CHAPTER 1

Introduction to Cisco IPsec Technology

IPsec Overview

A secure network starts with a strong security policy that defines the freedom of access to information and dictates the deployment of security in the network. Cisco Systems offers many technology solutions for building a custom security solution for Internet, extranet, intranet, and remote access networks. These scalable solutions seamlessly interoperate to deploy enterprise-wide network security. Cisco System's IPsec delivers a key technology component for providing a total security solution. Cisco's IPsec offering provides privacy, integrity, and authenticity for transmitting sensitive information over the Internet.

Cisco's end-to-end offering allows customers to implement IPsec transparently into the network infrastructure without affecting individual workstations or PCs. Cisco IPsec technology is available across the entire range of computing infrastructure: Windows 95, Windows NT 4.0, and Cisco IOS software.

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

IPsec's method of protecting IP datagrams takes the following forms:

- Data origin authentication
- Connectionless data integrity authentication
- Data content confidentiality
- Anti-replay protection
- Limited traffic flow confidentiality

IPsec protects IP datagrams by defining a method of specifying the traffic to protect, how that traffic is to be protected, and to whom the traffic is sent.

IPsec Business Applications

By implementing security at the IP level, an organization can ensure secure networking not only for applications that have security mechanisms but also for the many security-ignorant applications.

IPsec provides the capability to secure communications across a LAN, across private and public WANs, and across the Internet. Examples of its use include:

- Secure branch office connectivity over the Internet

  A company can build a secure virtual private network over the Internet or over a public WAN. This enables a business to rely heavily on the Internet and reduce its need for private networks, saving costs and network management overhead.

- Secure remote access over the Internet

  An end user whose system is equipped with IP security protocols can make a local call to an Internet Service Provider (ISP) and gain secure access to a company network. This reduces the cost of toll charges for traveling employees and telecommuters.

- Establishment of extranet and intranet connectivity with partners

  IPsec can be used to secure communication with other organizations, ensuring authentication and confidentiality and providing a key exchange mechanism.

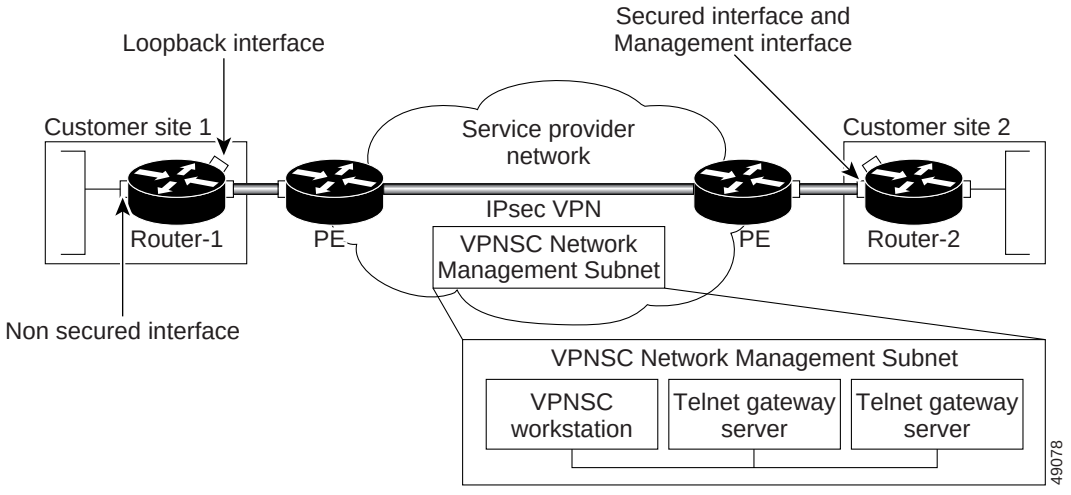
- Enhancement of electronic commerce security:

  Most efforts to date to secure electronic commerce on the Internet have relied upon securing Web traffic with SSL since that is commonly found in Web browsers and is easy to set up and run. There are new proposals that may utilize IPsec for electronic commerce.

The principal feature of IPsec that enables it to support these varied applications is that it can encrypt or authenticate all traffic at the IP level. Thus, all distributed applications, including remote logon, client/server, e-mail, file transfer, Web access, and so on, can be secured.

Figure 1-1 shows a typical IPsec usage scenario in a Cisco IPsec Solutions environment.

