SOLSOFT

HOW SECURE NETWORKS ARE MANAGED

SOLSOFT POLICY SERVER
WORKING WITH VPNS

VERSION 6.0.2

September 2004

Solsoft Policy Server

Working With VPNs

Software Version: 6.0.2

Revision 12

English Edition

The information contained in this document may be subject to modification without prior notice and Solsoft S.A. assumes no responsibility for any errors that may appear in it.

Revision 12, 2004/09/30
Manual reference: udoc-00335-en
Author: Ray Gallon

Contents

List of Tables

List of Figures

Preface

This guide presents features and functions of Solsoft Policy Server's VPN Module 1.0. Complete information on Solsoft Policy Server use and function is found in the *User Guide* and *Reference Manual*.

Audience

This guide is intended for network supervisors who will be responsible for implementing and maintaining network security policies using Solsoft Policy Server, and who wish to include configuration of IPsec based virtual private networks (VPNs) in their arsenal of security tools.

Purpose

This guide provides specific information on the concepts of IPsec VPNs and Solsoft's approach to configuring them with Solsoft Policy Server.

It also provides detailed procedures and tasks for implementing VPNs in Solsoft Policy Server, and a sample use case.

Related Documentation

Solsoft Policy Server Getting Started Guide

This document contains installation instructions, a list of new features in the current edition, and a guide to security-based policy management using Solsoft software.

Solsoft Policy Server User Guide

This document details the step-by-step procedures for using software to implement network security policies.

Solsoft Policy Server Reference Manual

This document contains detailed information regarding the functions of Solsoft software. It also includes details on the interface, error messages, warnings and logs.

Solsoft Policy Server Administration Guide

The Solsoft Policy Server *Administration Guide* explains how to use the Server Console of Solsoft Policy Server.

Solsoft Policy Server Language (NPL) Guide

(available on demand by writing to support@solsoft.com)
The Solsoft Policy Server *Language Guide* is a manual for advanced users. It details how to edit .NPL files to refine the filters generated by Solsoft Policy Server.

Typographical Conventions

The typographical conventions used in this manual are the following:

Standard Typographical Conventions Used in Text

Type	Significance
Normal	Text in this manual
`Courier`	Operator input and/or screen responses OR names of Solsoft Policy Server objects found in screen shots (e.g. `My_Firewall`).
`<Courier text in angle brackets>`	A keyboard press other than a character (e.g. `<enter>`)
Bold >Menu Trees	Used to indicate menu selections and hierarchical level in menu trees. Bold face type is also used in the document text to indicate: • Buttons • Labels in windows and views • Selections in tree lists.
Text in Italics	Used for emphasis, and to refer to other Solsoft Policy Server documents
Bold Italic text	Used to introduce a term defined in the glossary

Note: Windows and UNIX operating systems use different conventions for separating files in a directory structure. Where text in this document applies only to one operating system, the conventions of that operating system are followed. Where text applies equally to all operating systems, the UNIX convention of a forward slash ”/ “is used. Windows users should substitute a back slash “\ “in its place.

Special Typographical Conventions Used Only for Command Lines

Type	Significance
`Courier normal`	Operator input and/or screen responses
`Courier text in bold`	A defined **`keyword`** in a command syntax
`[Courier Text in square brackets]`	Optional items in a command line
`<Courier text in angle brackets>`	A token to be replaced by an actual value (in a command line) OR A keyboard press other than a character (e.g. `<enter>`).
`{Courier Text in curly braces}`	Curly braces are used in command lines to group elements which form a set of elements
`* (asterisk)`	The asterisk character, at the end of a list in a command line, indicates that the list can be expanded with any number of additional entries using the same format.
`| (vertical bar)`	The vertical bar character in a command line indicates an exclusive `or`

Note: Parentheses – `( )` – are used only when they form part of a command syntax and are thus required.

Chapter 1: Introduction to Solsoft Policy Server VPN Management

Graphical Management of IPsec Tunnels

Solsoft Policy Server's VPN Module 1.0 offers seamless integration of VPN management with other Solsoft Policy Server functions. It uses the same easy-to-use graphical interface and the same basic concepts to configure and manage tunnels.

With Solsoft Policy Server, you configure your tunnel once, and thereafter you only need to think about what services should pass through the tunnel.

Creating a tunnel is as simple as drawing a permission. In the Solsoft Security Designer, you draw a tunnel and configure it with a pre-defined tunnel policy - you can use policies provided by Solsoft, or you can create your own.

Once the tunnel is created and the group of IP addresses authorized to use it is defined (this is called a trust zone), all you need to do is define permissions between machines in the trust zone, just as you would any other Solsoft Policy Server permission. Solsoft Policy Server's Network Policy Engine™ (NPE) generates tunnels and filters according to the tunnel policy you have defined in advance, and the permissions you have defined inside the tunnel.

A Simple Use Case

Figure 1 shows a typical network for a small company with one branch. Each unit has a single internal network (labeled `Headquarters` and `Subdivision`) secured by an IPsec capable PEP (IPsec capability is indicated by the red key superimposed on the PEP icon). The only interconnection between the two units of the company is the Internet.

The company needs to send sensitive data between `Headquarters` and `Subdivision` but is worried about interception or tampering. The objective, thus, is to set up a tunnel between `PEP1` and `PEP2` that will allow secure, encrypted communication.

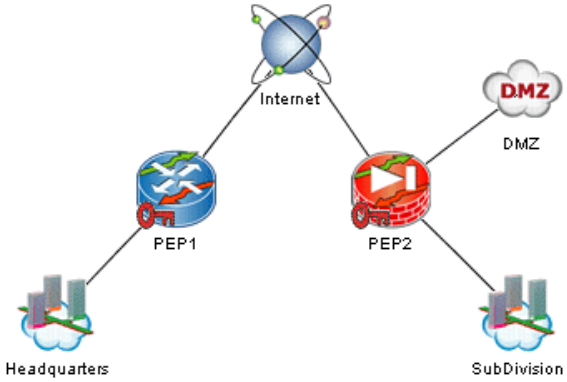


Figure 1: A Company And Branch Need Secure Communications

With Solsoft Policy Server you create a VPN to secure communications between these two networks using the following steps (refer to Figure 2):

1. Define a trust zone within which communication must be secure and assign objects to it (explained on page 59).
2. Assign a tunnel policy to the tunnel you are going to create (explained on page 63).
3. Draw a tunnel between `PEP1` and `PEP2` (explained on page 63).
4. Open permissions inside the tunnel (explained on page 75)
5. Compile and upload filters (explained in the *User Guide*).

.

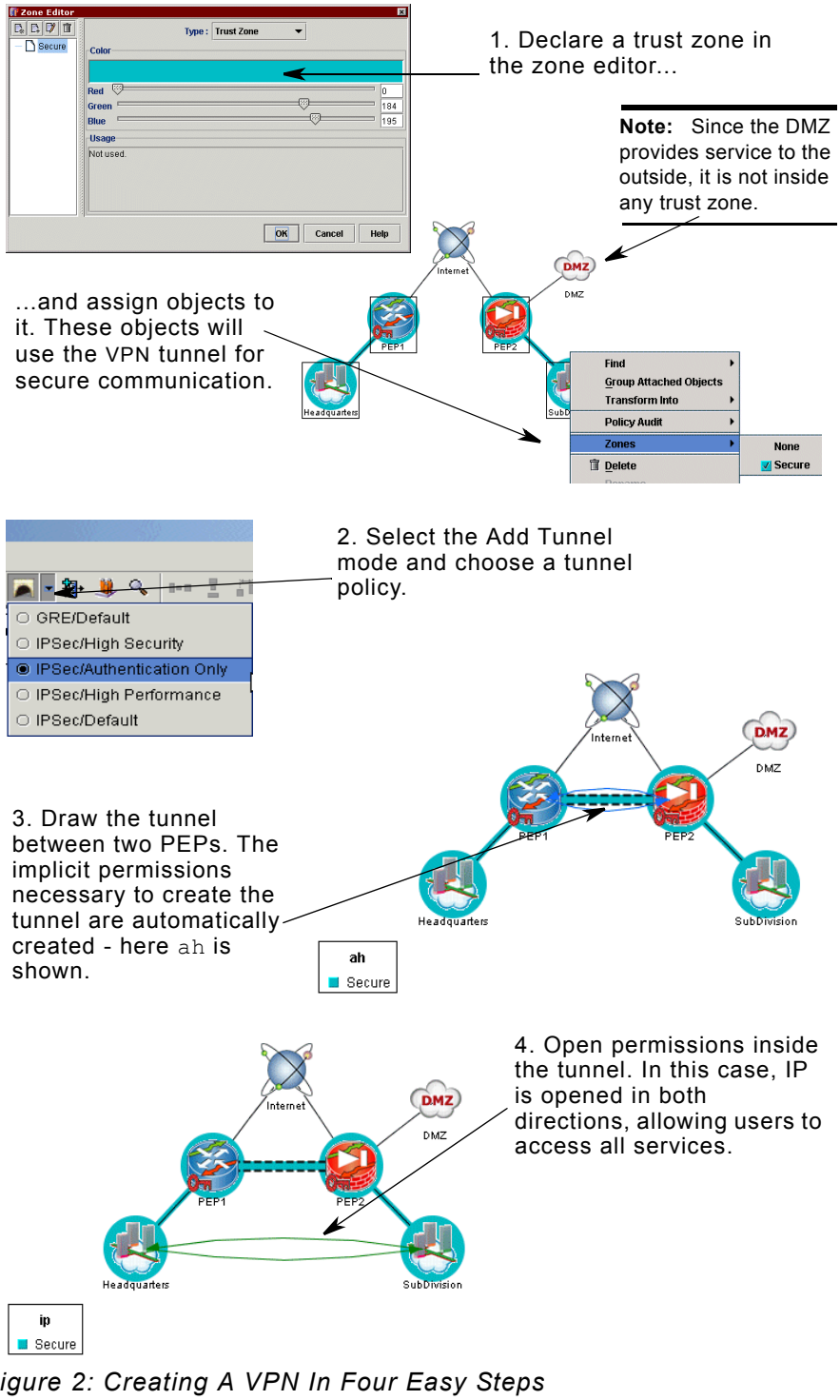


Figure 2: Creating A VPN In Four Easy Steps

You have now created a VPN - that's all there is to it!

What You Should Know Before Getting Started

License

You must have purchased and installed the special Solsoft Policy Server option for use with the VPN Module 1.0. If you do not have this license, you will not be able to configure tunnels on IPsec PEPs nor to take unmanaged tunnels in your global network into account in your logical network map.

Limitations

The following restrictions apply to Network Address Translation (NAT) when used together with a tunnel:

- By default, NAT rules are disabled inside a tunnel (except for static NAT rules) to allow use of internal addressing schemes. For static NAT rules, you *must* use NATed addresses.
- You cannot use NAT on a path which is used by `ah` or `esp`. (two protocols used for transport of IPsec packets).
- Device-specific limitations are given in Table 1 on page 47.

For more details, refer to "Using Tunnels and NAT Together" on page 47. For procedures, see "Verify Tunnel Properties" on page 70.

Chapter 2: IPsec Concepts

The Internet Protocol Security Standard (IPsec) is a set of open standards, working at the IP layer, to secure data communications by providing:

- Data confidentiality (privacy)
- Data authenticity:
  - Authentication
  - Integrity

IPsec can protect data by encapsulating packets in new, encrypted and authenticated packets, thus creating a “tunnel.” A secure zone which includes such tunnels is a virtual private network (VPN).

IPsec tunnels are brought up and down by a protocol which is called the Internet Key Exchange (IKE). This protocol also provides mutual authentication of the IPsec peers.

The resulting VPN offers the user a secure virtual connection through a network which the user considers to be otherwise untrustworthy.

This chapter gives an overview of standard IPsec and IKE concepts:

- Architecture
- Cryptography in IPsec VPNs
- IPsec
- IKE

Typical Use Cases

The two most common situations in which IPsec is used are:

- To establish a secured area within an internal network or between two distant internal networks (gateway to gateway).
- To allow remote access from a roaming client (no routing functions on the client side).

Definition of Terms and Concepts

Standard IPsec and IKE terms are defined in this chapter, along with the basic concepts of their operation. Additional terms and concepts associated with IPsec VPNs are defined in the context of Solsoft Policy Server’s VPN management approach. See *Definition of Terms* on page 38.

Architecture

Put simply, IPsec provides secure tunnels between two peers, such as two routers. You define which packets need protection (because they contain sensitive or secret information). These must always be sent via these secure tunnels, usually using an encryption scheme.

You define the parameters which should be used to protect these packets, by specifying characteristics of the tunnels. A tunnel is defined as secure when it provides the required combination of confidentiality, authentication and integrity for the user's data.

When the local IPsec peer sees a packet needing protection, it automatically sets up the appropriate secure tunnel and sends the packet through the tunnel to the remote peer.

Note: It is important to make a distinction between the use of the term *tunnel* in this document, and the *tunnel mode* of IPsec.

More precisely, a tunnel is a set of security associations (SA) that are established between two IPsec peers. These security associations define which protocols and algorithms should be applied to protected packets (e.g. encryption algorithm, authentication algorithm), and also specify the type of key to be used by the two peers.

Symmetric Configuration

Unlike Solsoft Policy Server permissions, which produce filters to be installed on a single device, IPsec tunnels require the symmetric configuration of two devices, one at either end of the tunnel. When you configure a tunnel in your Solsoft Policy Server logical network map (LNM), Solsoft Policy Server takes care of the calculations for you, independently of whether the IPsec devices are managed by Solsoft Policy Server or not.

What Is A Tunnel?

The word “tunnel” can be misleading, in that it may give the impression of a secure physical path through an unprotected net such as the Internet.

In reality, it functions *as if* there were a physical path. In Figure 3, `Network 1` wants to send a sensitive file to `Network 2` using the `ftp` protocol. The only physical connection they have is via the Internet. The solution is to protect the traffic so that when it is in the unprotected domain (Internet in Figure 3), no one can read or alter the data unless they are authorized to do so.

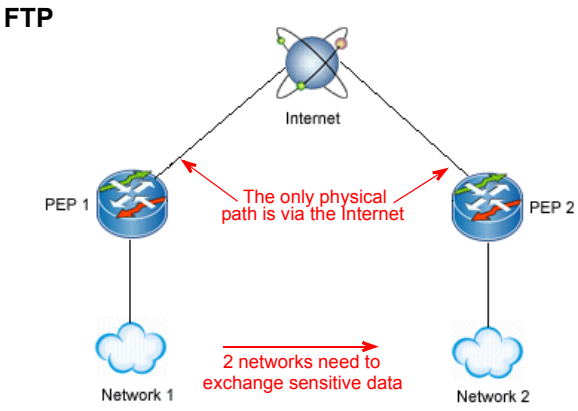


Figure 3: No Secure Path Exists

Protected traffic passes through the Internet or other unprotected network normally.

It is protected because it is:

- Encrypted.
- Authenticated by adding a “signature” to the packet.
- Encapsulated in a new IP Packet containing the encrypted data and the authentication data, which is transmitted using either AH or ESP, two specialized extensions to IP which perform many of the functions necessary to this process (see page 32 for details).

At the receiving end of the transmission, the packet is:

- Disencapsulated.
- Authenticated (the “signature” is verified).
- Decrypted, rendering it legible to an authorized user.

Whatever the nature of the communication (e.g. FTP, SMTP), it passes the untrusted zone as AH or ESP. Because they are encapsulated and, in the case of ESP, encrypted, these packets are difficult or impossible to read by unauthorized persons (see Figure 4).

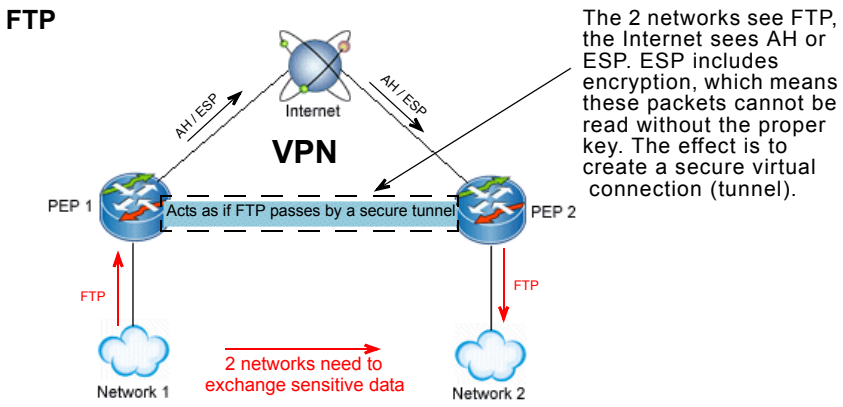


Figure 4: An IPsec VPN Makes Secure Communication Through Unprotected Networks Possible

The different parameters that determine the algorithms, and also which of the two protocols (AH or ESP) to use, are negotiated by IKE. Each tunnel has two IPsec SAs and two IKE SAs for each device, one for each direction of the traffic.

For authorized users, this process is transparent from end to end.

Note: A tunnel protects a specific set of IP addresses using specified services. It is not possible to set up a tunnel without specifying which traffic is protected.

Cryptography in IPsec VPNs

Cryptography is a discipline that develops methods for the confidential transmission of information.

In data communications, cryptography provides a suite of security services, notably:

- Data confidentiality (privacy) via encryption
- Data authenticity via the use of hash algorithms and other specialized processes. Authenticity has two components:
  - Authentication
  - Integrity

Encryption and Confidentiality

Encryption is a method of modifying the message content so that the encrypted message can only be read (“decrypted”) with the aid of an additional element of information (a “key”) which is known only to the intended recipient of the encrypted message.

The process of encryption is to apply an encryption algorithm to data or text, using an encryption key. This renders the information unreadable to anyone not possessing that key.

In order to read the information, you apply the complementary decryption algorithm (using the appropriate decryption key) to it. This renders the data or text back into its original form.

Data confidentiality results from data being protected so that it cannot be observed, read or interpreted by anyone not authorized to do so. Two types of encryption algorithms are used to ensure confidentiality:

- Symmetric algorithms (using a shared secret key)
- Asymmetric algorithms (using separate public and private keys)

Symmetric Algorithms

In this type of algorithm the encryption and decryption keys are identical. The key is applied to a clear text to create the encrypted text, and the same key is applied in reverse to decrypt it and restore the original.

This system is quite efficient, but requires that a key must be known, and shared, in advance by both parties: the sender and the receiver of the information. The key must be known only to the authorized parties, and must not fall into untrustworthy hands. This is not easy when the key is being shared between two distant sites.

Symmetric encryption algorithms used in IPsec include:

- RC5
- DES
- 3DES
- IDEA
- CAST
- Blowfish
- AES

Asymmetric Algorithms

In asymmetric encryption, the two keys are different and cannot be deduced or inferred, one from the other. In this scheme, one of these keys can be made public.

If the public key is used for encrypting the message, then anyone can cause a message to be encrypted, but only the possessor of the private key can decrypt and read it.

If the private key is used for encrypting the message, then anyone can decrypt and read the message. The interest of doing this is to authenticate the sender of the message - the encryption via the private key becomes a kind of “signature,” which is verified if decryption via the public key functions correctly. This second function corresponds more to authentication than to confidentiality.

The disadvantage of these schemes is that they are very slow. Thus, in practice, they are not used to encrypt data, but to encrypt and exchange in a secure manner a secret shared key which will be used for symmetric encryption.

Asymmetric encryption algorithms used in IPsec include:

- RSA
- El-Gamal

Authenticity

Data authenticity implies continuous access control for a communication. It includes two concepts:

- Authentication assures that the data is actually sent by the presumed sender.
- Data integrity verifies that data has not been altered during transmission.

It is quite common to see the word “authentication” used to mean both authentication and integrity. In this document, the combination will always be referred to as “authenticity.”

These two concepts are inseparable, and are usually provided by the same mechanism. This is logical if you think for a moment that authentication without integrity would permit an intruder to modify data along its path and have it accepted at the receiving end as authenticated.

In IPsec and IKE, the method used for authenticity is the message authentication code (MAC). The MAC is produced by a combination of encryption and the use of a hash algorithm.

Hash Algorithms

A hash algorithm is used to convert a string of a given length into a condensed string - i.e. a string of shorter, and generally fixed, length.

The result provides a unique “finger print” for the message, called a *digest*; it is easy to calculate, but very difficult and time consuming to reverse. Hashing is thus a “one-way” algorithm.

Hash algorithms used for authenticity are also generally required to be “collision-less” - i.e., it should be impossible to find two messages with identical digests.”

The combination of hashing and encryption (symmetric or asymmetric) is used to provide authenticity:

- Digital signature, using asymmetric encryption plus hashing, provides authenticity and non-repudiation of the message source.
- Digital seal, using HMAC, a special construction that applies a key to a hash algorithm, which provides authenticity without non-repudiation.

Digital Signatures

The process of providing a digital signature is as follows:

1. Alice wants to send a message to Bob. Before sending it, she hashes the message, producing a digest.
2. Alice applies an asymmetric encryption algorithm using her private key to the digest. This provides a digital signature which only she can generate, as only she has the private key.
3. Alice sends both the message and the signature to Bob.
4. Bob applies the same hash algorithm to the message, and uses Alice’s public key to decrypt the signature.
5. The digest produced by the hash algorithm and the digest that results from the decryption of the signature must be identical, or the authenticity of the message is not guaranteed.

Note: It would be possible simply to asymmetrically encrypt the data, however, as already noted, this is very slow. It is more general practice, therefore, to use asymmetric encryption of the digest (a much smaller file) only as the signature, and, where the message needs to be encrypted, to encrypt it symmetrically.

Digital Seals

The process of providing a digital seal is as follows:

1. Alice wants to send a message to Bob. Before sending it, she hashes the message, producing a digest.
2. Alice applies a hash algorithm to the message, which is further processed by the HMAC mechanism using a secret shared key. The resulting MAC is a seal. Nevertheless, Alice is not the only one capable of generating the seal, as Bob also has the same key.
3. Alice sends both the message and the seal to Bob.
4. Bob applies the same HMAC plus hash procedure to the message, using the same shared key.
5. The result of Bob's operation must produce an identical file to the seal sent by Alice. If they are not identical, the authenticity of the message is not guaranteed.

Note: It is also possible to generate a MAC entirely with encryption, or using several specialized methods. HMAC refers to a keyed hash. It is also possible to use a keyed MAC, in which the secret key appears before the digest, after the digest, or split and surrounding the digest.

Mutual Authentication and Session Key Sharing

As noted above, for data encryption, only symmetric algorithms are practical. Obviously, it is critical to keep the shared key secret in this scheme, as anyone who has it can decrypt the message and read it. Therefore, a system of session keys is used. Each session has its own key, and the exchange of keys must be authenticated. This being the case, authentication can be extended to the entire communication.

This presents two basic problems:

- A secure means for the correspondents to agree on a common key before beginning encrypted communications is essential.
- If many pairs of correspondents are involved, the number of keys can very soon become difficult to manage.

To solve these problems, a system of public key distribution is used. There are two methods of key distribution:

- Transport - the session key is asymmetrically encrypted by the sender using the receiver's public key, and transported to the receiver who decrypts it using his private key. The shared session key is used by both parties to encrypt and to read data.

- Generation - rather than send the actual key across an insecure network (even in encrypted form), a shared common key is generated by each party using publicly available information. In IPsec this is done using the Diffie-Hellman protocol.

The Diffie-Hellman Protocol

The Diffie-Hellman protocol uses public and private key parts to generate a secret session key which applies to a specific pair of correspondents.

Both public and private key parts are necessary to calculate the session key, but only the public parts are exchanged.

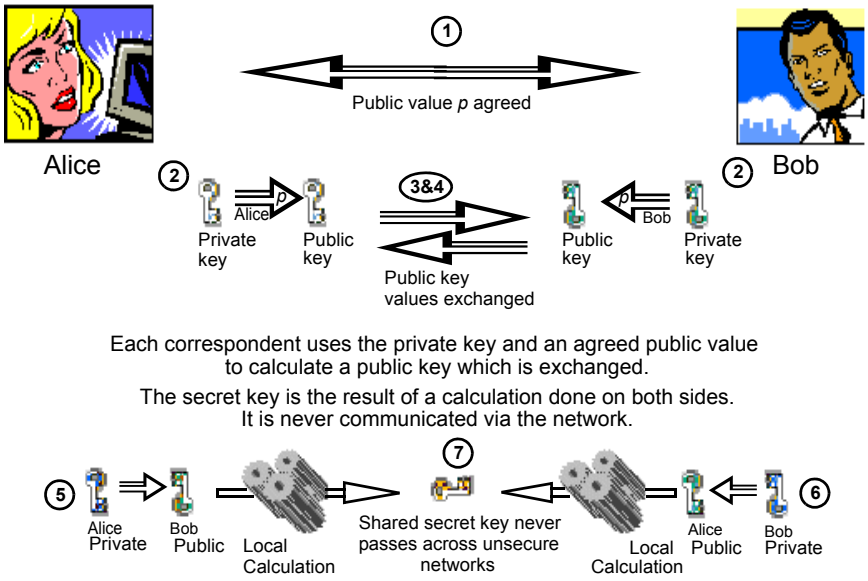


Figure 5: The Diffie-Hellman Protocol

The sequence is as follows (refer to Figure 5):

1. Alice and Bob wish to communicate over an encrypted link. They have agreed on a public value *p* that meets certain requirements of the Diffie-Hellman algorithm.
2. Alice chooses a second, secret value. This is her private key. She performs a calculation on it using *p*. Bob does the same thing.
3. The result of the calculations in Step 2 are the public keys for each correspondent (Alice and Bob).
4. Bob sends Alice his public key, and Alice sends Bob her public key.
5. Alice performs a calculation involving her own private key and Bob's public key.
6. Bob performs a calculation involving his own private key and Alice's public key.
7. The result of both calculations is the same. This is the common session key used to encrypt and decrypt the communications that pass between Bob and Alice.

