Configuring a VPN Using Easy VPN and an IPSec Tunnel

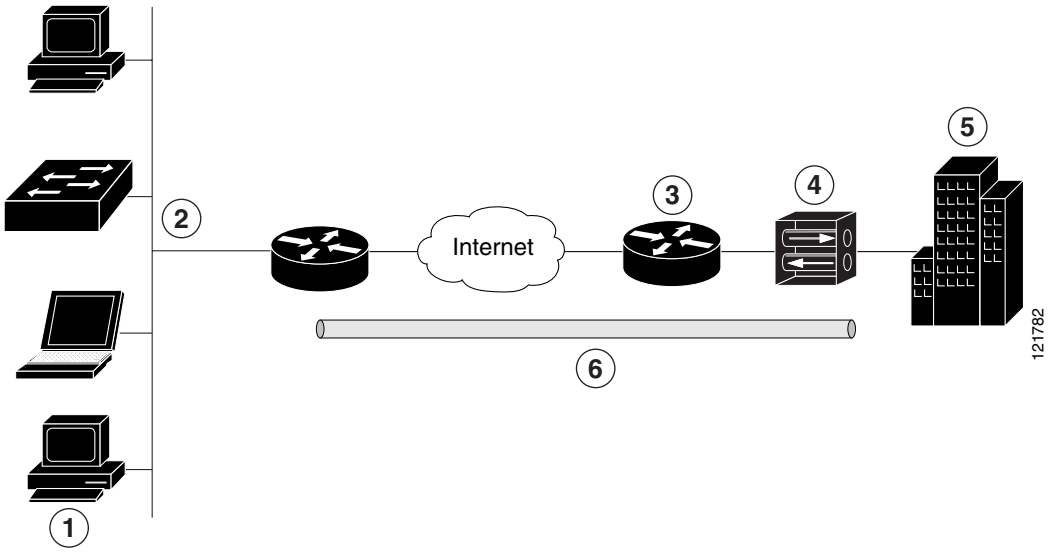
The Cisco 1800 series integrated services fixed-configuration routers support the creation of Virtual Private Networks (VPNs).

Cisco routers and other broadband devices provide high-performance connections to the Internet, but many applications also require the security of VPN connections which perform a high level of authentication and which encrypt the data between two particular endpoints.

Two types of VPNs are supported—site-to-site and remote access. Site-to-site VPNs are used to connect branch offices to corporate offices, for example. Remote access VPNs are used by remote clients to log in to a corporate network.

The example in this chapter illustrates the configuration of a remote access VPN that uses the Cisco Easy VPN and an IPSec tunnel to configure and secure the connection between the remote client and the corporate network. Figure 6-1 shows a typical deployment scenario.

Figure 6-1 Remote Access VPN Using IPSec Tunnel



1	Remote, networked users
2	VPN client—Cisco 1800 series integrated services router
3	Router—Providing the corporate office network access
4	VPN server—Easy VPN server; for example, a Cisco VPN 3000 concentrator with outside interface address 192.168.101.1
5	Corporate office with a network address of 10.1.1.1
6	IPSec tunnel

Cisco Easy VPN

The Cisco Easy VPN client feature eliminates much of the tedious configuration work by implementing the Cisco Unity Client protocol. This protocol allows most VPN parameters, such as internal IP addresses, internal subnet masks, DHCP server addresses, WINS server addresses, and split-tunneling flags, to be defined at a VPN server, such as a Cisco VPN 3000 series concentrator that is acting as an IPSec server.

An Easy VPN server–enabled device can terminate VPN tunnels initiated by mobile and remote workers who are running Cisco Easy VPN Remote software on PCs. Easy VPN server–enabled devices allow remote routers to act as Easy VPN Remote nodes.

