

# Virtual Private Networks

---

## Contents

Overview .....	10-4
VPN Tunnels .....	10-4
IP Security (IPSec) .....	10-4
IPSec Headers .....	10-5
Hash and Encryption Algorithms .....	10-6
IPSec VPN Tunnels .....	10-7
Security Associations (SAs) .....	10-7
IKE .....	10-8
VPN Overlay .....	10-13
Physical Setup .....	10-14
Configuring a VPN Using IPSec .....	10-15
Configuring IPSec with IKE .....	10-15
Configuring IPSec with Manual Keying .....	10-19
How the ProCurve Secure Router Processes IKE Policies and Crypto Maps .....	10-20
Configuration Tasks .....	10-23
Enabling Crypto Commands .....	10-23
Configuring IKE Policies .....	10-23
Peer ID .....	10-24
Initiate and Response Mode .....	10-26
Attribute Policy .....	10-28
Enabling NAT-Traversal (NAT-T) for a Client-to-Site VPN .....	10-31

Configuring a Peer's Remote ID and Preshared Key .....	10-32
Site-to-Site Configuration .....	10-33
Client-to-Site Configuration .....	10-34
Configuring a Remote ID List for a VPN that Uses Digital Certificates .....	10-34
Mapping the Remote ID to an IKE Policy and Crypto Map Entry .....	10-35
Defining Traffic Allowed over the VPN Tunnel .....	10-35
Restricting Specified Hosts .....	10-36
Permitting Local and Remote Networks .....	10-37
Applying the ACL to a Crypto Map .....	10-38
Example Configuration .....	10-39
Enabling Router Traffic to Servers at a Remote VPN Site .....	10-39
Configuring IPSec SA Parameters .....	10-40
Transform Sets .....	10-40
Crypto Maps .....	10-42
Applying a Crypto Map to an Interface .....	10-46
Granting Remote Users a Private Network Address with IKE Mode Config (Required for Client-to-Site VPNs) .....	10-47
IKE Mode Config .....	10-47
Configuring an IKE Client Configuration Pool .....	10-48
Applying the Pool to an IKE Policy .....	10-49
Using Extended Authentication (Xauth) (Optional) .....	10-49
Configuring an Xauth Server .....	10-50
Configuring an Xauth Host .....	10-53
Using Digital Certificates (Optional) .....	10-54
Overview .....	10-54
Obtaining Digital Certificates .....	10-57
Managing Certificates .....	10-61
Configuring a VPN using IPSec with Manual Keying .....	10-64
Configuring the Transform Set .....	10-65
Configuring Crypto Maps for Manual IPSec .....	10-67
Example Configuration .....	10-69
Monitoring a VPN .....	10-70

Troubleshooting a VPN That Uses IPSec .....	10-73
Tools and Procedures .....	10-73
Troubleshooting Commands .....	10-74
Checking WAN Connections .....	10-75
Determining the Source of the Problem: Permitting All Traffic in a VPN .....	10-75
Monitoring the IKE Process using Debug Commands .....	10-76
Comparing VPN Policies .....	10-80
Returning VPN Policies to Their Defaults .....	10-86
Quick Start .....	10-88
Configuring a Site-to-Site VPN .....	10-90
Configuring a Client-to-Site VPN .....	10-94
Obtaining Digital Certificates .....	10-101

## Overview

When your organization leases dedicated lines to establish a WAN, it is guaranteed a secure, private connection. Your organization controls what networks can access the private lines. However, leasing private lines can be costly. When you establish a WAN through the Internet, you capitalize on pre-existing public connections to link networks with a minimum of expense. However, because the Internet is public, it is also insecure. You can access your organization's networks through the Internet, but so can anyone else.

A virtual private network (VPN) addresses this problem.

## VPN Tunnels

A VPN provides secure, private connections across an insecure, public network, such as the Internet. A VPN achieves privacy by connecting sites together through secure tunnels. A tunnel is a point-to-point connection between two authorized endpoints. That is, the tunnel functions essentially as a virtual private line. It carries data only between specified, trusted peers.

In order for a tunnel to provide the same security as a private WAN connection, it must be able to authenticate the endpoints of the tunnel and ensure that only data from these endpoints accesses the tunnel. It may also encrypt confidential data to shield it from hackers. Several protocols have emerged to meet these criteria.

## IP Security (IPSec)

IPSec, which supports a variety of industry-standard authentication and encryption protocols, is a flexible, highly secure method of establishing a VPN. IPSec can operate in tunnel mode. That is, one or both of the endpoints of the tunnel can be gateway devices providing IPSec for hosts on a connecting network. The ProCurve Secure Router will be the gateway for your VPN, which you can configure to connect to another gateway in a site-to-site VPN or to multiple mobile users in a client-to-site VPN.

## IPSec Headers

Operating on the Network Level of the Open Systems Interconnection (OSI) model, IPSec authenticates the endpoints of a tunnel by encapsulating an IP packet with an IPSec header. The IPSec header is either an Authentication Header (AH) and/or an Encapsulation Security Payload (ESP) header.

The placement of the header in the packet differs according to the mode in which IPSec is operating. IPSec can operate in either transport or tunnel mode. In transport mode, the IPSec header encapsulates the payload at the Transport (TCP or UDP) Layer (Layer 4). An IP header then encapsulates the IPSec packet. (See Figure 10-1.)

Transport mode is typically used for local security applications. It provides flexibility and security, but it can be difficult and expensive to implement because the host must add the IPSec header before it adds an IP header and transmits the data. That is, every host must support IPSec.

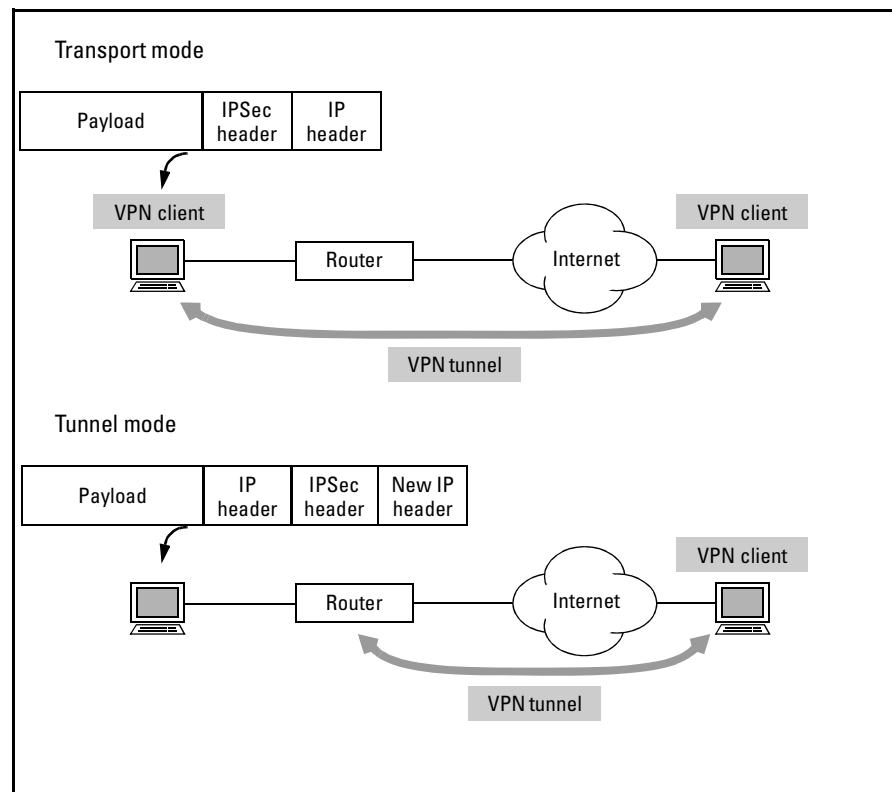


Figure 10-1. IPSec Transport and Tunnel Mode

IPSec tunnel mode, which acts at the Network Layer (Layer 3), allows a gateway device (such as a router) to provide IPSec support for many hosts. The router receives a packet already encapsulated with an IP header. It then encapsulates the IP packet with an IPSec header, adding a new IP header to direct the packet to the location where it will be processed. (See Figure 10-1.) The ProCurve Secure Router supports IPSec tunnel mode.

An AH header authenticates both the payload and the new IP header. An ESP header only authenticates the payload, but can also encrypt it. The tools AH and ESP use to secure data sent over the VPN tunnel are hash and encryption algorithms.

## Hash and Encryption Algorithms

To understand how algorithms secure data, you must understand the difference between a key and an algorithm. A key is a unique string of text; it is what the router actually combines with data in order to transform the data. An algorithm is a set method for transforming data; it specifies a series of permutations and functions performed on data using the unique key.

Both AH and ESP use hash algorithms to authenticate data. A hash algorithm uses a unique authentication key to condense data into a distinctive message digest. The host then appends the message digest to the data. When the remote host receives the complete packet, it uses the same authentication key and algorithm to de-condense the message digest and compare it to the data. If the two match, then the host knows both that:

- the data was sent by the host claimed as the source (because only this host also knows the unique authentication key)
- the data has not been tampered with en route

An encryption algorithm uses a unique key to transform data into a form readable only by a host using the same key.

AH and ESP support the following hash algorithms:

- Message Digest 5 (MD5)
- Secure Hash Algorithm (SHA)

ESP also supports the following encryption algorithms:

- Data Encryption Standard (DES)
- Triple DES (3DES)
- Advanced Encryption Standard (AES), with 128, 192, or 256 bit keys

## IPSec VPN Tunnels

A private WAN connection physically defines the path between two hosts over which data can be transmitted. Only authorized hosts can exchange data because only authorized hosts have access to the physical media that transmit the data. A VPN tunnel virtually simulates such a private connection. That is, it simulates the *privacy* of such a connection, not the physical connection itself, which is provided by the public network. In other words, what a private WAN connection controls physically—the data that can pass between two hosts—the VPN tunnel must control virtually.

The algorithms described above provide this control for IPSec VPN tunnels. Each tunnel is defined by a unique authentication and/or encryption key. Only authorized peers can exchange data because peers automatically drop all data except that whose integrity is confirmed by a message digest, which was generated using the shared authentication key. A unique encryption key may also transform data, effectively hiding it from potential hackers.

### Security Associations (SAs)

Every VPN tunnel is an individual, private connection between two peers defined by the unique set of authentication and encryption keys that secure it. IPSec maintains the definition for an individual tunnel in an IPSec Security Association (IPSec SA). The SA contains the tunnel's authentication and/or encryption keys as well as policies for how such keys are generated and managed.

When a host sends an IPSec packet to a peer, it first searches for an IPSec SA associated with a tunnel to that peer. If such an SA already exists, the host inserts the security parameter index (SPI) associated with the SA into the IPSec packet. When the remote host receives the packet, it matches the SPI to the corresponding SA stored in its system. (If the remote host cannot match the SPI, it discards the packet.) The remote host, which now knows which key was used to hash the data, can de-hash and authenticate it. The remote host can also look up the key it must use to decrypt data, if necessary.

Of course, when a host first initiates a VPN connection with a peer, it will be unable to find an associated SA; the SA has not yet been negotiated.

Your task, when you configure a VPN connection, is to define how the router will negotiate an SA to a specified peer. An SA can be created either manually or using Internet Key Exchange (IKE).

**Defining an SA Manually.** You can define the IPSec SA yourself, specifying the algorithms to be used to secure data, defining the SA's SPI, and inputting the actual keys. (See "Configuring a VPN using IPSec with Manual Keying" on page 10-64.) However, because this method of configuration is relatively insecure and complex, ProCurve Networking does not recommend it.

**Defining an SA Using IKE.** By far, the more secure and manageable solution for VPN configuration is to allow IKE to negotiate the IPSec SA. IKE regulates the process as hosts authenticate each other, agree upon hash and encryption algorithms, and generate the unique keys used to secure packets.

Using IPSec with IKE provides increased security because keys are randomly generated and periodically changed.

IKE also eases configuration. Your role is simply to configure IKE to exchange messages with certain, authorized peers and to define the security parameters that IKE proposes when negotiating the IPSec SA.

## IKE

IKE follows a set process to negotiate the IPSec SA and passes through two phases. The first phase establishes a preliminary tunnel, or IKE SA. The second phase establishes the IPSec SA. When you understand this process, you will find it much easier to configure your ProCurve Secure Router to make a VPN connection.

**IKE Phase 1.** During phase 1, IKE must fulfill three tasks:

- negotiate security parameters for the IKE SA
- generate the keys used to secure data sent using the IKE SA
- authenticate the endpoints of the tunnel (the two hosts)

Typically, therefore, IKE phase 1 involves three exchanges between hosts, or six total messages. (See Figure 10-2.)

**Security parameters.** In the first exchange, the host initiating the VPN connection sends a message to the remote host, proposing one or more security policies. Each policy specifies a hash algorithm, an encryption algorithm, and an authentication method. The remote host searches its IKE policies for one that matches one of the proposed policies. When it finds a match, it returns these security parameters to the original host.

If the remote host cannot find a match, the VPN connection fails. This is why it is very important that you match the IKE policies at both ends of the connection.



**Key generation.** You will recall that an algorithm is simply the set method for transforming data using a key. The key is what actually defines and secures the tunnel and it must be unique. When you use IKE, however, you only need to configure the algorithms IKE proposes in the first exchange. IKE generates the actual keys for you using the Diffie-Hellman Key Agreement Protocol. The Diffie-Hellman exchange takes place in the second set of exchanges of IKE phase 1.

The Diffie-Hellman protocol is a secure method for generating a unique, shared key without sending it over the connection and thus rendering it vulnerable to interception. Each host selects a private value, which is then modified (using prime number modulation) into a public value. Hosts exchange the public values. Each uses the other's public value and their own private value to compute a new value. The computation function is such that these values will be the same.

This shared value is the authentication or encryption key used to secure data in the final IKE phase 1 exchange and all IKE phase 2 exchanges. In this way, IPSec provides an additional layer of security; hosts transmit their authentication information in secured packets, and secured packets negotiate the IPSec SA itself.

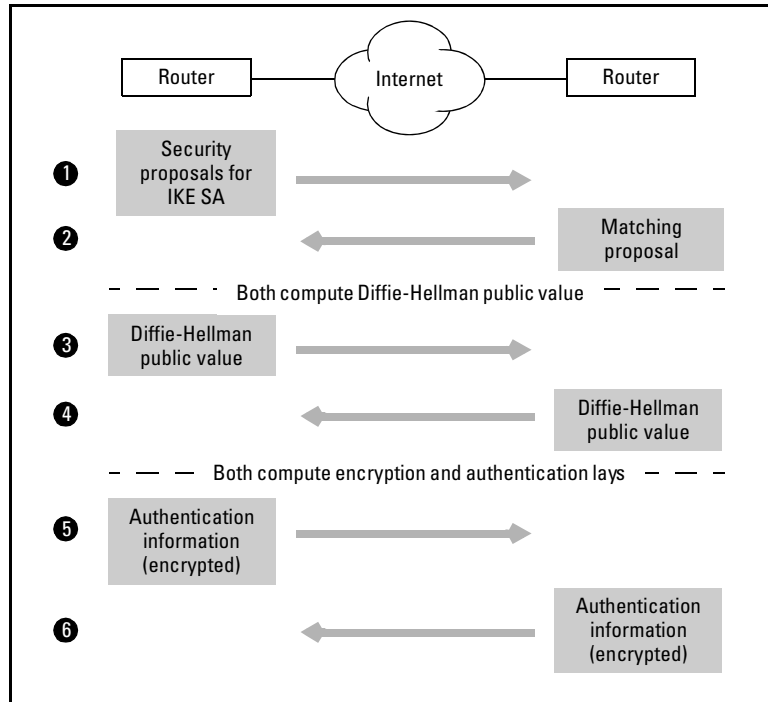


Figure 10-2. IKE Phase 1

**Authentication.** In the third IKE phase 1 exchange, hosts confirm each other's identities according to the method agreed upon in the first exchange. The method can be:

- preshared keys
- digital certificates

Preshared keys are *symmetric*. Hosts using preshared keys have determined the same secret value beforehand. They now exchange this value to authenticate each other, and the IKE SA is established.

Digital certificates use *asymmetric keys*. That is, each host receives two keys from a certificate authority (CA)—one to encrypt data and one to decrypt data. The host's *private key* encrypts data, which can then only be decrypted with that host's *public key*.

When authenticating itself, a host sends a certificate containing its identification information, its public key, and its CA's digital signature. The host then appends its own digital signature to the certificate, which it generates by hashing the certificate and encrypting it with its private key. The remote host who receives the certificate first extracts the public key and uses it to decrypt the digital signature. It then decondenses the signature and compares it to the certificate. A signature that matches the certificate testifies to the certificate's integrity.

The remote host next checks the CA's digital signature by decrypting it with the public key in the CA certificate (which the host must have loaded in its system). The CA's signature attests that the first host is who it claims to be. A certificate revocation list (CRL) issued by the CA tracks which hosts are trusted to join the VPN.

***IKE modes.*** IKE phase 1 can be initiated in one of two modes:

- main mode
- aggressive mode

*Main mode* consists of the exchange of six messages (three exchanges) as described above.

*Aggressive mode* condenses the process into three messages. First, the initiating host sends all necessary information: its IKE SA policy proposals, Diffie-Hellman public value, and either its preshared key or digital certificate. The remote host responds with the IKE SA policy it has selected, its Diffie-Hellman public value, its preshared key or certificate, and authentication for the session. The first host replies, authenticating the remote host and establishing the IKE SA. Aggressive mode is quicker than main. However, because it requires hosts to send identifying information before exchanges are encrypted, it is less secure.

In the Secure Router OS, you configure the IKE mode in an IKE policy. You configure the security proposals IKE uses in an IKE attribute policy.

Table 10-1 summarizes the configurations you must make for IKE phase 1.

**Table 10-1. IKE Phase 1 Exchanges**

IKE Phase 1 Exchange	Message Includes	You Must Configure	Reference
security proposal	<ul style="list-style-type: none"><li>• hash algorithm</li><li>• encryption algorithm</li><li>• authentication method</li><li>• Diffie-Hellman group</li><li>• IKE SA lifetime</li></ul>	IKE attribute policy	page 10-28
Diffie-Hellman key exchange	public value	—	—
authentication	preshared key or digital certificate	preshared key or certificate	page 10-32 or page 10-54

**IKE Phase 2.** The goal of IKE phase 2 is to negotiate the IPSec SA. (For this reason, even though IKE carries out both phases, phase 1 is associated with IKE policies and phase 2 with IPSec policies.) Like an IKE SA, an IPSec SA defines unique authentication and encryption keys, as well as other security parameters for the VPN connection. Keys generated during IKE phase 2 will secure all data exchanged over the lifetime of the VPN tunnel.

When negotiating the IPSec SA, IKE follows much the same process it did in IKE phase 1. The initiating host sends IP packets (now secured by the IKE SA), proposing one or more security policies. Each policy includes a hash algorithm and (if using ESP) an encryption algorithm.

The responding host searches its IPSec policies (referred to as crypto map entries when configuring the ProCurve Secure Router) for a match. When it finds a match, it returns the policy to the initiating host.

IKE then manages the generation and exchange of any hash and encryption keys. It also associates an SPI with the IPSec SA.

Peers can now transmit data securely over the VPN tunnel.

In the Secure Router OS, you will configure proposals for IKE phase 2 in a transform set and crypto map entry. Table 10-2 summarizes configurations you must make for IKE phase 2.

**Table 10-2. IKE Phase 2 Exchanges**

IKE Phase 2 Exchange	Message Includes	You Must Configure	Reference
security proposal	<ul style="list-style-type: none"> <li>• one to three algorithms:               <ul style="list-style-type: none"> <li>– AH hash</li> <li>– ESP encryption</li> <li>– ESP hash</li> </ul> </li> <li>• perfect forward secrecy (Diffie-Hellman) group (optional)</li> <li>• IPSec SA lifetime</li> </ul>	<ul style="list-style-type: none"> <li>• transform set containing the algorithm(s)</li> <li>• crypto map entry containing:               <ul style="list-style-type: none"> <li>– transform set</li> <li>– perfect forward secrecy group (optional)</li> <li>– IPSec SA lifetime</li> </ul> </li> </ul>	page 10-40
Diffie-Hellman key generation	public value	—	—

## VPN Overlay

You can also establish a VPN using Generic Routing Encapsulation (GRE) tunneling. A GRE tunnel establishes a virtual point-to-point connection between two routers across a common, public network such as the Internet.

You configure a tunnel that uses an address in the private network. The tunnel endpoints, however, use public IP addresses. The router encapsulates traffic that arrives on the tunnel with a GRE header and a new IP header, with a destination address of the remote tunnel endpoint.

The new IP header allows traffic to cross the public network to the remote tunnel endpoint. The GRE header renders the payload transparent to intervening routers in the public network. Only at the remote tunnel endpoint can a router decapsulate packets and send them on to their private network destination. In this way, traffic crosses from point to point in the private network through the public network, as if the public network did not exist.

GRE tunnels therefore offer some of the same advantages as a VPN established using IPSec:

- a virtual private point-to-point connection between remote routers
- a private connection carried cost-effectively through a public network

Disadvantages of GRE include:

- because packets are not encrypted, the tunnel is less secure
- each tunnel must be manually configured

GRE tunnels are commonly used to send multicasts through a network (such as the Internet) that cannot route multicast messages. For example, routing protocols such as RIP v2 and OSPF send multicast updates. A tunnel can encapsulate the updates and carry them through the network that does not support multicasts.

See *Chapter 11: Configuring a Tunnel with Generic Routing Encapsulation* to learn how to configure a VPN overlay with such tunnels.

---

## Physical Setup

You must purchase a 7100/7200 IPSec VPN module in order for your ProCurve Secure Router to support a VPN. Before installing the module, disconnect the ProCurve Secure Router from the power source. Slide the module into the encryption slot in the ProCurve Secure Router's rear panel until the module sits firmly against the chassis. Secure the screws and restore power to the router. You should refer to the "Installation Instructions" in the *ProCurve Secure Router 7100/7200 IPSec Module Quick Start Guide* for detailed instructions on how to install the module.