This method ensures that each pair of correspondents has its own secret shared session key, without any secret information passing over insecure communication links.

Perfect Forward Secrecy

When Diffie-Hellman is used to negotiate session keys, it ensures that even if a key is cracked, previous and subsequent keys are not compromised as subsequent keys are not derived from previous keys. This is called perfect forward secrecy (PFS).

Diffie-Hellman Group

The Diffie-Hellman group parameter is used to control the length of the calculated keys. It is normally agreed during the IKE session as part of the proposal.

IKE uses pre-defined Diffie-Hellman groups, numbered from 1 to 5. Groups 1,2 and 5 use modular exponentiation with increasing key length. Groups 3 and 4, which are rarely implemented, use elliptic curves, and have the advantage of allowing shorter keys for a comparable security.

IPsec

As mentioned at the beginning of this chapter, IPsec works at the IP layer, and protects individual IP datagrams (packets). This means it is transparent to applications, and does not require existing applications to be modified.

This section explains some important aspects of the modes and functioning of the IPsec protocol:

- Protection Modes
- AH and ESP protocol extensions
- Encapsulation Process
- Encryption algorithms
- Authentication algorithms

Protection Modes

There are two protection modes for IPsec operation:

- Transport mode
- Tunnel mode

Transport Mode

IPsec transport mode protects only the content of a packet. It is usable on terminal equipment only. It does not modify the IP header, and thus cannot guarantee that the packet passes into the hands of the proper correspondent. This method is suitable only for end-to-end security, such as communication between a client Policy Server and a server, or between an administration station and network equipment such as a router, or between two clients (Bob and Alice for example).

Tunnel Mode

Tunnel mode involves the encapsulation of an IP packet in a new packet. The source and destination IP addresses of the *new* IP packet are specified and authenticated (in the new header), and thus information can be routed beyond the terminal equipment to the correct destination and still be protected. In most cases this method of protection is preferable, and is the mode supported by this release of Solsoft Policy Server.

AH and ESP

The two different protocols available for transport of IPsec packets (in either mode) are:

- AH
- ESP

These are two extensions of the IP protocol. They are the principal protocols used to implement IPsec, and they are the protocols that apply the different techniques and algorithms needed for data security.

AH (Authentication Header)

This protocol provides authenticity assurance for IP datagrams. It embeds a special header in front of an existing IP packet, preceded by a new IP header and followed by the original IP header.

The AH header data is used in a calculation to determine that the packet does originate with the presumed sender, and that it has not been altered en route to the receiver.

Note: The authenticity of the new IP header is not assured by this process.

AH provides authentication only. It does not provide encryption.

ESP (Encapsulating Security Payload)

This protocol can provide confidentiality as well as authenticity of data. As the name implies, it encapsulates the original IP packets. It functions as follows:

- A new ESP trailer is added to the end of the packet.
- Encryption is applied, using one of the supported algorithms, to the entire packet:
  - IP header
  - Data
  - ESP trailer
- An ESP header is added in front of the encrypted datagram. This header is non-encrypted.
- A new IP header is added before the ESP header
- An authentication code is added behind the encrypted datagram.

Using this protocol, everything from the ESP header to the encrypted ESP trailer is authenticated as to origin and integrity, and confidentiality is maintained via the encrypted datagram which can only be read by authorized entities possessing the session key.

Note: It can readily be seen that the functions of confidentiality and authenticity are closely interlocked, and that the choice of methods, algorithms and protocols for each aspect of an IPsec communication can impact the possible options for the rest.

Encapsulation Process

The process of IPsec encapsulation is shown in Figure 6.

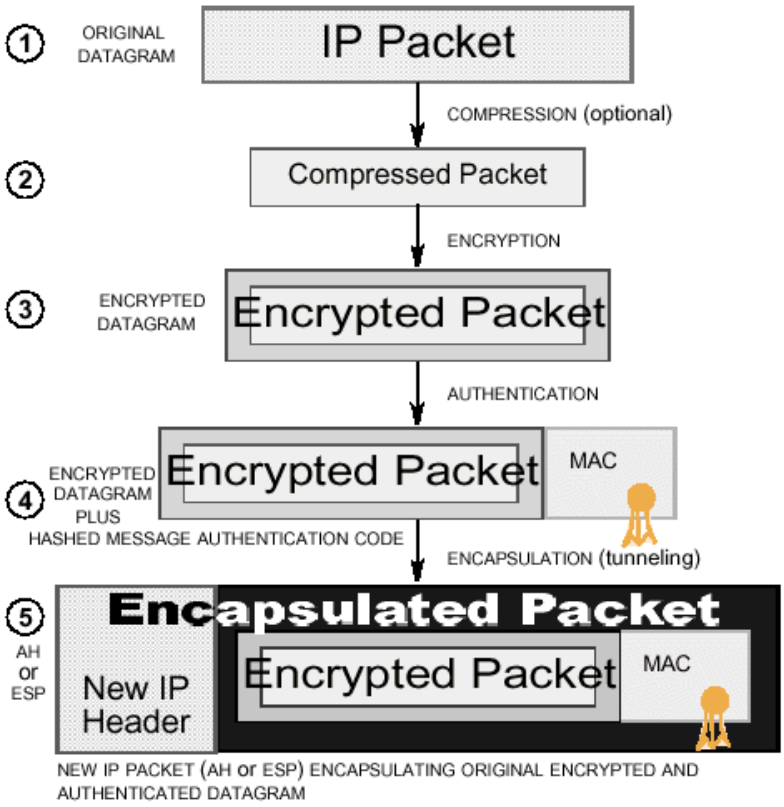


Figure 6: The IPsec Process

1. An IP Packet is identified as containing sensitive data that must pass via the IPsec tunnel.
2. The packet can be compressed to reduce the message size after encapsulation. This is optional but often recommended.
3. The packet can be compressed to reduce the message size after encapsulation. This is optional but often recommended.

4. A hash algorithm (negotiated by IKE at tunnel setup time) is applied to the encrypted packet to create a digital “signature” for the packet. The results of the hash operation are appended to the encrypted packet to provide authentication.
5. The encrypted packet plus authentication “signature” are encapsulated in a new IP packet (either AH or ESP) with its own IP header. This is the packet that is transmitted across the insecure network.

Encryption Algorithms Used

Solsoft Policy Server currently configures one of the following encryption algorithms for IKE communications or for data:

- DES
- 3DES

It is also possible to force data traffic to pass through the tunnel unencrypted, should you so desire.

Authentication Algorithms Used

Solsoft Policy Server currently configures one of the following hash algorithms for IKE communications:

- SHA-1
- MD5

For data, Solsoft Policy Server currently configures one of the following:

- HMAC-SHA-1
- HMAC-MD5

These are specific instances of the application of HMAC to an existing hash algorithm.

IKE

IKE is a protocol that brings up IPsec tunnels dynamically when needed (i.e. when there is traffic that needs protection).

IPsec can be configured without IKE, but IKE enhances IPsec by providing additional features, flexibility, and ease of configuration for IPsec. Tunnels configured by Solsoft Policy Server are always brought up (and down) by IKE.

IKE provides the following services:

- Negotiation and management of IKE parameters (security associations)
- Negotiation and management of IPsec parameters (security associations)
- Authenticated secure key exchange using the Diffie-Hellman protocol
- Mutual peer authentication using:
  - Shared secret keys
  - Public keys
- Identity protection in some modes

When bringing up a tunnel, IKE has to establish the capabilities of the two IPsec devices which will form the two ends of the tunnel, and the level of security that is required for the traffic that will use it. Since the capabilities of a remote device are often unknown, IKE negotiates the IPsec parameters according to the *proposals* of each device.

Proposals

In simplest terms, a proposal is a set of alternative parameters which IKE is authorized to offer to the remote device in order to arrive at a mutually acceptable SA. IKE provides a way to offer proposals to another device which will make the choice according to its proposals depending on its own capabilities. A proposal typically takes into account:

- The capabilities of the local device.
- Within the context of the known device capabilities, a set of parameters that will provide *at least* the minimum level of security needed.

As might be expected, each proposal is offered as an undivided package. Separate negotiation of each individual parameter could, in some situations, compromise security (high encryption without strong authentication, for example, would not provide the expected reliability).

At the time of tunnel creation, each device offers its own list of one or more proposals to the other. The devices will agree on the proposal to use as follows:

- The proposals accepted by each device must be compatible and capable of interoperating.
- From among the compatible pairs of proposals, the pair with the highest priority will be selected.

IKE Parameters

At the start of an IKE session, IKE must protect itself. IKE therefore negotiates an ISAKMP SA to protect IKE internal communications. The ISAKMP proposal includes these parameters:

- Algorithm to use for encrypting IKE communications.
- Hash algorithm to use for verification of IKE data integrity.
- Peer authentication method (private or public keys) for data origin authentication.
- Diffie-Hellman group to use for the key exchange.
- Lifetime of the SA.

IPsec Parameters

Once it has secured its own communications, IKE negotiates one or more IPsec SAs. These SAs protect traffic, and have a shorter lifetime than the ISAKMP SA. The IPsec proposal includes these parameters:

- Which IPsec protocol extension to use - ESP or AH?
- Use data compression, yes or no?
- Algorithm to use for encrypting traffic.
- Algorithm to use for authenticating data.
- If data compression is used, which algorithm?
- Diffie-Hellman group to use for PFS (in practice, usually the same as that used by IKE in the ISAKMP SA).
- Lifetime of the SA.

Note: One IKE SA can be used to negotiate several IPsec SAs.

Chapter 3: The Solsoft Policy Server Approach to VPN Management

Overview

Solsoft Policy Server is a product for distributed security policy provisioning and management. The Solsoft Policy Server approach to VPNs follows the same logic.

Thus, in Solsoft Policy Server you can define different aspects of the VPN according to your needs:

- You can define a tunnel between two devices and configure the tunnel with its IPsec and IKE parameters (tunnel policy).
- You can create multiple tunnel policies to create several different categories of tunnels and assign any tunnel in your logical network map (LNM) to the tunnel policy of your choice in a single click operation.
- You can modify the characteristics of a tunnel policy, which will automatically modify the characteristics of all tunnels using that policy.
- Once you have defined a tunnel and a trust zone (which together form the VPN), you can define permissions without worrying about the VPN. Traffic will be implicitly encrypted when needed.

Definition of Terms

This section defines important terms used in this document as they apply to VPN management in Solsoft Policy Server:

- Trust Zone
- Tunnel
- Virtual Private Network (VPN)
- IPsec Capabilities
- Tunnel Policy
- IPsec PEP

Trust Zone

A trust zone is a set of Solsoft Policy Server objects where all communications between two objects inside the zone use a path entirely inside it.

In other words, a trust zone is a group of objects amongst which you can communicate freely in confidence, knowing that any given communication between two objects in the zone will stay inside it, and thus never pass through an untrusted zone.

Limited Path Zone

A limited path zone is a Solsoft Policy Server feature, known in previous versions as a Perimeter Map, that lets you refine a security policy by describing a boundary which restricts the path packets may take when traveling through the global network. It has a common feature with a trust zone, that a flow entirely within the zone can never leave it.

Packets can, however, enter and leave a limited path zone from and to the exterior. Its special property is that once a packet leaves the area described by a limited path zone, it cannot return within its boundaries.

For more information on limited path zones, refer to the *User Guide*.

Tunnel

A tunnel is a virtual link defined by a set of IPsec parameters and IKE parameters applied to a pair of devices capable of implementing IPsec.

IPsec Capabilities

The IPsec Capabilities of a device are the set of algorithms and protocols supported by it.

Virtual Private Network (VPN)

Network: is a set of IP addresses that can communicate together freely.

Note: For purposes of VPN management, this definition varies slightly from the standard Solsoft Policy Server definition which defines a network as a set of IP addresses that can communicate together freely *without passing through a filtering device.*

Private: means that all communication between two points must be:

- Possible *only* between IP addresses identified as being within the private domain (i.e. trust zone), *and*
- Inaccessible and unreadable to any IP address outside the trust zone.

Virtual: building a VPN will group several networks into one virtual network, i.e. tunnels will provide virtual links between networks.

Thus, in Solsoft Policy Server, a trust zone that contains a tunnel is equivalent to a VPN.

Tunnel Policy

This is a list, containing sets of IKE and IPsec proposals which guarantee a given level of security desired by the user. When a tunnel policy is applied to a tunnel, only these proposals are authorized for use in the tunnel.

When at least one IKE and one IPsec proposal from the tunnel policy matches the capabilities of the managed IPsec device(s) to be connected by the tunnel, the policy is applicable.

The proposals in the tunnel policy are sorted by priority. When the tunnel policy is applied to the tunnel, all proposals which match device capabilities will be applied.

Note: The number of proposals that can be configured is limited to whichever is the lowest number of proposals supported by both of the devices.

When the tunnel is established, the proposal with the highest priority matching the capabilities of the managed IPsec device(s) will be used.

Note: The proposal actually used in the tunnel may be determined by the capabilities of a device which is not managed by, and is unknown to, Solsoft Policy Server. This is why a list of acceptable proposals is presented.

IPsec PEP An IPsec PEP is defined as a device capable of establishing and managing IPsec and IKE communications. An IPsec PEP can be a dedicated device that handles only this function, or it can be a filtering PEP which also has IPsec and IKE capability.

IPsec Definitions IPsec and IKE generic terms and concepts are discussed in Chapter 2, starting on page 21.

Managed and Unmanaged IPsec Devices

In some cases, you may wish to establish a tunnel between an IPsec PEP which is managed by Solsoft Policy Server, and another device which is not managed by Solsoft Policy Server.

It is also possible to have Solsoft Policy Server take into account a tunnel between two unmanaged IPsec devices, even though it does not control the devices directly.

In Solsoft Policy Server you can use the nexus object to represent an IPsec PEP which is not managed by it.

You can also use an unmanaged PEP if you know the capabilities of the unmanaged device and you want Solsoft Policy Server to take these capabilities into consideration.

Trust Zones

The trust zone is fundamental to Solsoft Policy Server's VPN management. As defined in this chapter, a trust zone is a set of Solsoft Policy Server objects amongst which you are able to communicate in confidence.

Trust zones follow these rules:

- A communication between two objects inside the same trust zone must remain inside the trust zone, and may not pass through any exterior path.

  This means that a communication cannot be intercepted by an external object unless the object has *explicit* permission to enter the trust zone.
- All communication from an object inside the trust zone to an object outside the trust zone, or from outside into the trust zone, is made as if the trust zone does not exist. Such communication must be explicitly authorized through permissions.
- All objects inside a trust zone are connected inside it - i.e. a path must exist between all objects in the trust zone that is entirely within it.

- When a tunnel is used to link two sets of objects in a trust zone which are separated by an untrustworthy path or domain (e.g. the Internet), the two trust zone segments are merged. Communication which must traverse the insecure link will automatically use the tunnel and be encrypted, to guarantee confidentiality.
- A trust zone can exist even if there is no tunnel inside it.

Note: The combination of trust zone plus tunnel, providing an unbroken secure communication path *which includes a virtual link*, is a VPN.

Trust Zone vs. Limited Path Zone

The Solsoft Policy Server limited path zone feature is a special case, more restrictive, of a trust zone. Both share a common property:

A communication between two points of the same zone must use a path inside the zone.

There are, however, some differences:

- When one end (source or destination) of a communication is not inside a limited path zone, the limited path zone still functions; it will not allow the communication to enter the perimeter once it has left it.
- When a communication passes through a limited path zone, and both source and destination are outside the limited path zone, the limited path zone also still functions, and a communication which leaves the perimeter cannot re-enter.
- A limited path zone does not require that all objects in it be connected inside it.
- Only a trust zone can be used to define a VPN.

Trust Zone vs. VPN

A trust zone need not have a tunnel inside it. A VPN can be defined as a trust zone which contains a tunnel.

How a Trust Zone is Implemented

When objects are assigned to a trust zone, Solsoft Policy Server generates filters for PEPs that contain interfaces connected to objects inside and outside the trust zone, such that flows inside the zone can never leave it (see Figure 7).

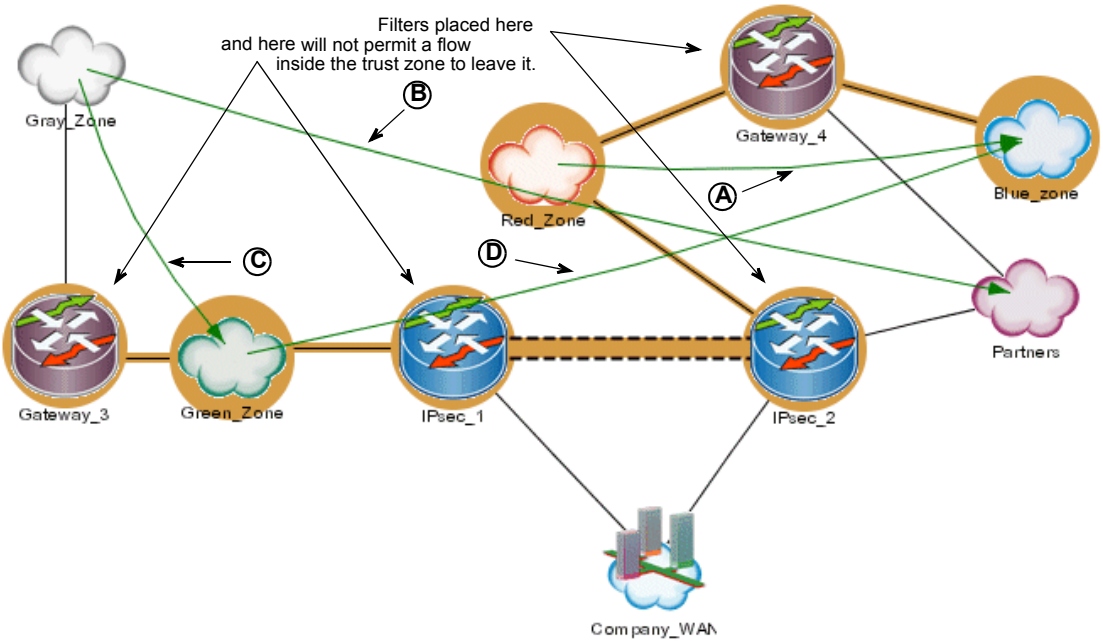


Figure 7: Examples Of Flows In And Out Of A Trust Zone

In Figure 7, the following are true (letters in bold face type below refer to callouts on the diagram):

A. The flow from `Red_Zone` to `Blue_Zone` is entirely inside the trust zone. It passes through `Gateway_4` and cannot pass through the `Partners` network (via `IPsec_2`).

B. The flow from `Gray_Zone` to `Partners` is between two objects entirely outside the trust zone. It passes through objects in the zone as if the trust zone did not exist, and passes unencrypted via the `Company_WAN` network, not through the tunnel.

C. The flow from `Gray_Zone` to `Green_Zone` is from an object outside the trust zone, to an object inside it. In this case, too, it passes as if the trust zone did not exist. This type of flow must be specifically authorized.

D. The flow from `Green_Zone` to `Blue_Zone` is entirely inside the trust zone (also a `VPN` in this case) and passes, encrypted, through the tunnel.

How Solsoft Policy Server Represents Tunnels

The following types of tunnels can occur in the security policy map:

- A tunnel between two IPsec PEPs managed by Solsoft Policy Server
- A tunnel between one IPsec PEP managed by Solsoft Policy Server and another which is unmanaged
- A tunnel between two IPsec PEPs, neither of which is managed by Solsoft Policy Server

This section explains how Solsoft Policy Server manages tunnels generally, and in each of these cases.

General Tunnel Behavior (Services)

In Solsoft Policy Server a tunnel has meaning only for communications which start and end inside the VPN which contains it. Other communication is excluded from the tunnel.

This means that the following are also true:

- A tunnel will be used only if included in at least one trust zone.
- The definition of which objects are allowed to use a given tunnel is determined by the trust zone which contains the tunnel.
- All permissions made inside the trust zone where the tunnel is a possible path will be implicitly encrypted (IPsec only) when passing through the tunnel.
- Permissions between two points of different trust zones (i.e. two trust zones not having an unbroken path between them - whether or not a tunnel is involved) will not use a tunnel, even if such exists. The tunnel is a possible path only when the communicating objects are on the *same* unbroken path (i.e. inside the same VPN).

Note: Encryption (IPsec only) only takes place in the tunnel portion of the communication path. The rest of the trust zone is, by definition, an area where unencrypted communication can pass in confidence. The logical consequence of this is that a communication between two points inside a trust zone which does not need to use the tunnel will not be encrypted.

IPsec Tunnel Policy Enforcement

A tunnel policy is automatically enforced if it is applicable. It is applicable if the two IPsec PEPs are able to accept at least one IPsec proposal and one IKE proposal contained in the policy.

There are three decision levels in the enforcing of a tunnel policy:

1. The user defines the set of all acceptable proposals, which are listed in order of priority in the tunnel policy.
2. All the proposals from the tunnel policy which match the capacities of the IPsec PEPs involved will be configured in each device, maintaining the priority order defined in the policy.

3. When the tunnel is established, the two IPsec PEPs will negotiate the best possible solution (as defined by the priorities in the policy) from among the configured proposals.

This system allows Solsoft Policy Server to manage situations where a single solution would fail due to problems of interoperability or incompatibility between the IPsec PEPs.

Both IPsec PEPs Are Managed

When both IPsec PEPs are managed by Solsoft Policy Server, Solsoft Policy Server generates the native IPsec configuration of both devices, and manages both ends of the tunnel and all communication passing through it.

Only One IPsec PEP Is Managed

In this scenario, there are two possible situations:

- The capacities of the unmanaged IPsec PEP are known.
- The capacities of the unmanaged IPsec PEP are unknown and/or uncontrolled by you.

The Capacities of the Unmanaged IPsec PEP Are Known

In this case, you can use an unmanaged PEP to represent the unmanaged IPsec PEP and manually configure its capabilities.

These capabilities will be taken into account by Solsoft Policy Server when configuring proposals from the tunnel policy.

The Capacities of the Unmanaged IPsec PEP Are Unknown and/or Uncontrolled

In this case, you use a nexus to represent the unmanaged IPsec PEP as it supports any IPsec and IKE capabilities. Solsoft Policy Server tunnel policy instantiation will take account only of the managed PEP capabilities.

At compilation time, NPE will generate a text file containing a generic set of IPsec parameters needed by the other side of the tunnel. This text file can then be passed to the person responsible for managing the other side of the tunnel (by fax, email or another method of your choice).

Both IPsec PEPs Are Unmanaged

In this scenario, you do not need to know the capabilities of either IPsec PEP, as you will not configure the devices in the LNM. The behavior is as follows:

- Solsoft Policy Server will recognize that the tunnel exists.
- If a trust zone containing the tunnel has been defined in Solsoft Policy Server, communications inside the VPN for which the tunnel is a possible path will be encrypted and use the tunnel.

How does Solsoft Policy Server Take an Unmanaged Tunnel Into Account?

All permissions inside a trust zone, for which an unmanaged tunnel is the only secure path, will be treated as follows:

- The permission, as in all Solsoft Policy Server permissions, has a source and destination for the service selected. In the example in Figure 8, the service is `http` and the source is the `East` network, destination is the `West` network.
- When the flow arrives at the outgoing tunnel Gateway (`IPsec_1` in Figure 8), the flow is converted into `ah` or `esp` flows, with source and destination being the two gateways of the tunnel. Thus, in Figure 8, the `ah`/`esp` flow has `IPsec_1` as its source and `IPsec_2` as its destination.
- at `IPsec_2` the flow is converted back to `http` and continues to its destination in the `West` network.

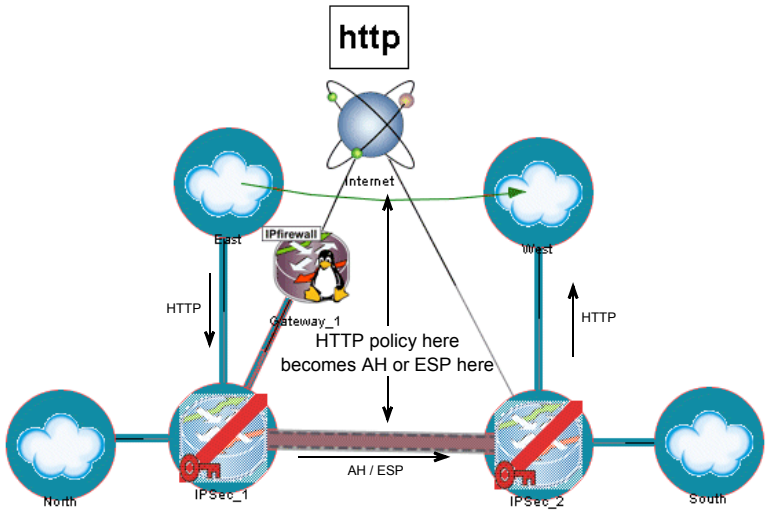


Figure 8: Flows In An Unmanaged Tunnel

The physical path used by the tunnel (across the Internet in the example used in Figure 8) may include additional PEPs (e.g. `Gateway_1` in Figure 8). Any such PEPs managed by Solsoft Policy Server will have filters applied to them with the following characteristics:

- The source and destination of the flow will be the tunnel endpoints (`IPsec_1` or `IPsec_2` in the example).