The Cisco Easy VPN client feature can be configured in one of two modes—client mode or network extension mode. Client mode is the default configuration and allows only devices at the client site to access resources at the central site. Resources at the client site are unavailable to the central site. Network extension mode allows users at the central site (where the VPN 3000 series concentrator is located) to access network resources on the client site.

After the IPSec server has been configured, a VPN connection can be created with minimal configuration on an IPSec client, such as a supported Cisco 1800 integrated services router. When the IPSec client initiates the VPN tunnel connection, the IPSec server pushes the IPSec policies to the IPSec client and creates the corresponding VPN tunnel connection.

Note The Cisco Easy VPN client feature supports configuration of only one destination peer. If your application requires creation of multiple VPN tunnels, you must manually configure the IPSec VPN and Network Address Translation/Peer Address Translation (NAT/PAT) parameters on both the client and the server.

Configuration Tasks

Perform the following tasks to configure your router for this network scenario:

- Configure the IKE Policy
- Configure Group Policy Information
- Apply Mode Configuration to the Crypto Map
- Enable Policy Lookup
- Configure IPSec Transforms and Protocols
- Configure the IPSec Crypto Method and Parameters
- Apply the Crypto Map to the Physical Interface
- Create an Easy VPN Remote Configuration

An example showing the results of these configuration tasks is shown in the section "Configuration Example."

Note The procedures in this chapter assume that you have already configured basic router features as well as PPPoE or PPPoA with NAT, DCHP and VLANs. If you have not performed these configurations tasks, see Chapter 1, "Basic Router Configuration," Chapter 3, "Configuring PPP over Ethernet with NAT," Chapter 4, "Configuring PPP over ATM with NAT," and Chapter 5, "Configuring a LAN with DHCP and VLANs" as appropriate for your router.

Configure the IKE Policy

Perform these steps to configure the Internet Key Exchange (IKE) policy, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**crypto isakmp policy** *priority*
	
	Example:
	`Router(config)# crypto isakmp policy 1`
	`Router(config-isakmp)#`
		Creates an IKE policy that is used during IKE negotiation. The priority is a number from 1 to 10000, with 1 being the highest.
		
		Also enters the Internet Security Association Key and Management Protocol (ISAKMP) policy configuration mode.

Step 2	**encryption** {**des** | **3des** | **aes** | **aes 192** | **aes 256**}
	
	Example:
	`Router(config-isakmp)# encryption 3des`
	`Router(config-isakmp)#`
		Specifies the encryption algorithm used in the IKE policy.
		
		The example specifies 168-bit data encryption standard (DES).

Step 3	**hash** {**md5** | **sha**}
	
	Example:
	`Router(config-isakmp)# hash md5`
	`Router(config-isakmp)#`
		Specifies the hash algorithm used in the IKE policy.
		
		The example specifies the Message Digest 5 (MD5) algorithm. The default is Secure Hash standard (SHA-1).

Step 4	**authentication** {**rsa-sig** | **rsa-encr** | **pre-share**}
	
	Example:
	`Router(config-isakmp)# authentication pre-share`
	`Router(config-isakmp)#`
		Specifies the authentication method used in the IKE policy.
		
		The example specifies a pre-shared key.

Step 5	**group** {**1** | **2** | **5**}
	
	Example:
	`Router(config-isakmp)# group 2`
	`Router(config-isakmp)#`
		Specifies the Diffie-Hellman group to be used in an IKE policy.

	Command or Action	Purpose

Step 6	**lifetime** *seconds*
	
	Example:
	`Router(config-isakmp)# lifetime 480`
	`Router(config-isakmp)#`
		Specifies the lifetime, 60–86400 seconds, for an IKE security association (SA).

Step 7	**exit**
	
	Example:
	`Router(config-isakmp)# exit`
	`Router(config)#`
		Exits IKE policy configuration mode, and enters global configuration mode.

