may also be used to obtain additional status information. When OCSP client issues a status request to an OCSP server, acceptance of the certificate in question is suspended until the server provides a response.

In order to use OCSP, the root CA must be configured to use this method instead of CRL. This setting is inherited by the subordinate CA's.

CRL Prefetch-Cache

Since the retrieval of CRL can take a long time (in comparison to the entire IKE negotiation process), VPN stores the CRLs in a CRL cache so that later IKE negotiations do not require repeated CRL retrievals.

The cache is pre-fetched:

- every two hours
- on policy installation
- when the cache expires

If the pre-fetch fails, the previous cache is not erased.

Note - The ICA requires the use of a CRL cache.

An administrator can shorten the lifetime of a CRL in the cache or even to cancel the use of the cache. If the CRL Cache operation is canceled, the CRL must be retrieved for each subsequent IKE negotiation, thus considerably slowing the establishment of the VPN tunnel. Because of these performance implications, we recommend that you only disable CRL caching when the level of security demands continuous CRL retrieval.

Special Considerations for the CRL Pre-fetch Mechanism

The CRL pre-fetch mechanism makes a "best effort" to obtain the most up to date list of revoked certificates. However, after the **cpstop, cpstart** commands have been executed, the cache is no longer updated. The Security Gateway continues to use the old CRL for as long as the old CRL remains valid (even if there is an updated CRL available on the CA). The pre-fetch cache

mechanism returns to normal functioning only after the old CRL expires and a new CRL is retrieved from the CA.

In case there is a requirement that after **cpstop**, **cpstart** the CRL's will be updated immediately, proceed as follows:

- After executing **cprestart**, run **crl_zap** to empty the cache, or:
- In the SmartConsole **Menu** > **Global Properties** > **Advanced** > **Configure > Certificates and PKI properties** select: **flush_crl_cache_file_on_install**.

When a new policy is installed, the cache is flushed and a new CRL will be retrieved on demand.

CRL Grace Period

Temporary loss of connection with the CRL repository or slight differences between clocks on the different machines may cause valid of CRLs to be considered invalid—and thus the certificates to be invalid as well. VPN overcomes this problem by supplying a CRL Grace Period. During this period, a CRL is considered valid even if it is not valid according to the CLR validity time.

Special Considerations for PKI

Using the Internal CA vs. Deploying a Third Party CA

The Internal CA makes it easy to use PKI for Check Point applications such as site-to-site and remote access VPNs. However, an administrator may prefer to continue using a CA that is already functioning within the organization, for example a CA used to provide secure email, and disk encryption.

Distributed Key Management and Storage

Distributed Key Management (DKM) provides an additional layer of security during the key generation phase. Instead of the Security Management Server generating both public and private keys and downloading them to the module during a policy installation, the management server instructs the module to create its own public and private keys and send (to the management server) only its public key. The private key is created and stored on the module in either a hardware storage device, or via software that emulates hardware storage. Security Management Server then performs certificate enrollment. During a policy installation, the certificate is downloaded to the module. The private key never leaves the module.

Local key storage is supported for all CA types.

DKM is supported for all enrollment methods, and can be configured as a default setting by selecting in **Global Properties > SmartConsole Customization > Configure > Certificates and PKI properties**, the option: **use_dkm_cert_by_default**

Note - Generating certificates for Edge devices does not support DKM and will be generated locally on the management even if **use_dkm_cert_by_default** is configured.

Configuration of PKI Operations

Trusting a CA – Step-By-Step

This section describes the procedures for obtaining a CA's own certificate, which is a prerequisite for trusting certificates issued by a CA.

In order to trust a CA, a CA server object has to be defined. The following sections deal with the various configuration steps required in different scenarios.

Trusting an ICA

A VPN module automatically trusts the ICA of the Security Management Server that manages it. No further configuration is required.

Trusting an Externally Managed CA

An externally managed CA is the ICA of another Security Management Server. The CA certificate has to be supplied and saved to disk in advance. To establish trust:

1. In **Object Explorer** (Ctrl+E), click **New** > **Server** > **More** > **Trusted CA**.
   The **Certificate Authority Properties** window opens.
2. Enter a **Name** for the CA object
3. Go to the **OPSEC PKI** tab and click **Get...**
4. Browse to where you saved the peer CA certificate and select it.
   The certificate details are shown. Verify the certificate's details. Display and validate the SHA-1 and MD5 fingerprints of the CA certificate.
5. Click **OK**.

Trusting an OPSEC Certified CA

The CA certificate has to be supplied and saved to the disk in advance.

Note - In case of SCEP automatic enrollment, you can skip this stage and fetch the CA certificate automatically after configuring the SCEP parameters.

The CA's Certificate must be retrieved either by downloading it using the CA options in the Certificate Authority object, or by obtaining the CA's certificate from the peer administrator in advance.

Then define the CA object according to the following steps:

1. In **Object Explorer** (Ctrl+E), click **New** > **Server** > **More** > **Trusted CA** or **Subordinate CA**.
   The **Certificate Authority Properties** window opens.
2. Enter a **Name** for the CA object.
3. On the **OPSEC PKI** tab:
   - For automatic enrollment, select **automatically enroll certificate**.
   - From the **Connect to CA with protocol**, select the protocol used to connect with the certificate authority, either SCEP, CMPV1 or CMPV2.

   Note - For entrust 5.0 and later, use CMPV1
4. Click **Properties**.

- **If you chose SCEP as** the protocol, in the **Properties for SCEP protocol** window, enter the CA identifier (such as example.com) and the Certification Authority/Registration Authority URL.
- If you chose cmpV1 as the protocol, in the **Properties for CMP protocol - V1** window, enter the appropriate IP address and port number. (The default port is 829).
- If you chose cmpV2 as the protocol, in the **Properties for CMP protocol -V2** window, decide whether to use direct TCP or HTTP as the transport layer.

  Note - If Automatic enrollment is not selected, then enrollment will have to be performed manually.

5. Choose a method for retrieving CRLs from this CA.

   If the CA publishes CRLs on HTTP server choose **HTTP Server(s)**. Certificates issued by the CA must contain the CRL location in an URL in the **CRL Distribution Point** extension.

   If the CA publishes CRL on LDAP server, choose **LDAP Server(s)**. In this case, you must define an LDAP Account Unit as well. See the *Security Management Server Administration Guide* for more details about defining an LDAP object.

   Make sure that **CRL retrieval** is checked in the **General** tab of the **LDAP Account Unit Properties** window.

   Certificates issued by the CA must contain the LDAP DN on which the CRL resides in the CRL distribution point extension.
6. Click **Get**.
7. If SCEP is configured, it will try to connect to the CA and retrieve the certificate. If not, browse to where you saved the peer CA certificate and select it.

   VPN reads the certificate and displays its details. Verify the certificate's details. Display and validate the SHA-1 and MD5 fingerprints of the CA certificate.
8. Click **OK**.
9. Install the Access Control Policy on the Security Gateway.

Certificate Revocation (All CA Types)

A certificate issued by the Internal Certificate Authority it is revoked when the certificate object is removed. Otherwise, certificate revocation is controlled by the CA administrator using the options on the **Advanced** tab of the CA object. In addition, the certificate must be removed from the Security Gateway.

A certificate cannot be removed if the Security Management Server infers from other settings that the certificate is in use, for example, that the Security Gateway belongs to one or more VPN communities and this is the only certificate of the Security Gateway.

To remove the certificate:

1. Open the **IPsec VPN** tab of the relevant Security Gateway.
2. In the Repository of **Certificates Available to the Gateway**, select the appropriate certificate and click **Remove**.

Certificate Recovery and Renewal

When a certificate is revoked or becomes expired, it is necessary to create another one or to refresh the existing one.

Recovery and Renewal with Internal CA

Removal of a compromised or expired certificate automatically triggers creation of a new certificate, with no intervention required by the administrator. To manually renew a certificate use the **Renew...** button on the VPN page of the Security Gateway object.

Note - A Security Gateway can have only one certificate signed by one CA. When the new certificate is issued, you will be asked to replace the existing certificate signed by the same CA.

CA Certificate Rollover

CA Certificate Rollover is a VPN feature that enables rolling over the CA certificates used to sign client and Security Gateway certificates for VPN traffic, without risk of losing functionality at transition.

To achieve a gradual CA certificate rollover, CA Certificate Rollover enables VPN support for multiple CA certificates generated by third-party OPSEC-compliant CAs, such as Microsoft Windows CA. By using multiple CA certificates, you can gradually rollover client and Security Gateway certificates during a transitional period when client and Security Gateway certificates signed by both CA certificates are recognized.

When a certificate is added to a CA that already has a certificate, the new certificate is defined as Additional and receives an index number higher by one than the highest existing certificate index number. The original certificate is defined as Main.

Only additional certificates can be removed. CA Certificate Rollover provides tools for adding and removing certificates, and for changing the status of a certificate from additional to main and from main to additional.

CA Certificate Rollover is for rolling over CA certificates with different keys. To add a CA certificate using the same key as the existing CA certificate (for example, to extend its expiration date), just Get the certificate from the OPSEC PKI tab of the CA properties, and do not use CA Certificate Rollover.

Managing a CA Certificate Rollover

By using multiple CA certificates, you can gradually rollover client and Security Gateway certificates during a transitional period when client and Security Gateway certificates signed by both CA certificates are recognized.

This section describes a recommended workflow for the most common scenario. For full details of the CLI commands, see CA Certificate Rollover CLI (on page 51).

Before you begin:

In SmartConsole, define a third-party OPSEC-compliant CA, such as Microsoft Windows CA, that is capable of generating multiple CA certificates. Generate the main CA certificate and define it in SmartConsole.

To roll over with a new CA certificate:

1. Generate from the third-party CA a second CA certificate in DER format (PEM is not supported), with different keys than the previous CA certificate. Copy the certificate to the Security Management Server.

2. Log into the Security Management Server, and stop Check Point processes:
   `cpstop`
3. Back up the Security Management Server database:
   `mcc backup`
4. Add the new CA certificate to the Security Management Server database's definitions for the third-party CA:
   `mcc add <CA> <CertificateFile>`
5. <CA> is the name of the CA as defined in the Security Management Server database. <CertificateFile> is the certificate filename or path.
6. The new CA certificate should now be defined as additional #1. Make sure with `mcc lca` or `mcc show` ("CA Certificate Rollover CLI" on page 51).
7. Start Check Point processes:
   `cpstart`
8. Install policy on all Security Gateways.

Use the new additional CA certificate to sign client certificates.

For performance reasons, as long as most clients are still using certificates signed by the old CA certificate, you should leave the new CA certificate as the additional one and the old certificate as the main one. As long as the new CA certificate is not the main CA certificate, do not use it to sign any Security Gateway certificates.

CA Certificate Rollover CLI

CA Certificate Rollover uses the VPN Multi-Certificate CA command set, as detailed in this section. VPN Multi-Certificate CA configuration commands should not be run when SmartConsole or Database Tool are logged in to the Security Management Server, or when Check Point processes are running.

To see usage instructions in the CLI, run the following without arguments:

```
mcc
```

Adding Matching Criteria to the Validation Process

While certificates of an externally managed VPN entity are not handled by the local Security Management Server, you can still configure a peer to present a particular certificate when creating a VPN tunnel:

1. Open the **VPN** page of the externally managed VPN entity.
2. Click **Matching Criteria...**
3. Choose the desired characteristics of the certificate the peer is expected to present, including:
   - The CA that issued it
   - The exact DN of the certificate
   - The IP address that appears in the **Subject Alternate Name** extension of the certificate. (This IP address is compared to the IP address of the VPN peer itself as it appears to the VPN module during the IKE negotiation.)
   - The e-mail address appearing in the **Subject Alternate Name** extension of the certificate

CRL Cache Usage

To cancel or modify the behavior of the CRL Cache:

1. Open the **Advanced Tab** of the Certificate Authority object.
2. To enable the CRL cache, check **Cache CRL on the module**.

   The cache should not be disabled for the ICA. In general, it is recommended that the cache be enabled for all CA types. The cache should be disabled (for non-ICAs) only if stringent security requirements mandate continual retrieval of the CRL.

   Note - The ICA requires the use of a CRL cache, and should never be disabled.
3. If CRL Cache is enabled, choose whether a CRL is deleted from the cache when it expires or after a fixed period of time (unless it expires first). The second option encourages retrieval of a CRL more often as CRLs may be issued more frequently than the expiry time. By default a CRL is deleted from the cache after 24 hours.

See: CRL Prefetch-Cache (on page 46) for information about CRL caching.

Modifying the CRL Pre-Fetch Cache

The behavior of the Pre-fetch catch can be altered via the Global properties:

1. Go to **Menu** > **Global Properties** > **Advanced** > **Configure > Certificates and PKI properties**
2. Select **Certificates and PKI Properties**.

Configuring CRL Grace Period

Set the CRL Grace Period values by selecting **Policy > Global Properties > VPN > Advanced**. The Grace Period can be defined for both the periods before and after the specified CRL validity period.

Configuring OCSP

To use OCSP, you must configure the CA object to use the OCSP revocation information method instead of the CRL method.

Use **GuiDBedit** to change the value of the field **ocsp_validation** to **true**. When set to true, the CA uses OCSP to make sure that certificates are valid. This is configured on the root CA and is inherited by the subordinate CAs.

To configure a CA to use OCSP, in GuiDBedit:

See sk37803 http://supportcontent.checkpoint.com/solutions?id=sk37803 for detailed instructions.

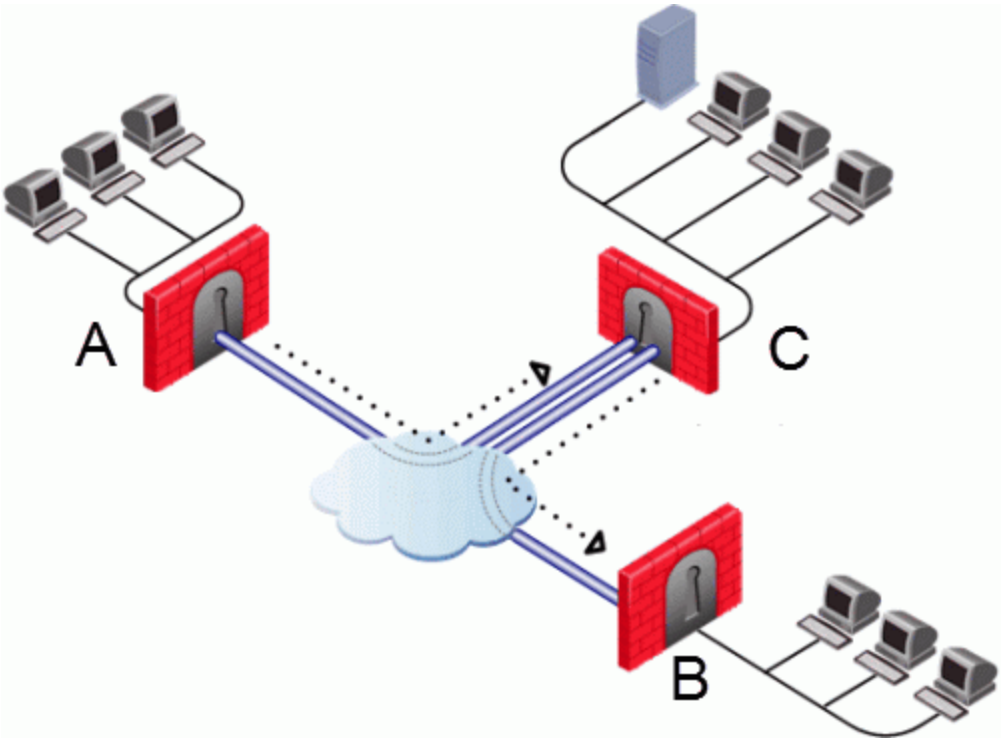
Domain Based VPN

In This Section:

Overview of Domain-based VPN

Domain Based VPN controls how VPN traffic is routed between Security Gateways and remote access clients within a community. To route traffic to a host behind a Security Gateway, you must first define the VPN domain for that Security Gateway. Configuration for VPN routing is done with SmartConsole or in the VPN routing configuration files on the Security Gateways.

In this figure, one of the host machines behind Security Gateway A tries to connect to a host computer behind Security Gateway B. For technical or policy reasons, Security Gateway A cannot establish a VPN tunnel with Security Gateway B. With VPN Routing, Security Gateways A and B can establish VPN tunnels through Security Gateway C.



Item	
A	Security Gateway A
B	Security Gateway B
C	Security Gateway C

VPN Routing and Access Control

VPN routing connections are subject to the same access control rules as any other connection. If VPN routing is correctly configured but a Security Policy rule exists that does not allow the connection, the connection is dropped. For example: a Security Gateway has a rule which forbids all FTP traffic from inside the internal network to anywhere outside. When a peer Security Gateway opens an FTP connection with this Security Gateway, the connection is dropped.

For VPN routing to succeed, a single rule in the Security Policy Rule Base must cover traffic in both directions, inbound and outbound, and on the central Security Gateway. To configure this rule, see Configuring the 'Accept VPN Traffic Rule (see "Configuring the 'Accept VPN Traffic Rule'" on page 55).

Configuring Domain Based VPN

Configure most common VPN routing scenarios through a VPN star community in SmartConsole.

You can also configure VPN routing between Security Gateways in the configuration file **$FWDIR/conf/vpn_route.conf**.

You can only configure VPN routing between Security Gateways that belong to a VPN community.

Configuring VPN Routing for Security Gateways through SmartConsole

To configure a VPN Routing in a star community in SmartConsole:

1. On the **Star Community** window, in the:
   a) **Center Gateways** section, select the Security Gateway that functions as the "Hub".
   b) **Satellite Gateways** section, select Security Gateways as the "spokes", or satellites.
2. On the **VPN Routing** page, **Enable VPN routing for satellites** section, select one of these options:
   - **To center and to other Satellites through center -** This allows connectivity between the Security Gateways, for example if the spoke Security Gateways are DAIP Security Gateways, and the Hub is a Security Gateway with a static IP address.
   - **To center, or through the center to other satellites, to internet and other VPN targets -** This allows connectivity between the Security Gateways as well as the ability to inspect all communication passing through the Hub to the Internet.
3. Create an appropriate Access Control Policy rule. Remember: one rule must cover traffic in both directions.
4. NAT the satellite Security Gateways on the Hub if the Hub is used to route connections from Satellites to the Internet.

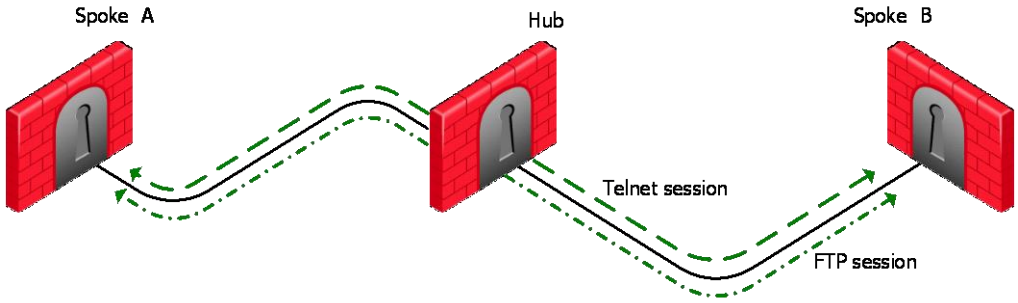
The two Dynamic Objects (DAIP Security Gateways) can securely route communication through the Security Gateway with the static IP address.

Configuration through the VPN Configuration File

For more granular control over VPN routing, edit the **vpn_route.conf** file in the **conf** directory of the Security Management Server.

The configuration file, **vpn_route.conf**, is a text file that contains the name of network objects. The format is: **Destination**, **Next hop**, **Install on Security Gateway** (with tabbed spaces separating the elements).

Consider a simple VPN routing scenario consisting of Center gateway (hub) and two Satellite gateways (spokes). All machines are controlled from the same Security Management Server, and all the Security Gateways are members of the same VPN community. Only Telnet and FTP services are to be encrypted between the Satellites and routed through the Center:



Although this can be done easily in a VPN star community, the same goal can be achieved by editing **vpn_route.conf**:

Destination	Next hop router interface	Install on
Spoke_B_VPN_Dom	Hub_C	Spoke_A
Spoke_A_VPN_Dom	Hub_C	Spoke_B

In this instance, Spoke_B_VPN_Dom is the name of the network object group that contains spoke B's VPN domain. Hub C is the name of the Security Gateway enabled for VPN routing. Spoke_A_VPN_Dom is the name of the network object that represents Spoke A's encryption domain. For an example of how the file appears:

```
Spoke_B_VPN_DOM   Hub_C        Spoke_A
Spoke_A_VPN_DOM   Hub_C        Spoke_ B
```

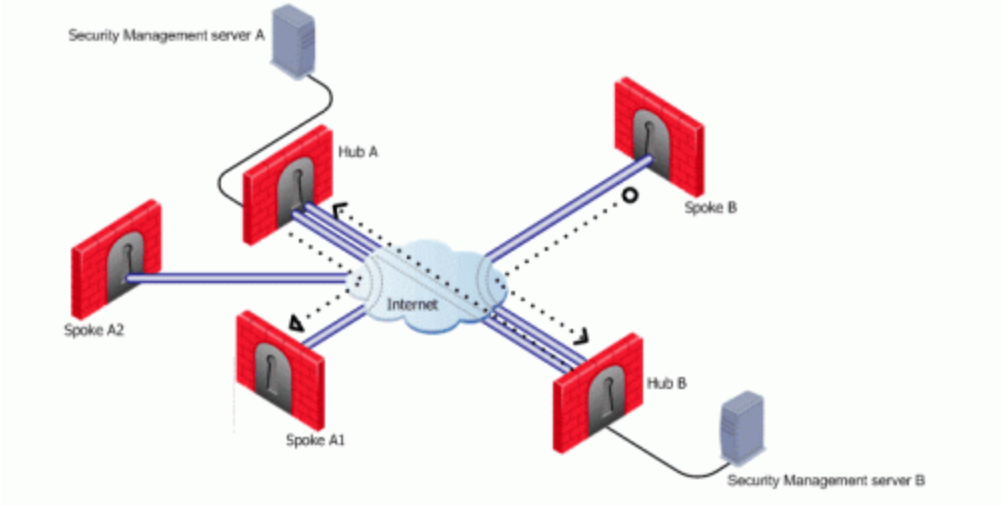
Configuring the 'Accept VPN Traffic Rule'

In SmartConsole:

1. Double click on a Star or Meshed Community.
2. On the **Encrypted Traffic** page, select **Accept all encrypted traffic**.
3. In a Star community, choose between accepting encrypted traffic on **Both center and satellite gateways** or **Satellite gateways only**.
4. Click **OK**.

Configuring Multiple Hubs

Consider two Hubs, A and B. Hub A has two spokes, spoke_A1, and spoke_A2. Hub B has a single spoke, spoke_B. In addition, Hub A is managed from Security Management Server A, while Hub B is managed via Security Management Server B:



For the two VPN star communities, based around Hubs A and B:

- Spokes A1 and A2 need to route all traffic going outside of the VPN community through Hub A
- Spokes A1 and A2 also need to route all traffic to one another through Hub A, the center of their star community
- Spoke B needs to route all traffic outside of its star community through Hub B

A_community is the VPN community of A plus the spokes belonging to A. B_community is the VPN community. Hubs_community is the VPN community of Hub_A and Hub_B.

Configuring VPN Routing and Access Control on Security Management Server A

The **vpn_route.conf** file on Security Management Server 1 looks like this:

Destination	Next hop router interface	Install on
Spoke_B_VPN_Dom	Hub_A	A_Spokes
Spoke_A1_VPN_Dom	Hub_A	Spoke_A2
Spoke_A2_VPN_Dom	Hub_A	Spoke _A1
Spoke_B_VPN_Dom	Hub_B	Hub_A

Spokes A1 and A2 are combined into the network group object "A_spokes". The appropriate rule in the Security Policy Rule Base looks like this:

Source	Destination	VPN	Service	Action
Any	Any	A_Community B_Community Hubs_Community	Any	Accept

Configuring VPN Routing and Access Control on Security Management Server B

The **vpn_route.conf** file on Security Management Server 2 looks like this:

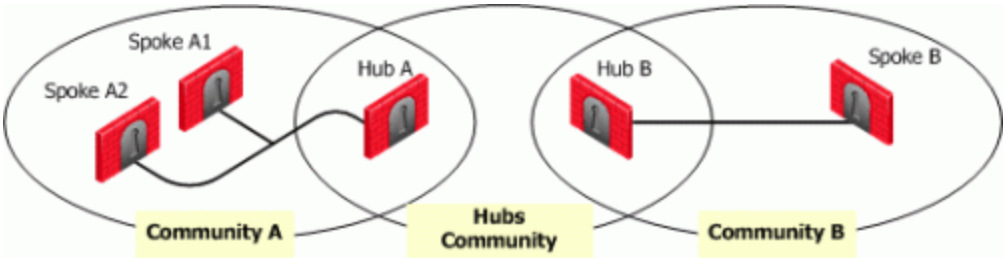
Destination	Next hop router interface	Install On
Spoke_A1_VPN_Dom	Hub_B	Spoke_B
Spoke_A2_VPN_Dom	Hub_B	Spoke_B
Spoke_A1_VPN_Dom	Hub_A	Hub_B
Spoke_A2_VPN_Dom	Hub_A	Hub_B

The appropriate rule in the Security Policy Rule Base looks like this:

Source	Destination	VPN	Service	Action
Any	Any	B_Community A_Community Hubs_Community	Any	Accept

For both **vpn_route.conf** files:

- "A_Community" is a star VPN community comprised of Hub_A, Spoke_A1, and Spoke_A2
- "B_Community" is a star VPN community comprised of Hub_B and Spoke_B
- "Hubs-Community" is a *meshed* VPN community comprised of Hub_A and Hub_B (it could also be a star community with the central Security Gateways meshed).