- The permission will be for `ah` or `esp` and *not* `http`.

GRE Tunnel Policy Enforcement

Routing Encapsulation (GRE) with IPsec is a solution that enables routing updates to be sent over the VPN, thus delivering greater network resiliency than IPsec-only solutions. Aside from providing a failover mechanism, GRE tunnels provide the ability to encrypt multicast/broadcast packets and non-IP protocols with IPsec over. Thus by using GRE with the support of AppleTalk and Novell IPX it is possible in a VPN solution.

GRE is the standard solution for service providers that want to provide managed IP VPN services across an established IP network. A significant advantage of GRE is that service providers can offer finer-grained QoS than with IPsec solutions. This is possible because the routers maintain visibility into IP packet header information necessary for fine-grain (application-level) QoS. In an IPsec packet, the header information is hidden.

In a GRE solution, traffic is restricted to a single provider network to enable end-to-end QoS control. This restriction of "on-net only" traffic also keeps GRE tunnels private without encryption.

Customers who require greater levels of security can also use "on-demand" application-level encryption such as secure connections in a Web browser. Alternatively, the entire connection may be encrypted, but at the cost of QoS visibility and granularity.

In summary, the advantages of GRE tunnels for service providers are:

- Encryption-optional tunneling
- Fine-grained QoS service capabilities, including application-level QoS
- Available as a Cisco IOS software feature on all Cisco routers and Layer 3 switches with routing capabilities
- IP-level visibility makes Cisco IOS software the platform of choice for building value-added
- services such as application-level bandwidth management
- more similar to IPsec tunnel mode.

Client-to-Gateway Tunnels

Solsoft Policy Server supports Client-to-Gateway Tunnels in IPsec mode on the following devices:

- Cisco PIX 6.0.1 and higher
- Cisco IOS 12.2.8.T and higher
- Cisco VPN 3000 3.0.1 and higher

Solsoft supports all VPN 3000 authentication server types:

- Internal Database

- RADIUS
- SecurID
- NT Domain

We manage only the user groups.

Using Tunnels and NAT Together

It is possible to apply or to disable NAT rules inside a tunnel, subject to the rules shown in Table 1.

Note: This table applies only to flows that pass through the tunnel - i.e. the option **Ignore All VPNs** is not set in the Permission Properties.

Note: For all PEPs, masquerading in a tunnel is prohibited by Solsoft Policy Server. To be able to make masquerading and a tunnel on the same PEP, you must either disable NAT in the tunnel or make the masquerading function of a destination that avoids passing through the tunnel.

Table 1: Rules for NAT in tunnels

Tunnel Endpoint	Application	Restrictions
All PEPs supporting IPsec or GRE	Static NAT rules in the tunnel.	**Disable NAT Rules in Tunnel** (see page 70) will not be applied to the static rules. You must use the NATed addresses to initiate connections that match a static NAT rule in a tunnel.
	You want to apply a NAT rule on a PEP which is on the physical path of a tunnel.	This application is not supported. You cannot use NAT on a path which is used by `ah` or `esp`.
Cisco IOS	NAT rules on the interface used as the endpoint of a tunnel.	This is only possible if: • No static NAT rules are used *and* • **Disable NAT Rules in Tunnel** is set to **Yes** (see page 70).

Table 1: Rules for NAT in tunnels (Continued)

Tunnel Endpoint	Application	Restrictions
Cisco Secure PIX Firewall	When: • You want to use the **Disable NAT Rules in Tunnel** option AND • The source address and the destination address of an encrypted flow overlaps with the source and destination addresses of an unencrypted flow (i.e. the **Ignore All VPNs** option has been set) for another service.	NPE will produce an error message at compilation time because the PIX cannot disable NAT on a per service basis, but only on the basis of source and destination addresses.
Cisco VPN 3000 Series	Disable NAT Rules in Tunnel	This application is not supported on this device. If NAT rules are set on the VPN 3000 in conjunction with this option, NPE will produce an error message at compilation time.
Shiva LanRover	Not supported	Change the destination of the NAT rule to avoid passing through the tunnel.
Nortel Contivity	only Dynamic	
Netscreen	only Dynamic	

The most common use of IPsec tunnels is to provide secure communication across the Internet via a VPN from one remote site to another inside the same organization. It is convenient, in this case, to continue to use internal addresses (thus non-routable from the outside) and not their translations. For this reason, NAT rules are disabled by default for flows authorized to pass through a tunnel.

There may be situations, however, where you want NAT rules to be activated. See "Verify Tunnel Properties" on page 70 for the procedure.

Fully Meshed/ Hub and Spoke VPNs

You can manage a fully-meshed VPN or a hub-and-spoke VPN with Solsoft Policy Server. The following requirements must be met:

- Each tunnel is a point-to-point tunnel.
- Only dedicated interfaces can be used to make tunnels.

See the section "Working with Fully-Meshed and Hub and Spoke Tunnels" on page 99.

VPN Implementation Phases

To implement a VPN in Solsoft Policy Server, you perform the following tasks:

- Declare the trust zone or VPN topology:
  - Networks
  - Filtering PEPs
  - IPsec PEPs
  - Nexus
- Define one or more tunnel policies.
- Create the tunnel.
- Define permissions in the VPN.

You can perform these operations in any order.

Declare the Trust Zone or VPN Topology

In this phase, you determine the objects which need to communicate in a secure fashion. You assign all these objects to a trust zone, to ensure communication between them rests inside the zone.

You add a tunnel to extend the trust zone through an untrustworthy network. In effect, you create a secure path through the untrustworthy network by building a VPN (trust zone plus tunnel).

The untrustworthy network could be the Internet, but it also could be an internal network, or part of an internal network, in which you do not have enough confidence to transmit sensitive data.

The steps of this phase are:

1. Declare a trust zone.
2. Determine which objects must communicate securely with each other.
3. Assign these objects to the same trust zone.

Define One or More Tunnel Policies

In this phase, you use the tunnel policy editor to create one or more tunnel policies. These are defined on the basis of the minimum level of security that will be acceptable to the user. It is, of course, possible to have multiple tunnel policies with different levels of security, which are applied to different tunnels according to need.

Solsoft provides templates for a number of pre-defined tunnel policies with different levels of security. They can be used “as is” or as the basis from which to define your own policies. You can also design a tunnel policy completely from scratch.

Create the Tunnel

In this phase, you draw the tunnel in the LNM and then assign parameters to it. The steps are:

1. Draw the tunnel (includes assigning a tunnel policy to it).
2. Check the tunnel configuration
3. Adjust the parameters of the tunnel

Define Permissions

In this phase, you define which flows are permitted in the tunnel. This is done in the same manner as for other permissions in Solsoft Policy Server.

All permissions where the source and the destination are in a trust zone of the same type connected by a tunnel will be encrypted by default.

You can set individual permission properties to require that a given flow will always be unencrypted, even if normally it would pass through the tunnel.

Note: When you attach a NAT to an interface, the tunnel must be considered as an interface. That is to say that if it is an outgoing interface, a translated packet will not be translated in the tunnel even if the NAT is attached to the interface used to make the tunnel.

Chapter 4: VPN Procedures

Configuring IPsec VPNs using Solsoft Policy Server's VPN Module 1.0 is done through a set of procedures which logically and flexibly allow you to manage many VPNs at the same time.

There are six main groups of procedures:

- Global Configuration
- Configuring IPsec PEPs
- Defining a VPN
- Defining Permissions in a VPN
- Managing Certificates and PKI
- Working with Tunnel Policies
- Working with Fully-Meshed and Hub and Spoke Tunnels

These can be done in any order, although the order of appearance mentioned above is the most usual.

In the sections that follow, you will often find a procedure guide at the beginning. This is your key to the tasks you need to perform for each procedure, and the order in which to perform them. Read it carefully before starting a new procedure for the first time.

Global Configuration

There are a few settings which affect the global functioning of the Solsoft Policy Server VPN Module 1.0. To be sure that everything will function the way you intend it to, perform the following procedures:

- Check the License
- Set Display Options
- Set Preferences

After checking the license, if default settings for display options and preferences suit your needs, there is no need to perform these procedures.

Default settings are:

- **View** menu (display options): all zones are shown by default. If you want to see the tunnel configuration, `uncheck not managed`.
- **Tools >Preferences** menu:
  - **Implicit Permissions** are shown.
  - **General >Pre-Shared Key** automatic generation is activated.

Check the License

You must have the proper license to use the VPN Module 1.0. Check your license conditions as follows:

1. Select **Help >About** Solsoft Policy Server from the menu bar. The About Solsoft Policy Server window opens.
2. In the About Solsoft Policy Server window, scroll to the bottom where licensed features are listed. You should see the following text:

```
VPN tunnels support is enabled and is licensed for:
  A maximum of <number> IPsec tunnels
  The generation of native IPsec configuration for:
        Cisco IOS
        Cisco Secure PIX Firewall
        Cisco VPN 3000 Series
```

   If this does not appear, either you are not licensed for VPN support, or your license is not functioning properly. Contact support@solsoft.com to resolve license problems.

Note: It is possible to have support for tunnels in Solsoft Policy Server without the ability to configure them. In this case, the “native IPsec configuration” line will be absent from the About Solsoft Policy Server window. You can take tunnels into account, but you are not licensed to create or use one.

When you create an IPsec PEP in the workspace, its icon is shown with a red key if you are licensed to configure tunnels. If not, the key is shown in gray (see Figure 10).

---

3. If your license is correct, go on to the next *Global Configuration* task.

Set Display Options

These options determine the way in which VPN-specific elements (trust zones, tunnels) are displayed in the logical network map.

1. Select the **View** menu bar option.

2. Among the options presented in the drop down menu are **Show Trust Zone** and **Show Limited Path Zone** (see Figure 9). Under most circumstances, you will want to select them both, otherwise you will not be able to see elements that are essential to configuring a new tunnel. Also check the VPN View to see all the VPNs.

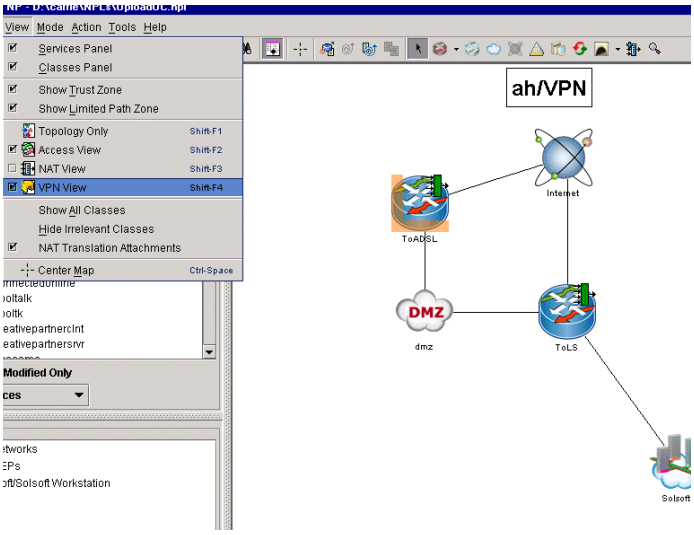


Figure 9: View Options

Note: During VPN configuration, you may want to temporarily de-select **Show Limited Path Zone** to avoid any possible confusion between these zones and a trust zone.

3. Go on to the next *Global Configuration* task.

Set Preferences

The settings in the Preferences tool control the display of implicit permissions and the automatic generation of pre-shared keys.

Set Display of Implicit Permissions

1. Select **Tools >Preferences** from the menu bar. The Preferences window opens.
2. Select **Implicit Permissions** from the tree list. The check box in the parameters area should be selected (this is the default). This will show permissions which are automatically generated by Solsoft Policy Server when a tunnel is created.

   These implicit permissions are necessary for the tunnel to function and are shown in the LNM as blue permission arrows.
3. Click **OK** to close the Preferences window or go on changing preferences.
4. Go on to the next task.

Set Automatic Pre-Shared Key Generation

1. Select **Tools >Preferences** from the menu bar. The Preferences window opens.
2. Select **General** from the tree list. In the **Pre-Shared Key** pane, check the check box labeled **Automatically Generate Pre-Shared Key** to activate this option (it is deactivated by default).
3. Click **OK** to close the Preferences window or go on changing preferences.
4. Go on to the next task.

BEHAVIOR OF THE SOLSOFT SECURITY DESIGNER WITH AUTOMATIC OR MANUAL PRE-SHARED KEYS

When the **Automatically Generate Pre-Shared Key** option is activated, the following are true:

- When a new tunnel is created that has a tunnel policy associated with it, and is defined between two PEPs that support IPsec, a pre-shared key will automatically be generated at creation time.
- If there is no tunnel policy, or if one of the PEPs does not support IPsec, no pre-shared key will be generated. The tunnel will be shown as invalid in the LNM.

When the **Automatically Generate Pre-Shared Key** option is not activated, the following are true:

- Since there is no pre-shared key, the tunnel will be shown as invalid in the LNM, even if all other parameters are OK.
- You must manually create the pre-shared key by using the **Create...** button in the Tunnel Properties window. If all other parameters are OK, the tunnel will be shown as valid in the LNM after this action.

In both cases, the following are true:

- If one of the two endpoint IPsec PEPs is not capable of accepting or using a pre-shared key, it will not be generated. The tunnel will be shown as invalid in the LNM.
- You can manually edit or regenerate the pre-shared key at any time by using the **Edit...** button in the Tunnel Properties window.

Note: The first time a pre-shared key is generated during a Solsoft Policy Server session takes a little longer than any subsequent generations during the same session. This is because the random number generator must initialize. Any additional key generation will need less time.

Configuring IPsec PEPs

The Solsoft Policy Server VPN Module 1.0 configures the VPN capabilities of Cisco IOS, Secure Pix and VPN 3000 Series PEPs which are properly equipped and running the right software versions (see prerequisites under "Enable IPsec on the PEP", below)

IPsec PEP configuration follows the same procedures (as detailed in the *User Guide)* as for filtering PEPs.

This section covers configuration items which are specific to IPsec PEPs

IPsec PEP Configuration Procedures

The procedures involved are given in the following sections, along with the tasks you need in order to perform each procedure.

TO CONFIGURE AN IPSEC PEP FOR VPN USE do the following tasks:

1. Enable IPsec on the PEP
2. Configure VPN-Specific Properties

TO PREPARE AN UNMANAGED IPSEC PEP FOR VPN USE do the following tasks:

1. Enable IPsec on the PEP
2. Make an IPsec PEP Unmanaged
3. Configure VPN-Specific Properties

Enable IPsec on the PEP

Prerequisite

The PEP must be one of:

- Cisco IOS running IOS software version 12.0 or higher
- Cisco Secure PIX Firewalls running software version 5.1 or higher
- Cisco VPN 3000 Series PEPs with software version 3.0 or higher.
- Nortel Contivity with software version 4.5.
- Shiva LanRover with software version 8.3 or higher.

Procedure

1. Open the PEP properties window by double clicking the PEP in the LNM.
2. Select **General Options** from the tree list.
3. In the **General Options** view, set the check box for **Has IPsec Module** to **Yes** (the default on PIX). The tree list expands to include a **VPN Options** view.
4. Go on to the next task as outlined in the appropriate procedure of the *IPsec PEP Configuration Procedures* above.

Configure VPN-Specific Properties

Prerequisite

The PEP must be one of:

- Cisco IOS running IOS software version 12.0 or higher
- Cisco Secure PIX Firewalls running software version 5.1or higher
- Cisco VPN 3000 Series PEPs with software version 3.0.

Procedure

From the PEP properties window:

1. Select **VPN Options** from the tree list
2. Set parameters as needed:
   - **DES Encryption Enabled** cannot be modified and is permanently set to **Yes**.
   - **3DES Encryption Enabled** - when set to **Yes** (default on the PIX), indicates that the PEP is equipped to handle this encryption algorithm.
3. Double click **VPN Options** or click the key beside it to expand its entry. The subentries **IKE Capabilities** and **IPsec Capabilities** contain no editable data - they are provided for information.
4. It is possible to assign the IPsec PEP to a trust zone now, from the **Zones** view. This is optional, as you can assign objects to a trust zone in a group from the logical network map (see "Assign Objects to a Trust Zone" on page 60*)*.
5. Click **OK** to close the properties window or go on setting PEP properties.
   Once IPsec has been activated in the Properties window and the Properties window has been closed, the PEP appears in the work-space with a key superimposed on it. The key is red if your license authorizes native configuration of tunnels, and is gray if your license only authorizes taking tunnels into account (see Figure 10).

Figure 10: IPsec PEP Display

6. Go on to the next task as outlined in the appropriate procedure of the *IPsec PEP Configuration Procedures* above.

Make an IPsec PEP Unmanaged

Use an unmanaged IPsec PEP as a tunnel endpoint when you do not manage the device from your LNM, and you know its capabilities.

Solsoft Policy Server takes the IPsec capabilities configured on the unmanaged IPsec PEP into account in calculating the proposals to offer from the tunnel policy.

Prerequisite

The PEP must be one of:

- Cisco IOS running IOS software version 12.0 or higher
- Cisco Secure PIX Firewalls running software version 5.1or higher
- Cisco VPN 3000 Series PEPs with software version 3.0.
- Nortel Contivity with software version 4.5.
- Shiva LanRover with software version 8.3 or higher.

Procedure

From the PEP properties window:

1. Select **General Options** from the tree list.
2. In the **General Options** view, set the check box for **Unmanaged by** Solsoft Policy Server to **Yes**.
3. Go on to the next task as outlined in the appropriate procedure of the *IPsec PEP Configuration Procedures* above.

An unmanaged IPsec PEP is represented in the LNM as shown at right.

Defining a VPN

A VPN consists of a trust zone which includes at least one tunnel, and thus also at least two IPsec PEPs. To define a VPN, you must perform the following procedures.

VPN Definition Procedures

You define the VPN by performing a series of procedures. The procedures have two purposes:

- Meet the immediate need of secure communication across an untrusted network.
- Create a configuration that is easily reusable, and that frees you from having to configure everything from scratch each time you need to set up a tunnel.

Declare a Trust Zone

You declare a trust zone in the Zone Editor. The procedure is as follows:

1. Select **Tools >Zone Editor** from the menu bar. The Zone Editor window opens.
2. Click the **Add Zone** action button  . The tree list shows the new **Zone 1** selected and the parameters area fills with controls.

3. Enter a new name for the trust zone using the **Rename Selected Zone** action button  . The Rename Zone dialog box appears.
4. Enter the new name in the field provided and click **OK** in the dialog box. The new name is now shown in the tree list.

Note: The name should provide an easy reminder of the purpose or function of the trust zone, to be recalled when assigning objects to it.

You cannot use the "/" character in a zone name.

5. The **Type:** field is set to **Trust Zone** by default. Confirm this, and change it if necessary.
6. Select the color for display of the trust zone by adjusting the color sliders in the parameters area.
7. Click **OK** to close the editor or continue adding more zones.

Note: At this point, the trust zone exists only as an association between name and display color. The zone is defined by assigning objects to it using the procedure that follows.

Assign Objects to a Trust Zone

The task below explains how to add a tunnel to a trust zones.

Procedure

1. Select the two tunnel ends (i.e. the IPsec PEPs) by clicking them while holding down the <Shift> key or the <Ctl> key.
2. Right click over one of the selected IPsec PEPs and click the check box next to the desired trust zone. The IPsec PEPs are now assigned to the zone.
3. From each IPsec PEP:
   a) Follow the physical path back into the network, noting the networks and PEPs to include in the trust zone.

Attention: You must ensure that a path is possible between all the objects inside the trust zone, and that this path does not leave the zone.

   b) Select the objects one by one using the <Shift> or <Ctl> click method described in Step 1 OR

      Use the lasso technique (left click and hold, and use the mouse to draw a box around the objects you want to select while holding down the <Shift> key).
   c) Right click over one of the selected objects and click the check box next to the desired trust zone. The objects are now assigned to the trust zone. Note the following rules for attaching objects to the trust zone:
      • Only networks, PEPs and Nexus objects can be assigned to a trust zone.

- Attached classes are assigned to a trust zone *only* via the networks to which they are attached. They cannot be assigned individually.
- Unattached classes can never be assigned to a trust zone.

Note: It is possible, and often desirable, to include more than two IPsec PEPs in a trust zone (when you want to build more than one tunnel, for example). However, it is good practice to start from the end points of a tunnel you know you want to create, and then decide which other objects you want to include in the trust zone, whether networks, PEPs or Nexus objects.

---

4. When you have completed Step 3 for both IPsec PEPs, verify the following:
   - Only trusted networks are included in the trust zone. Do *not* include untrustworthy or insecure networks (such as the Internet) in a trust zone, as this will defeat its purpose.
   - Objects on opposite sides of the tunnel which need secure communications with each other are included in the *same* trust zone.
   - On each side of the tunnel, a physical path exists which is entirely inside the trust zone, and permits all the objects in it to communicate.
5. If necessary, adjust the trust zone to conform to the conditions in Step 4.

   You have now defined your trust zone. Figure 11 shows an LNM in which two trust zones have been declared. There are also two limited path zones, but these are not shown as the display has been turned off in the **View** menu.

   Note that classes attached to networks in a trust zone are also shown as included in it, and that the network `DMZ`, with its attached class

`MailNotes Server`, are shown as belonging to both trust zones (which is, indeed, the case).

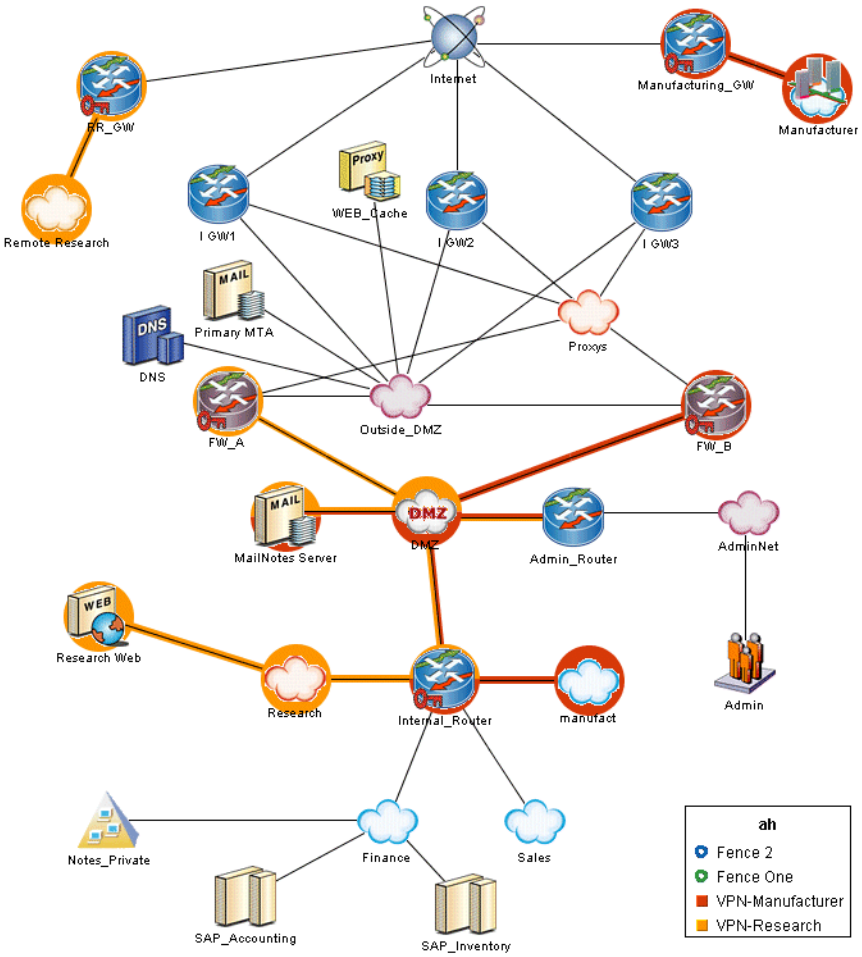


Figure 11: An LNM With Two Trust Zones

In both trust zones, all included objects are physically connected inside the trust zone, and thus tunnels established between IPsec PEPs in each trust zone will create functioning VPNs.

Remove Objects from a Trust Zone

Alternative methods for removing an object from a zone are available in the *User Guide*. Refer to that document for details.

1. Select the objects to be removed by holding down the `<Shift>` key while clicking on them.
2. Right click over one of the selected objects and select **Zones** from the shortcut menu. Click the check box next to the trust zone from which to remove the objects.

3. To remove the objects from all zones (trust zones *and* limited path zones) select **Zones >None** from the short cut menu.

