Chapter 5

Advanced VPN Concepts and Tunnel Monitoring

Solutions in this chapter:

- **Encryption Overview**
- **VPN Communities**
- **Policy-Based VPN**
- **Route-Based VPN**

- ☑ **Summary**
- ☑ **Solutions Fast Track**
- ☑ **Frequently Asked Questions**

Introduction

VPNs (virtual private networks) have emerged as a technology due to their ability to leverage an organization's existing infrastructure (including the Internet) to both create and augment existing communication links securely. Checkpoint VPNs are based on standard secure protocols defined in the Internet RFCs. This enables the creation of secure links between selected categories of network nodes such as the VPN-1 Power module. VPN implementations may be created as site-to -site VPNs to ensure secure links between gateways, or as remote access VPNs to ensure that the links between gateways and remote access clients are secured.

Check Point's VPN-1 Power is an integrated software solution that provides secure connectivity to corporate networks, remote and mobile users, branch offices and business partners on a wide range of open platforms and security appliances.

VPN-1 Power is composed of:

- VPN endpoints, including gateways, gateway clusters, and remote client software for mobile systems. These are the systems that negotiate the VPN link parameters.
- VPN trust entities such as the CheckPoint Internal Certificate Authority (ICA). ICA is a component of the VPN-1 Power suite and it is used in order to establish trust between SIC connections. It supports deployment from gateways, authenticating administrators, and third-entity servers. ICA supplies certificates for both internal gateways and remote access clients involved in the negotiation of a VPN link.
- VPN management tools such as SmartCenter Server and SmartDashboard make the implementation of VPNs using CheckPoint products relatively simple. SmartDashboard is the console used to manage the SmartCenter Server Management module. The SmartDashboard module includes the VPN Manager giving administrators the capability to define and deploy intranet and remote access VPNs across their organization.

Encryption Overview

Symmetric cryptographic systems use the same key for the encryption and decryption of data between the entities that are communicating. The material used to construct these keys needs to be exchanged securely. Information can be exchanged in a secure manner only if the key is held and used solely by the communicating entities and no other.

Internet Key Exchange (IKE) is used to allow both entities to produce the same symmetric key in parallel. The symmetric key then encrypts and decrypts the accepted IP packets that make up the bulk transfer of data between the VPN-1 Power peers. IKE constructs a VPN tunnel between the peers by authenticating the systems on both sides of the VPN tunnel and reaching an agreement on the encryption and integrity scheme to be used. A successful IKE negotiation results in a security association (SA) between the systems.

The accord resulting from exchanging keys and encryption methods needs to be performed in a secure manner. As such, IKE is comprised of two phases. IKE Phase I puts down the practicalities required for the second phase. Diffie-Hellman (DH) is used by the IKE protocol to exchange the material that the systems use to construct the symmetric keys. The Diffie-Hellman algorithm constructs an encryption key known as a "shared secret" from the private key of one entity and the public key of the system. Since the symmetric keys used in IPSec are derived from the DH key that is shared between the VPN peers, the symmetric keys are never in point of fact exchanged over the network.

IKE Overview

The IKE suite of protocols permits a pair of security gateways to:

- Dynamically establish a secure tunnel through which the security gateways are able to exchange tunnel and key information.
- Assemble user-level tunnels or Security Associations (SAs) that incorporate tunnel attribute negotiations and key management. SAs are able to be refreshed or concluded utilizing the same secure channel.

IKE is founded on the Oakley and SKEME key determination protocols. The ISAKMP framework for key exchange and security association establishment is used by VPN-1 and implemented by IKE to provide:

- Automatic key refreshment using a configurable timeout
- Support for public key infrastructure (PKI) authentication systems
- Anti-replay defense

IKE utilizes UDP port 500 to exchange IKE data across the security gateways. UDP port 500 packets are required to be permitted through any IP interface concerned in the connection of a security gateway peer.

Main Mode and Aggressive Mode

IKE Phase I negotiations are implemented in order to establish IKE SAs. The SAs shield the IKE phase II negotiations from an eavesdropping attack. IKE implements one of two possible modes in phase I negotiations, either main mode or aggressive mode. The selection of main or aggressive mode is a subject of substituting costs and benefits in either case. The primary makeup of the two modes is:

- Main mode
  - Is more secure because it protects the identities of the peers during negotiations
  - Provides superior proposal flexibility with more options than aggressive mode
  - Uses more resources and takes more time than aggressive mode because a greater number of messages are exchanged between peers
  - Exchanges six messages
- Aggressive mode
  - Exposes identities of the peers to eavesdropping, making it less secure than main mode
  - Is quicker than main mode because a smaller number of messages need to be exchanged between peers
  - Exchanges three messages

Renegotiating IKE and IPSec Lifetimes

IKE phase I uses more resources (especially the processor) than IKE phase II. In phase I, the Diffie-Hellman keys need to be created and the peers need to authenticate each time the setup occurs. As a consequence, IKE phase I is carried out less regularly than phase II. The IKE SA remains valid for only a definite period, following which the IKE SA needs to be renegotiated. An IPSec SA is legitimate for an even smaller period than phase I. As a result, numerous IKE phase II exchanges take place for each phase I exchange.

The timeframe between each IKE renegotiation is known as the lifetime. In most cases, a shorter lifetime will result in a more secure IPSec tunnel. The trade-off

is that the cost associated with the amount of processor intensive IKE negotiations is greater. By using a longer IKE lifetime, VPN connections may be brought up quicker. IKE phase I occurs once a day by default; IKE phase II occurs every hour by default (the time-out for each phase is configurable and may be changed). The IPSec lifetime is configurable based on the number of kilobytes that are transmitted using DBedit to edit the objects_5_0.c file. The pertinent properties are included within the community set:

- ike_p2_use_rekey_kbytes. Change from false (default) to true.
- ike_p2_rekey_kbytes. Modify to include the required rekeying value (default 50,000).

Perfect Forward Secrecy

The keys produced by peers during IKE phase II and used by IPSec are formulated from a series of random binary digits. These are exchanged among peers and are based on the DH key computed during IKE phase I negotiations.

The DH key is calculated once. This is then used multiple times by the IKE phase II negotiations. Because the keys that are used during IKE phase II are derived from the DH key calculated within the IKE phase I negotiations, a mathematical association is created between the DH keys. Consequently, reusing a single DH key will deteriorate the strength of any subsequent keys that are exchanged. This means that the compromise of a single key will make the compromise of all subsequent keys easier.

Perfect Forward Secrecy (PFS) covers the situation where the compromise of a current session key or long-term private key will not result in a compromise of previous or successive keys in cryptography. VPN-1 Power supports PFS with a PFS mode. Enabling PFS will result in a fresh DH key being constructed for the period of an IKE phase II negotiation, and being renewed for every subsequent key exchange. Since a new DH key is constructed for the duration of each IKE phase I negotiation, no dependency exists between these keys and those produced in subsequent IKE phase I negotiations.

Checkpoint recommends that you permit PFS in IKE phase II only in cases where the security requirements are large due to the increased overhead and latency. The DH group employed throughout PFS mode is configurable among groups 1, 2, 5, and 14, with group 2 (1042 bits) set as the default.

IP Compression

IP compression is a process that decreases the size of the data segment of the TCP/IP packet. This reduction can significantly improve performance on a VPN-1 device. IPSec as implemented in VPN-1 provides support for the Inflate/Deflate IP compression algorithm. Deflate is an elegant algorithm that adjusts how the compression of data is conducted based on the data contents. The choice of using IP compression is negotiated as a part of the IKE phase II negotiations. Although it can improve the efficiency of the VPN tunnel, IP compression is not enabled by default.

IP compression is essential for systems that use SecuRemote and SecureClient across with slow or high latency network links such as dialup modems. It can add compression in order to make the link seem faster. VPN-1 Power encryption scrambles TCP/IP packets in an irregular manner. The result is that encrypted data cannot be compressed and consequently bandwidth is reduced. Where IP compression is enabled, VPN-1 will compress the packets ahead of encryption. This effectively recovers the reduced bandwidth that can result from encryption.

IKE DoS Attacks

The IKE protocol necessitates that the receiving gateway allocate memory from the initial IKE phase I request packet that is received. The gateway replies, and accepts an additional packet, which is subsequently processed using the information collected from the initial packet.

An attacker can transmit numerous IKE initial packets that have a forged and invalid source IP address for each packet. The receiving gateway is required to reply to each of these packets, assigning memory for each new initial packet. This can devour all resources of the CPU preventing additional connections from being allocated to legitimate users.

The attacker transferring IKE packets can spoof a host that is permitted to initiate IKE negotiations. This is referred to as an identified source. The attacker can also spoof an IP address that is unknown to the receiving gateway (such as a SecuRemote or SecureClient, or VPN-1 Power gateway that uses a dynamic IP address). This is referred to as an unidentified source.