Figure 1-1 A Typical Cisco IPsec Solutions Scenario



The VPN Solutions Center 2.0 workstation and one or more Telnet Gateway servers function as the Network Operations Center (NOC). As shown in Figure 1-1, the VPN Solutions Center 2.0 workstation is typically placed inside the Service Provider “cloud.”

Organizations usually maintain LANs at dispersed locations. In this typical business scenario, traffic on each LAN does not need any special protection, but the devices on the LAN can be protected from the untrusted network with firewalls.

Since we live in a distributed and mobile world, the people who need to access the services on each of the LANs may be at sites across the Internet. This company can use IPsec protocols to protect their access. These protocols can operate in networking devices, such as a router or firewall that connects each LAN to the outside world, or they can operate directly on the workstation or server.

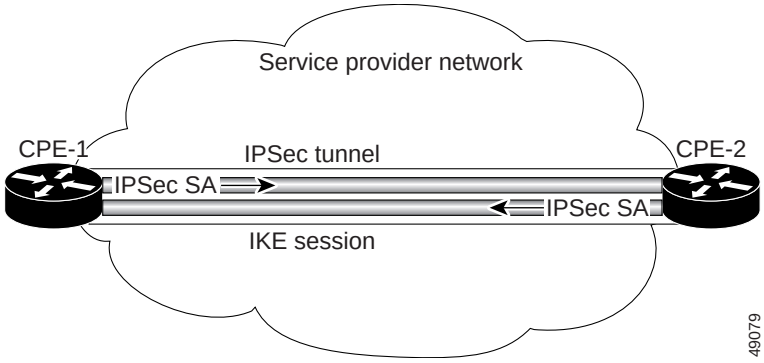
In Figure 1-1, the user workstation connected to one of the CPEs in a customer site can establish an IPsec tunnel with the network devices to protect all the subsequent sessions. After this tunnel is established, the workstation can have many different sessions with the devices behind these IPsec gateways. The packets going across the Internet will be protected by IPsec, but will be delivered onto each LAN as a normal IP packet.

How IPsec Works

IPsec provides secure *tunnels* between two peers, such as two routers. You define which packets are considered sensitive and should be sent through these secure tunnels, and you define the parameters which should be used to protect these sensitive packets, by specifying characteristics of these tunnels. Then, when the IPsec peer sees such a sensitive packet, it sets up the appropriate secure tunnel and sends the packet through the tunnel to the remote peer.

Figure 1-2 shows a high-level view of IPsec deployment across an IP network.

Figure 1-2 IPsec Deployed Across a Public IP Network



More accurately, these tunnels are sets of *security associations* (SAs) that are established between two IPsec peers. The security associations define which protocols and algorithms should be applied to sensitive packets, and also specify the keying material to be used by the two peers. Security associations are unidirectional and are established per security protocol (AH or ESP).

With IPsec you define what traffic should be protected between two IPsec peers by configuring access lists and applying these access lists to interfaces by way of *crypto map sets*. Therefore, traffic can be selected based on source and destination address, and optionally Layer 4 protocol, and port. The access lists used for IPsec only determine which traffic should be protected by IPsec, not which traffic should be blocked or permitted through the interface. Separate access lists define blocking and permitting at the interface.

A *crypto map* set can contain multiple entries, each with a different access list. The crypto map entries are searched in order—the router attempts to match the packet to the access list specified in that entry. It is good practice to place the most important crypto map entries at the top of the list.

When a packet matches a permit entry in a particular access list, and the corresponding crypto map entry is tagged as cisco, then CET is triggered, and connections are established if necessary. If the crypto map entry is tagged as ipsec-isakmp, IPsec is triggered.

If no security association exists that IPsec can use to protect this traffic to the peer, IPsec uses the Internet Key Exchange protocol (IKE) to negotiate with the remote peer to set up the necessary IPsec security associations on behalf of the data flow. The negotiation uses information specified in the crypto map entry as well as the data flow information from the specific access list entry.

If the crypto map entry is tagged as ipsec-manual, IPsec is triggered. If no security association exists that IPsec can use to protect this traffic to the peer, the traffic is dropped. In this case, the security associations are installed via the configuration, without the intervention of IKE. If the security associations did not exist, IPsec did not have all of the necessary pieces configured.

