(To view this Guide in its entirety, click here for the PDF version.)

Chapter 2 - Design Considerations
Chapter 3 - Scenarios (Part 1)
Chapter 4 - Scenarios (Part 2)

Introduction

Internet Protocol Security (IPSec) is a framework of open standards for ensuring secure private communications over IP networks. Based on standards developed by the Internet Engineering Task Force (IETF), IPSec ensures confidentiality, integrity, and authenticity of data communications across a public IP network. IPSec provides a necessary component for a standards-based, flexible solution for deploying a network-wide security policy.

This document covers the following information for network designers, system engineers, administrators, and users implementing IPSec on Cisco equipment:

- Performance factors
- Configuration issues
- Deployment issues
- Example scenarios with configuration files
- Review of interoperability among Cisco products and feature sets, and with other vendors' products
- Troubleshooting techniques
- Examples of debugging messages

Related Documents

For additional background information on IPSec, please refer to the Cisco IOS Privacy web page (which contains information about encryption, IPSec, and Cisco IPSec products), at the following URL:

`http://www.cisco.com/warp/public/732/net_foundation/privacy_identity.html`

Other documents provide detailed definitions for some of the IPSec components. These can be found at the following URL:

`http://www.ietf.org/html.charters/ipsec-charter.html`

They include the following Request for Comment (RFC) documents:

- RFC 2104—HMAC: Keyed-Hashing for Message Authentication
- RFC 2085—HMAC-MD5 IP Authentication with Replay Prevention
- RFC 2401—Security Architecture for the Internet Protocol
- RFC 2410—The NULL Encryption Algorithm and Its Use with IPSec
- RFC 2411—IP Security Document Roadmap
- RFC 2402—IP Authentication Header
- RFC 2412—The OAKLEY Key Determination Protocol
- RFC 2451—The ESP CBC-Mode Cipher Algorithms
- RFC 2403—The Use of HMAC-MD5-96 within ESP and AH

- RFC 2404—The Use of HMAC-SHA-1-96 within ESP and AH
- RFC 2405—The ESP DES-CBC Cipher Algorithm with Explicit IV
- RFC 2406—IP Encapsulating Security Payload (ESP)
- RFC 2407—The Internet IP Security Domain of Interpretation for ISAKMP
- RFC 2408—Internet Security Association and Key Management Protocol (ISAKMP)
- RFC 2409—The Internet Key Exchange (IKE)

IPSec Overview

The IPSec initiative has been proposed to offer a standard way of establishing authentication and encryption services between endpoints. This means not only standard algorithms and transforms, but also standard key negotiation and management mechanisms to promote interoperability between devices by allowing for the negotiation of services between these devices. The Internet Key Exchange (IKE), based on ISAKMP/Oakley, is the protocol used to manage the generation and handling of keys. It is also the protocol by which potential peer devices form Security Associations.

A Security Association (SA) is a negotiated policy or agreed way of handling the data that will be exchanged between two peer devices, an example of a policy item is the transform used to encrypt data. The active SA parameters are stored in the Security Association Database (SAD).

SAs for both IKE and IPSec are negotiated by IKE over various phases and modes:

- Phase 1: IKE negotiates IPSec SAs during this phase. Two modes can be used for phase 1:
  - Main mode is used in the vast majority of situations.
  - Aggressive mode is used under rare circumstances, given particular configuration parameters between two systems.

    The user has no control over which mode is chosen. The router automatically chooses a mode, depending on the configuration parameters set up on both peers.
- Phase 2: IKE negotiates IPSec SAs during this phase. The only phase 2 exchange is quick mode.

IPSec SAs terminate through deletion or by timing out. When the SAs terminate, the keys are also discarded. When subsequent IPSec SAs are needed for a flow, IKE performs a new phase 2 and, if necessary, a new phase 1 negotiation. A successful negotiation results in new SAs and new keys. New SAs can be established before the existing SAs expire, so that a given flow can continue uninterrupted.

The components of IPSec, SAs, and IKE, are covered in more detail later.

IPSec in Detail

Within the TCP/IP environment, IPSec protocols offer security services at the IP layer. These security services include access control, connectionless integrity, data origin authentication, protection against replay, confidentiality (encryption), and limited traffic-flow confidentiality.

These security services are provided by the Authentication Header (AH) and the Encapsulating Security Payload (ESP) protocols. Each protocol provides certain services and may be used separately or together, although it is not usually necessary to use both protocols together. Refer to *Security Architecture for IP*, RFC 2401, for further reading.

