Configuring the VPN Acceleration Module

This chapter contains the information and procedures needed to configure the VPN Acceleration Module (VAM) in the Cisco 7100 series routers, Cisco 7200 series routers, and Cisco 7401ASR routers. This chapter contains the following sections:

Overview

The VAM provides encryption services for any interface in Cisco 7100 series, Cisco 7200 series routers, and Cisco 7401ASR routers. If you have previously configured IPSec on the router and you install the VAM, the VAM automatically performs encryption services.

Note There are no interfaces to configure on the VAM.

This section only contains basic configuration information for enabling encryption and IPSec tunneling services. Refer to the "IP Security and Encryption" part of the *Security Configuration Guide* and the appropriate IOS Release version of the *Security Command Reference* guide for detailed configuration information on IPSec, IKE, and CA.

Configuration Tasks

On power up if the enabled LED is on, the VAM is fully functional and does not require any configuration commands. However, for the VAM to provide encryption services, you must complete the steps in the following sections:

- Using the EXEC Command Interpreter, page 4-2 (required)
- Configuring IKE, page 4-2 (required)
- Configuring IPSec, page 4-6 (required)

Note You can configure a static crypto map, create a dynamic crypto map, or add a dynamic crypto map into a static crypto map. Refer to the online publication, *Configuring the VPN Acceleration Module*.

Optionally, you can configure Certification Authority (CA) interoperability (refer to the "Configuring Certification Authority Interoperability" chapter in the *Security Configuration Guide*).

Using the EXEC Command Interpreter

You modify the configuration of your router through the software command interpreter called the *EXEC* (also called enable mode). You must enter the privileged level of the EXEC command interpreter with the **enable** command before you can use the **configure** command to configure a new interface or change the existing configuration of an interface. The system prompts you for a password if one has been set.

The system prompt for the privileged level ends with a pound sign (#) instead of an angle bracket (>). At the console terminal, use the following procedure to enter the privileged level:

Step 1 At the user-level EXEC prompt, enter the **enable** command. The EXEC prompts you for a privileged-level password as follows:

```
Router> enable

Password:
```

Step 2 Enter the password (the password is case sensitive). For security purposes, the password is not displayed. When you enter the correct password, the system displays the privileged-level system prompt (#):

```
Router#
```

This completes the procedure for entering the privileged level of the EXEC command interpreter.

Configuring IKE

To configure IKE, you would perform the following tasks:

If you do not specify a value for a parameter, the default value is assigned. For information on default values, refer to the "IP Security and Encryption" chapter of the *Security Command Reference* publication.

To configure a policy, use the following commands, starting in global configuration mode:

	Command	Purpose
Step 1	**crypto isakmp policy** *priority*	Identifies the policy to create, and enters config-isakmp command mode.
Step 2	**encryption {des | 3des}**	Specifies the encryption algorithm.
Step 3	**group** {**1** | **2**}	Specifies the Diffie-Hellman group identifier.

For detailed information on creating IKE policies, refer to the "Configuring Internet Key Exchange Security Protocol" chapter in the *Security Configuration Guide* publication.

Specifying an RSA Authentication Method

When you configure IKE policies, you specify one of the following RSA authentication methods:

- *RSA Signatures Method*—Using this method, you configure peers to obtain certificates from a certification authority (CA). (Note: You first configure the CA to issue certificates, as described in the "Configuring Certification Authority Interoperability" chapter of the *Cisco IOS Security Configuration Guide* for your Cisco IOS Release.)

  Peers use certificates to exchange public keys securely; each peer has the remote peer's public signature key. When both peers have valid certificates, they automatically exchange public keys as part of any IKE negotiation in which RSA signatures are used.

  Or, you can exchange public keys manually (see "Manually Configuring RSA Keys" section on page 4-3).