The IPSec VPN module enables the software features that support IPSec protocols and relieves the CPU of the overhead associated with processing the encryption algorithms.

After installing the IPSec module, you will be able to enter **crypto** commands in the CLI or use the VPN wizard in the Web browser interface.

## Configuring a VPN Using IPsec

In order to establish a VPN connection, you must define how the IPsec SA is to be negotiated and with what peers. The IPsec SA can be created either manually or using IKE. This guide will focus on IKE configuration, which is recommended. (To learn how to configure an IPsec SA manually, see “Configuring a VPN using IPsec with Manual Keying” on page 10-64.)

### Configuring IPsec with IKE

Your role is primarily to give IKE the information it needs to carry out IKE phase 1 and phase 2 with an authorized peer. You must also inform the router of what traffic to include in the VPN. VPN settings break down into five general categories:

- policies proposed during IKE phase 1 (IKE SA definitions)
- policies proposed during IKE phase 2 (IPsec SA definitions)
- authorized peer IDs
- VPN traffic (defined in an access control list [ACL])
- authentication information

**Policies for IKE Phase 1 (IKE SA Establishment).** You must configure at least one IKE policy. For each policy, you must define:

- the peer with which the router exchanges IKE messages
- the modes in which the router can initiate and respond to IKE
- the security parameter proposals

You configure the security parameter proposals in an attribute policy. Each policy contains an authentication method, a hash algorithm, and an encryption algorithm. You also select the Diffie-Hellman group, which specifies the length of the prime number used to generate shared keys, and the lifetime for the SA.

When the ProCurve Secure Router cannot find a match for a peer's IKE policy proposals, it terminates the connection. Therefore, you must be careful to configure the same settings on both sides of the connection. You can configure multiple attribute policies for an IKE policy to maximize the chances that peers come to an agreement.

You can refer to Table 10-3 for a summary of policies you can configure for the IKE SA. Each setting must match the peer's setting.

**Table 10-3. Policies for IKE Phase 1: IKE SA Establishment \*Must Match Peer**

Parameter	Options	Default	Configured in	Reference
*hash algorithm	<ul style="list-style-type: none"> <li>MD5</li> <li>SHA</li> </ul>	SHA	IKE attribute policy	page 10-28
*encryption algorithm	<ul style="list-style-type: none"> <li>DES</li> <li>3DES</li> <li>AES (128-bit)</li> <li>AES (192-bit)</li> <li>AES (256-bit)</li> </ul>	DES	IKE attribute policy	page 10-28
*authentication method	<ul style="list-style-type: none"> <li>preshared key</li> <li>DSS digital certificate</li> <li>RSA digital certificate</li> </ul>	preshared key	IKE attribute policy	page 10-28
*IKE SA lifetime	<ul style="list-style-type: none"> <li>60 to 86,400 seconds (1 minute to 1 day)</li> </ul>	8 hours	IKE attribute policy	page 10-28

Table 10-4 displays parameters for the modes in which the router will initiate and respond to IKE.

**Table 10-4. Policies for IKE Phase 1: IKE Mode**

Parameter	Options	Default	Configured in	Reference
initiate mode	<ul style="list-style-type: none"> <li>aggressive</li> <li>main</li> </ul>	main	IKE policy	page 10-26
respond mode	<ul style="list-style-type: none"> <li>aggressive</li> <li>main</li> <li>any mode</li> </ul>	any mode	IKE policy	page 10-26

**Policies for IKE Phase 2 (IPSec SAs Establishment).** You must configure the security parameters IKE proposes for the IPSec SA in a crypto map entry. Again, each policy must include a hash algorithm, and (if using ESP protocol) an encryption algorithm. You specify algorithms in one or more transform sets, which you then bind to the crypto map.

If you do not want IKE to refer to the keys created in IKE phase 1 when it generates the new keys for the IPSec SA, you must specify a perfect-forward secrecy (PFS) group. The PFS group defines the Diffie-Hellman group for the new keys.

You can also specify the lifetime for the VPN tunnel.



Refer to Table 10-5 for a summary of how you configure security policies for the IPSec SA. You do not have to specify the same algorithms and other options for the IKE SA and the IPSec SA. However, you must be sure to configure IPSec proposals that match your peer's.

**Table 10-5. Policies for IKE Phase 2: IPSec SA Establishment \*Must Match Peer**

Parameter	Options	Default	Configured in	Reference
*hash algorithm	<ul style="list-style-type: none"> <li>• MD5</li> <li>• SHA</li> </ul>	no default	transform set (which is then associated with a crypto map entry)	page 10-40
*encryption algorithm	<ul style="list-style-type: none"> <li>• DES</li> <li>• 3DES</li> <li>• AES (192-bit)</li> <li>• AES (128-bit)</li> <li>• AES (256-bit)</li> </ul>	no default	transform set (which is then associated with a crypto map entry)	page 10-40
*PFS group	<ul style="list-style-type: none"> <li>• Diffie-Hellman group 1</li> <li>• Diffie-Hellman group 2</li> </ul>	PFS not used	crypto map entry	page 10-42
*IPSec SA lifetime	<ul style="list-style-type: none"> <li>• 2560 to 536870912 kilobytes</li> <li>• 120 to 86,400 seconds (2 minutes to 24 hours)</li> </ul>	8 hours	crypto map entry	page 10-42

**Authorized Peer ID.** Typically, for a site-to-site VPN, the peer's remote ID is the IP address of the interface on the remote router that connects to the Internet. The remote ID can also be the device's domain name.

For a client-to-site VPN, you may need to allow remote users from many different locations. You should configure the ID for connecting to the mobile users as **any**. You have several options for configuring the ID that IKE will use when authenticating peers—for example, a wildcard email address purely as an identifier.

You configure the ID for the peers authorized to form the remote endpoint of the VPN tunnel in several locations (see Table 10-6):

- IKE policy
- remote ID and preshared key list
- crypto map entry

**Table 10-6. Authorized Peer ID**

Parameter	Options	Default	Configured in	Reference
peer ID (for establishing communications)	<ul style="list-style-type: none"> <li>• public IP address (site-to-site)</li> <li>• any (client-to-site)</li> </ul>	no default	<ul style="list-style-type: none"> <li>• IKE policy</li> <li>• crypto map entry</li> </ul>	<p>page 10-24</p> <p>page 10-42</p>
peer ID (for identification in a remote ID list)	<ul style="list-style-type: none"> <li>• public IP address</li> <li>• domain name</li> <li>• email address</li> <li>• ASN distinguished name (when using digital certificates only)</li> <li>• any</li> </ul>	no default	remote ID and preshared key list	page 10-32

You must configure the peer's ID in an IKE policy. The IKE policy controls the initiation of the IKE process. Setting a peer's address in the IKE policy allows IKE to send the first IKE message to it, establish an IKE SA, and, ultimately, open a VPN connection.

You should also add a remote ID for each peer (together with a preshared key, if used) to a list configured from the global configuration mode context.

If you want your ProCurve Secure Router to initiate IKE with a peer, you should configure the ID for this peer in a crypto map entry. When the router needs to create a VPN tunnel to a peer, it uses the ID set in the crypto map to reference an IKE policy.

You can map different peers to different crypto map entries and/or IKE policies to create various security levels according to your organization's needs.

**VPN Traffic.** You must also specify which LANs will connect through the VPN by matching the crypto map entry to an extended access control list (ACL). In the ACL, you add entries permitting traffic from a local network to a remote network.

For a client-to-site VPN, the remote network is the addresses on the private network that IKE mode config assigns to clients.

Table 10-7 refers you to the sections in which you will learn how to configure the selectors for VPN traffic.

**Table 10-7. Configuring VPN Traffic**

Parameter	Options	Default	Configured in	Reference
Local network(s)	subnet (IP range indicated by wildcard bits)	No default	extended ACL permit statement (source IP)	page 10-35
Remote network(s)	subnet (IP range indicated by wildcard bits)	No default	extended ACL permit statement (destination IP)	page 10-35

**Authentication Information.** You select whether IKE will use preshared keys or digital certificates for authentication in an IKE policy; however, you also must configure the actual authentication information that IKE sends. (See Table 10-8.)

If you select preshared keys, you must associate a peer's preshared key with its ID in the remote ID list configured from the global configuration mode context.

If you select a digital signature standard, you must load a CA and self certificate into the ProCurve Secure Router operating system. The local router will send the self certificate to authenticate itself to peers. You should also add the ID for authorized peers to the remote ID list so that peers can authenticate themselves to the local router. For example, if the certificates used in your network identify hosts by a certain domain name, you should add that domain name to the remote ID list.

**Table 10-8. Authentication Information**

Parameter	Options	Default	Configured in	Reference
preshared key	alphanumeric string (for example: mypassword)	no default	remote ID and preshared key list	page 10-32
digital certificate	<ul style="list-style-type: none"> <li>• DSS self certificate</li> <li>• RSA self certificate</li> </ul>	no default	<ul style="list-style-type: none"> <li>• remote ID list</li> <li>• CA profile</li> </ul>	<ul style="list-style-type: none"> <li>• page 10-32</li> <li>• page 10-57</li> </ul>

### Configuring IPSec with Manual Keying

You are strongly encouraged to use IKE to generate keys. However, if you must use manual keying, you will configure an inbound and an outbound key for each connection to a remote site. The local inbound key should match the remote outbound key and vice versa.

**Table 10-9. Inbound and Outbound Manually Configured Keys**

Parameter	Options	Default	Configured in	Reference
key protocol	<ul style="list-style-type: none"><li>• AH</li><li>• ESP</li></ul>	no default	<b>crypto map, set session-key</b> command	page 10-64
SPI	256 to 4294967295	no default	<b>crypto map, set session-key</b> command	page 10-64
encryption key	hex string	no default	<b>crypto map, set session-key</b> command	page 10-64
authentication key	hex string	no default	<b>crypto map, set session-key</b> command	page 10-64

Table 10-9 displays the parameters that you must configure to establish IPSec keys manually.

You must also configure all other settings discussed for IPSec with IKE, except those for IKE phase 1.

### How the ProCurve Secure Router Processes IKE Policies and Crypto Maps

When a packet arrives on a VPN interface, the ProCurve Secure Router follows a set procedure for deciding to which VPN tunnel it belongs, if any, and securing it according to the security policies established for that tunnel. (See Figure 10-3.)

As mentioned above, you can configure more than one crypto map entry and/or IKE policy. When you create a crypto map entry, you assign it an alphanumeric name and a map index between 0 and 65,535. Entries with the same name (but different index numbers) are grouped together as a single crypto map, which you assign to a WAN interface as a set.

When an outgoing packet is transmitted on the WAN interface, the ProCurve Secure Router reads the source and destination address in the packet's IP header. The router then searches the ACLs associated with the interface's crypto map to determine whether it needs to negotiate a VPN tunnel over which to send the packet. The router processes ACL in the crypto map entry with the lowest number first. If the router does not find a match in this ACL, it begins processing the crypto map entry with the next highest number. If the router never finds a match, it discards the packet. If the router finds that the packet matches a crypto map entry, for which an active IPSec SA that also

matches the packet already exists, then the router secures the packet with the keys contained in the SA, inserts the associated SPI, and forwards the packet to its destination.

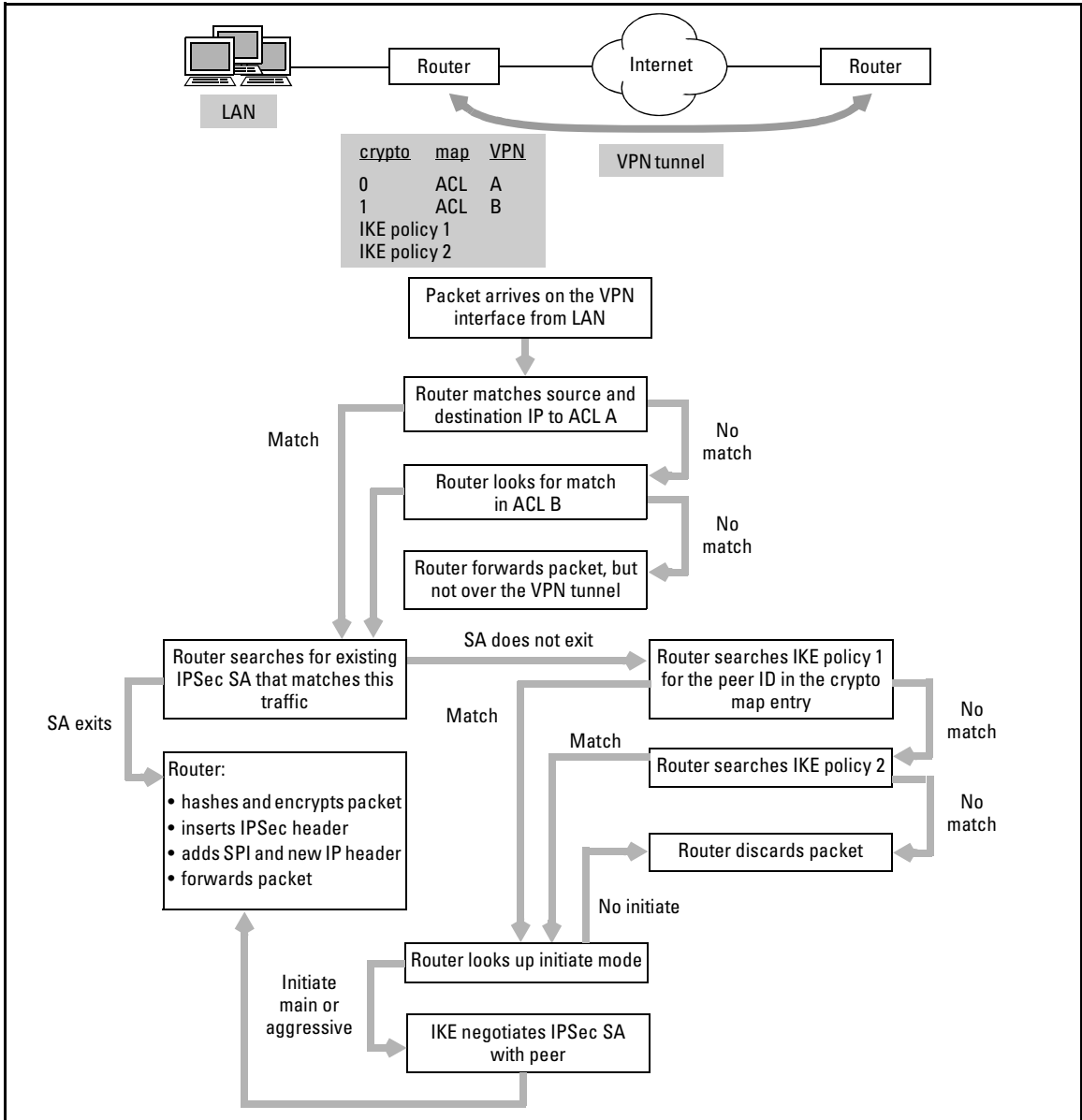


Figure 10-3. How the ProCurve Secure Router Processes Crypto Maps

If the packet does not match an active IPSec SA, then the ProCurve Secure Router looks up the IKE policy associated with the peer specified in the entry. It uses this policy to initiate IKE with the peer, establish an IKE SA, and negotiate an IPSec SA to secure the packet. (If the associated IKE policy does not allow the router to initiate IKE, the packet will be discarded.)

The ProCurve Secure Router also prioritizes IKE policies according to their index numbers, which can be between 1 and 10,000. The router processes the lowest-numbered IKE policy first.

Each IKE policy contains one or more IKE attribute policies, which are also numbered and processed from lowest to highest number. The attribute policy contains the settings IKE uses in its IKE phase 1 exchanges. You can configure different authentication methods, hash and encryption algorithms, and other security parameters in different IKE attribute policies in a single IKE policy. This makes it more likely that IKE will be able to negotiate an IKE SA with the peer.

When an incoming packet arrives on a WAN interface to which you have assigned a crypto map, the ProCurve Secure Router checks its SPI and searches for the matching IPSec SA. If it cannot find a match, it discards the packet. If the packet does not have a SPI, the router can attempt to negotiate an IPSec SA with the peer that sent the packet. It does so using lowest numbered IKE policy configured to initiate IKE with that peer.

When a peer initiates IKE with the router, the router responds using the lowest-number IKE policy that allows it to respond to that peer.

Table 10-10 summarizes how the router matches traffic to a VPN tunnel (IPSec SA) or to a policy for establishing a VPN tunnel.

**Table 10-10. How the Router Matches Traffic to VPN Policies**

<b>The Router Matches</b>	<b>To</b>	<b>According to</b>
an outgoing packet	<ul style="list-style-type: none"><li>• a crypto map entry</li><li>• an IPSec SA (if one exists)</li></ul>	source and destination IP (defined by the map entry's ACL and in the SA)
a crypto map entry	an IKE policy, which negotiates an IPSec SA	<ul style="list-style-type: none"><li>• peer ID</li></ul>
an incoming packet	an IPSec SA	SPI
IKE phase 1 message from peer	an IKE policy, which negotiates an IPSec SA	peer ID

## Configuration Tasks

In order to configure a VPN connect using IKE, you must:

- enable **crypto** commands
- configure an IKE policy
- configure an IKE attribute policy
- add an entry for the peer in a remote ID list
- configure a transform set
- specify VPN traffic in an ACL
- configure a crypto map entry
- apply the crypto map to an interface

If you are using preshared keys, you must also associate a peer with its preshared key in the remote ID list.

If you are using digital certificates, you must load a CA and a self certificate into the Secure Router OS.

If you are configuring a client-to-site VPN, you must also configure an IKE mode config pool. You can optionally enable Xauth.

## Enabling Crypto Commands

After you install the IPSec VPN module, enter the following command from the global configuration mode context:

```
ProCurve(config)# ip crypto
```

This command enables the **crypto** commands, which you use to configure the VPN.

For the greatest security and ease of management, you should configure IKE to manage peer authentication, key exchange, and negotiation of the VPN tunnel.

## Configuring IKE Policies

The IKE policy defines how an IKE SA with a specific peer will be negotiated. The settings you must configure in an IKE policy include:

- the peer's ID
- an attribute policy

You can also alter the default settings for:

- initiate mode
- response mode
- IKE SA security parameters stored in the attribute policy, including:
  - hash algorithm
  - encryption algorithm
  - Diffie-Hellman group
  - authentication method

To begin configuring an IKE policy, enter this command from the global configuration mode context:

**Syntax:** `crypto ike policy <index number>`

The index number determines the priority for the IKE policy and must be a value between 1 and 10,000. When the router needs to negotiate an IPSec SA with a peer or to respond to a peer's IKE negotiations, it searches IKE policies for one that matches the peer. Because the router begins with the policy with the lowest index number, the lower the index number, the higher the priority.

After entering the **crypto ike policy** command, you will enter the IKE policy configuration mode context, indicated by this prompt:

```
ProCurve(config)# crypto ike policy 1
ProCurve(config-ike)#
```

## Peer ID

The peer ID defines the gateway devices or VPN clients with which IKE can establish an IKE SA. You must add the peer ID for each peer in the VPN to at least one IKE policy.

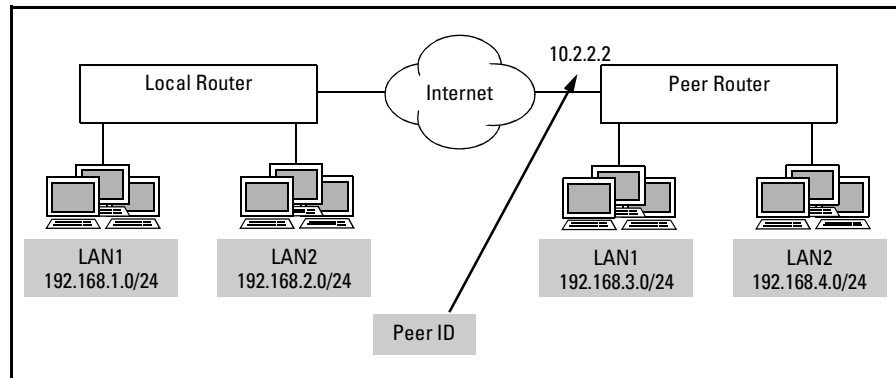
The peer ID type for an IKE policy is always an IP address. This is because IKE is responsible for initial communications with a potential VPN peer; it must know where to reach the peer.

**Site-to-Site Configuration.** The peer ID is the public IP address for the remote gateway device. Usually this is the IP address on the remote router for the interface connecting to the Internet.

To set the ID, enter:

**Syntax:** `peer <A.B.C.D>`





**Figure 10-4. Peer ID**

To configure Local Router shown in Figure 10-4, you should enter:

```
ProCurve(config-ike)# peer 10.2.2.2
```

Even in a VPN with several sites, your ProCurve Secure Router creates an individual VPN tunnel to each site. (Remember that VPN tunnels are point-to-point connections.) However, you can use the same IKE policy to negotiate the preliminary IKE SA for different VPN tunnels. Evaluate the security parameters required for each VPN connection and use the same IKE policy for connections at the same security level. To configure multiple peers, simply enter the command multiple times.

---

**Note**

When you configure more than one peer ID for a policy, it can no longer initiate IKE. If the local router must be able to initiate IKE, you should configure a separate IKE policy for each peer.

If you want IKE to negotiate different security parameters for connections to various sites (for example, a key using a less processor-intensive algorithm), you must configure a separate IKE policy for each site.

---

**Note**

If the remote gateway has a dynamic address, you must set the peer ID to **any**. The policy will not be able to initiate IKE.

**Client-to-Site Configuration.** A client-to-site VPN connects mobile users (such as telecommuters) to a private network through the individual users' Internet connection. It would not be feasible for you to configure a peer ID for each mobile user, even if they all had a static IP addresses. You should allow IKE to establish a VPN tunnel with any peer:

```
ProCurve(config-ike)# peer any
```

This setting does not open a security breach because the peer must still authenticate itself with a preshared key or digital certificate.