The following services are provided by AH and ESP:

- Access control prevents unauthorized use of a resource such as the network behind a security gateway. (A security gateway is an intermediate system such as a router or firewall that interfaces two networks and provides security services for the hosts on the internal side.)
- Connectionless integrity detects any modification to data within individual IP packets, regardless of their position in a data stream.
- Data origin authentication verifies the source of the data. Connectionless integrity and data origin authentication are joint services, and together they are generally referred to as authentication.
- Protection against replay is optional and is a service selected by the receiver. The sender must increment the sequence number used for antireplay; the receiver can then detect the arrival of duplicate IP packets within a constrained window by checking the sequence number.
- Confidentiality protects data from unauthorized disclosure by using encryption. Confidentiality may be selected independent of any other services. Limited traffic-flow confidentiality occurs by hiding the original source and destination addresses as part of the data when using ESP in tunnel mode. It is most effective if implemented at a security gateway, where traffic aggregation may be able to mask the true source destination patterns.

Because both AH and ESP provide access control, the real decision to make when choosing security services is whether you need authentication or both authentication and encryption.

Authentication Header

AH mode provides authentication for as much of the IP header as possible, as well as for all the upper-layer protocols of an IP datagram (for example, ESP, UDP, TCP, ICMP, and Internet Group Management Protocol [IGMP]). However, some of the IP header fields may change in transit and the value of these fields, when the packet arrives at the receiver, may not be predictable by the sender. The values of such fields cannot be protected by AH.

The AH integrity check value (ICV), usually a keyed hash using a shared secret value, is computed over:

- IP header fields that are either immutable in transit or that are predictable in value upon arrival at the endpoint for the AH SA
- The AH, next header, payload len, reserved, SPI, sequence number, the authentication data (which is set to zero for this computation), and explicit padding bytes (if any)
- The upper-layer protocol data, which is assumed to be immutable in transit

The following are immutable fields and are included in the AH ICV computation:

- Version
- Internet header length
- Total length
- Identification
- Protocol (this should be the value for AH)
- Source address
- Destination address (without loose or strict source routing)

The mutable but predictable field is destination address (with loose or strict source routing).

The following are mutable fields (zeroed prior to ICV calculation) and are not included in the computation of the AH ICV:

- Type of service (ToS)
- Flags

- Fragment offset
- Time to Live (TTL)
- Header checksum

The ToS field is excluded because some routers are known to change the value of this field, even though the IP specification does not consider ToS to be a mutable header field. The flags field is excluded because an intermediate router might set the don't fragment (DF) bit, even if the source did not select it. Because AH is applied only to nonfragmented IP packets, the fragment offset field must always be zero, and thus is excluded (although it is predictable). TTL is changed en route as a normal course of processing by routers. Therefore, its value at the receiver is not predictable by the sender. Header checksum will change if any of these other fields change, so its value upon reception cannot be predicted by the sender.

Encapsulating Security Payload

ESP performs encryption at the IP packet layer. It supports a variety of symmetric encryption algorithms. The default algorithm for IPSec is the 56-bit Data Encryption Standard using the cipher block chaining mode transform (DES-CBC).This cipher must be implemented to guarantee interoperability among IPSec products.

ESP has an optional field for authentication. It contains an ICV that is computed over the remaining part of the ESP less the authentication field itself. The length of the field varies with the authentication algorithm selected. If authentication is not chosen, the ICV is omitted. Authentication is calculated after the encryption is done. The current IPSec standard specifies HMAC (a symmetric signature scheme) with Security Hash Algorithm 1 (SHA1) and Message Digest 5 (MD5) as mandatory implementations.

The Modes of AH and ESP

The format of the AH and ESP headers, and the values contained therein, will vary according to the mode in which they are applied.

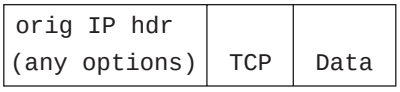
Transport Mode

Transport mode is used when both communicating peers are hosts. It may also be applied when one peer is a host and the other is a gateway, if that gateway is acting as a host (for example, a router sending SNMP traffic to an SNMP manager).

Transport mode has an advantage of adding only a few bytes to the header of each packet. With this choice, the original header is not protected. This setup allows the true source and destination addresses to be seen by intermediate devices. One advantage of not changing the original header is that Quality of Service (QoS), for example, can be processed from the information in the IP header. One disadvantage is that it is possible to do traffic analysis on the packets. Remember, transport mode can be used only if the two end devices are the ones providing IPSec protection. Transport mode cannot be used if an intermediate device, such as a router or firewall, is providing the IPSec protection.

Figure 1-1 depicts an original IPv4 packet.

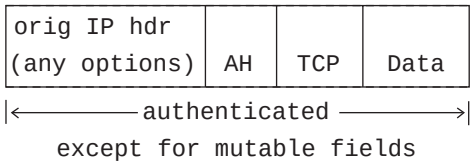
Figure 1-1 Original IPv4 packet



Authentication Header

AH services protect the external IP header along with the data payload (see Figure 1-2). It protects all the fields in the header that do not change in transport. The header goes after the IP header and before the ESP header, if present, and other higher-layer protocols.