Route Based VPN

In This Section:

Overview of Route-based VPN

The use of VPN Tunnel Interfaces (VTI) is based on the idea that setting up a VTI between peer Security Gateways is similar to connecting them directly.

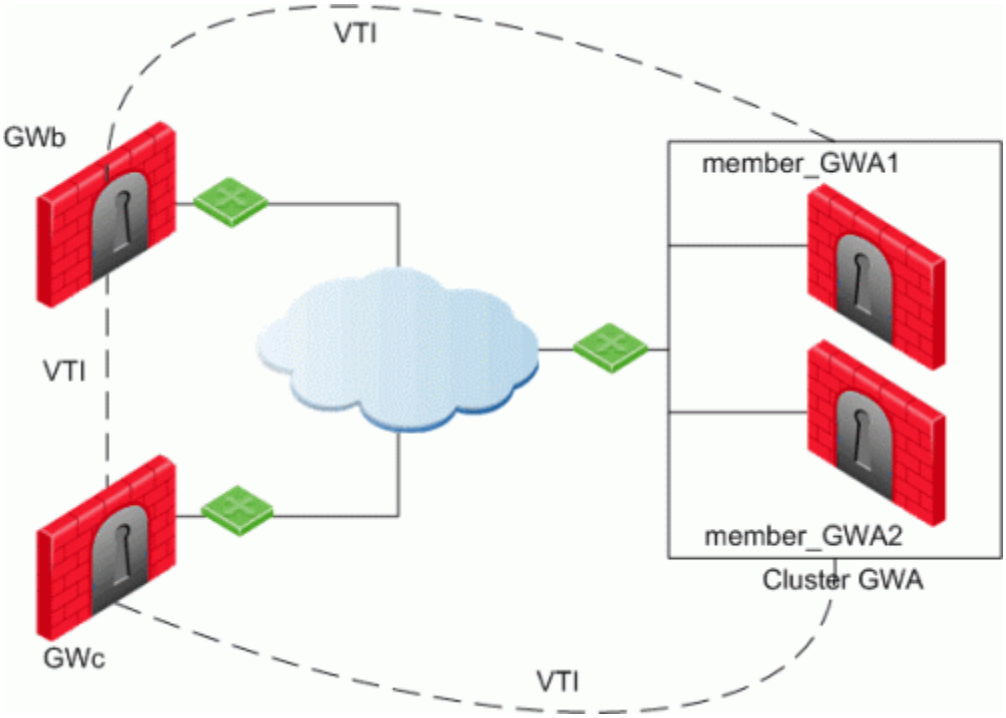
A VTI is an operating system level virtual interface that can be used as a Security Gateway to the VPN domain of the peer Security Gateway. Each VTI is associated with a single tunnel to a Security Gateway. The tunnel itself with all of its properties is defined, as before, by a VPN Community linking the two Security Gateways. Configure the peer Security Gateway with a corresponding VTI. The native IP routing mechanism on each Security Gateway can then direct traffic into the tunnel as it would for other interfaces.

All traffic destined to the VPN domain of a peer Security Gateway is routed through the "associated" VTI. This infrastructure allows dynamic routing protocols to use VTIs. A dynamic routing protocol daemon running on the Security Gateway can exchange routing information with a neighboring routing daemon running on the other end of an IPsec tunnel, which appears to be a single hop away.

Route Based VPN can only be implemented between two Security Gateways within the same community.

VPN Tunnel Interface (VTI)

A VPN Tunnel Interface is a virtual interface on a Security Gateway that is related to a VPN tunnel and connects to a remote peer. You create a VTI on each Security Gateway that connects to the VTI on a remote peer. Traffic routed from the local Security Gateway via the VTI is transferred encrypted to the associated peer Security Gateway.



In this scenario:

- There is a VTI connecting Cluster GWA and GWb
- There is a VTI connecting Cluster GWA and GWc
- There is a VTI connecting GWb and GWc

A virtual interface behaves like a point-to-point interface directly connected to the remote peer. Traffic between network hosts is routed into the VPN tunnel using the IP routing mechanism of the Operating System. Security Gateway objects are still required, as well as VPN communities (and access control policies) to define which tunnels are available. However, VPN encryption domains for each peer Security Gateway are no longer necessary. The decision whether or not to encrypt depends on whether the traffic is routed through a virtual interface. The routing changes dynamically if a dynamic routing protocol (OSPF/BGP) is available on the network.

When a connection that originates on GWb is routed through a VTI to GWc (or servers behind GWc) and is accepted by the implied rules, the connection leaves GWb in the clear with the local IP address of the VTI as the source IP address. If this IP address is not routable, return packets will be lost.

The solution for this issue is:

- Configure a static route on GWb that redirects packets destined to GWc from being routed through the VTI
- Not including it in any published route
- Adding route maps that filter out GWc's IP addresses

Having excluded those IP addresses from route-based VPN, it is still possible to have other connections encrypted to those addresses (i.e. when not passing on implied rules) by using domain based VPN definitions.

The VTI can be configured in two ways:

- Numbered
- Unnumbered

Numbered VTI

You configure a local and remote IP address for each numbered VPN Tunnel Interface (VTI). For each Security Gateway, you configure a local IP address, a remote address, and the local IP address source for outbound connections to the tunnel. The remote IP address must be the local IP address on the remote peer Security Gateway. More than one VTI can use the same IP Address, but they cannot use an existing physical interface IP address.

Unnumbered VTI

For unnumbered VTIs, you define a proxy interface for each Security Gateway. Each Security Gateway uses the proxy interface IP address as the source for outbound traffic. Unnumbered interfaces let you assign and manage one IP address for each interface. Proxy interfaces can be physical or loopback interfaces.

Using Dynamic Routing Protocols

VTIs allow the ability to use Dynamic Routing Protocols to exchange routing information between Security Gateways. The Dynamic Routing Protocols supported on Gaia are:

- BGP4
- OSPFv2
- RIPv1
- RIPv2

Configuring Numbered VTIs

Route Based VPN can only be implemented between two Security Gateways within the same community.

Enabling Route Based VPN

If you configure a Security Gateway for Domain Based VPN and Route Based VPN, Domain Based VPN takes precedence by default. To force Route Based VPN to take priority, you must create a dummy (empty) group and assign it to the VPN domain.

To force Route-Based VPN to take priority:

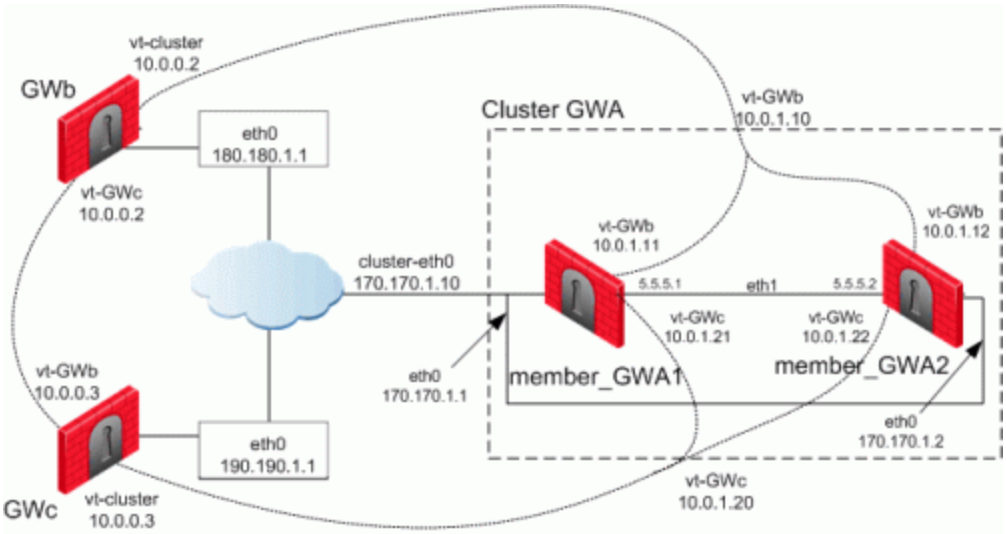
1. In the **Gateways & Servers** view, edit a Check Point Security Gateway.
2. Go to the **Network Management** > **VPN Domain** page.
3. Select **Manually define**.

4. Click **New** > **Group** > **Simple Group**.
5. Enter a **Name** and click **OK**.

Configuring Numbered VTIs

Using the new VPN Command Line Interface (VPN Shell), the administrator creates a VPN Tunnel Interface on the enforcement module for each peer Security Gateway, and "associates" the interface with a peer Security Gateway. The VPN Tunnel Interface may be numbered or unnumbered. For more information on the VPN Shell, see VPN Shell.

Every numbered VTI is assigned a local IP Address and a remote IP Address. Prior to configuration, a range of IP Addresses must be configured to assign to the VTIs.



A VTI connects:

- Cluster GWA and GWb
- Cluster GWA and GWc
- GWb and GWc

The devices in this scenario are:

ClusterXL:

- Cluster GWA
- member_GWA1
- member_GWA2

VPN Modules:

- GWb
- GWc

IP Configurations:

- Cluster GWA
- member_GWA1
- External Unique IP eth0: 170.170.1.1/24
- External VIP eth0: 170.170.1.10/24
- Sync Interface eth1: 5.5.5.1/24

- IP of VTI vt-GWb: Local: 10.0.1.11, Remote: 10.0.0.2
- VIP of VTI vt-GWb: 10.0.1.10
- IP of VTI vt-GWc: Local: 10.0.1.21, Remote: 10.0.0.3
- VIP of VTI vt-GWc: 10.0.1.20
- member_GWA2
- External Unique IP eth0: 170.170.1.2/24
- External VIP eth0: 170.170.1.10/24
- Sync Interface eth1: 5.5.5.1/24
- IP of VTI vt-GWb: Local: 10.0.1.12, Remote: 10.0.0.2
- VIP of VTI vt-GWb: 10.0.1.10
- IP of VTI vt-GWc: Local: 10.0.1.22, Remote: 10.0.0.3
- VIP of VTI vt-GWc: 10.0.1.20
- GWb
- External Unique IP eth0: 180.180.1.1/24
- IP of VTI vt-ClusterGWa: Local: 10.0.0.2, Remote: 10.0.1.10
- IP of VTI vt-GWc: Local: 10.0.0.2, Remote: 10.0.0.3
- GWc
- External Unique IP eth0: 190.190.1.1/24
- IP of VTI vt-ClusterGWa: Local: 10.0.0.3, Remote: 10.0.1.20
- IP of VTI vt-GWb: Local: 10.0.0.3, Remote: 10.0.0.2

VTIs in a Clustered Environment

When configuring numbered VTIs in a clustered environment, a number of issues need to be considered:

- Each member must have a unique source IP address.
- Every interface on each member requires a unique IP address.
- All VTIs going to the same remote peer must have the same name.
- Cluster IP addresses are required.

Configuring VTIs in a Clustered Environment

The following sample configurations use the same Security Gateway names and IP addresses used referred to in: Numbered VTIs (see "Configuring Numbered VTIs" on page 61)

Configuring member_GWA1

```
--------- Access the VPN shell Command Line Interface
[member_GWa2]# vpn shell
 ?              - This help
 ..             - Go up one level
 quit           - Quit
[interface   ] - Manipulate tunnel interfaces
```

```
[show        ] - Show internal data
[tunnels     ] - Manipulate tunnel data
--------- Add vt-GWb
VPN shell:[/] > /interface/add/numbered 10.0.1.12 10.0.0.2 GWb
Interface 'vt-GWb' was added successfully to the system
--------- Add vt-GWc
VPN shell:[/] > /interface/add/numbered 10.0.1.22 10.0.0.3 GWc
Interface 'vt-GWc' was added successfully to the system
---------- Verify configuration
VPN shell:[/] > /show/interface/detailed all
vt-GWb     Type:numbered  MTU:1500
           inet addr:10.0.1.12  P-t-P:10.0.0.2  Mask:255.255.255.255
           Peer:GWb  Peer ID:180.180.1.1  Status:attached

vt-GWc     Type:numbered  MTU:1500
           inet addr:10.0.1.22  P-t-P:10.0.0.3  Mask:255.255.255.255
           Peer:GWc  Peer ID:190.190.1.1  Status:attached

VPN shell:[/] > /quit
[member_GWa2]# ifconfig vt-GWb
vt-GWb    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.1.12  P-t-P:10.0.0.2  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)

[member_GWa2]# ifconfig vt-GWc
vt-GWc    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.1.22  P-t-P:10.0.0.3  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)
```

Configuring member_GWA2

```
 --------- Access the VPN shell Command Line Interface
[member_GWa2]# vpn shell
 ?             - This help
 ..            - Go up one level
 quit          - Quit                                  .
[interface   ] - Manipulate tunnel interfaces
[show        ] - Show internal data
[tunnels     ] - Manipulate tunnel data
--------- Add vt-GWb
VPN shell:[/] > /interface/add/numbered 10.0.1.12 10.0.0.2 GWb
Interface 'vt-GWb' was added successfully to the system
--------- Add vt-GWc
VPN shell:[/] > /interface/add/numbered 10.0.1.22 10.0.0.3 GWc
Interface 'vt-GWc' was added successfully to the system
---------- Verify configuration
VPN shell:[/] > /show/interface/detailed all
vt-GWb     Type:numbered  MTU:1500
           inet addr:10.0.1.12  P-t-P:10.0.0.2  Mask:255.255.255.255
           Peer:GWb  Peer ID:180.180.1.1  Status:attached

vt-GWc     Type:numbered  MTU:1500
           inet addr:10.0.1.22  P-t-P:10.0.0.3  Mask:255.255.255.255
           Peer:GWc  Peer ID:190.190.1.1  Status:attached

VPN shell:[/] > /quit
[member_GWa2]# ifconfig vt-GWb
```

```
vt-GWb    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.1.12  P-t-P:10.0.0.2  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)

[member_GWa2]# ifconfig vt-GWc
vt-GWc    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.1.22  P-t-P:10.0.0.3  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)
```

When configuring a VTI in a clustered environment and an interface name is not specified, a name is provided. The default name for a VTI is "vt-[peer Security Gateway name]". For example, if the peer Security Gateway's name is Server_2, the default name of the VTI is 'vt-Server_2'. For peer Security Gateways that have names that are longer than 12 characters, the default interface name is the last five characters plus a 7 byte hash of the peer name calculated to the give the interface a unique name.

After configuring the VTIs on the cluster members, you must configure in the SmartConsole the VIP of these VTIs.

In SmartConsole:

1. In the **Gateways & Servers** view, edit the Check Point Cluster.
2. In **Network Management** window, click **Get Interfaces**.

   The VTIs are shown in the Topology column as **Point to point**.

   Interfaces are members of the same VTI if these criteria match:

   - Remote peer name
   - Remote IP address
   - Interface name
3. Configure the VTI VIP. Select the interface and click **Edit**. Edit the interface in the **General** page of the interface object.
4. Click **OK** and install policy.

Configuring GWb

```
 --------- Access the VPN shell Command Line Interface
[GWb]# vpn shell
 ?             - This help
 ..            - Go up one level
 quit          - Quit
[interface   ] - Manipulate tunnel interfaces
[show        ] - Show internal data
[tunnels     ] - Manipulate tunnel data
--------- Add vt-GWa
VPN shell:[/] > /interface/add/numbered 10.0.0.2 10.0.1.10 GWa
Interface 'vt-GWa' was added successfully to the system
--------- Add vt-GWc
VPN shell:[/] > /interface/add/numbered 10.0.0.2 10.0.0.3 GWc
Interface 'vt-GWc' was added successfully to the system
---------- Verify configuration
VPN shell:[/] > /show/interface/detailed all
vt-GWa     Type:numbered  MTU:1500
           inet addr:10.0.0.2  P-t-P:10.0.1.10  Mask:255.255.255.255
```

```
           Peer:GWa  Peer ID:170.170.1.10  Status:attached

vt-GWc     Type:numbered  MTU:1500
           inet addr:10.0.0.2  P-t-P:10.0.0.3  Mask:255.255.255.255
           Peer:GWc  Peer ID:190.190.1.1  Status:attached

VPN shell:[/] > /quit
[GWb]# ifconfig vt-GWa
vt-GWa    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.0.2  P-t-P:10.0.1.10  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)

[GWb]# ifconfig vt-GWc
vt-GWc    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.0.2  P-t-P:10.0.0.3  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)
```

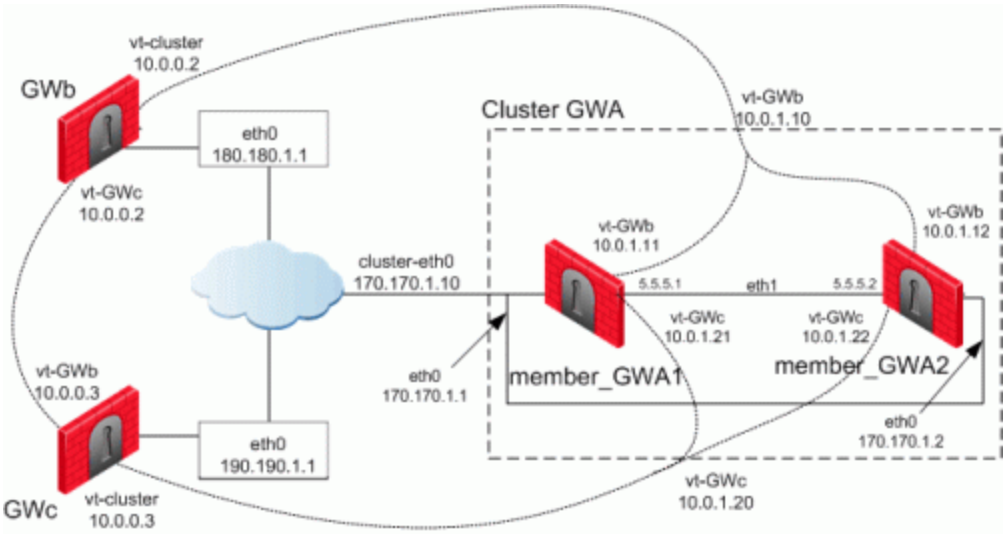
Configuring GWc

```
 --------- Access the VPN shell Command Line Interface
[GWc]# vpn shell
 ?             - This help
 ..            - Go up one level
 quit          - Quit
[interface   ] - Manipulate tunnel interfaces
[show        ] - Show internal data
[tunnels     ] - Manipulate tunnel data
--------- Add vt-GWa
VPN shell:[/] > /interface/add/numbered 10.0.0.3 10.0.1.20 GWa
Interface 'vt-GWa' was added successfully to the system
--------- Add vt-GWb
VPN shell:[/] > /interface/add/numbered 10.0.0.3 10.0.0.2 GWb
Interface 'vt-GWb' was added successfully to the system
---------- Verify configuration
VPN shell:[/] > /show/interface/detailed all
vt-GWa     Type:numbered  MTU:1500
           inet addr:10.0.0.3  P-t-P:10.0.1.20  Mask:255.255.255.255
           Peer:GWa  Peer ID:170.170.1.10  Status:attached

vt-GWb     Type:numbered  MTU:1500
           inet addr:10.0.0.3  P-t-P:10.0.0.2  Mask:255.255.255.255
           Peer:GWb  Peer ID:180.180.1.1  Status:attached

VPN shell:[/] > /quit
[GWc]# ifconfig vt-GWa
vt-GWa    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.0.3  P-t-P:10.0.1.20  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)

[GWc]# ifconfig vt-GWb
vt-GWb    Link encap:IPIP Tunnel  HWaddr
          inet addr:10.0.0.3  P-t-P:10.0.0.2  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
```

```
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:36 (36.0 b)
```

Enabling Dynamic Routing Protocols on VTIs

Using the example:



The following tables illustrate how the OSPF dynamic routing protocol is enabled on VTIs both for single members and for cluster members using SecurePlatform. Note that the network commands for single members and cluster members are not the same.

For more information on advanced routing commands and syntaxes, see the *Check Point Advanced Routing Suite - Command Line Interface* book.

To learn about enabling dynamic routing protocols on VTIs in Gaia environments, see VPN Tunnel Interfaces in the *R80.10 Gaia Administration Guide* http://supportcontent.checkpoint.com/documentation_download?ID=TBD.

When peering with a Cisco GRE enabled device, a point to point GRE tunnel is required. Use the following command to configure the tunnel interface definition:

ip ospf network point-to-point

```
--------- Launch the Dynamic Routing Module
[member_GWa1]# expert
Enter expert password:

You are in expert mode now.

[Expert@member_GWa1]# cligated
localhost>enable
localhost#configure terminal
--------- Enable OSPF and provide an OSPF router ID
localhost(config)#router ospf 1
localhost(config-router-ospf)#router-id 170.170.1.10
--------- Define interfaces/IP's on which OSPF runs (Use the cluster IP as
defined in topology) and the area ID for the interface/IP
localhost(config-router-ospf)#network 10.0.1.10 0.0.0.0 area 0.0.0.0
localhost(config-router-ospf)#network 10.0.1.20 0.0.0.0 area 0.0.0.0
--------- Redistribute kernel routes (this is only here as an example, please
see the dynamic routing book for more specific commands concerning
redistribution of routes)
```

```
localhost(config-router-ospf)#redistribute kernel
localhost(config-router-ospf)#exit
localhost(config)#exit
-------- Write configuration to disk
localhost#write memory
IU0 999 Configuration written to '/etc/gated.ami'
localhost#quit
```

Dynamic Routing on member_GWA2

```
 --------- Launch the Dynamic Routing Module
[member_GWa2]# expert
Enter expert password:

You are in expert mode now.

[Expert@member_GWa2]# cligated
localhost>enable
localhost#configure terminal
--------- Enable OSPF and provide an OSPF router ID
localhost(config)#router ospf 1
localhost(config-router-ospf)#router-id 170.170.1.10
--------- Define interfaces/IP's on which OSPF runs (Use the cluster IP as
defined in topology) and the area ID for the interface/IP
localhost(config-router-ospf)#network 10.0.1.10 0.0.0.0 area 0.0.0.0
localhost(config-router-ospf)#network 10.0.1.20 0.0.0.0 area 0.0.0.0
--------- Redistribute kernel routes (this is only here as an example, please
see the dynamic routing book for more specific commands concerning
redistribution of routes)
localhost(config-router-ospf)#redistribute kernel
localhost(config-router-ospf)#exit
localhost(config)#exit
-------- Write configuration to disk
localhost#write memory
IU0 999 Configuration written to '/etc/gated.ami'
localhost#quit
```

Dynamic Routing on GWb

```
 --------- Launch the Dynamic Routing Module
[GWb]# expert
Enter expert password:

You are in expert mode now.

[Expert@GWb]# cligated
localhost>enable
localhost#configure terminal
--------- Enable OSPF and provide an OSPF router ID
localhost(config)#router ospf 1
localhost(config-router-ospf)#router-id 180.180.1.1
--------- Define interfaces/IP's on which OSPF runs (Use the cluster IP as
defined in topology) and the area ID for the interface/IP
localhost(config-router-ospf)#network 10.0.1.10 0.0.0.0 area 0.0.0.0
localhost(config-router-ospf)#network 10.0.0.3 0.0.0.0 area 0.0.0.0
--------- Redistribute kernel routes (this is only here as an example, please
see the dynamic routing book for more specific commands concerning
redistribution of routes)
localhost(config-router-ospf)#redistribute kernel
localhost(config-router-ospf)#exit
localhost(config)#exit
-------- Write configuration to disk
localhost#write memory
IU0 999 Configuration written to '/etc/gated.ami'
localhost#quit
```

Dynamic Routing on GWC

```
--------- Launch the Dynamic Routing Module
[GWc]# expert
Enter expert password:

You are in expert mode now.

[Expert@GWc]# cligated
localhost>enable
localhost#configure terminal
--------- Enable OSPF and provide an OSPF router ID
localhost(config)#router ospf 1
localhost(config-router-ospf)#router-id 190.190.1.1
--------- Define interfaces/IP's on which OSPF runs (Use the cluster IP as
defined in topology) and the area ID for the interface/IP
localhost(config-router-ospf)#network 10.0.1.20 0.0.0.0 area 0.0.0.0
localhost(config-router-ospf)#network 10.0.0.2 0.0.0.0 area 0.0.0.0
--------- Redistribute kernel routes (this is only here as an example, please
see the dynamic routing book for more specific commands concerning
redistribution of routes)
localhost(config-router-ospf)#redistribute kernel
localhost(config-router-ospf)#exit
localhost(config)#exit
-------- Write configuration to disk
localhost#write memory
IU0 999 Configuration written to '/etc/gated.ami'
localhost#quit
```

Configuring VTIs in a Gaia Environment

To learn how to configure VTIs in Gaia environments, see VPN Tunnel Interfaces in the *R80.10 Gaia Administration Guide* http://supportcontent.checkpoint.com/documentation_download?ID=TBD.

Note - For VTIs between Gaia gateways and Cisco GRE gateways: You must manually configure hello/dead packet intervals at 10/40 on the Gaia gateway, or at 30/120 on the peer gateway. If not, OSPF will not get into Full state.

Configuring Anti-Spoofing on VTIs

In SmartConsole:

1. In the **Gateways & Servers** view, edit a Check Point Security Gateway.
2. Go to the **Network Management** page.
3. Click **Get Interfaces** to read the interface information on the Security Gateway computer.
4. Select an interface, and click **Edit**.
5. In the **Topology** section of the **General** page, click **Modify**.
6. In the **IP Addresses behind peer Security Gateway that are within reach of this interface** section, select:
   - **Not Defined** to accept all traffic.
   - **Specific** to choose a particular network. The IP addresses in this network will be the only addresses accepted by this interface.
7. In the **Perform Anti-Spoofing based on interface topology** section, select **Don't check packets from** to ensure Anti-Spoofing checks do not take place for addresses from certain internal networks coming into the external interface. Define a network object that represents those internal networks with valid addresses, and from the drop-down list, select that network object.