Delete a Trust Zone

You delete a trust zone in the Zone Editor. The procedure is as follows:

1. Select **Tools >Zone Editor** from the menu bar. The Zone Editor window opens.
2. In the tree list, select the trust zone you want to delete.
3. Click the **Remove Selected Zone** action button  . The zone is deleted.
4. Click **OK** to close the editor or continue deleting more zones.

Create an IPsec Tunnel

1. Enter the Add Tunnel mode by doing one of the following:
   - Select the tunnel icon  in the toolbar. A menu appears listing all the tunnel policies you have created (if any) plus the pre-defined tunnel policies furnished by Solsoft. The current default (the last tunnel policy that was applied) is indicated on the menu. Select the tunnel policy to use for the new tunnel.

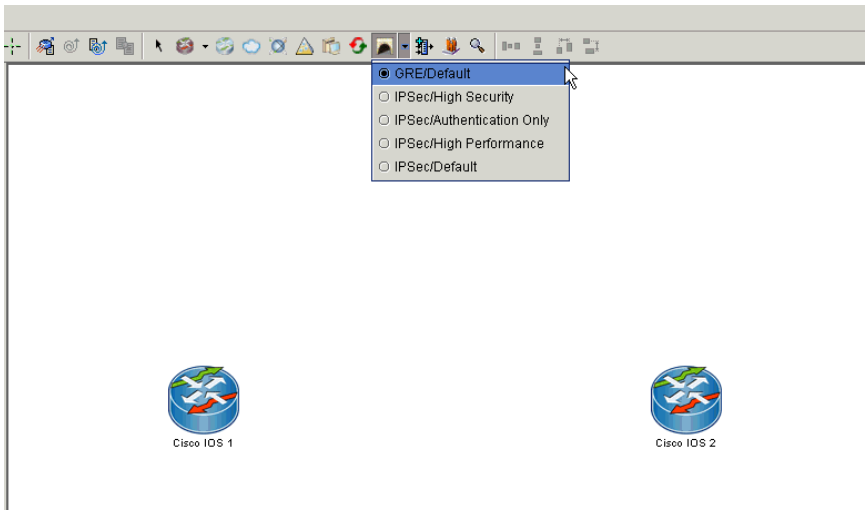


Figure 12: Choosing the type of Tunnel

   - Select **Mode >Add Tunnel** from the menu bar. In this case, the last tunnel policy used will be applied to the tunnel. If no previously used tunnel policy can be identified, no tunnel policy will be applied.

   See also "Working with Tunnel Policies" on page 93.

2. Place the cursor over one of the IPsec PEPs serving as a tunnel endpoint. Left click and hold the mouse button down. The cursor is transformed into the tunnel icon.

3. Drag the cursor to the other endpoint IPsec PEP and release the mouse button. If automatic pre-shared key generation is activated, a pause occurs while Solsoft Policy Server calculates the random key. If the tunnel is valid, a double dashed line representing the tunnel appears on the logical network map. The color associated with the trust zone fills the space between the dashed lines.

Note: When automatic pre-shared key generation is activated, both endpoint PEPs must be capable of accepting and using a pre-shared key. Otherwise the tunnel will be invalid.

Note: To avoid problems due to the mix of IPsec and NAT be sure that you follow the limitations on using NAT in tunnels - see "Using Tunnels and NAT Together" on page 47.

Reading The Result

Figure 13 shows the same LNM as Figure 11 with the addition of four tunnels and two new networks, connected through a new IPsec PEP which is not managed by Solsoft Policy Server.

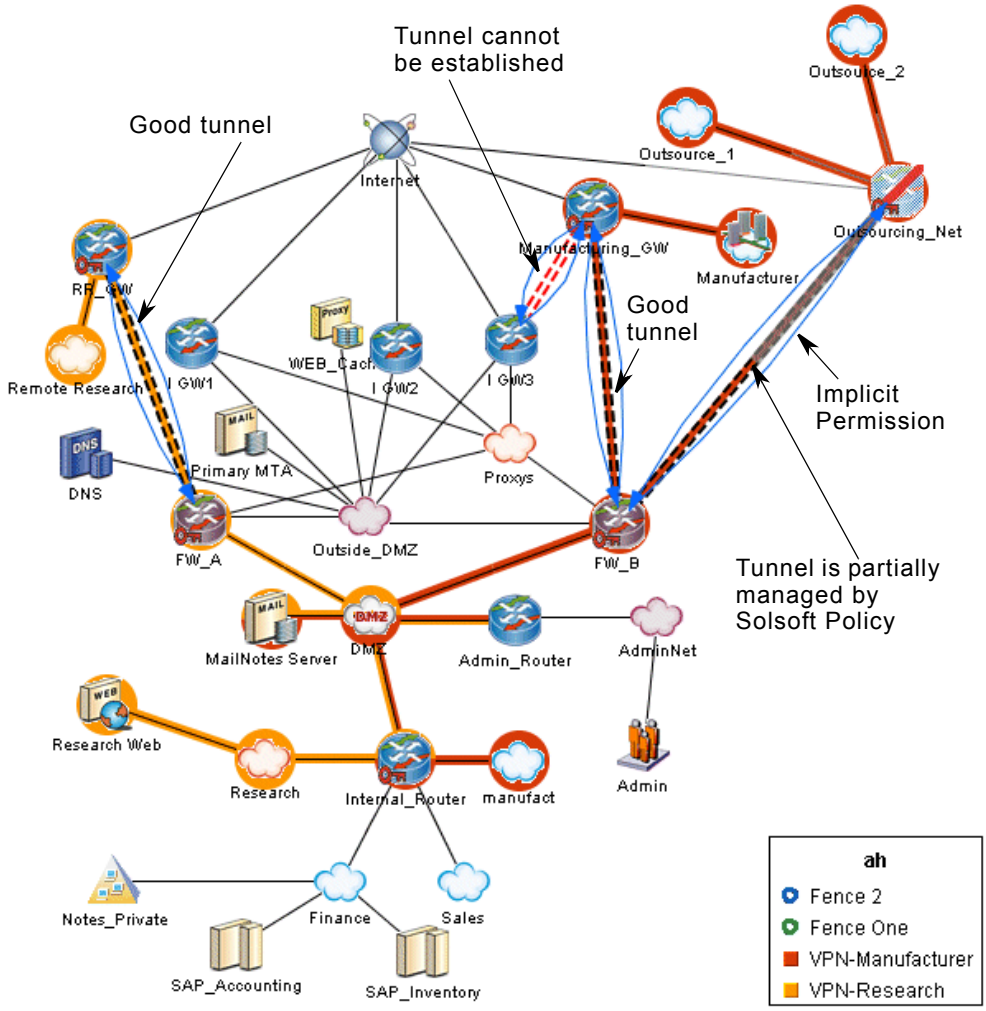


Figure 13: Two VPNs, Four Tunnels (One Bad)

Tunnel Colors: The color of the double dashed line indicates the state of the tunnel:

- Black lines indicate that the tunnel policy contains acceptable proposals for the two endpoint devices. The black lines are filled with the trust zone color (e.g. between `RR_GW` and `FW_A` or between `Manufacturing_GW` and `FW_B`).
- Red lines indicate that the tunnel is invalid. There can be several reasons for this:

- There is no tunnel policy assigned to the tunnel. In this case, select one in its Tunnel Properties window (see "Configure Tunnel Properties" on page 73.
- There is no valid proposal for the two endpoints. In this case, you must use a different tunnel policy or modify the one in use to provide proposals that conform to the capabilities of the two endpoint devices.
- You have not activated the AES encryption or 3DES encryption algorithms in the **PEP Properties >VPN Options view**.
- There is no pre-shared key, or the pre-shared key is not correct.
- The endpoints are not inside the same trust zone, in which case the tunnel is not filled with the trust zone color but it is valid. In this case you should complete trust zone assignment.
- One of the endpoint devices does not support IPsec tunnels.

In Figure 13 the tunnel between `Manufacturing_GW` and `I GW3` is not valid. The PEP`I GW3` does not support IPsec tunnels (no red key icon), and it is not inside the same trust zone as `Manufacturing_GW`. It is therefore shown in red.

- Gray lines indicate that the tunnel is not configured by Solsoft Policy Server.
- If the lines are gray over half their length and black over the other half, Solsoft Policy Server configures only one of the devices. The black lines will touch the managed device. See the tunnel between `Outsourcing_Net` (an unmanaged IPsec PEP) and `FW_B`.

Implicit Permissions: Permissions for basic services are implicitly required for a tunnel to work properly (IPsec, IKE, AH, ESP...). These will be created automatically when you define the tunnel if you have not disabled this function in the **Tools >Preferences >Implicit Permissions** window from the menu bar (the default is enabled). Implicit permissions are shown as blue arrows. They cannot be edited. Since Figure 13 shows the **ah** service view, implicit permissions are visible for each tunnel.

Create a Client-to-Gateway Tunnel

This is a particular type of IPsec Tunnel. You need to create it as you create other IPsec tunnels.

1. Define the server (not needed in the case of the Internal Database).
2. Define the User Group.
3. Define the Mapped User Group.
   You can associate it to a metaclass that contains several network when the User Group can initiate from different networks define on the map.
4. Associate the servers to the PEP.
5. Associate the Mapped User Group, the VPN Server and the network accessed through the VPN in the same Trust Zone.
6. Create the tunnel between the Mapped User Group and the PEP.

7. Configure the tunnel properties. The main ones are:
   - group password
   - the pool to use
   - split tunnel policy

   Other parameters are the same as the VPN 3000 User Group parameter.

Create a GRE Tunnel

. There are, however, a few differences.

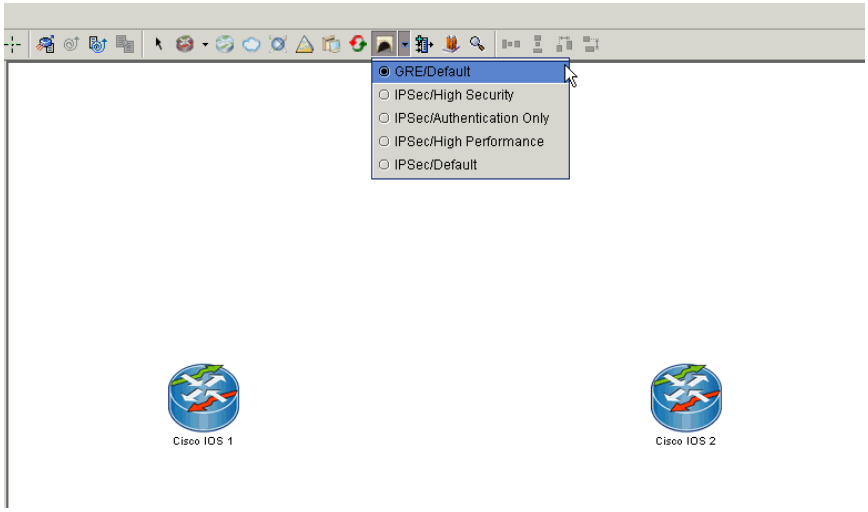


Figure 14: Choosing to create a GRE Tunnel

Procedure

1. Click the drop down list beside the **Add Tunnel** button on the toolbar, and select the **GRE/Default** policy.
2. You can create a GRE tunnel between two PEPs. Draw the tunnel by clicking on one object and dragging the mouse pointer to the other object.

Note: Only Cisco IOS supports GRE tunnels.

3. Open the tunnel properties window to configure the tunnel. If the **Show Errors** button appears, you can click it to show what you still need to configure.

Create an Encapsulated GRE Tunnel over an IPsec Tunnel

Prerequisites

- You must have two PEPs as the tunnel endpoints on your workspace.
- Both of the PEPs must support GRE and IPsec. See the Supported PEPs chapter of the Solsoft Policy Server Reference Manual.
- You must enable the option “Has IPsec Module” in the **PEP Properties >General Options** view, on both PEPs.

Procedure

1. Create a tunnel as in “Create a GRE Tunnel”.
2. Open the tunnel properties window and, from the tree list, select the Primary Tunnel view.

Check the **Has Carrier Tunnel** option so that it displays “Yes.” A new view, Carrier Tunnel, appears in the tree list.

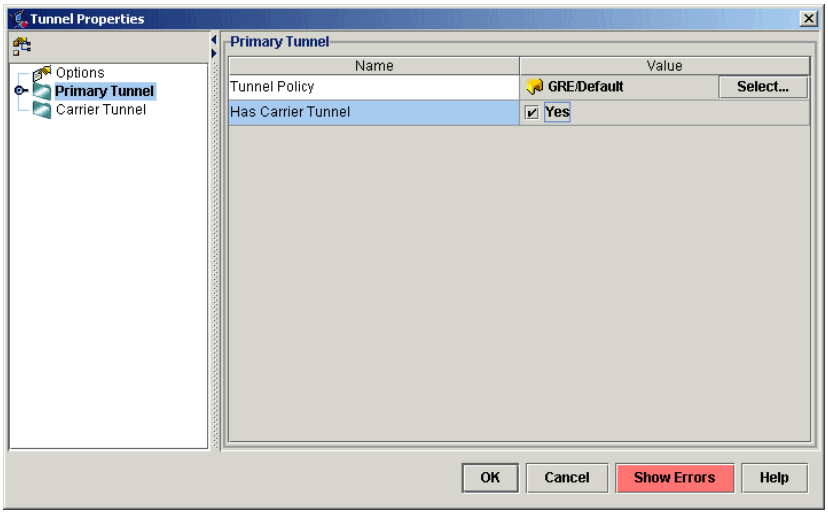


Figure 15: Primary Tunnel window

3. The carrier tunnel is the tunnel that encapsulates the GRE tunnel. Select the Carrier Tunnel view to configure the carrier tunnel.
4. In the Carrier Tunnel view, in the Tunnel Policy field, select IPsec/ Default.

Configure a Tunnel that Passes Through Dynamic NAT (NAT-T)

Solsoft Policy Server lets you configure NAT-T on an IPsec tunnel.

Note: Only certain PEPs support NAT-Traversal. See the PEP Specific Functions section in the Reference Manual for a list of supporting PEPs.

Prerequisites

- You have created a tunnel between two PEPs
- Both of the PEPs support NAT-T. See the PEP-Specific functions section in the Reference Manual.
- Neither of the PEPs have the TED option enabled. See, "Configure a Tunnel that has Dynamic IP Addresses" on page 69, for a description of the TED feature.

Procedure

1. Open the tunnel properties window and, from the tree list, select the **Primary Tunnel >Options** view.
2. Select the NAT-T option.

3. Make sure that the NAT Keepalive value, in seconds, is lower than the NAT expiration time. NAT expiration time is usually 20 seconds, but you can configure it directly on the device that performs the NAT.

Configure a Tunnel that has Dynamic IP Addresses

There are three methods of configuring tunnels when the PEP interfaces that serve as the tunnel's endpoints have dynamic IP addresses. The method Solsoft Policy Server uses depends on parameters you set in the Tunnel Properties Window, and in the Tunnel Policy Editor.

Table 2: Methods of mounting a tunnel with dynamic IP addresses

Tunnel Properties	Tunnel Policy Editor Parameters	Resulting Method	Description
Tunnel type = Default	IKE proposals include at least one "Pre-Shared Key (PSK)" method.	Masked Pre-Shared Key	The masked pre-shared key method means that the PEPs each have a key that they have somehow securely shared. The PEPs use this key to identify themselves. The key is "masked", or not associated with the PEP's IP address.
Tunnel type = Default	IKE proposals only contain the "RSA-Sig" method.	Certificate	The certificate method means that the PEPs' certificates, and not their IP addresses, prove their identity. **Note:** See "Managing Certificates and PKI" on page 76 to configure certificates on your workspace.
Tunnel type = TED	N/A	TED	Using TED, the PEPs send out probes in order to find their peer when mounting the tunnel.

Prerequisites

- You have created a tunnel between two PEPs

Procedure

1. Open the tunnel properties window and, from the tree list, select the **Primary Tunnel >Options** view.
2. In the **Tunnel type** field, select Default or TED.

Verify Tunnel Properties

Once you have defined the tunnel, you need to be sure that the configuration conforms to your needs. Follow these steps:

1. Read the Tunnel Properties Window. This window is divided into two areas: **Tunnel Options** and **Tunnel Status**. In the **Tunnel Options** area:
   - **Tunnel Policy:** shows the tunnel policy in force.
   - **Unmanaged by** Solsoft Policy Server**:** - a value of **No** (the default) indicates that the current tunnel will be configured by Solsoft Policy Server.
   - **Generate Implicit Tunnel Permissions:** - a value of **Yes** means that Solsoft Policy Server will automatically generate the implicit permissions needed to set up the tunnel (IPsec, IKE, AH, ESP...). The default value is a function of the Preferences settings.
   - **Disable NAT in Tunnel:** - The following values apply:
     - **No**: use NATed addresses in the tunnel.
     - **Static NAT**: disable only Static NAT on each PEP for traffic that enters the tunnel
     - **Dynamic NAT**: disable only Dynamic NAT on each PEP for traffic that enters the tunnel
     - **All NAT**: disable both Dynamic & Static NAT on each PEP for traffic that enters the tunnel

Note: For Static NAT, Dynamic NAT and All NAT, both peers try to disable what is requested on the tunnel and warn about a NAT that can't be disabled (for example, Cisco will not disable static NAT.)

   - **Tunnel Scope:** indicates the type of automatic optimization performed by Solsoft Policy Server when setting up the tunnel. Although represented as one “logical” tunnel in the LNM, at device level it is possible to set up one or several tunnels to handle the traffic. The Tunnel Scope parameter allows you to control the following three options:
     - **By IP Address** indicates that at device level, a tunnel is established for each pair of IP addresses, range of IP addresses or netmask that need to communicate across it. Allowing or denying specific services in each sub-tunnel is controlled by filters uploaded to the IPsec PEPs. This kind of optimization can reduce the number of negotiated tunnels at the device level,

increasing performance of devices by decreasing the number of tunnels.

- **By IP Address & Service** indicates that at device level, a tunnel is set up for each pair of IP addresses, range of IP addresses or netmask and for each service which needs to communicate across it. Services which are not authorized will have no tunnel established in which case there will be more tunnels negotiated than in the above optimization. This can impact performance of the device.
- **Single Tunnel** indicates that all traffic (all IP addresses and all services) authorized to use the tunnel passes through one single device-level tunnel. Allowing or denying specific services is controlled exclusively by filters uploaded to the IPsec PEPs. This kind of optimization allows only one tunnel, increasing performance. Filters produced to reduce the traffic allowed to use the tunnel may also impact performance.

Note: The tunnel scope calculation is done automatically. From the point of view of the user, you perform the same actions no matter which optimization scheme is in place. The same tunnel policy is applied to all device-level tunnels, and Solsoft Policy Server calculates them and/or the necessary filters for you.

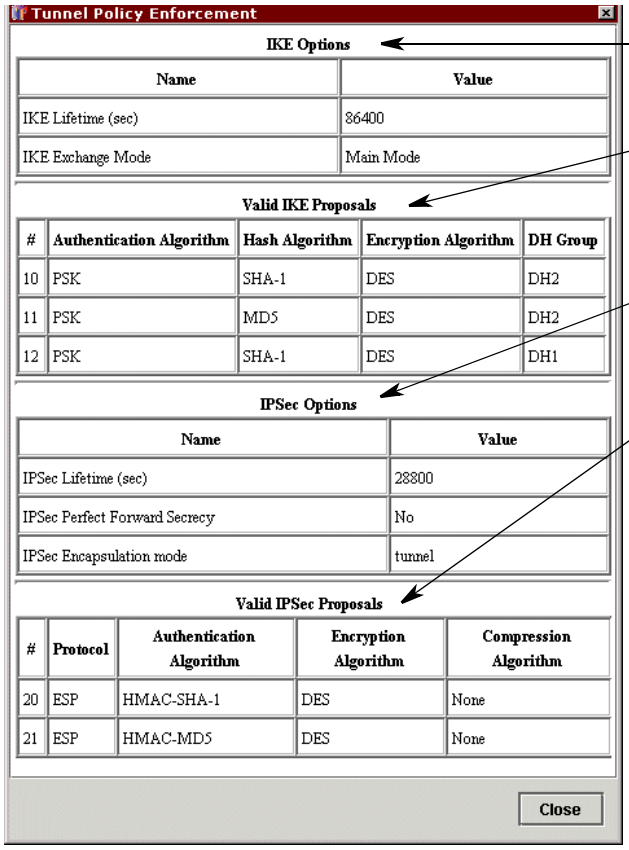
- The two `<PEP_name>` **Interface:** entries indicate which interface on the named device will be the effective tunnel endpoint. The option **All Relevant** allows any interface needing access to the tunnel to be a tunnel endpoint.

2. The **Tunnel Status** area has two components:
   - The **Show Information** button opens a window which shows, on each line, a proposal number (IKE or IPsec) and a reason why the proposal was rejected. This is not blocking unless *all* the proposals are rejected.
   - If the tunnel is invalid for any reason, the button is marked **Show Errors** in red. This corresponds with red display of the dashed lines of the tunnel in the LNM.

Note: A tunnel is invalid if there is no pre-shared key. See "Configure Tunnel Properties" on page 73.

- The **Show Valid IPsec Parameters** button opens the Tunnel Policy Enforcement window, which allows you to see all the proposals that are allowed by both IPsec PEPs (see Figure 16). It displays the options and proposals (IKE and IPsec) from the tunnel policy in force which will actually be uploaded to the IPsec PEP to be used in negotiating a tunnel. Modifications are not possible from this window.

- Click **Close** when you have finished verifying the information.



IKE options in force

Valid IKE proposals (accepted by Solsoft Policy Server for upload to the IPsec PEP)

IPsec options in force

Valid IPsec proposals (accepted by Solsoft Policy Server for upload to the IPsec PEP)

Note: Proposals are shown in order of priority from top down. The proposal number from the tunnel policy is shown in the extreme left hand column of the

Figure 16: Tunnel Policy Enforcement Window

3. To set the options for the devices that are the end-points of the tunnel, click the name of the appropriate device. You can:
   - generate static routing
   - generate filters on the tunnel interface
     By clicking **Yes**, you will enable anti-spoofing.
   - auto generate the tunnel IP address
   - auto generate the tunnel interface name
     To set the beginning number of the interface name, go to the menu bar **Tools>Properties for Current Policy>Generation Options>GRE parameters.**

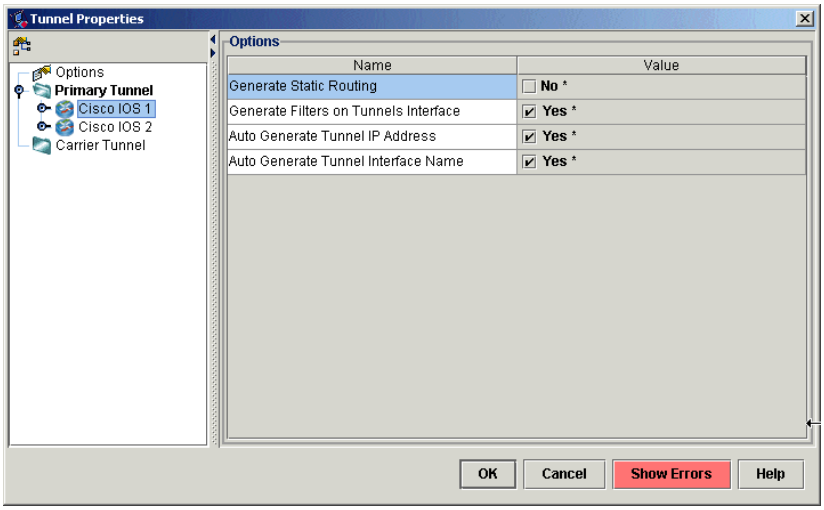


Figure 17: Tunnel Properties Interface options

Note: These parameters are kept and used from one upload to another even if the tunnel has been modified in other ways or another tunnel removed.

Configure Tunnel Properties

To configure the tunnel properties, you simply continue on from the verification process.

From inside the Tunnel Properties Window:

1. Modify parameters that you need to change. See "Verify Tunnel Properties" for explanations of the parameters.
2. The **Tunnel Policy:** combo box gives you the list of all policies previously defined in the Tunnel Policy Editor (those provided by Solsoft and policies you have created yourself). Select a different tunnel policy from the list if needed.
   If none of the policies fits your needs, you can open the Tunnel Policy Editor by clicking the **Edit...** button. See the section "Working with Tunnel Policies" on page 93 for procedures to use.

3. The **Pre-Shared Key:** option allows you to manually create or edit the pre-shared key. If no key exists, the button is marked **Create...** If a key already exists, it will be marked **Edit...**

   You can either generate the pre-shared keys from the Solsoft Security Designer main menu (see "Generate Pre-Shared Keys for Several Tunnels at Once" on page 74), or you can do so one tunnel at a time, as in the following procedure:

   a) Click the **Create...** (or **Edit...**) button. The Pre-Shared Key dialog box opens. Any existing key is displayed in the text box.
   b) To create a new pre-shared key click the **Generate Random Key** button below the text area.
      You can change an existing key by editing it in the text box or by regenerating a new random key using the **Generate Random Key** button.
   c) Click **OK** to close the dialog box when key generation has terminated.