**Default IKE Policy.** An IKE policy whose peer is set to **any** also acts as the default policy. You can store only one such policy in the running-config. You should always assign this policy the highest index number (lowest priority) so that the router will process other policies, matching specific peers, first.

### Initiate and Response Mode

By default, an IKE policy allows the router to initiate IKE in main mode and respond to IKE in any mode. Depending on your VPN topology and security needs, you might need to alter these settings. The local router must be able to respond to the mode in which the remote peer initiates. If the local router initiates IKE, it must do so in a mode to which the remote peer can respond. And, of course, at least one peer must be able to initiate and the other, to respond.

View Table 10-11 for guidelines on how you should configure IKE modes to connect to various types of peers.

**Table 10-11. IKE Modes**

Parameter	Options	Default	Static peer	Dynamic peer	Mobile peer
initiate mode	<ul style="list-style-type: none"><li>aggressive</li><li>main</li></ul>	main	match peer's respond mode	no initiate	no initiate
respond mode	<ul style="list-style-type: none"><li>aggressive</li><li>main</li><li>any mode</li></ul>	any mode	match peer's initiate mode	match peer's initiate mode	match peer's initiate mode

**Site-to-Site Configuration.** Typically, you can leave the initiate and respond modes at their defaults.

However, if the remote router takes a dynamic address, the local router cannot initiate IKE. To prevent the router from initiating IKE, enter:

```
ProCurve(config-ike)# no initiate
```

Conversely, if the WAN interface on your ProCurve Secure Router has a dynamic address, it *must* initiate IKE. It will not, however, be able to respond to IKE because the remote router will not know where to send the first IKE message. You should configure the local router to initiate IKE only:

```
ProCurve(config-ike)# no respond
```

In addition to configuring the router to initiate and respond to IKE, you can configure the mode in which it does so. You will recall that IKE main mode is more secure, though it consumes more bandwidth. (See “IKE Phase 1” on page 10-8 in the chapter overview for information.)

To set the mode to which the router will respond to IKE, enter:

**Syntax:** respond [main | aggressive | anymode]

By default, the router is set to the **anymode** option. You can tighten security by only allowing the router to respond to IKE only in main mode. This option is particularly attractive when your VPN uses preshared keys as the authentication method. When they use aggressive mode, peers send their preshared keys before exchanges are encrypted. When your router uses aggressive mode, you risk making a connection with a peer whose identity has been compromised. To prevent the router from responding to IKE aggressive mode, enter:

```
ProCurve(config-ike)# respond main
```

Conversely, you can set the mode in which the router itself will initiate IKE:

**Syntax:** initiate [main | aggressive]

By default, the router initiates IKE in the more secure, main mode. You can allow the router to initiate IKE in aggressive mode. Be aware that although this option speeds IKE negotiations, it can expose your authentication information. Enter:

```
ProCurve(config-ike)# initiate aggressive
```

**Client-to-Site Configuration.** The router cannot initiate IKE with mobile users in a client-to-site configuration. Enter the following command:

```
ProCurve(config-ike)# no initiate
```

Setting the respond mode to **main** can cause problems in a client-to-site VPN: main mode requires the peer to use an IP address for its ID, but you may need to use a different type of ID for mobile users. Generally, you should keep the default setting, **anymode**, which allows the router to respond to IKE in either mode.

## Attribute Policy

The attribute policy contains the security parameters IKE proposes in its first phase 1 message:

- authentication method
- hash algorithm
- encryption algorithm
- IKE SA lifetime
- Diffie-Hellman group

The authentication method determines whether peers will exchange pre-shared keys or digital certificates before establishing the IKE SA. The hash and encryption algorithms determine how data transmitted using the IKE SA will be transformed. The Diffie-Hellman group specifies the length of the prime number IKE will use when generating the keys for this transformation.

You must configure at least one attribute policy. Enter:

**Syntax:** attribute <polycynumber>

The valid range for a polycynumber is 1 to 65,535.

IKE always proposes the security parameters configured in the attribute policy with the lowest number first. Numbering the first attribute policy you configure higher than 1 leaves room for updates in your organization's security policies. For example:

```
ProCurve(config-ike)# attribute 10
```

All attribute policy settings must match those of the peer. You can only configure a single IKE policy for each peer. However, you can make IKE more flexible and raise the chances of establishing a connection by configuring multiple attribute policies for that IKE policy.

The attribute policy is accessible only to the IKE policy in which you configure it. This means that you cannot assume IKE can propose parameters to one peer that you have configured for another peer.

**Table 10-12. Attribute Policy Settings: Match Peer's Settings**

Parameter	Options (Most to Least Secure)	Default	Command Syntax
hash algorithm	<ul style="list-style-type: none"> <li>• SHA</li> <li>• MD5</li> </ul>	SHA	<b>hash [md5   sha]</b>
encryption algorithm	<ul style="list-style-type: none"> <li>• AES (256-bit key)</li> <li>• AES (192-bit)</li> <li>• 3DES</li> <li>• AES (128-bit)</li> <li>• DES</li> </ul>	DES	<b>encryption [aes-256-cbc   aes-192-cbc   3des   aes-128-cbc   des]</b>
authentication method	<ul style="list-style-type: none"> <li>• RSA digital certificate</li> <li>• DSS digital certificate</li> <li>• preshared key</li> </ul>	preshared key	<b>authentication [rsa-sig   dss-sig   pre-share]</b>
IKE SA lifetime	60 to 86,400 seconds (1 minute to 1 day)	8 hours	<b>lifetime &lt;seconds&gt;</b>
Diffie-Hellman key group	<ul style="list-style-type: none"> <li>• group 1</li> <li>• group 2</li> </ul>	group 1	<b>group [1   2]</b>

You can leave the attribute policy settings at their defaults or customize them according to your organization's security policies. Refer to Table 10-12 for the commands for setting these policies. (See "IKE Phase 1" on page 10-8 in the chapter overview for more information on selecting either preshared keys or digital certificates.)

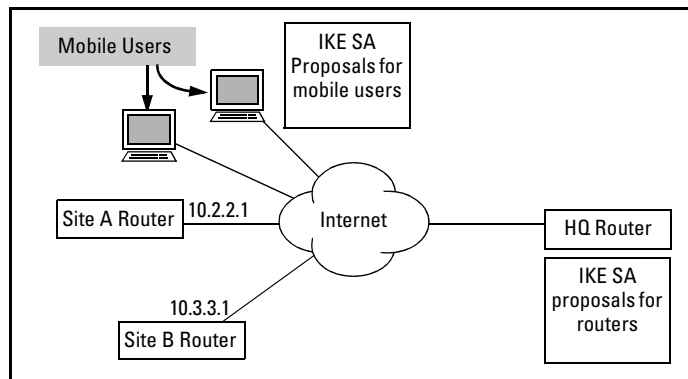
---

**Note**

You must configure at least one attribute policy for each IKE policy even if you do not alter its default settings.

**Example Configuration.** Figure 10-5 illustrates a VPN between headquarters and two branch offices. The VPN must also allow mobile users remote access. The company has established the security parameters shown beneath the headquarters router for IKE SAs. However, because mobile users' clients might not support these options, the company decides to allow greater flexibility for client-to-site IKE SAs. You would configure two IKE policies on the headquarters router. The policy for the branch office sites would include the

stronger security parameters. The policy for the mobile clients would include a higher-priority attribute policy for the preferred security parameters, but also an attribute policy with lower security options.



**Figure 10-5. Example IKE Proposals**

You would complete these steps:

1. Create the IKE policy for initiating IKE phase 1 with routers at remote sites.

```
ProCurve(config)# ip crypto
ProCurve(config)# crypto ike policy 1
ProCurve(config-ike)# peer 10.2.2.1
ProCurve(config-ike)# peer 10.3.3.1
ProCurve(config-ike)# respond main
```

2. Configure the high-security IKE SA proposals in an attribute policy. The same proposals must be configured on the remote routers.

```
ProCurve(config-ike)# attribute 10
ProCurve(config-ike-attribute)# authentication dss-sig
ProCurve(config-ike-attribute)# encryption 3des
ProCurve(config-ike-attribute)# hash sha
ProCurve(config-ike-attribute)# lifetime 240
ProCurve(config-ike-attribute)# group 2
```

3. Create the IKE policy for responding to IKE phase 1 from mobile users. The router cannot initiate IKE with mobile users. Leave the respond mode at the default, **anymode**.

```
ProCurve(config-ike-attribute)# crypto ike policy 10
ProCurve(config-ike)# peer any
ProCurve(config-ike)# no initiate
```

4. Configure the high security IKE SA proposals in an attribute policy:

```
ProCurve(config-ike)# attribute 10
ProCurve(config-ike-attribute)# authentication dss-sig
ProCurve(config-ike-attribute)# encryption 3des
ProCurve(config-ike-attribute)# lifetime 240
ProCurve(config-ike-attribute)# group 2
```

5. Configure a second set of IKE SA proposals for mobile users in a lower priority (higher index) attribute policy:

```
ProCurve(config-ike-attribute)# attribute 20
ProCurve(config-ike-attribute)# authentication dss-sig
ProCurve(config-ike-attribute)# encryption des
ProCurve(config-ike-attribute)# hash md5
ProCurve(config-ike-attribute)# group 1
```

## Enabling NAT-Traversal (NAT-T) for a Client-to-Site VPN

By default, the ProCurve Secure Router allows peers to request that a VPN tunnel use NAT-T.

Remote VPN users may be behind a device that performs network address translation (NAT) on packets destined to the Internet. When a packet passes through a NAT device, the device changes the packet's IP address. If NAT is performed on packets before they are encrypted, as in a site-to-site VPN between two gateway devices, then the packets pass over the VPN connection without difficulty. However, in a client-to-site VPN, client software encrypts packets before the NAT device alters them. As a result of this alteration, packets will fail the IPSec integrity check.

Some client software provides for this problem; however, other software applications (such as those using the L2TP protocol) do not.

NAT-T uses UDP encapsulation to address the incompatibility between NAT and IPSec. UDP encapsulates the IPSec packet in a UDP/IP header. The NAT device changes the address in this header without tampering with the IPSec packet.

Peers agree to use NAT-T during IKE negotiations by exchanging a pre-determined, known value that indicates that they support NAT-T. When the peers exchange the Diffie-Hellman values, they also send NAT Discovery (NAT-D) packets that include hashes of their source and destination IP addresses and ports. Because one peer's source IP address should be the other's destination address, and vice versa, the hashes should match. If they do not, the peers know that somewhere between the two an address was translated.

If the peers discover NAT, then they encapsulate packets in the UDP/IP header. The peer behind the NAT device should also use a one-byte UDP packet that ensures that it keeps the same NAT assignment for the duration of the VPN tunnel.

You specify whether the ProCurve Secure Router will allow a peer to use NAT-T in the IKE policy used to negotiate an IKE SA with that peer. Enter the following command from the IKE policy configuration mode context to set the NAT-T policy:

**Syntax:** nat-traversal [v1 | v2] [allow | disable | force]

By default, the router allows a peer to request either NAT-T version 1 or NAT-T version 2. Enter the following commands to return the router to the defaults:

```
ProCurve(config-ike)# nat-traversal v1 allow
ProCurve(config-ike)# nat-traversal v2 allow
```

NAT-T may affect performance because it adds 200 bytes in the IKE security association negotiations and 20 bytes to each IPSec packet. Also, IPSec must use ESP rather than AH to encapsulate the packet. If you want to prevent peers from using a particular NAT-T version or from using NAT-T at all, use the **disable** keyword. For example:

```
ProCurve(config-ike)# nat-traversal v1 disable
```

If, on the other hand, you want the router to end negotiations unless the peer agrees to use NAT-T, use the **force** keyword with one of the version options.

## Configuring a Peer's Remote ID and Preshared Key

You should add the peer's remote ID to a list configured from the global configuration mode context. IKE uses the settings configured in this list when negotiating an IKE SA with a remote peer. Particularly, IKE uses the information specified in this list to authenticate the peer. When using preshared keys as the authentication method, you must also associate the remote ID with a preshared key. This list is like a username and password database for the VPN. The remote ID is like the username, and the preshared key, the password.

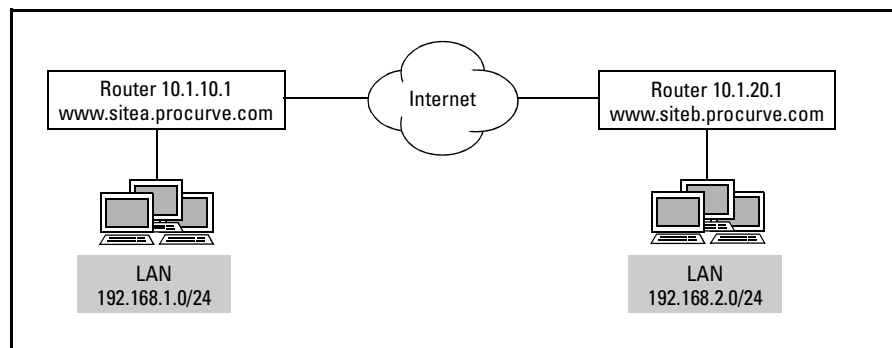
You enter one of the commands shown in Table 10-13 for each peer with which you want to establish a VPN connection. For example, enter this command from the global configuration mode context:

```
ProCurve(config)# crypto ike remote-id fqdn siteb.procurve.com preshared-key
mysecret
```



**Table 10-13. Remote ID Types**

Remote ID Type	Example (Figure 10-6)	Wildcard	Command Syntax
IP address	10.1.20.1	10.1.0.0 0.0.255.255	<b>crypto ike remote-id address</b> <b>&lt;A.B.C.D&gt; &lt;wildcard bits&gt;</b>
domain name	siteb.procurve.com	*procurve.com	<b>crypto ike remote-id fqdn</b> <b>&lt;domain name&gt;</b>
ASN distinguished name (for use with digital certificates only)	"CN=RouterB, C=US, S=CA, L=Roseville, O=ProCurve, OU=TechSupport"	"CN=*, C=*, S=*, L=*, O=ProCurve, OU=**"	<b>crypto ike remote-id asn1-dn</b> <b>&lt;distinguished name&gt;</b>
email address	user@procurve.com	*@procurve.com	<b>crypto ike remote-id user-fqdn</b> <b>&lt;email address&gt;</b>
any	—	—	<b>crypto ike remote-id any</b>



**Figure 10-6. Remote ID for VPN Peer**

### Site-to-Site Configuration

You can identify the peer by its:

- IP address (gateway's public address)
- fully-qualified domain name (FQDN)
- abstract syntax notation distinguished name (ASN-DN) (only when using digital certificates)
- email address

You should identify the peer in the way most supported by your organization's policies. You can also use the wildcard character (\*) to ease configuration. For example, if you are connecting multiple sites that all use your organization's domain name, you might want to enter an FQDN that consists of a wildcard character and your organization's domain name so that you only have to enter one command. This option is, of course, less secure.

For example, you could configure both routers shown in figure 10-6 with this remote ID for the peer:

```
ProCurve(config)# crypto ike remote-id fqdn *procurve.com preshared-key mysecret
```

---

**Note**

---

If the routers are using IKE main mode, you must use an IP address for the remote ID.

### Client-to-Site Configuration

When your organization uses preshared keys, you may specify the peer's remote ID as **any**. For example, enter:

```
ProCurve(config)# crypto ike remote-id any preshared-key mysecret
```

You can also use the wildcard character with your organization's domain name or with a set of email addresses if the ID applies to all remote clients. The remote ID is purely for identifying the client; email addresses do not have to be valid. See Table 10-13 on page 10-33 for the command syntax for specifying the remote ID.

IKE main mode requires an IP address for the remote ID. You can use **any** rather than a domain name or email address if your VPN uses main mode.

### Configuring a Remote ID List for a VPN that Uses Digital Certificates

If your VPN uses digital certificates, you must enter the remote ID specified in the peer's digital certificate. A digital certificate can identify a host in several different ways including:

- IP address
- FQDN
- email address
- ASN-DN

If peers' digital certificates use ASN-DNs, you must enter the fields exactly as they are in the certificate. You can use the wildcard character (\*) for some of the fields. See Table 10-13 on page 10-33 for the command syntax for specifying the remote ID.

## Mapping the Remote ID to an IKE Policy and Crypto Map Entry

You can associate a peer's remote ID with a specific IKE policy and crypto map entry. This option can ease configuration and troubleshooting. You can quickly associate a peer with the policies that the router proposes to it.

To associate IKE and IPSec policies with a peer, enter this command:

**Syntax:** `crypto ike remote-id [address <A.B.C.D> | any | asn-dn <distinguished name> | fqdn <domain name> | user-fqdn <email address>] [preshared-key <key>] [ike-policy <policy number> [crypto map <mapname> <map index>]]`

For example, enter a command such as this:

```
ProCurve(config)# crypto ike remote-id address 10.2.2.1 preshared-key mysecret ike-policy 2 crypto map VPN 20
```

---

### Note

If you associate the remote ID with a crypto map entry that has not yet been configured, the Secure Router OS will automatically create such an entry. See “Crypto Maps” on page 10-42 for instructions on configuring a crypto map.

Take care to associate the remote ID with the IKE policy and/or crypto map that includes that peer's correct public IP address.

If you are only configuring one IKE policy and crypto map entry, you need not use this option.

## Defining Traffic Allowed over the VPN Tunnel

You define which networks connect over an individual VPN tunnel as follows:

1. Create an extended ACL.
2. Add entries to the ACL denying any hosts not authorized to access the VPN.
3. Add entries to the ACL permitting traffic from the local network to the remote network.
4. Apply the ACL to the crypto map entry that defines the tunnel's IPSec SA.

Extended ACLs allow you to select traffic according to its source and destination IP address (among other fields in the IP header). To create an ACL that selects traffic transmitted between two networks, enter the following command:

**Syntax:** ip access-list extended <listname>

An ACL listname is alphanumeric and case-sensitive. For example:

```
ProCurve(config)# ip access-list extended VPNTraffic
```

### Restricting Specified Hosts

You can enforce your organization's security policies by restricting certain hosts from accessing the VPN tunnel. By default, the ACL excludes all hosts not explicitly permitted. However, if certain hosts who should not be able to access the VPN are on a permitted subnet, you will need to explicitly deny them, as follows:

**Syntax:** deny ip [any | host <source A.B.C.D> | hostname <source hostname> | <source A.B.C.D> <wildcard bits>] [any | host <destination A.B.C.D> | hostname <destination hostname> | <destination A.B.C.D> <wildcard bits>]

Use the **host** keyword to deny a single host. Use wildcard bits to specify a range of address. The wildcard bits operate on reverse logic from subnet masks; a bit corresponding to a 1 is ignored. In effect, the digital number corresponding to the wildcard bits in an octet is the number of hosts that can be selected.

In this example, you exclude hosts 99 and 192 through 223 in the 192.168.1.0 /24 network from the VPN:

```
ProCurve(config-ext-nacl)# deny ip host 192.168.1.99 any
ProCurve(config-ext-nacl)# deny ip 192.168.1.192 0.0.0.31 any
```

---

#### Note

In order to exclude a specific host or hosts from a permitted subnet, you must enter the deny entry before the permit entry. This is because the ProCurve Secure Router processes ACL entries in order and stops processing the list as soon as it finds a match.

You can also deny specific hosts as a valid destination for traffic carried over the VPN tunnel. For example:

```
ProCurve(config-ext-nacl)# deny ip any host 192.168.3.99
```

## Permitting Local and Remote Networks

You will need to add a permit statement specifying each local network allowed to access the VPN tunnel as the source IP address. The destination depends on the type of VPN.

---

### Note

The IP addresses selected by the ACL must match the peer's configuration exactly. For example, if the peer's configuration specifies that remote network 192.168.3.0 /24 is part of the VPN, but not remote network 192.168.4.0 /24, you must permit only 192.168.3.0 /24 as a valid destination.

---

**Site-to-Site Configuration.** The destination is the remote network or networks that participate in the VPN. The source is the local VPN network or networks.

You permit traffic to and from a network with this command:

**Syntax:** permit ip [any | host <source A.B.C.D> | hostname <source hostname> | <source A.B.C.D> <wildcard bits>] [any | host <destination A.B.C.D> | hostname <destination hostname> | <destination A.B.C.D> <wildcard bits>]

Wildcard bits allow you to select an entire subnet or range of subnets in one entry. However, if the remote gateway device connects to more than one non-contiguous subnet, you must enter separate permit statements to allow traffic from every local subnet to every remote subnet included in the VPN.

Wildcard bits operate on reverse logic from subnet masks. A one indicates that the router is to ignore the bit and zero indicates that the router is to check it. For example, the wildcard bits in the following entry allow you to select an entire class C network for the source and for the destination of VPN traffic:

```
ProCurve(config-ext-nacl)# permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255
```

You can also select a range of subnets. For example, an organization has divided the network 10.1.0.0 /16 into /24 subnets. Each site includes 16 /24 subnets, which means that, considered as a whole, the site is a /20 network. That is, Site A includes subnets 10.1.0.0 /24 through 10.1.15.0 /24, which can be summarized as 10.1.0.0 /20. Site B include subnets 10.1.16.0 /24 through 10.1.31.0 /24, which can be summarized as 10.1.16.0 /20. (Every time you double the number of subnets, you decrease the prefix length by one.)

A quick rule-of-thumb for specifying a range of /24 subnets such as these is that the number in the third octet plus one shows the number of subnets in the range.

To permit traffic from Site A to Site B, you enter:

```
ProCurve(config-ext-nacl)# permit ip 10.1.0.0 0.0.15.255 10.1.16.0 0.0.15.255
```

You can also use wildcard bits to include only part of a subnet, according to topology of your VPN.