   Objects selected in the **Don't check packets from** drop-down menu are disregarded by the Anti-Spoofing enforcement mechanism.
8. Under **Spoof Tracking** select **Log**, and click **OK**.

Configuring Unnumbered VTIs

Working with unnumbered interfaces eliminates the need to assign two IP addresses per interface (the local IP, and the remote IP Address), and the need to synchronize this information among the peers.

If the VPN Tunnel Interface is unnumbered, local and remote IP addresses are not configured. This interface is associated with a proxy interface from which the virtual interface inherits an IP address. Traffic initiated by the Security Gateway and routed through the virtual interface will have the physical interface's IP Address as the source IP.

To configure unnumbered VTIs for Gaia:

1. In Gaia WebUI, select **Interface Management > Network Interfaces**.
2. Click **Add > VPN Tunnel**.
3. In the **Add/Edit** window that opens, configure these parameters:
   - **VPN Tunnel ID** - an integer from 1 to 99, and Gaia automatically adds `vpnt` prefix to the Tunnel ID
   - **Remote Peer Name** - alpha-numeric Peer ID, as defined for the Remote Peer Name in the VPN community. You must define the two peers in the VPN community before you define the VTI.

- **VPN Tunnel Type** - select **Unnumbered**
- **Local Address** - leave empty for unnumbered VTI
- **Remote Address** - leave empty for unnumbered VTI
- **Physical Device** - the name of the local peer interface (the loopback interface can also be configured as the local peer interface)

Routing Multicast Packets Through VPN Tunnels

Multicast is used to transmit a single message to a select group of recipients. IP Multicasting applications send one copy of each datagram (IP packet) and address it to a group of computers that want to receive it. This technique addresses datagrams to a group of receivers (at the multicast address) rather than to a single receiver (at a unicast address). The network is responsible for forwarding the datagrams to only those networks that need to receive them. PIM is required for this feature and it is only available on SecurePlatform Pro and IPSO.

For more information on Multicasting, see *"Multicast Access Control"* in the *R80.10 Security Gateway Technical Administration Guide* http://supportcontent.checkpoint.com/documentation_download?ID=46524.

Multicast traffic can be encrypted and forwarded across VPN tunnels that were configured using VPN tunnel interfaces (virtual interfaces associated with the same physical interface). All participant Security Gateways, both on the sending and receiving ends, must have a virtual interface for each VPN tunnel and a multicast routing protocol must be enabled on all participant Security Gateways.

For more information on virtual interfaces, see Configuring a Virtual Interface Using the VPN Shell.

To enable multicast service on a Security Gateway functioning as a rendezvous point, add a rule to the security policy of that Security Gateway to allow only the specific multicast service to be accepted unencrypted, and to accept all other services only through the community. Corresponding access rules enabling multicast protocols and services should be created on all participating Security Gateways. For example:

Source	Destination	VPN	Service	Action	Track
Multicast_ Gateways	Multicast_ Gateways	Any Traffic	igmp pim	accept	log
Sample_Host	Multicast_Group_ Address	Sample_ Community	Multicast_ Service_Group	accept	log

Tunnel Management

In This Section:

Overview of Tunnel Management

The VPN tunnel transports data securely. You can manage the types of tunnels and the number of tunnels with these features:

- *Permanent Tunnels* - Keeps VPN tunnels active to allow real-time monitoring capabilities.
- *VPN Tunnel Sharing* - Provides greater interoperability and scalability between Security Gateways. It also controls the number of VPN tunnels created between peer Security Gateways.

See the status of all VPN tunnels in SmartView Monitor. For details see *Monitoring Tunnels* in the *Logging and Monitoring Administration Guide*.

Permanent Tunnels

As companies have become more dependent on VPNs for communication to other sites, uninterrupted connectivity has become more crucial than ever before. Therefore it is essential to make sure that the VPN tunnels are kept up and running. Permanent Tunnels are constantly kept active and as a result, make it easier to recognize malfunctions and connectivity problems. Administrators can monitor the two sides of a VPN tunnel and identify problems without delay.

Each VPN tunnel in the community may be set to be a Permanent Tunnel. Since Permanent Tunnels are constantly monitored, if the VPN tunnel is down, then a log, alert, or user defined action, can be issued. A VPN tunnel is monitored by periodically sending "tunnel test" packets. As long as responses to the packets are received the VPN tunnel is considered "up." If no response is received within a given time period, the VPN tunnel is considered "down." Permanent Tunnels can only be established between Check Point Security Gateways. The configuration of Permanent Tunnels takes place on the community level and:

- Can be specified for an entire community. This option sets every VPN tunnel in the community as permanent.
- Can be specified for a specific Security Gateway. Use this option to configure specific Security Gateways to have permanent tunnels.
- Can be specified for a single VPN tunnel. This feature allows configuring specific tunnels between specific Security Gateways as permanent.

Permanent Tunnels in a MEP Environment

In a *Multiple Entry Point* (MEP) environment, VPN tunnels that are active are rerouted from the predefined primary Security Gateway to the backup Security Gateway if the primary Security Gateway becomes unavailable. When a Permanent Tunnel is configured between Security Gateways in a MEP environment where RIM is enabled, the satellite Security Gateways see the center Security Gateways as "unified." As a result, the connection will not fail but will fail over to another center Security Gateway on a newly created permanent tunnel. For more information on MEP see Multiple Entry Point VPNs (see "Multiple Entry Point (MEP) VPNs" on page 114).

Tunnel Testing for Permanent Tunnels

Check Point uses a proprietary protocol to test if VPN tunnels are active, and supports any site-to-site VPN configuration. Tunnel testing requires two Security Gateways, and uses UDP port 18234. Check Point tunnel testing protocol does not support 3rd party Security Gateways.

Terminating Permanent Tunnels

Once a Permanent Tunnel is no longer required, the tunnel can be shut down. Permanent Tunnels are shut down by deselecting the configuration options to make them active and re-installing the policy.

Dead Peer Detection

In addition to Tunnel Testing, *Dead Peer Detection* (DPD) is a different method to test if VPN tunnels are active. *Dead Peer Detection* does support 3rd party Security Gateways and supports permanent tunnels with interoperable devices based on IKEv1/IKEv2 DPD (IKEv1 DPD is based on RFC 3706). It uses IPsec traffic patterns to minimize the number of messages required to confirm the availability of a peer.

The tunnel testing mechanism is the recommended keepalive mechanism for Check Point to Check Point VPN gateways because it is based on IPsec traffic and requires an IPsec established tunnel. DPD is based on IKE encryption keys only.

DPD has two modes:

- **DPD responder mode** - Requires an R77.10 or higher gateway managed by an R77 or higher management server.
- **Permanent tunnel mode based on DPD -** Requires an R77.10 or higher gateway managed by an R77.10 or higher management.

DPD Responder Mode

In this mode the Check Point gateway sends the IKEv1 DPD Vendor ID to peers from which the DPD Vendor ID was received.

To enable DPD Responder Mode:

1. Run on each gateway:
   ```
   ckp_regedit -a SOFTWARE/CheckPoint/VPN1 forceSendDPDPayload -n 1
   ```

2. Enable the keep_IKE_SAs property in GuiDBedit to prevent a problem where the Check Point gateway deletes IKE SAs:
   a) In **SmartConsole**, go to **Menu** > **Global Properties** > **Advanced** > **Advanced Configuration** > **VPN advanced properties** > **VPN IKE properties**.
   b) Change keep_IKE_SAs to **true**.

To disable DPD Responder Mode:

1. Run on each gateway:
   ```
   ckp_regedit -d SOFTWARE/CheckPoint/VPN1 forceSendDPDPayload
   ```

Note - Enable the keep_IKE_SAs property in GuiDBedit to prevent a problem where the Check Point gateway deletes IKE SAs. The DPD mechanism is based on IKE SA keys. In some situations, the Check Point gateway deletes IKE SAs and a peer, usually a 3rd Party gateway, sends DPD requests without response and concludes that the Check Point gateway is down. The peer can then delete the IKE and IPsec keys, which causes encrypted traffic from the Check Point gateway to be dropped by the remote peer.

Permanent Tunnel Mode Based on DPD

DPD can monitor remote peers with the permanent tunnel feature. All related behavior and configurations of permanent tunnels are supported.

To configure DPD for a permanent tunnel, the permanent tunnel must be in the VPN community. After you configure the permanent tunnel, configure Permanent Tunnel mode Based on DPD. There are different possibilities for permanent tunnel mode:

- **tunnel_test (default)** - The permanent tunnel is monitored by tunnel test (as in earlier versions). It works between Check Point gateways only. Keepalive packets are always sent.
- **dpd** - The active DPD mode. A peer receives DPD requests at regular intervals (10 seconds). DPD requests are only sent when there is no traffic from the peer.
- **passive** - The passive DPD mode. Peers do not send DPD requests to this peer. Tunnels with passive peers are monitored only if there is IPsec traffic and incoming DPD requests.

  Note: To use this mode for only some gateways, enable the `forceSendDPDPayload` registry key on Check Point remote peers.

To enable DPD monitoring:

On each VPN gateway in the VPN community, configure the tunnel_keepalive_method property, in GuiDBedit. This includes 3rd Party gateways. (You cannot configure different monitor mechanisms for the same gateway).

1. In GuiDBedit, go to **Network Objects** > **network_objects** > **<gateway>** > **VPN**.
2. For the **Value**, select a permanent tunnel mode.
3. Save.
4. Install policy on the gateways.

Optional Configuration

- **IKE Initiation Prevention** - By default, when a valid IKE SA is not available, a DPD request message triggers a new IKE negotiation. To prevent this behavior, set the property **dpd_allowed_to_init_ike** to **false**.

Edit the property in GuiDBedit under **Network Objects** > **network_objects** > **<gateway Name>** > **VPN**.

- **Delete IKE SAs for dead peer** - Based on RFC 3706, a VPN gateway has to delete IKE SAs from a dead peer. This functionality is enabled, by default.

  To disable this feature, set the **DPD_DONT_DEL_SA** environment variable to **0:**

  - To do this temporarily, run:

    ```
    cpstop
    export DPD_DONT_DEL_SA=0
    cpstart
    ```

  - To do this permanently:

    (i) Add this line to the `$CPDIR/tmp/.CPprofile.sh file`:

    DPD_DONT_DEL_SA=0 ; export DPD_DONT_DEL_SA

    (ii) Reboot

    Note: To re-enable the feature, remove the DPD_DONT_DEL_SA environment variable.

VPN Tunnel Sharing

For a VPN community, the VPN tunnel sharing configuration is set on the **Tunnel Management** page of the **Community Properties** window.

For a specific Security Gateway, the configuration is set on the **VPN Advanced** page of the Security Gateway properties window.

VPN Tunnel Sharing provides greater interoperability and scalability by controlling the number of VPN tunnels created between peer Security Gateways. Configuration of VPN Tunnel Sharing can be set on both the VPN community and Security Gateway object.

- **One VPN Tunnel per each pair of hosts** - A VPN tunnel is created for every session initiated between every pair of hosts.
- **One VPN Tunnel per subnet pair**- Once a VPN tunnel has been opened between two subnets, subsequent sessions between the same subnets will share the same VPN tunnel. This is the default setting and is compliant with the IPsec industry standard.
- **One VPN Tunnel per Security Gateway pair**- One VPN tunnel is created between peer Security Gateways and shared by all hosts behind each peer Security Gateway.

In case of a conflict between the tunnel properties of a VPN community and a Security Gateway object that is a member of that same community, the "stricter" setting is followed. For example, a Security Gateway that was set to **One VPN Tunnel per each pair of hosts** and a community that was set to **One VPN Tunnel per subnet pair**, would follow **One VPN Tunnel per each pair of hosts**.

Tunnel test is a proprietary Check Point protocol used to see if VPN tunnels are active. Tunnel testing requires two Security Gateways and uses UDP port 18234. Third party gateways do not support tunnel testing.

Configuring Tunnel Features

To configure Tunnel Management options:

1. In SmartConsole, click **Object Explorer** (Ctrl+E)
2. Click **New** > **VPN Community** and choose **Star Community** or **Meshed community**.
3. Click **Tunnel Management**. and configure the tunnel settings:
   - Permanent Tunnels (on page 71)
   - Tracking Options (on page 76)
   - VPN Tunnel Sharing (on page 74)

Permanent Tunnels

In the **Star Community** or **Meshed community** object, on the **Tunnel Management** page, select **Set Permanent Tunnels.** These are the options:

- **On all tunnels in the community**
- **On all tunnels of specific Security Gateways**
- **On specific tunnels in the community**

To configure all tunnels as permanent, select **On all tunnels in the community**. Clear this option to terminate all Permanent Tunnels in the community.

To configure on all tunnels of specific Security Gateways:

1. Select **On all tunnels of specific gateways** and click **Select Gateways**.

   The **Select Gateway** window is displayed.

   To terminate Permanent Tunnels connected to a specific Security Gateway, highlight the Security Gateway and click **Remove**.
2. To configure the Tracking options for a specific Security Gateway, highlight a Security Gateway and click **Gateway Tunnel Properties**.

To configure on specific tunnels in the community:

1. Select **On specific tunnels in the community** and click **Select Permanent Tunnels**.

   The **Select Permanent Tunnels** window opens.
2. Double click in the white cell that intersects the Security Gateways where a permanent tunnel is required.

   The **Tunnel Properties** window is displayed.
3. Click **Set these tunnels to be permanent tunnels**.

   To terminate the Permanent Tunnel between these two Security Gateways, clear **Set these tunnels to be permanent tunnels**.
4. Click **OK**.

Advanced Permanent Tunnel Configuration

In SmartConsole:

1. Click **Menu** > **Global Properties**.

   The **Global Properties** window shows.

2. Select **Advanced** > **Configure**.

   The **Advanced configuration** window shows.
3. Click **VPN Advanced Properties > Tunnel Management** to see the five attributes that may be configured to customize the amount of tunnel tests sent and the intervals in which they are sent:
   - **life_sign_timeout** - Set the amount of time the tunnel test runs without a response before the peer host is declared 'down.'
   - **life_sign_transmitter_interval** - Set the time between tunnel tests.
   - **life_sign_retransmissions_count** - When a tunnel test does not receive a reply, another test is resent to confirm that the peer is 'down.' The Life Sign Retransmission Count is set to how many times the tunnel test is resent without receiving a response.
   - **life_sign_retransmissions_interval** - Set the time between the tunnel tests that are resent after it does not receive a response from the peer.
   - **cluster_status_polling_interval** - (Relevant for HA Clusters only) - Set the time between tunnel tests between a primary Security Gateway and a backup Security Gateway. The tunnel test is sent by the backup Security Gateway. When there is no reply, the backup Security Gateway will become active.

Tracking Options

You can configure alerts to stay updated on the status of permanent VPN tunnels.

To configure logs and alerts for VPN tunnel status:

1. In the properties of the VPN Community, open the **Tunnel Management** page.
2. In **Tunnel down track**, select the alert when a tunnel is down.
3. In **Tunnel up track**, select the alert when a tunnel is up.

   The alerts are configured for the tunnels that are defined as permanent, based on the settings on the page.

See status of all VPN tunnels in SmartView Monitor.

To open SmartView Monitor:

1. Open SmartConsole > Logs & Monitor.
2. Click **New Tab**.
3. Click **Tunnel & User Monitoring**.

For more details, see *Monitoring Tunnels* in the *R80.10 Logging and Monitoring Administration Guide*.

Link Selection

In This Section:

Link Selection Overview

Link Selection is a method to define which interface is used for incoming and outgoing VPN traffic as well as the best possible path for the traffic. With the Link Selection mechanisms, the administrator can choose which IP addresses are used for VPN traffic on each Security Gateway.

Link Selection has many configuration options to enable you to control VPN traffic. These options include:

- Use probing to choose links according to their availability.
- Use Load Sharing for Link Selection to distribute VPN traffic over available links.
- Use Service Based Link Selection to control bandwidth use.

Configuration settings for remote access clients can be configured together or separately from the Site-to-Site configuration.

Configuring IP Selection by Remote Peer

There are several methods that can determine how remote peers resolve the IP address of the local Security Gateway. These settings are configured in **Security Gateway Properties > IPsec VPN > Link Selection**. Remote peers can connect to the local Security Gateway with these settings.

Always Use This IP Address:

Configure a certain IP address that is always used. The options are:

- **Main address** - The VPN tunnel is created with the Security Gateway main IP address, specified in the **IP Address** field on the **General Properties** page of the Security Gateway.
- **Selected address from topology table** - The VPN tunnel is created with the Security Gateway using a selected IP address chosen from the drop down menu that lists the IP addresses configured in the **Topology** page of the Security Gateway.

- **Statically NATed IP** - The VPN tunnel is created using a NATed IP address. This address is not required to be listed in the topology tab.

Calculate IP Based on Network Topology:

The VPN tunnel is created using an IP address of an external interface on the local gateway. The remote peer selects the first IP address on the list of external interfaces. This method is best when the main IP address of the local gateway is unknown and the external interfaces change frequently.

Use DNS Resolving:

This method is required for Dynamically Assigned IP (DAIP) Security Gateways. A VPN tunnel to a DAIP Security Gateway can only be initiated using DNS resolving since the IP address of the DAIP Security Gateway cannot be known in advance. If using this method for a non-DAIP Security Gateway, the IP address must be defined in the **Topology** tab. Without DNS resolving, a DAIP Security Gateway can only initiate the first connection between two peers. The second connection can be initiated by the peer Security Gateway as long as the IP address of the DAIP Security Gateway has not changed. The options are:

- **Full hostname** - Enter the full Fully Qualified Domain Name (FQDN). The DNS host name that is used is "Security Gateway_name.domain_name." For example, if the object name is "john" and the domain name is "smith.com" then the FQDN will be "john.smith.com."
- **Security Gateways name and domain name (specified in global properties)** - The Security Gateway name is derived from the **General Properties** page of the Security Gateway and the domain name is derived from the Global Properties page.

Use Probing Redundancy Mode:

When more than one IP address is available on a Security Gateway for VPN, Link Selection may employ the RDP probing method to determine which link will be used. The RDP probing method is implemented using a proprietary protocol that uses UDP port 259. This protocol is proprietary to Check Point and works only between Check Point entities. (Note that it does not comply with RDP as specified in RFC 908/1151). IP addresses you do not want to be probed (i.e., internal IP addresses) may be removed from the list of IP's to be probed. Once a Security Gateway maps the links' availability, a link selection per connection can be made according to the following redundancy modes:

- **High Availability** (default setting)

  In High Availability mode the VPN tunnel uses the first IP address to respond, or the primary IP address if a primary IP is configured and active. If the chosen IP address stops responding, the connection fails over to another responding IP address. If a primary IP address is configured, the VPN tunnel will stay on the backup IP address until the primary one becomes available again.

  Note that if **one time probing** is configured, the VPN tunnel will stay on the first chosen IP address until the next time policy is installed. See **ongoing probing and onetime probing** methods below.

- **Load Sharing**

  In Load Sharing mode the encrypted traffic is distributed among all available links. Every new connection ready for encryption uses the next available link in a round robin manner. When a link becomes unavailable, all of its connections are distributed among the other available links. A link's availability is determined using RDP probing.

The peer Security Gateway that responds to the connection will route the reply traffic through the same route that it was received on, as long as that link is available.

Although the VPN tunnel traffic can be routed through multiple links in Load Sharing mode, only one VPN tunnel is generated. IKE sessions are arbitrarily routed though one of the available links.

Load Sharing is supported on Security Gateways of version R71 and higher. If a Security Gateway of version R71 or higher is configured to use the Load Sharing redundancy mode, Security Gateways of versions before R71 will use the High Availability redundancy mode when routing traffic to the R71 or higher Security Gateways.

Load Sharing is supported on all platforms for incoming traffic. For outgoing traffic, VPN traffic between peers with Load Sharing configuration is not accelerated by IPSO acceleration devices. Load Sharing is not supported on UTM-1 Edge devices.

Probing Settings:

Additional settings related to probing are set in **Link Selection > IP Selection by Remote Peer > Use probing > Configure > Probing Settings**

- **Probe all addresses defined in the topology tab** - choose to include all addresses defined in the topology tab for the Security Gateway in the probing
- **Probe the following addresses** - Specify the addresses that you want to include in the probing.
- **Primary address** -Optionally, to choose a primary address, select the check box and choose one of the included IP addresses from the drop down menu as the Primary Address. A primary IP address is only used with the High Availability probing mode. If Load Sharing is configured, the primary address is ignored. Enabling a primary IP address has no influence on the IP selected for outgoing VPN traffic. If the remote Security Gateway connects to a peer Security Gateway that has a primary IP address defined, then the remote Security Gateway will connect to the primary address (if active) regardless of network speed (latency) or route metrics.
- **Use probing method**

  Choose one of the following probing methods.
  - **Using ongoing probing** (default setting) - When a session is initiated, all possible destination IP addresses continuously receive RDP packets. The RDP probing is activated when a connection is opened and continues as a background process.
  - **Using one time probing** - When a session is initiated, all possible destination IP addresses receive an RDP session to test the route. These results are used until the next time that a policy is installed.

Note - UDP RDP packets are not encrypted. The RDP mechanism only tests connectivity.

Last Known Available Peer IP Address

The IP address used by a Security Gateway during a successful IKE negotiation with a peer Security Gateway, is used by the peer Security Gateway as the destination IP address for the next IPsec traffic and IKE negotiations that it initiates. This is only the case when the Link Selection configuration does not use probing.

Configuring Outgoing Route Selection

For outbound traffic, there are different methods that can be used to determine which path to use when connecting with a remote peer. These settings are configured in **Security Gateway Properties > IPsec VPN > Link Selection**.

When Initiating a Tunnel

- **Operating system routing table** (default setting) - Using this method, the routing table is consulted for the available route with the lowest metric and best match for the VPN tunnel traffic.
- **Route based probing** - This method also consults the routing table for an available route with the lowest metric and best match. However, before a route is chosen, it is tested for availability using RDP probing. The Security Gateway then selects the best match (highest prefix length) active route with the lowest *metric*. This method is recommended when there is more than one external interface.

  If you selected the **IP Selection by Remote Peer** setting of **Use probing** with **Load Sharing**, it also affects **Route based probing** link selection. In this case, **Route based probing** distributes the outgoing encrypted traffic among all available links. All possible links to the peer Security Gateway are derived from the routing table and the link's availability is tested using RDP probing. Every new connection ready for encryption uses the next available link in a round robin manner.

  Route based probing enables the use of **On Demand Links (ODL)**, which are triggered upon failure of all primary links. You can run a script to activate an **On Demand Link** when all other links with higher priorities become unavailable. For more information, see On Demand Links ("On Demand Links (ODL)" on page 93).

  For IKE and RDP sessions, Route based probing uses the same IP address and interface for responding traffic.

  Route based probing is supported on the SecurePlatform, Gaia, Linux, and IPSO platforms. VPN traffic between peers with Load Sharing probing mode and Route Based probing configuration will not be accelerated by IPSO acceleration devices.

  Note:

  The High Availability mechanism is based on:

  - `resolver_session_interval (30)` - Defines for how many seconds the remote peer status (up or down) stays valid
  - `resolver_ttl (10)` - Defines how many seconds the gateway waits before it decides that a remote peer is down

  Some network protocols (for example, TCP) might timeout in the time between link failure and the next attempt to resolve. Administrators can decrease these default values. Note that high resolution frequency can overload the gateway. This configuration also changes the default resolution timeouts for the MEP mechanism.

  For L2 links, there must be routes to the peer's encryption domains through the local L2 interface device.

When Responding to a Remotely Initiated Tunnel

When responding to a remotely initiated tunnel, there are two options for selecting the interface and next hop that are used. *These settings are only relevant for IKE and RDP sessions.*

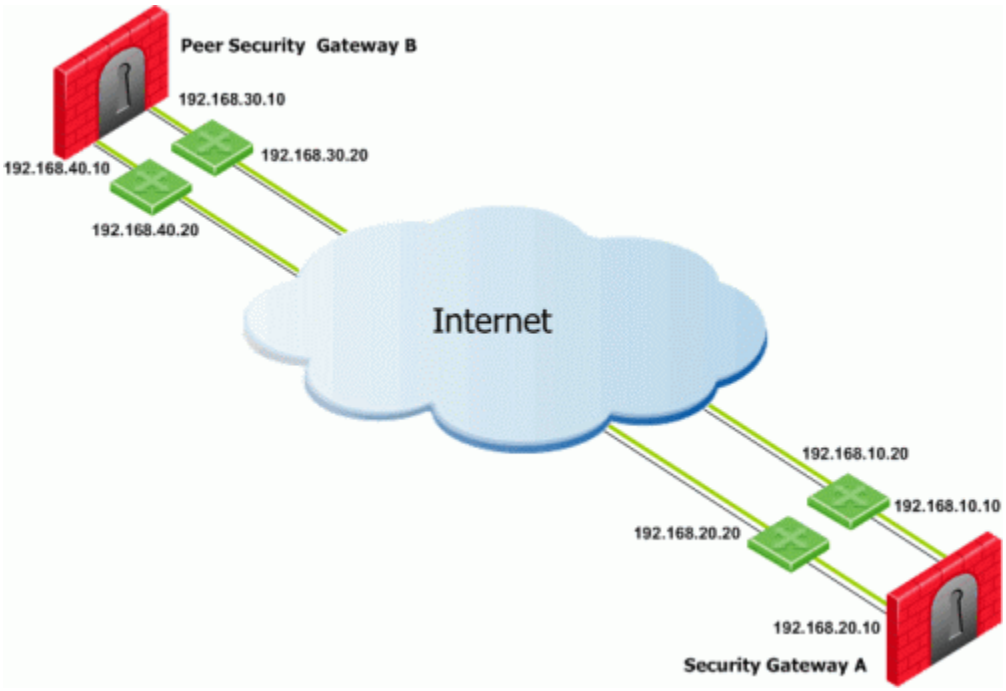
These settings are configured in **Link Selection > Outgoing Route Selection > Setup > Link Selection - Responding Traffic** window.