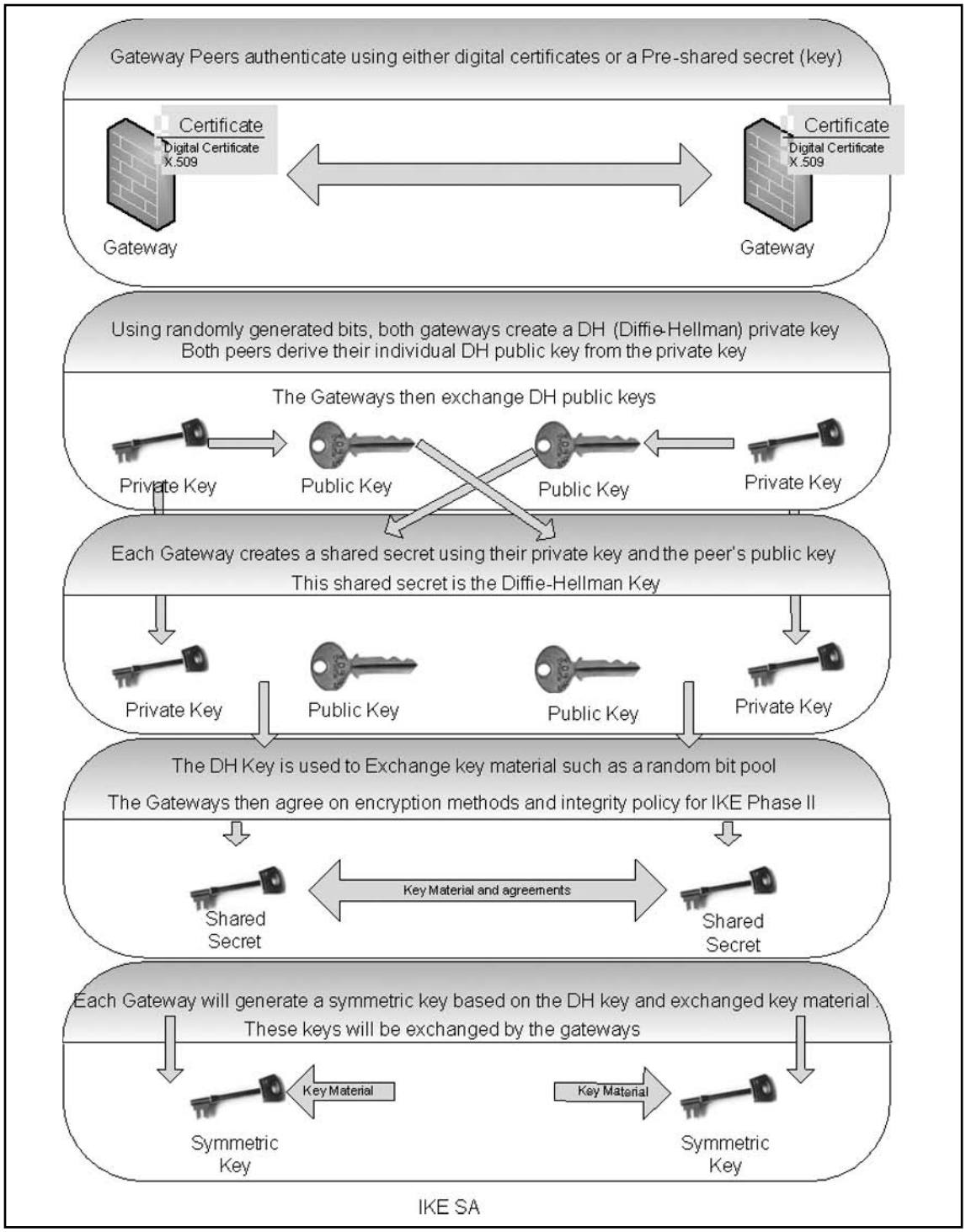
IKE Phase I

During IKE phase I:

- The peers authenticate using either certificates or via a preshared secret. Other authentication methods are accessible if one of the peers is a remote access client.
- A Diffie-Hellman key is produced. The makeup of the Diffie-Hellman protocol results in each peer being able to autonomously create the shared secret. The shared secret is a key that is known only to the peers in the negotiation.
- *Key material*, which is composed of random bits and other mathematical data, is sent with a concurrence between the peers as to the methods that the IKE phase II negotiation will use, which are exchanged among the peers.

The generation of the Diffie Hellman Key is slow and uses a lot of resources, causing degradation in performance. The result of IKE phase I is an IKE SA. This is an agreement on keys and methods that will be used in IKE phase II. Figure 5.1 illustrates the process that occurs throughout IKE phase I. It does not inevitably reproduce the actual order of events due to a variety of real-world occurrences (such as packet loss and retransmits).

Figure 5.1 IKE Phase I (IKE Gateway Exchange)



IPSEC Phase II

IKE phase II is encrypted based on the keys and methods agreed upon during the IKE phase I negotiation. The key material exchanged through IKE phase II is used for constructing the IPSec keys. The conclusion of phase II negotiations results is the IPSec Security Association (SA) being created. The IPSec SA is an agreement on the keys to be used and methods that are to be implemented in the IPSec communications. IPSec takes place based on the keys and methods agreed upon in the IKE phase II negotiations (see Figure 5.2).

Once the IPSec keys have been produced and exchanged by the gateways, the systems can begin the transfer of encrypted data (see Figure 5.3).

Figure 5.2 IKE Phase II Negotiations

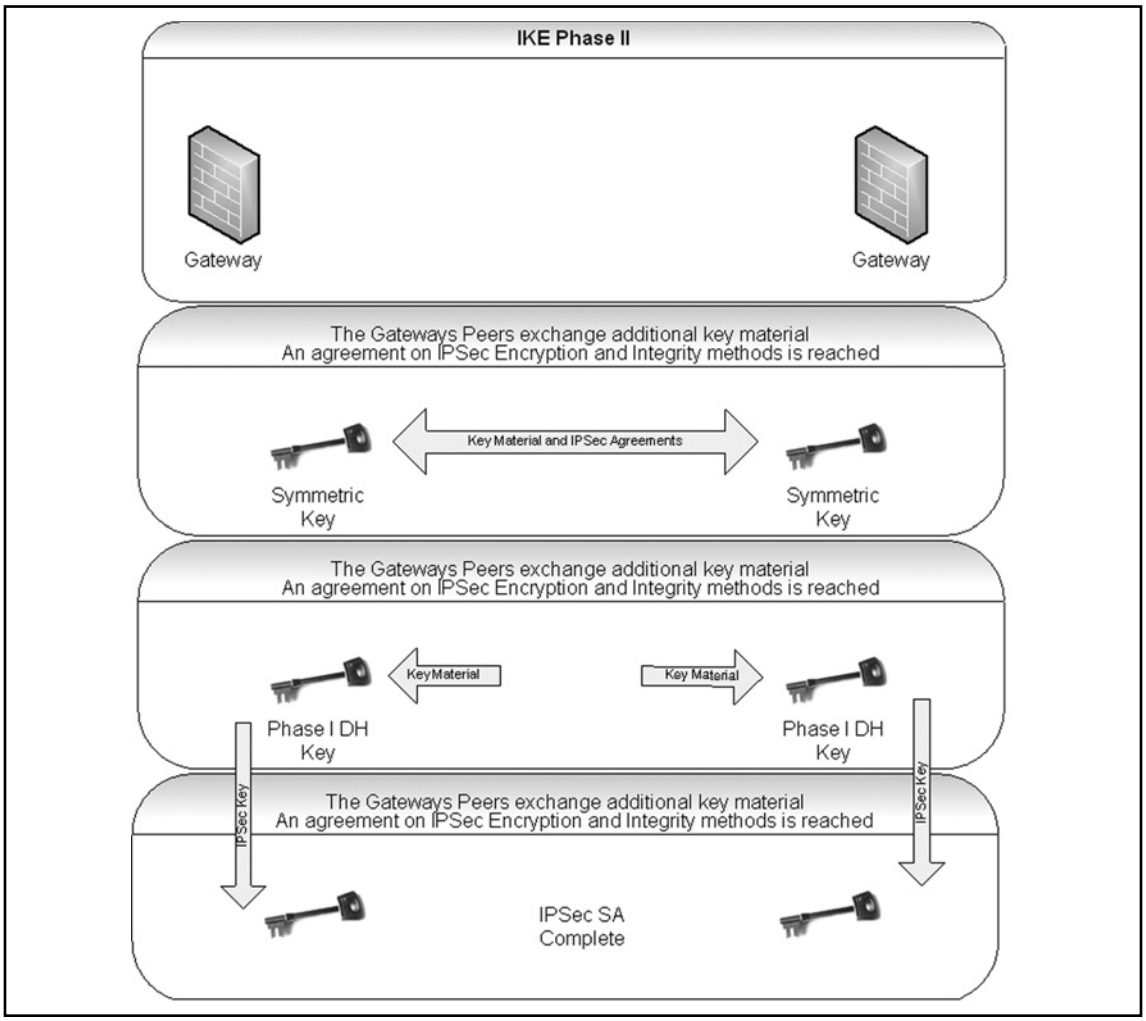
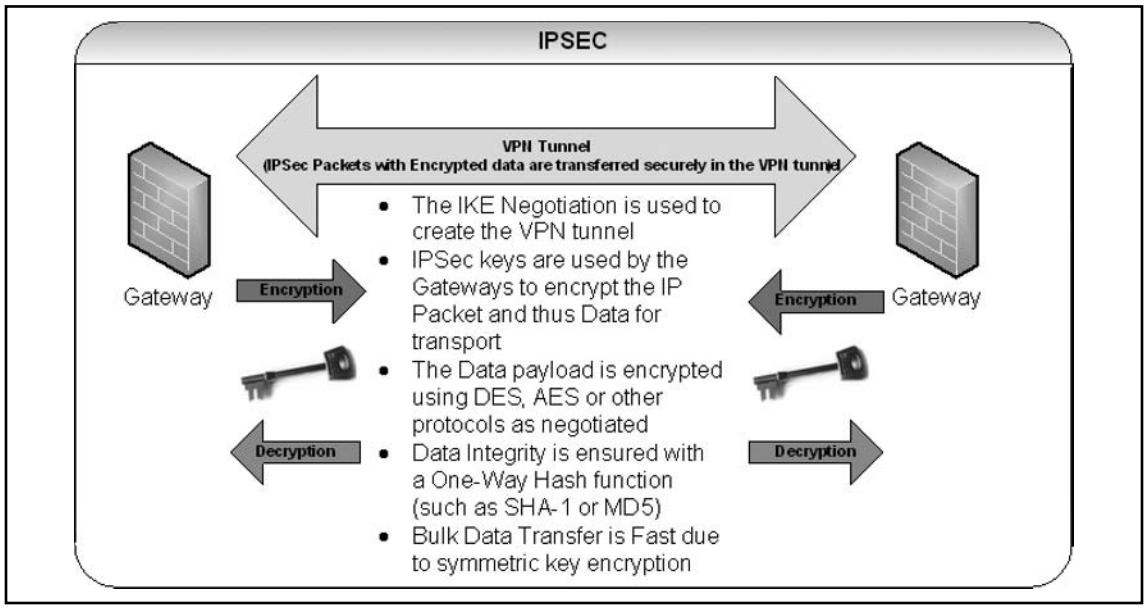


Figure 5.3 IPSec and the VPN Tunnel



Configuring Advanced IKE Properties

IKE is configured in two places:

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

1. On the VPN Community Network Object VPN Properties page, select:
   - Encryption methods used by IKE phase I and II
   - Integrity methods used by IKE phase I and II
2. On the Advanced Settings > Advanced VPN Properties page, select:
   - The Diffie-Hellman group to use
   - How often to renegotiate the IKE Security Associations
   - Whether the system will use aggressive mode (the default is to use Main mode)
   - Whether Perfect Forward Secrecy will be used, and which Diffie-Hellman group would require it.