**Client-to-Site Configuration.** The router uses IKE mode config to assign remote users addresses on the private network after they have established an IKE SA with the router. (Refer to “Granting Remote Users a Private Network Address with IKE Mode Config (Required for Client-to-Site VPNs)” on page 10-47 for more information on IKE mode config.) In the permit statement, the local network is the source. The addresses in the IKE client configuration pool used for the connection are the destination:

```
ProCurve(config-ext-nacl)# permit ip 192.168.1.0 0.0.0.255 100.1.1.0 0.0.0.255
```

For more information on configuring ACLs, see *Chapter 5: Applying Access Control to Router Interfaces*.

## Applying the ACL to a Crypto Map

After configuring the ACL, you should associate it with a crypto map entry. Create the crypto map entry and move to the crypto map configuration mode context. (You will learn how to configure a crypto map in “Crypto Maps” on page 10-42.)

From the crypto map configuration mode context, enter:

**Syntax:** match address <ACL listname>

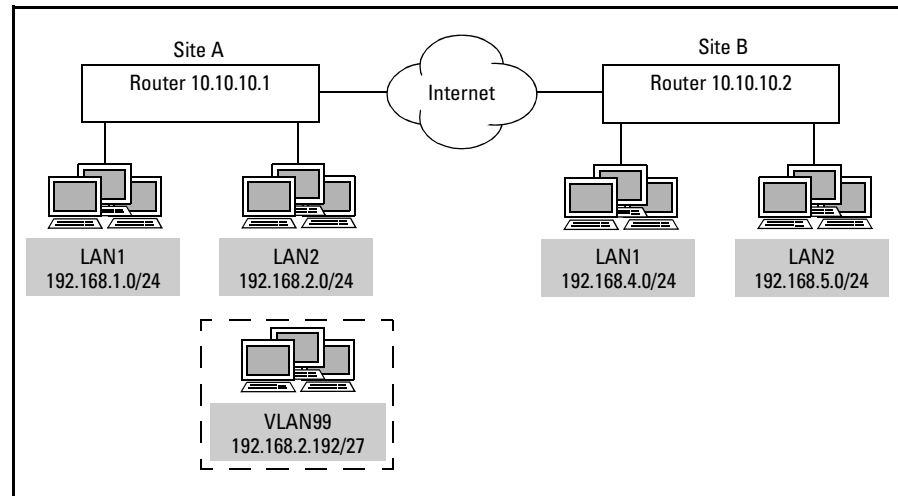
For example:

```
ProCurve(config-crypto-map)# match address VPNTraffic
```

Remember that the ACL defines the traffic permitted over an individual VPN tunnel. That is, it defines, according to source and destination IP address, which packets will be secured by a specific IPSec SA. Even though you can add any number of permit and deny statements to the ACL, you should only add statements for traffic destined to networks behind a single gateway device. If your VPN connects multiple remote sites, you will need to configure an ACL and crypto map entry for each site.

## Example Configuration

Figure 10-7 illustrates a VPN between two remote sites, each of which includes two LANs. At Site B, only one LAN is allowed in the VPN. At Site A, independent on-site contractors have been assigned addresses in VLAN 99—192.168.2.192 to 192.168.2.223. These contractors are not authorized to connect to Site B.



**Figure 10-7. Configuring an ACL for VPN Traffic**

Enter the following commands to define traffic on Router A permitted to access the VPN tunnel to Site B:

```
ProCurve(config)# ip access-list extended VPNTraffic
ProCurve(config-ext-nacl)# deny ip 192.168.2.192 0.0.0.31 any
ProCurve(config-ext-nacl)# permit ip 192.168.1.0 0.0.0.255 192.168.4.0 0.0.0.255
ProCurve(config-ext-nacl)# permit ip 192.168.2.0 0.0.0.255 192.168.4.0 0.0.0.255
ProCurve(config)# crypto map VPN 10 ipsec-ike
ProCurve(config-crypto-map)# match address VPNTraffic
```

## Enabling Router Traffic to Servers at a Remote VPN Site

The ProCurve Secure Router can send traffic to a server on its own behalf. For example, it can download a file from a TFTP server at a remote VPN site. Typically, a router takes the source addresses for the packets that it sends to a server from the IP address of the interface used to reach that server. In our example, packets that the ProCurve Secure Router sends to the TFTP server

pass through the WAN interface and so receive the router's public IP address. However, only traffic from local private networks can access the VPN tunnel, so the traffic cannot reach its destination.

You can force all traffic sent to a server to use the IP address of LAN interface so that it can access the remote VPN site.

Enter one of these commands from the global configuration mode context:

**Syntax:** ip [tftp | sntp | ftp] source-interface <interface ID>

**Syntax:** snmp-server source-interface <interface ID>

For example, you can set Ethernet 0/1 as the source interface for all traffic sent to a TFTP server.

```
ProCurve(config)# ip tftp source-interface eth 0/1
```

## Configuring IPSec SA Parameters

You configure the security parameters that IKE proposes during IKE phase 2 for the IPSec SA in:

- a transform set
- a crypto map entry

### Transform Sets

A transform set contains the hash and encryption algorithms used to secure data transmitted over the VPN tunnel. To create a transform set, complete these steps:

1. Name the transform set.
2. Select AH or ESP. (See "IPSec Headers" on page 10-5 in the chapter overview for more information on the difference between these two protocols.)



3. Specify the algorithms:
  - a. If using AH, you can select:
    - an AH hash algorithm
  - b. If using ESP, you can select:
    - an encryption algorithm
    - a hash algorithm (optional)
  - c. If using AH and ESP, you can select:
    - an AH hash algorithm
    - an ESP encryption algorithm (optional)
    - an ESP hash algorithm (optional)
4. Configure tunnel mode.

**Table 10-14. Transform Sets**

Set Protocol	Algorithm Types	Algorithm Options (Most to Least secure)	Command Syntax
AH	hash algorithm	<ul style="list-style-type: none"> <li>• SHA</li> <li>• MD5</li> </ul>	<b>crypto ipsec transform-set</b> <b>&lt;setname&gt; [ah-sha-hmac   ah-md5-hmac]</b>
ESP	<ul style="list-style-type: none"> <li>• encryption algorithm</li> <li>• hash algorithm (optional, unless encryption is not used)</li> </ul>	<ul style="list-style-type: none"> <li>• encryption:               <ul style="list-style-type: none"> <li>– AES (256-bit key)</li> <li>– AES (192-bit)</li> <li>– 3DES</li> <li>– AES (128-bit)</li> <li>– DES</li> <li>– None</li> </ul> </li> <li>• hash:               <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> </ul>	<b>crypto ipsec transform-set</b> <b>&lt;setname&gt; [esp-aes-256-cbc   esp-aes-192-cbc   esp-3des   esp-aes-128-cbc   esp-des   esp-null] [esp-sha-hmac   esp-md5-hmac]</b>
AH and ESP	<ul style="list-style-type: none"> <li>• AH hash algorithm</li> <li>• ESP encryption algorithm</li> <li>• ESP hash algorithm (optional)</li> </ul>	<ul style="list-style-type: none"> <li>• AH hash:               <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> <li>• ESP encryption:               <ul style="list-style-type: none"> <li>– AES (256-bit key)</li> <li>– AES (192-bit)</li> <li>– 3DES</li> <li>– AES (128-bit)</li> <li>– DES</li> <li>– None</li> </ul> </li> <li>• ESP hash:               <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> </ul>	<b>crypto ipsec transform-set</b> <b>&lt;setname&gt; [ah-sha-hmac   ah-md5-hmac] [esp-aes-256-cbc   esp-aes-192-cbc   esp-3des   esp-aes-128-cbc   esp-des   esp-null] [esp-sha-hmac esp-md5-hmac]</b>

You complete the first four steps in a single command entered from the global configuration mode context. Refer to Table 10-14 for the exact command syntax for configuring a transform set. Enter commands such as the following:

```
ProCurve(config)# crypto ipsec transform-set T1 ah-sha-hmac esp-3des
ProCurve(config)# crypto ipsec transform-set T2 ah-md5-hmac esp-aes-128-cbc esp-
sha-hmac
ProCurve(config)# crypto ipsec transform-set T3 esp-des
ProCurve(config)# crypto ipsec transform-set T4 esp-3des esp-sha-hmac
```

The setname is alphanumeric.

---

**Note**

If you want to use ESP transformation without encryption, use the **esp-null** keyword. If you select this option, however, you must include an ESP hash algorithm. For example:

```
ProCurve(config)# crypto ipsec transform-set T5 esp-null esp-md5-hmac
```

---

After you name the transform set and add the algorithms, you will enter the transform set configuration mode context. Specify tunnel mode, which allows the ProCurve Secure Router to act as a gateway device for hosts on connected LANs:

```
ProCurve(cfg-crypto-trans)# mode tunnel
```

## Crypto Maps

You configure a crypto map entry to specify the security parameters that IKE will propose during phase 2. These settings determine what kind of authentication and encryption keys will define and secure the VPN tunnel. You also specify the peer with which the IPSec SA will be established and the networks involved in the VPN.

For each crypto map, you must specify:

- peer's remote ID (unless the router will only respond to, not initiate, IPSec requests)
- hash and/or encryption algorithms (a transform set)
- the traffic carried over the VPN tunnel (an extended ACL)

You can optionally specify:

- a perfect forward secrecy (PFS) group
- an IPSec SA lifetime

To create a crypto map entry, enter the following command from the global configuration mode context:

**Syntax:** `crypto map <mapname> <map index> [ipsec-ike | ipsec-manual]`

The mapname is an alphanumeric string. You can configure a set of crypto map entries that have the same name but different map indexes, which you apply together to an interface. This is how, for example, you would configure a VPN to multiple sites.

The map index is a number between 0 and 65,535 that indicates to the ProCurve Secure Router in what order to process entries.

The **ipsec-manual** keyword allows you to enter keys manually rather than have IKE generate them automatically. This option is less secure and more complicated to configure. (For more instructions, see “Configuring a VPN using IPSec with Manual Keying” on page 10-64.)

To configure a crypto map entry that uses IKE, you would enter, for example:

```
ProCurve(config)# crypto map VPN 0 ipsec-ike
```

After creating the crypto map entry, you will enter the crypto map configuration mode:

```
ProCurve(config-crypto-map)#
```

**Peer’s Remote ID.** You must set one and only one peer ID.

In a site-to-site VPN, the peer’s remote ID is the ID of the gateway device for the remote networks. Unlike the remote ID configured in the remote ID list, this must be an IP address—the WAN interface that connects to the Internet on the remote router.

The router uses the peer ID in the crypto map to select an IKE policy for communicating with the peer. (In other words, the peer ID in the crypto map entry must match the peer ID in the IKE policy used to establish the IKE SA.) Use the following command to set the peer’s ID:

**Syntax:** `set peer <A.B.C.D>`

For example:

```
ProCurve(config-crypto-map)# set peer 10.2.2.1
```

Unlike an IKE policy, you can only set one peer for the crypto map entry. This is because the crypto map entry actually defines the VPN tunnel, and a VPN tunnel is a point-to-point connection.

---

**Note**

---

If the remote gateway has a dynamic address, you cannot set the peer ID. The router will respond to requests to open a VPN tunnel. If both routers have dynamic addresses, you cannot establish a VPN.

For client-to-site configurations, you do not need to set a peer. The router will use the crypto map entry to respond to requests from mobile users to connect to the private network.

**IKE Policy.** You can also explicitly associate the crypto map entry with an IKE policy. Enter this command from the crypto map configuration mode context:

**Syntax:** `ike-policy <policy number>`

Make sure that the policy you specify includes the same peer that you set for the crypto map entry.

**Hash and Encryption Algorithms.** You must assign at least one transform set to the crypto map entry. (The transform set contains the AH or ESP algorithms that IKE uses to secure the VPN tunnel.) Use the **set** command to specify one or more transform sets by name:

**Syntax:** `set transform-set <setname> [<additional setname>]`

You can assign each crypto map entry up to six transform sets.

Using more than one transform set makes it more likely that IKE will be able to select a security policy compatible with the peer's. You can also assign the same transform set to more than one crypto map entry. IPSec SAs negotiated using these entries will use the same algorithms (although each SA will have its own unique keys).

For example, you have configured three transform sets named T1, T2, and T3 to contain different security algorithms. IKE first proposes the algorithms in set T1. If these do not match the peer's settings, then IKE proposes the algorithms in set T2, and so forth. You would enter:

```
ProCurve(config-crypto-map)# set transform-set T1 T2 T3
```

**Traffic Carried over the VPN Tunnel.** To specify which traffic will be carried over the VPN tunnel (in other words which networks make up the VPN), you must match the crypto map entry to an extended ACL:

**Syntax:** match address <listname>

For example:

```
ProCurve(config-crypto-map)# match address VPNTraffic
```

The extended ACL selects packets according to their source and destination IP address. You configure the ACL to permit traffic between the local and remote networks included in the VPN. The local networks should all connect to the local ProCurve Secure Router and the remote networks should all connect to the remote gateway device.

You cannot attempt to add entries to the ACL and connect to more than one site through the same crypto map entry. If you are configuring a VPN that connects to more than two sites, you should configure a new crypto map entry to establish an IPSec SA with each gateway device. These map entries should have the same map name but different index numbers.

(Configuring an ACL is described in “Defining Traffic Allowed over the VPN Tunnel” on page 10-35.)

**Table 10-15. Crypto Map Entry Settings: Match Peer’s Settings**

Parameter	Options (From Most to Least Secure)	Default	Command Syntax
hash and encryption algorithms (contained in a transform set)	up to six transform sets each set contains up to three algorithms, one each of: <ul style="list-style-type: none"> <li>• AH hash algorithm:               <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> <li>• ESP encryption algorithm:               <ul style="list-style-type: none"> <li>– DES</li> <li>– 3DES</li> <li>– AES (192-bit)</li> <li>– AES (128-bit)</li> <li>– AES (256-bit)</li> </ul> </li> <li>• ESP hash algorithm               <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> </ul>	no default	<b>set transform-set</b> <b>&lt;setname1&gt; [&lt;setname2&gt;</b> <b>[&lt;setname3&gt;] [&lt;setname4&gt;</b> <b>[&lt;setname5&gt;] [&lt;setname6&gt;]</b>

Parameter	Options (From Most to Least Secure)	Default	Command Syntax
PFS group	<ul style="list-style-type: none"> <li>Diffie-Hellman group 2</li> <li>Diffie-Hellman group 1</li> </ul>	PFS not used	<b>set pfs [group2   group1]</b>
IPsec SA lifetime	<ul style="list-style-type: none"> <li>2560 to 536,870,912 kilobytes</li> <li>120 to 86,400 seconds (2 minutes to 24 hours)</li> </ul>	8 hours	<b>set security-association lifetime [kilobytes &lt;kilobytes&gt;   seconds &lt;seconds&gt;]</b>

You can find the commands for configuring IPsec SA security parameters in Table 10-15.

**PFS Group.** By default, IKE refers back to the keys defined by the IKE SA when generating the keys for the IPsec SA during IKE phase 2. PFS enhances security by generating entirely new keys. This ensures that even if one key is comprised, other keys remain secure. If you want to use PFS, you must specify which Diffie-Hellman group IKE uses to generate the IPsec SA keys. For example, you could enter this command from the crypto map configuration mode context:

```
ProCurve(config-crypto-map)# set pfs group2
```

**IPsec SA Lifetime.** You can define the lifetime of an IPsec SA (that is, a VPN connection) in kilobytes and in seconds. See Table 10-15 for the command syntax. Enter commands such as:

```
ProCurve(config-crypto-map)# set security-association lifetime kilobytes 1000000
ProCurve(config-crypto-map)# set security-association lifetime seconds 9600
```

If you set the SA lifetime in both kilobytes and seconds, the VPN connection will close after whichever limit is reached first.

## Applying a Crypto Map to an Interface

In order for the crypto map to take effect, you *must* apply it to an interface. When you apply the crypto map to an interface, you apply the entire set of crypto map entries with the same name. Configuring multiple crypto map entries with the same name but different index numbers lets you establish a VPN connection with multiple peers. It also allows you to create different levels of security for different sets of traffic by matching entries to various ACLs.

You should apply the crypto map to the logical interface on which traffic will be transmitted. Typically this is a WAN interface that connects the Internet. Valid interfaces include:

- PPP interfaces
- Frame Relay subinterfaces
- HDLC interfaces
- ATM subinterfaces
- Ethernet interfaces
- Ethernet subinterfaces (VLAN interfaces)
- demand interfaces

---

**Note**

---

An interface must have an IP address before you can apply a crypto map to it.

From the appropriate interface configuration mode context, enter:

**Syntax:** `crypto map <mapname>`

For example, if your organization has a PPP connection to the Internet, you would enter:

```
ProCurve(config-ppp 1)# crypto map VPN
```

## Granting Remote Users a Private Network Address with IKE Mode Config (Required for Client-to-Site VPNs)

At times you will need to grant remote users a virtual IP address on your organization's private network. This option is typically used for client-to-site VPNs—for example, for telecommuters. These users tunnel into the VPN, often through their home Internet connection, and unlike users at a remote site, they are not part of a single private network which the ProCurve Secure Router can associate with the VPN. IKE mode config assigns these mobile users a virtual private address for as long as they communicate with the VPN gateway.

### IKE Mode Config

The gateway device uses IKE mode config to send configurations such as an IP address to a remote user during IKE negotiations. IKE mode config allows a relatively small pool of mobile users to access the VPN from remote locations. (IKE mode config is not designed for wide-scale management.)

The remote user requests an IP address from the ProCurve Secure Router between IKE phase 1 and phase 2 negotiations. It may also request addresses for Domain Name System (DNS) and NetBIOS Windows Internet Naming Service (WINS) servers. These servers will translate domain names to IP addresses for the user.

The router uses IKE mode config to issue the remote user an address from a client configuration pool. The peer now has an address in a network permitted in the VPN and can establish an IPSec SA with the router. IKE mode config allows remote users to function as members of the private network. These VPN users appear as internal users on the network, which they can then browse freely.

---

**Note**

---

When you configure the ACL specifying which traffic is carried over the VPN connection, you must include the addresses from the client configuration pool as the permitted destination.

### Configuring an IKE Client Configuration Pool

A client configuration pool contains the information the router needs to issue configurations. For each pool, you must configure:

- a range of IP addresses—You should use a separate network from those used by local hosts.

You can optionally configure:

- DNS server address (up to two)
- WINS server address (up to two)

You create the client configuration pool from the global configuration mode context:

**Syntax:** `crypto ike client configuration pool <poolname>`

For example:

```
ProCurve(config)# crypto ike client configuration pool VPNUsers
ProCurve(config-ike-client-pool)#
```

Next, specify the range of IP addresses that the router can assign to remote users:

**Syntax:** `ip-range <first A.B.C.D> <final A.B.C.D>`



For example, include the entire 192.168.100.0 /24 subnet:

```
ProCurve(config-ike-client-pool)# ip-range 192.168.100.1 192.168.100.254
```

Use the commands shown in Table 10-16 to configure optional configurations such as server addresses.

**Table 10-16. IKE Client Configuration Pools**

Parameter	Function	Command Syntax
address range	The router assigns these addresses to clients with which it has established an IKE SA.	<b>ip-range &lt;first A.B.C.D&gt; &lt;final A.B.C.D&gt;</b>
DNS server	The DNS server resolves hostnames into IP addresses for the client.	<b>dns-server &lt;server1 A.B.C.D&gt; [&lt;server2 A.B.C.D&gt;]</b>
WINS server	The WINS server resolves hostnames into dynamic IP addresses for the client.	<b>netbios-name-server &lt;server1 A.B.C.D&gt; [&lt;server2 A.B.C.D&gt;]</b>

### Applying the Pool to an IKE Policy

Move to the configuration mode context for the IKE policy that users will use to establish the IKE SA. Then enter this command:

**Syntax:** client configuration pool <poolname>

For example:

```
ProCurve(config-crypto-ike)# client configuration pool VPNUsers
```

### Using Extended Authentication (Xauth) (Optional)

In a site-to-site VPN, IKE authenticates the remote gateway device. However, if your organization's security policies require it, you can also configure the router to authenticate individual remote VPN users. When you enable an Xauth server on the ProCurve Secure Router, the router requests authentication information from a remote user between establishing the IKE SA and the IPSec SA. You can also use Xauth for increased security in client-to-site VPNs since many clients, including the ProCurve VPN Client, support Xauth.

You can also use Xauth to authenticate the gateway device itself. You can configure the ProCurve Secure Router to be an Xauth host, and authenticate itself to a peer that requires Xauth. However, if you enable the Xauth host on an IKE policy, the router cannot also use the Xauth server with that policy. For this reason, you might want to use the Xauth server with IKE policies that respond to IKE and the Xauth host with policies that only initiate IKE.

## Configuring an Xauth Server

Complete the following steps:

1. Configure an authentication, authorization, and accounting (AAA) list to inform the Xauth server which database to search for usernames and passwords.
2. Enable the Xauth server in an IKE policy.

If you have not already done so, you will also need to configure the local username database or RADIUS server group.

**Configuring a Username Database.** If Xauth will use the router's local database to authenticate users, you should make sure that entries for all authorized users have been added to the database. You can check the list of usernames and passwords by viewing the running-config.

```
ProCurve# show running-config
!
username administrator password procurve
username juan password mypassword
username sara password mysecret
!
```

**Figure 10-8. Viewing Passwords in the Local Username Database**

If necessary, add entries to the local database from the global configuration mode context:

**Syntax:** `username <username> password <password>`

For example:

```
ProCurve(config)# username rodriguez password procurve
```

---

### Note

The router also uses the local database list to permit access to the router. If the users you are entering for Xauth do not have authority to access or configure the router, you should configure a RADIUS or TACACS+ server for Xauth.

---

**Configuring RADIUS and TACACS+.** If Xauth will be using a RADIUS or TACACS+ server database, you must enable the router to contact the server.