- **Use outgoing traffic configuration** - Select this option to choose an interface using the same method selected in the **Outgoing Route Selection** section of the **Link Selection** page.
- **Reply from the same interface** - This option sends the returning traffic through the same interface and next hop IP that it arrived in.

Note - When Route Based Probing is enabled, **Reply from the same interface** is the selected method and cannot be changed.

Using Route Based Probing

The local Security Gateway, using RDP probing, considers all possible routes between itself and the remote peer Security Gateway. The Security Gateway then decides on the most effective route between the two Security Gateways:



In this scenario, Security Gateway A has two external interfaces, 192.168.10.10 and 192.168.20.10. Peer Security Gateway B also has two external interfaces: 192.168.30.10 and 192.168.40.10.

For Security Gateway A, the routing table reads:

Destination	Netmask	Next hop	Metric
192.168.40.10	255.255.255.0	192.168.10.20	1
192.168.40.10	255.255.255.0	192.168.20.20	2
192.168.30.10	255.255.255.0	192.168.10.20	3
192.168.30.10	255.255.255.0	192.168.20.20	4

For Security Gateway B, the routing table reads:

Destination	Netmask	Next hop	Metric
192.168.20.10	255.255.255.0	192.168.40.20	1
192.168.20.10	255.255.255.0	192.168.30.20	2
192.168.10.10	255.255.255.0	192.168.40.20	3
192.168.10.10	255.255.255.0	192.168.30.20	4

If all routes for outgoing traffic from Security Gateway A are available, the route from 192.168.10.10 to 192.168.40.10 has the lowest metric (highest priority) and is therefore the preferred route.

Configuring Source IP Address Settings

The source IP address used for outgoing packets can be configured for sessions initiated by the Security Gateway. These settings are configured in **Security Gateway Properties > IPsec VPN > Link Selection > Outgoing Route Selection > Source IP address settings**.

When initiating a VPN tunnel, set the source IP address using one of the following:

- **Automatic (derived from the method of IP selection by remote peer)** - The source IP address of outgoing traffic is derived from the method selected in the **IP Selection by Remote Peer** section.
  - If **Main address** or **Selected address from topology table** are chosen in the **IP Selection by Remote Peer** section, then the source IP when initiating a VPN tunnel is the IP specified for that method.
  - If **Calculate IP based on network topology**, **Statically NATed IP, Use DNS resolving** or **Use probing** is chosen in the **IP Selection by Remote Peer** section, then the source IP when initiating a VPN tunnel is the IP address of the chosen outgoing interface.
- **Manual**:
  - **Main IP address -** The source IP is derived from the **General Properties** page of the Security Gateway.
  - **Selected address from topology table -** The selected IP from the drop down menu becomes the source IP.
  - **IP address of chosen interface -** The source IP is the same IP of the interface where the traffic is being routed through.

These settings are relevant for RDP and IKE sessions. When responding to an IKE session, use the **reply_from_same_IP (default: true)** attribute to follow the settings in the **Source IP address settings** window or to respond from the same IP.

Note - When Route Based Probing is enabled, **reply_from_same_IP** will be seen as **true.**

Outgoing Link Tracking

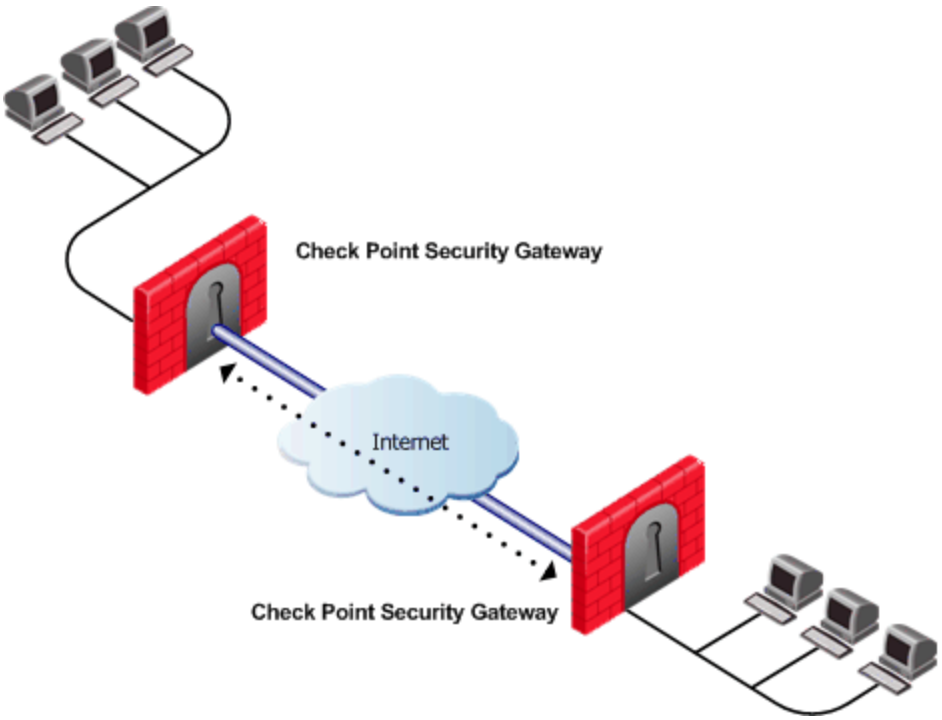
When **Outgoing link tracking** is activated on the local Security Gateway, the Security Gateway sends a log for every new resolving decision performed with one of its remote VPN peers. If **Use Probing** is configured on the local Security Gateway for Remote Peer resolving, or if Route Based Probing is activated on the local Security Gateway, log entries are also created for all resolving changes. For example, if a link in use becomes unavailable and a new available link is chosen, a log entry is issued.

Link Selection Scenarios

Link Selection can be used in many environments. This section describes various scenarios and how Link Selection should be configured in each scenario.

Gateway with a Single External Interface

This is the simplest scenario, where the local Security Gateway has a single external interface for VPN:



How do peer Security Gateways select an IP address on the local Security Gateway for VPN traffic?

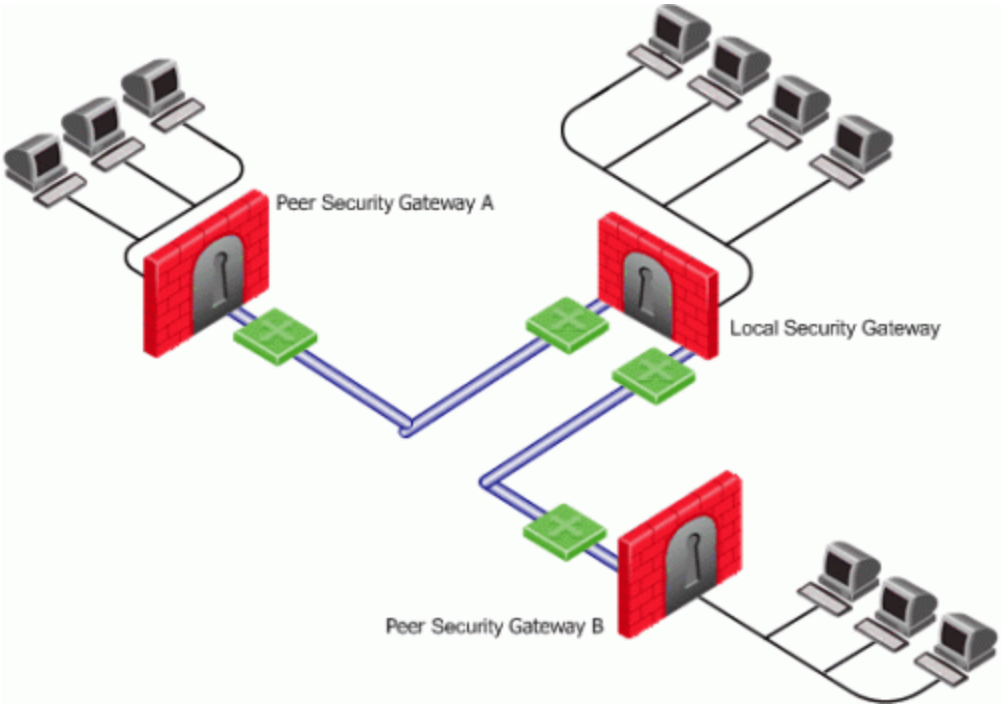
Since there is only one interface available for VPN, to determine how remote peers determine the IP address of the local Security Gateway, select the following from the **IP Selection by Remote Peer** section of the Link Selection page:

- Select **Main address** or choose an IP address from the **Selected address from topology table** drop down menu.
- If the IP address is located behind a static NAT device, select **Statically NATed IP**.

Gateway with Several IP Addresses Used by Different Parties

In this scenario, the local Security Gateway has a point-to-point connection from two different

interfaces. Each interface is used by a different remote party:

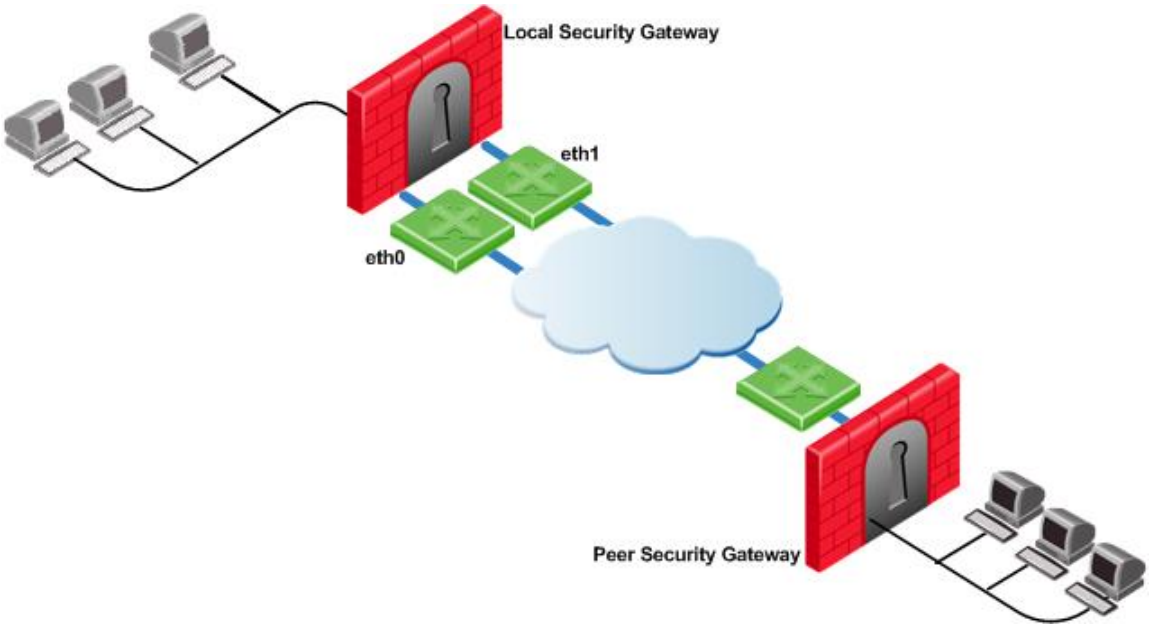


The local Security Gateway has two IP addresses used for VPN. One interface is used for VPN with a peer Security Gateway A and one interface for peer Security Gateway B.

To determine how peer Security Gateways discover the IP address of the local Security Gateway, enable **one-time probing** with **High Availability** redundancy mode. Since only one IP is available for each peer Security Gateway, probing only has to take place one time.

Gateway with an Interface Behind a Static NAT Device

In this scenario, the local Security Gateway has two external interfaces available for VPN. The address of interface eth0 is being translated using a NAT device:



To determine how peer Security Gateways discover the IP address of the local Security Gateway, use **ongoing probing** with **High Availability** redundancy mode. In order for the Static NAT IP

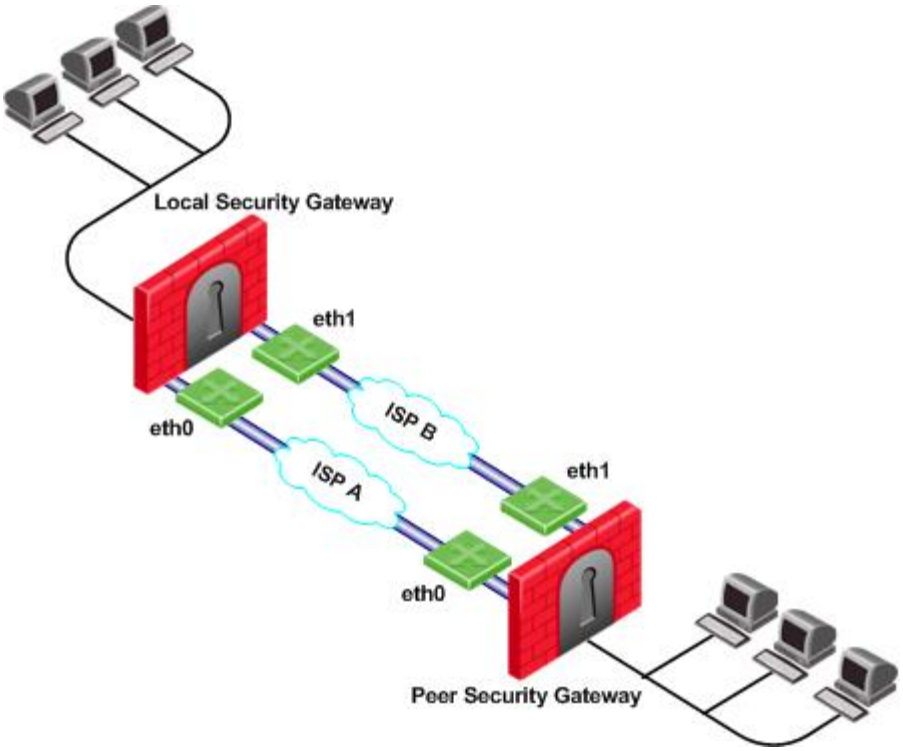
address to be probed, it must be added to the **Probe the following addresses** list in the **Probing Settings** window.

Utilizing Load Sharing

Depending on your configuration, there are many ways to use Load Sharing to distribute VPN traffic among available links between the local and peer Security Gateways.

Load Sharing with Multiple External Interfaces on Each End

In the following scenario, the local and peer Security Gateways each have two external interfaces available for VPN traffic.

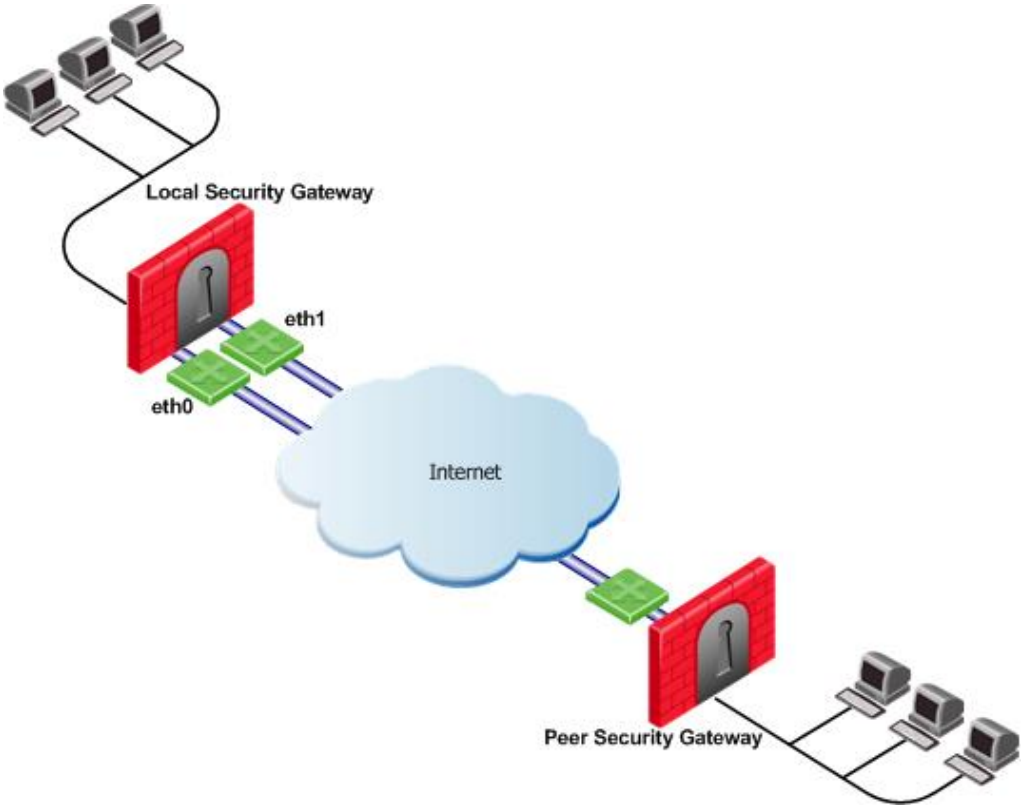


To utilize both external interfaces by distributing VPN traffic among all available links, use the Probing redundancy mode of **Load Sharing** on both Security Gateways. You can also specify that only certain external interfaces should be probed by putting only those interfaces in the **Probe the following addresses** list in the **Probing Settings** window. If one link goes down, traffic will automatically be rerouted through the other link.

To enable this configuration, make sure that your routing table allows packet flow back and forth between both eth0 interfaces and packet flow back and forth between both eth1 interfaces. Then Link Selection can reroute the VPN traffic between these available links.

Load Sharing with Multiple External Interfaces on One End

In the following scenario, the local Security Gateway has two external interfaces available for VPN traffic. The peer Security Gateway has one external interface for VPN traffic.



To utilize both external interfaces and distribute VPN traffic between the available links, use the Probing redundancy mode of **Load Sharing** on the local Security Gateway. Then the peer Security Gateway will distribute its outgoing VPN traffic between interfaces eth0 and eth1 of the local Security Gateway.

If the default, **Operating system routing table**, setting in the **Outgoing Route Selection** section is selected, the local Security Gateway will only use one of its local interfaces for outgoing VPN traffic; the route with the lowest metric and best match to reach the single IP address of the peer Security Gateway, according to the routing table.

If you want to distribute the outgoing VPN traffic on both outbound links from the local Security Gateway as well, select **Route Based Probing** in the Outgoing Route Selection on the Link Selection page of the local Security Gateway.

Service Based Link Selection

Service Based Link Selection enables administrators to control outgoing VPN traffic and bandwidth use by assigning a service or a group of services to a specific interface for outgoing VPN routing decisions. The encrypted traffic of an outgoing connection is routed through the configured interface according to the traffic's service. The links to the peer Security Gateway are derived from the routing table and the link's availability is tested using RDP probing.

If all links through the interface assigned to a specific service stop responding to RDP probing, a link failover will occur by default, as in any other probing mode. When a link through the assigned interface is restored, new outgoing connections are assigned to it, while existing connections are maintained over the backup link until they are completed.

It is possible to configure the traffic of a specific service not to fail over. In this case, traffic of the configured service will only be routed through interfaces assigned to this service, even if these interfaces stop responding to RDP.

If the same service is assigned to more than one interface, this service's traffic is distributed between the configured interfaces. Every new outgoing encrypted connection uses the next available link in a round robin manner.

All traffic from services that are not assigned to a specific interface is distributed among the remaining interfaces. If all links through these interfaces are down, the traffic is distributed among the interfaces that are configured for specific services.

Service Based Link Selection configuration requires enabling the following features:

- IP Selection by Remote Peer – Load Sharing probing mode
- Outgoing Route Selection – Route based probing
- Service Based Link Selection configuration file on the management server

Service Based Link Selection is supported on Security Gateways of version R71 and higher. It is supported on the SecurePlatform, Gaia, Linux, and IPSO platforms. VPN traffic between peers with Service Based Link Selection configuration is not accelerated by IPSO acceleration devices. Service Based Link Selection is not supported on UTM-1 Edge devices.

Configuring Service Based Link Selection

To configure Service Based Link Selection:

1. In the **Link Selection** page, in the IP Selection by Remote Peer section, select:
   - **Use probing. Redundancy mode**
   - **Load Sharing**
2. In the Outgoing Route Selection section, select **Route based probing**.

Edit the Service Based Link Selection configuration in the `$FWDIR/conf/vpn_service_based_routing.conf` configuration file on the management server.

Fill in each line in the configuration file to specify the target Security Gateway, the interface for outgoing routing, and the service (or services group) to route through this interface. Use the names defined in the SmartConsole GUI. Fill in all of the details for each Security Gateway on which you want to configure Service Based Link Selection.

The fields are:

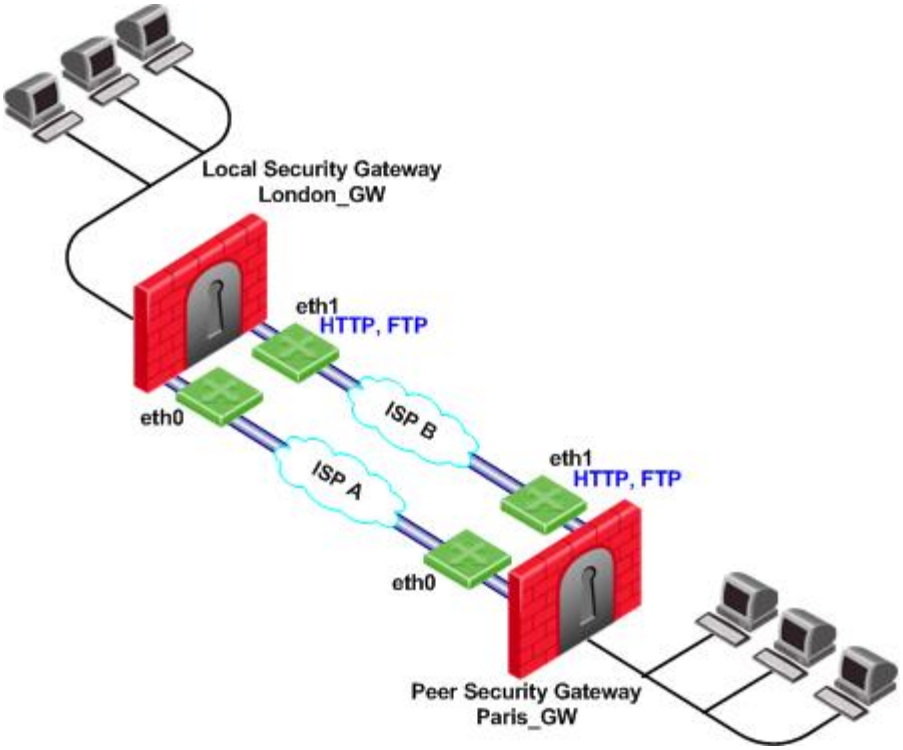
- **Gateway** – Security Gateway name (the name of the VPN Security Gateway or the cluster)
- **Interface** – Interface name
- **Service** – Service or services group name
- **dont_failover** – (Optional) If this string is present, traffic of the configured service will only be routed through interfaces configured for this service and will not fail over to another interface.

Service Based Link Selection Scenarios

The following scenarios provide examples of how Service Based Link Selection can be utilized.

Service Based Link Selection with Two Interfaces on Each End

In the scenario below, the local and peer Security Gateways each have two external interfaces for VPN traffic.



In this example, interface eth1 of both Security Gateways is dedicated to HTTP and FTP traffic. All other traffic is routed to interface eth0.

If the available link through eth1 stops responding to RDP probing, HTTP and FTP traffic will fail over to eth0. It is possible to specify that HTTP and FTP traffic should only be routed through eth1 even if the link through eth1 stops responding. Specify this by including the dont_failover flag when editing the Service Based Link Selection configuration file.

All other traffic that is not HTTP or FTP will be routed through eth0. If the link through eth0 stops responding to RDP probing, all traffic will be routed through eth1.