Configure Group Policy Information

Perform these steps to configure the group policy, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**crypto isakmp client configuration group** {**group-name** | **default**}
	
	Example:
	`Router(config)# crypto isakmp client configuration group rtr-remote`
	`Router(config-isakmp-group)#`
		Creates an IKE policy group containing attributes to be downloaded to the remote client.
		Also enters the Internet Security Association Key and Management Protocol (ISAKMP) group policy configuration mode.

Step 2	**key** *name*
	
	Example:
	`Router(config-isakmp-group)# key secret-password`
	`Router(config-isakmp-group)#`
		Specifies the IKE pre-shared key for the group policy.

Step 3	**dns** *primary-server*
	
	Example:
	`Router(config-isakmp-group)# dns 10.50.10.1`
	`Router(config-isakmp-group)#`
		Specifies the primary Domain Name System (DNS) server for the group.
		Note You may also want to specify Windows Internet Naming Service (WINS) servers for the group by using the **wins** command.

Step 4	**domain** *name*
	
	Example:
	`Router(config-isakmp-group)# domain company.com`
	`Router(config-isakmp-group)#`
		Specifies group domain membership.

	Command or Action	Purpose

Step 5	**exit**
	
	Example:
	`Router(config-isakmp-group)# exit`
	`Router(config)#`
		Exits IKE group policy configuration mode, and enters global configuration mode.

Step 6	**ip local pool** {**default** | *poolname*} [*low-ip-address* [*high-ip-address*]]
	
	Example:
	`Router(config)# ip local pool dynpool 30.30.30.20 30.30.30.30`
	`Router(config)#`
		Specifies a local address pool for the group.
		For details about this command and additional parameters that can be set, see the *Cisco IOS Dial Technologies Command Reference*.

Apply Mode Configuration to the Crypto Map

Perform these steps to apply mode configuration to the crypto map, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**crypto map** *map-name* **isakmp authorization list** *list-name*
	
	Example:
	`Router(config)# crypto map dynmap isakmp authorization list rtr-remote`
	`Router(config)#`
		Applies mode configuration to the crypto map and enables key lookup (IKE queries) for the group policy from an authentication, authorization, and accounting (AAA) server.

Step 2	**crypto map** *tag* **client configuration address** [**initiate** | **respond**]
	
	Example:
	`Router(config)# crypto map dynmap client configuration address respond`
	`Router(config)#`
		Configures the router to reply to mode configuration requests from remote clients.

Enable Policy Lookup

Perform these steps to enable policy lookup through AAA, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**aaa new-model**
	
	Example:
	`Router(config)# aaa new-model`
	`Router(config)#`
		Enables the AAA access control model.

Step 2	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
	
	Example:
	`Router(config)# aaa authentication login rtr-remote local`
	`Router(config)#`
		Specifies AAA authentication of selected users at login, and specifies the method used.
		
		This example uses a local authentication database. You could also use a RADIUS server for this. For details, see the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference*.

Step 3	**aaa authorization** {**network** | **exec** | **commands** *level* | **reverse-access** | **configuration**} {**default** | *list-name*} [*method1* [*method2...*]]
	
	Example:
	`Router(config)# aaa authorization network rtr-remote local`
	`Router(config)#`
		Specifies AAA authorization of all network-related service requests, including PPP, and specifies the method of authorization.
		
		This example uses a local authorization database. You could also use a RADIUS server for this. For details, see the *Cisco IOS Security Configuration Guide* and *Cisco IOS Security Command Reference*.

Step 4	**username** *name* {**nopassword** | **password** *password* | **password** *encryption-type encrypted-password*}
	
	Example:
	`Router(config)# username Cisco password 0 Cisco`
	`Router(config)#`
		Establishes a username-based authentication system.
		
		This example implements a username of *Cisco* with an encrypted password of *Cisco*.

Configure IPSec Transforms and Protocols

A transform set represents a certain combination of security protocols and algorithms. During IKE negotiation, the peers agree to use a particular transform set for protecting data flow.