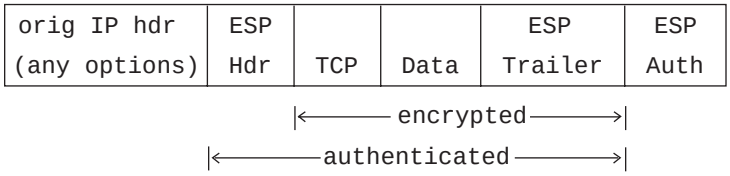
Figure 1-2 IP Packet after applying AH in Transport Mode



Encapsulating Security Payload

In transport mode, the IP payload is encrypted and the original headers are left intact. The ESP header is inserted after the IP header and before the upper-layer protocol header (see Figure 1-3). The upper-layer protocols are encrypted and authenticated along with the ESP header. ESP does not authenticate the IP header itself. This authentication is important only for IPv6 because there are no known attacks that would require an IPv4 header to be authenticated. Please note that higher-layer information, such as port numbers in the layer 4 header, are not available because they are part of the encrypted payload.

Figure 1-3 IP Packet after applying ESP in Transport Mode



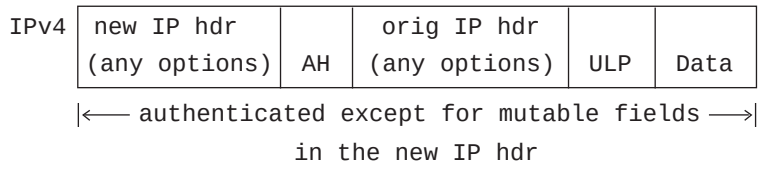
Tunnel Mode

Tunnel mode is used between two gateway devices, or between a host and a gateway if that gateway is the conduit to the actual source or destination. In tunnel mode, the entire original IP packet is encrypted and becomes the payload of a new IP packet. The new IP header has the destination address of its IPSec peer. This setup allows for tunneling of IP from a protected host (that is not doing IPSec itself) through a router or firewall usually to another router or firewall, both acting as security gateways. One of the advantages of tunnel mode is that intermediate devices, such as routers, can do the encryption without modifying end systems. All the information from the original packet, including the headers, is protected. Tunnel mode protects against traffic analysis because, although the IPSec tunnel endpoints can be determined, the true source and destination endpoints cannot be determined because the information in the original IP header has been encrypted.

Authentication Header Example

In this case, all of the original header is authenticated and the new IP header is protected in the same way as the IP header in transport mode (see Figure 1-4).

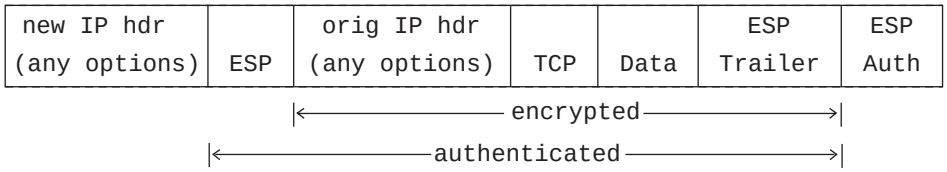
Figure 1-4 IP Packet after applying AH in Tunnel Mode



Encapsulating Security Payload

When ESP is used and confidentiality is selected in tunnel mode, the original IP header is well-protected because the entire original IP datagram is encrypted (see Figure 1-5). With an ESP authentication mechanism, the original IP datagram and the ESP header are included; however, the new IP header is not included in the authentication. This fact is important only for IPv6 because there are no known attacks that would require an IPv4 header to be authenticated. When both authentication and encryption are selected, encryption is performed first, before authentication. One reason for this order of processing is that it facilitates rapid detection and rejection of replayed or bogus packets by the receiving node. Prior to decrypting the packet, the receiver can detect the problem and potentially reduce the impact of denial-of-service attacks.

Figure 1-5 IP Packet after applying ESP in Tunnel Mode



To summarize, when you want to make sure that certain data from a known and trusted source (authentication) gets transferred with integrity and does *not* need confidentiality, use the AH protocol. AH protects the upper-layer protocols and the IP header fields that do not change in transit, such as the source and destination addresses. AH cannot protect those fields that change in transit, such as the TTL field. Protection means that the values cannot be changed without detection, so the IPSec node will reject any altered IP datagram. To clarify, AH does not protect against someone sniffing the wire and seeing the headers and data. However, because headers and data cannot be changed without the change being detected, changed packets would get rejected.

If you need to keep data private (confidentiality), then you *must* use ESP. ESP will encrypt the upper-layer protocols in transport mode and the entire original IP datagram in tunnel mode so that neither are readable off the wire. ESP can also provide authentication for the packets. However, when using ESP in transport mode, the outer IP original header is not protected; in tunnel mode, the new IP header is not protected. Users will probably implement tunnel mode more than transport mode during initial IPSec usage.