- The frequency at which the IPSec security associations are to be renegotiated
- The option to use Support IP compression

3. On the Gateway Network Object
4. On the VPN Advanced page, deselect Support Key exchange for subnets if the SA is to be calculated for each host or peer. The option to support key exchange for subnets is the default.

IKE Policies

The IKE policy defines the combination of security parameters that are to be used during the IKE SA negotiation. IKE policies are configured on the communicating security gateway peers with the requirement that there has to be at minimum one policy on the local peer that matches at least one policy on the remote peer. If this is not the case the two peers are not able to negotiate an IKE SA successfully. Without an IKE SA no data transfer can occur between the peers.

IKE policies are global to the system. Each IPSec tunnel uses the same set of policies when negotiating IKE SAs. The agreed-on IKE SA between the local system and a remote security gateway may fluctuate due to a dependence on the IKE policies used by each remote peer. The initial set of IKE policies the peer uses is always the same and independent of the peer with which the VPN-1 system is negotiating.

During negotiation, VPN-1 may skip IKE policies that require parameters that are not configured or available for the remote security gateway with which the IKE SA is being negotiated.

It is possible to define multiple IKE policies, with each policy having a different combination of security parameters. A default IKE policy that contains default values for every policy parameter is possible. This policy is used only when IKE policies are not configured and IKE is required.

The following sections describe each of the parameters contained in an IKE policy.

Priority

Priority allows more secure policies to be given preference throughout the negotiation process. During IKE negotiation all policies are scanned, one at a time, starting from the highest priority policy and ending with the lowest priority policy. The first policy that the peer security gateway accepts is used for that IKE session. This procedure is repeated for every IKE session that needs to be established.

Encryption

A specific encryption transform can be applied to each IKE policy.

Hash Function

An individual hash function can be specified for an IKE policy.

IKE also uses an authentication algorithm during IKE exchanges. This authentication algorithm is automatically set to the HMAC version of the specified hash algorithm. Therefore, you cannot have the hash function set to MD5 and authentication algorithm set to HMAC-SHA.

Authentication Mode

As part of the IKE protocol, one security gateway needs to authenticate the other security gateway to make sure that the IKE SA is established with the intended entity.

Digital Certificates (Using RSA Algorithms)

For digital certificate authentication, an initiator signs message interchange data using his private key and a responder uses the initiator's public key to verify this signature. Classically, the public key is exchanged via messages containing an X.509v3 certificate that provides a level of assurance that the peer's identity as represented in the certificate is associated with a particular public key.

Preshared Keys

With preshared key authentication mode, the same secret string needs to be configured on both security gateways before the gateways can authenticate each other. It is not advisable to share a preshared key among multiple pairs of security gateways as this will reduce the security level of all gateways using this key.

Diffie-Hellman Group

An IKE policy must specify which Diffie-Hellmann group is to be used during the symmetric key generation phase of IKE.

Lifetime

Similar to a user SA, an IKE SA does not (and should not) last ad infinitum. Consequently, VPN-1 gives the capability to specify a lifetime parameter for an IKE policy. The timer for the lifetime parameter begins when the IKE SA is established using IKE.

IKE SA Negotiation

As the initiator of an IKE SA, VPN-1 sends its IKE policies to the remote peer. If the peer has an IKE policy that matches the encryption, hash, authentication method, and Diffie-Hellmann group settings, the peer returns the matching policy. The peers use the lesser lifetime setting as the IKE SA lifetime. In the event that no match is established, the IKE SA will not succeed, and a log entry is constructed.

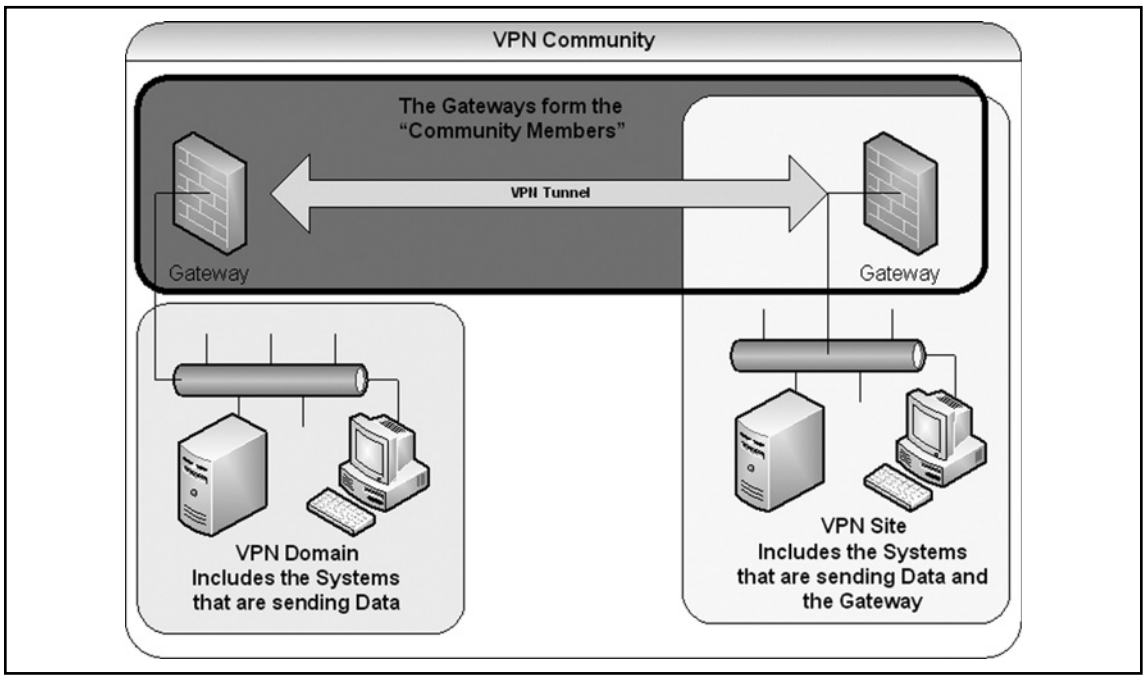
When acting as the responder to an IKE negotiation, VPN-1 receives all IKE policies from the remote security gateway. VPN-1 will then examine its own list of IKE policies to confirm whether a matching policy is present, beginning from the highest priority down. If it finds a match, that policy can be successfully negotiated. The SA lifetime is negotiated to the lesser of the two lifetimes, and failures are logged.

VPN Communities

Creating VPN tunnels between gateways is simplified using the configuration of the VPN communities feature. A VPN community is a collection of VPN-enabled gateways and peers that are able to communicate using a VPN tunnel. In order to understand VPN communities, several terms need to be defined:

- **VPN community member** refers to the gateway that resides at one end of a VPN tunnel (see Figure 5.4).
- **VPN domain** refers to the hosts behind the gateway. The VPN domain can be the entire network that lies within the protected segment of the gateway or just a segment of that network.
- **VPN site** includes the community member plus VPN domain; an archetypal VPN site would be the branch office of a corporation with multiple locations.
- **VPN community** refers to the collection of VPN tunnels/links and their attributes.
- **Domain-based VPN** covers the process of routing VPN traffic based on the encryption domain behind each gateway in the community. A star community permits satellite gateways to communicate with all others through center gateways.
- **Route-based VPN** occurs when traffic is routed within the VPN community based on the static or dynamic routing information that is configured on the operating systems of the gateways.

Figure 5.4 Terminology Used with Checkpoint VPNs



A Checkpoint SmartCenter Server can manage multiple VPN communities allowing communities to be created and organized according to their particular needs.

Remote Access Community

A Remote Access Community is a category of VPN community created particularly for users that typically work from remote locations, defined as those places that are outside of the corporate LAN. These kinds of community make certain secure communication occurs between users and the corporate LAN.