Once established, the set of security associations (outbound, to the peer) is then applied to the triggering packet as well as to subsequent applicable packets as those packets exit the router. *Applicable packets* are packets that match the same access list criteria that the original packet matched. For example, all applicable packets could be encrypted before being forwarded to the remote peer. The corresponding inbound security associations are used when processing the incoming traffic from that peer.

If IKE is used to establish the security associations, the security associations will have lifetimes set so that they periodically expire and require renegotiation, thus providing an additional level of security.

Multiple IPsec tunnels can exist between two peers to secure different data streams, with each tunnel using a separate set of security associations. For example, some data streams might be just authenticated while other data streams must both be encrypted and authenticated.

Access lists associated with IPsec crypto map entries also represent which traffic the router requires to be protected by IPsec. Inbound traffic is processed against the crypto map entries—if an unprotected packet matches a permit entry in a particular access list associated with an IPsec crypto map entry, that packet is dropped because it was not sent as an IPsec-protected packet.

Crypto map entries also include *transform sets*. A transform set is an acceptable combination of security protocols, algorithms, and other settings to apply to IPsec protected traffic. During the IPsec security association negotiation, the peers agree to use a particular transform set when protecting a particular data flow.

IPsec implements network layer encryption and authentication, embedding end-to-end security within the network architecture. The advantage to this is that individual applications do not need to be modified to take advantage of strong security. All packets routed through the network are automatically secured.

The Benefits of IPsec Technology

The benefits of IPsec are as follows:

- When IPsec is implemented in a firewall or router, it provides strong security that can be applied to all traffic crossing the perimeter. Traffic within a company or workgroup does not incur the overhead of security-related processing.
- IPsec is below the transport layer (TCP, UDP), so is transparent to applications.

  There is no need to change software on a user or server system when IPsec is implemented in the firewall or router. Even if IPsec is implemented in end systems, upper layer software, including applications, is not affected.

- IPsec can be transparent to end users. There is no need to train users on security mechanisms, issue keying material on a per-user basis, or revoke keying material when users leave the organization.
- IPsec can provide security for individual users if needed. This feature is useful for offsite workers and also for setting up a secure virtual subnetwork within an organization for sensitive applications.

The Scope of IPsec

IPsec provides three main facilities:

- An authentication-only function, referred to as Authentication Header (AH)
- A combined authentication/ encryption function called Encapsulating Security Payload (ESP)
- A key exchange function. For virtual private networks, both authentication and encryption are generally desired, because it is important both to a) assure that unauthorized users do not penetrate the virtual private network, and b) assure that eavesdroppers on the Internet cannot read messages sent over the virtual private network.

  Because both features are generally desirable, most implementations are likely to use ESP rather than AH. The key exchange function allows for manual exchange of keys as well as an automated scheme.

Cisco IPsec Technologies

Cisco IPsec includes the following technologies:

- IPsec

  IPsec uses encryption technology to provide data confidentiality, integrity, and authenticity between participating peers in a private network. Cisco provides full Encapsulating Security Payload (ESP) and Authentication Header (AH) support.

- Internet Key Exchange (IKE)

  The Internet Key Exchange (IKE) provides security association management. IKE authenticates each peer in an IPsec transaction, negotiates security policy, and handles the exchange of session keys. Cisco has been leading the standardization effort for IKE by writing IETF Internet drafts and by making a freeware version of IKE available on the Internet. For details, see the “Internet Key Exchange Security (IKE) Protocol” section on page 1-13.

- Certificate management

  Cisco supports the X509.V3 certificates for device authentication during IKE negotiation. Certificate management includes the use of the Simple Certificate Enrollment Protocol (SCEP), a protocol for communicating with Certification Authorities (CA). This certificate solution supports hierarchical certificate structures and the cross-certification necessary for a public key infrastructure (PKI) solution.

The component technologies include the following:

- Diffie-Hellman

  Diffie-Hellman is a public-key cryptography protocol that allows two parties to establish a shared secret over an unsecure communications channel. IKE uses Diffie-Hellman to establish session keys. VPN Solutions Center supports two Diffie-Hellman groups: Group 1—a MODP group with a 768-bit modulus; Group 2—a MODP group with a 1024-bit modulus.

- DES

  The Data Encryption Standard (DES) encrypts packet data.

- MD5/SHA algorithms

  The Message Digest 5/SHA hash algorithms authenticate packet data.