During IKE negotiations, the peers search in multiple transform sets for a transform that is the same at both peers. When such a transform set is found, it is selected and applied to the protected traffic as a part of both peers' configurations.

Perform these steps to specify the IPSec transform set and protocols, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**crypto ipsec transform-set** *transform-set-name transform1* [*transform2*] [*transform3*] [*transform4*]
	
	Example:
	`Router(config)# crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac`
	`Router(config)#`
		Defines a transform set—an acceptable combination of IPSec security protocols and algorithms.
		
		See the *Cisco IOS Security Command Reference* for detail about the valid transforms and combinations.

Step 2	**crypto ipsec security-association lifetime** {**seconds** *seconds* | **kilobytes** *kilobytes*}
	
	Example:
	`Router(config)# crypto ipsec security-association lifetime seconds 86400`
	`Router(config)#`
		Specifies global lifetime values used when IPSec security associations are negotiated.
		
		See the *Cisco IOS Security Command Reference* for details.

Note With manually established security associations, there is no negotiation with the peer, and both sides must specify the same transform set.

Configure the IPSec Crypto Method and Parameters

A dynamic crypto map policy processes negotiation requests for new security associations from remote IPSec peers, even if the router does not know all the crypto map parameters (for example, IP address).

Perform these steps to configure the IPSec crypto method, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**crypto dynamic-map** *dynamic-map-name dynamic-seq-num*
	
	Example:
	`Router(config)# crypto dynamic-map dynmap 1`
	`Router(config-crypto-map)#`
		Creates a dynamic crypto map entry and enters crypto map configuration mode.
		
		See the *Cisco IOS Security Command Reference* for more detail about this command.

Step 2	**set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*]
	
	Example:
	`Router(config-crypto-map)# set transform-set vpn1`
	`Router(config-crypto-map)#`
		Specifies which transform sets can be used with the crypto map entry.

	Command or Action	Purpose

Step 3	**reverse-route**
	
	Example:
	`Router(config-crypto-map)# reverse-route`
	`Router(config-crypto-map)#`
		Creates source proxy information for the crypto map entry.
		
		See the *Cisco IOS Security Command Reference* for details.

Step 4	**exit**
	
	Example:
	`Router(config-crypto-map)# exit`
	`Router(config)#`
		Returns to global configuration mode.

Step 5	**crypto map** *map-name seq-num* [**ipsec-isakmp**] [**dynamic** *dynamic-map-name*] [**discover**] [**profile** *profile-name*]
	
	Example:
	`Router(config)# crypto map static-map 1 ipsec-isakmp dynamic dynmap`
	`Router(config)#`
		Creates a crypto map profile.

Apply the Crypto Map to the Physical Interface

The crypto maps must be applied to each interface through which IP Security (IPSec) traffic flows. Applying the crypto map to the physical interface instructs the router to evaluate all the traffic against the security associations database. With the default configurations, the router provides secure connectivity by encrypting the traffic sent between remote sites. However, the public interface still allows the rest of the traffic to pass and provides connectivity to the Internet.

Perform these steps to apply a crypto map to an interface, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**interface** *type number*
	
	Example:
	`Router(config)# interface fastethernet 0`
	`Router(config-if)#`
		Enters the interface configuration mode for the interface to which you want the crypto map applied.

	Command or Action	Purpose

Step 2	**crypto map** *map-name*
	
	Example:
	`Router(config-if)# crypto map static-map`
	`Router(config-if)#`
		Applies the crypto map to the interface.
		
		See the *Cisco IOS Security Command Reference* for more detail about this command.

Step 3	**exit**
	
	Example:
	`Router(config-crypto-map)# exit`
	`Router(config)#`
		Returns to global configuration mode.

Create an Easy VPN Remote Configuration

The router acting as the IPSec remote router must create an Easy VPN remote configuration and assign it to the outgoing interface.