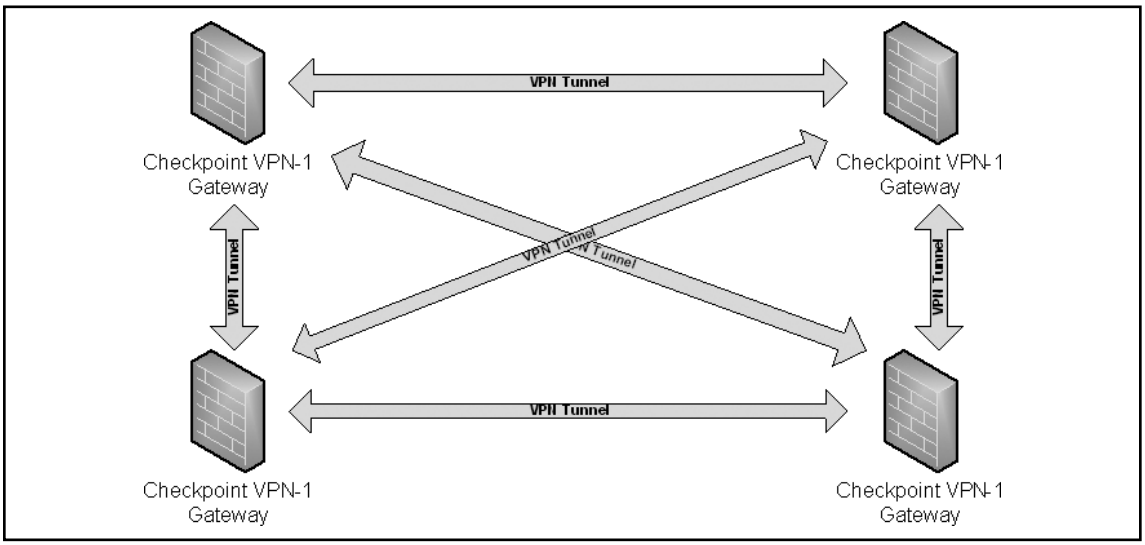
Remote Access Community Changes

Defining services in the clear in the community (available in gateway-to-gateway communities) is not supported if one of the internally managed members is an earlier version than NG FP3.

Mesh Topology

A mesh topography is a VPN community in which a VPN site can construct a VPN tunnel with any other VPN site in the community (see Figure 5.5).

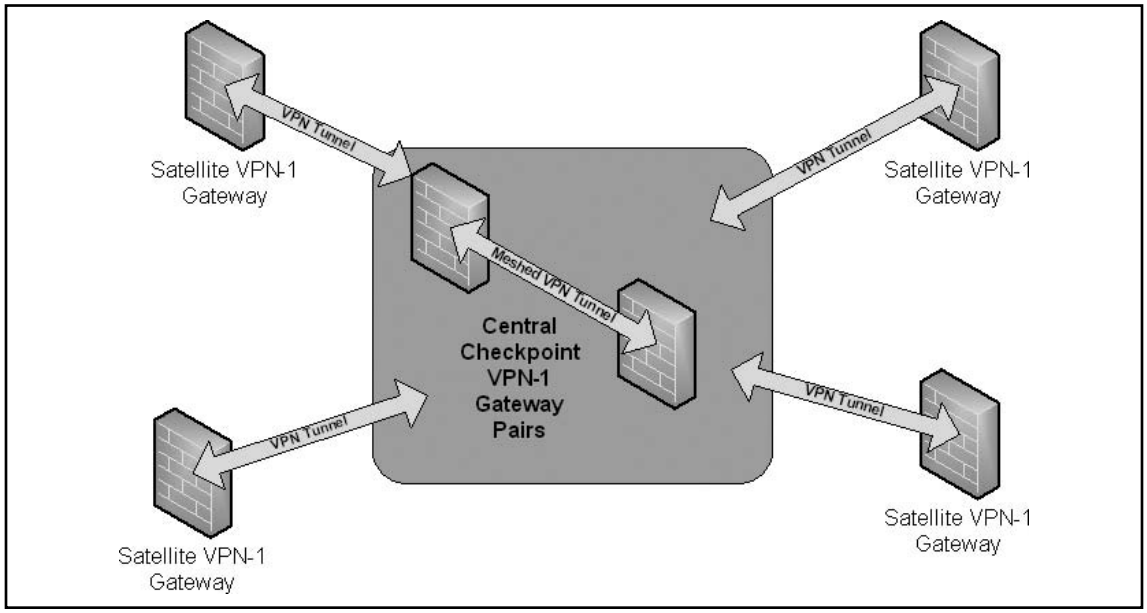
Figure 5.5 Checkpoint VPN-1 Basic Mesh Community



Star Topology

A star topography is a VPN community consisting of central or hubs gateways that are connected with satellite or spokes gateways (see Figure 5.6). A satellite system can create a tunnel only with other sites whose gateways are defined as central in this kind of community.

Figure 5.6 Checkpoint VPN-1 Star VPN Community



A satellite gateway is unable to generate a VPN tunnel with a gateway that is also defined as a satellite gateway. Central gateways can generate VPN tunnels with other central gateways only if the Mesh Center Gateways option has been selected on the Central Gateways tab for the Star Community Properties window on the VPN-1 Policy.

VPN Routing

VPN routing connections are subject to the equivalent access control rules as any other connection. When VPN routing is appropriately configured but a Security Policy rule is also configured to not allow the connection, the connection will be dropped by VPN-1. For instance, if a gateway has a rule set to reject all SMTP traffic from inside the internal network to a network exterior to the VPN-1 host, if a peer gateway opens an SMTP connection on the blocked network with this gateway, the connection is still rejected.

If VPN routing is to be successful, at least one rule in the Security Policy rule base must cover traffic in both directions, inbound and outbound, and on the central gateway.

In order to be able to route traffic to a host behind a gateway, an encryption domain must be configured for that gateway. The configuration for VPN routing is executed either directly through SmartDashboard or by editing the VPN routing configuration files on the gateways.

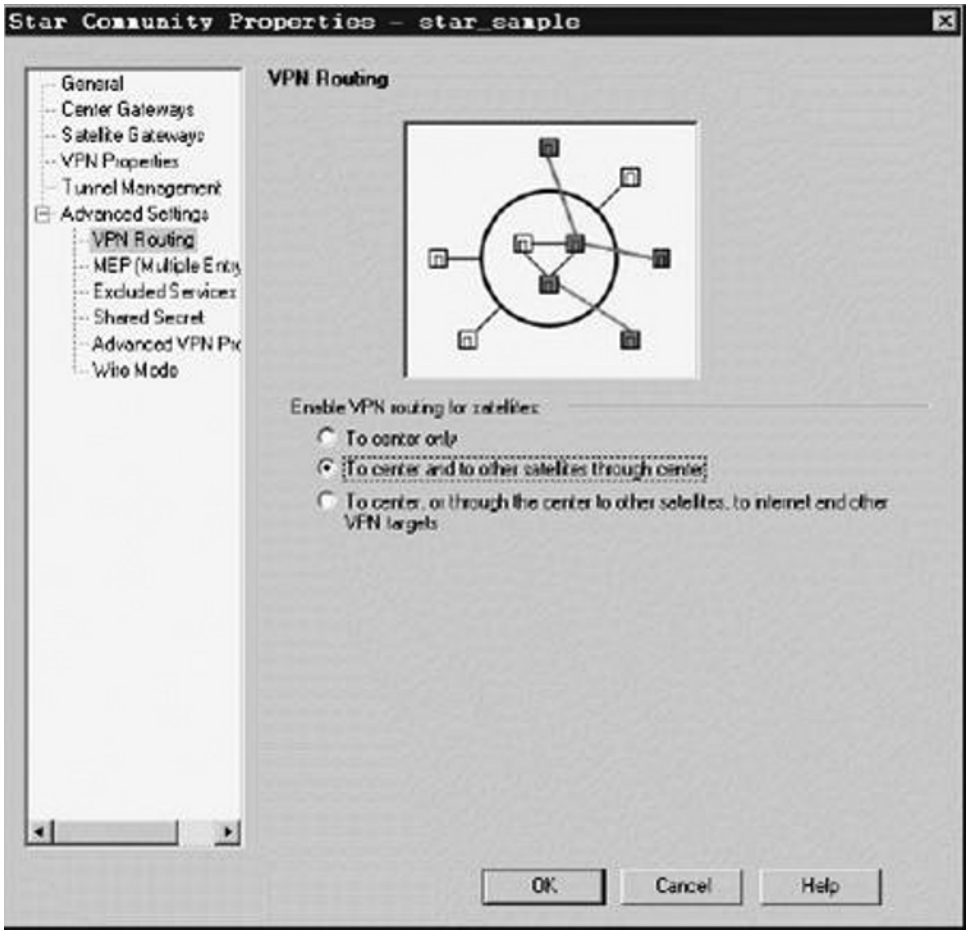
VPN routing cannot be configured between gateways that are not contained within a VPN community.

Configuring VPN Routing for Gateways via SmartDashboard

For uncomplicated hub or spoke networks and in circumstances where there is only a single hub, the easiest method is to configure a VPN star community in SmartDashboard (see Figure 5.7). The following steps demonstrate how this is done:

1. On the Star Community properties window, Central Gateways page, select the gateway that is to function as the hub.
2. On the Satellite Gateways page, select gateways as the spokes, or satellites.

Figure 5.7 Configuring a Satellite Community Using a Star Topography



3. On the VPN Routing page, select the Enable VPN routing for satellites option and then select one of the following options:
   - To center and to other satellites through center. This allows connectivity between the gateways, for example if the spoke gateways are DAIP gateways, and the hub is a 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 gateways as well as the ability to inspect all communication passing through the Hub to the Internet.
4. Create a suitable access control rule in the Security Policy Rule Base remembering that one rule must cover traffic in both directions.
5. NAT the satellite gateways on the hub if the hub is used to route connections from satellites to the Internet.

NOTE

Disabling NAT in the community (available in Star and Meshed communities in the Advanced VPN Properties tab) is not supported if one of the internally managed members is an earlier version than NG FP3.

The two DAIP gateways can securely route communication through the gateway with the static IP address. In order to configure the VPN routing option to center and to other satellites through center with remote office branch office (RO_BO) gateways the following steps must be completed:

1. Create a network object in the VPN-1 policy editor that holds the VPN domains of all the VPN-1 RO_BO gateways managed by SmartLSM.
2. Edit the vpn_route.conf file, so that this network object appears in the Router column. This is the center gateway of the star community.
3. Install the vpn_route.conf file that was created on all LSM profiles that participate in the VPN community.