- *RSA Encrypted Nonces Method*—Using this method, each peer has the public keys of the other peers. Unlike RSA signatures, the RSA encrypted nonces method does not use certificates to exchange public keys. Instead, you use one of the following methods:

  - Manually configure RSA keys (see "Manually Configuring RSA Keys" section on page 4-3).
  - Ensure that an IKE exchange, using RSA signatures with certificates, has already occurred between the peers. (The peer public keys are exchanged during the RSA-signatures-based IKE negotiations, if certificates are used.)

    You specify two policies: a higher-priority policy with RSA encrypted nonces, and a lower-priority policy with RSA signatures. RSA signatures are used the first time because peers do not yet have each other's public keys. Future IKE negotiations use RSA encrypted nonces because public keys have been exchanged.

    Note This alternative requires that you have CA support configured.

Note The RSA encrypted nonces feature on VAM cards became enabled beginning with Cisco IOS Release 12.3(10).

- *Preshared Keys Authentication Method*—Using this method, you configure preshared keys as described in the "Configuring Preshared Keys" section on page 4-6.

  If RSA encryption is configured and signature mode is negotiated (and certificates are used for signature mode), the peer requests both signature and encryption keys. The router requests as many keys as the configuration will support. If RSA encryption is not configured, it will just request a signature key.

Manually Configuring RSA Keys

When you specify the RSA encrypted nonces authentication method, and you are not using a certification authority, you manually configure the RSA keys by performing the following tasks at each IPSec peer that uses RSA encrypted nonces in an IKE policy:

- Generating RSA Keys, page 4-4
- Setting ISAKMP Identity, page 4-4
- Specifying Other Peers RSA Public Keys, page 4-5

Generating RSA Keys

To generate RSA keys, use the following commands starting in global configuration mode:

	Command	Purpose
Step 1	`Router (config)# crypto key generate rsa [usage-keys]`	Generates RSA keys.
Step 2	`Router# show crypto key mypubkey rsa`	Displays the generated RSA public key (in EXEC mode).

Repeat the above tasks at each peer (without CA support) that uses RSA encrypted nonces in an IKE policy.

Setting ISAKMP Identity

This topic describes how to set the ISAKMP identity for each peer that uses pre-shared keys in an IKE policy.

When two peers use IKE to establish IPSec security associations, each peer sends its identity to the remote peer. Each peer sends either its host name or its IP address, depending on how you have the router's ISAKMP identity set.

By default, a peer's ISAKMP identity is the peer's IP address. If appropriate, you could change the identity to be the peer's host name instead. As a general rule, set all peers' identities the same way—either all peers should use their IP address, or all peers should use their host name. If some peers use their host name and some peers use their IP address to identify themselves to each other, IKE negotiations could fail if a remote peer's identity is not recognized and a DNS lookup is unable to resolve the identity.

To set a peer's ISAKMP identity, use the following commands in global configuration mode:

	Command	Purpose
Step 1	`Router (config)# crypto isakmp identity {address | hostname}`	**At the local peer**: Specifies the peer's ISAKMP identity by IP address or by host name.[1]
Step 2	`Router (config)# ip host hostname address1 [address2...address8]`	**At all remote peers**: If the local peer's ISAKMP identity was specified using a host name, maps the peer's host name to its IP address(es) at all the remote peers. (This step might be unnecessary if the host name/address is already mapped in a DNS server.)

1. See the **crypto isakmp identity** command description for guidelines for when to use the IP address vs. the host name.

Repeat these tasks for each peer that uses pre-shared keys in an IKE policy.

Specifying Other Peers RSA Public Keys

At each peer, specify the other peers RSA public keys by using the following commands starting in global configuration mode:

	Command	Purpose
Step 1	`Router (config)# crypto key pubkey-chain rsa`	Enters public key chain configuration mode.

Step 2	`Router (config-pubkey-c)# named-key key-name [encryption | signature]`
	or
	`Router (config-pubkey-c)# addressed-key key-address [encryption | signature]`
		Indicates which remote peer's RSA public key you are going to specify. Enters public key configuration mode.
		If the remote peer uses its host name as its ISAKMP identity, use the **named-key** command and specify the remote peer's fully qualified domain name (such as somerouter.example.com) as the key-name.
		If the remote peer uses its IP address as its ISAKMP identity, use the **addressed-key** command and specify the remote peer's IP address as the key-address.