The Service Based Link Selection configuration file for this environment should appear as follows:

Gateway	Interface	Service	[dont_failover]
London_GW	eth1	http	
London_GW	eth1	ftp	
Paris_GW	eth1	http	
Paris_GW	eth1	ftp	

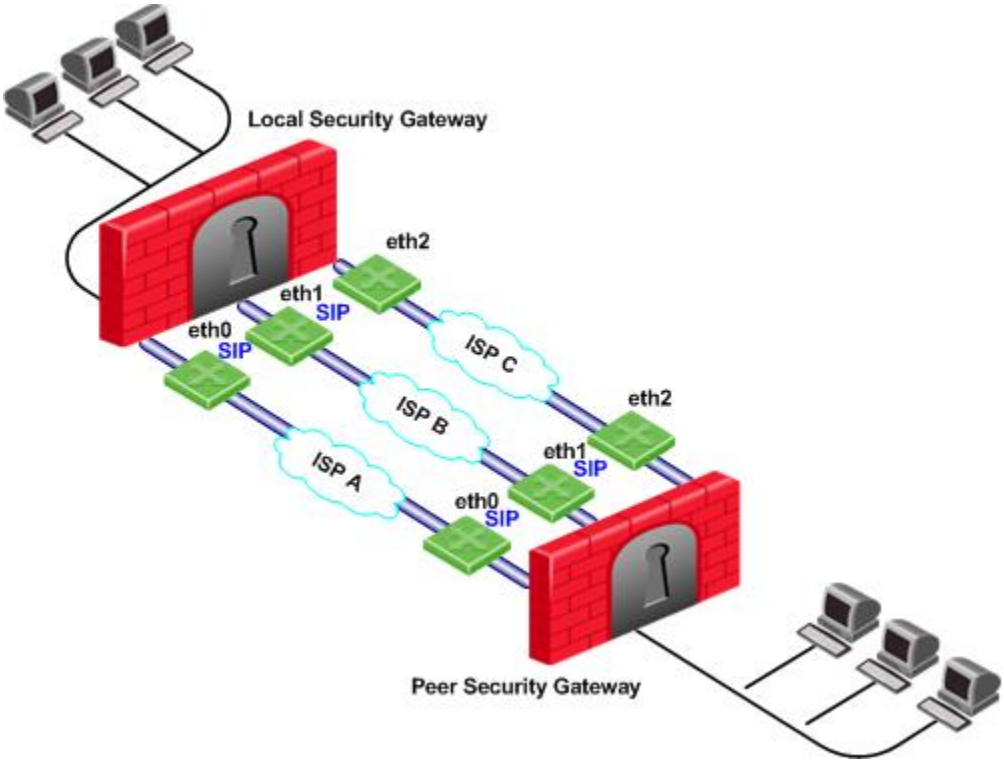
Alternatively, in SmartConsole, you can create a Services Group that includes HTTP and FTP services. In the example below, this group is called **http_ftp_grp**. Using this group, the Service Based Link Selection configuration file for this environment should appear as follows:

Gateway	Interface	Service	[dont_failover]
London_GW	eth1	http_ftp_grp	

Paris_GW	eth1	http_ftp_grp	

Service Based Link Selection with Multiple Interfaces on Each End

In the following scenario, the local and peer Security Gateways each have three external interfaces available for VPN.

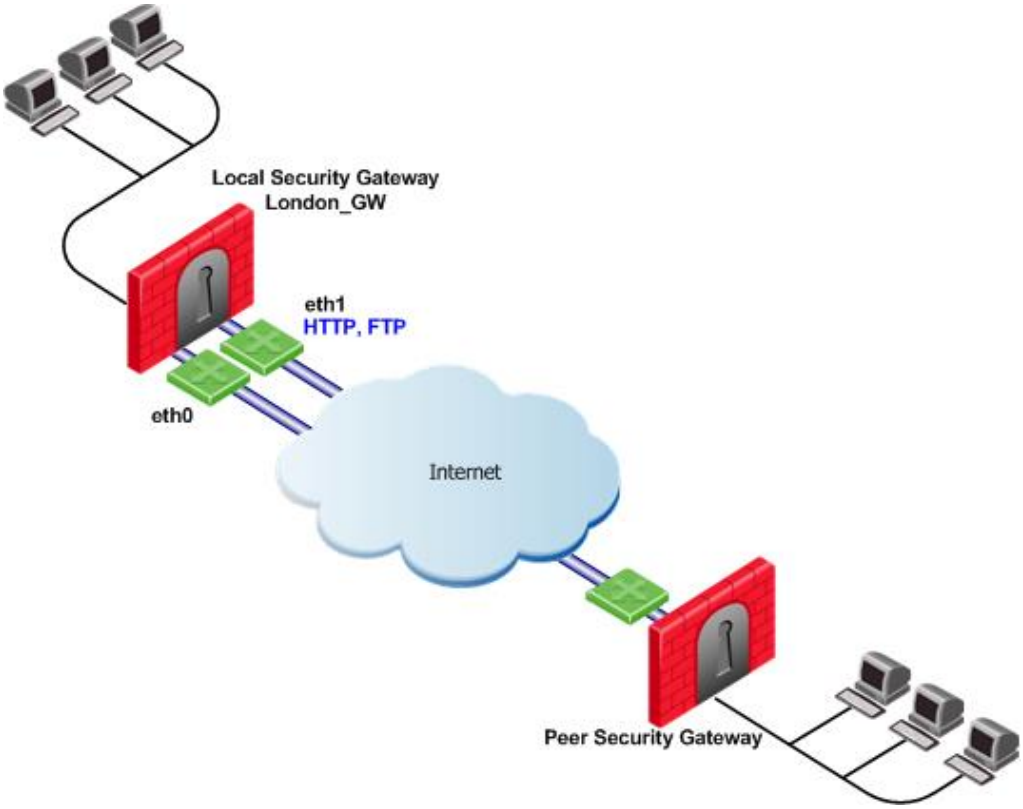


To utilize all three external interfaces and distribute the VPN traffic among the available links, Link Selection Load Sharing and Route based probing should be enabled. To control your bandwidth use, dedicate one or more links to a specific service or services using Service Based Link Selection. In this scenario, interfaces eth0 and eth1 of both Security Gateways are dedicated to SIP traffic. SIP traffic is distributed between eth0 and eth1. All other traffic is routed through eth2.

If either the link through eth0 or the link through eth1 stops responding to RDP probing, SIP traffic will fail over to the other SIP interface. If the link through eth2 stops responding to RDP probing, all traffic will be routed though eth0 or eth1.

Service Based Link Selection with Two Interfaces on One End

In the following scenario, the local Security Gateway has two external interfaces available for VPN traffic. The peer Security Gateway has a single external interface for VPN traffic.



To utilize all external interfaces and distribute the VPN traffic among the available links, Link Selection Load Sharing and Route based probing should be enabled on the local Security Gateway, **London_GW**. To control your bandwidth use, dedicate interface eth1 of the local Security Gateway to HTTP and FTP traffic using Service Based Link Selection. The local Security Gateway will route outgoing HTTP and FTP connections through interface eth1. All other traffic, not HTTP or FTP, will be routed through eth0.

In this scenario, HTTP and FTP traffic should not fail over. HTTP and FTP traffic should only be routed through interface eth1, even if the link through interface eth1 stops responding to RDP probing. This is configured by specifying the `dont_failover` flag.

The Service Based Link Selection configuration file for this environment should appear as follows:

Gateway	Interface	Service	[dont_failover]
London_GW	eth1	http	dont_failover
London_GW	eth1	ftp	dont_failover

Since the Service Based Link Selection configuration is only relevant for outgoing traffic of the local Security Gateway, the peer Security Gateway can send HTTP and FTP traffic to either interface of the local Security Gateway. The outgoing VPN traffic of the peer Security Gateway is distributed between interfaces eth0 and eth1 of the local Security Gateway.

Trusted Links

Trusted Links allows you to set an interface as "trusted" for VPN traffic so that traffic sent on that link will not be encrypted. You may want to set up a trusted link if you are confident that the link is already encrypted and secure and you do not need a second encryption.

If you configure an interface as trusted, traffic routed through that interface will be sent unencrypted, while traffic sent through other interfaces will still be encrypted.

Trusted interfaces should be configured symmetrically on the local and peer Security Gateways. If only one side of the link is configured as trusted for VPN traffic, clear traffic received by a non-trusted interface will be dropped by the peer Security Gateway.

If you have configured a specific link as trusted for VPN traffic and you are using probing, the probing method considers all links, including the trusted link, when choosing a link for a connection.

The probing method chooses the link according to these criteria:

- The configured redundancy mode, High Availability or Load Sharing
- If Service Based Link Selection is configured.

If the trusted link is chosen for a connection, the traffic is not encrypted. If another, non-trusted, link is chosen, the traffic is encrypted.

In an MEP configuration, trusted links are only supported for connections initiated by a peer Security Gateway to a MEP Security Gateway. Unencrypted VPN connections routed through a trusted interface and initiated by a MEP Security Gateway may be dropped by the peer Security Gateway.

Trusted links are not supported in Traditional mode. In Traditional mode, trusted link settings are ignored and VPN traffic is always encrypted.

Trusted links are supported on Security Gateways of version R71 and higher.

Note - Trusted links are not supported by IPSO acceleration devices. IPSO acceleration devices ignore trusted links settings and will encrypt traffic routed through these links.

Configuring Trusted Links

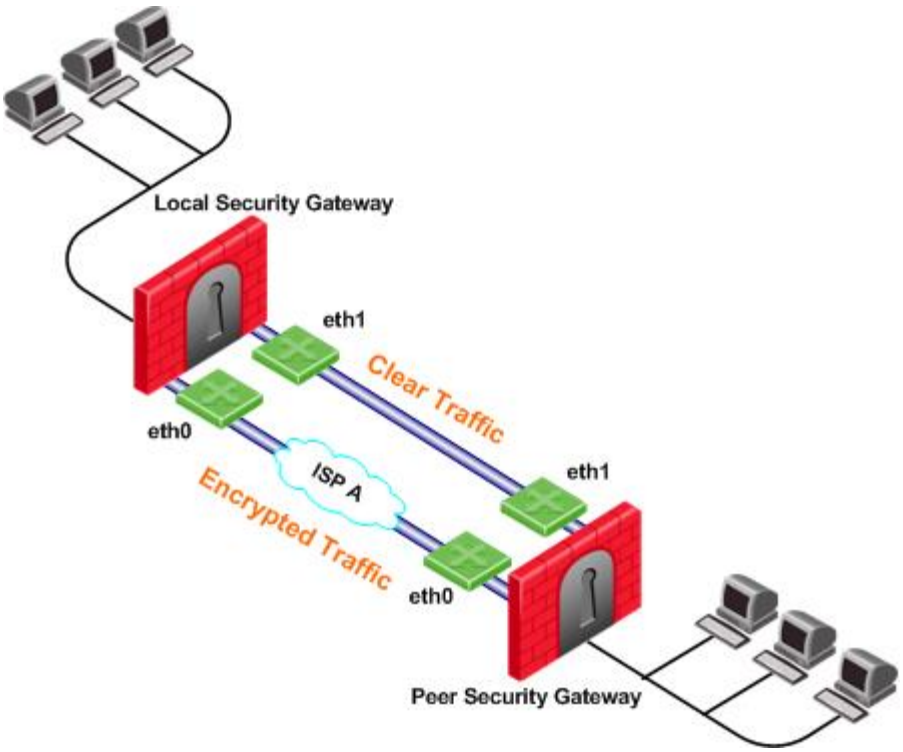
Use GuiDBedit, the Check Point Database Tool to configure Trusted Links.

To configure a trusted link:

1. In GuiDBedit go to **Network objects > network_objects**.
2. Select the Security Gateway that you want to edit.
3. Search for the interface that you want to configure as trusted from within the interfaces set. The interface name appears in the `officialname` attribute
4. Within the trusted interface set, change the value of the `vpn_trusted` attribute to `true` (default value: false).
5. Configure trusted interfaces symmetrically on the peer Security Gateways. If only one side of the link is configured as trusted for VPN traffic, clear traffic received by a non-trusted interface will be dropped by the peer Security Gateway.
6. Save changes.

Trusted Links Scenarios

In the following scenario, both the local and peer Security Gateways have two external interfaces available for VPN traffic. Interface eth1 on both Security Gateways has been configured as a trusted interface. Therefore traffic sent from eth1 of the local Security Gateway will be sent unencrypted and will be accepted by interface eth1 of the peer Security Gateway, and vice versa.

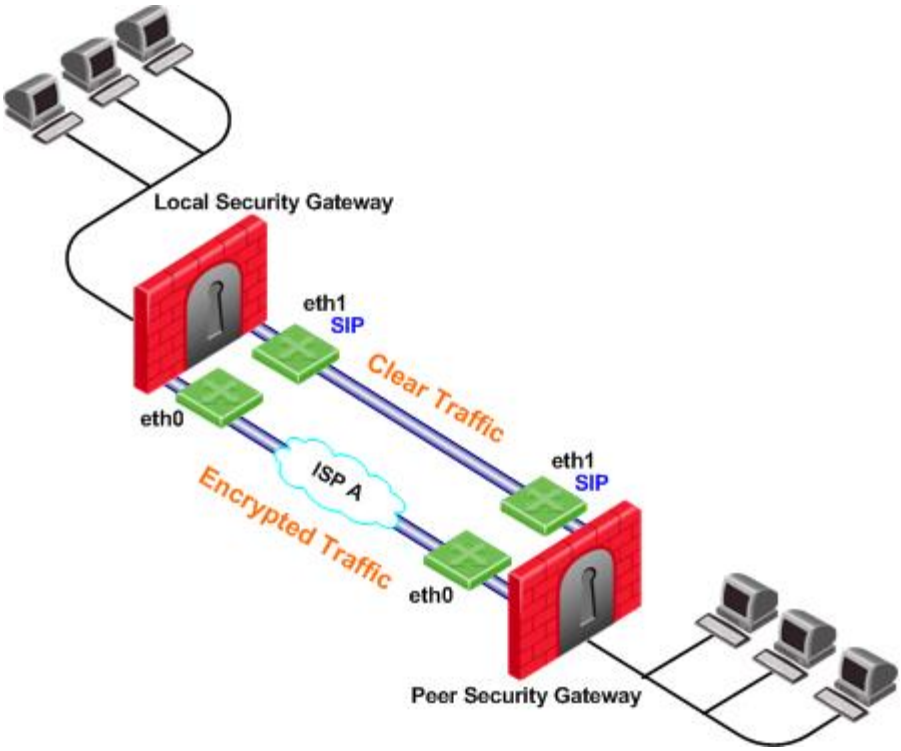


If the probing redundancy mode is High Availability and the trusted link is configured as the **Primary IP address**, the trusted link will be used for VPN traffic. If the trusted link stops responding to RDP probing, the link through Interface eth0 will be used for VPN traffic and traffic will be encrypted.

If the probing redundancy mode is Load Sharing, the VPN traffic will be distributed between the available links. Connections routed through interface eth0 will be encrypted while connections routed through the trusted link will not be encrypted.

Using Trusted Links with Service Based Link Selection

In the following scenario, the local and peer Security Gateways have two external interfaces available for VPN traffic. Interface eth1 on both Security Gateways is configured as a trusted interface for VPN traffic since encryption is not needed on that link. In addition, interface eth1 of both Security Gateways is dedicated to SIP traffic using Service Based Link Selection.

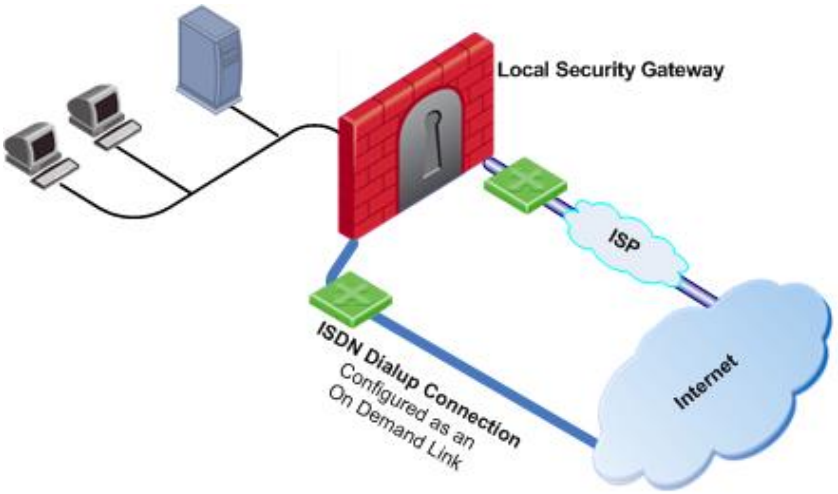


SIP traffic is routed through the trusted link between the two eth1 interfaces and will not be encrypted. If the trusted link stops responding to RDP probing, SIP traffic will be routed through the eth0 interfaces and will be encrypted.

All other traffic that is not SIP is encrypted and routed through the interface eth0 link. However, if interface eth0 stops responding to RDP probing, all the traffic will be routed through the trusted link and will not be encrypted.

On Demand Links (ODL)

Route based probing enables use of an On Demand Link (ODL), which is triggered upon failure of all primary links. When a failure is detected, a custom script is used to activate the ODL and change the appropriate routing information. The ODL's metric must be set to be larger than a configured minimum in order for it to be considered an ODL.



The Security Gateway has two external links for Internet connectivity: one to an ISP, the other to an ISDN dialup. The ISDN dialup connection is configured as an On Demand Link.

On the Security Gateway, the Route Based Probing mechanism probes all of the non-On Demand Links and selects the active link with the lowest metric. In this case, it probed the ISP link. A script is run to activate the On Demand Link when all other links with higher priorities become unavailable. When the link becomes available again, a shutdown script is run automatically and the connection continues through the link with the ISP.

Note - On Demand Links are probed only once using a single RDP session. Fail over between On Demand Links is not supported.

Configuring On Demand Links

You can enable On Demand Links only if you enabled Route Based Probing. Configure On Demand Links commands in GuiDBedit, the Check Point Database Tool.

Property	Description
use_on_demand_links	Enables on-demand links. The default is FALSE. Change to TRUE.
on_demand_metric_min	Defines the minimum metric level for an on-demand link. This value must be equal to or higher than the configured minimum metric.
on_demand_initial_script	The name of the on-demand script, which runs when all not-on-demand routes stop responding. Put the script in the `$FWDIR/conf` directory.
on_demand_shutdown_script	This script is run when the failed links become available. Put the script in the `$FWDIR/conf` directory.

If you do not want to use GuiDBedit, you can configure the `use_on_demand_links` and `on_demand_metric_min` commands in SmartConsole:

1. In SmartConsole, click **Menu > Global Properties > Advanced > Configure**.
2. In **VPN Advanced Properties**, click **Link Selection**.

3. Click **use_on_demand_links** to enable On Demand Links.
4. Set the minimum metric level for an On Demand Link next to the **on_demand_metric_min** command.

Link Selection and ISP Redundancy

ISP Redundancy enables reliable Internet connectivity by allowing a single or clustered Security Gateway to connect to the Internet via redundant ISP connections. As part of standard VPN installation, it offers two modes of operation:

- Load Sharing mode
- Primary/Backup mode

Configuring Link Selection and ISP Redundancy

Configure Link Selection and ISP Redundancy in the **Other** > **ISP Redundancy** page of the Gateway object:

- **Load Sharing** mode connects to both ISPs while sharing the load of outgoing connections between the ISPs according to a designated weight assignment. New connections are randomly assigned to a link. If a link fails, all new outgoing connections are directed to the active link. This configuration effectively increases the WAN bandwidth while providing connectivity protection. The assigned ISP Links weight is only supported for firewall traffic.
- **Primary/Backup** mode connects to an ISP through the primary link, and switches to a backup ISP if the primary ISP link fails. When the primary link is restored, new outgoing connections are assigned to it, while existing connections are maintained over the backup link until they are complete.

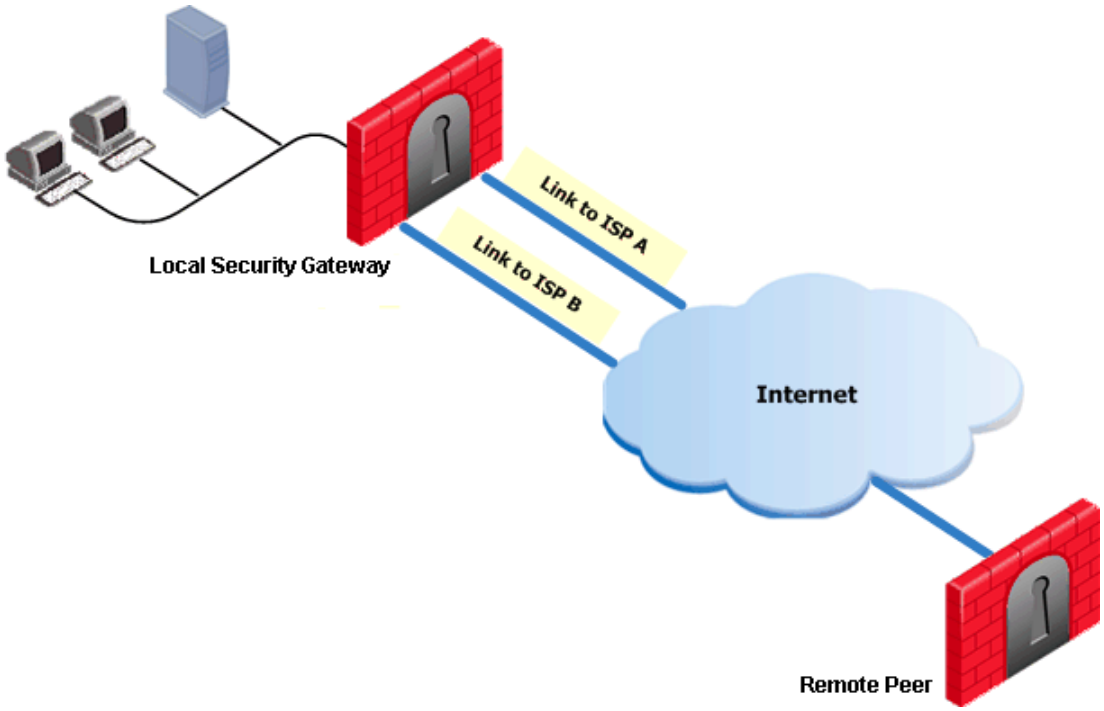
The settings configured in the **ISP Redundancy** window are by default, applied to the **Link Selection** page and will overwrite any pre-existing configuration. The following settings carry over:

- When ISP Redundancy is configured, the default setting in the Link Selection page is **Use ongoing probing**. However, Link Selection only probes the ISPs configured in the **ISP Redundancy** window. This enables connection failover of the VPN tunnel if connectivity to one of the Security Gateway interfaces fails.
- If the ISP Redundancy mode is **Load Sharing**, the Probing redundancy mode in the Link Selection page is also **Load Sharing**.
- If the ISP Redundancy mode is **Primary/Backup**, the Probing redundancy mode in the Link Selection page is **High Availability**.
    - The Primary ISP link of the ISP redundancy is set as the Primary Address of the Link Selection probing. The Primary Address is set under: **IP Selection by Remote Peer > Use Probing > Configure** (or **View** if the settings are derived from the ISP Redundancy settings).

If you do not want the ISP Redundancy settings to affect the Link Selection settings, on the ISP Redundancy page, clear the check box that says **Apply settings to VPN traffic** and configure the required VPN settings on the **Link Selection** page. This may apply when you want to route VPN traffic differently than the firewall traffic. For example, if you want to use Load Sharing for firewall traffic and High Availability for VPN traffic, or if you want to use different primary ISPs for firewall and VPN traffic.

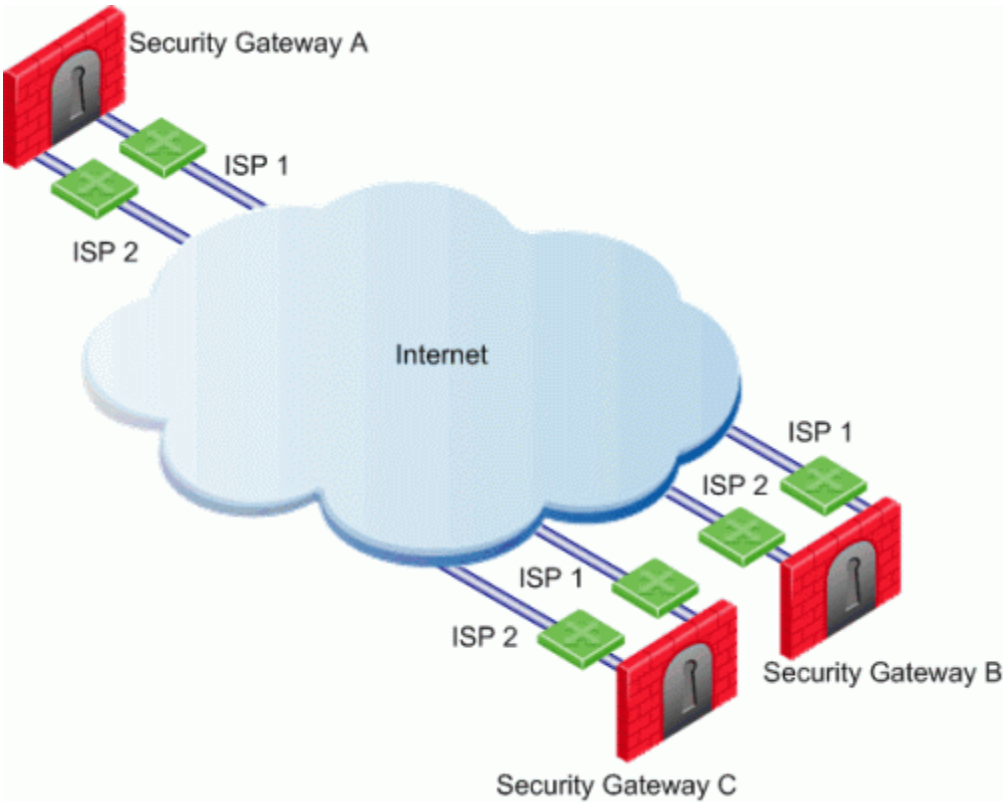
Link Selection and ISP Redundancy

In the following scenario, the local Security Gateway maintains links to ISPs A and B, both of which provide connectivity to the Internet using ISP Redundancy.



In the **Topology > ISP Redundancy** window, configure the ISP Redundancy settings, such as ISP Links and Redundancy mode. The ISP Redundancy settings are applied by default to VPN traffic. The derived Link Selection settings are visible in the **IPsec VPN > Link Selection** window.

In the following scenario, the **Apply settings to VPN traffic o**n the **ISP Redundancy** page was cleared and there are different setting configured for Link Selection and ISP Redundancy.