Route Injection

A Route Injection Mechanism (RIM) enables a VPN-1 Power gateway to use dynamic routing protocols to disseminate the encryption domain of a VPN-1

Power peer gateway to the internal network and then instigate back connections. RIM updates the local routing table of the VPN-1 Power gateway to incorporate the encryption domain of the VPN-1 Power peer when a VPN tunnel is formed.

It is only possible to enable a RIM when permanent tunnels are configured for the community. Permanent tunnels are kept alive by tunnel test packets. This tunnel will be considered and marked as being down if it fails to reply to the test packet. Consequently, the RIM will delete the route to the failed link from the local routing table, triggering the neighboring dynamic routing enabled devices to update their routing information. This will redirect all traffic destined to travel across the VPN tunnel to a predefined alternative path.

There are two possible methods to configure RIM:

- **Automatic RIM**. RIM can automatically inject the route to the encryption domain of the peer gateways.
- **A Custom Script**. A script can be used to specify tasks that RIM will execute according to particular needs.

Route injection can be integrated with MEP functionality in order to route return packets back through the same MEP gateway. With SecurePlatform installed on the gateway or if the operating system is IPSO or Linux, automatic RIM can be enabled using the GUI. A custom script can be used on these systems but custom-written scripts are not required.

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 be established only between

Check Point 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.
- A specific gateway. Use this option to configure specific gateways to have permanent tunnels.
- A single VPN tunnel. This feature allows configuring specific tunnels between specific gateways as permanent.

Wire Mode

Wire Mode was intended to improve connectivity by permitting existing connections to fail over effectively by bypassing firewall enforcement. By definition, traffic within a VPN community is private and if correctly implemented, secure. In a lot of cases, the firewall and the rule on the firewall concerning VPN connections is redundant. The firewall can be bypassed by VPN connections by defining internal interfaces and communities as "trusted" using Wire Mode.

When a packet reaches a gateway, the gateway asks itself two questions about it:

- Is this information coming from a "trusted" source?
- Is this information going to a "trusted" destination?

If the answer to both questions is yes, the VPN-1 Stateful inspection will not be enforced at the gateway and the traffic between the trusted interfaces bypasses the firewall if the VPN community to which both gateways are associated is designated as Wire Mode enabled.

As stateful inspection does not occur, there is no possibility of packets being discarded. The VPN connection is no different from any other connection along a dedicated wire. This is the meaning of Wire Mode. With stateful inspection not being conducted, the dynamic routing protocols that do not endure state verification from a nonwire mode configuration may currently be deployed. Wire mode thus assists Route-based VPNs.

Consider a scenario where:

- Gateway M1 and gateway M2 are both *wire mode enabled* and have trusted internal interfaces
- The community where gateway M1 and gateway M2 reside is wire mode enabled

- Host 1 residing behind gateway S1 is communicating through a VPN tunnel with Host 2 residing behind gateway M1
- MEP is configured for gateway M1 and gateway M2 with gateway M1 being the primary gateway and gateway M2 as the backup

In this scenario, if the gateway M1 fails, the connection fails over to the gateway M2. Any packet leaving Host 2 will be redirected by the router behind gateway M1 to gateway M2 given that gateway M2 is designated as the backup gateway. Without wire mode, stateful inspection would be enforced at gateway M2 and the connection is dropped since packets that come into a gateway whose session was initiated through a different gateway are regarded by VPN-1 as out-of-state packets. Given that gateway M2's internal interface is "trusted," and wire mode in enabled on the community, no stateful inspection is executed and gateway M2 will successfully maintain the connection without losing any information.

Consider a scenario where:

- Wire mode is enabled on Center gateway C without an internal trusted interface being specified
- The community is wire mode enabled
- Host 1 is residing behind Satellite gateway A and wants to open a connection through a VPN tunnel with Host 2, which is located behind Satellite gateway B.

In a Satellite community, Center gateways are used to route traffic between Satellite gateways within the community.

In this scenario, traffic from the Satellite gateways is only rerouted by gateway C and cannot pass through gateway C's firewall. Consequently, stateful inspection does not need to take place at gateway C. Given that wire mode is enabled, making them trusted on the community and on gateway C, stateful inspection is bypassed. Stateful inspection does take place on gateways A and B in this example.

In an alternate scenario:

- Gateway A belongs to Community 1
- Gateway B belongs to Community 2
- Gateway C belongs to Communities 1 and 2
- Wire mode is enabled on Center gateway C set without an internal trusted interface being specified

- Wire mode is enabled on both communities
- Host 1 resides behind Satellite gateway A, which wants to open a connection through a VPN tunnel with Host 2, which is behind Satellite gateway B

Wire mode can be enabled for routing VPN traffic involving two gateways that are not members of the same community. Because Gateway C is a member of both communities, it hence recognizes both communities as trusted. When Host 1 behind gateway A commences a connection to Host 2 behind gateway B, gateway C is used to route traffic between the two communities. As the traffic is not in reality entering gateway C, there is no need for stateful inspection to occur at that gateway. Stateful inspection does take place on gateways A and B.

PKI Solutions

X.509-based PKI solutions present the infrastructure that enables organizations to establish trust relationships connecting 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 since they can verify the CA's signature, and then can rely on the information in the certificate. The most important information in the certificate is the association of the entity with the public key.

IKE standards advocate the use of PKI in VPN environments where strong authentication is necessary. A VPN-1 Power module taking part in a VPN tunnel establishment must have an RSA key pair and a certificate issued by a trusted CA. The certificate holds details about the module's identity, its public key, CRL retrieval details, and is signed by the CA.

As soon as two entities attempt to establish a VPN tunnel, each system supplies its peer with random information signed by its private key and with the certificate that contains the public key. The certificate facilitates the establishment of a trust relationship linking the gateways. Each gateway uses the peer gateway's public key to confirm the source of the signed information and the CA's public key to confirm the certificate's authenticity. As a result, the corroborated certificate is used to authenticate the peer.

Every deployment of Check Point SmartCenter server includes an Internal Certificate Authority (ICA) that can issue VPN certificates for the VPN modules it controls. These VPN certificates simplify the creation of VPNs connecting the modules.

Difficulties can occur when integration with other PKI solutions is required; for instance:

- A VPN must be established with a VPN-1 Power module administered through an external SmartCenter server. For instance, the peer gateway belongs to another organization that makes use of Check Point products, and its certificate is signed by its own SmartCenter server's ICA.
- A VPN must be established with a non-Check Point VPN entity. In this instance, the peer's certificate is signed by a third-entity CA.
- An organization may settle on, using a third-entity CA to generate certificates for its VPN-1 Power modules.

PKI Deployments and VPN

Following are some sample CA deployments:

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

Policy-Based VPN

Common VPN routing scenarios can be configured using a VPN star community. Not all VPN routing configuration may be handled through SmartDashboard. VPN star or mesh routing between gateways can be also be configured by editing the configuration file $FWDIR\conf\vpn_route.conf.

VPN routing cannot be configured between gateways that do not belong to a VPN community.

vpn_route.conf

For further control above VPN routing, edit the *vpn_route.conf* file in the *conf* directory of the SmartCenter Server. The configuration file, *vpn_route.conf*, is a text file that contains the name of network objects. The format is defined by Destination, Next hop, Install on Gateway. It uses tabs to separate the elements.

Think about a simple VPN routing scenario consisting of a hub and two spokes where all systems are controlled by the same SmartCenter management Server, and all VPN-1 Power enforcement modules are members of the matching VPN community.

Only Telnet and FTP services are to be encrypted between the spokes and routed through the hub. Although this could be done easily by configuring a VPN star community, the same objective can be accomplished by editing vpn_route.conf (see Table 5.1):

Table 5.1 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 VPN-1 Power gateway enabled for VPN routing. Spoke_A_VPN_Dom is the name of the network object that represents Spoke A's encryption domain.

Route-Based VPN

The use of VPN Tunnel Interfaces (VTI) introduces a new method of configuring VPNs called Route-based VPN. This method is based on the concept that initiating a VTI between peer gateways is reminiscent of connecting them directly.

A VTI is an operating system level virtual interface that can be used as a gateway to the encryption domain of the peer gateway. Every VTI is associated with a single tunnel to a VPN-1 Power peer gateway. The tunnel itself with all its properties is defined by a VPN community connecting the two gateways. The peer gateway should also be configured with an analogous VTI. The native IP routing mechanism on each gateway can then direct traffic into the tunnel as it would for any other type of interface.

All traffic destined to the encryption domain of a peer gateway will be routed through the associated VTI. This infrastructure allows dynamic routing protocols to use VTIs. A dynamic routing protocol daemon running on the VPN-1 Power gateway

can swap routing information with a neighboring routing daemon running on the other end of an IPSec tunnel that looks as if it is a single hop away.

Route-based VPN is supported using SecurePlatform and Nokia IPSO 3.9 (and greater) platforms and can be implemented only between two gateways within the same community.

Virtual Tunnel Interfaces

A VPN Tunnel Interface is a virtual interface on a VPN-1 module that is linked with an existing VPN tunnel and is used by IP routing as a point-to-point interface directly associated to a VPN peer gateway (see Figure 5.8).