Step 3	`Router (config-pubkey-k)# address ip-address`	If you used a fully qualified domain name to name the remote peer in Step 2 (using the **named-key** command), you can optionally specify the remote peer's IP address.
Step 4	`Router (config-pubkey-k)# key-string key-string`	Specifies the remote peer RSA public key. This is the key viewed by the remote peer's administrator previously when he generated his router RSA keys.
Step 5	`Router (config-pubkey-k)# quit`	Returns to public key chain configuraiton mode.
Step 6	—	Repeat Steps 2 through 4 to specify the RSA public keys of all the other IPSec peers that use RSA encrypted nonces in an IKE policy.
Step 7	`Router (config-pubkey-c)# exit`	Returns to global configuration mode.

Repeat the above tasks at each peer that uses RSA encrypted nonces in an IKE policy.

To view RSA public keys while or after you configure them, use the following command in EXEC mode:

	Command	Purpose
Step 1	`Router# show crypto key pubkey-chain rsa {name key-name | address key-address}`	Displays a list of all the RSA public keys stored on your router, or displays details of a particular RSA public key stored on your router.

Configuring Preshared Keys

To configure preshared keys, perform these tasks at each peer that uses preshared keys in an IKE policy:

- First, set the ISAKMP identity of each peer. Each peer's identity should be set to either its host name or by its IP address. By default, a peer's identity is set to its IP address. Setting ISAKMP identities is described in the section "Setting ISAKMP Identity."
- Next, specify the shared keys at each peer. Note that a given preshared key is shared between two peers. At a given peer you could specify the same key to share with multiple remote peers; however, a more secure approach is to specify different keys to share between different pairs of peers.

To specify preshared keys at a peer, use the following commands in global configuration mode:

	Command	Purpose

Step 1	`Router(config)# crypto isakmp key` *keystring* `address peer-address`
	
	or
	
	`Router(config)# crypto isakmp key` *keystring* `hostname peer-hostname`
		At the local peer: Specifies the shared key to be used with a particular remote peer.
		
		If the remote peer specified its ISAKMP identity with an address, use the **address** keyword in this step; otherwise use the **hostname** keyword in this step.

Step 2	`Router(config)# crypto isakmp key` *keystring* `address peer-address`
	
	or
	
	`Router(config)# crypto isakmp key` *keystring* `hostname peer-hostname`
		At the remote peer: Specifies the shared key to be used with the local peer. This is the same key you just specified at the local peer.
		
		If the local peer specified its ISAKMP identity with an address, use the **address** keyword in this step; otherwise use the **hostname** keyword in this step.

Step 3	–	Repeat Steps 1 and 2 for each remote peer.

Remember to repeat these tasks at each peer that uses preshared keys in an IKE policy.

Configuring IPSec

After you have completed IKE configuration, configure IPSec at each participating IPSec peer. This section contains basic steps to configure IPSec and includes the tasks discussed in the following sections:

- Creating Crypto Access Lists, page 4-6 (required)
- Defining Transform Sets, page 4-7 (required)
- Verifying the Configuration, page 4-9 (optional)

For detailed information on configuring IPSec, refer to the “Configuring IPSec Network Security” chapter in the *Security Configuration Guide* publication.

Creating Crypto Access Lists

Crypto access lists define which IP traffic will be protected by encryption.

Note IKE uses UDP port 500. The IPSec Encapsulation Security Protocol (ESP) and Authentication Header (AH) protocols use protocol numbers 50 and 51. Ensure that your interface access lists are configured so that protocol numbers 50, 51, and UDP port 500 traffic are not blocked at interfaces used by IPSec. In some cases you might need to add a statement to your access lists to explicitly permit this traffic.