First, specify the IP address of the server from the global configuration mode context:

**Syntax:** radius-server host [<A.B.C.D>| <hostname>]

**Syntax:** tacacs-server host [<A.B.C.D>| <hostname>]

You can enter either the server's IP address or its hostname. For example:

```
ProCurve(config)# radius-server host 10.2.3.4
```

You can specify more than one server. Depending on the type of server that you specify, the router automatically creates a default group with the name *radius* that includes all RADIUS servers, or with the name *tacacs* that includes all TACACS+ servers.

You can also place select servers into a group. Use these two commands to group RADIUS servers:

**Syntax:** aaa group server radius <groupname>

**Syntax:** server <A.B.C.D>

Use these two commands to group TACACS+ servers:

**Syntax:** aaa group server tacacs <groupname>

**Syntax:** server <A.B.C.D>

For example:

```
ProCurve(config)# aaa group server tacacs MyServers
ProCurve(config-sg-tacacs)# server 10.2.3.4
ProCurve(config-sg-tacacs)# server10.3.2.1
```

**Configuring an AAA List.** The Xauth server can search for remote VPN users' usernames and passwords in three locations:

- a RADIUS server's database
- a TACACS+ server's database
- the router's local username database

To inform the Xauth server where it should search, create an AAA list and specify the database as an authentication method. You name the list and add entries for the authentication method with the same global configuration mode command. Refer to Table 10-17 for the command syntax for creating an AAA list for Xauth.

**Table 10-17. AAA List Authentication Methods**

Database Location	Keyword	Command Syntax
router	<b>local</b>	<b>aaa authentication login</b> <b>&lt;aaa listname&gt; local</b>
RADIUS server or servers	<b>group</b>	<b>aaa authentication login</b> <b>&lt;aaa listname&gt; group</b> <b>[radius   &lt;groupname&gt;]</b>
TACACS+ server or servers	<b>group</b>	<b>aaa authentication login</b> <b>&lt;aaa listname&gt; group</b> <b>[tacacs   &lt;groupname&gt;]</b>

When you use the **group** keyword to specify RADIUS databases, you can either enter the name of a configured group of servers, or you can use the **radius** keyword, which selects all RADIUS servers. Similarly, you can either enter **group <groupname>** for a specific set of TACACS+ servers or **group tacacs**, which specifies all TACACS+ servers.

If you want the Xauth server to search more than one database, you should specify all these locations in the order in which you want the server to search them. For example, you could configure the Xauth server to search the router's local database first and then, if this database does not include the host's username, the database of any RADIUS server with which the router can communicate. Enter:

```
ProCurve(config)# aaa authentication login xauth local group radius
```

**Enabling the Xauth Server.** You enable the Xauth server in an IKE policy.

First, create the IKE policy (or move to the configuration mode context for a pre-existing policy) and set the peer ID. For a site-to-site VPN, the peer ID is that of gateway device behind which the hosts you want to authenticate are located. For a client-to-site VPN, the peer ID will typically be **any**.

Then, still in the IKE policy configuration mode context, enable the Xauth server and specify the name of the AAA list configured for Xauth:

**Syntax:** client authentication server list *<aaa listname>*

For example:

```
ProCurve(config-ike)# client authentication server list xauth
```

## Configuring an Xauth Host

The ProCurve Secure Router can act as an Xauth host and authenticate itself to a peer that requires Xauth.

Complete the following steps:

1. Create or move to the configuration mode context of the IKE policy for the peer that requires Xauth.

**Syntax:** `crypto ike policy <policy number>`

2. Select the type of authentication.

**Syntax:** `client authentication host xauth-type [generic | otp | radius]`

3. Enable the Xauth host by setting the username and password it will send to the peer.

**Syntax:** `client authentication host username <username> password <password>`

### Setting the Username and Password for Generic Authentication.

Generic authentication is the default type. Obtain your username and password from your VPN peer. From the IKE policy configuration mode context, enter the username and password:

**Syntax:** `client authentication host username <username> password <password>`

For example:

```
ProCurve(config-ike)# client authentication host username VPNPeer password  
MyPassword
```

### Setting the Username and Password for RADIUS Authentication.

1. Change the authentication type to RADIUS:

**Syntax:** `client authentication host xauth-type radius`

2. Enter the username and password you have obtained from your peer.

For example:

```
ProCurve(config-crypto-ike)# client authentication host xauth-type radius  
ProCurve(config-crypto-ike)# client authentication host username VPNPeer  
password MyPassword
```

**Setting the Username, Password, and Passphrase for One-time Password (OTP) Authentication.** OTP provides increased security by using a passphrase to generate a series of passwords, each of which is used only once. This prevents hackers from intercepting and hijacking an authorized VPN user's authentication information.

Complete these steps to configure OTP authentication:

1. Change the authentication type to OTP.

**Syntax:** client authentication host xauth-type otp

2. Specify the username, password, and passphrase:

**Syntax:** client authentication host username <username> password <password> passphrase <passphrase>

For example:

```
ProCurve(config-crypto-ike)# client authentication host xauth-type otp
ProCurve(config-crypto-ike)# client authentication host username VPNPeer
password MyPassword passphrase MyPassphrase
```

## Using Digital Certificates (Optional)

This section explains how to obtain certificates for the ProCurve Secure Router. You should refer to this section only if you selected a digital signature standard for the authentication method of at least one IKE attribute policy.

### Overview

As discussed in the chapter overview, digital certificates rely on asymmetric keys. Each host is issued two keys by its CA: a public key and a private key. The public key decrypts data encrypted by its private key.

A host authenticates itself with a certificate, to which it appends its digital signature. It creates the digital signature by hashing the certificate and then encrypting the hash with its private key. The certificate itself consists of:

- the host's identification information
- the host's public key
- the function used to hash the certificate
- the CA's digital signature

When the peer receives the digital certificate, it extracts the host's public key and hash function. It decrypts and unhashes the signature and compares it to the certificate. If they match, the peer knows that no one has tampered with the certificate en route.

In order to fully authenticate a host, the peer must also have the CA's certificate in its system. This certificate includes the CA's public key, which the peer uses to verify the CA's signature. A genuine CA signature attests that the holder of a certificate is who it says it is. Your CA should also issue you a certificate revocation list (CRL), which lists current and expired certificates of hosts that you trust to access your VPN.

Because a host can freely distribute its public key, it can authenticate itself to anyone who trusts its CA. However, no one can pose as the host because only the host's unshared, private key can encrypt and "sign" the certificate.

In summary, digital certificates present two important security advantages over preshared keys:

- A host can authenticate itself to anyone who accepts the integrity of its CA, not just to those with whom it entrusts a shared secret.
- Because a host can authenticate itself without having to share its private key, it need never expose the key, verbally, in writing, or over the Internet.

The entire system for authentication with digital certificates—the individual hosts, their certificates, and trusted CAs—is called the public key infrastructure (PKI).

**CAs.** The first step in obtaining a certificate is selecting a CA. In some ways, the CA is the most vulnerable point in the PKI. Digital certificates rely on robust algorithms and asymmetric keys that hackers cannot crack. However, strong certificates do not protect against a hacker who obtains a certificate using false credentials because the certificate itself is valid. For this reason, it is very important that your CA be reputable and trusted. It should have vigorous standards for ensuring that it issues certificates only to hosts submitting their own authentic information.

**Digital Signature Standards.** CAs can use one of several algorithms to encrypt data. The ProCurve Secure Router supports:

- Digital Signature Standard (DSS)
- Rivest-Shamir-Adleman Signature (RSA)

DSS, which is the U.S. government authentication standard, uses the Digital Signature Algorithm (DSA) to create public and private keys.

RSA is the most commonly used algorithm and is extremely secure.

Your CA will tell you which standard it uses. You should configure this standard in the IKE attribute policy. (See the discussion of authentication methods in “IKE Phase 1” on page 10-8.)

**CA Servers.** You use a CA server to obtain certificates from a CA. If the CA server that you select supports Simple Certificate Enrollment Protocol (SCEP), the ProCurve Secure Router can download and import certificates from it automatically.

Otherwise, you will have to navigate the server’s Web site to request and download certificates. You then paste these into the ProCurve Secure Router’s CLI.

You will need to use the server to obtain at least two certificates:

- a CA certificate
- a personal, or self, certificate

**CA Certificate.** The router uses the CA certificate to decrypt and check the CA’s digital signature. A CA includes its signature in all the certificates that it issues, attesting that the identification information is accurate for the host holding the certificate. The router’s system must include a CA certificate for each CA from which it receives a certificate and from which it accepts certificates.

The CA certificate can be either a root certificate, which a CA issues to itself, or a subordinate certificate, which a CA issues to a subordinate CA.

**Self Certificate.** The self certificate is the certificate the router uses to authenticate its own identity. It includes:

- the router’s identification information
- the router’s public key
- the CA’s signature

When the router sends a certificate to a peer, it adds its own signature by encrypting the certificate with its private key.



## Obtaining Digital Certificates

First, select a CA server.

If your CA server supports SCEP, you must complete three steps to load the necessary certificates into the ProCurve Secure Router's operating system:

1. Create a CA profile.
2. Load the CA certificate.
3. Generate a self certificate request.

If your CA server does not support SCEP, you must complete four steps to load the certificates manually:

1. Create a CA profile.
2. Load the CA certificate.
3. Generate a self certificate request.
4. Import a self certificate and CRL. The CRL, which lists certificates issued to hosts and when they expire, allows the router to determine whether a peer's certificate is still valid.

These are only the steps you need to complete on the ProCurve Secure Router. For manual loading, you will also need to download the CA certificate from your CA server, submit the request for a self certificate to the server, and download the self certificate and CRL. This guide assumes that the certificates necessary for each step are ready to be copied and pasted into the CLI.

**Creating a CA Profile.** You must configure a profile for a CA before you can load its certificate into the system. Create a CA profile from the global configuration mode context:

**Syntax:** `crypto ca profile <profile name>`

For example:

```
ProCurve(config)# crypto ca profile MyCA
ProCurve(ca-profile)#
```

In the CA's profile, you *must* configure the enrollment method.

If the ProCurve Secure Router will automatically load your certificates using SCEP, you should input the URL for your CA server's Web site:

**Syntax:** `enrollment url [http://<FQDN>]/[<client program name>]`

For example:

```
ProCurve(ca-profile)# enrollment url http://isakmp-test.ssh.fi/
```

The domain name should be fully qualified. If you do not include a program name, the router will use the default program `pkiclient.exe`.

If you will be loading certificates manually, use this option for the command:

```
ProCurve(ca-profile)# enrollment terminal
```

---

**Note**

---

The **url** and **terminal** options are mutually exclusive, and the most recently entered option takes precedence. For example, if you enter a URL for your CA server and then enter **enrollment terminal**, the URL will be erased.

Refer to Table 10-18 for the commands for specifying various information about the router in the CA profile. The ProCurve Secure Router OS uses this information when you configure it to generate the self certificate request to this CA.

Entering this information now is optional. You can also enter this information later, in a dialog box, when you actually generate the request. Note that your VPN peer must match this information for the local router in its remote ID list.

**Table 10-18. Adding Information for a Self Certificate Request to a CA Profile**

Information	Command Syntax
IP address	<b>ip-address</b> <A.B.C.D>
domain name	<b>fqdn</b> <domain name>
email address	<b>email-address</b> <email address>
subject name	<b>subject-name</b> <name>
serial number	<b>serial-number</b>
password	<b>password</b> <password>

**Loading a CA Certificate.** You load a CA certificate from the global configuration mode context:

**Syntax:** `crypto ca authenticate <profile name>`

Specify the profile configured for the CA. For example:

```
ProCurve(config)# crypto ca authenticate MyCA
```

If you are using automatic enrollment, you only need to enter the command. Then press **y** to accept the certificate that the OS automatically loads.

If you are obtaining the certificate manually, follow the directions in the CLI to cut and paste the certificate into the command line. (See Figure 10-9.)

You should have already downloaded the certificate from your CA server. The certificate must be in Privacy Enhanced Mail (PEM) format.

```

ProCurve(config)# crypto ca authenticate MyCA
Enter the base 64 encoded CA certificate. End with two consecutive
carriage returns or the word "quit" on a line by itself:
-----BEGIN X509 CERTIFICATE-----
MIICTDCCAbWgAwIBAgICAS0wDQYJKoZIhvcNAQEFBQAwwjELMAkGA1UEBhMCRkxx
JDAiBgNVBAoTG1NTSCBDb21tdW5pY2F0aW9ucyBTZWV1cm10eTERMA8GA1UECXMl
V2ViIHRlc3QxEjAQBgNVBAMTCVRlc3QgQ0EgMTAeFw0wMzAxMDkxNjI1MTVaFw0w
MzEyMzEyMzU5NTlaMFoxCzAJBgNVBAYTAkZJMSQwIgyYDVQKExtTU0gggQ29tbXVu
aWNhdGlvbnMgU2VjdXJpdHkxETAPBgNVBAsTCFdlYiB0ZXN0MRIwEAYDVQQDEw1U
ZXN0IENBIDUwDQYJKoZIhvcNAQEBBQADgYSAMIghAAoGBAI3wb1DaZUvk7L+d
sQxr8hD7YFSqU1Ty6xJFKj7DzgulhU9w5JIt83qxeXp1aMcjhK//00feFhM41EH+
JNi3Qk4Hbcwqtzmz4jFW58ib0GSWq9LR7hFdakDVKQJtICPLM9zZ8PY1Red04wwiH
IGCPKBZJd1/FjC3wyaw4CKgnJ5jTAgEloyMwITALBgNVHQ8EBAMCAYYwEgYDVR0T
AQH/BAGwBgEB/wIBMjANBgkqhkiG9w0BAQUFAAOBgQB0keEUE3E5bleCBKUMOKguX
zu8K0TlPkFtC3y37j3Ub4CRKcRuwbt2qLwfdZAwYfxTBb6C+0o4Diyi2dBqIBTnW
7Qami34yS/3ebz0LF4PZTLj9SUP1mIp6Dyf2trky3AQQN4JHFGdShThY2+ehlRjF
z7FLEJ7/xDDhd2I3IN5W9A==
-----END X509 CERTIFICATE-----
Hash: 81df9e48f5e9e8f4409ab407ce9c72ce
* Do you accept this certificate? [y]
CA certificate was successfully added.
ProCurveSR7102dl(config)#
  
```

**Figure 10-9. Manually Loading a CA Certificate**

**Note**

In order to load a CA certificate automatically, the CA's URL must be configured in its profile. You do not have to configure a URL if you are loading the certificate manually, but you still must configure a CA profile.

**Generating a Self Certificate Request.** After you load a CA certificate, you must request a self certificate from the CA. From the global configuration mode context, enter:

**Syntax:** `crypto ca enroll <profile name>`

The OS will then initiate a dialog with you. (See Figure 10-10.) The OS will ask you to enter any information that you have not already configured from the CA profile configuration mode context.

```

ProCurve(config)# crypto ca enroll MyCA
**** Press CTRL+C to exit enrollment request dialog. ****
* Enter signature algorithm (RSA or DSS) [rsa]:
* Enter the modulus length to use [512]:1024
* Enter the subject name as an X.500 (LDAP)
DN:Router,C=US,L=Roseville,S=CA
--The subject name in the certificate will be
CN=CN=Router,C=US,L=Roseville,S=CA.
* Include IP address in subject alternate name [n]:y
* Enter IP address or name of interface to use:10.10.10.1
* Include fully qualified domain name [n]:
* Include an email address [n]:
Generating request (including keys)....
.....Done
* Display certificate request to terminal? [y]
----BEGIN CERTIFICATE REQUEST-----
MIIBoJCCAQsCAQAwQDELMAKGA1UEBhMCVVMxCzAJBgNVBAGTAkFMMRMwEQYDVQQH
EwpIdW50c3ZpbGx1MQ8wDQYDVQQDEwZSb3V0ZXIwZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALCZJHE4Bx1+ZzJQUFEQwfGcI2vE9RR68KqRmwGr+hZyMi49Eg85
UIBCUvCe15u/P4JYS9d//jT37+uh+jZxNvSUiPa99XatHcIcVLWQHZrn9+vWYReu
7418ZEBx/ETPKa6XqSpFqJ5ZG38VUaN05r2gwW+aZtsfxOyMUupstSypAgMBAAGg
IjAgBgkqhkiG9w0BCQ4xEzARMA8GA1UdEQQIMAaHBAoKCgEwDQYJKoZIhvcNAQEF
BQADgYEAidRLWBpm5pNn38VylSXqvTrEAZtWNRSTxBEMnObbg+buHPIQet1bOpu
QF1wmBJUehLkVYlnmO4Di6IFcJbnF0AD/Jcoiw5jgRZdbcWSpIOg5uqXFpdlbtbg4
pIY+ZWviGolKZH00EJuuzywFUF74QPScVf6Ci6eI1619cYntP14=
-----END CERTIFICATE REQUEST-----
* Redisplay certificate request to terminal? [n]

```

**Figure 10-10. Requesting a Self Certificate**

When you submit the request to the CA, the CA will add this information to the router's self certificate. When your VPN peer or peers define your router as one of the remote peers allowed to connect to their VPN, they must match the value entered for one of these fields.

If you are obtaining certificates manually, answer yes (y) when the CLI asks if you want to display the certificate request to the terminal. Copy the request and submit it to your CA server.

If you are obtaining certificates automatically, the OS will submit the request for you. It will also automatically load the self certificate and a CRL into the CA profile. You will then have completed obtaining your certificates.

**Importing a Self Certificate and CRL.** You only need to complete this step if you obtaining certificates manually.

After your CA server has sent you a self certificate and CRL, you must import them into the CA profile configured on the router. Enter the following commands from the global configuration mode context:

**Syntax:** `crypto ca import <profile name> [certificate | crl]`

For example:

ProCurve(config)# `crypto ca import MyCA certificate`

Figure 10-11 illustrates how you should import the self certificate.

**Note**

Even if you already have a certificate ready to import, you have to load a CA certificate into the profile before you can import a self certificate or CRL.

```

ProCurveSR7102dl(config)# crypto ca import MyCA certificate
Enter the PM-encoded certificate. End with two consecutive
carriage returns or the word "quit" on a line by itself:
-----BEGIN X509 CERTIFICATE-----
MIICZTCCAc6gAwIBAgIEP5/c2TANBgqhkiG9w0BAQUFADBAMQswCQYDVQQGEWJG
STEkMCIGA1UEChMbU1NIEENvbW11bmljYXRpb25zIFNlY3VyaXR5MREwDwYDVQQL
EwhXZWlmdGVzdDESMBAGA1UEAaXzVGVzdBQSAxMjE4XDAzMTAyOTAwMDAwMFMX
DTAzMTIwMTAwMDAwMFowQDELMAkGA1UEBhMCVVMxZCZAJBgNVBAGTAkFMMRMwEQYD
VQHEWpI dW50c3ZpbGx1MQ8wDQYDVQQDEwZSb3V0ZXIwZ8wDQYJKoZIhvcNAQEB
BQADgY0AMIGJAoGBALCZJHE4Bx1+ZzJQUFEQwFgC I2vE9RR68KqRmwGr+hZyMi49
Eg85UIBCUvCe15u/P4JYS9d//jT37+uh+jZxNvSUIPa99XatHcIcVLWQHZrn9+vW
YReu7418ZEbx/ETPKa6XqSpFqJ5ZG38VUaN05r2gww+aZtsfxOyMUupstSypAgMB
AAGjUjBQMAsgA1UdDwQEAwIFoDAPBgNVHRECDAGhWKCgoBMDAGA1UdHwQpMCCw
JaAjoCGGH2h0dHA6Ly9sZGFwLnNzaC5maS9jcmxzL2NhMS5jcmwwDQYJKoZIhvcN
AQEFBQADgYEAObiCh0AtS1q1Ic0lfg1huYUcczLkqYm2UQ6uSvi0rpmgiEnIVqH+
y8at3D4Mr1xCGzTqSuXf7uAxCHwkjwS6OVw2wERicy9X1j28XzKjZGpo3Z6aQPax
ZBUvo6EcYtKxtiD1ONTzrxqBC9bvcV0pipzWleTY1pwPKzRvCuopiqA=
-----END X509 CERTIFICATE-----
quit
Success!
ProCurveSR7102dl (config)#
  
```

**Figure 10-11. Manually Importing a Self Certificate**

**Managing Certificates**

The certificates configured on the ProCurve Secure Router vouch for your organization's identity and integrity. It is very important that the information in them be correct and up to date.

**Viewing Certificates.** You can use the **show crypto ca** commands to view:

- certificates
- CRLs
- CA profiles

Enter the command from the enable mode context:

**Syntax:** show crypto ca [certificates | crls | profiles]

For example:

```
ProCurve# show crypto ca certificates
```

The **certificates** option shows both CA and self certificates. View certificates to verify that the information in them is correct. You should also keep track of when your certificates expire and periodically update them.

If the information in a self certificate is incorrect, you should view the CA profile. Information may have been miskeyed into the profile, which would cause the OS to submit incorrect information to the CA server.

Figure 10-12 shows a sample display of certificates loaded on a router.

```

ProCurve# show crypto ca certificates
CA Certificate
  Status: Available
  Certificate Serial Number: 012d ← Use when deleting
  Subject Name: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test
CA 1
  Issuer: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test CA 1
  CRL Dist. Pt: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test
CA 1
  Start date is Jan  9 16:25:15 2003 GMT
  End date is Dec 31 23:59:59 2003 GMT
  Key Usage:
                                Use when deleting
Self Certificate
  Status: Available
  Certificate Serial Number: 3f9fdcd9
  Subject Name: /C=US/ST=CA/L=Roseville/CN=Router ← Router identification
  Issuer: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test CA 1
  CRL Dist. Pt: /C=FI/O=SSH Communications Security/OU=Web test/CN=Test
CA 1
  Start date is Oct 29 00:00:00 2003 GMT
  End date is Dec  1 00:00:00 2003 GMT
  Key Usage:

```

**Figure 10-12. Viewing Certificates**

An up-to-date CRL is also very important. The CRL determines which certificates the router accepts from peers and thus determines which peers can access the VPN.