ESP can also be applied with the choice of the NULL encryption algorithm. When NULL is selected, no privacy is provided, thus it is recommended that an ESP authentication algorithm be used to afford integrity and authenticity to the data. The use of the ESP NULL algorithm with authentication is almost analogous to AH, although there are still differences in the fields used in the authentication calculation, as described in the preceding section.

Table 1-1 summarizes the services provided by the different IPSec protocols in **transport mode**.

Table 1-1 IPSec Transport Mode Services

Security Service	AH	ESP	AH & ESP
Access Control	Yes	Yes	Yes
Connectionless Integrity Data Origin Authentication	Authenticates parts of original IP header along with upper-layer protocols	Encrypts and optionally authenticates upper-layer protocol; does not authenticate original IP header	Encrypts and authenticates upper-layer protocol; authenticates parts of original IP header
Antireplay Service	Optional	Optional	Optional
Confidentiality	No	Yes	Yes
Limited Traffic-Flow Confidentiality	No	No	No

Table 1-2 summarizes the services provided by the different IPSec protocols in **tunnel mode**.

Table 1-2 IPSec Tunnel Mode Services

Security Service	AH	ESP	AH & ESP
Access Control	Yes	Yes	Yes
Connectionless Integrity Data Origin Authentication	Authenticates all of original IP datagram and parts of new IP header	Encrypts and optionally authenticates original IP datagram; does not authenticate new IP header	Encrypts and authenticates original IP datagram; authenticates parts of new IP header
Antireplay Service	Optional	Optional	Optional
Confidentiality	No	Yes	Yes
Limited Traffic-Flow Confidentiality	No	Yes	Yes

Table 1-3 Summarizes the algorithms required to be supported in an IPSec implementation.

Table 1-3 IPSec Required Algorithms

Security Service	AH	ESP
Authentication and Integrity	HMAC-MD5 (128-bit hash) HMAC-SHA (160-bit hash) Both calculated on payload and non-mutable fields of the IP header	HMAC-MD5 HMAC-SHA Both calculated on the payload of the encrypted IP datagram and ESP header
Encryption and Privacy		56-bit DES-CBC 3 DES with 3 unique keys

Security Associations

As mentioned in the overview, the establishment of SAs facilitates an IPSec-based conversation between two devices. Each device must agree on the policies or rules of their conversation by negotiating these policies with their potential peer.

IKE provides negotiation, peer authentication, key management, and key exchange. IKE negotiates a "contract" between the two IPSec endpoints. The SA is used to keep track of all the details of this negotiation for a single IPSec session. In fact, one of the main functions of IKE is to establish and maintain the SAs. The details that are tracked include:

- The mode of the authentication algorithm used in AH, and the keys to the authentication algorithm
- The mode of the encryption algorithm used in ESP, and the keys to the encryption algorithm
- The authentication algorithm, mode used in ESP, and the keys
- The SA lifetime (also the lifetime of the corresponding keys)
- The SA destination address
- The identities on whose behalf this SA is established

SAs should be viewed as the instantiation of security policy for a given data flow within the IKE/IPSec environment.

An SA is unidirectional, so it represents a simplex connection for the traffic. For example, when a bidirectional TCP session exists between two systems, A and B, there will be one SA from A to B and a separate SA from B to A. Two SAs (one in each direction) are required. The SA reflects the security services provided by the use of AH or ESP, but not both. If both AH and ESP are applied to a unidirectional traffic stream, then two SAs are created for the traffic stream.

An SA is uniquely identified by an IP destination address, a security protocol (AH or ESP) identifier, and a unique Security Parameter Index (SPI, pronounced spy) value. A SPI is a 32-bit number assigned to the initiator of the SA request by the receiving IPSec endpoint. The SPIs for AH and ESP are not the same for a given traffic flow (for example, IP|AH(500)|ESP(99999)|…). The SPI value is a field of both the AH and ESP headers. It is used to identify the appropriate SA in all the communications between the two end nodes. On receiving a packet, the destination address, protocol, and SPI are used to determine the SA, which allows the node to authenticate or decrypt the packet according to the security policy configured for that SA.

Below is a short piece of some debugging messages using both the **`debug crypto isakmp`** and **`debug crypto ipsec`** commands. This short section of debug display was captured after IKE had been negotiated. The IKE and IPSec processes are creating two IPSec SAs, one in each direction. Notice the IP addresses for the two SAs. The source and destination addresses of the encryption endpoints are reversed on a bidirectional communication session.