To create crypto access lists, use the following commands in global configuration mode:

Step	Command	Purpose
Step 1	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**log**] or **ip access-list extended** *name*	Specifies conditions to determine which IP packets are protected.[1] (Enable or disable encryption for traffic that matches these conditions.) We recommend that you configure "mirror image" crypto access lists for use by IPSec and that you avoid using the **any** keyword.
Step 2	Add **permit** and **deny** statements as appropriate.	Adds permit or deny statements to access lists.
Step 3	**end**	Exits the configuration command mode.

1.You specify conditions using an IP access list designated by either a number or a name. The **access-list** command designates a numbered extended access list; the **ip access-list extended** command designates a named access list.

For detailed information on configuring access lists, refer to the "Configuring IPSec Network Security" chapter in the *Security Configuration Guide* publication.

Defining Transform Sets

A transform set is a combination of security protocols and algorithms. During the IPSec security association negotiation, peers agree to use a specific transform set to protect a particular data flow.

To define a transform set, use the following commands, starting in global configuration mode:

	Command	Purpose
Step 1	**crypto ipsec transform-set** *transform-set-name transform1* [*transform2* [*transform3*]]	Defines a transform set and enter crypto transform configuration mode. **Note** Complex rules define which entries you can use for the transform arguments. These rules are explained in the command description for the **crypto ipsec transform-set** command, and Table 4-1 provides a list of allowed transform combinations.
Step 2	**mode** [**tunnel** | **transport**]	Changes the mode associated with the transform set. The mode setting is applicable only to traffic whose source and destination addresses are the IPSec peer addresses; it is ignored for all other traffic. (All other traffic is in tunnel mode only.)

	Command	Purpose
Step 3	**end**	Exits the crypto transform configuration mode to enabled mode.
Step 4	**clear crypto sa** or **clear crypto sa peer** {*ip-address* | *peer-name*} or **clear crypto sa map** *map-name* or **clear crypto sa spi** *destination-address protocol spi*	Clears existing IPSec security associations so that any changes to a transform set take effect on subsequently established security associations (SAs). (Manually established SAs are reestablished immediately.) Using the **clear crypto sa** command without parameters clears out the full SA database, which clears out active security sessions. You may also specify the **peer**, **map**, or **entry** keywords to clear out only a subset of the SA database.

:Table 4-1 shows allowed transform combinations.

Table 4-1 Allowed Transform Combinations

AH Transform[1]		ESP Encryption Transform[1]		ESP Authentication Transform[2]	
Transform	**Description**	**Transform**	**Description**	**Transform**	**Description**
ah-md5-hmac	AH with MD5 (HMAC variant) authentication algorithm	**esp-3des**	ESP with 168-bit Triple DES encryption algorithm	**esp-md5-hmac**	ESP with MD5 (HMAC variant) authentication algorithm
ah-sha-hmac	AH with SHA (HMAC variant) authentication algorithm	**esp-des**	ESP with 56-bit DES encryption algorithm	**esp-sha-hmac**	ESP with SHA (HMAC variant) authentication algorithm
		esp-null	ESP transform without cipher		

1. Pick one transform option.
2. Pick one transform option, but only if you selected esp-null or ESP encrypting transform.

To create crypto map entries that use IKE to establish the security associations, use the following commands, starting in global configuration mode:

	Command	Purpose
Step 1	**crypto map** *map-name seq-num* **ipsec-isakmp**	Creates the crypto map and enters crypto map configuration mode.
Step 2	**match address** *access-list-id*	Specifies an extended access list. This access list determines which traffic is protected by IPSec and which is not.

	Command	Purpose
Step 3	**set peer** {*hostname* | *ip-address*}	Specifies a remote IPSec peer. This is the peer to which IPSec-protected traffic can be forwarded. Repeat for multiple remote peers.
Step 4	**set transform-set** *transform-set-name1 [transform-set-name2...transform-set-name6]*	Specifies which transform sets are allowed for this crypto map entry. Lists multiple transform sets in order of priority (highest priority first).
Step 5	end	Exits crypto map configuration mode.
Step 6	Repeat these steps to create additional crypto map entries as required.	

Verifying the Configuration

Some configuration changes take effect only after subsequent security associations are negotiated. For the new settings to take effect immediately, clear the existing security associations.