Attention: The pre-shared key *must* be a genuine random key to guarantee security. Words or names such as are often used for passwords are subject to attack and easily crackable.

Generate Pre-Shared Keys for Several Tunnels at Once

The Solsoft Security Designer lets you generate keys for selected tunnels in the workspace all at once. This is a shortcut for opening the properties box of each tunnel and generating its pre-shared key.

In order to do this, you must set an option, one time, on each tunnel.

Procedure

1. Open the tunnel's properties box and, from the tree list, select the **Primary Tunnel >Tunnel Options** view.
2. Make sure the option "**Enable Pre-Shared Key Regeneration**" is checked.
3. Close the tunnel's properties box.
4. Repeat steps 1 - 3 for each tunnel on your workspace for which you want to automate the regeneration of pre-shared keys.
5. From the Solsoft Security Designer menu bar, choose **Action >Regenerate Pre-Shared Keys**.

Defining Permissions in a VPN

Once you have defined and configured your tunnel, you define the traffic which will use it. Here the advantage of the Solsoft approach becomes apparent: to change traffic in a tunnel, you simply change permissions in your map. No need to reconfigure the tunnel itself. Solsoft Policy Server's Network Policy Engine calculates tunnels

and filters as a function of the settings you have defined for the tunnel and for the IPsec PEPs in the trust zone, and the permissions you draw.

Modify VPN-Related Permission Properties

VPN-related permission properties are configured in the same way (as detailed in the *User Guide)* as for other permissions. This section covers configuration of properties which are specific to permissions inside a VPN.

1. Open the properties window for the permission you want to modify by double clicking it OR

   Right click the permission whose properties you want to change and select **Properties...** from the short cut menu OR

   Select the permission whose properties you want to change and click the properties toolbar icon OR

   Select the permission whose properties you want to change and then select **Edit >Properties...** from the menu bar.

   The permission's parameters are displayed in the **<permission_name> Properties** window.
2. Select **Zones & VPNs** from the tree list.
3. Set parameters as follows:
   - **Ignore All VPNs** - if checked, the flow allowed by this permission will not be encrypted and will not pass through the tunnel. It will use the untrusted network as if the tunnel did not exist. This is useful if you have traffic which is passing inside the trust zone but is not of a sensitive nature. You can reduce overhead on the tunnel by sending such traffic in the clear. Permissions configured this way are displayed in the LNM as shown in Figure 18.

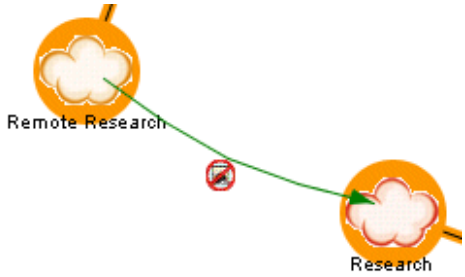


Figure 18: A Permission With Ignore All VPNs Set

   - **Ignore Zones** - if checked, the flow allowed by this permission will pay no attention to either trust zones or limited path zones, even if the source and destination are both inside one or more of the same zones. In the absence of other routing controls, such traffic will take any available physical path between the two objects.

Note: This setting is not strictly speaking VPN-specific, but it appears together with a VPN function. It has been moved from its position in previous editions of Solsoft Policy Server, where it used to be found under **Routing** and was labeled **Ignore Perimeter Maps**.

4. Click **OK** to close the properties window.

Managing Certificates and PKI

In order to authenticate PEPs using secure cryptographic certificates, you must represent certificate authority servers on your workspace, and create public keys for the PEPs. This section describes the procedures required to manage certificates and PKI in Solsoft Policy Server.

Define a VPN with Certificate Authority Servers

1. Create the topology: add the networks and peers that will participate in the VPN. See "Defining a VPN" on page 59.
2. In the Tunnel Policy Editor, create or make sure you have a tunnel policy that has only the “RSA-Sig” method in its IKE Proposals.
3. Create the Trust Zone and the tunnel using the policy defined above. For details on creating Trust Zones, see "Declare a Trust Zone" on page 59.
4. Add CRL distribution point (CDP) servers to your workspace.
   Use the Add Class button to add a class, then open the class’ properties window, type an IP address, and add a CDP server.

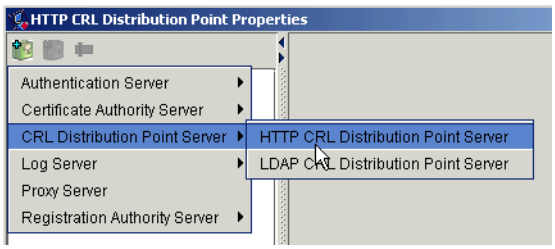


Figure 19: Add a CRL distribution point server to the workspace

   You must know the CDP servers used: the one that is static and those that are written in the generated certificates.
5. Add a CA/RA server to your workspace and reference its associated CDP.

To add the CA/RA server, use the Add Class button to add a class, then open the class' properties window, type an IP address, and add a CA (or RA) server.

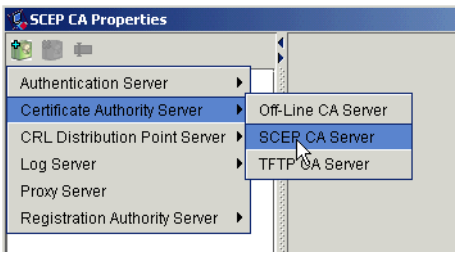


Figure 20: Add a CA server to the workspace

To reference the CDP from the CA/RA server, choose the CDP from the CA server's CRL Distribution List view.

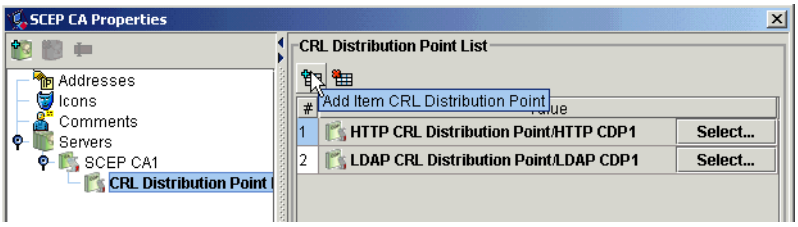


Figure 21: Reference a CDP server from a CA server

You must know the URL if you use SCEP or TFTP.

In case of TFTP, you must ensure that the router is able to write a file to the TFTP server to enroll.

6. Define, if using the SCEP protocol, an HTTP proxy to communicate with the CA or RA.

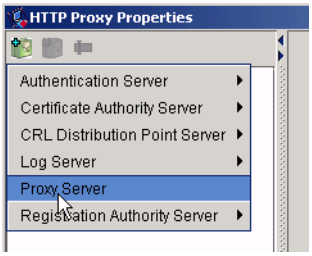


Figure 22: Add an HTTP proxy server to the workspace

You must know if a proxy is used and on which service it is listening.

7. Reference the CA/RA and proxy (if needed) on PEPs with the appropriate enrollment properties.

For a detailed description of these enrollment properties, see PEP Properties in the *Reference Manual* or the online help.

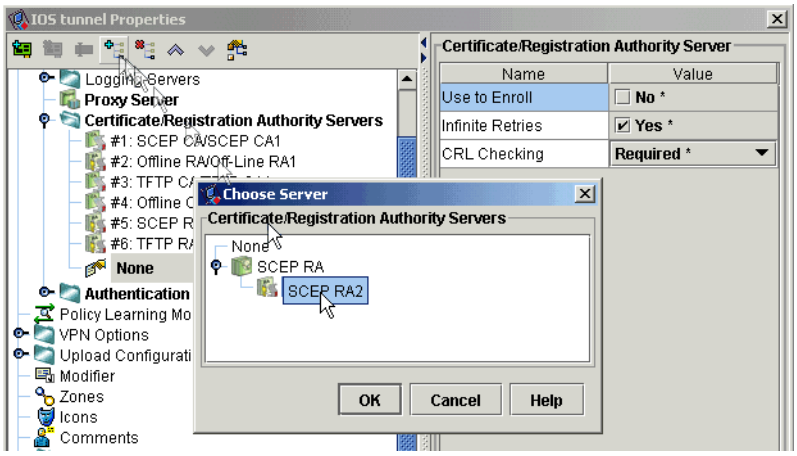


Figure 23: Add an HTTP proxy server to the workspace

8. Define a class that represents an NTP server and create an NTP permission from the PEP to that class.

When you compile and upload this configuration, you can start enrolling your PEPs on the certificate authority servers.

Get a CA's Certificate

Prerequisite

You must have defined a PKI network on your workspace, as described in "Define a VPN with Certificate Authority Servers" on page 76.

Procedure

1. Launch an external commands window.

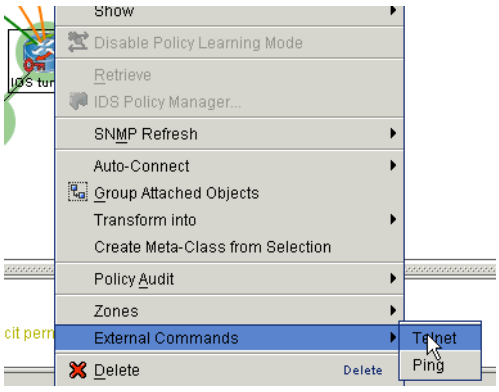


Figure 24: Launching an external command

2. Type one of the commands described below. This step depends on the type of your CA server. You define the type in the **Class properties >Servers >Server** view. The possible types are SCEP, TFTP, and Offline.
   a) On a SCEP server, you request the CA certificate via SCEP with the command: "`crypto ca authenticate <trustpoint name>`".
   This command authenticates the CA to your router by obtaining the self-signed certificate of the CA that contains the public key of the CA. Because the CA signs its own certificate, you should manually authenticate the public key of the CA by contacting the CA administrator when you perform this command.
   In the case of an RA, registration authority signing and encryption certificates will be returned from the CA as well as the CA certificate.
   b) On a TFTP server, you request the CA certificate from the TFTP server via the command: "`crypto ca authenticate <trustpoint name>`".
   This enhanced subcommand specifies that TFTP should be used to send the enrollment requests and to retrieve the certificate of the CA and the certificate of the router. The `file_specification` (**PEP properties >Application Servers >Certificate/Registration Authority Servers >Server view** in the "TFTP File Specification" field) is optional. However, if the `file_specification` is defined, the router will append an extension onto the file specification. When you enter the `crypto ca authenticate` command, the router will retrieve the CA's certificate from the specified TFTP server. As appropriate, the router will append the extension "`.ca`" to the filename or the fully qualified domain name (FQDN). (If the file specification is not defined, the router's FQDN will be used.)
   The file must contain the certificate of the CA in binary or base 64-encoded format.

Table 3: Examples of TFTP certificate file names

TFTP File Specification Field	The router's FQDN	File that the router will read from the TFTP server
`TFTPfiles/router1`	doesn't matter	`TFTPfiles/router1.ca`
<empty>	`router1.cisco.com`	`router1.cisco.com.ca`

   c) On an offline server, you copy/paste the CA certificate via the command: "`crypto ca authenticate <trustpoint name>`".
   For example, you should see something similar on the external commands screen:

```
Router(config)# crypto ca trustpoint MS
Router(ca-trustpoint)# crypto ca authenticate MS
Enter the base 64 encoded CA certificate.
```

```
End with a blank line or the word "quit" on a line
by itself
-----BEGIN CERTIFICATE-----
[...]
-----END CERTIFICATE-----
Certificate has the following attributes:
Fingerprint:D6C12961 CD78808A 4E02193C 0790082A
% Do you accept this certificate? [yes/no]:y
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

Generate a Key Pair for the PEP

Prerequisites

- You must have defined a PKI network on your workspace, as described in "Define a VPN with Certificate Authority Servers" on page 76.
- Before issuing this command, ensure that your router has its host name and IP domain name configured (with the `hostname` and `ip domain-name` commands). You will be unable to complete the `crypto key generate rsa` command without a host name and IP domain name. (This is not true only when you generate a named-key-pair.)

Procedure

1. Launch an external commands window (see Figure 24).
2. Type the following command to generate the keys:

   ```
   crypto key generate rsa [usage-keys | general-keys]
   [ <key-pair-label> ] [modulus <modulus-length> ]
   ```

   This command has the following parameters

Table 4: The PKI command: crypto key generate rsa

Parameter	Description
`usage-keys` (Optional)	Specifies that two special-usage key pairs should be generated, instead of one general-purpose key pair.
`general-keys` (Optional)	Specifies that the general-purpose key pair should be generated. If you generate general-purpose keys, only one pair of RSA keys will be generated. This pair will be used with IKE policies specifying either RSA signatures or RSA-encrypted nonce. Therefore, a general-purpose key pair might be used more frequently than a special-usage key pair.

Table 4: The PKI command: crypto key generate rsa

Parameter	Description
`key-pair-label` (Optional)	Specifies the name of the key pair that router will use. (If this argument is enabled, you must specify either usage-keys or general-keys.). Named key pairs allow you to have multiple RSA key pairs, enabling the Cisco IOS software to maintain a different key pair for each identity certificate.
`modulus length`	When you generate RSA keys, you will be prompted to enter a modulus length. A longer modulus could offer stronger security but takes longer to generate (see Table 1 for sample times) and takes longer to use. A length of less than 512 bits is normally not recommended. (In certain situations, the shorter modulus may not function properly with IKE, so Cisco recommends using a minimum modulus of 1024 bits.)

3. Type the following commands to associate the key pair with the certificates that the PEP will request when it enrolls.

   ```
   crypto ca trustpoint <trustpoint name>
   rsakeypair key-label [key-size [encryption-key-size]]
   ```

   The `rsakeypair` command has the following parameters

Table 5: The PKI command: rsakeypair

Parameter	Description
`key-label`	Will be generated during enrollment if it does not already exist or if the `auto-enroll regenerate` command was issued.
`key-size`	To generate the key and specify the encryption key size to request separate encryption, signature keys, and certificates. The `key-size` and `encryption-key-size` must be equal.

Note: Step 3 is not required if you generated an unlabeled RSA key pair. In that case the unlabeled key pair will be used.

Enroll a PEP on a Certificate Authority Server

Prerequisites

- You must have defined a PKI network on your workspace, as described in "Define a VPN with Certificate Authority Servers" on page 76.

Procedure

1. Launch an external commands window (see Figure 24).
2. Type one of the commands described below. This step depends on the type of your CA server. You define the type in the **Class properties >Servers >Server** view. The possible types are SCEP, TFTP, and Offline.
   a) On a SCEP server, you enroll the PEP via SCEP with the command: "`crypto ca enroll <trustpoint name>`".
   This command requests certificates from the CA for all of your router's RSA key pairs. This task is also known as enrolling with the CA. (Technically, enrolling and obtaining certificates are two separate events, but they both occur when this command is issued.)
   Your router needs a signed certificate from the CA for each RSA key pairs of your router; if you previously generated general purpose keys, this command will obtain the one certificate corresponding to the one general purpose RSA key pair. If you previously generated special usage keys, this command will obtain two certificates corresponding to each of the special usage RSA key pairs.
   If you already have a certificate for your keys you will be unable to complete this command; instead, you will be prompted to remove the existing certificate first. (You can remove existing certificates with the `no certificate` command.)
   For example, you should see something similar on the external commands screen:

```
router(config)# crypto ca enroll myca
%
% Start certificate enrollment..
% Create a challenge password. You will need to
verbally provide this password to the CA
Administrator in order to revoke your certificate.
For security reasons your password will not be saved
in the configuration. Please make a note of it.
Password: <mypassword>
Re-enter password: <mypassword>
% The subject name in the certificate will be:
myrouter.example.com
% Include the router serial number in the subject
name? [yes/no]: yes
% The serial number in the certificate will be:
03433678
```

```
% Include an IP address in the subject name
[yes/no]? yes
Interface: ethernet0/0
Request certificate from CA [yes/no]? yes
% Certificate request sent to Certificate Authority
% The certificate request fingerprint will be
displayed
% The 'show crypto ca certificate' command will also
show the fingerprint.
myrouter(config)#
```

Some time later, the router receives the certificate from the CA and displays the following confirmation message:

```
myrouter(config)# Fingerprint: 01234567 89ABCDEF
FEDCBA98 75543210
%CRYPTO-6-CERTRET: Certificate received from
Certificate Authority
myrouter(config)#
```

If there is a problem with the certificate request and the certificate is not granted, the following message is displayed on the console instead:

```
%CRYPTO-6-CERTREJ: Certificate enrollment request
was rejected by Certificate Authority
```

b) On a TFTP server, generate the request and take it via copy/paste: `crypto ca enroll <trustpoint name>`“.
When you enroll the router via the `crypto ca enroll command`, you are prompted for information regarding the enrollment. The filename that is to be written is already determined at this point, and an extension of "`.req`" is appended to indicate that this is a certificate request.
For usage keys, two requests are generated and two certificates are expected to be granted. Thus, the extension for the certificate requests are "`-sign.req`" and "`-encr.req`".
Import the certificates generated from the CA via copy/paste:
`crypto ca import <trustpoint name> certificate`
After you enter the `crypto ca import command`, the router will attempt to fetch the granted certificate via TFTP using the same filename that was used to send the request, except that "`.req`" extension will be replaced by a "`.crt`" extension. (The certificates are expected to be base 64-encoded PCKS#10 format certificates.) The router will parse the files it receives, verify the certificates, and insert the certificates into the internal certificate database.

c) On an offline server, Generate the request and take it via copy/paste: `crypto ca enroll <trustpoint name>`.
You may wish to manually cut-and-paste certificate requests and certificates when you do not have a network connection between

the router and CA. After entering the `crypto ca enroll` command, the base 64-encoded certificate request will then be displayed on the terminal.
For example, you should see something similar on the external commands screen:

```
Router(config)#crypto ca enroll MS
% Start certificate enrollment..
% The subject name in the certificate will
be:Router.cisco.com
% Include the router serial number in the subject
name?
[yes/no]:n
% Include an IP address in the subject name? [no]:n
Display Certificate Request to terminal? [yes/no]:y
Signature key certificate request -
Certificate Request follows:
[...]
---End - This line not part of the certificate
request---
Redisplay enrollment request? [yes/no]:
Encryption key certificate request -
Certificate Request follows:
[...]
---End - This line not part of the certificate
request---
Redisplay enrollment request? [yes/no]:
n
```

Import the certificates generated from the CA via copy/paste:
`crypto ca import <trustpoint name> certificate.`
You enter the `crypto ca import` command to enter the granted certificate. With cut-and-paste, the base 64-encoded certificate will be accepted from the console terminal. Certificate input ends after you enter "quit" on a line by itself.
For example, you should see something similar on the external commands screen:

```
Router(config)#crypto ca import MS certificate
Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line
by itself
[...]
% Router Certificate successfully imported
Router(config)#
Router(config)#crypto ca import MS certificate
```

```
Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line
by itself
[...]
% Router Certificate successfully imported
```

Verify your PKI Configuration

After performing manual certificate enrollment (via TFTP or cut-and-paste), verify your configuration to be sure that you successfully completed all steps. Use the following commands to do this:

The command `show crypto ca certificates` verifies information about your certificate.

The command `show crypto ca trustpoints` displays the trustpoints that are configured in the router.

Remove the CA from the Router

If you do not want to use a CA anymore, you must connect to the device and explicitly remove the trustpoint configuration using the following command:

```
no crypto ca trustpoint <trustpoint name>
```

Remove a Key Pair from the Router

If you want to remove the key pair generated on the PEP:

```
crypto key zeroize rsa [<key-pair-label>]
```

Note: This command cannot be undone. After you save your configuration, and after RSA keys have been deleted, you cannot use certificates or the CA or participate in certificate exchanges with other IP Security (IPsec) peers unless you reconfigure CA interoperability by regenerating RSA keys, getting the CA's certificate, and requesting the PEP's certificate again.

Checking Your Work

This section is supplementary to the information provided in the *User Guide* on this subject. Refer to that document if you are not already familiar with it.

There are two VPN-related checking procedures:

- Show Generated VPN Configuration
- Through Tunnel Audit.

Show Generated VPN Configuration

This function enables you to see formatted displays of text files generated by the NPE compilation process.

This function will display the accepted proposals from the tunnel policy, IKE and IPsec options, permissions and other useful parameters in a generic format that allows you to manually configure the IPsec device:

1. Right click an IPsec PEP which is a tunnel endpoint.
2. From the shortcut menu, select **Show >Generated VPN Configuration**.
3. If you have not compiled your filters since last modifying them, you will see a message box asking you if you want to compile. Click **Compile** to proceed with the compilation.

Attention: If you click **Ignore** the display will show the *last compiled* generic VPN configuration, and *not* the work you are currently doing.

4. If you select **Compile**, you will see the Compilation Finished window displaying the results of the compilation:
   - If the compilation fails, close the window and fix the problem, then try again.
   - If the compilation succeeds, immediately after you close the Compilation Finished window the Show Generated VPN Configu-

ration window opens. See Figure 25.

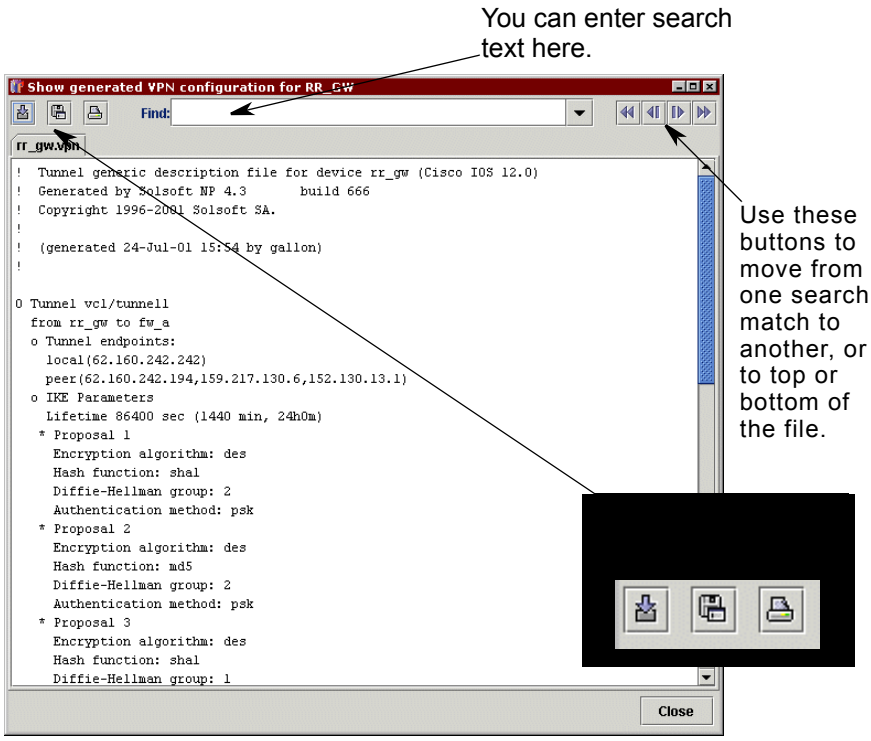


Figure 25: Show Generated VPN Configuration Window

See "The .VPN File" on page 90 for a guide to reading the file.

Through Tunnel Audit

Once you have defined your VPN and the permissions that will use the tunnel, assess your work using the Solsoft Policy Server audit feature. This will display all permissions allowed to pass through the tunnel. Only “Through” audits are valid on a tunnel.

Prerequisites

- The tunnel to be audited must be valid.
- Solsoft recommends that you compile filters before auditing. Error messages and warnings at compilation time are a valuable first check for problems and will trap VPN errors that would otherwise occur at audit time.

Procedure

1. Right click on the tunnel to be audited.
2. Select **Policy Audit >Through...** from the shortcut menu. A message box advises you to wait for the results, with the option to **Cancel** the operation by clicking the button.
3. The result is displayed in the Audit Results window. You will see one of two displays:

- If you have not successfully compiled the filters before auditing, and there is a problem, you will see compiler error and warning messages, similar to Figure 26. Close the window and correct the error(s), then try the audit again.
- If you have no errors, you will see a window similar to Figure 27. Refer to the *User Guide* for details on how to read an audit.

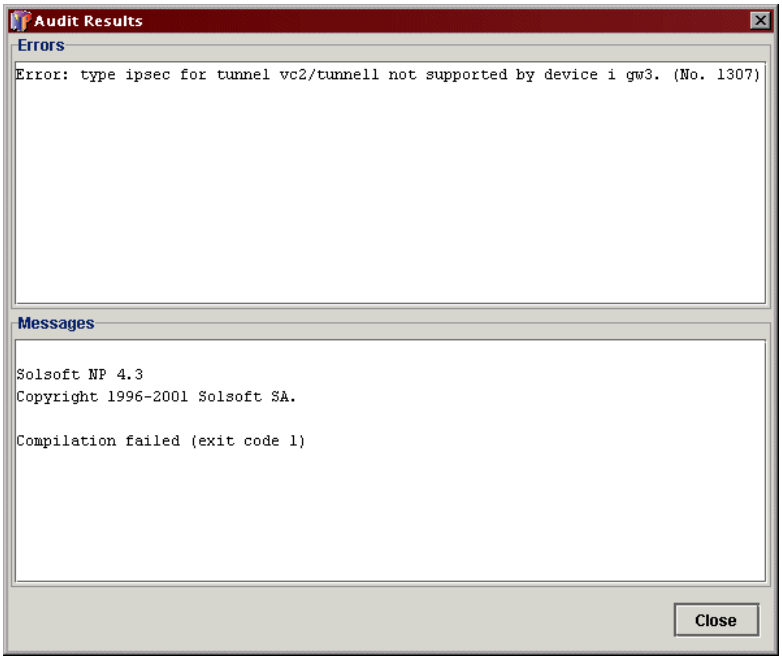


Figure 26: Audit Results Error Window

Attention: Error messages will occur if *any* tunnel is invalid, not just the tunnel being audited, or for non-VPN related reasons.

Note: If you need to communicate with Solsoft support to resolve your problem, the error number in parenthesis at the end of the message will help support personnel to quickly identify the error.

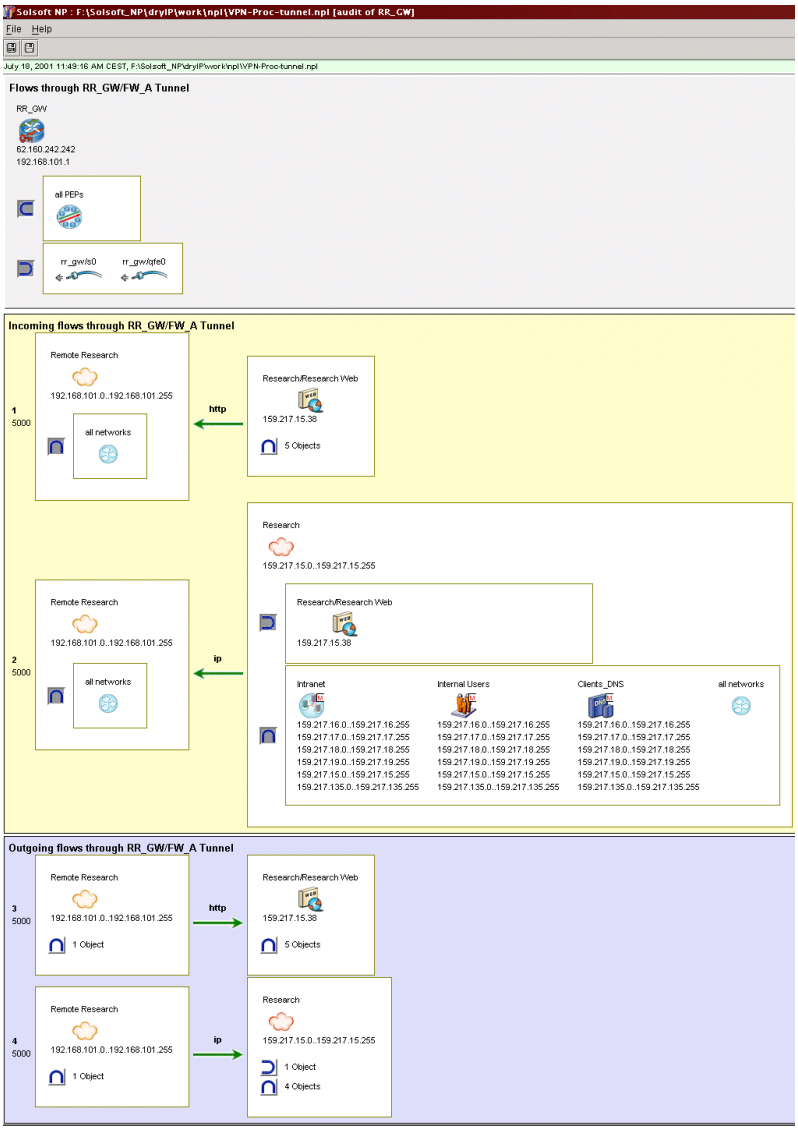


Figure 27: Successful Tunnel Audit Results

4. Go on to the next task.

Compile and Upload

Compiling and uploading filters for VPNs are automatically included in the global compile and upload processes. These are detailed in the *User Guide*.

The .VPN File

When filters are compiled, an additional file is generated for each device serving as a tunnel endpoint. This file is found in the same directory as the filter files (by default `<NPDIR>/work/filters` where `<NPDIR>` is the top directory where Solsoft Policy Server is installed). It is named as follows:

```
<devicename>.vpn
```

where `<devicename>` is the name of an IPsec PEP which is a tunnel endpoint.

This is the same file displayed by the Show Generated VPN Configuration function.

Case When One Tunnel Endpoint is an Unmanaged IPsec PEP

In the case where a tunnel in the LNM connects a managed IPsec PEP with an unmanaged IPsec PEP, NPE generates a `.vpn` file containing for both devices.

The file is human readable and can be edited in a text editor. If the capabilities of the unmanaged IPsec PEP have been correctly configured in the LNM, the `.vpn` file presents, in generic form, all the necessary information to manually configure the unmanaged device so that it will be compatible with its own capabilities and with the tunnel policy configured in Solsoft Policy Server.

Case When One Tunnel Endpoint is a Nexus

A `.vpn` file is only generated for the IPsec PEP end of the tunnel, whether the PEP is managed or not. No file can be generated for a nexus.

Since nothing is known by Solsoft Policy Server about the device represented by a nexus, it is assumed to be capable of everything, and only the capabilities of the managed device are taken into account in generating the tunnel configuration.

Note: Since Solsoft Policy Server can not know anything about the nexus, it is a good idea to be informed in advance of its capabilities, and to take them into account when choosing or creating a tunnel policy for the managed IPsec PEP.

The `.vpn` file for the IPsec PEP can then be exchanged with the other side of the tunnel by any normal means that you choose (email, fax, etc.) to help the other side configure its device.

Header Information

Here is an example of a `.vpn` file for communication between a managed Cisco Secure PIX firewall and an unmanaged Cisco IOS router:

```
: Tunnel generic description file for device fw_b (Cisco
Secure PIX Firewall 5.3)
: Generated by Solsoft 5.3 build 002
: Copyright 1996-2003 Solsoft SA.
:
: (generated 24-Jul-03 14:50 by user)
:
```

Global Tunnel Data

```
O Tunnel vc4/tunnel1
  from fw_b to outsourcing_net
  o Tunnel endpoints:
    local(62.160.242.205,152.130.13.35,159.217.130.45)
    peer(183.234.17.5)
```

IKE Parameters and Proposals

```
 o IKE Parameters
    Lifetime 86400 sec (1440 min, 24h0m)
   * Proposal 1
     Encryption algorithm: des
     Hash function: sha1
     Diffie-Hellman group: 2
     Authentication method: psk
   * Proposal 2
     Encryption algorithm: des
     Hash function: md5
     Diffie-Hellman group: 2
     Authentication method: psk
   * Proposal 3
     Encryption algorithm: des
     Hash function: sha1
     Diffie-Hellman group: 1
     Authentication method: psk
  o Pre-Shared Key:
zVbZ_`1gumff:mP@^:0Ho7A%gnPaw%zn09fe%5ho/Dp9f$vBPO/
Ra]Qhp_@'^umtSV&b;_:+,KSTq9O2tU'79$^;9<6)94n;CZW__ql3p>[d}D
d_9t:=p2;3&xvvIYP
  o IKE identities: IP addresses (IPV4_ADDR)
```

IPsec parameters and proposals

```
o IPsec Parameters
    Lifetime 28800 sec (480 min, 8h0m)
    Perfect Forward Secrecy: no
   * Proposal 1
```

```
Protocol:esp
Mode: tunnel
Authentification algorithm: hmac-sha1
Encryption algorithm: des
Compression algorithm: null
```

Tunnel negotiated at the device level and associated permissions

```
 o Tunnel scope
  allow from(159.217.19.0/159.217.19.255) to(183.234.19.0/
183.234.19.255) ip-proto(all) srcPort(all) destPort(all)
    allow from(159.217.19.0/159.217.19.255) to(183.234.19.0/
183.234.19.255) ip-proto(all) srcPort(all) destPort(all)
  o Traffic allowed inside tunnels:
    allow from(159.217.19.0/159.217.19.255) to(183.234.19.0/
183.234.19.255) ip-proto(TCP[6]) srcPort(21)
destPort(1024..65535)
    allow from(159.217.19.0/159.217.19.255) to(183.234.19.0/
183.234.19.255) ip-proto(TCP[6]) srcPort(20)
destPort(1024..65535)
    allow from(159.217.19.0/159.217.19.255) to(183.234.19.0/
183.234.19.255) ip-proto(TCP[6]) srcPort(20)
destPort(1024..65535)
    allow from(159.217.19.0/159.217.19.255) to(183.234.19.0/
183.234.19.255) ip-proto(all) srcPort(all) destPort(all)
  o Explicitly unencrypted permissions:
    allow from(62.160.242.205)
to(62.160.242.245,183.234.17.5) ip-proto(UDP[17])
srcPort(500) destPort(500)
    allow from(62.160.242.205)
to(62.160.242.245,183.234.17.5) ip-proto(ESP[50])
srcPort(all) destPort(all)
    allow from(62.160.242.205)
to(62.160.242.245,183.234.17.5) ip-proto(AH[51])
srcPort(all) destPort(all)
    allow from(62.160.242.205)
to(62.160.242.245,183.234.17.5) ip-proto(UDP[17])
srcPort(500) destPort(500)
    allow from(62.160.242.205)
to(62.160.242.245,183.234.17.5) ip-proto(ESP[50])
srcPort(all) destPort(all)
    allow from(62.160.242.205)
to(62.160.242.245,183.234.17.5) ip-proto(AH[51])
srcPort(all) destPort(all)
    allow from(152.130.13.35)
to(62.160.242.245,183.234.17.5) ip-proto(UDP[17])
srcPort(500) destPort(500)
    allow from(152.130.13.35)
to(62.160.242.245,183.234.17.5) ip-proto(ESP[50])
srcPort(all) destPort(all)
```