The configuration specifics are as follows:

- Tunnel mode
- Protecting all the Telnet traffic originating from 192.168.X.X to 172.16.24.X

Please note that ISAKMP in the parser is really referring to IKE. The bolded text is used for emphasis.

```
ISAKMP (113): Creating IPSec SAs
        inbound SA from 10.213.55.2 to 10.16.157.2 (proxy 172.16.24.0 to 192.168.0.0)
        has spi 295377457 and conn_id 116 and flags 4
        lifetime of 3600 seconds
        lifetime of 4608000 kilobytes
       outbound SA from 10.16.157.2 to 10.213.55.2 (proxy 192.168.0.0 to 172.16.24.0)
        has spi 51978768 and conn_id 117 and flags 4
        lifetime of 3600 seconds
        lifetime of 4608000 kilobytes
IPSEC(key_engine): got a queue event...
IPSEC(initialize_sas): processing an sa request of size 148
IPSEC(initialize_sas): ,
  (kei) dest= 10.16.157.2, src= 10.213.55.2,
    dest_proxy= 192.168.0.0/255.255.0.0/6/0,
    src_proxy= 172.16.24.0/255.255.255.0/6/23,
    protocol= 3, transform= 2, hmac_alg= 0,
    lifedur= 0xE10s and 0x465000kb,
    spi= 0x119B1A31(295377457), conn_id= 116, keysize= 0, flags= 0x4
IPSEC(initialize_sas): ,
  (kei) src= 10.16.157.2, dest= 10.213.55.2,
    src_proxy= 192.168.0.0/255.255.0.0/6/0,
    dest_proxy= 172.16.24.0/255.255.255.0/6/23,
    protocol= 3, transform= 2, hmac_alg= 0,
    lifedur= 0xE10s and 0x465000kb,
    spi= 0x3192210(51978768), conn_id= 117, keysize= 0, flags= 0x4
IPSEC(create_sa): sa created,
  (sa) sa_dest= 10.16.157.2, sa_prot= 50,
    sa_spi= 0x119B1A31(295377457),
    sa_trans= esp-des , sa_conn_id= 116,
  (identity) local= 10.16.157.2, remote= 10.213.55.2,
    local_proxy= 192.168.0.0/255.255.0.0/6/0,
    remote_proxy= 172.16.24.0/255.255.255.0/6/23
IPSEC(create_sa): sa created,
  (sa) sa_dest= 10.213.55.2, sa_prot= 50,
    sa_spi= 0x3192210(51978768),
    sa_trans= esp-des , sa_conn_id= 117,
  (identity) local= 10.16.157.2, remote= 10.213.55.2,
    local_proxy= 192.168.0.0/255.255.0.0/6/0,
    remote_proxy= 172.16.24.0/255.255.255.0/6/23
```

Now the SAs have been created. To view the IPSec SAs that are active in the system, you use the **show crypto ipsec sa** command.

```
router-1603#sh crypto ipsec sa

interface: Serial0
    Crypto map tag: combined, local addr. 10.16.157.2

local  ident (addr/mask/prot/port): (192.168.29.212/255.255.255.255/6/0)
   remote ident (addr/mask/prot/port): (172.16.10.19/255.255.255.255/6/21)
   current_peer: 10.213.55.2
     PERMIT, flags={origin_is_acl,ident_is_ipsec,}
    #pkts encaps: 17, #pkts encrypt: 0, #pkts digest 17
    #pkts decaps: 15, #pkts decrypt: 0, #pkts verify 15
    #send errors 3, #recv errors 0

     local crypto endpt.: 10.16.157.2, remote crypto endpt.: 10.213.55.2
     path mtu 1500, media mtu 1500
     current outbound spi: BC311E4

     inbound esp sas:
      spi: 0x119B1A31(295377457)
        transform: esp-des ,
        in use settings ={Tunnel, }
        slot: 0, conn id: 116, crypto map: combined
        sa timing: remaining key lifetime (k/sec): (413694/3475)
        IV size: 8 bytes
        replay detection support: N

     inbound ah sas:

     outbound esp sas:
      spi: 0x3192210(51978768)
        transform: esp-des ,
   in use settings ={Tunnel, }
        slot: 0, conn id: 117, crypto map: combined
        sa timing: remaining key lifetime (k/sec): (413694/3475)
        IV size: 8 bytes
        replay detection support: N

     outbound ah sas:
```

The SA concept allows for the building of secure virtual private networks (VPNs). It allows you to differentiate between data flows and provide differing levels of security to them.

Key Management

Internet Key Exchange

IKE is the facilitator and manager of IPSec-based conversations. IKE is based on ISAKMP/Oakley, or the Internet Security Association Key Management Protocol/Oakley, which gives the reader a good understanding of the actual role of the protocols, establishment of security associations, and the management of keys. So IKE is a derivative of ISAKMP/Oakley specifically for IPSec.