In this scenario:

- Security Gateways A, B, and C each have two interfaces configured as ISP links.
- **ISP Redundancy** is configured on Security Gateway A.
- Security Gateway A should use ISP 1 in order to connect to Security Gateway B and ISP 2 in order to connect to Security Gateway C. If one of the ISP links becomes unavailable, the other ISP should be used.

In this scenario, the administrator of Security Gateway A needs to:

- Clear the **Apply settings to VPN traffic** box in the **ISP Redundancy** window.
- Reconfigure the **Outgoing Route Selection** to **Route Based Probing** in the **Link Selection** window.
- Configure the routing table so that ISP 1 is the highest priority for peer Security Gateway B and ISP 2 has the highest priority for peer Security Gateway C.

Link Selection with non-Check Point Devices

RDP probing, the probing method used for certain Link Selection features, is proprietary to Check Point and only works between Check Point entities. It is not supported with non-Check Point devices.

Since RDP probing is not active on non-Check Point gateways, the following results apply if a Check Point Security Gateway sends VPN traffic to a non-Check Point gateway:

- **Use probing** cannot be used by locally managed Check Point Security Gateways to determine the IP address of non-Check Point devices. Any of the other methods available from the **IP Selection by Remote Peer** section can be used.
- **Load Sharing** and **Service Based Link Selection** do not work with non-Check Point gateways. If Load Sharing or Service Based Link Selection is enabled on the local Security Gateway, but the peer is a non-Check Point device, the local Security Gateway will only use one link to the non-Check Point device: the best match (highest prefix length) link with the lowest metric.
- If **Route based probing** is selected as the **Outgoing Route Selection** method, for VPN traffic to a non-Check Point device, the local Security Gateways will always use the best match (highest prefix length) link with the lowest metric.

Route Injection Mechanism

In This Section:

Overview of Route Injection

Route Injection Mechanism (RIM) enables a Security Gateway to use dynamic routing protocols to propagate the encryption domain of a VPN peer Security Gateway to the internal network and then initiate back connections. When a VPN tunnel is created, RIM updates the local routing table of the Security Gateway to include the encryption domain of the VPN peer.

Note - Route Injection is not currently supported for IPv6.

RIM can only be enabled when permanent tunnels are configured for the community. Permanent tunnels are kept alive by tunnel test packets. When a Security Gateway fails to reply, the tunnel will be considered 'down.' As a result, RIM will delete the route to the failed link from the local routing table, which triggers neighboring dynamic routing enabled devices to update their routing information accordingly. This will result in a redirection of all traffic destined to travel across the VPN tunnel, to a pre-defined alternative path.

There are two possible methods to configure RIM:

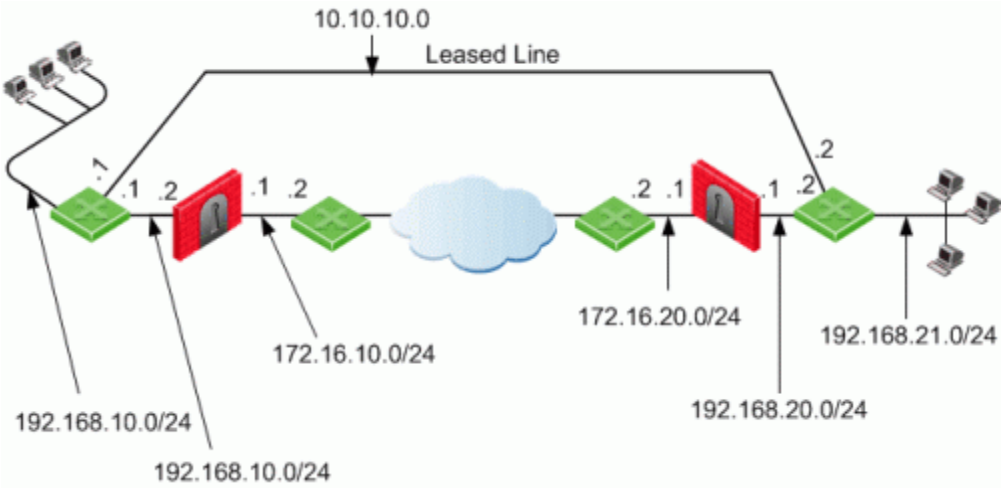
- Automatic RIM - RIM automatically injects the route to the encryption domain of the peer Security Gateways.
- Custom Script - Specify tasks for RIM to perform according to specific needs.

Route injection can be integrated with MEP functionality (which route return packets back through the same MEP Security Gateway). For more information on MEP, see Multiple Entry Point VPNs (see "Multiple Entry Point (MEP) VPNs" on page 114).

Automatic RIM

Automatic RIM can be enabled using the GUI when the operating system on the Security Gateway is SecurePlatform, IPSO or Linux. Although a custom script can be used on these systems, no custom-written scripts are required.

In this scenario:



- Security Gateways 1 and 2 are both RIM and have a dynamic routing protocol enabled.
- R1 and R4 are enabled routers.
- When a VPN tunnel is created, RIM updates the local routing tables of Security Gateway 1 and Security Gateway 2 to include the encryption domain of the other Security Gateway.
- Should the VPN tunnel become unavailable, traffic is redirected to the leased line.

The routing tables for the Security Gateways and routers read as follows. Entries in bold represent routes injected into the Security Gateways local routing tables by RIM:

For Security Gateway 1:

Destination	Netmask	Security Gateway	Metric
0.0.0.0	0.0.0.0	172.16.10.2	1
192.168.21.0	**255.255.255.0**	**172.16.10.2**	**1**
192.168.11.0	255.255.255.0	192.168.10.1	1

Security Gateway 2:

Destination	Netmask	Security Gateway	Metric
0.0.0.0	0.0.0.0	172.16.20.2	1
192.168.11.0	**255.255.255.0**	**172.16.20.2**	**1**
192.168.21.0	255.255.255.0	192.168.20.1	1

R1 (behind Security Gateway 1):

Destination	Netmask	Security Gateway	Metric
0.0.0.0	0.0.0.0	192.168.10.2	1
192.168.21.0	255.255.255.0	192.168.10.2	1
192.168.21.0	255.255.255.0	10.10.10.2	2

R4 (behind Security Gateway 2):

Destination	Netmask	Security Gateway	Metric
0.0.0.0	0.0.0.0	192.168.20.2	1
192.168.11.0	255.255.255.0	192.168.20.2	1
192.168.11.0	255.255.255.0	10.10.10.1	2

Custom Scripts

Custom scripts can be run on any Security Gateway in the community. These scripts are executed whenever a tunnel changes its state, i.e. goes "up" or "down." Such an event, for example, can be the trigger that initiates a dial-up connection.

A script template **custom_rim** (with a **.sh** or **.bat** extension depending on the operating system) is provided in the **$FWDIR/Scripts** directory.

Sample customized script:

```
#!/bin/sh

# This script is invoked each time a tunnel is configured with the RIM option
# and the tunnel changed state.
#
# You may add your custom commands to be invoked here.

# Parameters read from command line.
RIM_PEER_Security Gateway=$1
RIM_NEW_STATE=$2
RIM_HA_STATE=$3
RIM_FIRST_TIME=$4
RIM_PEER_ENC_NET=$5

case "${RIM_NEW_STATE}" in
      up)
      # Place your action for tunnels that came up
      ;;
      down)
      # Place your action for tunnel that went down
      ;;
esac
```

Where:

- RIM_PEER_Security Gateway: Peer Security Gateway
- RIM_NEW_STATE: Change in the state of the Security Gateway, i.e. up or down.
- RIM_HA_STATE: State of a single Security Gateway in a cluster (i.e., standby or active).
- RIM_FIRST_TIME: The script is executed separately for each network within the peer's encryption domain. Although the script might be executed multiple times on a peer, this parameter will only be transferred to the script with the value of '1' the first time the script runs on the peer. The value '1' indicates that this is the first time this script is being executed. The next time the script is executed, it is transferred with the value of '0' and the parameter is disregarded. For example, you may send an email alert to the system administrator the moment a tunnel goes down.

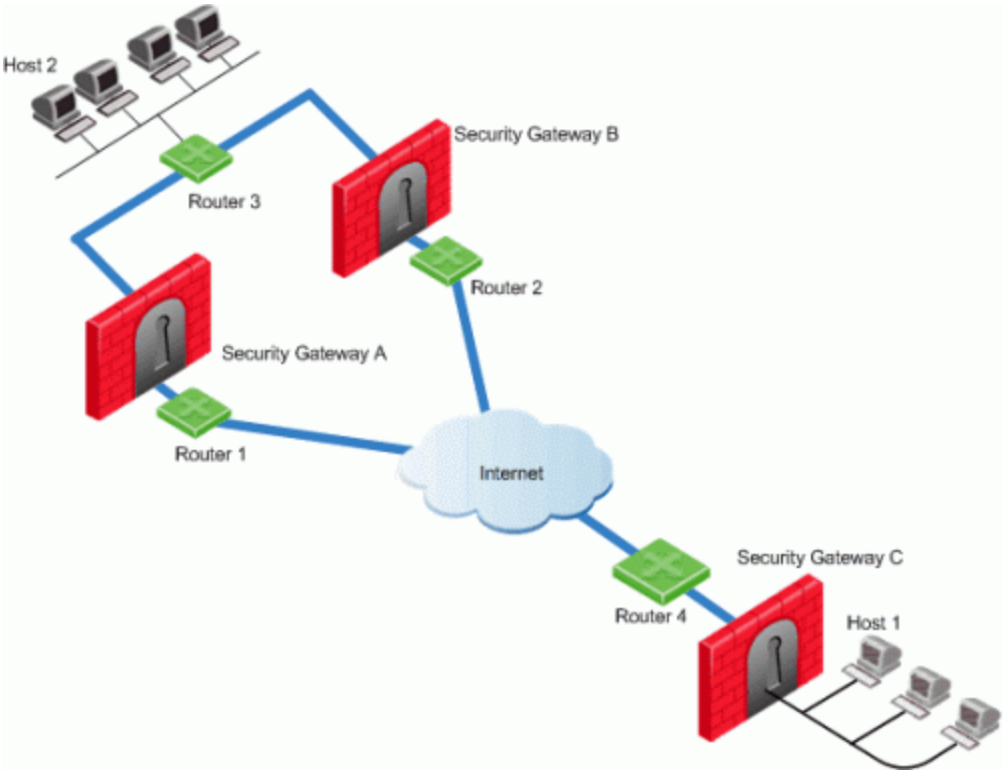
- RIM_PEER_ENC_NET: VPN domain of the VPN peer.

Injecting Peer Security Gateway Interfaces

The **RIM_inject_peer_interfaces** flag is used to inject into the routing tables the IP addresses of the peer Security Gateway in addition to the networks behind the Security Gateway.

For example, after a VPN tunnel is created, RIM injects into the local routing tables of both Security Gateways, the encryption domain of the peer Security Gateway. However, when RIM enabled Security Gateways communicate with a Security Gateway that has Hide NAT enabled, the peer's interfaces need to be injected as well.

In this scenario:



- Security Gateways A and B are both RIM enabled and Security Gateway C has Hide NAT enabled on the external interface ("hiding" all the IP addresses behind it).
- Host 1, behind Security Gateway C, initiates a VPN tunnel with Host 2, through Security Gateway A.
- Router 3 holds routes to all the hosts behind Security Gateway C. Router 3 however, does not have the Hide NAT IP address of Security Gateway C and as a result, cannot properly route packets back to host 1.

This solution for routing the packets back properly is twofold:

1. Select the flag **RIM_inject_peer_interfaces** in the **Global Properties** page. This flag will inject router 3 with all of the IP addresses of Security Gateway C including the Hide NAT address.
2. Configure the router not to propagate the information injected to other Security Gateways. If the router is not configured properly, using the previous example, could result in Security Gateway B routing traffic to Security Gateway C through Security Gateway A.

Configuring RIM

Configuring RIM in a Star Community

1. Open the **Star Community > Tunnel Management** page.
2. In the **Permanent Tunnels** section, select **Set Permanent Tunnels**. The following Permanent Tunnel (see "Permanent Tunnels" on page 71) modes are then made available:
   - **On all tunnels in the community**
   - **On all tunnels of specific Security Gateways**
   - **On specific tunnels in the community**

When choosing tunnels, keep in mind that RIM can only be enabled on tunnels that have been configured to be permanent (see "Configuring Tunnel Features" on page 75). **On all tunnels in the community** must be selected if MEP is enabled on the community.

1. Select **Enable Route Injection Mechanism (RIM)**.
2. Click **Settings...**

   The **Star Community** Settings window opens

   Decide if:
   - RIM should run automatically on the central or satellite Security Gateways (Gaia, SecurePlatform, or IPSO only).
   - A customized script should be run on central or satellite Security Gateways whenever a tunnel changes its states (goes up or down).

   You can also configure the tracking options (on page 76).
3. If a customized script is run, edit **custom_rim** (**.sh** or **.bat**) script in the **$FWDIR/Scripts** directory on each of the Security Gateways.

Configuring RIM in a Meshed Community:

1. Open the **Meshed Community properties > Tunnel Management** page.
2. In the **Permanent Tunnels** section, select **Set Permanent Tunnels**. The following Permanent Tunnel modes are then made available:
   - **On all tunnels in the community**
   - **On all tunnels of specific Security Gateways**
   - **On specific tunnels in the community**

For more information on these options, see Permanent Tunnels (on page 71).

When choosing tunnels, keep in mind that RIM can only be enabled on tunnels that have been configured to be permanent. To configure permanent tunnels, see Configuring Tunnel Features (on page 75).

1. Select **Enable Route Injection Mechanism (RIM)**.
2. Click **Settings...**

   The **Route Injection Mechanism** Settings window open

   Decide if:
   - RIM should run automatically on the Security Gateways (SecurePlatform, IPSO or Linux only).
   - A customized script should be run on the Security Gateway whenever a tunnel changes its state (goes up or down).

For tracking options, see Tracking Options (on page 103).

3. If a customized script is run, edit **custom_rim** (**.sh** or **.bat**) script in the **$FWDIR/Scripts** directory on each of the Security Gateways.

Enabling the RIM_inject_peer_interfaces flag

To enable the **RIM_inject_peer_interfaces** flag:

1. In SmartConsole, click **Menu > Global Properties**.
2. Go to **Advanced** > **Configure**.
   The **Advanced Configuration** window opens
3. Click **VPN Advanced Properties** > **Tunnel Management**.
4. Select **RIM_inject_peer_interfaces**.
5. Click **OK**.

Tracking Options

Several types of alerts can be configured to keep administrators up to date on the status of Security Gateways. The **Tracking** settings can be configured on the Community object, in the **Tunnel Management** > **Enable Route Injection Mechanism** > **Settings** page. The options are **Log**, **Popup Alert, Mail Alert, SNMP Trap Alert**, and **User Defined Alert**.

Configuring RIM on Gaia

In Gaia, the Route Injection Mechanism adds routes directly to the kernel. For the routes to remain in the Kernel, you must configure this option.

To set kernel routes using the CLI:

1. Run: `set kernel-routes on`.
2. Run: `save config`.

To set kernel routes using the WebUI:

1. In the tree view, click **Advanced Routing > Routing Options**.
2. In the **Kernel Options** area, select the **Kernel Routes** option.
3. Click **Apply**.

Gaia Gateways in a Star VPN Community

For RIM to work, the Gaia gateways in a star VPN community must publish the routes of the satellite networks to the router. For Gaia gateways to publish routes, run these CLI commands on all gateways at the center of the community:

1. `set routemap <Routemap Name> id <ID Number>`
   For example:
   `set routemap RIM id 5`
2. `set routemap <Routemap Name>  id <ID Number> match protocol kernel`
   For example:
   `set routemap RIM id 5 match protocol kernel`

3. `Set ospf export-routemap <Routemap Name> preference 1 on`

   For example:

   `set ospf export-routemap RIM preference 1 on`

4. `set routemap <Routemap Name> id <ID Number> allow`

   For example:

   `set routemap RIM id 5 allow`

5. `set routemap <Routemap Name> id <ID Number> on`

   For example:

   `set routemap RIM2 id 10 on`

6. `set routemap <Routemap Name> id <ID Number> match nexthop <IP of OSPF Interface of the other RIM GW> on`

   For example:

   `set routemap RIM2 id 10 match nexthop <10.16.50.3> on`

7. `set routemap <Routemap Name> id <ID Number> restrict`

   For example:

   `set routemap RIM2 id 10 restrict`

8. `set ospf import-routemap <Routemap Name> preference 1 on`

   For example:

   `set ospf import-routemap RIM2 preference 1 on`

9. `save config`

Wire Mode

In This Section:

Overview of Wire Mode

Wire Mode improves connectivity by allowing existing connections to fail over successfully by bypassing firewall enforcement. Traffic within a VPN community is, by definition, private and secure. In many cases, the firewall and the rule on the firewall concerning VPN connections is unnecessary. Using *Wire Mode*, the firewall can be bypassed for VPN connections by defining internal interfaces and communities as "trusted".

When a packet reaches a Security Gateway, the Security Gateway asks itself two questions regarding the packet(s):

Is this information coming from a "trusted" source?

Is this information going to a "trusted" destination?

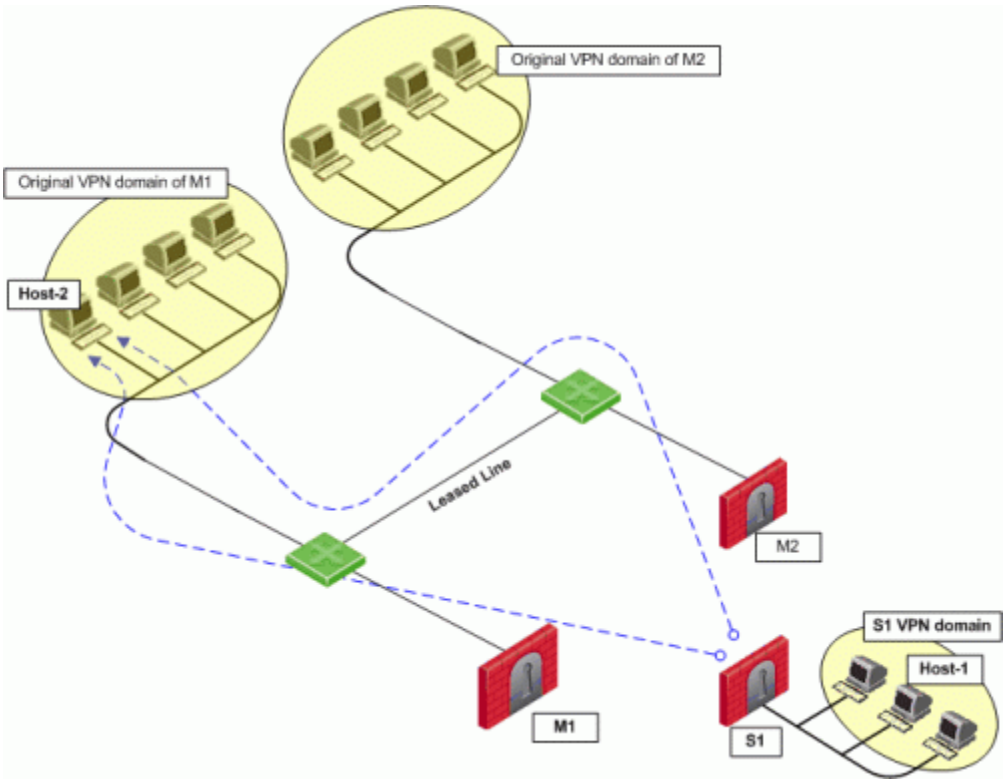
If the answer to both questions is yes, and the VPN Community to which both Security Gateways belong is designated as "*Wire Mode* enabled," stateful inspection is not enforced and the traffic between the trusted interfaces bypasses the firewall. Since no stateful inspection takes place, no packets can be discarded. The VPN connection is no different from any other connection along a dedicated wire. This is the meaning of "*Wire Mode*." Since stateful inspection no longer takes place, dynamic routing protocols (which do not survive state verification in non-wire mode configuration) can now be deployed. *Wire Mode* thus facilitates Route Based VPN (on page 58).

Wire Mode Scenarios

Wire mode can be used to improve connectivity and performance in different infrastructures. This section describes scenarios that benefit from the implementation of wire mode.

Wire Mode in a MEP Configuration

In this scenario:

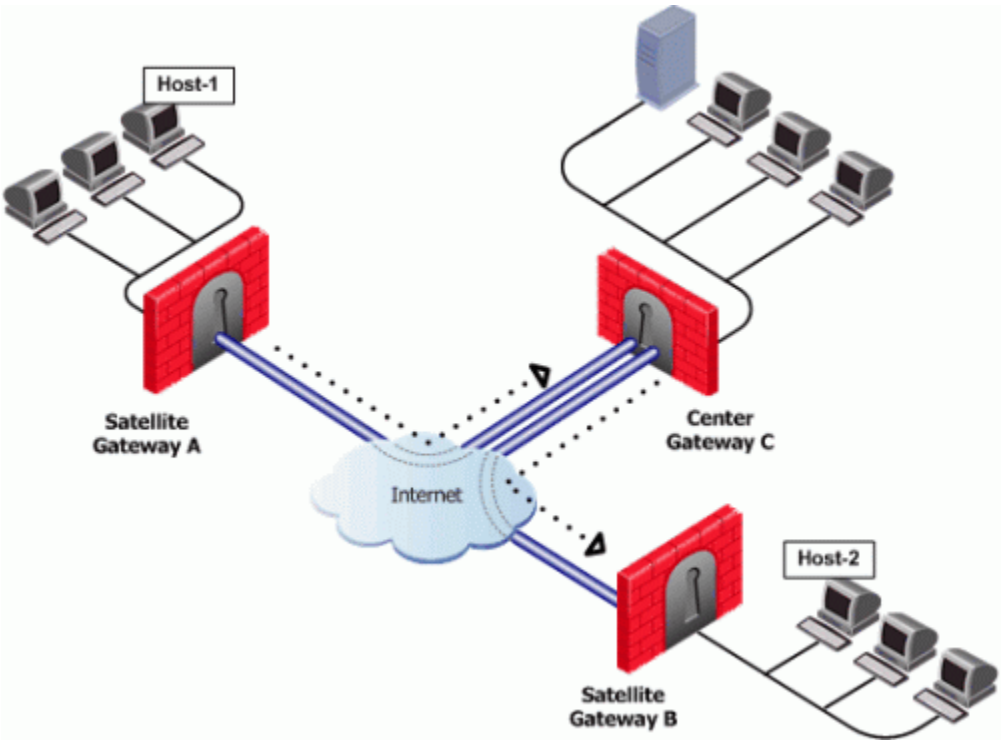


- Security Gateway M1 and Security Gateway M2 are both wire mode enabled and have trusted internal interfaces.
- The community where Security Gateway M1 and Security Gateway M2 reside, is wire mode enabled.
- Host 1, residing behind Security Gateway S1 is communicating through a VPN tunnel with Host 2 residing behind Security Gateway M1.
- MEP ("Multiple Entry Point (MEP) VPNs" on page 114) is configured for Security Gateway M1 and Security Gateway M2 with Security Gateway M1 being the primary Security Gateway and Security Gateway M2 as the backup.

In this case, if Security Gateway M1 goes down, the connection fails over to Security Gateway M2. A packet leaving Host 2 will be redirected by the router behind Security Gateway M1 to Security Gateway M2 since Security Gateway M2 is designated as the backup Security Gateway. Without wire mode, stateful inspection is enforced at Security Gateway M2 and the connection is dropped. Packets that come into a Security Gateway whose session was initiated through a different Security Gateway, are considered "out-of-state" packets. Since Security Gateway M2's internal interface is "trusted," and wire mode in enabled on the community, no stateful inspection is performed and Security Gateway M2 will successfully continue the connection without losing any information.

Wire Mode with Route Based VPN

In this scenario:



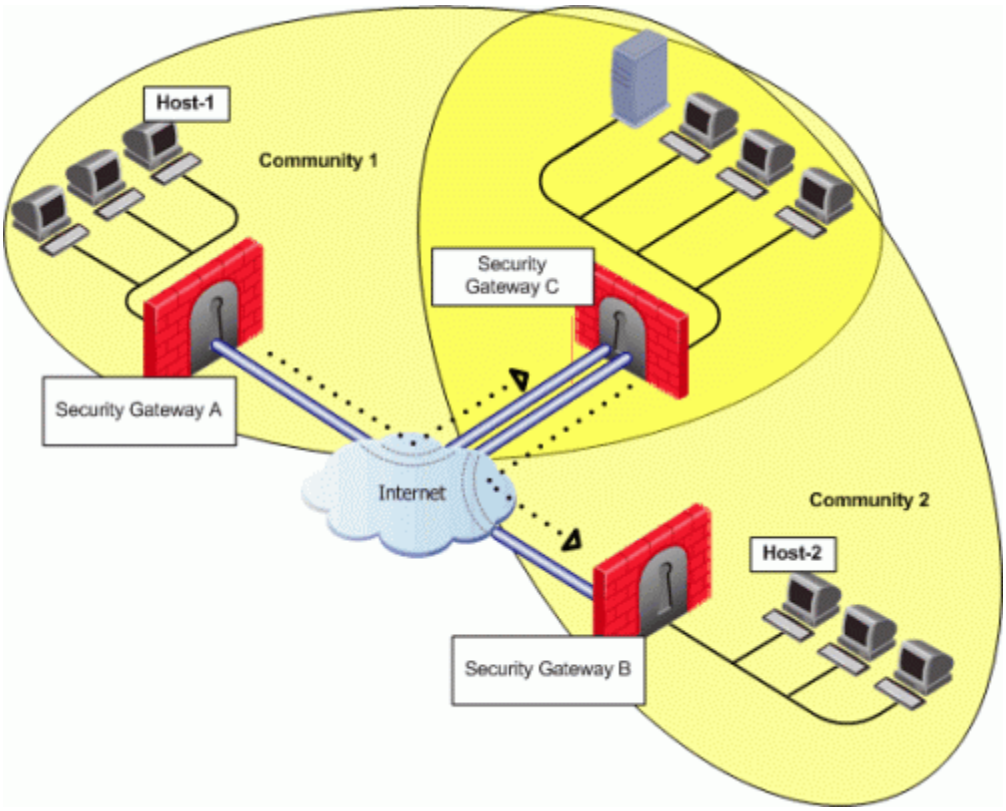
- Wire mode is enabled on Center Security Gateway C (without an internal trusted interface specified).
- The community is wire mode enabled.
- Host 1 residing behind Satellite Security Gateway A wishes to open a connection through a VPN tunnel with Host 2 behind Satellite Security Gateway B.