```
    allow from(152.130.13.35)
to(62.160.242.245,183.234.17.5) ip-proto(AH[51])
srcPort(all) destPort(all)
    allow from(152.130.13.35)
to(62.160.242.245,183.234.17.5) ip-proto(UDP[17])
srcPort(500) destPort(500)
    allow from(152.130.13.35)
to(62.160.242.245,183.234.17.5) ip-proto(ESP[50])
srcPort(all) destPort(all)
    allow from(152.130.13.35)
to(62.160.242.245,183.234.17.5) ip-proto(AH[51])
srcPort(all) destPort(all)
    allow from(159.217.15.0/159.217.15.255)
to(192.168.101.0/192.168.101.255) ip-proto(TCP[6])
srcPort(23) destPort(1024..65535)
    allow from(159.217.15.0/159.217.15.255)
to(192.168.101.0/192.168.101.255) ip-proto(TCP[6])
srcPort(23) destPort(1024..65535)
    allow from(192.168.101.0/192.168.101.255)
to(159.217.15.0/159.217.15.255) ip-proto(TCP[6])
srcPort(1024..65535) destPort(23)
```

Working with Tunnel Policies

Tunnel policies provide the means to define common parameters that will be shared by various tunnels. GRE tunnel policies have no common parameters except being of the same type. IPsec tunnel policies provide lists of IKE and IPsec proposals which Solsoft Policy Server is authorized to configure for a given tunnel. Solsoft provides a number of pre-defined tunnel policies which provide different levels of security. You can use one of these, modify it to suit your needs, or create your own tunnel policy from an empty template.

Note: Tunnel policies are most often constructed to provide a guaranteed minimum level of security.

At least one proposal option should provide the minimum security level that you will accept when using this policy.

At least one more should provide the maximum level that you will accept (for example, it is necessary to balance security needs with throughput speeds, and this may limit how strong a level of encryption you want to use, even under the best of circumstances).

Open the Tunnel Policy Editor

1. Select **Tools >Tunnel Policy Editor** from the menu bar. The Tunnel Policy Editor window opens. Copies of the pre-defined tunnel policies provided by Solsoft already appear in the tree list.

2. Do one of the following:
   - Double click an existing policy you want to modify (or click the key to its left) to expand the tree list for the policy.
   - Create a new template copy from which to develop a new policy.

Create a New Copy of a Template

Note: If this is the first time you are using a given template, a copy already exists in the Tunnel Policy Editor. You do not need to create a new copy, though you probably will want to rename the existing template (go directly to Step 2 below).

1. Click the **Add a Tunnel Policy** action button  above the tree list. A menu appears with the different pre-defined templates. Select the one you want to use as a basis for your new tunnel policy.

Note: Use **Empty** if you want to build a tunnel policy from scratch.

2. Select the new tunnel policy and click the **Rename a Tunnel Policy** action button  . The Rename Tunnel Policy dialog box appears.
3. Enter the name for the tunnel policy in the field.
4. Click **OK** on the dialog box.

Define IKE Options and Proposals

1. Double click your new empty tunnel policy or click on the key to its left to expand its tree list.
   The **Empty** tunnel policy is a blank template allowing you to generate custom tunnel policies for situations not treated by the pre-defined ones provided by Solsoft.
2. Click **IKE Options** in the tree list. Enter the IKE lifetime in seconds in the **IKE Lifetime** text entry box. The IKE exchange mode cannot be modified.
3. Click **IKE Proposals** in the tree list. The parameters area is empty.
4. Click the **New Proposal** action button  in the **Proposal List** pane. The parameters area fills with data and controls.
5. Select the hash and encryption algorithms to use to protect IKE communications, the Diffie-Hellman group number, and the authentication method.
6. Configure keepalives. Keepalives are messages that the PEPs send periodically to verify that the peer is still active, and hence that the tunnel is working. Note that keepalives are called “Dead Peer Detection” on recent versions of Cisco PEPs.
   d) Choose whether or not to enable the keepalive function.
   e) Enter an idle keepalive value: this is the interval, in number of seconds, after which the PEPs send keepalive messages to verify that their peer is still online and maintaining the tunnel.
   f) Enter a keepalive retry value: this is the interval after which the PEP will resend a keepalive message if it does not get a response.

The minimum and maximum keepalive values depend on your device.

7. Repeat Steps 4 to 6 to create as many proposal options as you need.
8. Use the arrow buttons above the list to change the position of the proposal in the list. Select the proposal you want to move, then use the up arrow to move the proposal up the hierarchy, or the down arrow to move it down.

Note: Remember that the proposals are offered in their list order, and the first acceptable proposal will be uploaded to the device as the first priority proposal. Each subsequent acceptable proposal will be uploaded in the next priority order to the limit of the number of proposals in the list, or the capacity of the device.

Define IPsec Options and Proposals

1. If you are not already in your new tunnel policy, double click it or click on the key to its left to expand its tree list.
2. Click **IPsec Options** in the tree list. Enter values as follows:
   - Enter the IPsec lifetime in seconds in the **IPsec Lifetime** text entry box.
   - Select the Diffie-Hellman group for Perfect Forward Secrecy from the **IPsec Perfect Forward Secrecy** combo box.

Note: The option **First valid IKE group** will reuse the first valid IKE proposal in the same tunnel policy as the PFS value.

The value **None** disables PFS.

   - The IPsec Encapsulation mode cannot be modified.
3. Click **IPsec Proposals** in the tree list. The parameters area is empty.
4. Click the **New Proposal** action button  in the **Proposal List** pane. The parameters area fills with data and controls.
5. Select the protocol to use (AH or ESP) and the authentication method for data.

   With ESP, you also choose the encryption algorithm to protect data communications.
6. You can also choose a compression algorithm to use on data before encapsulation, or, if you prefer, no compression (this is the default).
7. Repeat Steps 4 - 8 to create as many proposal options as you need.
8. Use the arrow buttons above the list to change the position of the proposal in the list. Select the proposal you want to move, then use the up arrow to move the proposal up the hierarchy, or the down arrow to move it down.

Note: Remember that the proposals are offered in their list order, and the first acceptable proposal will be uploaded to the device as the first priority proposal. Each subsequent acceptable proposal will be uploaded in the next priority order to the limit of the number of proposals in the list, or the capacity of the device.

9. Do one of the following:
   - Click **OK** to close the Tunnel Policy Editor
   - Go on making other changes in the Tunnel Policy Editor.

Modify or Add IKE Options and Proposals

1. Double click the tunnel policy you wish to modify or click the key to its left to expand its tree list.
2. Select **IKE Options** in the tree list. Modify the IKE lifetime in seconds in the **IKE Lifetime** text entry box. The IKE exchange mode cannot be modified.
3. Click **IKE Proposals** in the tree list. In the **Proposal List** pane, select the proposal you wish to modify.
4. Change values as needed in the **Proposal Description** pane.
5. If you wish to add a new proposal, click the **New Proposal** action action button  in the **Proposal List** pane.
   In the **Proposal Description** pane, select the hash and encryption algorithms to use to protect IKE communications, and the Diffie-Hellman group number. The authentication method cannot be modified.
6. If you wish to delete a proposal, select it and click the **Delete Selected Proposal** action button  in the **Proposal List** pane.
7. When you have finished making modifications, reorder the proposals according to priority using the up and down arrow action buttons (see *Define IKE Options and Proposals* for more details).

Modify or Add IPsec Options and Proposals

1. Double click the tunnel policy you wish to modify or click the key to its left to expand its tree list.
2. Select **IPsec Options** in the tree list. Modify settings as follows:
   - Modify the IPsec lifetime in seconds in the **IPsec Lifetime** text entry box.
   - Change the Diffie-Hellman group for Perfect Forward Secrecy in the **IPsec Perfect Forward Secrecy** combo box.

Note: The option **First valid IKE group** will reuse the first valid IKE proposal in the same tunnel policy as the PFS value.

The value **None** disables PFS.

   - The IPsec Encapsulation mode has three choice:
     - Automatic: this will set Transport mode only when it is a carrier tunnel.

3. Click **IPsec Proposals** in the tree list. In the **Proposal List** pane, select the proposal you wish to modify.
4. Change values as needed in the **Proposal Description** pane.
5. If you wish to add a new proposal, click the **New Proposal** action button  in the **Proposal List** pane.

   In the **Proposal Description** pane, select the protocol to use (AH or ESP) and the authentication method for data.

   With ESP, you also choose the encryption algorithm to protect data communications.

   You can also choose a compression algorithm to use on data before encapsulation, or, if you prefer, no compression (this is the default).
6. If you wish to delete a proposal, select it and click the **Delete Selected Proposal** action button  in the **Proposal List** pane.
7. When you have finished making modifications, reorder the proposals according to priority using the up and down arrow action buttons (see *Define IPsec Options and Proposals* for more details).
8. Do one of the following:
   - Click **OK** to close the Tunnel Policy Editor
   - Go on making other changes in the Tunnel Policy Editor.

Delete a Tunnel Policy

1. In the tree list, click the policy you wish to delete.
2. Click the **Remove a Tunnel Policy** action button  . The tunnel policy is deleted.
3. Do one of the following:
   - Click **OK** to close the Tunnel Policy Editor
   - Go on making other changes in the Tunnel Policy Editor.

Working with Fully-Meshed and Hub and Spoke Tunnels

This feature lets you define a global VPN more easily and rapidly in only one action. It allows you to share the same tunnel properties among all the PEPs contained in a meta-class.

This feature is available only with the Solsoft Policy Server Enterprise Edition.

This feature is not available for a Remote VPN.

Note: If you want to set up fully-meshed or hub-and-spoke VPNs, and you have Cisco IOS devices, Solsoft recommends that you use the DMVPN feature. DMVPN is a Cisco feature simplifies grouped-tunnel VPNs. See the Solsoft *Working with Cisco IOS and HSRP* document for more information on DMVPN.

Create a Fully-Meshed VPN

1. Create the topology that contains the VPN devices and the networks that participate in the VPN.
2. Associate the networks to protect and the PEP in a Trust Zone.
3. Create a meta-class that contains all the PEPs that participate in the fully-meshed VPN.
4. On the toolbar, click the Tunnel button.

5. Point the cursor to the meta-class, and click the cursor so that a looped tunnel appears. This “loopback” tunnel is a tunnel between the meta-class and itself. This allows tunnels to be fully-meshed because each PEP has a tunnel going to every other PEP included the meta-class.

6. Open the tunnel properties box to set its properties.

7. The specific parameter **Use the Shortest Encryption Path** appears only when one end of the tunnel is a meta-class. This parameter allows specifying if the encrypted traffic between peers contained in the meta-class can use only one tunnel or can use several tunnels to reach its final destination.

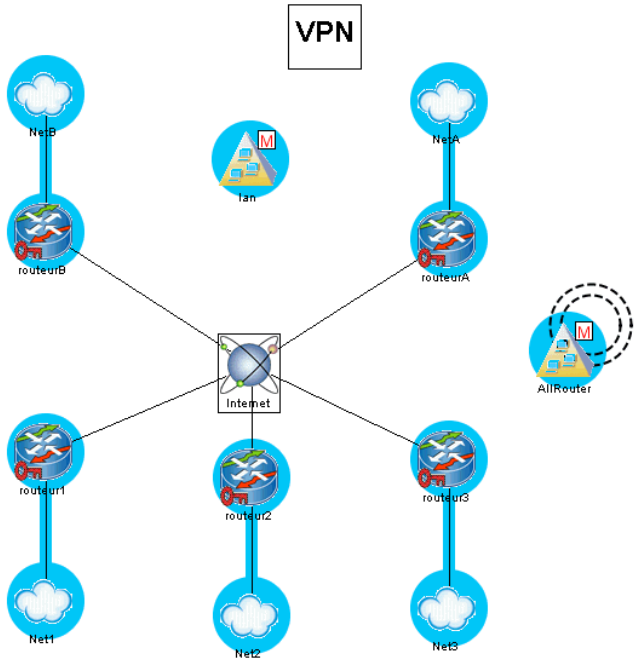


Figure 28: An Example of a Fully Meshed Tunnel Group

Creating a Hub and Spoke

In the figure below a hub and spoke VPN configuration has been made for the entire network.

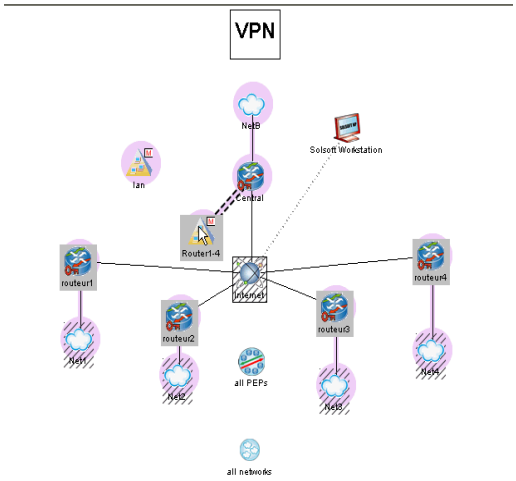


Figure 29: An Example of a Hub and Spoke Tunnel Group

1. Fill in the property box for the tunnel correctly.

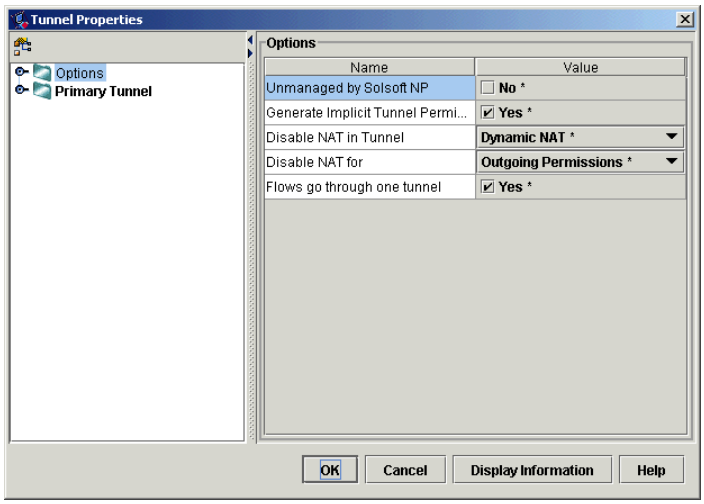


Figure 30: Tunnel Properties: Primary Tunnel: All Relevant

Limitations

- All PEPs contained in a meta-class must be of the same type. Otherwise, the tunnel is considered invalid.

Tunnel properties

- The tunnel properties windows will allow setting all the properties of a current tunnel except the interfaces.

- A parameter **Use the Shortest Encryption Path** indicates if encrypted traffic can use several or only one tunnel. This property appears only when one end of the tunnel is a meta-class that will be the relevant interface.

Chapter 5: Sample Use Cases

This chapter gives examples of two cases where VPN tunnels are used to secure communications.

A Small Company/Branch/Supplier Network

Figure 31 shows a company network with one branch, represented by the networks `Headquarters` and `SubDivision`. Each is protected by an IPsec PEP, and connected through the Internet.

In addition, the company has very close relations with a supplier, represented by the network `OEM Supplier`. The supplier network is protected by a third IPsec PEP which is managed by Solsoft Policy Server.

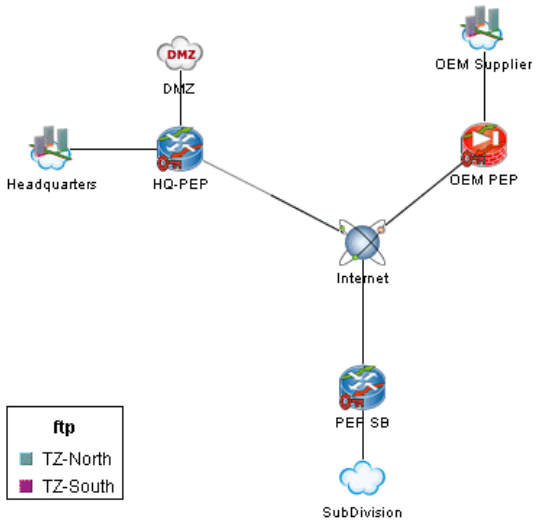


Figure 31: A Company/Branch/Supplier Network

The secure communications needs are as follows:

- All services need to pass freely but securely between `Headquarters` and `SubDivision`.
- The `OEM Supplier` network needs access to an Oracle database and to a SAP server inside `Headquarters` and no other services.

Build the VPNs

To meet these needs, two VPNs will be constructed:

- One for open but secure internal communication between company branches
- One for limited secure communications with the outside supplier

This requires the construction of two trust zones, each with an associated tunnel as shown in Figure 32.

Each trust zone contains only elements that need to communicate with each other. Note that some objects are assigned to both trust zones.

The tunnels connect the two parts of each trust zone separated by the Internet as follows:

- Between `HQ-PEP` and `PEP SB`
- Between `HQ-PEP` and `OEM PEP`

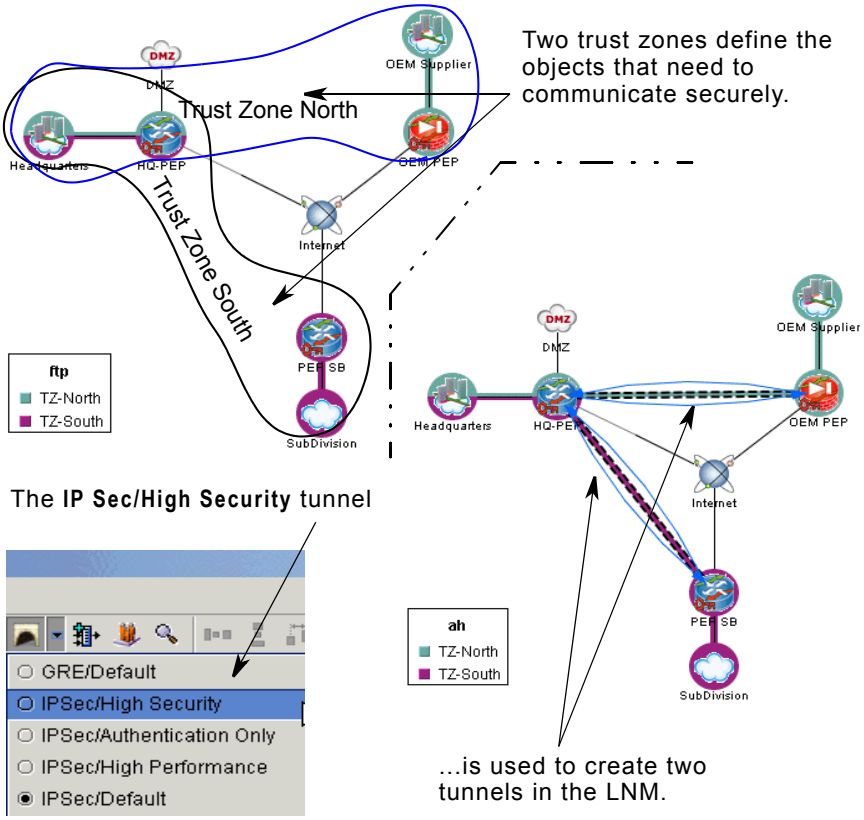


Figure 32: Two Trust Zones and Two Tunnels Declared

Note: Since the DMZ provides service to the outside, it is not inside any trust zone.

Trust zone procedures are found starting on page 59.

Procedures for building tunnels are found starting on page 63.

Open Permissions

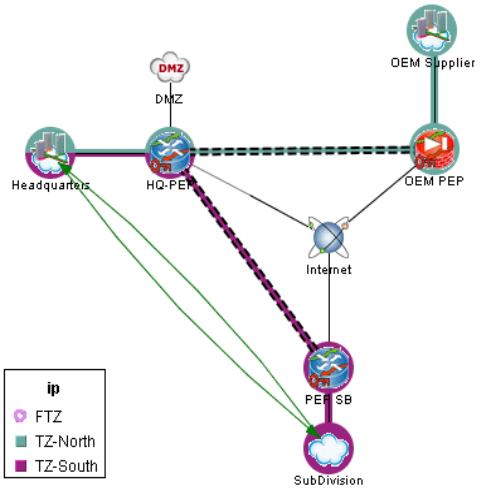
There are three services needed, as explained above:

- IP
- SQLNET (for the Oracle database)
- SAP

These serve two different groups of users: internal and external. Procedures for opening permissions in a VPN are found starting on page 75.

Provide Internal Service

Open an `ip` permission between `Headquarters` and `SubDivision` as shown in Figure 33.



This permission allows all machines in headquarters and subdivision locations to use all `ip` services in either direction.

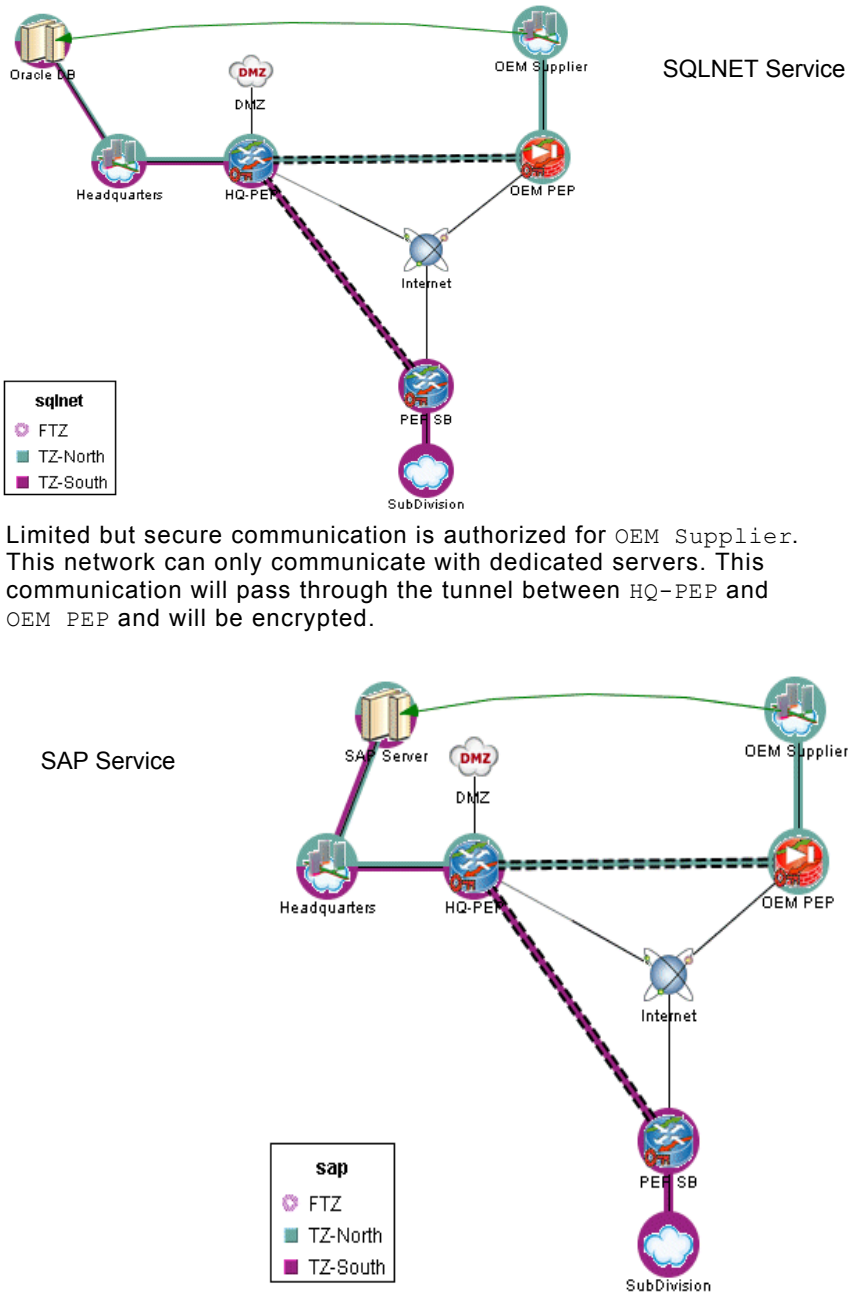
All such communication will pass through the tunnel between `HQ-PEP` and `PEP SB` and will be encrypted.

Figure 33: IP Flow For Internal Service

Provide External Service

Create two classes inside `Headquarters`. One for the Oracle server, the other for SAP service.

Open flows for `sqlnet` from `OEM Supplier` to the class `Oracle DB` (inside `Headquarters`) and flows for `sap` from `OEM Supplier` to the class `SAP Server` (inside `Headquarters`) as shown in Figure 34.



Limited but secure communication is authorized for `OEM Supplier`. This network can only communicate with dedicated servers. This communication will pass through the tunnel between `HQ-PEP` and `OEM PEP` and will be encrypted.

Figure 34: IP Flow For External (OEM) Service

Audit the Tunnels

To verify the flows which will pass through the tunnel, use the audit function. Do a through audit of each tunnel (see extract, Figure 35), and a through audit on all interfaces of each PEP (see extract, Figure 36).

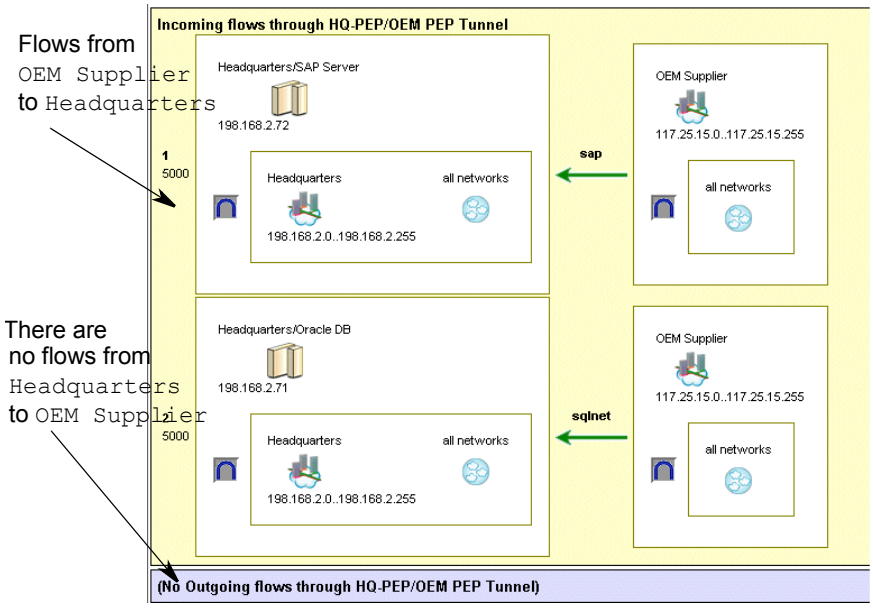


Figure 35: Through Audit of Tunnel from `HQ-PEP` to `OEM PEP` (extract)

The interest of doing these two audits is:

- With the audit of the tunnel, you see all the flows which pass through it.
- With the audit of the PEP, you see flows that pass through, and flows that do not pass through, the tunnel, and can easily identify if there are errors.

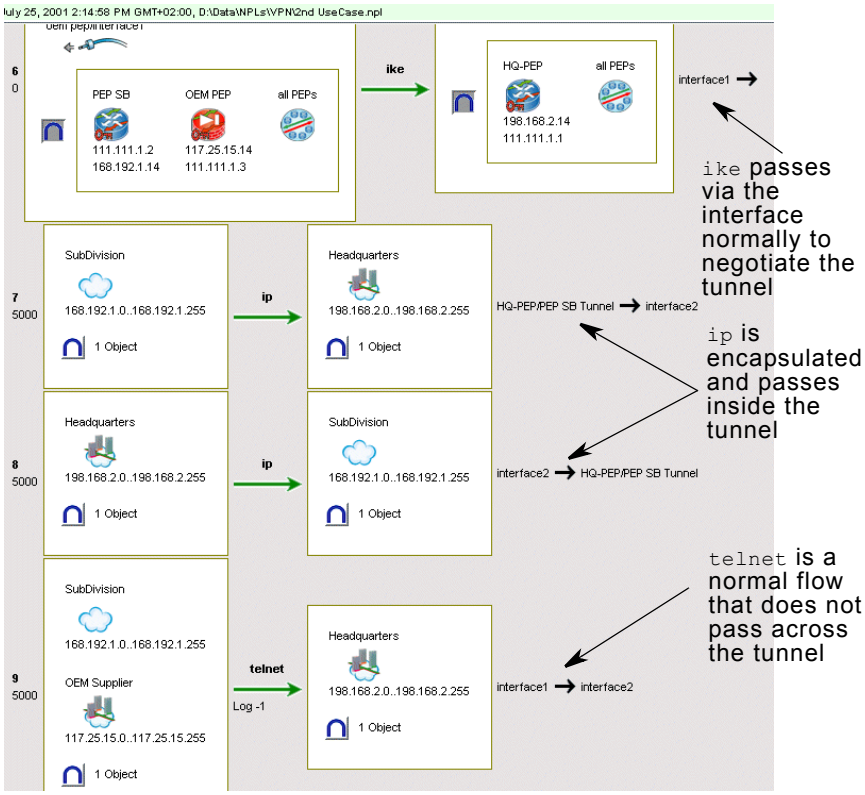


Figure 36: Through Audit of `HQ-PEP` (all interfaces - extract)

Once the VPNs and permissions have been defined and verified, it's time to compile and upload the filters to the PEPs. This is explained in the *User Guide*.

A Larger Distributed Network

Figure 37 shows a company network with distributed operations and an outside manufacturer. The research function is distributed across two sites, represented by the networks `Research` and `Remote Research`.

The manufacturing function is distributed between a local site (`manufact`) and an external supplier (`Manufacturer`).

Each distributed function needs secure communication across the Internet.

The company internal networks are protected by an IPsec PEP firewall (`PIX`). Between the firewall and the Internet are a DMZ and a network of proxy servers, both routed to the via a redundant pair of PEPs (`I GW1` and `I GW3`). These are contained in a limited path zone in order to prevent packets from one internal address to another from passing via the Internet unnecessarily.

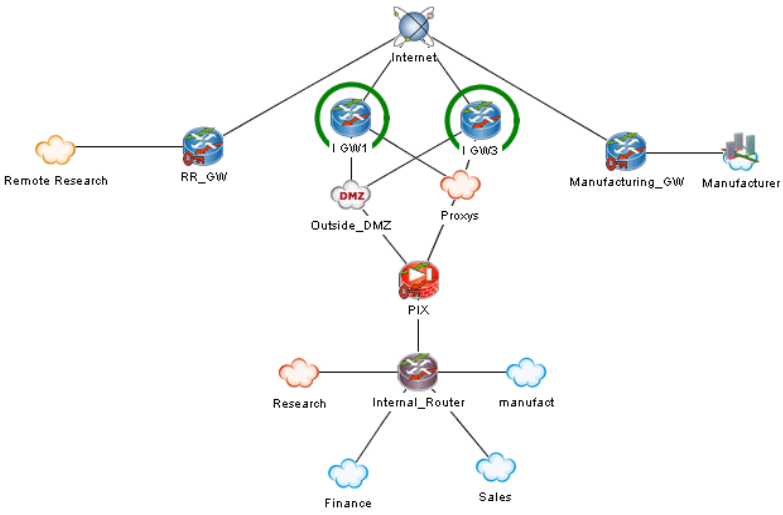


Figure 37: A Distributed Network With Outside Manufacturer

Build the VPNs

As in the previous example, each pair of sites needing secure communication is placed inside a trust zone: one for research, one for manufacturing. The tunnels to build will connect the following PEPS (see Figure 38):

- `RR_GW` and `PIX`
- `Manufacturing_GW` and `PIX`

Trust zone procedures are found starting on page 59.

Procedures for building tunnels are found starting on page 63.

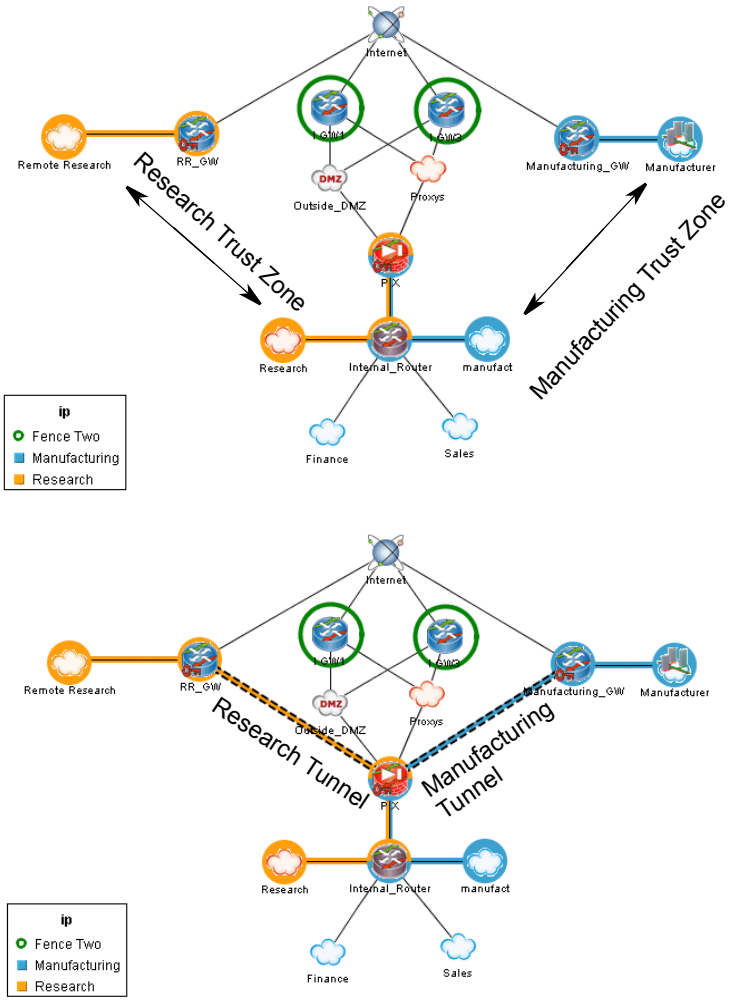


Figure 38: Trust Zones and Tunnels For Distributed Functions

The effect of these actions is to create several distinct sections, or partitions in the network:

- An internal network at the main site
- A distributed research network between two company sites
- A distributed manufacturing network between a company site and an outside supplier
- An outside communication area, to provide servers and proxies available to the general public and partners who do not need secure communications with the internal networks.

This partitioning strategy is illustrated in Figure 38. The secure communications needs of each group are different, and different permissions can be configured for each of them.

Note: Obviously, the LNM shown here does not include all the classes representing the various servers for DNS, Web, Mail and other services. These are normally displayed only when affected by a permission, in the view for the service concerned. Refer to the *User Guide* for details.

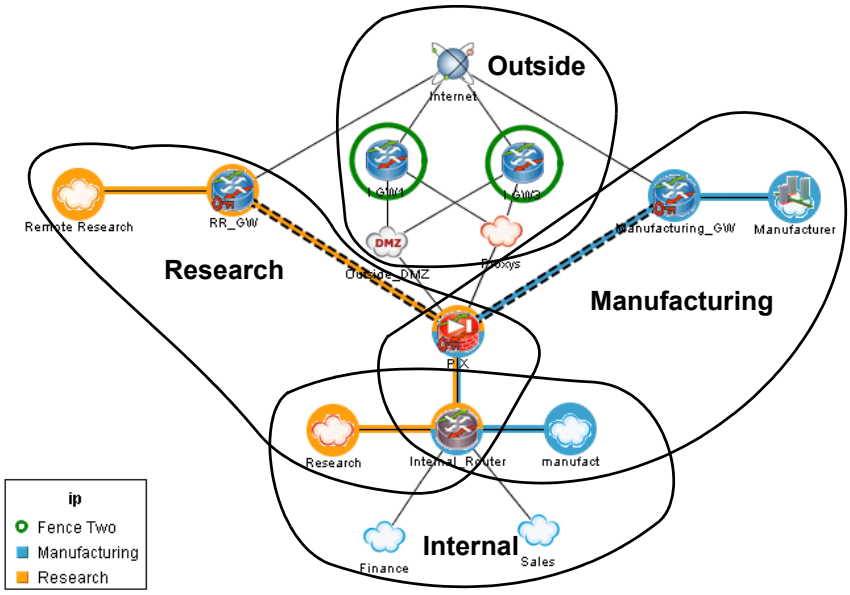


Figure 39: The Network Is Partitioned In Four Sections

It can be readily seen that sections which are protected by secure communications overlap in this example. On the other hand, there is no overlap between these secure sections and the outside section, which remains totally isolated both physically and logically (through appropriate permissions applied via Solsoft Policy Server).

Open Permissions

User needs can be complex. A network may need secured access for a service in one direction, and unsecured access for the same service may suffice in another. Such is the case in this example with `http`.

The research department has its own internal web server, based at the main site. The remote research group needs secure access to this intranet. At the same time, all the major groups in research and marketing need and want access to the company's public web site. As this site is public, secured access is not required.

Figure 40 shows the permissions for `http`.

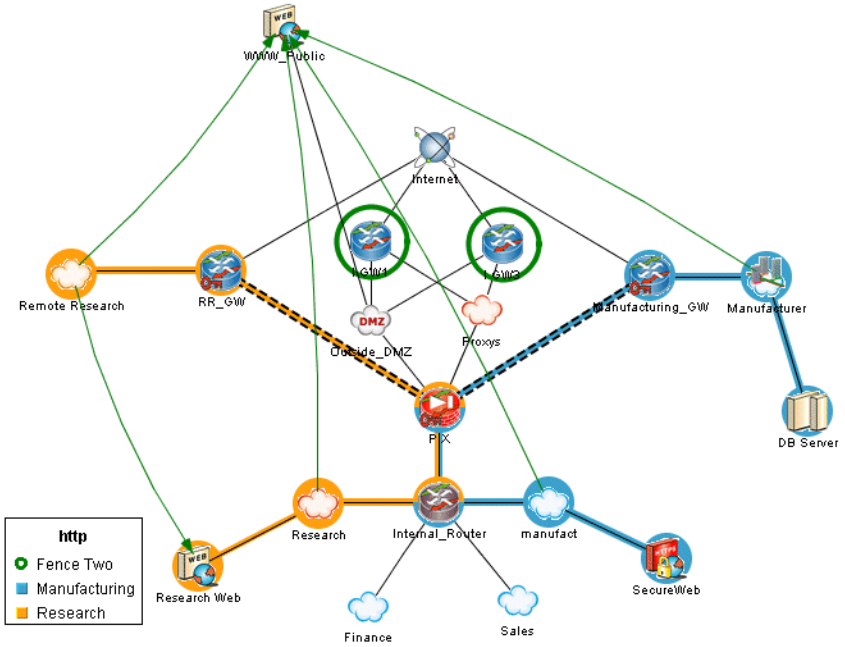


Figure 40: Secured And Unsecured `http`

Secure access to `Research Web` is provided via the VPN. Other accesses to `WWW_Public` pass directly via the Internet and not through either VPN.

Note that even networks within a trust zone can communicate freely with the public web server. This server is not in either trust zone, and so communication passes from inside to outside and back as if the trust zone did not exist.