**Deleting Certificates.** Follow this process to delete a certificate:

1. View the certificate using the **show** command.
2. Find the certificate's serial number.
3. Move to the global configuration mode context and access the **certificate chain** command set for the corresponding CA profile:

**Syntax:** crypto ca certificate chain <profile name>

4. Delete the certificate:

**Syntax:** no certificate [ca <serial number> | <serial number>]

The **ca** keyword indicates that the serial number refers to a CA certificate. If you want to delete a self certificate, enter the serial number without the **ca** keyword.

For example to delete the self certificate shown in Figure 10-12, enter:

```
ProCurve(config)# crypto ca certificate chain MyCA  
ProCurve(config-cert-chain)# no certificate 3f9fdcd9
```

---

**Note**

---

The Secure Router OS uses the commands in the **certificate chain** command set to load certificates. You should only use these commands to delete certificates.

**Managing CRLs.** A CRL is a list of digital certificate subscribers. It includes information about each subscriber's certificates, including:

- current status
- date of issue
- CA from which the certificate was obtained

The CRL also lists revoked certificates, accompanied by the cause for the revocation.

IKE uses the CRL to help determine whether a peer can be trusted to connect over the VPN tunnel. To keep your private network secure, you should make sure that the CA profile contains an up-to-date CRL.

To delete a CRL:

1. Access the **certificate chain** command set for the corresponding CA profile:

**Syntax:** `crypto ca certificate chain <profile name>`

2. Delete the CRL:

```
ProCurve(config-cert-chain)# no crl
```

If the ProCurve Secure Router OS does not contain a CRL, the router will accept all certificates signed by the CA as authentic.

## Configuring a VPN using IPSec with Manual Keying

IKE manages the generation of keys automatically using the Diffie-Hellman key exchange protocol. Using IKE offers several advantages. IKE:

- relieves an often over-extended IT staff from configuring cumbersome keys
- eliminates the need to communicate keys between two sites, thus closing a vulnerability window
- periodically changes keys for heightened security



For these reasons, you are advised to always use IKE with IPSec.

However, if you are establishing a VPN with a site that does not support IKE, you will have to use manual keying. To maintain security and reduce the chance of misconfigurations, you should only use manual keying to connect two sites managed by the same IT staff.

To configure a VPN with manual keying, you must complete all steps described for configuring IPSec with IKE except those related to IKE phase 1. You must:

1. Install the IPSec VPN module and enable **crypto** commands.
2. Define the networks included in the VPN.
3. Configure a transform set.
4. Configure a crypto map entry.
5. Apply the crypto map to a WAN interface

This section will explain how to configure a crypto map entry that uses manual keying. (You perform the other steps exactly as you would to configure IPSec with IKE. See “Configuring IPSec with IKE” on page 10-15 for instructions.)

When you use manual keying, you take over IKE’s task during phase 2 and define the keys for the IPSec SA. (The router does not establish a preliminary SA, which eliminates the purpose of IKE phase 1.)

In the crypto map entry, you must define:

- hash and/or encryption algorithms (in a transform set)
- inbound and outbound SPIs
- a unique inbound key for each algorithm
- a unique outbound key for each algorithm
- an IPSec SA lifetime

You must also define:

- the peer’s remote ID
- traffic allowed to access the tunnel

## Configuring the Transform Set

The transform set contains the algorithms used to secure data. You create the transform set from the global configuration mode context with this command:

**Syntax:** crypto ipsec transform-set <setname> [ah-sha-hmac | ah-md5-hmac] [esp-aes-256-cbc | esp-aes-192-cbc | esp-3des | esp-aes-128-cbc | esp-des | esp-null] [esp-sha-hmac esp-md5-hmac]

You must select at least one algorithm. You can select one each of an AH hash, ESP encryption, or an ESP hash algorithm. (See Table 10-19.) For example, enter:

```
ProCurve(config)# crypto ipsec transform-set T1 ah-md5-hmac esp-3des esp-sha-hmac
```

See “Transform Sets” on page 10-40 to learn more about transform sets.

**Table 10-19. Transform Sets**

Transform Set Protocol	Algorithm Types	Algorithm Options (Most Secure to Least Secure)	Command Syntax
AH	hash algorithm	<ul style="list-style-type: none"> <li>• SHA</li> <li>• MD5</li> </ul>	<b>crypto ipsec transform-set &lt;setname&gt; [ah-sha-hmac   ah-md5-hmac]</b>
ESP	<ul style="list-style-type: none"> <li>• encryption algorithm</li> <li>• hash algorithm (optional, unless encryption is not used)</li> </ul>	<ul style="list-style-type: none"> <li>• encryption: <ul style="list-style-type: none"> <li>– AES (256-bit key)</li> <li>– AES (192-bit)</li> <li>– 3DES</li> <li>– AES (128-bit)</li> <li>– DES</li> <li>– None</li> </ul> </li> <li>• hash: <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> </ul>	<b>crypto ipsec transform-set &lt;setname&gt; [esp-aes-256-cbc   esp-aes-192-cbc   esp-3des   esp-aes-128-cbc   esp-des   esp-null] [esp-sha-hmac   esp-md5-hmac]</b>
AH and ESP	<ul style="list-style-type: none"> <li>• AH hash algorithm</li> <li>• ESP encryption algorithm</li> <li>• ESP hash algorithm (optional)</li> </ul>	<ul style="list-style-type: none"> <li>• AH hash: <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> <li>• ESP encryption: <ul style="list-style-type: none"> <li>– AES (256-bit key)</li> <li>– AES (192-bit)</li> <li>– 3DES</li> <li>– AES (128-bit)</li> <li>– DES</li> <li>– None</li> </ul> </li> <li>• ESP hash: <ul style="list-style-type: none"> <li>– SHA</li> <li>– MD5</li> </ul> </li> </ul>	<b>crypto ipsec transform-set &lt;setname&gt; [ah-sha-hmac   ah-md5-hmac] [esp-aes-256-cbc   esp-aes-192-cbc   esp-3des   esp-aes-128-cbc   esp-des   esp-null] [esp-sha-hmac esp-md5-hmac]</b>

The algorithms you choose determine the minimum length for the key that defines the IPSec SA. For example, 3DES uses a 192-bit key. You will input the key in HEX (rather than true ANSI). Table 10-20 displays the minimum key lengths for various algorithms.

**Table 10-20. Key Lengths for Standard Algorithms**

Algorithm	Minimum Key Length in Bits	Minimum Key length in HEX
SHA	160	20
MD5	128	16
AES	<ul style="list-style-type: none"> <li>• 128</li> <li>• 192</li> <li>• 256</li> </ul>	<ul style="list-style-type: none"> <li>• 16</li> <li>• 24</li> <li>• 32</li> </ul>
DES	64	8
3DES	192	24

### Configuring Crypto Maps for Manual IPsec

You define the IPsec SA in a crypto map entry. First create a crypto map entry that uses manual keying:

**Syntax:** `crypto map <mapname> <map index> [ipsec-ike | ipsec-manual]`

For example:

```
ProCurve(config)# crypto map VPN 20 ipsec-manual
```

**Specifying the Transform Set.** Associate the crypto map entry with up to six transform sets. Use this command, entered from the crypto map configuration mode context:

**Syntax:** `set transform-set <setname> [<setname2>] [<setname3>] [<setname4>] [<setname5>] [<setname6>]`

**Defining the Keys.** You then use the `set session-key` command to define the keys that secure the IPsec SA. In this command, you specify:

1. whether the key is for inbound or outbound traffic
2. the key protocol (AH or ESP)
3. SPI
4. encryption key (for ESP)
5. authentication key (optional for ESP)

Use the commands shown in Table 10-21 to configure the keys for the IPsec SA. When you enter the key in HEX, do *not* enter the initial 0x for each character.

Each crypto map entry should include one inbound and one outbound key for the protocol(s) selected in the associated transform sets. If you have selected more than one transform set, then the key must meet the longest minimum length requirement.

When the router transmits a packet selected by this crypto map entry's ACL, it encrypts and hashes the packet using the outbound keys. It also inserts the *outbound* SPI into the IPSec header. When the peer router receives the packet, it matches the SPI to an *inbound* session-key configured in its crypto map entry. It then uses the associated keys to decrypt and de-hash the packet. Therefore, you must match the *outbound* SPI and keys on one router to the *inbound* SPI and keys on the peer router, and vice versa.

**Table 10-21. Manual Keys**

Key Type	Key Matches	Function	Command Syntax
inbound ESP key	peer's outbound key	decrypts (and authenticates) data received from a peer	<b>set session-key inbound esp</b> <b>&lt;SPI&gt; cipher &lt;HEX key&gt;</b> <b>[authenticator &lt;HEX key&gt;]</b>
outbound ESP key	peer's inbound key	encrypts (and authenticates) data sent to a peer	<b>set session-key outbound esp</b> <b>&lt;SPI&gt; cipher &lt;HEX key&gt;</b> <b>[authenticator &lt;HEX key&gt;]</b>
inbound AH key	peer's outbound key	authenticates data received from a peer	<b>set session-key inbound ah &lt;SPI&gt;</b> <b>&lt;HEX key&gt;</b>
outbound AH key	peer's inbound key	authenticates data sent to a peer	<b>set session-key outbound ah</b> <b>&lt;SPI&gt; &lt;HEX key&gt;</b>

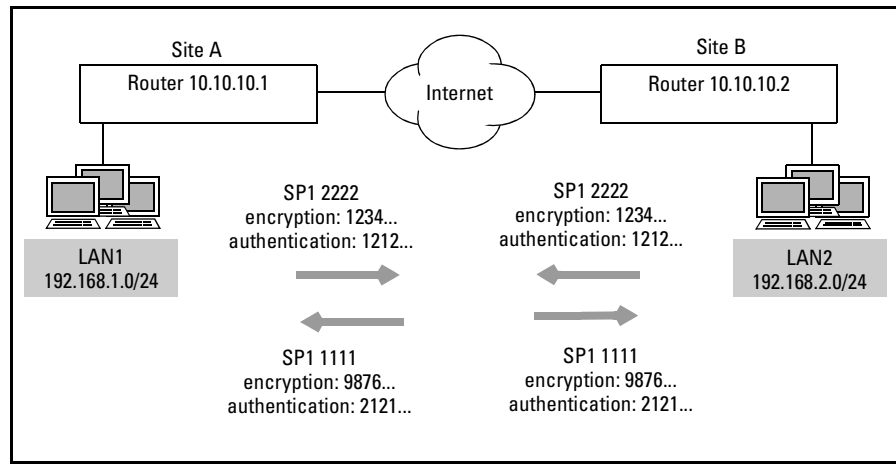
**Other Crypto Map Entry Configurations.** In the crypto map entry, you must also define the peer's remote ID—its public IP address. For example, enter:

```
ProCurve(config-crypto-map)# set peer 10.2.2.1
```

You also must match the crypto map to an ACL. This ACL should permit traffic between the local and remote networks that are included in the VPN:

```
ProCurve(config-crypto-map)# match address VPNTraffic
```

See “Crypto Maps” on page 10-42 for more detailed discussion on setting these and other parameters, such as the IPSec SA lifetime.



**Figure 10-13. Example VPN Configuration with Manual Keying**

### Example Configuration

Figure 10-13 shows Site A and Site B, whose LANs need to connect through the Internet. Site A's inbound key and SPI match Site B's outbound key and SPI and vice versa. The following are the configurations for Router A:

```
ProCurve(config)# ip crypto
ProCurve(config)# ip access-list extended VPNTraffic
ProCurve(config-ext-nacl)# permit ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
ProCurve(config-ext-nacl)# exit
ProCurve(config)# crypto ipsec transform-set T1 esp-3des esp-md5-hmac
ProCurve(config)# crypto map VPN 0 ipsec-manual
ProCurve(config-crypto-map)# set transform-set T1
ProCurve(config-crypto-map)# set session-key outbound esp 2222 cipher
123456789876543212345678 authenticator 2121212121212121
ProCurve(config-crypto-map)# set session-key inbound esp 1111 cipher
987654321234567898765432 authenticator 1212121212121212
ProCurve(config-crypto-map)# exit
ProCurve(config)# int ppp 1
ProCurve(config-ppp 1)#crypto map VPN
```

## Monitoring a VPN

You can monitor the VPN tunnels supported on your router. Enter this enable mode command to view all active SAs:

**Syntax:** show crypto [ike | ipsec] sa

Enter the **ike** keyword to view IKE SAs, which are open only temporarily to allow peers to negotiate a VPN connection securely.

Enter the **ipsec** keyword to view IPSec SAs, which are the VPN tunnels over which secured data travels. The router establishes a separate SA for each connection between a peer on the local network and a remote peer. (See Figure 10-14.)

```
IPSec Security Associations: Total IPSec SAs: 2

Peer IP Address: 10.2.2.1
Mode-config Address: 192.168.100.1
Direction: Inbound
SPI: 0x9AF31804 (2599622660)
Encapsulation: ESP
RX Bytes: 0
Selectors: Src:192.168.100.1/255.255.255.255 Port:ANY Proto:ALL IP
           Dst:10.1.30.0/255.255.255.0 Port:ANY Proto:ALL IP
Hard Lifetime: 28760
Soft Lifetime: 0
Crypto Map: VPN 10

Peer IP Address: 192.168.5.23
Mode-config Address: 192.168.100.1
Direction: Outbound
SPI: 0xB4E0AE5F (3034623583)
Encapsulation: ESP
TX Bytes: 0
Selectors: Src:10.1.30.0/255.255.255.0 Port:ANY Proto:ALL IP
           Dst:192.168.100.1/255.255.255.255 Port:ANY Proto:ALL IP
Hard Lifetime: 28760
Soft Lifetime: 28670
Crypto Map: VPN 10
```




Figure 10-14. Viewing an IPSec SA

If you determine that a VPN connection has been established that should not have been, you can enter one of these enable mode commands to terminate it:

**Syntax:** clear crypto ipsec sa entry <A.B.C.D> [ah | esp] <SPI>

**Syntax:** clear crypto ipsec sa peer <A.B.C.D>

Use the **entry** keyword to clear one of the SAs that displays when you enter the **show crypto ipsec sa** command. You can also find the SPI and the protocol in the display.

Use this command to clear all IPSec SAs established using a specific crypto map:

**Syntax:** clear crypto ipsec sa map <mapname>

For example, if you change the security policies defined in a crypto map, you must clear SAs already established using the map in order for the new settings to take effect.

You can clear all IPSec SAs on the router with this command:

**Syntax:** clear crypto ipsec sa

To clear IKE SAs, enter this command:

**Syntax:** clear crypto ike sa [<policy number>]

If you only want to clear the IKE SAs associated with a particular IKE policy, enter the number of that policy. For example, when you change the security policies in an IKE policy, you should clear any SAs already established using the old policies.

You can use **show** commands displayed in Table 10-22 to view:

- IKE and IPSec SAs
- IKE policies
- transform sets
- crypto maps
- remote ID and preshared key list
- client configuration pools
- ACLs

**Table 10-22. VPN show Commands**

<b>View</b>	<b>Command Syntax</b>
all IKE SAs	<b>show crypto ike sa</b>
all IPsec SA	<b>show crypto ipsec sa</b>
all IPsec SA to a specific peer	<b>show crypto ipsec sa address &lt;A.B.C.D&gt;</b>
all IPsec SA established with a specific crypto map	<b>show crypto ipsec sa map &lt;mapname&gt;</b>
all IKE policies	<b>show crypto ike policy</b>
specific IKE policy	<b>show crypto ike policy &lt;pollicynumber&gt;</b>
all transform sets	<b>show crypto ipsec transform-set</b>
specific transform set	<b>show crypto ipsec transform-set &lt;setname&gt;</b>
all crypto maps	<b>show crypto map</b>
specific crypto map	<b>show crypto map &lt;mapname&gt;</b>
specific crypto map entry	<b>show crypto map &lt;mapname&gt; &lt;mapindex&gt;</b>
crypto map associated with a specific interface	<b>show crypto map interface &lt;interface type&gt; &lt;interface number&gt;</b>
remote ID and preshared key list	<b>show crypto ike remote-id</b>
all client configuration pools	<b>show crypto ike client configuration pool</b>
specific client configuration pool	<b>show crypto ike client configuration pool &lt;poolname&gt;</b>
all ACLs	<b>show ip access-lists</b>
specific ACL	<b>show ip access-lists &lt;listname&gt;</b>
portion of the running-config dealing with the VPN configuration	<b>show running-config crypto</b>



## Troubleshooting a VPN That Uses IPSec

When you have correctly configured a VPN, it should quickly go up. You can verify that the VPN has been established by pinging a location on the remote network from the local network. The first few packets may be dropped while IKE negotiates the IPSec SA and establishes the VPN tunnel. However, after several attempts you should be able to ping the remote location.

If you cannot ping the remote location, you will need to troubleshoot the VPN connection.

### Tools and Procedures

Because you must configure many settings on at least two devices, many of which must match exactly, it is easy to make a misconfiguration. Searching for the problem can be frustrating, particularly if you do not know where to begin looking for it.

The following procedure will help you pinpoint a problem:

1. Check WAN connections.
2. Apply an ACL that permits all traffic to the crypto map.
3. Activate crypto debug messages and reattempt the connection.

---

### Caution

---

Debug messages can be very draining on the router's processor and can compromise network performance.

4. Compare configurations between the local router and the peer and make any necessary changes.
5. If you cannot find the peer's settings, return VPN policies to their defaults and reattempt the connection.

The first step winnows out problems with the Physical (Layer 1) and Data Link (Layer 2) connection to the Internet. The second step is designed to check for a problem in selecting traffic for the VPN before you waste time looking for mismatched security policies. In the third step, you track IKE negotiations to discover where the process of establishing the connection breaks down. In the fourth step, you search for mismatched configurations, using the knowledge you gained in the second step. Fix any misconfigurations so that

the local router's settings for this VPN connection exactly match those of the peer. If you are unable to learn the peer's settings, you can try using default settings to connect to the peer in the fifth step.

## Troubleshooting Commands

The tools you will use as you follow this procedure are the **show** and **debug** commands, which are enable mode commands. Preface the commands with the **do** keyword to execute them from the configuration mode contexts so that you can fix problems while you troubleshoot.

You can use the **show** commands displayed in Table 10-22 on page 10-72 to view:

- IKE and IPSec SAs
- IKE policies
- transform sets
- crypto maps
- remote ID and preshared key list
- client configuration pools
- ACLs

**Debug** commands display messages in real-time as they are received on the router. The debug messages displayed when you enter the commands shown in Table 10-23 give you valuable information about the IKE process.

---

### Note

---

Debug messages are processor-intensive and can seriously degrade network performance. Take care when using **debug** commands.

**Table 10-23. VPN debug Commands**

View	Command Syntax
all IKE debug messages	<b>debug crypto ike</b>
IKE key management messages (i.e. IKE phase 1 and 2)	<b>debug crypto ike negotiation</b>
IPSec debug messages (messages received after the tunnel has been established)	<b>debug crypto ipsec</b>
digital certificate messages	<b>debug crypto pki</b>
IKE client configuration messages	<b>debug crypto ike client configuration</b>
Xauth messages	<b>debug crypto ike client authentication</b>

## Checking WAN Connections

Before you waste time searching through convoluted configurations for an error, you should verify that your connection to the Internet (or other public network) is up.

Check that the Physical (Layer 1) connection is good and the Data Link (Layer 2) state is open. See the *Basic Management and Configuration Guide, Chapter 6: Configuring the Data Link Layer Protocol for E1, T1, and Serial Interfaces* or *Chapter 7: ADSL WAN Connections* for tips on troubleshooting a WAN connection.

## Determining the Source of the Problem: Permitting All Traffic in a VPN

Problems with VPN configurations can be broken down into two general categories:

- mismatched security parameters
- problems with the network addressing, including errors in:
  - peer's ID
  - networks permitted to access the VPN tunnel

Often you will have reason to believe that mismatched security parameters are a problem. A debug message such as “NO\_PROPOSAL\_CHOSEN” will appear.

However, when you do not know where to start looking for a problem, it is often advisable to rule out problems with network addressing before searching through the many security parameters.

Permitting all traffic in the VPN allows you to determine whether you can reach the peer at all.

Move to the configuration mode context for the ACL that selects VPN traffic. Add an entry that permits all traffic:

```
ProCurve(config-ext-nacl)# permit ip any any
```

Again attempt to make the connection. If the tunnel remains shut, incompatible security policies are most likely at fault. (It is also possible that the ACL has a “deny ip any any” entry at the beginning.)

However, if the tunnel opens, then you know that you have a problem with the ACL. Enter:

**Syntax:** show ip access-list <listname>

Review the ACL, looking for miskeyed entries or problems with the wildcard bits. Remember that for a client-to-site VPN, the destination should be the network in the IKE client configuration pool. See *Chapter 5: Applying Access Control to Router Interfaces* for more information on how to correctly configure an extended ACL.

To change an ACL entry, first enter the **no** form of the faulty entry to remove it from the list. Then enter the correct entry. Do *not* simply enter the correct entry without removing the incorrect one. The router processes ACLs in the order in which you enter the commands, so the faulty entry may continue to cause problems unless entirely removed.

### Monitoring the IKE Process using Debug Commands

To monitor the IKE process, enter:

```
ProCurve# debug crypto ike
```

---

#### **Note**

---

You should deactivate any active debug messages (enter **undebug all**) before activating the IKE messages.

You will receive a great many debug messages from IKE as it attempts three times to establish a connection. Look at the final messages first as these will give you a clue to the source of the problem. (Table 10-24 gives some examples of messages that appear due to common problems with the VPN.)