IKE provides three modes for exchanging key information and setting up IKE SAs. The first two modes are phase 1 exchanges, which are used to set up the initial secure channel. The other mode is the phase 2 exchange, which negotiates IPSec SAs. The two modes in phase 1 are main mode and aggressive mode, and the phase 2 mode is called quick mode. The basic idea is to bootstrap an IKE SA to provide a protected pipe for subsequent protected IKE exchanges between the IKE peers, and then use phase 2 quick mode with the IKE SA to negotiate the IPSec SAs. Only IPSec uses the IPSec SA for protecting traffic.

Main mode has three two-way exchanges between the initiator and receiver. In the first exchange, the algorithms and hashes are agreed upon. The second exchange uses Diffie-Hellman to agree on a shared secret and to pass nonces (random numbers sent to the other party, signed and returned to prove their identity—they are signed only if encrypted nonces or digital signatures are being used). The third exchange verifies the identity of the other side.

In aggressive mode, fewer exchanges are done and with fewer packets. On the first exchange, almost everything is squeezed in—the proposed SA (algorithm, hashes, and mode), Diffie-Hellman public value, a nonce that the other party signs, and an ID packet, which can be used to verify identity via a third party. The receiver sends everything back that is needed to complete the exchange. The only thing left is for the initiator to confirm the exchange. The weakness of using the aggressive mode is that both sides have exchanged information before there is a secure channel. Therefore, it is possible to snoop the wire and discover who formed the new SA. However, aggressive mode is faster than main mode.

Cisco IOS software automatically determines the correct phase 1 mode to use.

Quick mode occurs after IKE has established the secure tunnel. Every packet is encrypted in quick mode. Both negotiation of the IPSec SA and deriving the key material needed by IPSec are accomplished in quick mode.

Before IKE will proceed, it is a requirement that the potential parties agree upon a way to authenticate themselves to each other. This authentication method is negotiated during the IKE phase 1 main mode exchange.

IKE Authentication

The IKE protocol is very flexible and supports multiple authentication methods as part of the phase 1 exchange. The two entities must agree on a common authentication protocol through a negotiation process. At this time, the following mechanisms are generally implemented:

- Preshared keys are the same keys preinstalled on each host. IKE peers authenticate each other by computing and sending a keyed hash of data that includes the preshared key. If the receiving peer is able to independently create the same hash using its preshared key, then it knows that both parties must share the same secret, thus authenticating the other party. Preshared keys are a nonpublic key option. As with manual keys, each peer shares a secret key, which has been exchanged out-of-band and configured into the router. Although, like manual keys, this scenario has some limitations for scaling, preshared keys scale better than manual IPSec keys and provide a more robust environment. First, it requires fewer keys to configure—one key per peer instead of two keys or more. Second, the preshared key is used in the phase 1 exchange, and the phase 2 derivation is still performed. This scenario adds to the scalability by allowing multiple streams between peers. Also, the ability of each side to demonstrate knowledge of this secret (without mentioning it) authenticates the exchange. Third, manual IPSec mode does not allow for rekeying of IPSec SAs between peers, although IKE does. Fourth, the keys derived are of better quality because the Diffie-Hellman key derivation is used. Finally, the actual work required at each peer makes it easier to implement.
- Public key cryptography requires that each party generate a pseudorandom number (a nonce) and encrypt it in the other party's public key. Authentication occurs when each party decrypts the other party's nonce with a local private key (and other publicly and privately available information) and then uses the decrypted nonce to compute a keyed hash. This system provides for deniable transactions. That is, either side of the exchange can plausibly deny that they took part in the exchange. Currently, only the RSA public key algorithm is supported.
- With the digital signature, each device digitally signs a set of data and sends it to the other party. This method is similar to the previous one, except that it provides nonrepudiation. Currently both the RSA public key algorithm and the Digital Signature Standard (DSS) are supported.

The Importance of Certificates

Sharing keys does not scale to a large network. One method that does scale is encapsulation of the public key in a digital certificate authenticated by a Certificate Authority (CA). A CA is a trusted third party who can bind a public key to an identity. In this case, that includes the identity of devices, especially router and firewall devices. Within a CA domain, each device needs only its own certificate and the public key of the CA in order to authenticate every other device within the domain. The two encryption endpoints exchange these preexisting certificates that they have obtained from the CA.

The issuance of the certificate by the CA should be done at the time you install the device. It is not done for each IKE exchange. That is, after a device obtains a certificate, it does not need to contact the CA during the connection setup process. This setup avoids the reliability problems associated with a centralized key server.

The Cisco Certificate Enrollment Protocol (CEP) manages the certificate life-cycle process. This protocol supports operations such as certificate enrollment, certificate revocation, and certificate access allowing Cisco routers to participate within a public key infrastructure (PKI).