In a satellite community, Center Security Gateways are used to route traffic between Satellite Security Gateways within the community.

In this case, traffic from the Satellite Security Gateways is only rerouted by Security Gateway C and cannot pass through Security Gateway C's firewall. Therefore, stateful inspection does not need to take place at Security Gateway C. Since wire mode is enabled on the community and on Security Gateway C, making them trusted, stateful inspection is bypassed. Stateful inspection, however, does take place on Security Gateways A and B.

Wire Mode Between Two VPN Communities

In this scenario:



- Security Gateway A belongs to Community 1.
- Security Gateway B belongs to Community 2.
- Security Gateway C belongs to Communities 1 and 2.
- Wire mode is enabled on Center Security Gateway C (without an internal trusted interface specified).
- Wire mode is enabled on both communities.
- Host 1 residing behind Satellite Security Gateway A wishes to open a connection through a VPN tunnel with Host 2 behind Satellite Security Gateway B.

Wire mode can also be enabled for routing VPN traffic between two Security Gateways which are not members of the same community. Security Gateway C is a member of both communities and therefore recognizes both communities as trusted. When host 1 behind Security Gateway A initiates a connection to host 2 behind Security Gateway B, Security Gateway C is used to route traffic between the two communities. Since the traffic is not actually entering Security Gateway C, there is no need for stateful inspection to take place at that Security Gateway. Stateful inspection, however, does take place on Security Gateways A and B.

Special Considerations for Wire Mode

Wire mode is supported on SecurePlatform and Gaia platforms. It does not work with IPv6.

Configuring Wire Mode

Wire mode is configured in two places:

1. Community Properties (meshed or star)
2. Security Gateway Properties

Enabling Wire Mode on a VPN Community

1. In SmartConsole, open the **Object Explorer**, select the VPN community to be configured and click **Edit**.
2. Open the **Wire Mode** page.
3. To enable Wire Mode on the community, select **Allow uninspected encrypted traffic between Wire mode interfaces of the Community members.**
4. To enable Wire Mode Routing, select **Wire Mode Routing - Allow members to route uninspected encrypted traffic in VPN routing configurations**.

Enabling Wire Mode on a Specific Security Gateway

1. In SmartConsole, open the **Gateways & Servers** view, select the relevant Security Gateway and click **Edit**.
2. Open the **IPsec VPN** > **VPN Advanced** page.
3. To enable Wire Mode on the Security Gateway, select **Support Wire Mode (and Wire mode routing...)**
4. Click **Add** to include the interfaces to be trusted by the selected Security Gateway.
5. Select **Log Wire mode traffic** to log wire mode activity.

Directional VPN Enforcement

In This Section:

Overview of Directional VPN

When a VPN community is selected in the VPN column of the Security Policy Rule Base, the source and destination IP addresses can belong to any of the Security Gateways in the community. In other words, the traffic is bidirectional; any of the Security Gateways can be the source of a connection, any of the Security Gateways can be the destination endpoint. But what if the administrator (in line with the company's security policy) wished to enforce traffic in one direction only? Or to allow encrypted traffic to or from Security Gateways *not* included in the VPN community? To enable enforcement within VPN communities, VPN implements Directional VPN.

Directional VPN specifies where the source address must be, and where the destination address must be. In this way, enforcement can take place:

- Within a single VPN community
- Between VPN communities

Directional Enforcement within a Community

The figure shows a simple meshed VPN community called *MyIntranet*. VPN traffic within the MyIntranet Mesh is bidirectional; that is, either of the Security Gateways (or the hosts behind the Security Gateways in the VPN domains) can be the source or destination address for a connection.

Source	Destination	VPN	Service	Action	Track
Any	Any	MyIntranet => MyIntranet MyIntranet =>internal_clear internal_clear => MyIntranet	telnet	accept	log
Any	Any	MyIntranet	telnet	accept	log

The match conditions are represented by a series of compound objects. The match conditions enforce traffic in the following directions:

- To and from the VPN Community via VPN routing (**MyIntranet => MyIntranet**)
- From the Community to the local VPN domains (**MyIntranet =>internal_clear**)
- From the local VPN domains to the VPN community (**internal_clear => MyIntranet**)

Configurable Objects in a Direction

The table shows all the objects that can be configured in a direction, including three new objects created for Directional VPN:

Name of Object	Description
Remote Access	Remote Access community
Site2SiteVPN	Regular Star/Mesh community
Any Traffic	Any traffic
All_GwToGw	All Site2Site communities
All_Communities	All Site2Site and RemoteAccess communities
External_clear	For traffic outside the VPN community
Internal_clear	For traffic between local domains within the community

Note - Clear text connections originating from the following objects are not subject to enforcement:

- Any Traffic
- External_clear
- Internal_clear

There is *no limit* to the number of VPN directions that can be configured on a single rule. In general, if you have many directional enforcements, consider replacing them with a standard bidirectional condition.

Directional Enforcement between Communities

VPN Directional Enforcement can take place between two VPN communities. In this case, one gateway must be configured as a member of both communities and the enforcement point between them. Every other peer gateway in both communities must have a route entry to the enforcement point gateway in its `vpn_route.conf` file.

To add a route entry to the enforcement point gateway:

On the management module of each gateway in the community (except for the enforcement point gateway), add an entry in the $FWDIR/conf/vpn_route.conf file:

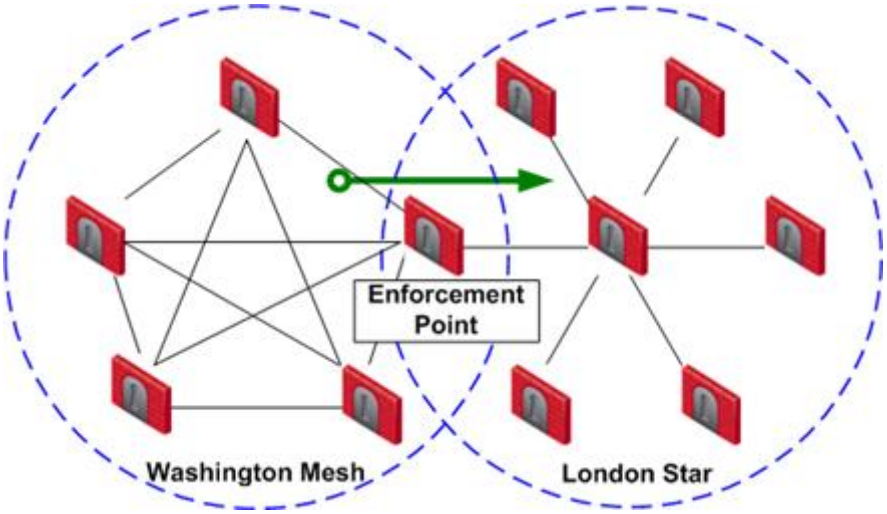
Destination	Next hop router interface	Install on
`<destination_community_obj>`	`<enforcement_point_gw>`	`<managed_FW_object>`

These are the variable in the entry:

- `destination_community_obj` - a network object for the combined encryption domain of the community
- `enforcement_point_gw` - the gateway that is a member of both communities and transfers the encrypted traffic between them

- `managed_FW_object` - all community members that are managed by the management module

In the example below, Washington is a Mesh community, and London is a VPN Star.



The directional VPN rule below must be configured for the enforcement point gateway in the Access Control Policy Rule Base:

Source	Destination	VPN	Services & Applications	Action
Any	Any	Washington => London	Any	accept

The rule is applied to all VPN traffic that passes through the enforcement point gateway between the Washington and London communities. If a connection is opened from a source in the Washington Mesh, and the destination is in the London Star, the connection is allowed. Otherwise, the connection is denied.

Note - The Directional Enforcement applies only to the first packet of a connection. If the connection is permitted, the following packets of this connection are also permitted, including the packets in the opposite direction.

Configuring Directional VPN Within a Community

To configure Directional VPN within a community:

1. In the **Global Properties** > **VPN** > **Advanced** page, select **Enable VPN Directional Match in VPN Column**.
2. In the VPN column of the appropriate rule, select **Directional Match Condition**.

   The **New Directional Match Condition** window opens.
3. In the **Traffic reaching from** drop-down box, select the object for *Internal_clear* (the source).
4. In the **Traffic leaving to** drop-down box, select the relevant community object (the destination).
5. Add another directional match in which the relevant community object is both the source and destination.

   This allows traffic from the local domain to the community, and within the community.
6. Click **OK**.

Configuring Directional VPN Between Communities

To configure Directional VPN between communities:

1. In the **Global Properties** > **VPN** > **Advanced** page, select **Enable VPN Directional Match in VPN Column**.
2. In the VPN column of the appropriate rule, select **Directional Match Condition**.
   The **New Directional Match Condition** window opens.
3. In the **Traffic reaching from** drop-down box, select the source of the connection.
4. In the **Traffic leaving to** drop-down box, select the destination of the connection
5. Click **OK**.

Multiple Entry Point (MEP) VPNs

In This Section:

Overview of MEP

Multiple Entry Point (MEP) is a feature that provides a High Availability and Load Sharing solution for VPN connections. A Security Gateway on which the VPN module is installed provides a single point of entry to the internal network. It is the Security Gateway that makes the internal network "available" to remote machines. If a Security Gateway should become unavailable, the internal network too, is no longer available. A MEP environment has two or more Security Gateways both protecting and enabling access to the same VPN domain, providing peer Security Gateways with uninterrupted access.

VPN High Availability Using MEP or Clustering

Both MEP and Clustering are ways of achieving High Availability and Load Sharing. However:

- Unlike the members of a ClusterXL Security Gateway Cluster, there is no physical restriction on the location of MEP Security Gateways. MEP Security Gateways can be geographically separated machines. In a cluster, the clustered Security Gateways need to be in the same location, directly connected via a *sync* interface.
- MEP Security Gateways can be managed by different Security Management Server; cluster members must be managed by the same Security Management Server.
- In a MEP configuration there is no "state synchronization" between the MEP Security Gateways. In a cluster, all of the Security Gateways hold the "state" of all the connections to the internal network. If one of the Security Gateways fails, the connection passes seamlessly over (performs *failover*) to another Security Gateway, and the connection continues. In a MEP configuration, if a Security Gateway fails, the current connection is lost and one of the backup Security Gateways picks up the *next* connection.
- In a MEP environment, the decision which Security Gateway to use is taken on the remote side; in a cluster, the decision is taken on the Security Gateway side.

Implementation

MEP is implemented via a proprietary *Probing Protocol* (PP) that sends special UDP RDP packets to port 259 to discover whether an IP is reachable. This protocol is proprietary to Check Point and does not conform to RDP as specified in RFC 908/1151.

Note - These UDP RDP packets are not encrypted, and only test the availability of a peer.

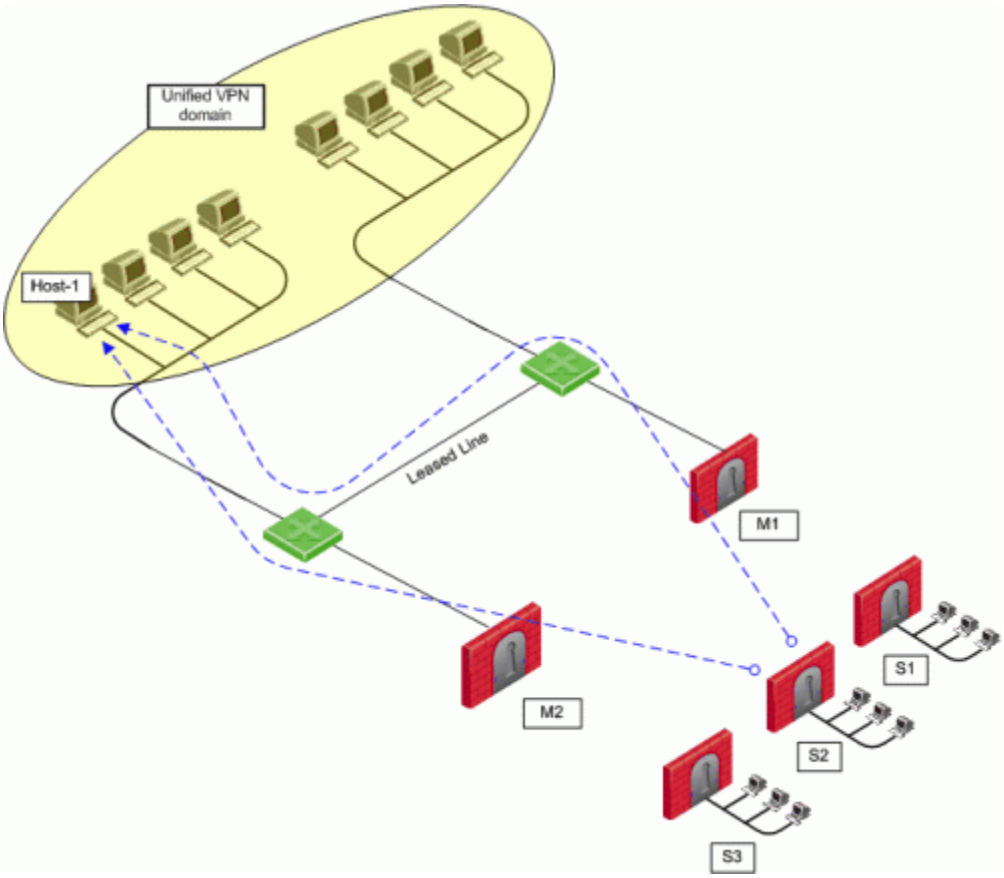
The peer continuously probes or polls all MEP Security Gateways in order to discover which of the Security Gateways are "up", and chooses a Security Gateway according to the configured selection mechanism. Since RDP packets are constantly being sent, the status of all Security Gateways is known and updated when changes occur. As a result, all Security Gateways that are "up" are known.

There are two available methods to implement MEP:

- Explicit MEP - Only Star communities with more than one central Security Gateway can enable explicit MEP, providing multiple entry points to the network behind the Security Gateways. When available, Explicit MEP is the recommended method.
- Implicit MEP - Implicit MEP is supported in all scenarios where fully or partially overlapping encryption domains exist or where Primary-Backup Security Gateways (on page 123) are configured. When upgrading from a version prior to NGX (R60) where Implicit MEP was already configured, the settings previously configured will remain.

Explicit MEP

In a site to site Star VPN community, explicit MEP is configured in the VPN community object. When MEP is enabled, the satellites consider the "unified" VPN domain of all the Security Gateways as the VPN domain for each Security Gateway. This unified VPN domain is considered the VPN domain of each Security Gateway:



In the figure, a Star VPN community has two central Security Gateways, M1 and M2 (for which MEP has been enabled) and three satellite Security Gateways — S1, S2, and S3. When S2 opens a connection with host-1 (which is behind M1 and M2), the session will be initiated through either M1

or M2. Priority amongst the MEP Security Gateways is determined by the MEP entry point selection mechanism.

If M2 is the selected entry point and becomes unavailable, the connection to host-1 fails over to M1. Returning packets will be rerouted using RIM or IP Pool NAT. For more information about returning packets, see Routing Return Packets (on page 124).

There are four methods used to choose which of the Security Gateways will be used as the entry point for any given connection:

- Select the closest Security Gateway to source (First to respond)
- Select the closest Security Gateway to destination (By VPN domain)
- Random Selection (for Load distribution)
- Manually set priority list (MEP rules)

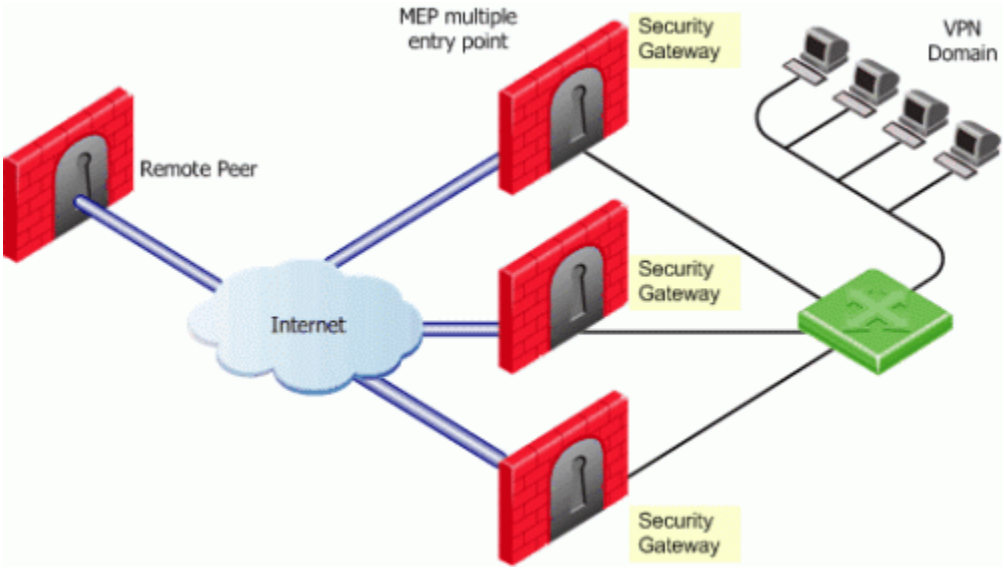
If either "By VPN domain" or "Manually set priority list" is selected, then **Advanced** options provide additional granularity.

MEP Selection Methods

- **First to Respond**, in which the first Security Gateway to reply to the peer Security Gateway is chosen. An organization would choose this option if, for example, the organization has two Security Gateways in a MEP configuration - one in London, the other in New York. It makes sense for peers located in England to try the London Security Gateway first and the NY Security Gateway second. Being geographically closer to the peers in England, the London Security Gateway will be the first to respond, and becomes the entry point to the internal network. See: First to Respond (on page 122).
- **VPN Domain,** is when the destination IP belongs to a particular VPN domain, the Security Gateway of that domain becomes the chosen entry point. This Security Gateway becomes the primary Security Gateway while other Security Gateways in the MEP configuration become its backup Security Gateways. See: By VPN Domain (on page 118).
- **Random Selection**, in which the remote peer randomly selects a Security Gateway with which to open a VPN connection. For each IP source/destination address pair, a new Security Gateway is randomly selected. An organization might have a number of machines with equal performance abilities. In this case, it makes sense to enable load distribution. The machines are used in a random and equal way. See: Random Selection (on page 119).
- **Manually set priority list,** Security Gateway priorities can be set manually for the entire community or for individual satellite Security Gateways. See: Manually Set Priority List (on page 119).

First to Respond

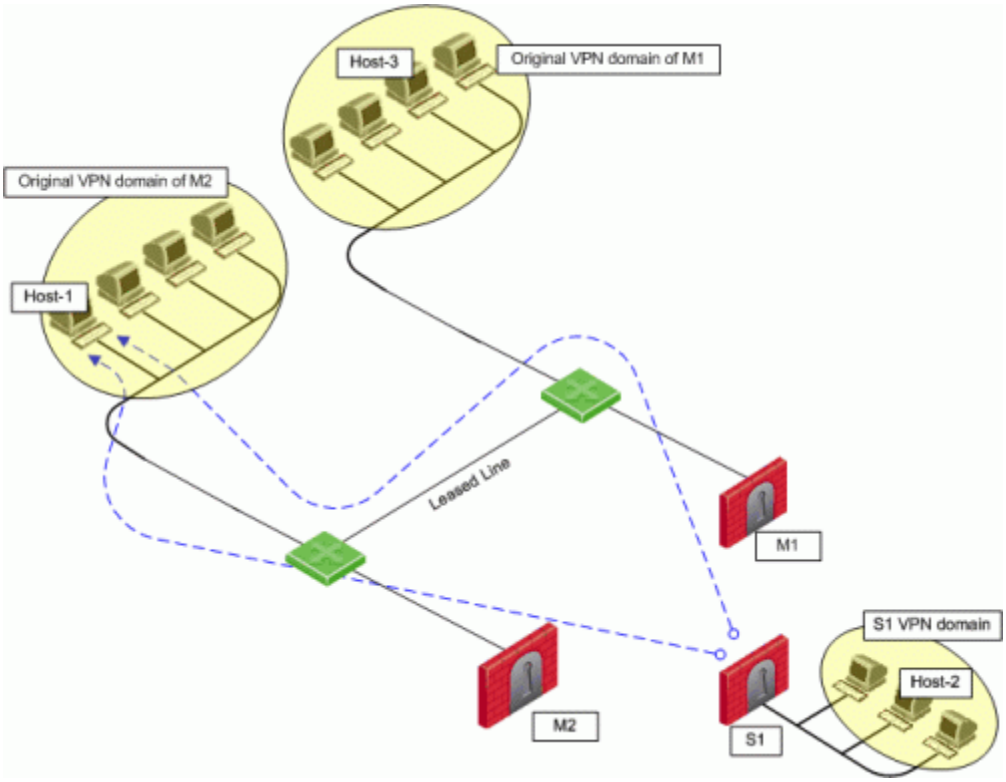
When there is no primary Security Gateway, all Security Gateways share "equal priority". When all Security Gateways share equal priority:



- Remote peers send RDP packets to all the Security Gateways in the MEP configuration.
- The first Security Gateway to respond to the probing RDP packets gets chosen as the entry point to network. The idea behind *first to respond* is proximity. The Security Gateway which is "closer" to the remote peer responds first.
- A VPN tunnel is opened with the first to respond. All subsequent connections pass through the chosen Security Gateway.
- If the Security Gateway ceases to respond, a new Security Gateway is chosen.

By VPN Domain

Before you enable MEP, each IP address belongs to a specific VPN domain. With **By VPN Domain**, the Security Gateway of that domain becomes the chosen entry point. In the figure, the VPN Star community has two central MEP Security Gateways (M1 and M2, each of which *have their own VPN domains*), and remote satellite S1.



Host-2 (in the VPN domain of satellite S1 initiates a connection with host-1. The connection can be directed through either M1 or M2. However, host-1 is within M2's original VPN domain. For this reason, M2 is considered the Security Gateway "closest" to the destination IP Address. M2 is therefore considered the primary Security Gateway and M1 the backup Security Gateway for Host-1. If there were additional Security Gateways in the center, these Security Gateways would also be considered as backup Security Gateways for M2.

If the VPN domains have fully or partially overlapping encryption domains, then more than one Security Gateway will be chosen as the "closest" entry point to the network. As a result, more than one Security Gateway will be considered as "primary." When there are more than one primary or backup Security Gateways available, the Security Gateway is selected using an additional selection mechanism. This advanced selection mechanism can be either (See Advanced Settings (on page 121)):

- First to Respond
- Random Selection (for load distribution)

For return packets you can use RIM on the center Security Gateways. If RIM is also enabled, set a metric with a lower priority value for the leased line than the VPN tunnel. The satellite S1 might simultaneously have more than one VPN tunnel open with the MEP Security Gateways, for example M2 as the chosen entry point for host-1 and M1 as the chosen entry point for host-3. While both M1 and M2 will publish routes to host-1 and host-3, the lower priority metric will ensure the leased line is used only when one of the Security Gateways goes down.

Random Selection

Using this method, a different Security Gateway is randomly selected as an entry point for incoming traffic. Evenly distributing the incoming traffic through all the available Security Gateways can help prevent one Security Gateway from becoming overwhelmed with too much incoming traffic.

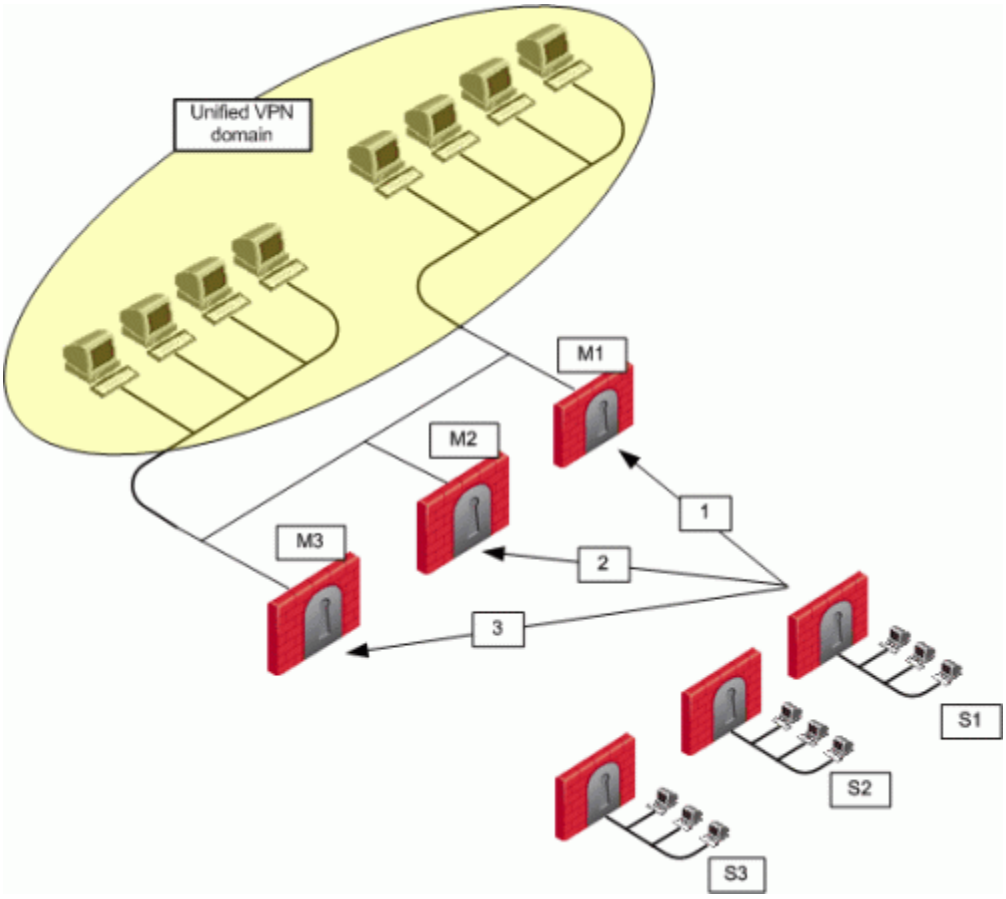
The Security Gateways are probed with RDP packets, as in all other MEP configurations, to create a list of responding Security Gateways. A Security Gateway is randomly chosen from the list of responding Security Gateways. If a Security Gateway stops responding, another Security Gateway is (randomly) chosen.