To clear (and reinitialize) IP Sec security associations, use one of the commands in Table 4-2 in global configuration mode:

Table 4-2 Commands to Clear IP Sec Security Associations

Command	Purpose
clear crypto sa or **clear crypto sa peer** {*ip-address* | *peer-name*} or **clear crypto sa map** *map-name* or **clear crypto sa spi** *destination-address protocol spi*	Clear IPSec security associations (SAs). Using the **clear crypto sa** command without parameters clears out the full SA database, which clears out active security sessions. You may also specify the **peer**, **map**, or **spi** keywords to clear out only a subset of the SA database.

The following steps provide information on verifying your configurations:

Step 1 Enter **show crypto ipsec transform-set** to view your transform set configuration:

```
Router# show crypto ipsec transform-set
Transform set combined-des-md5: {esp-des esp-md5-hmac}
   will negotiate = {Tunnel,},
Transform set t1: {esp-des esp-md5-hmac}
   will negotiate = {Tunnel,},
Transform set t100: {ah-sha-hmac}
   will negotiate = {Transport,},
Transform set t2: {ah-sha-hmac}
   will negotiate = {Tunnel,},
   {esp-des}
   will negotiate = {Tunnel,},
```

Step 2 Enter **show crypto map** [**interface** *interface* **| tag** *map-name]* to view your crypto map configuration:

```
Router# show crypto map
Crypto Map: "router-alice" idb: Ethernet0 local address: 172.21.114.123
Crypto Map "router-alice" 10 ipsec-isakmp
        Peer = 172.21.114.67
        Extended IP access list 141
```

```
            access-list 141 permit ip
                source: addr = 172.21.114.123/0.0.0.0
                dest:   addr = 172.21.114.67/0.0.0.0
          Current peer: 172.21.114.67
          Security-association lifetime: 4608000 kilobytes/120 seconds
          PFS (Y/N): N
          Transform sets={t1,}
```

Step 3 Enter **show crypto ipsec sa** [**map** map-name | **address** | **identity** | **detail** | **interface**] to view information about IPSec security associations.

```
Router# show crypto ipsec sa
interface: Ethernet0
    Crypto map tag: router-alice, local addr. 172.21.114.123
   local ident (addr/mask/prot/port): (172.21.114.123/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (172.21.114.67/255.255.255.255/0/0)
    current_peer: 172.21.114.67
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest 10
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify 10
    #send errors 10, #recv errors 0
     local crypto endpt.: 172.21.114.123, remote crypto endpt.: 172.21.114.67
     path mtu 1500, media mtu 1500
     current outbound spi: 20890A6F
     inbound esp sas:
      spi: 0x257A1039(628756537)
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 26, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y
     inbound ah sas:
     outbound esp sas:
      spi: 0x20890A6F(545852015)
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 27, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y
     outbound ah sas:
interface: Tunnel0
   Crypto map tag: router-alice, local addr. 172.21.114.123
   local ident (addr/mask/prot/port): (172.21.114.123/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (172.21.114.67/255.255.255.255/0/0)
   current_peer: 172.21.114.67
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 10, #pkts encrypt: 10, #pkts digest 10
    #pkts decaps: 10, #pkts decrypt: 10, #pkts verify 10
    #send errors 10, #recv errors 0
     local crypto endpt.: 172.21.114.123, remote crypto endpt.: 172.21.114.67
     path mtu 1500, media mtu 1500
     current outbound spi: 20890A6F
     inbound esp sas:
      spi: 0x257A1039(628756537)
        transform: esp-des esp-md5-hmac,
        in use settings ={Tunnel,}
        slot: 0, conn id: 26, crypto map: router-alice
        sa timing: remaining key lifetime (k/sec): (4607999/90)
        IV size: 8 bytes
        replay detection support: Y
     inbound ah sas:
     outbound esp sas:
```

```
     spi: 0x20890A6F(545852015)
       transform: esp-des esp-md5-hmac,
       in use settings ={Tunnel,}
       slot: 0, conn id: 27, crypto map: router-alice
       sa timing: remaining key lifetime (k/sec): (4607999/90)
       IV size: 8 bytes
       replay detection support: Y
   outbound ah sas:
```

For a detailed description of the information displayed by the **show** commands, refer to the "IP Security and Encryption" chapter of the *Security Command Reference* publication.