The general procedure is to first generate an RSA key pair on the router. The generated keys are saved in the private configuration in NVRAM. This private configuration is never displayed to the user or backed up to another device. A single CA must be defined for the router using its fully qualified Domain Name (FQDN) or IP address, the CA's HTTP CGI script path, and optionally, the certificate revocation list (CRL) query URL. Second, the CA certificate must be retrieved. In order to authenticate the certificate of another device, the router must query the CA to obtain the CA certificate containing the CA public key. Because the router has no means of

validating the CA certificate automatically, the key should be authenticated manually by contacting the CA administrator and comparing the fingerprints of the certificate. The validated certificate is stored in the configuration, and the public key from the CA certificate is added to the RSA public key chain. Third, request that the CA generate a certificate for the key pair of the router by sending an enrollment request. A dialog is begun in which the user-defined certificate attributes are entered (a challenge password, a user addressable name, and optionally, the router serial number and IP addresses). The CA operator may cease to manually validate the request, and the fingerprints can be compared. The last step is receiving the certificate. Because there may be a significant delay, the certificate is retrieved asynchronously.

Note that there are situations in which IKE cannot establish policy and keys for IPSec. If there is no certificate defined in the router and there are only public key-based authentication methods in IKE policy, or if there is no certificate and no preshared keys for the peer (either shared directly by address or by a host name that has been configured with that address), IKE cannot negotiate with the peer and IPSec will not be able to work.

Configuration Overview

This section presents a sample IPSec and IKE configuration for the Cisco IOS software. The configuration will be broken down into major areas discussed as follows:

- Configuring IKE Security Associations
- Configuring IPSec Security Associations
- Configuring Crypto Maps
- Applying the Crypto Map and Triggering IPSec
- Configuring IPSec with Manual Keys

Configuring IKE Security Associations

Following is a sample configuration for setting up the IKE SA. The parameters for the SA are negotiated during IKE main mode. This example uses preshared keys as the authentication method. The actual shared key is shown in boldface type and must be the same on both peers.

```
crypto isakmp policy 4
 hash md5
 authentication pre-share
crypto isakmp key iketest address 10.213.55.2
```

This defines the key as being shared with the peer based on the address of the peer. It is also possible to share a key based on the host name of a peer.

```
crypto isakmp key farFrobbarNitz hostname mypeer.some.com
```

To share a key based on host name, it is necessary for that host name to be configured in the router already or available via DNS. For instance,

```
ip host mypeer.some.com 10.12.12.2  10.4.4.2  192.168.1.33
```

would define the host mypeer.some.com to have any one of those three addresses. If IKE was asked to initiate a connection to 192.168.1.33 and there was no preshared key based on that address, it would check whether a preshared key based on the host name existed for a host configured to have that address. If a match was found, any preshared key protection suites configured in the router would be offered to the peer.

There is another caveat with sharing keys based on a host name. The IKE process on a router assumes an identity during its negotiations. The possible identities are IPv4 address and Fully Qualified Domain Name. Using the following commands:

```
crypto isakmp identity address
crypto isakmp identity hostname
```

It is possible to instruct IKE to assume an address-based or host name-based identity, respectively. If two peers are configured to share host name-based preshared keys, they must also be configured to use their host name as their identity. Similarly, if they share address-based preshared keys, they must use their address identity.

Configuring IPSec Security Associations

After IKE main mode has established the IKE SA, the peers will use quick mode to negotiate the parameters that form the basis of the IPSec SA. The following commands are an example of configuring these SA parameters; they include such parameters as determining the lifetime for which the SA will remain valid, the types of headers (AH, ESP, or AH and ESP) required, and the algorithms applied to the data exchanged between the peers.

```
crypto ipsec security-association lifetime seconds 120
!
crypto ipsec transform-set barney esp-des esp-sha-hmac
crypto ipsec transform-set fred esp-des esp-md5-hmac
 mode transport
crypto ipsec transform-set wilma esp-des
```

Configuring Crypto Maps

The IKE and IPSec SA parameters are brought together in a structure called a crypto map. This crypto map defines the peer to be contacted and the SA parameters to negotiate with that peer. The final piece of the crypto map is to define the traffic that will be used as the trigger mechanism for the IKE and IPSec SA establishment.

```
crypto map flintstone 1 ipsec-isakmp
 set peer 10.213.55.2
 set transform-set barney wilma fred
 match address 101
```

When this router receives traffic destined for a network that matches the rules defined in access list 101, it will initiate an IKE main mode exchange with the peer 10.213.55.2. It initiates IKE because of the ipsec-isakmp type of the crypto map. When the IPSec SA is negotiated, this router will make three offers, barney, wilma, and fred, each of which specifies different SA parameters. The peer 10.213.55.2 will choose the offer that matches the SA definitions configured for this device. If there is more than one match, the more secure offer will be chosen. So, for example, if wilma and barney were both suitable to peer 10.213.55.2, it would choose barney because this SA offers addition security by the application of an authentication algorithm with ESP.