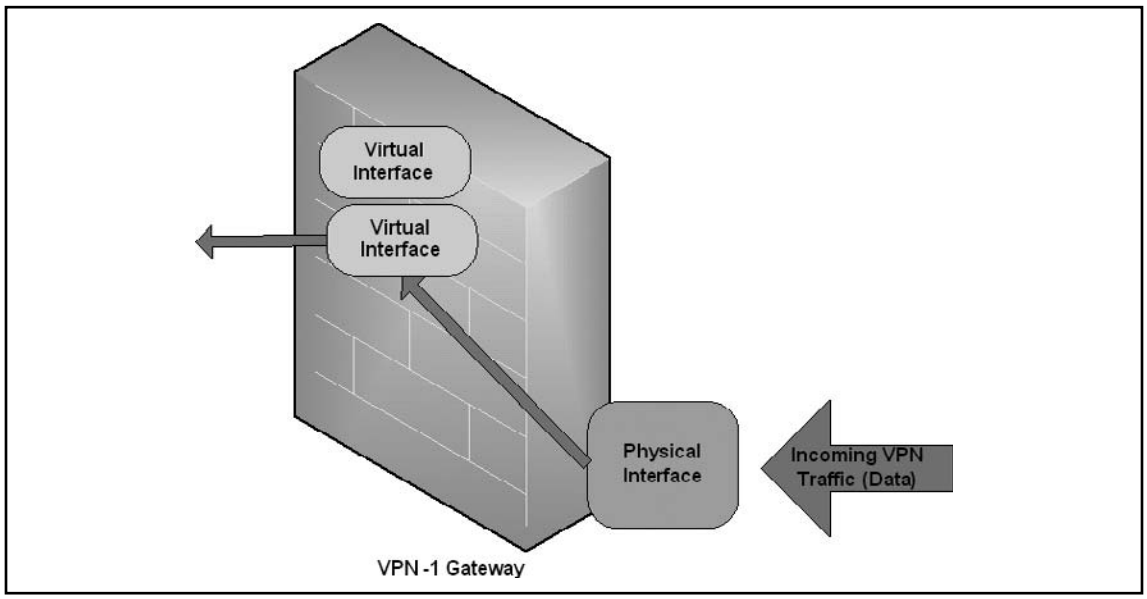
The VPN routing process of an outbound packet can be described as follows:

- An IP packet with destination address X is matched against the routing table.
- The routing table indicates that IP address X should be routed through a point-to-point link, which is the VPN Tunnel Interface that is related by way of peer gateway Y.
- VPN-1 kernel captures the packet as it goes into the virtual tunnel interface.
- The packet is encrypted using the appropriate IPSec Security Association parameters with peer gateway Y as defined in the VPN community, and the new packet receives the peer gateway Y's IP address as the destination IP.
- Based on the new destination IP, the packet is rerouted by VPN-1 into the physical interface, according to the proper routing table entry for Y's address.

The opposite is done for inbound packets:

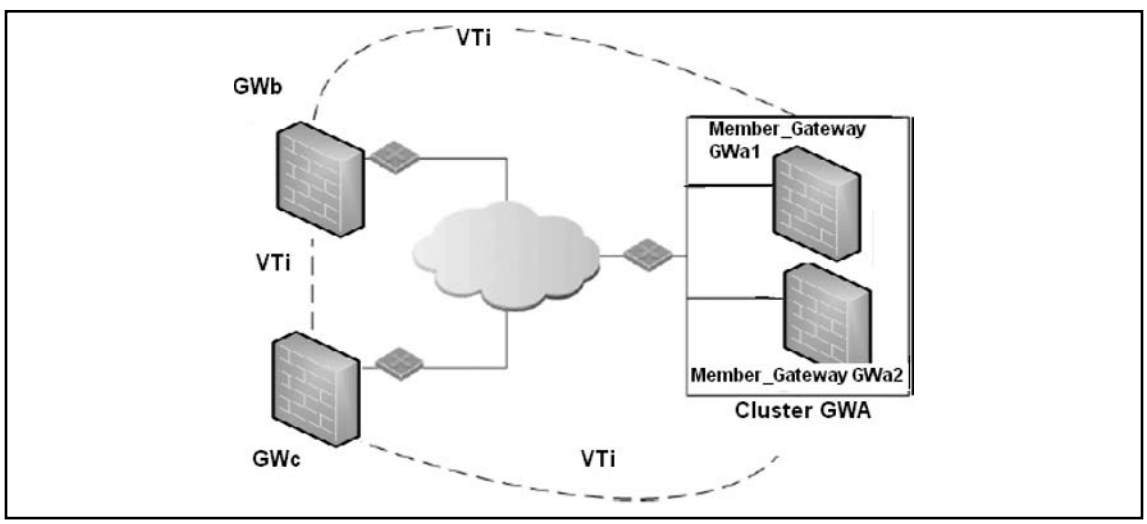
- An IPSec packet enters the system coming from gateway Y.
- VPN-1 intercepts the packet on the physical interface.
- VPN-1 identifies the originating VPN peer gateway.
- VPN-1 decapsulates the packet, and extracts the original IP packet.
- VPN-1 detects that a VPN Tunnel Interface exists for the peer VPN gateway, and reroutes the packet from the physical interface to the associated VPN Tunnel Interface.
- The packet enters the IP stack through the VPN Tunnel Interface.

Figure 5.8 Virtual Interface Routing



In a Route-based VPN, VTIs are created on the local gateway (see Figure 5.9). Each VTI is associated with a corresponding VTI on a remote VPN-1 Power peer. Traffic routed from the local gateway via the VTI is transferred encrypted to the associated VPN-1 Power peer gateway.

Figure 5.9 Route-Based VPNs



In a scenario demonstrated in Figure 5.9:

- 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 performs similar to a point-to-point interface directly linked to the remote VPN-1 Power peer. Traffic between network hosts is routed into the VPN tunnel using the IP routing mechanism of the operating system. Gateway objects are still necessary, as well as VPN communities and access control policies to describe which tunnels are accessible. Nevertheless, VPN encryption domains for each individual peer gateway are no longer required. The assessment as to whether or not to encrypt data is dependent on whether the traffic is routed through a virtual interface. The routing changes dynamically if a dynamic routing protocol such as OSPF or BGP is offered on the network.

Notes from the Underground...

Changes in Dynamic Routing

For NGX (R60) and above, the dynamic routing suite has been incorporated into SecurePlatform Pro. The administrator runs a daemon on the gateway to publish the changed routes to the network.

When a connection that originates on GWb is routed through a VTI to GWc (or a server 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:

- Configuring a static route on GWb that redirects packets intended 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

After excluding the IP addresses from a route-based VPN, it is still possible to have other connections encrypted to those addresses by using domain-based VPN definitions. An example would be when not passing on implied rules. The VTI may be configured in two ways:

- Numbered
- Unnumbered

Numbered VTI

If the VPN Tunnel Interface is numbered, the interface is assigned a local IP address and a remote IP address. The local IP address will be the source IP for the connections originating from the gateway and going through the VTI. VTIs may share an IP address but cannot use a previously existing IP address associated with a physical interface. Numbered interfaces are supported only by the SecurePlatform operating system.

Unnumbered VTI

If the VTI is unnumbered, local and remote IP addresses are not configured. Unnumbered VTIs must be assigned a proxy interface. The proxy interface is used as the source IP for outbound traffic. Unnumbered interfaces do away with the need to allocate and manage an IP address per interface. Unnumbered interfaces are supported only on the Nokia IPSO 3.9 platform and above.

Nokia IPSO interfaces may be physical or loopback.

Dynamic VPN Routing

VTIs allow the ability to use Dynamic Routing Protocols to exchange routing information between gateways. The Dynamic Routing Protocols that are supported include:

- BGP4
- OSPF
- RIPv1 (SecurePlatform Pro only)
- RIPv2 (SecurePlatform Pro only)

VPN Directional Match

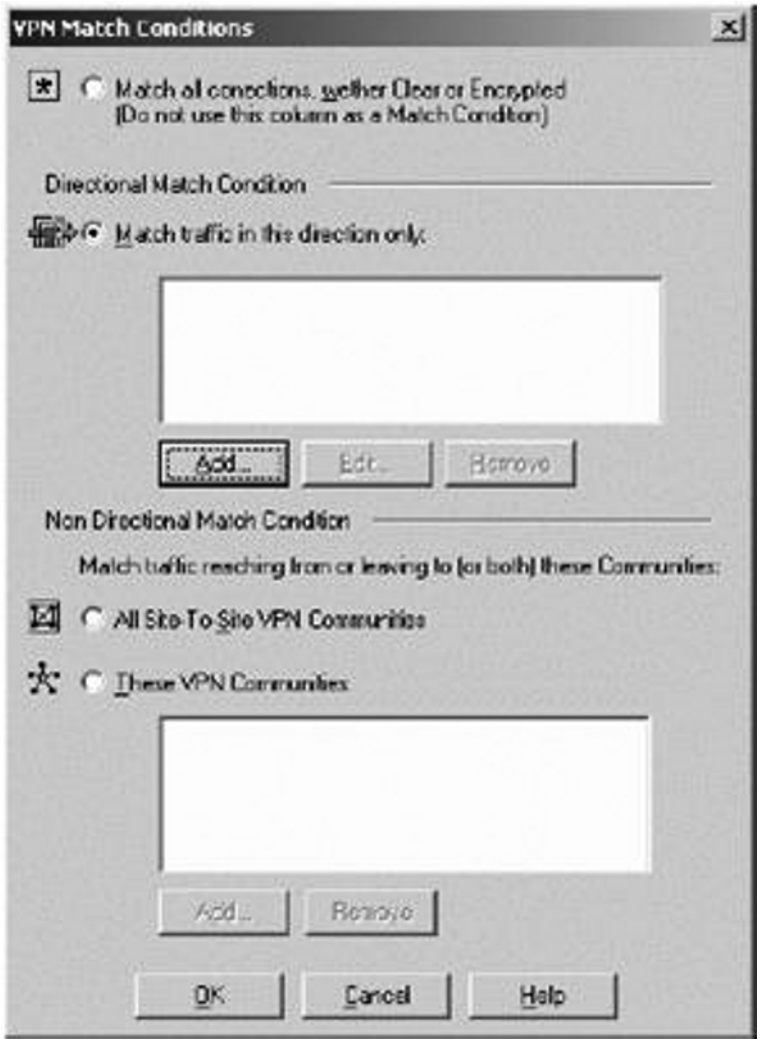
To configure Directional VPN within a community:

1. In Global Properties > VPN page > Advanced > Select Enable VPN Directional Match in VPN Column.
2. In the VPN column of the appropriate rule, right-click on the VPN community. From the pop-up menu, select Edit Cell….
   The VPN Match Conditions window opens.
3. Select Match traffic in this direction only, and click Add…
   The Directional VPN Match Condition window opens.
4. In the Match on traffic reaching the Gateway from: drop-down box, select the object for internal_clear. (the source).
5. In the Match on traffic leaving the Gateway to: box, select the relevant community object (the destination).
6. 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.
7. Click OK.