A new Security Gateway is randomly selected for every source/destination IP pair. While the source and destination IP's remain the same, the connection continues through the chosen Security Gateway.

In such a configuration, RIM is not supported. IP Pool NAT must be enabled to ensure return packets are correctly routed through the chosen Security Gateway.

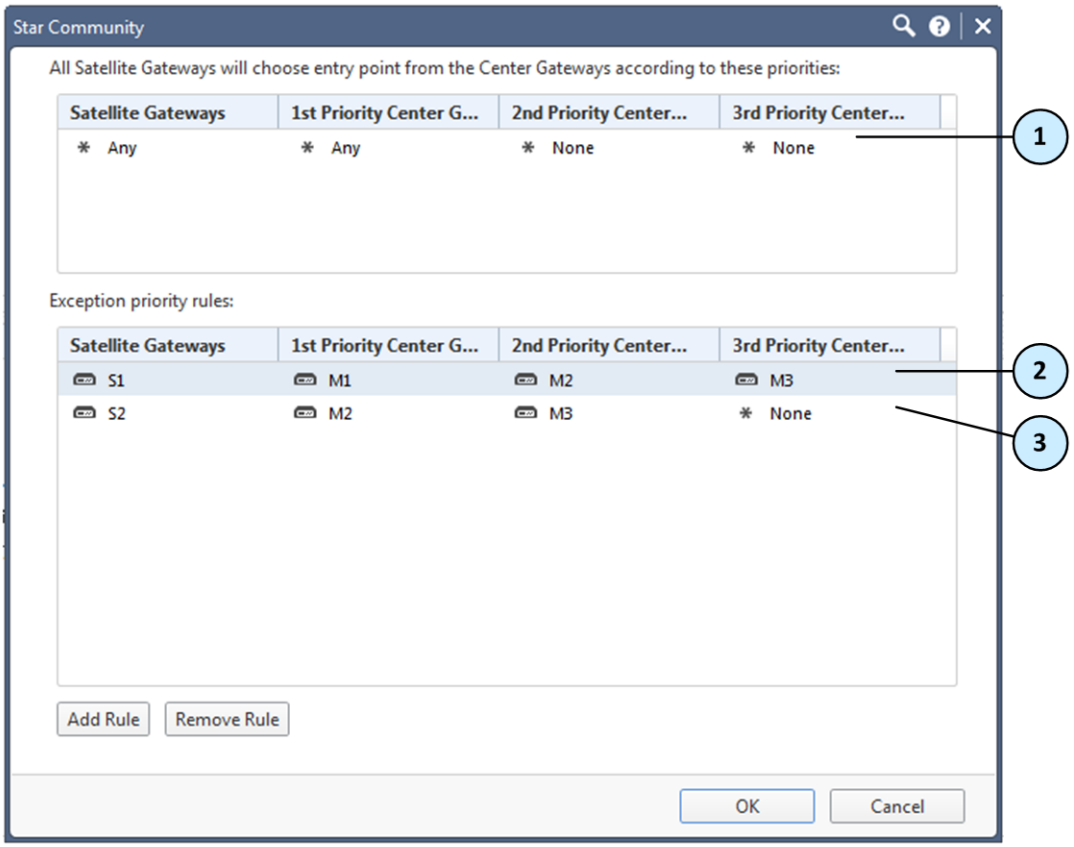
Manually Set Priority List

The Security Gateway that will be chosen (from the central Security Gateways in the star community) as the entry point to the core network can be controlled by manually setting a priority per source Security Gateway. Each priority constitutes a MEP Rule:



In the figure, three MEP members (M1, M2, M3) provide entry points to the network for three satellite Security Gateways (S1, S2, S3). Satellite S1 can be configured to try the Security Gateways in the following order: M1, M2, M3, giving the highest priority to M1, and the lowest priority to M3. Satellite S2 can be configured to try the Security Gateways in the following order: M2, M3 (but not to try M1).

Each of these priorities constitutes a MEP rule in the MEP manual priority list window:



Item	Description
1	Default MEP Rule
2	First MEP Rule
3	Second MEP Rule

The MEP manual priority list window is divided into the default rule, and rules which provide exceptions to the default rule. The default MEP rule takes effect when:

- No MEP rules are defined
- When the source of the connection cannot be found in the Exception priority rules

The Exception priority rules section contains three priority levels: primary, secondary, and tertiary. While there are only three priority levels,

- The same priority can be assigned to several central Security Gateways
- The same rule can be assigned to several satellite Security Gateways
- A priority level can be left blank

In the second MEP rule below:

Satellite Gateways	1st Priority Center G...	2nd Priority Center...	3rd Priority Center...
S1	M1	M2	M3
S2 S3	M2	M3 M1	None

Central Security Gateways M3 and M1 have equal priority. The same rule is being applied to satellites S2 and S3.

When more than one Security Gateway is assigned the same priority level, which Security Gateway will be chosen is resolved according to the Advanced settings. See Advanced Settings (on page 121).

Advanced Settings

In some instances, more than one Security Gateway is available in the center with no obvious priority between them. For example — as shown in the second example of the second MEP rule, above — more than one Security Gateway is assigned "second" priority. In this scenario, **Advanced** options are used to decide which Security Gateway is chosen: *First to Respond*, or *Random Selection*. (Choose Random selection to enable load balancing between the Security Gateways.)

When "manually set priority list" is the MEP selection mechanism, *RIM is supported.* RIM can be configured with "manually set priority list" because the "random selection" mechanism available on the **Advanced** button is different from the random selection mechanism used for MEP.

For the "random selection" mechanism employed for MEP, a different Security Gateway is selected for each IP source/destination pair. For the random selection mechanism available from the **Advanced** button, a single MEP entry point is randomly selected and then used for all connections, and does not change according to source/destination pair. Load distribution is therefore achieved since every satellite Security Gateway is randomly assigned a Security Gateway as its entry point. This makes it possible to enable RIM at the same time.

Tracking

If the tracking option is enabled for MEP, the following information is logged by each satellite Security Gateway:

- The resolved peer Security Gateway (a Security Gateway in the MEP)
- The priority of the resolved Security Gateway (primary, secondary, tertiary)
- Whether the resolved Security Gateway is responding

For example, in the scenario shown in the Manually Set Priority List (on page 119) section, satellite S1 opens a connection to the VPN domain that includes Security Gateways M1, M2, and M3. M1 is the resolved peer. If tracking is enabled, the log reads:

```
Resolved peer for tunnel from S1 to the MEP that contains M1, M2, and M3,
is: M1 (Primary Security Gateway, responding).
```

Implicit MEP

There are three methods to implement implicit MEP:

- *First to Respond*, in which the first Security Gateway to reply to the peer Security Gateway is chosen. An organization would choose this option if, for example, the organization has two Security Gateways in a MEP configuration - one in London, the other in New York. It makes sense for VPN-1 peers located in England to try the London Security Gateway first and the NY Security Gateway second. Being geographically closer to VPN peers in England, the London Security Gateway is the first to respond, and becomes the entry point to the internal network. See: First to Respond (on page 122).
- *Primary-Backup*, in which one or multiple backup Security Gateways provide "high availability" for a primary Security Gateway. The remote peer is configured to work with the primary Security Gateway, but switches to the backup Security Gateway if the primary goes down. An

organization might decide to use this configuration if it has two machines in a MEP environment, one of which is stronger than the other. It makes sense to configure the stronger machine as the primary. Or perhaps both machines are the same in terms of strength of performance, but one has a cheaper or faster connection to the Internet. In this case, the machine with the better Internet connection should be configured as the primary. See: Primary-Backup Security Gateways (on page 123).

- *Load Distribution*, in which the remote VPN peer randomly selects a Security Gateway with which to open a connection. For each IP source/destination address pair, a new Security Gateway is randomly selected. An organization might have a number of machines with equal performance abilities. In this case, it makes sense to enable load distribution. The machines are used in a random and equal way. See: Random Selection (on page 119).

Implicit MEP is supported if the Security Gateways with overlapping encryption domains are in the same community. If they are located in different communities, only one of the Security Gateways will be used for this encryption domain.

Note - When upgrading from a version prior to NGX R60 where Implicit MEP was already configured, the settings previously configured will remain.

First to Respond

When there is no primary Security Gateway, all Security Gateways share "equal priority." When all Security Gateway's share "equal priority":

- Remote VPN peers send RDP packets to all the Security Gateways in the MEP configuration.
- The first Security Gateway to respond to the probing RDP packets gets chosen as the entry point to network. The idea behind *first to respond* is "proximity". The Security Gateway which is "closer" to the remote VPN peer responds first.
- A VPN tunnel is opened with the first to respond. All subsequent connections pass through the chosen Security Gateway.
- If the Security Gateway ceases to respond, a new Security Gateway is chosen.

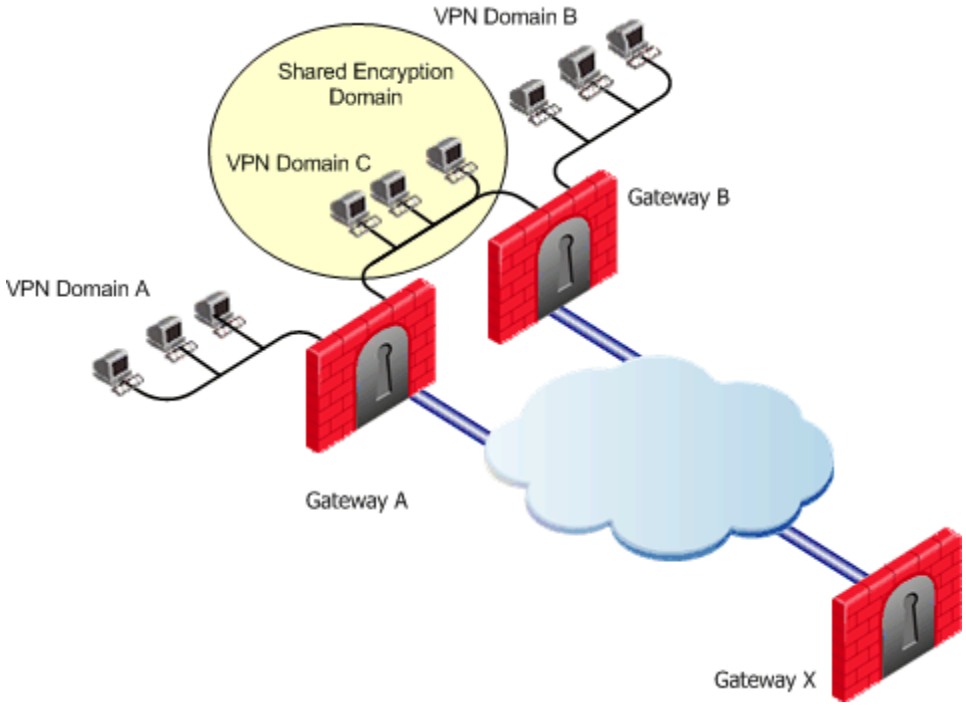
In a star community, RDP packets are sent to the Security Gateways and the first to respond is used for routing only when:

There is more than one center Security Gateway, **and**

One of the following VPN routing options was selected:

- **To center and to other satellites through center**
- **To center, or through the center to other satellites, to internet and other VPN targets**
  This setting is found on the **Community Properties > VPN Advanced > VPN Routing** page.

In this scenario:



- MEP is **not** enabled on the community
- First to respond method is used
- Security Gateway X accesses VPN domain A through Security Gateway A
- Security Gateway X accesses VPN domain B through Security Gateway B
- Security Gateway X accesses VPN domain C through Security Gateway A or B

In a star community, RDP packets are sent to the Security Gateways and the first to respond is used for routing when:

There is more than one center Security Gateway, and

One of the following VPN routing options was selected:

- **To center and to other satellites through center**
- **To center, or through the center to other satellites, to internet and other VPN targets**
  This setting is found on the **Community Properties > VPN Advanced > VPN Routing** page.

Primary-Backup Security Gateways

Backup Security Gateways provide redundancy for primary Security Gateways. If the primary Security Gateway fails, connections go through the backup.

The first Security Gateway is configured as the "primary," and the second Security Gateway as the "backup." If the primary Security Gateway fails, for whatever reason, the remote VPN peer detects that the link has gone down and works through the backup Security Gateway. The backup gateway inherits the complete VPN domain of the primary. Failover within an existing connection is not supported; the current connection is lost.

When the primary Security Gateway is restored, new connections go through the primary Security Gateway while connections that already exist will continue to work through the backup Security Gateway.

Note - When using the Primary-Backup Security Gateways method, the encryption domains should not overlap

Load Distribution

To prevent any one Security Gateway from being flooded with connections, the connections can be evenly shared amongst all the Security Gateways to distribute the load. When all Security Gateways share equal priority (no primary) and are MEP to the *same* VPN domain, it is possible to enable load distribution between the Security Gateways. The Security Gateways are probed with RDP packets, as in all other MEP configurations, to create a list of responding Security Gateways. A Security Gateway is randomly chosen from the list of responding Security Gateways. If a Security Gateways stops responding, a new Security Gateway is (randomly) chosen.

A new Security Gateway is randomly selected for every source/destination IP pair. While the source and destination IP's remain the same, the connection continues through the chosen Security Gateway.

Routing Return Packets

To make sure return packets are routed correctly, the MEP Security Gateway can make use of either:

- IP pool NAT (static NAT) or
- Route Injection Mechanism

IP Pool Network Address Translation (NAT)

IP pool NAT is a type of NAT in which source IP addresses from remote VPN domains are mapped to an IP address drawing from a pool of registered IP addresses. In order to maintain symmetric sessions using MEP Security Gateways, the MEP Security Gateway performs NAT using a range of IP addresses dedicated to that specific Security Gateway and should be routed within the internal network to the originating Security Gateway. When the returning packets reach the Security Gateway, the Security Gateway restores the original source IP address and forwards the packets to the source.

RIM

Route Injection Mechanism (RIM) enables a Security Gateway to use a dynamic routing protocol to propagate the encryption domain of a VPN peer Security Gateway to the internal network. When a VPN tunnel is created, RIM updates the local routing table of the Security Gateway to include the encryption domain of the VPN peer.

When a tunnel to a MEP Security Gateway goes down, the Security Gateway removes the appropriate "return route" from its own local routing table. This change is then distributed backwards to the routers behind the Security Gateway.

RIM is based both on the ability of the Security Gateway to update its local routing table, and the presence of the a dynamic routing protocol to distribute the change to the network behind the Security Gateway. There is little sense in enabling RIM on the Security Gateway if a dynamic routing protocol is not available to distribute changes.

When MEP is enabled, RIM can be enabled only if permanent tunnels are enabled for the whole community. In a MEP configuration RIM is available when using the *First to Respond*, *Manual set*

priority list, and *VPN Domain* mechanisms. In the first two options, satellite Security Gateways "see" the center Security Gateways as unified as if one tunnel is connecting them. As a result, only the chosen MEP Security Gateway will inject the routes. In *VPN Domain* MEP, it could be that all MEP Security Gateways will inject the routes, which requires configuring the routers behind the MEP Security Gateways to return packets to the correct Security Gateway.

RIM is not available when *Random Selection* is the selected entry point mechanism.

For more information on RIM, see Route Injection Mechanism (on page 98).

Special Considerations

1. If one of the central Security Gateways is an externally managed Security Gateway:
   - The VPN domain of the central Security Gateways will not be automatically inherited by an externally managed Security Gateway
   - The RIM configuration will not be automatically downloaded
2. UTM-1 Edge Security Gateways cannot be configured as a MEP Security Gateway but can connect to MEP Security Gateways.
3. DAIP Security Gateways require DNS resolving in order to be configured as MEP Security Gateways.

Configuring MEP

To configure MEP, decide on:

1. The MEP method
   - Explicit MEP - See Explicit MEP (on page 115).
   - Implicit MEP - See Implicit MEP (on page 121).
2. If required, method for returning reply packets:
   - IP pool NAT
   - RIM - To configure RIM, see Configuring RIM (on page 102).

Configuring Explicit MEP

Explicit MEP is only available in Site-to-Site Star VPN communities where multiple central Security Gateways are defined.

To configure MEP:

1. Open the **Star Community** object and go to the **MEP** page.
2. Select **Enable center gateways as MEP**.
3. Select an entry point mechanism:
   - First to respond
   - By VPN domain
   - Random selection
   - Manual priority list

If you select **By VPN domain** or **Manually set priority list**, in the **Advanced**, section choose **First to respond** or **Random selection** to resolve how more than one Security Gateway with equal priority should be selected.

If you select **Manually set priority list**, click **Set** to create a series of MEP rules.

4. Select a **Tracking** option, if required.

Configuring Implicit MEP

Configuring Implicit First to Respond

When more than one Security Gateway leads to the same (overlapping) VPN domain, they are in a MEP configuration. The first Security Gateway to respond is chosen. To configure *first to respond*, define that part of the network that is shared by all the Security Gateways into a single group and assign that group as the VPN domain.

Before you start, make sure that **Load Distribution** is not enabled:

1. In SmartConsole, go to **Menu** > **Global Properties** > **VPN** > **Advanced**.
2. Clear the **Enable load distribution for Multiple Entry Points** option.

To configure First to Respond MEP:

1. For each Security Gateway in the VPN domain, run `vpn overlap_encdom`.
2. In SmartConsole, create a host group and assign all these Security Gateways to it.
3. In each Security Gateway object go to the **Network Management** > **VPN Domain** page,
4. Select **Manually defined**.
5. Select the host group of MEP gateways that you defined in step 2.
6. Click **OK**.
7. Install the Access Control Policy on all Security Gateways.

Configuring Implicit Primary-Backup

Configure the VPN Domain that includes the Primary gateway and another domain that includes only the backup gateway. Configure each gateway as either the Primary gateway or a backup gateway.

To configure the primary gateway:

1. Open **Global Properties** window > **VPN > Advanced**, select **Enable Backup Gateway**.
2. In the Object Explorer, click **New** > **Network Group** and create a group of gateways to act as backup gateways.
3. Edit the Primary gateway object and open the **IPsec VPN** page.
4. Select **Use Backup Gateways**, and select the group of backup gateways.

   This gateway is the primary gateway for this VPN domain.
5. For each backup gateway, make a VPN domain that does not include IP addresses that are in the Primary VPN domain or the other backup domains.

   If the backup gateway already has a VPN domain, you must make sure that its IP addresses do not overlap with the other VPN domains.

   a) Create a group of IP addresses not in the other domains, or a group that consists of only the backup gateway.

b) In the backup network object, go to the **Network Management** > **VPN Domain** section, select **Manually defined**.

c) Select the group.

6. Click **OK**.
7. Install the policy.

Configuring Implicit Load Distribution

To configure implicit MEP for random gateway selection:

1. Click **Menu** > **Global Properties**.
2. Open the **VPN** > **Advanced** page.
3. Select **Enable load distribution for Multiple Entry Point configurations (Site to Site connections)**.
4. Define the same VPN domain for all the gateways:

   a) Create a group of the gateways.

   b) In each gateway network object, go to the **Network Management** > **VPN Domain** page, and select **Manually defined**.

   c) Select the group.

5. Click **OK**.
6. Install the Access Control Policy.

Configuring IP Pool NAT

To configure IP pool NAT for site to site VPN:

1. In Menu > **Global Properties**, open the **NAT** page, and click **Enable IP Pool NAT**.
2. Set tracking options for address exhaustion and for address allocation and release.
3. For each Security Gateway, create a network object that represents the IP pool NAT addresses for that Security Gateway. The IP pool can be a network, group, or address range. For example:
   - Open the **Object Explorer** (Ctrl+E) and click **New** > **Network Object** > **Address Range** > **Address Range**. The **New Address Range** window opens.
   - On the **General** tab, enter the first IP and last IP of the address range.
   - Click **OK**.
4. On the Security Gateway object where IP pool NAT translation is performed, in the **NAT > IP Pool NAT** page, select either
   - **Allocate IP Addresses** from, and select the address range you created, OR
   - **Define IP Pool addresses on Security Gateway interfaces**. If you choose this option, you need to define the IP Pool on each required interface, in the **Interface Properties** window, **IP Pool NAT** tab.
5. In the **IP Pool NAT** page, select either (or all):
   - **Use IP Pool NAT for VPN clients connections**
   - **Use IP Pool NAT for Security Gateway to Security Gateway connections**
   - **Prefer IP Pool NAT over Hide NAT**
6. Click **Advanced...**

- Decide after how many minutes unused addressees are returned to the IP pool.
- Click **OK** twice.

7. Edit the routing table for each internal router, so that packets with an IP address assigned from the NAT pool are routed to the appropriate Security Gateway.

Resolving Connectivity Issues

In This Section:

IPsec NAT-Traversal

NAT-T (NAT traversal or UDP encapsulation) makes sure that IPsec VPN connections stay open when traffic goes through gateways or devices that use NAT.

When an IP packet passes through a network address translator device, it is changed in a way that is not compatible with IPsec. To protect the original IPsec encoded packet, NAT traversal encapsulates it with an additional layer of UDP and IP headers.

For IPsec to work with NAT traversal, these protocols must be allowed through the NAT interface(s):

- IKE - UDP port 500
- IPsec NAT-T - UDP port 4500
- Encapsulating Security Payload (ESP) - IP protocol number 50
- Authentication Header (AH) - IP protocol number 51

Configuring NAT-Traversal

To configure NAT-T for site-to-site VPN:

1. Open the **Gateway Properties** of a gateway that has IPsec VPN enabled.
2. Select **IPsec VPN** > **VPN Advanced.**
3. Make sure that **Support NAT traversal (applies to Remote Access and Site to Site connections)** is selected.

   NAT-Traversal is enabled by default when a NAT device is detected.

Advanced NAT-T Configuration

These variables are defined for each gateway and control NAT-T for site-to-site VPN:

Item	Description	Default Value
offer_nat_t_initator	Initiator sends NAT-T traffic	true
offer_nat_t_responder_for_known_gw	Responder accepts NAT-T traffic from known gateways	true
force_nat_t	Force NAT-T even if there is no NAT-T device	false

The variables can be viewed or changed in GuiDBedit under:

TABLE > **Network Objects** > **network_objects** > **<gateway_object>** > **VPN**.

VPN Command Line Interface

In This Appendix

VPN Commands

These commands relate to VPN and are also documented in the *R80.10 Command Line Interface Reference Guide* http://supportcontent.checkpoint.com/documentation_download?ID=TBD.

VPN Command Line interface

Command	Description
VPN	This command and subcommands are used for working with various aspects of VPN. VPN commands executed on the command line generate status information regarding VPN processes, or are used to stop and start specific VPN services.
vpn compreset	This command resets the compression/decompression statistics to zero.
vpn compstat	This command displays compression/decompression statistics.
vpn crl_zap	This command is used to erase all Certificate Revocation Lists (CRLs) from the cache.
vpn crlview	This command retrieves the Certificate Revocation List (CRL) from various distribution points and displays it for the user.
vpn debug	This command instructs the VPN daemon to write debug messages to the log file: **$FWDIR/log/vpnd.elg.**
vpn drv	This command installs the VPN kernel (vpnk) and connects it to the Firewall kernel (fwk), attaching the VPN driver to the Firewall driver.
vpn export_p12	This command exports information contained in the network objects database and writes it in the PKCS#12 format to a file with the p12 extension.
vpn macutil	This command is related to Remote Access VPN, specifically Office mode, generating a MAC address per remote user. This command is relevant only when allocating IP addresses via DHCP.
vpn mep_refresh	This command causes all MEP tunnels to fail-back to the best available gateway, providing that backup stickiness has been configured.

Command	Description
vpn nssm_toplogy	This command generates and uploads a topology (in NSSM format) to a IPSO NSSM server for use by IPSO clients.
vpn overlap_encdom	This command displays all overlapping VPN domains. Some IP addresses might belong to two or more VPN domains. The command alerts for overlapping encryption domains if one or both of the following conditions exist: • The same VPN domain is defined for both Security Gateways • If the gateway has multiple interfaces, and one or more of the interfaces has the same IP address and netmask.
vpn sw_topology	This command downloads the topology for a SofaWare Security Gateway.
vpn ver	This command displays the VPN major version number and build number.
vpn tu	This command launches the TunnelUtil tool which is used to control VPN tunnels.