**Table 10-24. Debug Messages**

Message	Possible Problem	Best Next Step
<b>NO_PROPOSAL_CHOSEN</b>	incompatible security parameters	Determine whether negotiations failed at IKE phase 1 or phase 2.
<b>IKEStartNegotiation: could not find an IKE policy to use</b>	no IKE policy is configured for the peer set in the crypto map entry	Compare peer ID in the crypto map entry and IKE policy.
<ul style="list-style-type: none"> <li>• <b>IkeGetPreSharedKey failed</b></li> <li>• <b>IKEIDWaitProcess</b></li> </ul>	invalid authentication information	<ul style="list-style-type: none"> <li>• Double-check your preshared key with your peer.</li> <li>• Double-check the ID in the remote ID list and verify that it matches the peer's. If you are using digital certificates, make sure that the remote ID exactly matches that in authorized certificates.</li> <li>• Renew your certificate and CRL.</li> </ul>

The key to interpreting debug messages in order to pinpoint a problem with a VPN connection is understanding how IPSec, and particularly IKE, establish the VPN tunnel. IKE follows a set process for communicating with and authenticating a peer, negotiating security parameters, and bringing up first the IKE SA and then the IPSec SA, or VPN tunnel. By tracking this process, you can pinpoint exactly where the IKE negotiations derail. You will then know where to look for a misconfiguration.

IKE completes the following steps:

1. IKE phase 1 (main or aggressive mode)
  - a. proposes (or accepts) security parameters (main mode messages 1 or 2, aggressive mode message 1 or 2) including:
    - i. a hash algorithm
    - ii. a encryption algorithm
    - iii. an authentication method
    - iv. an IKE SA lifetime
  - b. generates keys using Diffie-Hellman key exchange (main mode message 3 or 4, aggressive mode message 1 or 2)
  - c. authenticates the peer and establishes the IKE SA (main mode message 5 or 6, aggressive mode message 3)

2. IKE phase 2 (quick mode)
  - a. proposes (or accepts) security parameters including:
    - i. a hash algorithm (optional for ESP)
    - ii. an encryption algorithm (optional for AH)
    - iii. an IPSec SA lifetime
  - b. generates keys
  - c. establishes the IPSec SA

When you scan debug messages for clues to the source of a problem, pay particular attention to messages that indicate the step that IKE is performing. You can then determine what settings you need to modify. You will learn more about specific problems and debug messages in the following pages.

IKE phase 2 problems are nearly always caused by incompatible security proposals for the IPSec SA. IKE phase 1, on the other hand, involves more steps and can go wrong in various ways. If you determine that problems begin in IKE phase 1, you should then zero in on the message that fails. Look for the message that IKE sends over and over. (See Table 10-25.)

**Table 10-25. IKE Debug Messages**

Message That Repeats	Possible Problem	Best Next Step
<b>main mode message 1</b>	incompatible IKE modes or security parameters	Compare IKE attribute policy with the peer's settings.
<b>main mode message 5</b>	invalid authentication information	Double-check preshared keys and certificates.
<b>aggressive mode message 1</b>	incompatible IKE modes or security parameters	Compare IKE attribute policy with peer's settings.
<b>aggressive mode message 3</b>	invalid authentication information	Double-check preshared keys and certificates.
<b>quick mode message 1</b>	incompatible IPSec security parameters	Compare crypto map entry and transform set settings with the peer's settings.

**Incompatible Security Parameters.** When you receive the NO\_PROPOSAL\_CHOSEN message, you need to determine which proposal was incompatible: the proposal sent during IKE phase 1 for the IKE SA or the proposal sent during IKE phase 2 for the IPSec SA.

A quick way to determine which phase failed is to enter:

```
ProCurve# show crypto ike sa
```

If the CLI shows an IKE SA for the connection, you know that it at least completed IKE phase 1.

You can also scroll through the debug messages looking for signs of the IKE phase that generated the problems. (See Table 10-25 above.) Look for messages that repeat several times—for example, “sending main mode message 1”; they indicate that the router cannot complete the step. Table 10-26 shows other messages associated with problems in a particular IKE phase.

**Table 10-26. Debug Messages**

Messages Associated with IKE Phase 1 Problems	Messages Associated with IKE Phase 2 Problems
<b>IKEDeletelsakmpSA</b>	<b>IKFindIPSecSAbySPI</b>
<b>IANA for protocol: Isakmp</b>	<b>IANA for protocol: IPSec</b>

Once you have determined which IKE phase is causing your problem, you should move to “Comparing VPN Policies” on page 10-80. This section will help you determine which specific policy is causing IKE to fail.

**Peer ID is Invalid.** Continuously repeating “IKStartNegotiation” messages indicate that the router is unable to even reach the peer to begin IKE negotiations. This problem can have several sources:

- The peer ID in crypto map entry is incorrect.
- The peer ID in IKE policy is incorrect.
- The IKE policy does not allow you to initiate IKE with this peer.

See Table 10-27 for debug messages associated these problems.

**Table 10-27. IKStartNegotiation Debug Messages**

Attribute	Problem	Best Next Step
<b>Can not initiate on a Respond only policy</b>	The IKE policy for the peer is set to <b>no initiate</b> .	Change the IKE initiate mode in the policy to <b>main</b> or <b>aggressive</b> .
<b>Could not find an IKE policy to use</b>	The peer ID in the crypto map entry does not match the peer ID in any IKE policy.	Check the peer ID in the crypto map entry and IKE policy and change the incorrect setting.
<b>Already in process of negotiation IKRetryTimeOut: Retrying 1st phase</b>	The peer ID in the crypto map entry and IKE policy are incorrect.	Verify that you have configured the correct public IP address for the peer.

To check the peer ID in an IKE policy or crypto map entry, enter commands such as the following:

**Syntax:** show crypto map [*<mapname>* *<mapindex>*]

**Syntax:** show crypto ike policy

You can also view all crypto maps by entering the **show crypto map** command without a mapname and index.

To change the initiate mode for IKE, move to the IKE policy configuration mode context and enter:

**Syntax:** initiate [main | aggressive]

**Invalid Authentication Information.** If IKE sends or receives **main mode message 5** again and again, it is unable to authenticate the peer. Check the preshared key for the peer in the running-configuration:

```
ProCurve# show running-config
```

If you are using digital certificates, you should verify that your certificate is up to date and valid. You might also need to change your CRL. See “Managing Certificates” on page 10-61 for more information on viewing and deleting digital certificates.

## Comparing VPN Policies

Depending on where you discovered IKE negotiations breaking down, you should check configurations for:

- IKE policies (IKE phase 1)
- transform sets (IKE phase 2)
- crypto maps (IKE phase 2)

**Comparing IKE Policies.** All security parameters should match the peer's. If possible, have your peer attempt to initiate a VPN connection with the local router. You can then find the settings proposed by the peer in the debug messages.

When viewing debug messages, first determine whether the proposals are those of the local or the remote peer. Figure 10-15 shows sample debug messages that display when the local router initiates IKE with the peer. If the peer had initiated IKE, the first debug message would have read:

```
Received first message of main mode
```



Scroll through the debug messages until you see the message for the relevant IKE phase: “IANA: for proposal ISAKMP” (phase 1). (See Figure 10-15.)

An Isakmp proposal is the proposal for the IKE SA. In the debug messages, look underneath the proposal message for the TRANSFORM ATTRIBUTES. These are the security proposals. Each proposal includes six attributes, marked “SA Attrib.” The actual setting for the attribute is shown below as the “Value.”

```
2005.08.13 14:20:49 1: Sent out first message of main mode
2005.08.13 14:20:49 <POLICY: 1> PAYLOADS: SA,PROP,TRANS,VID,VID,VID
2005.08.13 14:20:49 SA PAYLOAD
2005.08.13 14:20:49 DOI: 1
2005.08.13 14:20:49 Situation: 1
2005.08.13 14:20:49 PROPOSAL PAYLOAD
2005.08.13 14:20:49 Proposal No.: 1
2005.08.13 14:20:49 IANA No. for protocol: ISAKMP (1)
2005.08.13 14:20:49 Size of the variable SPI field: 0
2005.08.13 14:20:49 Number of transforms offered: 1
2005.08.13 14:20:49 TRANSFORM PAYLOAD
2005.08.13 14:20:49 Transform Number: 1
2005.08.13 14:20:49 IANA Transform ID: IKE Key (1)
2005.08.13 14:20:49 TRANSFORM ATTRIBUTES
2005.08.13 14:20:49 SA Attrib: Group Description (4)
2005.08.13 14:20:49 Length: 2
2005.08.13 14:20:49 Value: DH Group 1 (1)
2005.08.13 14:20:49 SA Attrib: Authentication Method (3)
2005.08.13 14:20:49 Length: 2
2005.08.13 14:20:49 Value: Pre-shared Key (1)
2005.08.13 14:20:49 SA Attrib: Encryption Algorithm (1)
2005.08.13 14:20:49 Length: 2
2005.08.13 Value: Seconds (1)
2005.08.13 14:20:49 SA Attrib: Life Time (12)
2005.08.13 14:20:49 Length: 4
2005.08.13 14:20:49 Value: (28800)
```

“Sent” indicates that these are the local router’s policies.

**Figure 10-15. IKE Debug Messages: IKE Phase 1 Security Parameters**

You can compare the peer’s settings to yours in two ways:

- Initiate a connection with the peer and view the debug messages with the local proposals
- View the IKE attribute policy used with the peer by entering:  
ProCurve# show crypto ike policy

Table 10-28 shows where in the local router’s running-config you can find the settings that should match the policies you see proposed by the peer.

**Table 10-28. TRANSFORM ATTRIBUTES (IKE SA Security Proposals)**

SA Attribute	Value Options	Remote Setting	Router Configuration	Options	Local Setting
Group Description	<ul style="list-style-type: none"> <li>DH Group 1</li> <li>DH Group 2</li> </ul>		IKE attribute policy: <b>group</b>	<ul style="list-style-type: none"> <li>1</li> <li>2</li> </ul>	
Authentication Method	<ul style="list-style-type: none"> <li>pre-shared key</li> <li>DSS</li> <li>RSA</li> </ul>		IKE attribute policy: <b>authentication</b>	<ul style="list-style-type: none"> <li>pre-share</li> <li>dss-sig</li> <li>rsa-sig</li> </ul>	
Encryption Algorithm	<ul style="list-style-type: none"> <li>DES</li> <li>3DES</li> <li>128</li> <li>192</li> <li>256</li> </ul>		IKE attribute policy: <b>encryption</b>	<ul style="list-style-type: none"> <li>des</li> <li>3des</li> <li>aes-128-cbc</li> <li>aes-192-cbc</li> <li>aes-256-cbc</li> </ul>	
Authentication Algorithm	<ul style="list-style-type: none"> <li>SHA1</li> <li>MD5</li> </ul>		IKE attribute policy: <b>hash</b>	<ul style="list-style-type: none"> <li>sha</li> <li>md5</li> </ul>	
Life Type	<ul style="list-style-type: none"> <li>seconds</li> <li>kilobytes</li> </ul>		only supports seconds	—	
Life Time	<ul style="list-style-type: none"> <li>number of seconds</li> </ul>		IKE attribute policy: <b>lifetime</b>	<b>60-86,400</b>	

Reconfigure any settings that do not match.

**Comparing IPSec Policies.** You can track IKE messages to verify that IKE has entered phase 2. You should see such messages as:

- sending main mode message 5
- received main mode message 5
- sending main mode message 6
- received main mode message 6
- sending aggressive mode message 3
- received aggressive mode message 3
- sending quick mode message 1
- received quick mode message 1

When IKE cannot progress past quickmode message 1, it is unable to negotiate the IPSec SA. If possible, have your peer attempt to initiate a connection with you. In this way you can search through the debug messages for the peer's IPSec SA proposal and determine which settings do not match local settings.

```
2005.08.13 14:25:03 peer 10.1.1.1: Received first message of quick mode
2005.08.13 14:25:03 <POLICY: 1> PAYLOADS:
2005.08.13 14:25:03 HASH, SA, PROP, TRANS, NONCE, ID, ID
2005.08.13 14:25:03 HASH PAYLOAD
2005.08.13 14:25:03 SA PAYLOAD
2005.08.13 14:25:03 DOI: 1
2005.08.13 14:25:03 Situation: 1
2005.08.13 14:25:03 PROPOSAL PAYLOAD
2005.08.13 14:25:03 Proposal No.: 1
2005.08.13 14:25:03 IANA No. for protocol: IPSec ESP (3)
2005.08.13 14:25:03 Size of the variable SPI field: 4
2005.08.13 14:25:03 Number of transforms offered: 1
2005.08.13 14:25:03 SPI for the proposal: 2866043823
2005.08.13 14:25:03 TRANSFORM PAYLOAD
2005.08.13 14:25:03 Transform Number: 1
2005.08.13 14:25:03 IANA Transform ID: DES (2)
2005.08.13 14:25:03 TRANSFORM ATTRIBUTES
2005.08.13 14:25:03 SA Attrib: Authentication Algorithm (5)
2005.08.13 14:25:03 Length: 2
2005.08.13 14:25:03 Value: MD5 (1)
2005.08.13 14:25:03 SA Attrib: Encapsulation Mode (4)
2005.08.13 14:25:03 Length: 2
2005.08.13 14:25:03 Value: Tunnel (1)
2005.08.13 14:25:03 SA Attrib: Life Type (1)
2005.08.13 14:25:03 Length: 2
2005.08.13 14:25:03 Value: Seconds (1)
2005.08.13 14:25:03 SA Attrib: Life Time (2)
2005.08.13 14:25:03 Length: 4
2005.08.13 14:25:03 Value: (28800)
```

“Received” indicates that these are the local peer’s policies.

Encryption algorithm

IPSec is using ESP headers

Figure 10-16. IKE Debug Messages: IKE Phase 2 Security Proposals

Figure 10-16 illustrates how you can find the security parameters proposed by the peer.

Search for the “IANA No for proposal: IPSec” message. An IPSec proposal is the proposal for the IPSec SA. Beneath it should be an “IANA Transform ID” and “TRANSFORM ATTRIBUTES.” The IANA Transform ID is the encryption algorithm for ESP. The transform attributes are the other IPSec SA security proposals. Each proposal includes four attributes, marked “SA Attrib.” The actual setting for the attribute is shown below as the “Value.”

Table 10-29 and Table 10-30 show where in the local router's running-config you can find the settings that should match the IPSec security policies proposed by the peer.

**Table 10-29. IANA Transform ID**

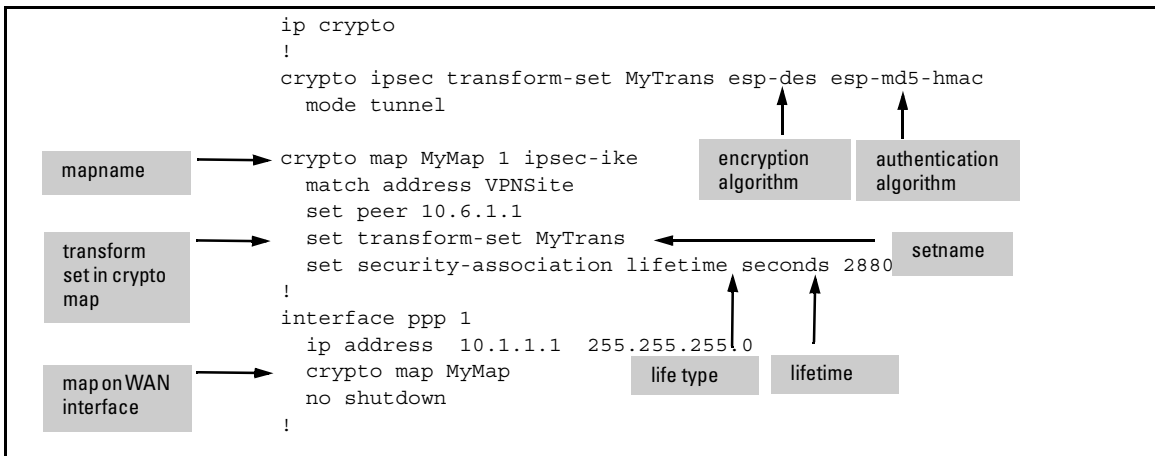
Message	Value Options	Remote Setting	Setting in Running-Config	Options	Local Setting
IANA Transform ID	<ul style="list-style-type: none"> <li>• 256</li> <li>• 192</li> <li>• 3DES</li> <li>• 128</li> <li>• DES</li> <li>• NULL</li> </ul>		Find the transform set: <pre>crypto map &lt;mapname&gt; &lt;mapindex&gt; set transform-set &lt;setname&gt;</pre> View the setting: <pre>crypto ipsec transform-set &lt;setname&gt;</pre>	<ul style="list-style-type: none"> <li>• esp-aes-256-cbc</li> <li>• esp-aes-192-cbc</li> <li>• esp-3des</li> <li>• esp-aes-128-cbc</li> <li>• esp-des</li> <li>• esp-null</li> </ul>	

**Table 10-30. IPSec SA TRANSFORM ATTRIBUTES**

SA Attribute	Value Options	Remote Setting	Setting in the Running-Config	Options	Local Setting
Authentication Algorithm	<ul style="list-style-type: none"> <li>• SHA1</li> <li>• MD5</li> </ul>		Find the transform set: <pre>crypto map &lt;mapname&gt; &lt;mapindex&gt; set transform-set &lt;setname&gt;</pre> View the setting: <pre>crypto ipsec transform-set &lt;setname&gt;</pre>	<ul style="list-style-type: none"> <li>• ah-sha-hmac</li> <li>• ah-md5-hmac</li> <li>• esp-sha-hmac</li> <li>• esp-md5-hmac</li> </ul>	
Encapsulation Mode	<ul style="list-style-type: none"> <li>• transport</li> <li>• tunnel</li> </ul>		<pre>crypto ipsec transform-set &lt;setname&gt; mode</pre>	tunnel	

SA Attribute	Value Options	Remote Setting	Setting in the Running-Config	Options	Local Setting
Life Type	<ul style="list-style-type: none"> <li>seconds</li> <li>kilobytes</li> </ul>		<b>crypto map</b> <b>&lt;mapname&gt;</b> <b>&lt;mapindex&gt;</b> <b>set security-</b> <b>association</b> <b>lifetime</b>	<ul style="list-style-type: none"> <li>kilobytes</li> <li>seconds</li> </ul>	
Life Time	<ul style="list-style-type: none"> <li>Number of seconds</li> <li>Number of kilobytes</li> </ul>		<b>crypto map</b> <b>&lt;mapname&gt;</b> <b>&lt;mapindex&gt;:</b> <ul style="list-style-type: none"> <li><b>set security-</b> <b>association</b> <b>lifetime</b> <b>seconds</b> <b>&lt;seconds&gt;</b></li> <li><b>set security-</b> <b>association</b> <b>lifetime</b> <b>kilobytes</b> <b>&lt;kilobytes&gt;</b></li> </ul>	<ul style="list-style-type: none"> <li>120 to 86,400</li> <li>2560 to 536,870,912</li> </ul>	

Figure 10-17 shows a sample VPN configuration and where you would look for the local settings.



**Figure 10-17. Sample VPN Settings in Running-Config**

You can compare the peer's settings to yours in two ways:

- Initiate a connection with the peer and view the debug messages with the local proposals
- View the VPN configurations on the local router for the connection

To view the configuration on the local router, you can view the running-config as shown above in 10-17. You can also zero in on VPN configurations only by following these steps:

1. View crypto maps:

```
ProCurve# show crypto map
```

Find the map that is set to the peer you are attempting to reach. Pay particular attention to which transform sets have been assigned to the crypto map entry used for the connection. You can also compare the PFS group and IPSec SA lifetime settings with those proposed by the peer.

2. View the transform set:

**Syntax:** show crypto ipsec transform-set <setname>

For example:

```
ProCurve# show crypto ipsec transform-set T1
```

Compare the algorithms in the transform sets used by the crypto map entry with those proposed by the peer.

If necessary, reconfigure any mismatched IPSec settings by creating a new transform set or modifying settings in the crypto map entry.

### Returning VPN Policies to Their Defaults

It is best to resolve incompatible VPN policies by contacting the remote site and agreeing upon the settings for both the IKE and IPSec SA. However, if the connection must go up immediately and you have no way to contact the remote site, you can return the IKE policy to its defaults in hopes that the peer is using these default settings.

To return VPN policies to their defaults:

1. Move to the configuration mode context for the IKE policy used to establish the connection.
2. Create a new attribute policy:  

```
ProCurve(config-crypto-ike)# attribute 90
```
3. Exit to the global configuration mode context.

4. Return the crypto map settings to the defaults:

```
ProCurve(config-crypto map)# no set pfs
ProCurve(config-crypto map)# no security-association lifetime
```

Try to ping the remote location from the local network. If the connection goes up, you know that you had a problem with the security policies. You should contact the IT staff at the remote site and agree upon which settings to use to enforce your organization's security policies.

If the connection still does not go up, then the peer may not be using default settings. You must contact the peer and bring your security settings into agreement.

You can also try changing the IKE respond mode to **anymode** and the initiate mode to the mode not currently used. Move to the IKE policy for the peer and enter:

```
ProCurve(config-ike)# respond anymode
ProCurve(config-ike)# initiate [aggressive | main]
```

The peer may also be using different algorithms to secure the IPSec SA. The Secure Router OS does not set any default algorithms for the permanent VPN connection. You can try the settings automatically established when you configure a VPN using the VPN wizard in the Web browser interface, which are:

- ESP 3DES for the encryption key
- ESP MD5 for an authentication key
- no PFS group
- 28,800 second SA lifetime

Enter this command to configure the transform set:

```
ProCurve(config)# crypto ipsec transform-set <setname> esp-3des esp-md5-hmac
```

You might also have a problem with your addressing. Verify the peer's public address, which should be set in the crypto map, IKE policy, and, if you are using main mode, the remote ID list. Also, if the peer has a dynamic address, you cannot initiate the VPN connection. The peer must initiate a connection with the local router.

## Quick Start

This section provides the commands you must enter to quickly configure:

- a site-to-site VPN
- a client-to-site VPN
- digital certificates

Only a minimal explanation is provided. If you need additional information about any of these options, see “Contents” on page 10-1 to locate the section and page number that contains the explanation you need.