To configure Directional VPN between communities:

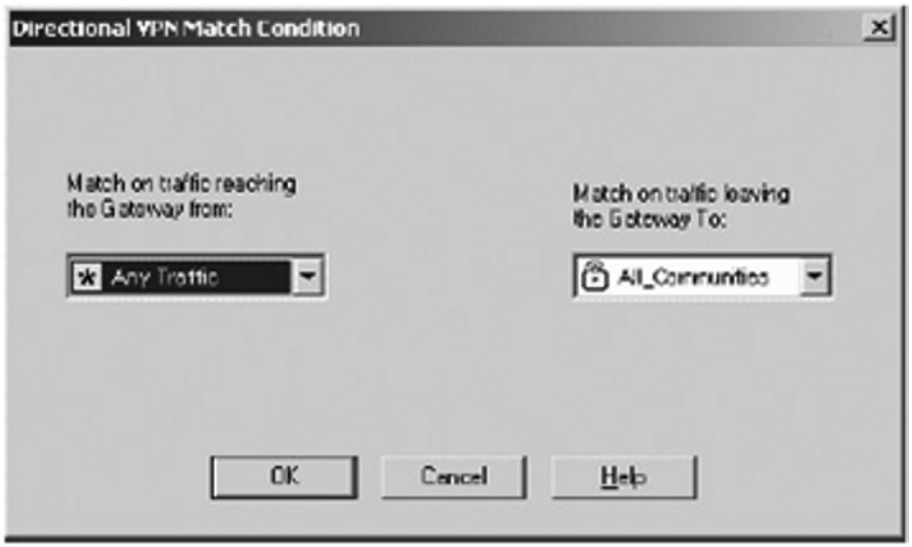
1. In Global Properties > VPN page > Advanced > Select Enable VPN Directional Match in VPN Column.
2. Right-click inside the VPN column of the right rule. From the pop-up menu, select Edit Cell or Add Direction.
   The VPN Match Conditions Window will open (see Figure 5.10).

Figure 5.10 VPN Match Conditions Window



3. Click Add. The Directional VPN Match Window will open (see Figure 5.11).

Figure 5.11 Directional VPN Match Window



4. From the drop-down box on the left of Figure 5.11, pick the source of the connection.
5. From the drop-down box on the right of Figure 5.11, pick the connection's destination.
6. Click OK.

Nokia Configuration

A Route-based VPN is supported only by SecurePlatform and Nokia IPSO 3.9 (or above) platforms and can be implemented only when two gateways within the same community are involved.

Local and remote IP addresses are not configured if the VTI is unnumbered. Unnumbered VTIs must be allocated a proxy interface. The proxy interface is used as the source IP for outbound traffic. Unnumbered interfaces do away with the need to allocate and manage an IP address per interface. Unnumbered interfaces are supported only by the Nokia IPSO 3.9 (or greater) platform.

Nokia IPSO interfaces may be physical or loopback.

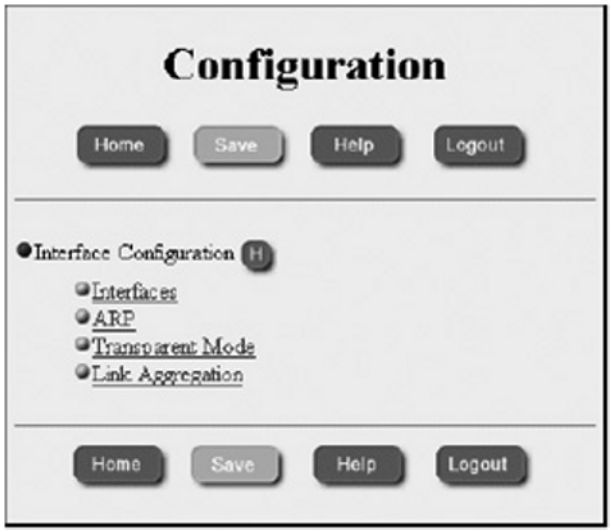
When a VTI connects a Nokia machine and a SecurePlatform host, a loopback interface has to be configured and defined in the Topology tab of the gateway. In Nokia Network Voyager:

1. Login and the window in Figure 5.12 will appear.

Figure 5.12 Initial Nokia Screen in Browser

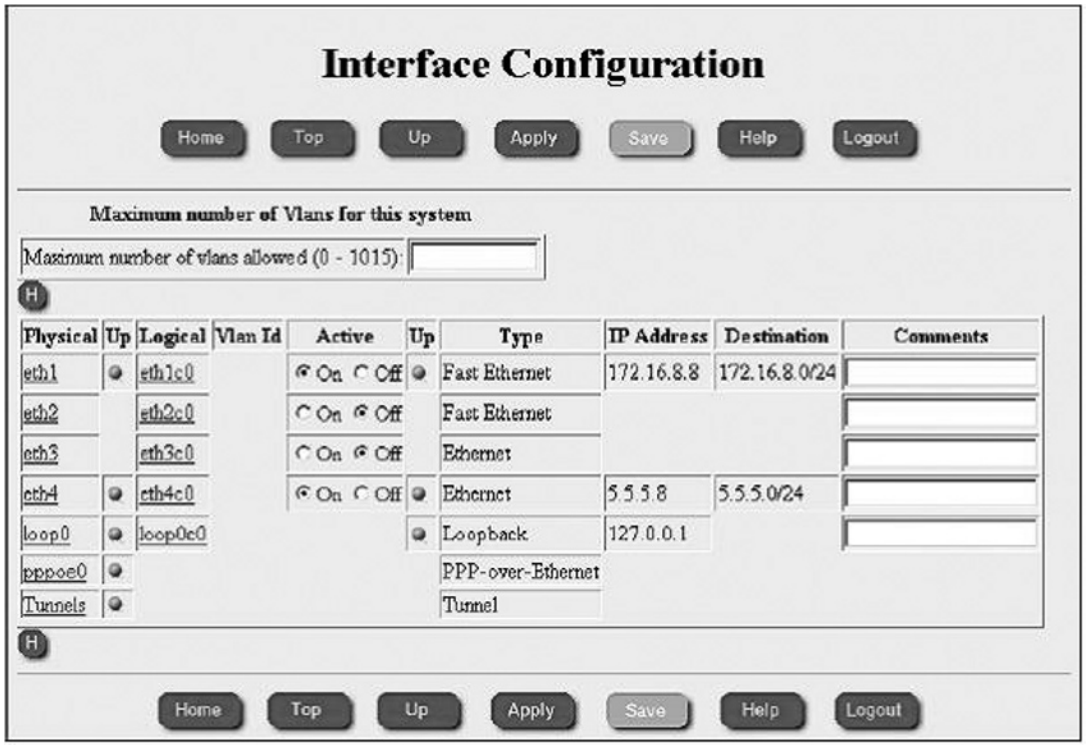
2. Click Interface Configuration.
3. On the Configuration page, click Interfaces (see Figure 5.13).

Figure 5.13 Nokia Configuration Window



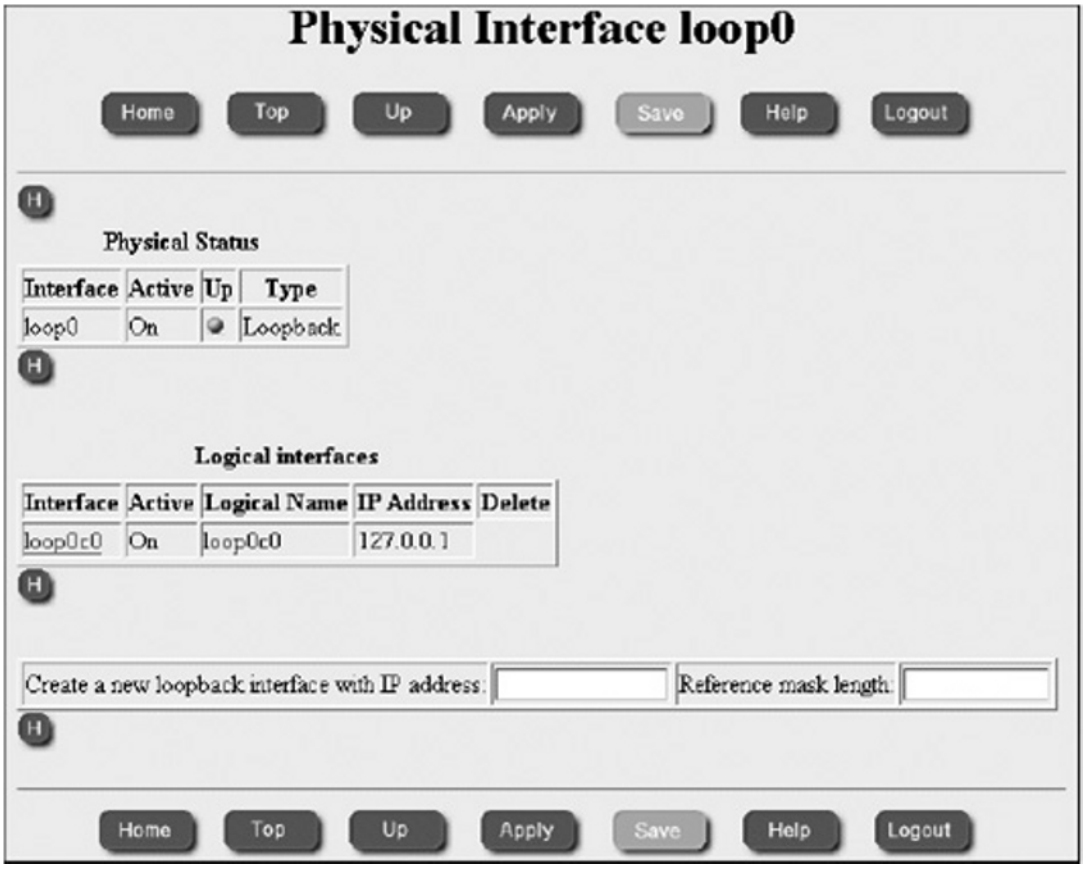
4. On the Interface Configuration page, click loop0 (see Figure 5.14).