Perform these steps to create the remote configuration, beginning in global configuration mode:

	Command or Action	Purpose

Step 1	**crypto ipsec client ezvpn** *name*
	
	Example:
	`Router(config)# crypto ipsec client ezvpn ezvpnclient`
	`Router(config-crypto-ezvpn)#`
		Creates a Cisco Easy VPN remote configuration, and enters Cisco Easy VPN remote configuration mode.

Step 2	**group** *group-name* **key** *group-key*
	
	Example:
	`Router(config-crypto-ezvpn)# group ezvpnclient key secret-password`
	`Router(config-crypto-ezvpn)#`
		Specifies the IPSec group and IPSec key value for the VPN connection.

Step 3	**peer** {*ipaddress* | *hostname*}
	
	Example:
	`Router(config-crypto-ezvpn)# peer 192.168.100.1`
	`Router(config-crypto-ezvpn)#`
		Specifies the peer IP address or hostname for the VPN connection.
		
		Note A hostname can be specified only when the router has a DNS server available for hostname resolution.

Step 4	**mode** {**client** | **network-extension** | **network extension plus**}
	
	Example:
	`Router(config-crypto-ezvpn)# mode client`
	`Router(config-crypto-ezvpn)#`
		Specifies the VPN mode of operation.

	Command or Action	Purpose

Step 5	**exit**
	
	Example:
	`Router(config-crypto-ezvpn)# exit`
	`Router(config)#`
		Returns to global configuration mode.

Step 6	**interface** *type number*
	
	Example:
	`Router(config)# interface fastethernet 0`
	`Router(config-if)#`
		Enters interface configuration mode.
		
		Note For routers with an ATM WAN interface, this command would be **interface atm 0**.

Step 7	**crypto ipsec client ezvpn** *name* [**outside** | **inside**]
	
	Example:
	`Router(config-if)# crypto ipsec client ezvpn ezvpnclient outside`
	`Router(config-if)#`
		Assigns the Cisco Easy VPN remote configuration to the WAN interface, causing the router to automatically create the NAT or PAT and access list configuration needed for the VPN connection.

Step 8	**exit**
	
	Example:
	`Router(config-crypto-ezvpn)# exit`
	`Router(config)#`
		Returns to global configuration mode.

Verifying Your Easy VPN Configuration

```
Router# show crypto ipsec client ezvpn
Tunnel name :ezvpnclient
Inside interface list:vlan 1
Outside interface:fastethernet 0
Current State:IPSEC_ACTIVE
Last Event:SOCKET_UP
Address:8.0.0.5
Mask:255.255.255.255
Default Domain:cisco.com
```

Configuration Example

The following configuration example shows a portion of the configuration file for the VPN and IPSec tunnel described in this chapter.

```
!
aaa new-model
!
aaa authentication login rtr-remote local
aaa authorization network rtr-remote local
aaa session-id common
!
username Cisco password 0 Cisco
```

```
!
crypto isakmp policy 1
    encryption 3des
    authentication pre-share
    group 2
    lifetime 480
!
crypto isakmp client configuration group rtr-remote
    key secret-password
    dns 10.50.10.1 10.60.10.1
    domain company.com
    pool dynpool
!
crypto ipsec transform-set vpn1 esp-3des esp-sha-hmac
!
crypto ipsec security-association lifetime seconds 86400
!
crypto dynamic-map dynmap 1
    set transform-set vpn1
    reverse-route
!
crypto map static-map 1 ipsec-isakmp dynamic dynmap
crypto map dynmap isakmp authorization list rtr-remote
crypto map dynmap client configuration address respond

crypto ipsec client ezvpn ezvpnclient
    connect auto
    group 2 key secret-password
    mode client
    peer 192.168.100.1
!

interface fastethernet 0
    crypto ipsec client ezvpn ezvpnclient outside
    crypto map static-map
!
interface vlan 1
    crypto ipsec client ezvpn ezvpnclient inside
!
```