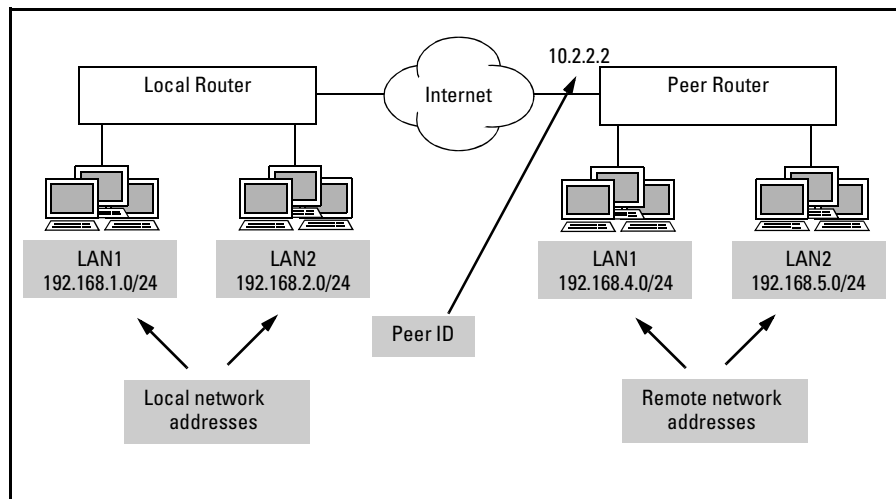
**Table 10-31. Quick Start Settings for a Site-to-Site VPN**

Parameters	Options	Obtain Setting From	Your Setting
peer ID	IP address (A.B.C.D)	remote router's public IP address	
peer's remote ID	<ul style="list-style-type: none"><li>• IP address (A.B.C.D)</li><li>• fully-qualified domain name (FQDN)</li><li>• email address</li><li>• abstract syntax notation distinguished name (ASN-DN), for digital certificates only</li></ul>	remote router	
preshared key (if using)	alphanumeric string	match peer	
IKE policy number	1 to 10,000	—	
initiate mode	<ul style="list-style-type: none"><li>• main</li><li>• aggressive</li><li>• none</li></ul>	peer's respond mode—however, at least one side must be able to initiate	
attribute policy number for IKE SA proposals	1 to 65,535	—	
IKE authentication method	<ul style="list-style-type: none"><li>• preshared keys</li><li>• DSS digital certificate</li><li>• RSA digital certificates</li></ul>	match peer	
IKE SA authentication algorithm	<ul style="list-style-type: none"><li>• MD5</li><li>• SHA-1</li></ul>	match peer	



Parameters	Options	Obtain Setting From	Your Setting
IKE SA encryption algorithm	<ul style="list-style-type: none"> <li>• DES</li> <li>• 3DES</li> <li>• AES 128-bit</li> <li>• AES 192-bit</li> <li>• AES 256-bit</li> </ul>	match peer	
IKE SA lifetime	60 to 86,400 seconds	match peer	
IPSec SA proposals	<ul style="list-style-type: none"> <li>• AH</li> <li>• ESP</li> <li>• AH and ESP</li> </ul>	match peer	
transform setname	alphanumeric string	—	
AH authentication algorithm	<ul style="list-style-type: none"> <li>• MD5</li> <li>• SHA-1</li> </ul>	match peer	
ESP encryption algorithm	<ul style="list-style-type: none"> <li>• DES</li> <li>• 3DES</li> <li>• AES 128-bit</li> <li>• AES 192-bit</li> <li>• AES 256-bit</li> <li>• None (null)</li> </ul>	match peer	
ESP authentication algorithm (optional, unless you select ESP null)	<ul style="list-style-type: none"> <li>• MD5</li> <li>• SHA-1</li> </ul>	match peer	
IPSec SA lifetime type	<ul style="list-style-type: none"> <li>• kilobytes</li> <li>• seconds</li> </ul>	match peer	
IPSec SA lifetime in kilobytes (optional)	2560 to 536,870,912 kilobytes	match peer	
IPSec SA lifetime in seconds (optional)	120 to 86,400 seconds	match peer	
ACL listname	alphanumeric string	—	
local VPN network(s)	—	local network address(es) and subnet mask(s)	
remote VPN network(s)	—	remote network address(es) and subnet mask(s)	
hosts in the VPN networks denied access to the tunnel (optional)	host address (<A.B.C.D>) range of host addresses (<A.B.C.D> <wildcard bits>)	organizational policy	

Parameters	Options	Obtain Setting From	Your Setting
crypto mapname	alphanumeric string	same name for every entry establishing a connection on the same interface	
crypto map index number	0 to 65,535	different index number for every entry establishing a connection to a different site	



**Figure 10-18. Sample VPN Configuration**

## Configuring a Site-to-Site VPN

You can print Table 10-31 and fill it out with the settings for your VPN. You can then use the worksheet to complete the quick start commands. If the local router will connect to more than one site, print and fill out a worksheet for each site.

Figure 10-18 displays a simplified sample network, which you can consult as you configure your router.

Complete these steps to configure a site-to-site VPN using IPSec with IKE:

1. Install the IPSec VPN module.
2. Enable VPN functions:

```
ProCurve(config)# ip crypto
```

3. Create an IKE policy:  
**Syntax:** `crypto ike policy <IKE policynumber>`
4. Configure the initiate mode:  
**Syntax:** `[no] initiate [main | aggressive]`  
For example:  
`ProCurve(config-crypto-ike)# initiate aggressive`  
If the peer has a dynamic address, set the mode to **no initiate**.
5. Set the peer ID or peer IDs:  
**Syntax:** `peer [any | <peer A.B.C.D>]`
6. Create an attribute policy:  
**Syntax:** `attribute <attribute policynumber>`
7. Enter settings for the IKE SA, including authentication method, authentication algorithm, encryption algorithm, Diffie-Hellman group, and IKE SA lifetime:  
**Syntax:** `authentication [dss-sig | pre-share | rsa-sig]`  
**Syntax:** `hash [md5 | sha]`  
**Syntax:** `encryption [3des | aes-128-cbc | aes-192-cbc | aes-256-cbc | des]`  
**Syntax:** `group [1 | 2]`  
**Syntax:** `lifetime <seconds>`
8. If so desired, repeat steps 7 and 8 to configure multiple attribute policies. IKE proposes the policy with the lowest number first.
9. If so desired, repeat steps 4 through 9 to configure multiple IKE policies. You can use the same policy with more than one peer, but you should usually use a different policy to connect to a remote site from the policy used to connect to mobile users. (The router cannot initiate IKE with mobile users.)

10. Exit to the global configuration mode and configure algorithms for the IPsec SA in a transform set:
  - AH protocol:  
**Syntax:** crypto ipsec transform-set <setname> [ah-md5-hmac | ah-sha-hmac]
  - ESP protocol:  
**Syntax:** crypto ipsec transform-set <setname> [esp-des | esp-3des | esp-aes-128-cbc | esp-aes-192-cbc | esp-aes-256-cbc | esp-null] [esp-md5-hmac | esp-sha-hmac]
  - AH and ESP protocol:  
**Syntax:** crypto ipsec transform-set <setname> [ah-md5-hmac | ah-sha-hmac] [esp-des | esp-3des | esp-aes-128-cbc | esp-aes-192-cbc | esp-aes-256-cbc | esp-null] [esp-md5-hmac | esp-sha-hmac]
11. Set the mode to tunnel:  
ProCurve(cfg-crypto-trans)# mode tunnel
12. If so desired, repeat steps 11 and 12 to configure another transform set.
13. Specify the traffic allowed over the tunnel in an ACL:
  - a. Create an extended ACL:  
**Syntax:** ip access-list extended <listname>
  - b. Add deny statements for hosts not allowed to access the tunnel.  
**Syntax:** deny ip [any | host <source A.B.C.D> | hostname <source hostname> | <source A.B.C.D> <wildcard bits>] [any | host <destination A.B.C.D> | hostname <destination hostname> | <destination A.B.C.D> <wildcard bits>]  
  
For example:  
ProCurve(config-ext-nacl)# deny ip host 192.168.10.112 any
  - c. Add permit statements from the local VPN networks to the remote VPN networks:  
**Syntax:** permit ip [any | host <source A.B.C.D> | hostname <source hostname> | <source A.B.C.D> <wildcard bits>] [any | host <destination A.B.C.D> | hostname <destination hostname> | <destination A.B.C.D> <wildcard bits>]  
  
You use wildcard bits, which operate on reverse logic from subnet masks, to specify the range of addresses. For example, to select a network with the subnet mask 255.255.255.0, enter:  
ProCurve(config-ext-nacl)# permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255
14. Configure a crypto map entry:  
ProCurve(config)# crypto map <mapname> <map index> ipsec-ike

15. Specify one peer only for the crypto map entry:  
**Syntax:** set peer <peer A.B.C.D>
16. You can associate the crypto map entry with the IKE policy configured for the remote peer.  
**Syntax:** ike-policy <policy number>
17. Assign up to six transform sets to the crypto map entry:  
**Syntax:** set transform-set <setname1> [<setname2>] [<setname3>] [<setname4>] [<setname5>] [<setname6>]
18. Apply the ACL to the crypto map entry:  
**Syntax:** match address <ACL listname>
19. Set IPsec SA lifetime (unless accepting default). You can configure it in kilobytes, seconds, or both:  
**Syntax:** set security-association lifetime [kilobytes <kilobytes> | seconds <seconds>]
20. If the router is connecting to more than one remote site, repeat steps 14 through 18 for each site. Use the same mapname for each entry, but a different map index number. You can also configure a crypto map entry to connect to mobile users. (See “Configuring a Client-to-Site VPN” on page 10-94).
21. Exit to the global configuration mode context. Configure a remote ID list that contains authentication information for remote peers. If you are using preshared keys for authentication, associate the preshared key with the peer. You can optionally associate a peer with the IKE policy and crypto map entry that should be used with that peer.

For the remote ID, you can specify:

- IP address:

**Syntax:** crypto ike remote-id address <peer A.B.C.D> [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]

- fully-qualified domain name (FQDN):

**Syntax:** crypto ike remote-id fqdn <peer FQDN> [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]

- email address:

**Syntax:** crypto ike remote-id user-fqdn <peer email address> [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]

- distinguished name (with digital certificates only):

**Syntax:** crypto ike remote-id asn1-dn <distinguished name> [ike-policy <policy number>] [crypto map <mapname> <map sequence>]

You can use the \* wildcard character to configure a remote ID that matches multiple remote peers.

22. Apply the crypto map to the WAN interface that connects to the Internet. Move to the logical interface configuration mode context and enter:

**Syntax:** crypto map <mapname>

For example:

```
ProCurve(config)# int ppp 1
ProCurve(config-ppp 1)# crypto map VPN
```

The local and remote gateways must also somehow exchange routing information. You can use BGP to communicate routes to your ISP, which then tunnels them to the remote router. (See *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR.*) You can also tunnel a multicast routing protocol such as RIP or OSPF through the Internet yourself using a GRE tunnel. See *Chapter 11: Configuring a Tunnel with Generic Routing Encapsulation.*

## Configuring a Client-to-Site VPN

You can print Table 10-32 and fill it out with the settings for your VPN. You can then use the worksheet to complete the quick start commands.

**Table 10-32. Quick Start Settings for a Client-to-Site VPN**

Parameters	Options	Obtain Setting From	Your Setting
peer ID	any	—	any
peer's remote ID	<ul style="list-style-type: none"> <li>• IP address (A.B.C.D)</li> <li>• fully-qualified domain name (FQDN)</li> <li>• email address</li> <li>• abstract syntax notation distinguished name (ASN-DN), for digital certificates only</li> <li>• any</li> </ul>	mobile users—You should either use <b>any</b> or wildcards to match multiple users. If you are using digital certificates, the remote ID should match the corresponding field in authorized certificates.	
preshared key (if using)	alphanumeric string	match peer	
IKE mode config poolname	alphanumeric string	—	
range of private addresses for IKE mode config to assign to mobile users	first A.B.C.D last A.B.C.D	organizational policy	
DNS server(s) for IKE mode config (optional)	A.B.C.D	organizational policy	
WINS (NetBIOS) server(s) for IKE mode config (optional)	A.B.C.D	organizational policy	
IKE policy number	1 to 10,000	highest on the router	
initiate mode	none	—	no initiate
attribute policy number for IKE SA proposals	1 to 65,535	—	
IKE authentication method	<ul style="list-style-type: none"> <li>• preshared keys</li> <li>• DSS digital certificate</li> <li>• RSA digital certificates</li> </ul>	match peer	
IKE SA authentication algorithm	<ul style="list-style-type: none"> <li>• MD5</li> <li>• SHA-1</li> </ul>	match peer	
IKE SA encryption algorithm	<ul style="list-style-type: none"> <li>• DES</li> <li>• 3DES</li> <li>• AES 128-bit</li> <li>• AES 192-bit</li> <li>• AES 256-bit</li> </ul>	match peer	
IKE SA lifetime	60 to 86,400 seconds	match peer	

**Virtual Private Networks**  
Quick Start

Parameters	Options	Obtain Setting From	Your Setting
IPSec SA proposals	<ul style="list-style-type: none"> <li>• AH</li> <li>• ESP</li> <li>• AH and ESP</li> </ul>	match peer	
transform setname	alphanumeric string	—	
AH authentication algorithm	<ul style="list-style-type: none"> <li>• MD5</li> <li>• SHA-1</li> </ul>	match peer	
ESP encryption algorithm	<ul style="list-style-type: none"> <li>• DES</li> <li>• 3DES</li> <li>• AES 128-bit</li> <li>• AES 192-bit</li> <li>• AES 256-bit</li> <li>• None (<b>null</b>)</li> </ul>	match peer	
ESP authentication algorithm (optional, unless you select ESP null)	<ul style="list-style-type: none"> <li>• MD5</li> <li>• SHA-1</li> </ul>	match peer	
IPSec SA lifetime type	<ul style="list-style-type: none"> <li>• kilobytes</li> <li>• seconds</li> </ul>	match peer	
IPSec SA lifetime in kilobytes (optional)	2560 to 536,870,912 kilobytes	match peer	
IPSec SA lifetime in seconds (optional)	120 to 86,400 seconds	match peer	
ACL listname	alphanumeric string	—	
local VPN network(s)	range of host addresses (< <b>A.B.C.D</b> > < <b>wildcard bits</b> >)	local network address(es) and subnet mask(s)	
network in the IKE mode config pool	first A.B.C.D last A.B.C.D	network address(es) and subnet mask(s)	
hosts in the VPN networks denied access to the tunnel (optional)	<ul style="list-style-type: none"> <li>• host address (&lt;<b>A.B.C.D</b>&gt;)</li> <li>• range of host addresses (&lt;<b>A.B.C.D</b>&gt; &lt;<b>wildcard bits</b>&gt;)</li> </ul>	organizational policy	
crypto mapname	alphanumeric string	same name for every entry establishing a connection on the same interface	
crypto map index number	0 to 65,535	different index number for every entry establishing a connection to a different site	



1. Install the IPsec VPN module.
2. Enable VPN functions:  
`ProCurve(config)# ip crypto`
3. Configure an IKE mode config pool:  
**Syntax:** `crypto ike client configuration pool <poolname>`
4. Specify the range of private network addresses in the pool:  
**Syntax:** `ip-range <first A.B.C.D> <last A.B.C.D>`
5. You can also specify server addresses for clients in the pool:  
**Syntax:** `dns-server <A.B.C.D> [<A.B.C.D>]`  
**Syntax:** `netbios-name-server <A.B.C.D> [<A.B.C.D>]`
6. Create an IKE policy:  
**Syntax:** `crypto ike policy <IKE policynumber>`  
For example:  
`ProCurve(config)# crypto ike policy 10`
7. Prevent the router from initiating IKE:  
`ProCurve(config-crypto-ike)# no initiate`
8. Set the peer ID:  
`ProCurve(config-crypto-ike)# peer any`
9. Apply the IKE client pool to the IKE policy:  
**Syntax:** `client configuration pool <poolname>`
10. Create an attribute policy:  
**Syntax:** `attribute <attribute policynumber>`
11. Enter settings for the IKE SA, including authentication method, authentication algorithm, encryption algorithm, Diffie-Hellman group, and IKE SA lifetime:  
**Syntax:** `authentication [dss-sig | pre-share | rsa-sig]`  
**Syntax:** `hash [md5 | sha]`  
**Syntax:** `encryption [3des | aes-128-cbc | aes-192-cbc | aes-256-cbc | des]`  
**Syntax:** `group [1 | 2]`  
**Syntax:** `lifetime <seconds>`
12. If so desired, repeat steps 11 and 12 to configure multiple attribute policies. The router uses the policy with the lowest number first.

13. If so desired, configure another IKE policy to connect to a remote site. (See “Configuring a Site-to-Site VPN” on page 10-90.)
  14. Exit to the global configuration mode and configure algorithms for the IPsec SA in a transform set:
    - AH protocol:  
**Syntax:** crypto ipsec transform-set <setname> [ah-md5-hmac | ah-sha-hmac]
    - ESP protocol:  
**Syntax:** crypto ipsec transform-set <setname> [esp-des | esp-3des | esp-aes-128-cbc | esp-aes-192-cbc | esp-aes-256-cbc | esp-null] [esp-md5-hmac | esp-sha-hmac]
    - AH and ESP protocol:  
**Syntax:** crypto ipsec transform-set <setname> [ah-md5-hmac | ah-sha-hmac] [esp-des | esp-3des | esp-aes-128-cbc | esp-aes-192-cbc | esp-aes-256-cbc | esp-null] [esp-md5-hmac | esp-sha-hmac]
  15. Set the mode to tunnel:  
ProCurve(cfg-crypto-trans)# mode tunnel
  16. If so desired, repeat steps 15 and 16 to configure another transform set.
  17. Specify the traffic allowed over the tunnel in an ACL:
    - a. Create an extended ACL:  
**Syntax:** ip access-list extended <listname>
    - b. Add deny statements for hosts not allowed to access the tunnel:  
**Syntax:** deny ip [any | host <source A.B.C.D> | hostname <source hostname> | <source A.B.C.D> <wildcard bits>] [any | host <destination A.B.C.D> | hostname <destination hostname> | <destination A.B.C.D> <wildcard bits>]
- For example:
- ```
ProCurve(config-ext-nacl)# deny ip host 192.168.10.112 any
```

- c. Add permit statements from the local VPN networks to the network addresses in the IKE mode config pool:

**Syntax:** permit ip [any | host <source A.B.C.D> | | hostname <source hostname> | <source A.B.C.D> <wildcard bits>] [any | host <destination A.B.C.D> | hostname <destination hostname> | <destination A.B.C.D> <wildcard bits>]

You use wildcard bits, which operate on reverse logic from subnet masks, to specify the range of addresses. The destination network address is the network that contains the addresses specified for the IKE mode config pool. For example:

```
ProCurve(config-ext-nacl)# permit ip 192.168.10.0 0.0.0.255 192.168.100.0 0.0.0.255
```

18. Configure a crypto map entry:

**Syntax:** crypto map <mapname> <map index> ipsec-ike

19. You can associate the crypto map entry with the IKE policy configured for the remote peer.

**Syntax:** ike-policy <policy number>

20. Assign up to six transform sets to the crypto map entry:

**Syntax:** set transform-set <setname1> [<setname2>] [<setname3>] [<setname4>] [<setname5>] [<setname6>]

21. Apply the ACL to the crypto map entry:

**Syntax:** match address <ACL listname>

22. Set the IPSec SA lifetime (unless accepting default). You can configure it in kilobytes, seconds, or both:

**Syntax:** set security-association lifetime [kilobytes <kilobytes> | seconds <seconds>]

23. If the router is also connecting to remote sites, configure a map entry for each site. (See “Configuring a Site-to-Site VPN” on page 10-90.) Use the same mapname for each entry, but a different map index number.

24. Exit to the global configuration mode context. Configure a remote ID list that contains authentication information for remote peers. If you are using preshared keys for authentication, associate the preshared key with the peer. You can optionally associate a peer with the IKE policy and crypto map entry that should be used with that peer.

For the remote ID, you can specify:

- **any** (often used for multiple mobile users):  
**Syntax:** `crypto ike remote-id any [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]`
- IP address:  
**Syntax:** `crypto ike remote-id address <peer A.B.C.D> [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]`
- fully-qualified domain name (FQDN):  
**Syntax:** `crypto ike remote-id fqdn <peer FQDN> [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]`
- email address:  
**Syntax:** `crypto remote-id user-fqdn <peer email address> [preshared-key <preshared key>] [ike-policy <policy number>] [crypto map <mapname> <map sequence>]`
- distinguished name (with digital certificates only):  
**Syntax:** `crypto ike remote-id asn1-dn <distinguished name> [ike-policy <policy number>] [crypto map <mapname> <map sequence>]`

Use the wildcard character (\*) to make the remote ID entry apply to multiple mobile users. This allows you to use the same IKE policy to respond to all mobile users.

25. Apply the crypto map to the WAN interface that connects to the Internet. Move to the logical interface configuration mode context and enter:

**Syntax:** `crypto map <mapname>`

For example:

```
ProCurve(config)# int ppp 1
ProCurve(config-ppp 1)# crypto map VPN
```

## Obtaining Digital Certificates

If you have selected a digital certificate standard for the IKE authentication method, you must obtain a certificate for the router. These instructions give the steps for obtaining a certificate automatically using SCEP. See configuration instructions in “Using Digital Certificates (Optional)” on page 10-54 to learn how to obtain certificates manually.

Complete the following steps to obtain digital certificates:

1. Select a CA server.
2. Configure a profile for the CA:  
**Syntax:** `crypto ca profile <profile name>`
3. Select automatic enrollment:  
**Syntax:** `enrollment url http://<CA server's FQDN>/<filename>`
4. Exit to global configuration mode and download the CA certificate:  
**Syntax:** `crypto ca authenticate <profile name>`
5. Accept the certificate by pressing **y**.
6. Generate a self certificate request:  
**Syntax:** `crypto ca enroll <profile name>`
7. Fill in the local router's information as prompted in the dialog box.

**Virtual Private Networks**  
Quick Start