Figure 5.14 Interface Configurations on a Nokia VPN-1



5. Enter an IP address in the Create a new loopback interface with IP address field and the value '30' in the Reference mask length field On the Physical Interface loop0 page (see Figure 5.15).

Figure 5.15 Loopback Configurations on a Nokia VPN-1



6. Click Apply.

   The Physical Interface loop0 page will refresh and displays the recently configured loopback interface.

7. Click Save.

Secure Platform Configuration

First Time Setup Using the Command Line is conducted after the installation from the CD has been completed and the computer has been rebooted. A first time setup is required in order to:

- Configure the network settings
- Apply the license
- Select which products will be installed
- Perform the SmartCenter initial setup, if selected

Perform the first time setup, as follows:

1. Run the sysconfig command from the console to configure SecurePlatform, using a text interface.
2. The command line setup wizard begins, and guides you through the first-time configuration.
3. Select "n" to proceed to the next menu, or "q" to exit the Wizard, and press Enter.
4. If you selected "n" and pressed Enter, the Network Configuration menu options are displayed. They are:
   - Host Name (Set/Show Host Name)
   - Domain Name (Set/Show Domain Name)
   - Domain Name Servers (Add/Remove/Show Domain Name Servers)
   - Network Connections (Add/Configure/Remove/Show Connection)
   - Routing (Set/Show Default Gateway)
5. You must configure the following:
   - The system's hostname
   - The domain name, and up to three DNS servers
   - The system's network interfaces
   - The default gateway for the system

6. Enter the preferred option number and press Enter.
   The Choose an action menu operation options will display.
7. Enter the preferred operation option number and press Enter.

HERE ARE SOME TIPS ON HOW TO GO BACK

- Select "e" and press Enter to return to the previous menu.
- Select "p" and press Enter to return to the previous menu, or select "q" and press Enter to exit the Wizard.

8. When you have completed the Network Configuration, choose "n" and press Enter to advance to the next menu, Time and Date Configuration. In the Time and Date Configuration menu it is possible to enter and set the current date and time and to set the time zone for the system.

Routing

This page enables you to manage the routing table on your device. It is possible to add a static or default route, or delete them.

NOTE

You cannot edit an existing route. To modify a specific route, delete it and create a new route in its place. Be careful not to delete a route that allows you connect to the device.

To delete a route, select the specific route checkbox and click Delete.

To configure routing, on the Routing Table page, click New. The Add Route drop-down box is displayed.

The options are:

- Route
- Default Route

To add a new route:

1. Select Route. The Add New Route page appears. Supply the following:
   - Destination IP Address
   - Destination Netmask
   - Interface (from the drop-down box)
   - Gateway
   - Metric
2. Click Apply.

To add a default route:

1. Select Default Route. The Add Default Route page appears. Supply the following:
   - Gateway
   - Metric
2. Click Apply.

Summary

This chapter described the process of connecting Check Point Firewall-1/VPN-1 using an IPSec Virtual Private Network (VPN). With the ability to share facilities and reduce costs when compared to traditional routed networks over dedicated facilities, VPNs have become an integral component of many organizations' infrastructures. When the ability to rapidly link enterprise offices, small and home offices, and mobile workers in both gateway peer connections and as mobile hosts is added this benefit only increases.

CheckPoint VPN-1 allows network administrators to customize their security and quality of service requirements as needed for specific applications. VPN-1 can scale to meet sudden demands, especially when provider-provisioned on shared infrastructure, and is able to reduce the operational expenditure associated with network support and facilities.

In this chapter we have discussed the benefits and capabilities of Checkpoint VPN-1 including IPSec and the associated IKE negotiation phases. Deployment of VPNs in the enterprise DMZ is done primarily through the following three models. All these models are supported using Checkpoint VPN-1:

- VPN termination at the edge router
- VPN termination at the corporate firewall
- VPN termination at a dedicated appliance

Each of these deployment models presents its own complexities that must be addressed for the VPN-1 topology to be successfully implemented.

Solutions Fast Track

Encryption Overview

- ☑ VPNs are able to afford Privacy, authenticity, data integrity.
- ☑ In IPSec, the Key exchange is public (asymmetric) and the session encryption is symmetric to provide the best level of performance.
- ☑ In IKE Phase I, the peers authenticate using either certificates or via a preshared secret. Other authentication methods are accessible if one of the peers is a remote access client. A Diffie-Hellman key is produced. The makeup

of the Diffie-Hellman protocol results in each peer being able to autonomously create the shared secret. The shared secret is a key that is known only to the peers in the negotiation.

- ☑ IKE Phase II is encrypted based on the keys and methods agreed upon during the IKE Phase I negotiation. The key material exchanged through IKE Phase II is used for constructing the IPSec keys.
- ☑ Perfect Forward Secrecy (PFS) covers the situation where the compromise of a current session key or long-term private key will not result in a compromise of previous or successive keys in cryptography.
- ☑ Double-check encryption rule properties on each gateway to ensure they are identical.
- ☑ Make sure key exchange rules (if any) are above your stealth rule.
- ☑ The four topologies supported by VPN-1 include mesh (fully and partially), star topology, hub-and-spoke, and remote access.
- ☑ The distinction between a star topology and a hub-and-spoke topology is that in a star topology, the branch or stub networks are not able to communicate with one another and are limited to communications with the central hub.
- ☑ There are three main choices for the symmetric key encryption schemes in IPSec with VPN-1. These are DES, 3DES, and AES.
- ☑ Message integrity is provided using the MD5, SHA-1, or HMAC hash algorithms. When using a HMAC algorithm, it is not possible to mix separate hash and HMAC algorithms (for instance by using MD5 and HMAC-SHA-1 together).
- ☑ Before an IPSec VPN tunnel can be established, the session parameters must be negotiated using Internet Key Exchange.
- ☑ IPSec security policies define the traffic permitted to enter the VPN tunnel.

Configuring SecuRemote/SecureClient VPNs

- ☑ SecuRemote/SecureClient can be used with dial-up or Ethernet adapters.
- ☑ Secure Domain Login is possible with SecuRemote/SecureClient.
- ☑ Several methods exist for automatically updating site topology.

VPN Tunnel Interfaces (VTI)

- ☑ The VTI may be configured in two ways: numbered or unnumbered.
- ☑ If the VPN Tunnel Interface is numbered, the interface is assigned a local IP address and a remote IP address.
- ☑ VTIs may share an IP address but cannot use a previously existing IP address associated with a physical interface.
- ☑ Numbered interfaces are supported only by the SecurePlatform operating system.
- ☑ If the VTI is unnumbered, local and remote IP addresses are not configured.
- ☑ Unnumbered VTIs must be assigned a proxy interface and do away with the need to allocate and manage an IP address per interface.
- ☑ The proxy interface is used as the source IP for outbound traffic.
- ☑ Unnumbered interfaces are supported only by the Nokia IPSO 3.9 platform and above.

Frequently Asked Questions

Q: Why can't I connect to a host in my peer's VPN domain?

A: This may not be allowed by policy. Ensure your policy allows the connection to and from the host and that traffic is allowed in both directions. Just because a VPN domain has been configured does not mean that you have set up the rules to allow the connection.

Q: What does it signify when "No response from peer: Scheme IKE" occurs in VPN-1's logs and you cannot initiate a VPN?

A: Confirm that fwd and isakmpd are both running on your peer gateway. Isakmpd listens on UDP port 500 to negotiate the IKE parameters. The netstat command may be used to check whether the port is listening. If the port is listening and you are still receiving the error, check that UDP 500 has not been blocked in the VPN-1 rules.

Q: What does it mean when you receive the error "No proposal chosen" in the logs and no VPN is initiated by the firewall?

A: The encryption rule properties diverge on the peer gateways. One gateway may support an encryption method that another doesn't. When VPN-1 is negotiating the IKE phase I parameters, it needs to have a common set of encryption methods with its peer gateway or the negotiation will fail. For instance, if one gateway supports only DES and the other supports only AES, then no VPN may be completed.

Q: Why is the VPN slow when I check Enable PFS?

A: Perfect Forward Secrecy (PFS) covers the situation where the compromise of a current session key or long-term private key will not result in a compromise of previous or successive keys in cryptography. Enabling PFS will result in a fresh DH key being constructed for the period of an IKE phase II negotiation, and being renewed for every subsequent key exchange. As new DH keys are constructed for the duration of each IKE phase I negotiation, no dependency exists between these keys and those produced in subsequent IKE phase I negotiations. This process increases the load on the VPN-1 gateway making the VPN slower. Checkpoint recommends implementing PFS only where there is a critical security need.

Q: How do I know that the key exchange is secure? Couldn't an attacker just sniff the key exchange and compromise the process?

A: IPSec key exchanges are based on a calculation of the symmetric keys, whereas the session is protected using the Diffie-Hellman (DH) protocol. In this negotiation, the keys are never sent over the network, but are created separately on each gateway.