Configuration Examples

This section provides the following configuration examples:

- Configuring IKE Policies Example, page 4-11
- Configuring IPSec Configuration Example, page 4-11

Configuring IKE Policies Example

In the following example, three IKE policies are created, with policy 15 as the highest priority, policy 20 as the next priority, and the existing default priority as the lowest priority. It also creates a preshared key to be used with policy 20 with the remote peer whose IP address is 192.168.224.33, and a RSA encryption key with policy 30.

```
crypto isakmp policy 15
 encryption 3des
 hash md5
 authentication rsa-sig
 group 2
 lifetime 5000
crypto isakmp policy 20
 authentication pre-share
 lifetime 10000
crypto isakmp key 1234567890 address 192.168.224.33
crypto isakmp policy 30
 encr 3des
 authentication rsa-encr
crypto key pubkey-chain rsa
 addressed-key 11.0.0.2
  address 11.0.0.2
  key-string
   305C300D 06092A86 4886F70D 01010105 00034B00 30480241 009E227B F7F489E2
   E980D39F 4A981644 C8A103F4 3CB1EFB1 CE8EDCC5 8E7BFDFC 6C4BCB3D 62BE76F3
   5E5F7F43 F0841163 D234138C 09725BA6 B30F50C5 63615E0B 45020301 0001
  quit
```

Configuring IPSec Configuration Example

The following example shows a minimal IPSec configuration where the security associations will be established via IKE:

An IPSec access list defines which traffic to protect:

```
access-list 101 permit ip 10.0.0.0 0.0.0.255 10.2.2.0 0.0.0.255
```

A transform set defines how the traffic will be protected. In this example, transform set "myset1" uses DES encryption and SHA for data packet authentication:

```
crypto ipsec transform-set myset1 esp-des esp-sha
```

Another transform set example is "myset2," which uses Triple DES encryptions and MD5 (HMAC variant) for data packet authentication:

```
crypto ipsec transform-set myset2 esp-3des esp-md5-hmac
```

A crypto map joins together the IPSec access list and transform set and specifies where the protected traffic is sent (the remote IPSec peer):

```
crypto map toRemoteSite 10 ipsec-isakmp
 match address 101
 set transform-set myset2
 set peer 10.2.2.5
```

The crypto map is applied to an interface:

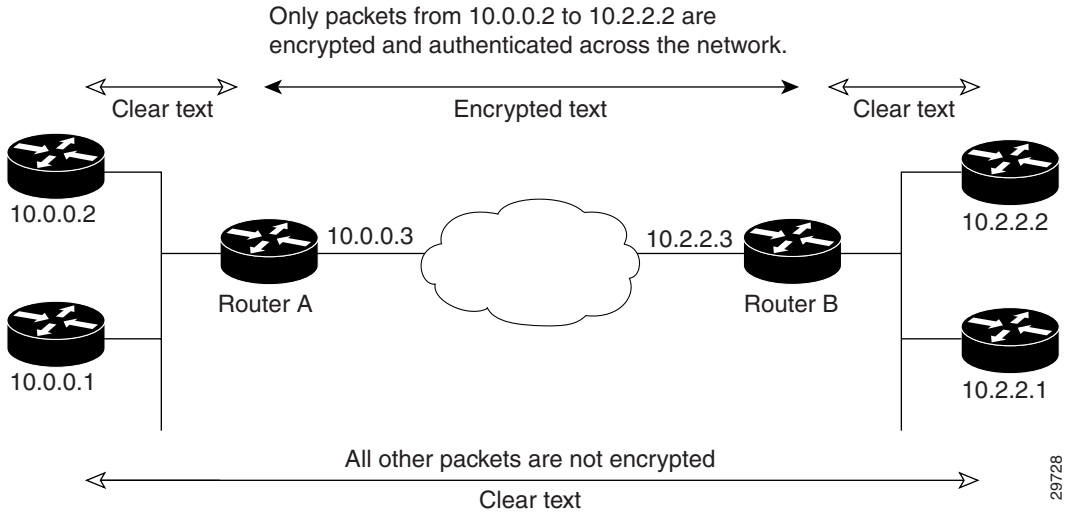
```
interface Serial0
 ip address 10.0.0.2
 crypto map toRemoteSite
```

Note In this example, IKE must be enabled.