Using VPNSC Templates to Customize Configuration Files

The Template Manager in the VPN Solutions Center software is a provisioning system that provides fast, flexible, and extensible Cisco IOS command generation capability. The Template Manager defines standard templates to generate Cisco IOS configurations for common provisioning tasks, such as common IPv4, QoS, and VPN provisioning. For details, see Chapter 8, "Provisioning with the VPN Solutions Center Template Manager."

- A *template file* is a file created by the Template Manager that stores a VPN Solutions Center template definition.
- A *template data file* is a text file that stores variable values to generate the template file. A valid data file contains name-value pairs for all the variables defined in a template. Each template file can be associated with multiple data files; however, note that each data file can only be associated with a single template. You can select which data file to use to generate a template. The filename suffix for data files is *.dat*.
- A *template configuration file* is an IOS configuration file that stores the Cisco IOS commands created by the Template Manager. A template configuration file can be either a partial or complete configuration file. When you generate a template configuration file using a particular data file, the template configuration filename is the same as the data file's name.

The template data files are tightly linked with its corresponding template. You can use a data file and its associated template to create a template configuration file. The template configuration file is merged with (either appended to or prepended to) the VPNSC configlet. VPN Solutions Center downloads the combined configlet to the edge device router.

You can apply the same template to multiple edge devices, assigning the appropriate template data file for each device. Each template data file includes the specific data for a particular device (for example, the management IP address or host name of each device).

The template files and data files are in XML format. The template file, its data files, and all template configuration file files are mapped to a single directory.

- VPN Solutions Center creates the initial VPNSC configlet. Through the Template Manager, you can create a template configuration file. You can then associate a template configuration file with a service request, which effectively merges the VPNSC configlet and the template configuration file. For details on this process, see the "Integrating VPN Solutions Center Templates with a Service Request" section on page 4-25. You can then download this merged VPNSC configlet to the target router (or routers).
- You can also create a template configuration file and download it directly to a router as described in the "Provisioning a Template Configuration File Directly to a Router" section on page 8-22.

Uses for the Template Function

Service providers can use the Template Manager to enhance VPN Solutions Center functionality. Because the Telnet Gateway Server (TGS) supports console access to VPN Solutions Center targets, you can use the Template Manager to provide initial configuration for any service provider core device or edge device.

The Template Manager can be used as a stand-alone tool to generate complete configuration files that you can download to any VPN Solutions Center target.

Some of the additional uses for templates are as follows:

- IOS firewall provisioning
- Add a set of commands that VPN Solutions Center does not include to a service request; for example, provisioning ATM Class of Service.
- Use the template feature to apply Class of Service using IP connectivity.
- Download a VPN Solutions Center service request and an Cisco IOS configuration file in one download operation through the console. This edge device staging method would create a template and apply the service request in one step.

Security and Encryption Overview

An important characteristic of IP networks is that the network layer is entirely uniform; it is the only network layer that is uniform. As a result, any communication going through an IP network must use the IP protocol. In other network layers, different protocols operate (depending on the network's architecture and types of communication). However, eventually all communications must go through the network layer, and for all IP networks, IP is the only one protocol in that layer. Consequently, if the IP (network) layer is secure, the network is secure.

Types of Security Attacks

IP-based data is vulnerable to hackers' tampering and eavesdropping. IP's strength is that it has small, manageable packets of electronic information that can be routed quickly and easily. These chunks of information create breaks in the data stream that allow them to be transmitted efficiently through the network. However, the way IP routes these packets causes large IP networks to be vulnerable to a number of security attacks, such as *spoofing*, *sniffing*, and *session highjacking*.

Spoofing is an attack that involves one machine on a network masquerading as another. Sniffing is an attack that involves an eavesdropper listening in on communications between two other parties. Session hijacking is an attack in which a hacker uses both spoofing and sniffing to take over an established communications session and pretends to be one of the parties involved.

In each of these forms of network attack, an unauthorized individual gains access to private company information. To remedy the problem, an international group organized under the Internet Engineering Task Force (IETF) created the IPsec protocol suite, a set of IP protocols that provide security services at the network level. IPsec is based on state-of-the-art cryptographic technology that makes secure data authentication and privacy on large networks a reality.

IPsec Encryption Technologies

The IPsec protocol suite has a foundation of powerful encryption technologies. The suite adds security services to the IP layer in a way that is compatible with both the existing IPv4 standard and the emerging IPv6 standard.