On the other hand, `Research`, `Remote Research` and `Research Web` are all contained in the same trust zone, which, connected by a tunnel, form a VPN. Thus, all `http` traffic between them will be encrypted and pass through the tunnel.

Logical Path vs. Physical Path

In order for a packet to arrive at `Research Web` site from `Remote Research`, it has to follow a physical path that takes it through the Internet via PEP `RR_GW`, through one of the `I GW` gateways, probably to a proxy server, and into the internal network via `PIX`.

Many of the devices and networks on this physical path are not included in the trust zone and are not trustworthy. This is the advantage of a VPN. Solsoft Policy Server will automatically calculate the filters necessary for PEPs such as `I GW1` and `I GW3` for `AH` or `ESP` to provide the logical path which is the VPN (trust zone plus tunnel).

More Permissions

The inside manufacturing group also has a web which the outside supplier needs to access. But this web has a secure server, and the `https` protocol is available. In this case, it is not necessary to use the tunnel for secure communications.

Since all the objects involved are in the same VPN, we must explicitly set the permission to pass outside the tunnel (see Figure 41).

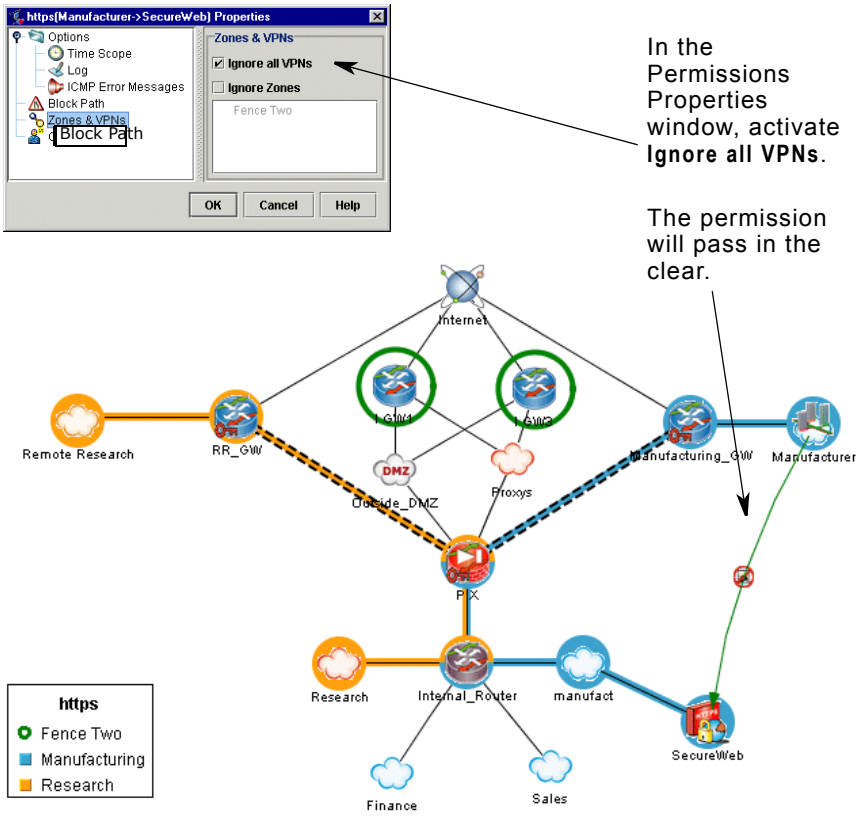


Figure 41: `https` Will Pass Outside The VPN

The manufacturing group at the main site is largely administrative. The fabrication is done by the outside supplier. Thus, the `manufact` group needs `sqlnet` access to the database located at `Manufacturer`. This is sensitive information and must pass via the tunnel. Nothing simpler than to draw its permission (Figure 42)!

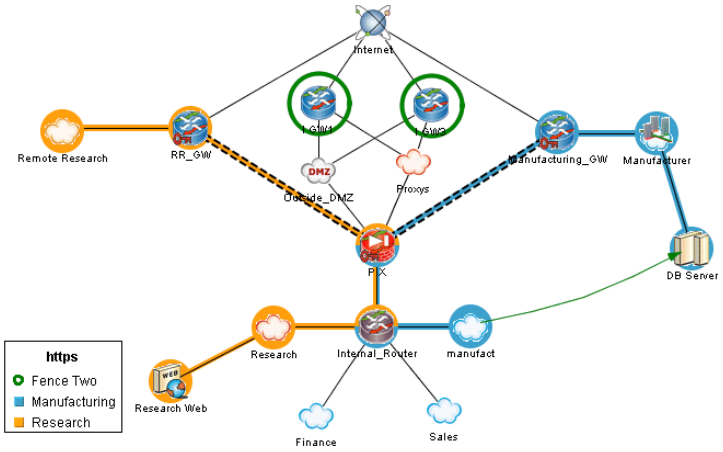


Figure 42: `sqlnet` Permission

Continue in this manner for other needed permissions. Procedures for opening permissions in a VPN are found starting on page 75.

Audit the Tunnels

As always, the audit function is indispensable to verify the flows which will pass through the tunnel. The figures that follow show points of interest extracted from different audits of this network.

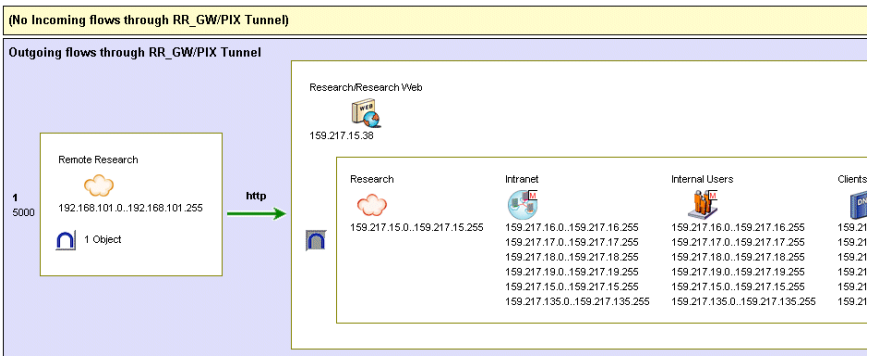


Figure 43: Through Audit Of The Research Tunnel (extract)

Figure 43 shows part of a through audit of the research tunnel. The secure `http` flow is clearly shown.

In Figure 44 and Figure 45, extracts of through audits of two PEPs in the network are shown:

- `PIX` is a tunnel endpoint.
- `I GW1` is not in the trust zone, but on the physical path used by the VPN.

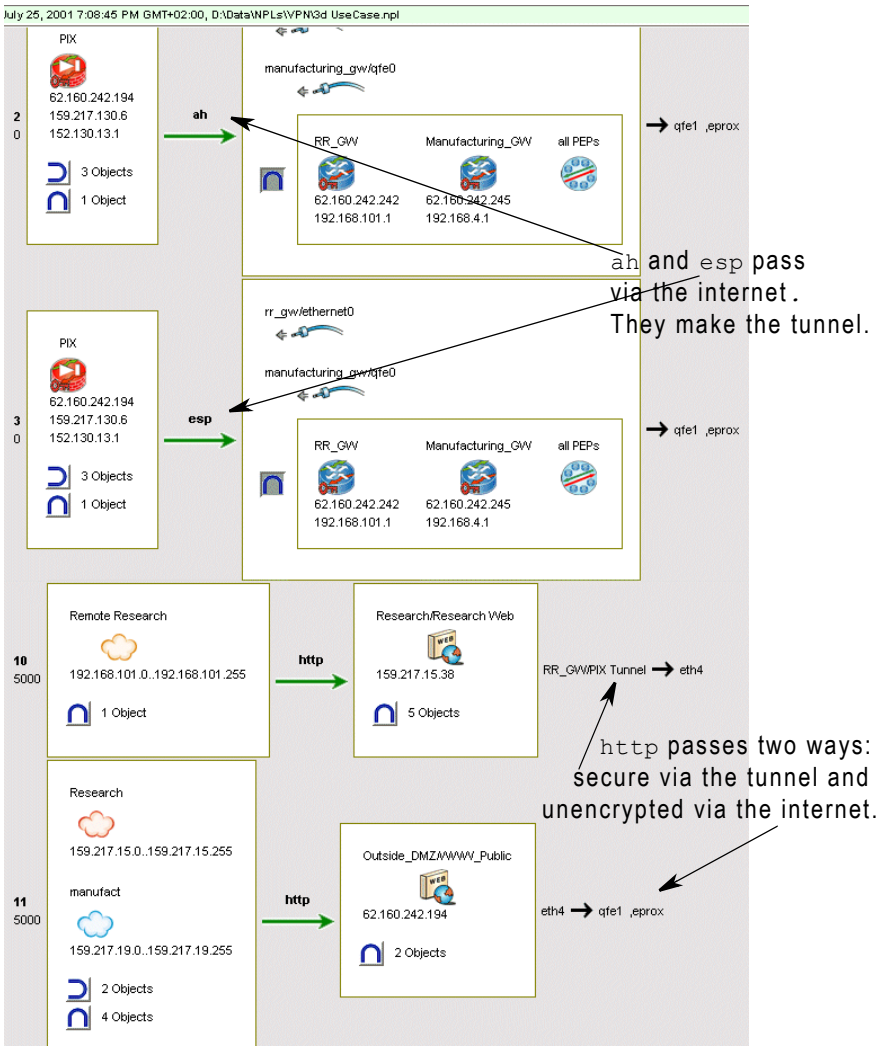
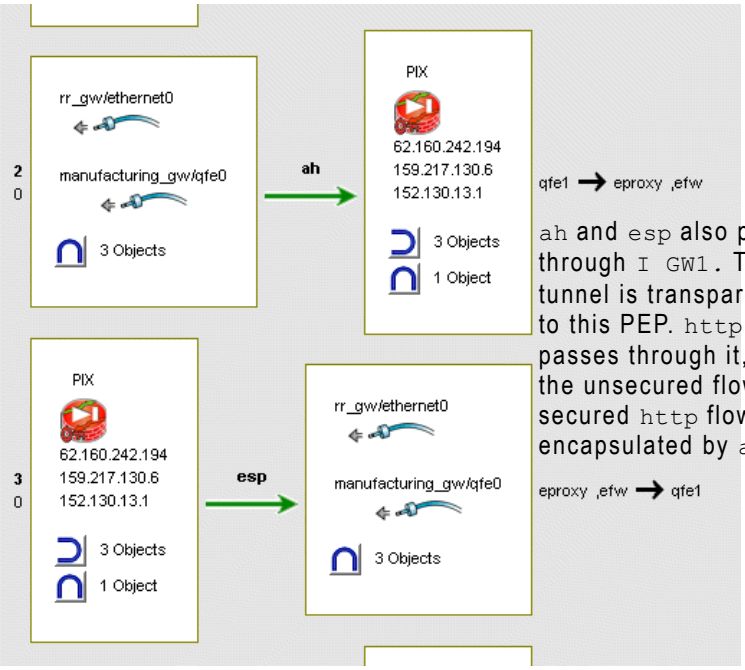


Figure 44: Through Audit Of `PIX` (extract)



`ah` and `esp` also pass through `I GW1`. The tunnel is transparent to this PEP. `http` also passes through it, but only the unsecured flows. The secured `http` flow is encapsulated by `ah` or `esp`.

Figure 45: Through Audit Of `I GW1` (extract)

Of course, the last step is to compile and deploy (upload) the filters!

A Client-to-Gateway Tunnel

This is a particular type of IP Sec Tunnel. You need to create it as you create other IP Sec tunnels.

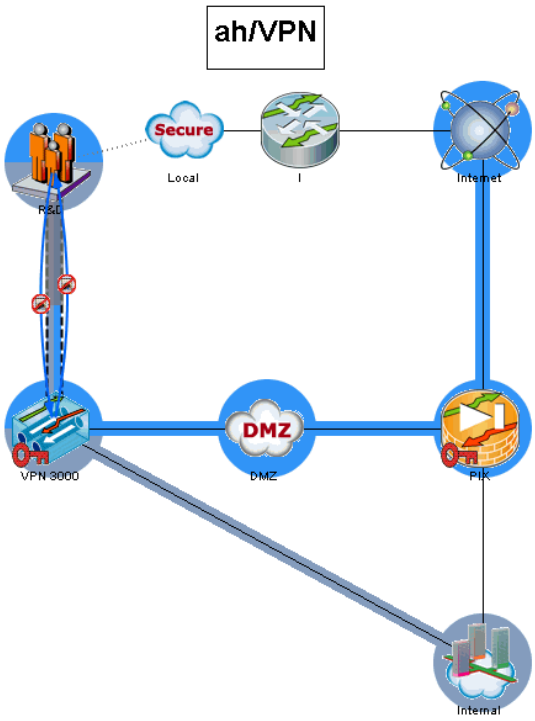


Figure 46: Client-Gateway Tunnel

Procedure

Step 1: Define the server (not needed in the case of an Internal Database).

Step 2: Define the User Group.

Step 3: Define the Mapped User Group.
You can associate it to a metaclass that contains several networks when the User Group can initiate from different networks defined on the map.

Step 4: Associate the servers to the PEP.

Step 5: Associate the Mapped User Group, the VPN Server and the network accessed through the VPN in the same Trust Zone.

Step 6: Create the tunnel between the Mapped User Group and the PEP.

Step 7: Configure the tunnel properties. The main ones are:

- group password
- the pool to use
- split tunnel policy

Other parameters are the same as for the various \User Group parameters.

Abbreviations

Table 6:

Abbreviation	Meaning
ACL	Access Control List
AH	Authentication Header
API	Application Programming Interface
CBAC	Context-Based Access Control
CIDR	Classless Internet Domain Routing
DES	Data Encryption Standard
3DES	Triple Data Encryption Standard
DMZ	Demilitarized Zone
DNS	Domain Name System
ESP	Encapsulating Security Payload
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
GIF	Graphics Interchange Format
HTML	Hypertext Mark-up Language
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IKE	Internet Key Exchange
IP	Internet Protocol
IPsec	Internet Protocol Security
ISAKMP	Internet Security Association and Key Management Protocol
LNM	Logical Network Map
NAT	Network Address Translation
NNM	HPOV Network Node Manager
NPC	Solsoft NPL Compiler
NPL	Solsoft Policy Server Language

Table 6:

Abbreviation	Meaning
PAT	Port Address Translation
PEP	Policy Enforcement Point
PFS	Perfect Forward Secrecy
POP3	Post Office Protocol 3
PPP	Point-to-Point Protocol
PSN	Packet Switched Network
RFC	Request For Comments
RPC	Remote Procedure Call
SA	Security Association
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol over Internet Protocol
Telnet	Telecommunications Network Protocol
TFTP	Trivial File Transfer Protocol
URL	Uniform Resource Locator
UDP	User Datagram Protocol
VPN	Virtual Private Network

Glossary

Table 7:

Term	Definition
Access Control List (ACL)	A sequential list of permit and deny conditions that define the connections permitted to pass through a device.
address inclusion highlighting	The effect of moving the mouse pointer over an object, highlighting any addresses that this object includes. All completely included objects are highlighted with a solid gray box and partially included objects are highlighted with gray stripes.
anti-spoofing	A method used to protect a network against IP spoofing attacks. A packet's source and destination IP addresses are verified to ensure that they are appropriate to the interface through which the packet passes. For example, that a packet entering the local network from the outside carries an external source IP address.
Application Programming Interface (API)	A well-defined set of functions, syntax or languages that enable application programs to communicate with one another and exchange data.
audit	See policy audit.
authentication	A process that ensures that data is actually sent by the presumed sender. *See also* authenticity.
authenticity	The combined functions of authentication *(which see also)* and integrity *(which see also)*.
class	An isolated IP address or grouping of IP addresses associated to one Network object and sharing common management requirements.
client	A computer system or process that requests a service of another computer system or process (a server) using some kind of protocol and accepts the server's responses. A client is part of a client-server software architecture.
Context-based Access Control (CBAC)	Protocol on Cisco's IOS Firewall that provides internal users with secure access control for each application and for all traffic across network perimeters. CBAC enhances security by scrutinizing both source and destination addresses and by tracking each application's connection status.

Table 7:

Term	Definition
data authentication	see authentication
data integrity	see integrity
Demilitarized Zone (DMZ)	A computer or a network located outside the trusted or secure network but still protected from the unsecured network (internet). Network administrators often isolate public resources such as HTTP servers in a DMZ so that an intruder who succeeds in breaching security cannot continue on to the internal network.
denial of service attack	An attack with the purpose of overwhelming the target with spurious data to the point where it is no longer able to respond to legitimate service requests, in contrast to an attack whose purpose is to penetrate the target system.
deny permission	The specified service is denied on the service flow. When a service is denied it is displayed in red with a bar at the mid-point.
Domain Name System (DNS) resolution	Provides the corresponding IP addresses for a particular hostname.
dynamic translation	NAT dynamic translation is where there is a set of (internal) addresses translated using a pool of reusable (external) addresses. See also Pool and PAT.
encryption	A method of modifying message content so that the encrypted message can only be read (“decrypted”) with the aid of an additional element of information (a “key”) which is known only to the sender and the intended recipient of the encrypted message.
extranet	In contrast to the Internet, which provides universal access to network-based information, and an intranet, which is accessible only within an enterprise, an extranet enables a company and its partners or customers to collaborate, communicate and exchange documents in a secured network environment. Extranets typically utilize virtual private networks that allow authorized users to access specific information, such as technical documentation or inventory information (see Virtual Private Network).
File Transfer Protocol (FTP)	A widely-used TCP-based protocol for copying files between hosts.
filtering device	A network device which provides IP filtering capability.

Table 7:

Term	Definition
firewall	A combination of hardware and software resources positioned between the local (trusted) network and the Internet. The firewall ensures that all communication between an organization's network and the Internet conform to the organization's security policy. Firewalls track and control communications, deciding whether to pass, reject, encrypt or log communications.
flow	Basic component of a service, consisting of a protocol, its source and destination ports, and its direction.
frame	The packet transmitted by the data link layer.
frame relay	An advanced packet switched network that transmits variable length data using Permanent Virtual Circuits over digital networks. Less error correction is built into frame relay than X25 which results in faster transmission rates.
gateway	A device positioned between two networks through which all communications between the networks must pass.
global network	The global network is that part of the user's physical network which is controlled by security policies using Solsoft Policy Server.
hash algorithm	An algorithm used in authentication. Converts a string of a given length into a condensed string (i.e. of shorter, and generally fixed, length). The result provides a unique "finger print" for the message; it is easy to calculate, but very difficult and time consuming to reverse.
hostname	A logical name that is assigned to an IP address.
Hypertext Transfer Protocol (HTTP)	A standard protocol for exploring the web. HTTP is implemented as a request/response protocol. A client requests the transfer of a page from the web server. The web server responds with the contents of that page.
icon	Objects in the Solsoft Policy Server toolbar.
inclusion highlighting	The effect of pointing at an object in the logical network map and highlighting associated objects. See Address Inclusion Highlighting.
integrity	A process that verifies that data have not been altered during transmission. *See also* authenticity.

Table 7:

Term	Definition
Internet Control Message Protocol (ICMP)	This internet layer protocol provides an error reporting mechanism and control messages to the TCP/IP control suite.
Internet Key Exchange (IKE)	A protocol to provide key management and origin authentication in conjunction with IPsec.
Internet Protocol (IP)	The network layer for the TCP/IP protocol suite. IP is connection-less and provides the logical addressing used for hosts in a TCP/IP network. IP is also used in determining whether a packet must be routed to a remote network or sent on the local network.
Internet Protocol Security (IPsec)	An encryption and authentication scheme supporting multiple encryption and authentication algorithms.
intranet	An internal private network, managed according to internet protocols, but accessible only inside the organization.
IP masquerading	A networking function similar to NAT which allows computers connected to the Linux gateway to invisibly access the internet using a single IP address. To other machines on the Internet this outgoing traffic will appear to be from the Linux gateway itself.
IPsec PEP	A device capable of establishing and managing IPsec and IKE communications. An IPsec PEP can be a dedicated device that handles only this function, or it can be a PEP which also has IPsec and IKE capability.
limited path zone	Functionality allowing the restriction of the number of possible flow paths between objects. A given flow can never re-enter an LPZ that it has left. It is a special case of a trust zone, *(see also Trust Zone)*.
logical network map (LNM)	The graphical representation in the Solsoft Security Designer of the global network.
machine	Device with an IP address (server, PC, or workstation).
metaclass	A grouping of IP addresses (machines, classes, networks, PEPs) independent from a specific network.
network	One or more networks which share common management requirements (e.g. subsidiary, department, DMZ, etc.).

Table 7:

Term	Definition
Network Address Translation (NAT)	A mechanism for replacing an address or a set of addresses by another address or set of addresses in a network packet.
Network Policy Engine (NPE)	The compiler that generates filter files containing the rules and configuration parameters as specified by the user in the Solsoft Security Designer.
nexus	A PEP that exists in the logical network map but is not managed by Solsoft Policy Server.
Solsoft Policy Server Language (NPL)	High Level universal language generated in text format by the NPE™ once the compilation process is completed.
object	A network, PEP, class or service flow in the Solsoft Security Designer workspace.
packet	A unit of data as sent across a network.
Packet Switched Network (PSN)	A network that delivers data by breaking it into smaller packets. These packets are routed from the source to the destination host on an individual basis. Each packet may take may take a different route through the network but is reassembled at the destination host.
perfect forward secrecy (PFS)	A system providing dynamic change of derived secret session keys. This ensures that even if a key is cracked, previous and subsequent keys are not compromised as subsequent keys are not derived from previous keys.
permission	An object in the Solsoft Security Designer (Arrow), authorizing a defined IP service, consisting of one or more protocols.
Point-to-Point Protocol (PPP)	A method for transmitting packets over serial point-to-point links, such as a dial-up line.
policy audit	A functionality that allows viewing of an object's global permissions over all configured IP services. Auditing is achieved by clicking on a given object and selecting **Policy Audit**.
Policy Enforcement Point (PEP)	A filtering device where a security policy is enforced.
pool	A dynamic NAT method where addresses are translated using a pool of reusable (external) addresses.

Table 7:

Term	Definition
Port Address Translation (PAT)	A dynamic NAT method similar to Pool except that the router changes the source port of the translated packet, and no further filtering is required on the source port in a TCP/UDP packet. This allows more internal addresses than external addresses to be in use at a given time.
Post Office Protocol 3 (POP3)	A protocol used by email clients to retrieve email messages from a server using TCP as the transport protocol.
privacy	Protection of data so that it cannot be observed, read or interpreted by anyone not authorized to do so.
protocol	A method of packaging information as it is transferred between two networked hosts.
Remote Procedure Call (RPC)	Allows a program on one computer to execute a program running on a server computer
refresh from SNMP	A process of configuration discovery where Solsoft Policy Server polls the current PEP configuration to update PEP information.
rollback filters	The process of rolling back filters involves uploading a previous version or the original version of the filters, effectively overwriting the most recently applied version.
router	See filtering device and/or policy enforcement point.
security association (SA)	A set of IPsec and IKE parameters that describes how two or more entities will configure security services to provide a required level of security.
security policy	A security policy is the collection of rules (authorizations and prohibitions) for different services, applied to a specific segment of the user's global network. The segment will usually consist of one or more networks, PEPs or classes in the logical network map.
server	A computer which provides some service for other computers connected to it via a network.
service	A defined IP service composed of one or more flows.
service group	A "super service" made by grouping together several services under one service name.
Simple Mail Transfer Protocol (SMTP)	A protocol providing mail transfer between two hosts using TCP as the transport protocol.

Table 7:

Term	Definition
Simple Network Management Protocol (SNMP)	An application-level protocol that allows management of network devices.
static translation	NAT static translation is where there is one corresponding external address for each internal address.
subnet	A physically independent network segment, which shares a network address with other portions of the network. Subnets enable greater security from unauthorized internal access by dividing the intranet into discrete managed portions.
subnet mask	Determines the dividing point between the network and host portions of an IP address.
Telecommunications Network Protocol (Telnet)	A remote terminal protocol enabling any terminal to log in to another host.
template	Functionality allowing the replication of one defined access policy over several like network configurations.
template instance list	A list of Instance Definitions to be applied, in turn, to each instantiation of a specific Template Object.
Transmission Control Protocol (TCP)	A transport-level protocol that provides guaranteed, connection-oriented delivery of data.
Transmission Control Protocol over Internet Protocol (TCP/IP)	The common name for the suite of protocols that allows connectivity between heterogeneous environments.
trust zone	A set of Solsoft Policy Server objects where all communications between two objects inside the zone use a path entirely inside it. See also limited path zone.
tunnel	A virtual link defined by a set of IPsec parameters and IKE parameters applied to a pair of devices capable of implementing IPsec.
tunnel policy	A list in Solsoft Policy Server, containing sets of IKE and IPsec proposals which guarantee a given level of security desired by the user. When a tunnel policy is applied to a tunnel, only these proposals are authorized for use in the tunnel.

Table 7:

Term	Definition
Uniform Resource Locator (URL)	The combination of the protocol, fully qualified domain name and content that you want to view in a web browser. For example: http://www.solsoft.com/index.htm specifies that the transfer protocol is http the domain name is www.solsoft.com and the content to be viewed is index.htm.
User Datagram Protocol (UDP)	A transport-layer protocol providing connection-less, non-guaranteed delivery of data on a network. Applications using UDP as their transport protocol must provide their own acknowledgement mechanism to provide reliable delivery.
upload policy	Functionality allowing the installation of the generated ACLs or filters on the filtering equipment.
voice over IP	The use of the internet as a transport medium for digitized voice data.
Virtual Private Network (VPN)	A network with some public segments in which data passing over its public segments is encrypted to achieve secure communications. A VPN is significantly less expensive and more flexible than a dedicated private network.
workspace	In the main window of the Solsoft Security Designer where your logical network map is displayed.
zero subnetting	An option available on some PEPs that support the transmission of both the network address and the subnet mask when routing tables are created. The use of zero subnetting allows for a subnet mask of all zeros to be implemented.

Index

How to Contact Us

USA

Solsoft Inc.
2065 Landings Drive
Mountain View, CA 94043-0827
Tel: +1 (650) 428-2800
Fax: +1 (650) 428-2804

Information:
info@solsoft.com
Web site: www.solsoft.com

Solsoft Southern Europe

Solsoft SA
130, rue Victor Hugo
92300 Levallois-Perret
FRANCE
Tel: +33 (0)1 47 15 55 00
Fax: +33 (0)1 47 15 55 09

Solsoft Central Europe

Prinzenallee 740549
Düsseldorf
GERMANY
Tel: +49 (0)211 52 391 550
Fax: +49 (0)211 52 391 507

Solsoft UK Ltd

1 Berkeley Street
London - W1J 8DJ
UNITED KINGDOM
Tel:+44 (0)20 7016 9090
Fax: +44 (0)20 7016 9100