Basic IPSec Configuration Illustration

The following is an example of an IPSec configuration in which the security associations are established through IKE. In this example an access list is used to restrict the packets that are encrypted and decrypted. In this example, all packets going from IP address 10.0.0.2 to IP address 10.2.2.2 are encrypted and decrypted and all packets going from IP address 10.2.2.2 to IP address 10.0.0.2 are encrypted and decrypted. Also, one IKE policy is created.

Figure 4-1 Basic IPSec Configuration



Router A Configuration

Specify the parameters to be used during an IKE negotiation:

```
crypto isakmp policy 15
  encryption des
  hash md5
  authentication pre-share
  group 2
  lifetime 5000

crypto isakmp key 1234567890 address 10.2.2.3
crypto isakmp identity address
```

Note In the above example, the encryption DES of policy 15 would not appear in the written configuration because this is the default value for the encryption algorithm parameter.

A transform set defines how the traffic will be protected:

```
crypto ipsec transform-set auth1 ah-md5-hmac esp-des esp-md5-hmac
  mode tunnel
```

A crypto map joins the transform set and specifies where the protected traffic is sent (the remote IPSec peer):

```
crypto map toRemoteSite 10 ipsec-isakmp
set peer 10.2.2.3
set transform-set auth1
```

The crypto map is applied to an interface:

```
  interface Serial0
  ip address 10.0.0.3
crypto map toRemoteSite
```

An IPSec access list defines which traffic to protect:

```
access-list 101 permit ip host 10.0.0.2 host 10.2.2.2
```

```
access-list 101 permit ip host 10.0.0.3 host 10.2.2.3
```

Router B Configuration

Specify the parameters to be used during an IKE negotiation:

```
crypto isakmp policy 15
  encryption des
  hash md5
  authentication pre-share
  group 2
  lifetime 5000

crypto isakmp key 1234567890 address 10.0.0.3
crypto isakmp identity address
```

A transform set defines how the traffic will be protected:

```
crypto ipsec transform-set auth1 ah-md5-hmac esp-des ah-md5-hmac
  mode tunnel
```

A crypto map joins the transform set and specifies where the protected traffic is sent (the remote IPSec peer):

```
crypto map toRemoteSite 10 ipsec-isakmp
  set peer 10.0.0.3
  set transform-set auth1
```

The crypto map is applied to an interface:

```
interface Serial0
  ip address 10.2.2.3
  crypto map toRemoteSite
```

An IPSec access list defines which traffic to protect:

```
access-list 101 permit ip host 10.2.2.2 host 10.0.0.2
access-list 101 permit ip host 10.2.2.3 host 10.0.0.3
```

Troubleshooting Tips

To verify that Cisco IOS software has recognized VAM, enter the **show diag** command and check the output. For example, when the router has the VAM in slot 1, the following output appears:

```
Router# show diag 1

    Slot 1:
            VAM Encryption/Compression engine. Port adapter
            Port adapter is analyzed
            Port adapter insertion time 00:04:45 ago
            EEPROM contents at hardware discovery:
            Hardware Revision        :1.0
            PCB Serial Number        :15485660
            Part Number              :73-5953-04
            Board Revision           :
            RMA Test History         :00
            RMA Number               :0-0-0-0
            RMA History              :00
            Deviation Number         :0-0
```

```
Product Number           :CLEO
Top Assy. Part Number    :800-10496-04
CLEI Code                :
EEPROM format version 4
EEPROM contents (hex):
  0x00:04 FF 40 02 8A 41 01 00 C1 8B 31 35 34 38 35 36
  0x10:36 30 00 00 00 82 49 17 41 04 42 FF FF 03 00 81
  0x20:00 00 00 00 04 00 80 00 00 00 00 CB 94 43 4C 45
  0x30:4F 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
  0x40:20 C0 46 03 20 00 29 00 04 C6 8A FF FF FF FF FF
  0x50:FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
  0x60:FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
  0x70:FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

To see if the VAM is currently processing crypto packets, enter the **show pas vam interface** command. The following is sample output:

```
Router# show pas vam interface

    Interface VAM 1/1 :
            ds:0x632770C8        idb:0x62813728
            Statistics of packets and bytes that through this interface:
                    18 packets in                    18 packets out
                  2268 bytes in                    2268 bytes out
                     0 paks/sec in                    0 paks/sec out
                     0 Kbits/sec in                   0 Kbits/sec out
                    83 commands out                  83 commands acknowledged
            ppq_full_err   :0            ppq_rx_err       :0
            cmdq_full_err  :0            cmdq_rx_err      :0
            no_buffer      :0            fallback         :0
            dst_overflow   :0            nr_overflow      :0
            sess_expired   :0            pkt_fragmented   :0
            out_of_mem     :0            access_denied   :0
            invalid_fc     :0            invalid_param    :0
            invalid_handle :0            output_overrun   :0
            input_underrun :0            input_overrun    :0
            key_invalid    :0            packet_invalid   :0
            decrypt_failed :0            verify_failed    :0
            attr_invalid   :0            attr_val_invalid :0
            attr_missing   :0            obj_not_wrap     :0
            bad_imp_hash   :0            cant_fragment    :0
            out_of_handles :0            compr_cancelled  :0
            rng_st_fail    :0            other_errors     :0
            633 seconds since last clear of counters
```

When the VAM processes packets, the “packets in” and “packets out” counter changes. Counter “packets out” represents the number of packets directed to the VAM. Counter “packets in” represents the number of packets received from the VAM.

Note In versions prior to Cisco IOS Release 12.2(5)T and Cisco IOS Release 12.1 (10)E, upon reboot trap configurations are lost and need to be re-entered.

To see if the IKE/IPSec packets are being redirected to the VAM for IKE negotiation and IPSec encryption and decryption, enter the **show crypto eli** command. The following is sample output when Cisco IOS software redirects packets to VAM:

```
Router# show crypto eli
Encryption Layer:  ACTIVE
 Number of crypto engines = 1.

 CryptoEngine-0 (slot-1) details.
 Capability-IPSec :IPPCP   , 3DES, RSA
 IKE-Session   :    0 active,  5120 max, 0 failed
 DH-Key        :    0 active,  5120 max, 0 failed
 IPSec-Session :    0 active, 10230 max, 0 failed
```

When the software crypto engine is active, the **show crypto eli** command yields no output.

During bootup or OIR, when the Cisco IOS software agrees to redirect crypto traffic to the VAM, it prints a message similar to the following:

```
%ISA-6-INFO:Recognised crypto engine (0) at slot-1
    ...switching to hardware crypto engine
```

To disable the VAM, use the configuration mode **crypto card shut** command, as follows:

```
Router(config)# crypto card shut 1
    Router#
    3w4d:%ISA-6-SHUTDOWN:VAM shutting down
    3w4d:%ISA-6-INFO:Crypto Engine 0 in slot 1 going DOWN
    3w4d:...switching to software crypto engine
```

Monitoring and Maintaining the VAM

Use the commands below to monitor and maintain the VAM:

Command	Purpose
`Router# show pas isa interface`	Displays the ISA interface configuration.
`Router# show pas isa controller`	Displays the ISA controller configuration.
`Router# show pas vam interface`	Verifies the VAM is currently processing crypto packets.
`Router# show pas vam controller`	Displays the VAM controller configuration.
`Router# Show version`	Displays integrated service adapter as part of the interfaces.