Transport Mode and Tunnel Mode

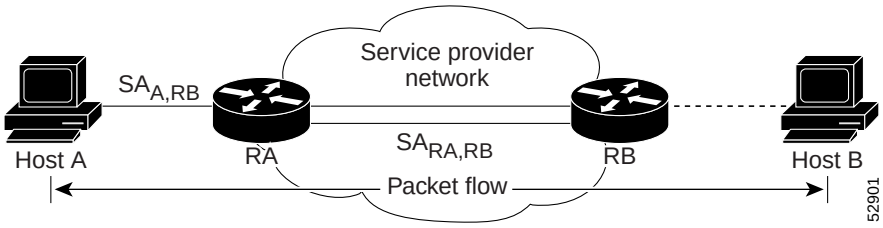
IPsec supports two encryption modes: *Transport mode* and *Tunnel mode*. *Transport mode* encrypts only the data portion (payload) of each packet and leaves the packet header untouched. Transport mode is applicable to either gateway or host implementations, and provides protection for upper layer protocols as well as selected IP header fields.

Tunnel mode is more secure than Transport mode because it encrypts both the payload and the header. IPsec in Tunnel mode is normally used when the ultimate destination of a packet is different than the security termination point. This mode is also used in cases when the security is provided by a device that did not originate packets, as in the case of VPNs.

Tunnel mode is often used in networks with unregistered IP addresses. The unregistered address can be tunneled from one gateway encryption device to another by hiding the unregistered addresses in the tunneled packet.

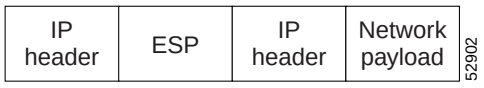
Figure 1-3 shows a typical network using IPsec in Tunnel mode:

Figure 1-3 IPsec in Tunnel Mode



In Tunnel mode, IPsec encapsulates an IP packet with IPsec headers and adds an outer IP header, as shown in Figure 1-4.

Figure 1-4 IPsec Tunnel Mode Packet Format



An IPsec Tunnel mode packet has two IP headers—an inner header and an outer header. The inner header is constructed by the host; the outer header is added by the device that is providing security services. IPsec defines Tunnel mode for both the Authentication Header (AH) and Encapsulating Security Payload (ESP).

IPsec standards define several new packet formats, such as an Authentication Header (AH) to provide data integrity and the Encapsulating Security Payload (ESP) to provide confidentiality. IPsec parameters between devices are negotiated with the Internet Key Exchange (IKE) protocol, formerly referred to as the Internet Security Association Key Management Protocol (ISAKMP/Oakley).

IKE can use digital certificates for device authentication. The Encapsulating Security Payload and the Authentication Header use cryptographic techniques to ensure data confidentiality and digital signatures that authenticate the data's source.

The IP packet is the fundamental unit of communications in IP networks. IPsec handles encryption at the packet level, and the protocol it uses is the ESP. ESP supports any type of symmetric encryption. The default standard built into ESP that assures basic interoperability is 56-bit DES.

Using IPsec to Secure the IP Layer

IPsec is a framework of open standards developed by the Internet Engineering Task Force (IETF) that provides security for transmission of sensitive information over unprotected networks such as the Internet. It acts at the network level and implements the following standards:

- IPsec
- Internet Key Exchange (IKE)
- Data Encryption Standard (DES)
- MD5 (HMAC variant)
- SHA (HMAC variant)
- Authentication Header (AH)
- Encapsulating Security Payload (ESP)

Essentially, if the IPsec suite is used where IP is normally used (in the network layer), communications are secured for all applications and for all users more transparently than would be the case if any other approach was employed.With IPsec, a service provider can create a secure VPN as needed and with any other device that is using the IPsec standard. Because IPsec works with both existing and future IP standards, regular IP networks can still be used to carry data. The sending and receiving devices must be IPsec compliant, but the rest of the network between the sender and recipient does not have to be IPsec compliant.

The primary strength of the IPsec approach is that security works at a low network level. As a result, IP is transparent to the average user, and IPsec-based security services also function behind the scenes to ensure that all network communications are secure. IPsec meets a broad range of security needs and allows different networks around the world to interconnect and to communicate securely. In addition, IPsec offers almost infinite scalability with transparent and reliable service, no matter how demanding a company's security needs.

The Encapsulating Security Payload (ESP)

The Encapsulating Security Payload (ESP) contains six parts as described below. The first two parts are not encrypted, but they are authenticated. Those parts are as follows:

- The Security Parameter Index (SPI) is an arbitrary 32-bit number that tells the device receiving the packet what group of security protocols the sender is using for communication. Those protocols include the particular algorithms and keys, and how long those keys are valid.
- The Sequence Number is a counter that is incremented by 1 each time a packet is sent to the same address and uses the same SPI. The sequence number indicates which packet is which, and how many packets have been sent with the same group of parameters. The sequence number also protects against replay attacks. Replay attacks involve an attacker who copies a packet and sends it out of sequence to confuse communicating devices.

The remaining four parts of the ESP are all encrypted during transmission across the network. Those parts are as follows:

- The *Payload Data* is the actual data that is carried by the packet.
- The *Padding*, from 0 to 255 bytes of data, allows certain types of encryption algorithms to require the data to be a multiple of a certain number of bytes. The padding also ensures that the text of a message terminates on a four-byte boundary (an architectural requirement within IP).
- The *Pad Length* field specifies how much of the payload is padding rather than data.

- The *Next Header* field, like a standard *IP Next Header* field, identifies the type of data carried and the protocol.

  The ESP is added after a standard IP header. Because the packet has a standard IP header, the network can route it with standard IP devices. As a result, IPsec is backwards-compatible with IP routers and other equipment even if that equipment isn't designed to use IPsec. ESP can support any number of encryption protocols. It's up to the user to decide which ones to use. Different protocols can be used for every person a user communicates with. However, IPsec specifies a basic DES-Cipher Block Chaining mode (CBC) cipher as the default to ensure minimal interoperability among IPsec networks. ESP's encryption capability is designed for symmetric encryption algorithms. IPsec employs asymmetric algorithms for such specialized purposes as negotiating keys for symmetric encryption.

Tunneling with ESP

Tunneling takes an original IP packet header and encapsulates it within the ESP. Then, it adds a new IP header containing the address of a gateway device to the packet. Tunneling allows a user to send illegal IP addresses through a public network (like the Internet) that otherwise would not accept them. Tunneling with ESP offers the advantage of hiding original source and destination addresses from users on the public network. Hiding these addresses reduces the power of traffic analysis attacks. A traffic analysis attack employs network monitoring techniques to determine how much data and what type of data is being communicated between two users.

The ESP Authentication Field

The ESP Authentication field contains an Integrity Check Value (ICV), which functions as a digital signature that is computed over the remaining part of the ESP. The ESP Authentication field varies in length depending on the authentication algorithm used. This field can be omitted entirely if authentication is not needed for the ESP. Authentication is calculated on the ESP packet once encryption is complete. The current IPsec standard requires HMAC (a symmetric signature scheme) with hashes SHA1 and MD5 as algorithms for IPsec-compliant hardware and software in the ESP packet's Authentication field.

The Integrity Check Value supports symmetric type authentication. The sending device encrypts a hash of the data payload and attaches it as the authentication field. The receiving device confirms that nothing has been tampered with and that the payload did come from the correct source device.

The Authentication Header (AH)

The IPsec suite's second protocol, the Authentication Header (AH), provides authentication services. The AH may be applied alone, together with the ESP, or in a nested fashion when tunnel mode is used. Authentication provided by the AH differs from what is provided in the ESP in that the ESP's authentication capabilities do not protect the IP header that lies in front of the ESP, although an encapsulated IP header in tunneling mode is protected. The AH services protect this external IP header, along with the entire contents of the ESP packet. The AH does not protect all of the fields in the external IP header because some change in transit, and the sender cannot predict how they might change. The AH protects everything that does not change in transit. In the packet, the AH is located after the IP header but before the ESP (if present) or other higher level protocol, such as TCP. Like the ESP, the AH can implement tunneling mode. Also, like the ESP, IPsec requires specific algorithms to be available for the AH to be implemented.

Security Associations (SA)

The Authentication Header and Encapsulating Security Payload protocols are the building blocks of IPsec. The encryption services provided by the AH and ESP are powerful tools for keeping data secret, for verifying its origin, and for protecting it from undetected tampering. But these tools will not work unless there is a carefully designed infrastructure to work with them. VPN security succeeds or fails depending on the reliability and scalability of this infrastructure.

Secure communication with authentication and encryption requires negotiation, an exchange of keys, and a capability to keep track of the keys. The way that IPsec keeps track of the details, as well as which keys and algorithms to use, is by bundling everything together in a *Security Association (SA)*. An association is a one-way relationship between a sender and a receiver that affords security services to the traffic carried on it. The SA groups together all the elements needed for two parties to communicate securely.

If a peer relationship is needed for two-way secure exchange, two security associations are required. Security services are afforded to an SA for the use of AH or ESP, but not both. A security association is uniquely identified by three parameters:

- *Security Parameter Index (SPI)*

  The SPI assigns a bit string to this SA that has local significance only. The SPI is carried in the AH and ESP headers to enable the receiving system to select the SA under which a received packet will be processed.

- *IP destination address*

  Currently, only unicast addresses are allowed; this is the address of the destination endpoint of the SA, which may be an end-user system or a network system, such as a firewall or router.

- *Security protocol identifier*

  This indicates whether the association is an AH or ESP security association. Hence, in any IP packet, the security association is uniquely identified by the destination address in the IPv4 or IPv6 header and the SPI in the enclosed extension header (AH or ESP).

An IPsec implementation includes a security association database that defines the parameters associated with each SA. A security association is defined by the following parameters:

- *Sequence number counter*

  A 32-bit value used to generate the sequence number field in AH or ESP headers

- *Sequence counter overflow*

  A flag indicating whether overflow of the sequence number counter should generate an auditable event and prevent further transmission of packets on this SA

- *Anti-replay window*

  Used to determine whether an inbound AH or ESP packet is a replay, by defining a sliding window within which the sequence number must fall

- *AH information*:

  Authentication algorithm, keys, key lifetimes, and related parameters being used with AH

- *ESP information*

  Encryption and authentication algorithm, keys, initialization values, key lifetimes, and related parameters being used with ESP

- *Lifetime of this security association*

  A time interval or byte count after which an SA must be replaced with a new SA (and new SPI) or terminated, plus an indication of which of these actions should occur

- *IPsec protocol mode*

  Tunnel, transport, or wildcard (required for all implementations); these modes are discussed later in this chapter (XREF)

- *Path MTU*

  Any observed path maximum transmission unit (maxi-mum size of a packet that can be transmitted without fragmentation) and aging variables (required for all implementations)

The key management mechanism that is used to distribute keys is coupled to the authentication and privacy mechanisms only by way of the security parameters index. Hence, authentication and privacy have been specified independent of any specific key management mechanism.

The SA is the secure channel through the public network. The SA also lets the system construct classes of security channels. If more secure safeguards are needed, more care can be taken, and the rules of the SA can be changed to specify stronger measures.

Internet Key Exchange Security (IKE) Protocol

Internet Key Exchange (IKE) is a protocol of choice for protocol negotiation and key exchange through the Internet. IKE enables an agreement to be negotiated on which protocols, algorithms, and keys should be used. It ensures secure authentication services from the beginning of the exchange. It manages keys securely after they have been agreed upon, and it exchanges those keys safely.

IKE provides four capabilities:

- Provides the means for parties to agree on which protocols, algorithms, and keys to use.
- Ensures from the beginning of the exchange that you are talking to the right person.
- Manages those keys after they have been agreed upon.
- Ensures that key exchanges are handled safely.

Key exchange is closely related to security association management. When a security association is created, keys must be exchanged. IKE wraps them together, and delivers them as an integrated package. IPsec specifies that compliant systems support manual keying as well. As a result, manual key exchange is possible in certain situations.

However, for most large enterprises, manual key exchange is impractical. Thus, IKE is expected to continue to negotiate SAs and exchange keys automatically through public networks. IKE functions in two phases:

- Phase 1: Two IKE peers establish a secure channel for performing ISAKMP operations.
- Phase 2: The two peers negotiate general purpose security associations.

An IKE peer is an IPsec-compliant node capable of establishing IKE channels and negotiating SAs. IKE provides three modes for the exchange of keying information and setting up IKE security associations: *Main mode*, *Aggressive mode*, and *Quick mode*.

Main Mode

Main mode provides a way to establish the first phase of an IKE SA, which is then used to negotiate future communications. The first step, securing an IKE SA, occurs in three two-way exchanges between the sender and the receiver. In the first exchange, the sender and receiver agree on basic algorithms and hashes. In the second exchange, public keys are sent for a Diffie-Hellman exchange. *Nonces* (random numbers each party must sign and return to prove their identities) are then exchanged. In the third exchange, identities are verified, and each party is assured that the exchange has been completed.

Aggressive Mode

Aggressive mode provides the same services as main mode. It establishes the phase one SA, and operates in much the same manner as main mode except that it is completed in two exchanges instead of three.

In aggressive mode, the sender generates a Diffie-Hellman pair at the beginning of the exchange, doing as much as is reasonable with the first packet (proposing an SA, passing the Diffie-Hellman public value, sending a nonce to the other party to sign, and so on). The recipient then sends back a consolidation of all three response steps that occur in main mode.

The result is that aggressive mode accomplishes as much as main mode, with one exception. Aggressive mode does not provide identity protection for communicating parties. In other words, in aggressive mode, the sender and recipient exchange identification information before they establish a secure channel where the information is encrypted. As a result, a hacker monitoring an aggressive mode exchange can determine who has just formed a new SA. Aggressive mode's value, though, is speed.

Quick Mode

After two parties have established a secure channel using either aggressive mode or main mode, they can use Quick mode. Quick mode has two purposes—to negotiate general IPsec security services and to generate newly keyed material. Quick mode is much simpler than both main and aggressive modes. Quick mode packets are always encrypted under the secure channel (or an IKE SA established in phase 1) and start with a hash payload that is used to authenticate the rest of the packet. Quick mode determines which parts of the packet are included in the hash.

Key refreshing can be done in two different ways:

- If perfect forward secrecy is not needed, Quick mode can refresh the keying material already generated in main or aggressive mode with additional hashing. The sender and recipient can then exchange nonces through the secure channel, and use them to hash the existing keys.
- If perfect forward secrecy is desired, an additional Diffie-Hellman exchange is requested through the existing SA, and the keys can be changed that way. Basic quick mode is a three-packet exchange.

Perfect Forward Secrecy

A user can reduce the risk of hackers deciphering a message through the use of larger and larger keys. But, the larger the key, the slower encryption is accomplished, and network performance also decreases. Use of fairly large keys and frequent changes of them is a good compromise. However, the challenge is coming up with ways to generate these new keys.

A method to generate a new key that does not depend on the current key is needed. Then, if a hacker knows the current key, he or she will know only a small amount of information. The hacker would have to find out an entirely unrelated key to get to the next part. This concept is called *perfect forward secrecy.* The way that perfect forward secrecy is done through IKE is called "Diffie-Hellman."

A Diffie-Hellman exchange allows two users who wish to communicate with each other to randomly generate keys that are similar to a public/private key pair. Each user sends a public key value to the other. Each then combines the public key they receive with the private key they just generated using the Diffie-Hellman combination algorithm. The resulting value is the same on both sides. No other users in the world can come up with the same key from the two public keys that traveled across the Internet, because the final key depends on each user's private key, which is secret.

The derived Diffie-Hellman key can be used either as a session key for subsequent exchanges or to encrypt another randomly generated key. Diffie-Hellman allows new shared keys, that are independent of older keys, to be generated for symmetric encryption, thus providing perfect forward secrecy. Because symmetric encryption operates quickly, Diffie-Hellman is valuable to network communications.

Certification Authority (CA)

The final component of the IPsec-compliant secure VPN is the Certification Authority (CA). Certification Authority interoperability is provided in support of the IPsec standard. It permits Cisco IOS devices and CAs to communicate so that your Cisco IOS device can obtain and use digital certificates from the CA. Although IPsec can be implemented in your network without the use of a CA, using a CA provides manageability and scalability for IPsec

While not an integral part of IPsec, the CA is, nevertheless, a critical element in the public key infrastructure. A CA is a trusted third party, an entity whose identity has already been established and proven. The CA's role is to vouch for the identities of people with whom a user is trying to communicate.

When verifying online communications, the CA software issues certificates tying together the following three elements:

- An individual's identity
- The public key the individual uses to "sign" online communications
- The CA's public key (used to sign and authenticate communications)

The CA defends against the "middle-man" hacker who attempts to work his way into key exchanges. Whenever an exchange is initiated, users sign their communications packages with their digital signatures. Those signatures are checked against the ones on record with the CA; they must match. Users then check the CA certificate's signature with the CA's signature. They have to match too. Otherwise, an SA cannot be established and no communications can take place.