Applying the Crypto Map and Triggering IPSec

Two tasks remain before the router configuration is complete. The crypto map defined above must be applied to a router interface before it will be triggered.

```
interface Serial0
 ip address 10.213.55.1 255.255.255.0
crypto map flintstone
```

As traffic is queued for outbound transit, it will be compared with the rules defined in access list 101 below. This list is referenced in the flintstone crypto map via match address 101. If the traffic is a match, this will initiate IKE for the first packet in the flow to the peer device. Subsequent packets are processed as per the SA definitions until the expiration of an SA lifetime or the manual clearing of an SA. If the IPSec SA expires, IKE quick mode will be triggered and a new IPSec SA established. If the IKE SA expires, main mode will be reexecuted, followed by quick mode.

```
access-list 101 permit ip 192.168.1.0 0.0.0.255 192.168.0.0 0.0.0.255
access-list 101 permit ip 192.168.1.0 0.0.0.255 172.17.0.0 0.0.255.255
```

Configuring IPSec and Manual Keying

IKE provides for the dynamic creation of SAs and is the preferred method to use with IPSec; however, the standard also requires the implementation of manual SA configuration. The following section discusses manual configuration.

Manual keying involves a "face-to-face" exchange of keys and, because this method of transportation is not a scalable solution for IPSec, it is outside the scope of this document. Although manual keying is specified as mandatory and Cisco IOS software supports it, Cisco does not recommend that users actually implement it. A benefit of manual keying is that it allows Cisco to work with other vendors' products.

Briefly, the process involves the typing of remote keys during the configuration of Cisco routers. Each crypto map requires multiple keys. For AH authentication, there is a key for both the outbound and inbound sessions. For ESP, there is a cipher and authentication key for both the outbound and inbound sessions.

Following is an example from a Cisco router configured with manual keying. It illustrates the selection of ESP doing authentication, encryption, and AH:

```
crypto map local-side 8 ipsec-manual
 set peer 144.254.1.2
 set session-key inbound esp 4098 cipher
  AAAA1234567890AAAA1234567890AAAA1234567890AAA0
 set session-key inbound esp 4099 auth 1234567890AAAA1234567890AAAA1234567890AAAA1234
 set session-key outbound esp 4096 cipher
  BBBB1234567890BBBB1234567890BBBB1234567890BBB0
 set session-key outbound esp 4097 auth 1234567890BBBB1234567890BBBB1234567890BBBB1234
 set session-key inbound ah 3073 CCCC1234567890CCCC1234567890CCCC1234567890CCCC0
 set session-key outbound ah 3072 DDDD1234567890DDDD1234567890DDDD1234567890DDD0
 set transform-set combrfc
 match address 103
!
```

Following is an example of the configuration from the peer router that acts as the crypto endpoint on the remote side:

```
crypto map remote-side 8 ipsec-manual
 set peer 144.254.1.1
 set session-key inbound esp 4096 cipher
  BBBB1234567890BBBB1234567890BBBB1234567890BBB0
 set session-key inbound esp 4097 auth 1234567890BBBB1234567890BBBB1234567890BBBB1234
 set session-key outbound esp 4098 cipher
  AAAA1234567890AAAA1234567890AAAA1234567890AAA0
 set session-key outbound esp 4099 auth 1234567890AAAA1234567890AAAA1234567890AAAA1234
 set session-key inbound ah 3072 DDDD1234567890DDDD1234567890DDDD1234567890DDDD0
 set session-key outbound ah 3073 CCCC1234567890CCCC1234567890CCCC1234567890CCC0
 set transform-set combrfc
 match address 103
!
```

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems Europe s.a.r.l.
Parc Evolic, Batiment L1/L2
16 Avenue du Quebec
Villebon, BP 706
91961 Courtaboeuf Cedex
France
http://www-europe.cisco.com
Tel: 33 1 69 18 61 00
Fax: 33 1 69 28 83 26

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Headquarters
Nihon Cisco Systems K.K.
Fuji Building, 9th Floor
3-2-3 Marunouchi
Chiyoda-ku, Tokyo 100
Japan
http://www.cisco.com
Tel: 81 3 5219 6250
Fax: 81 3 5219 6001

Cisco Systems has more than 200 offices in the following countries. Addresses, phone numbers, and fax numbers are listed on the Cisco Connection Online Web site at http://www.cisco.com/offices.

Argentina • Australia • Austria • Belgium • Brazil • Canada • Chile • China • Colombia • Costa Rica • Croatia • Czech Republic • Denmark • Dubai, UAE Finland • France • Germany • Greece • Hong Kong • Hungary • India • Indonesia • Ireland • Israel • Italy • Japan • Korea • Luxembourg • Malaysia Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland • Portugal • Puerto Rico • Romania • Russia • Saudi Arabia • Singapore Slovakia • Slovenia • South Africa • Spain • Sweden • Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela