Using IPSec to Construct Secure Virtual Private Networks

The Internet has become a popular, low-cost backbone infrastructure. Its universal reach has led many companies to consider constructing a secure Virtual Private Network (VPN) over the public Internet. The challenge in designing a VPN for today′s global business environment will be to exploit the public Internet backbone for both intra-company and inter-company communication while still providing the security of the traditional private, self-administered corporate network.

Traditional corporate networks were often administered by their owners, data traveled over private facilities, and very little traffic left or entered the corporate network. In such self-contained environments, these networks were generally considered to be secure. VPNs will extend the reach of the classical corporate network, exploiting the global span of the public Internet rather than relying on private backbones. However, there will be many significant challenges. No single entity owns the Internet or sets its policies. Data from many different sources will flow through its common backbone infrastructure and within its routers. As e-business proliferates, more and more data will flow between companies. This model differs radically from that of the traditional self-contained, self-administered corporate network.

Within the Internet Engineering Task Force (IETF), the IP Security (IPSec) working group has developed a framework for network layer security. IPSec protocols will support data origin authentication, data integrity, data confidentiality, key management, and management of security associations. IPSec is a flexible framework for providing network layer security. Earlier security protocols often protected a portion of an end-to-end path, or they forced you to impose the same protection everywhere along the path. IPSec provides complete end-to-end network layer security, while giving you the opportunity to tailor the security coverage on a segment-by-segment basis along any given path.

Within the IPSec framework, a company can configure secure end-to-end solutions that can accommodate both locally-attached users and remote-access users, and can support communications both within the company and between different companies. This paper outlines relevant security issues, and illustrates how IPSec can provide secure end-to-end solutions in a variety of typical e-business configurations.

CONTENTS

FIGURES

1. VPN OVERVIEW

Virtual Private Networks (VPNs) will exploit the worldwide reach of the public Internet to provide secure, cost-effective intra-company and inter-company communications. The breadth of the Internet and its growth in popularity have positioned it as the external network of choice for connecting intranets securely, both within and between companies.

Traditional corporate networks were largely self-contained. They were operated by the corporation; they used private facilities (e.g., leased lines or frame relay circuits); they carried only internal traffic; and they were generally inaccessible by outside users. In this closed, tightly controlled environment, the traditional corporate network was usually considered to be secure.

But today′s corporate networks are evolving toward a new business model: the extended corporate network is now a collection of physically separated *intranets* interconnected over the public Internet, and inter-company communication is now a business necessity. This new business model presents security exposures that were not present the traditional corporate network model. Designers and operators of Virtual Private Networks must understand these exposures and protect against them. A well thought out company-wide *VPN Security Policy* will be critical to assure that your VPN will interconnect a set of intranets—both within a single company and between companies—with security as good as, or better than, that of traditional self-contained, self-administered corporate networks.

The following sections will briefly review the main features of the IPSec protocols, will discuss the non-IPSec components that can play a part in a deployed VPN, will outline the typical end-to-end communications path for each of three example scenarios, will consider the security exposures, and will then outline a secure IPSec-based solution for each of them. To make the discussion more concrete, we will examine three basic scenarios that cover common business uses for VPNs:

- **Branch Office Interconnection:** a VPN that enables communications between physically separated intranets that are members of a single corporate network,
- **Inter-company Connections:** a VPN that enables secure communications between intranets of different companies, using the public Internet as a backbone
- **Remote Access:** a VPN that enables secure communications between a remote host and its home corporate network

2. AN OVERVIEW OF IPSEC

The IPSec Working Group of the IETF has defined an open architecture and an open framework, known as ″IPSec″. IPSec is called a framework because it provides a stable, long lasting base for providing network layer security. It can accommodate today′s cryptographic algorithms, and can also accommodate newer, more powerful algorithms as they become available. IPv6 implementations are required to support IPSec, and IPv4 implementations are strongly recommended to do so. In addition to providing the base security functions for the Internet, IPSec furnishes flexible building blocks from which robust, secure Virtual Private Networks can be constructed.

The IPSec Working Group has concentrated on defining protocols to address several major areas:

- *Data Origin Authentication* verifies that each datagram was originated by the claimed sender
- *Data integrity* verifies that the contents of the datagram were not changed in transit, either deliberately or due to random errors
- *Data confidentiality* conceals the cleartext of a message, typically by using encryption
- *Replay protection* assures that an attacker can not intercept a datagram and play it back at some later time
- *Automated management of cryptographic keys and security associations* assures that a company′s VPN Policy can be conveniently and accurately implemented throughout the extended network with little or no manual configuration. These functions make it possible for a VPN′s size to be scaled to whatever size a business requires.

The principal IPSec protocols are:

- IP Authentication Header (AH) provides data origin authentication, data integrity, and replay protection
- IP Encapsulating Security Payload (ESP)provides data confidentiality, data origin authentication, data integrity, and replay protection
- Internet Security Association and Key Management Protocol (ISAKMP) provides a method for automatically setting up security associations and managing their cryptographic keys.

Within the layered communications stack model, the Network layer is the lowest layer that can provide end-to-end security. Network-layer security protocols provide blanket protection for all upper-layer application data carried in the payload of an IP datagram, without requiring a user to modify the applications.

2.1 AUTHENTICATION HEADER (AH)

The IP Authentication Header provides connectionless (that is, per-packet) integrity and data origin authentication for IP datagrams, and also offers protection against replay. Data integrity is assured by the ″checksum″ generated by a message authentication code (for example, MD5); data origin authentication is assured by including a secret shared key in the data to be authenticated; and replay protection is provided by use of a sequence number field within the AH Header. In the IPSec vocabulary, these three distinct functions are lumped together and simply referred to by the name *authentication*.

The algorithms used by the AH protocol are known as *hashed message authentication codes (HMAC)*. HMAC applies a conventional keyed message authentication code two times in succession, as shown in schematic form in Figure 1: first to a secret key and the data, and then to the secret key and the output of the first round. Since the underlying message authentication code in Figure 1 is MD5, this algorithm would be referred to as *HMAC-MD5*. The AH protocol also supports the use of HMAC-SHA. The mechanics are the same, but in this case, the Secure Hash Algorithm (SHA) is used as the base message authentication code rather than MD5.

AH protects the entire contents of an IP datagram except for certain fields in the IP header (called ″mutable fields″) that could normally be modified while the datagram is in transit[1]. For purposes of calculating an integrity check value, the mutable fields are treated as if they contained all zeros. The integrity check value is carried in the ″AH Header″ field shown in Figure 2.

AH can be applied in either of two modes: *transport mode* or *tunnel mode* (see Figure 2). Figure 2 shows how the AH protocol operates on an original IP datagram in each of these two modes.

- In transport mode, the original datagram′s IP header is the outermost IP header, followed by the AH header, and then the payload of the original IP datagram. The entire original datagram, as well as the AH Header itself, is authenticated, and any change to any field (except for the mutable fields) can be detected. Note that all information in the datagram is in cleartext form, and therefore is subject to eavesdropping while it is in transit.
- In tunnel mode, a new IP header is generated for use as the outer IP header of the resultant datagram. The source and destination address of the new header will generally differ from those used in the original header[2]. The new header is

[1] For example, the Time to Live field must decremented by each router along a path, so it can not be covered by the AH integrity check value.

[2] Since routing protocols operate only on the outermost IP header, a common case is to have the destination address of the new (outer) header point to an intermediate box, such as a firewall, where the AH processing will take place.

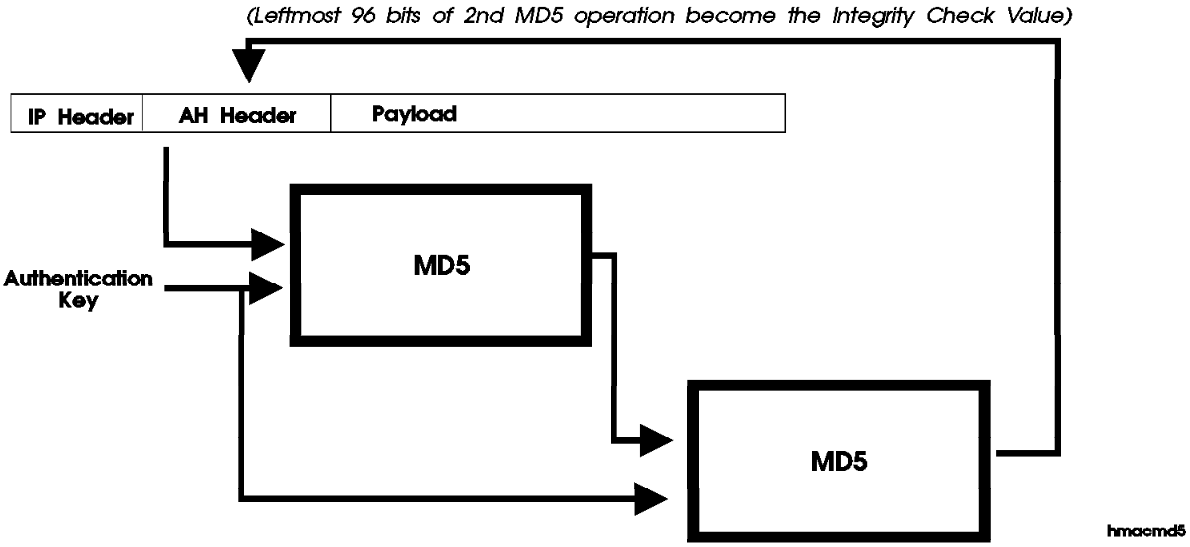

(Leftmost 96 bits of 2nd MD5 operation become the Integrity Check Value)
IP Header
AH Header
Payload
Authentication Key
MD5
MD5
hmacmd5

Figure 1. Hashed Message Authentication Codes. *HMAC applies a conventional Message Authentication Code twice in succession.*

then followed by the AH header, and then by the original datagrams in its entirety, both its IP header and the original payload. The entire datagram (new IP Header, AH Header, IP Header, and IP Payload) is protected by the AH protocol. Any change to any field (except the mutable fields) in the tunnel mode datagram can be detected. Note that all information in the datagram is in cleartext form, and therefore is subject to eavesdropping while it is in transit.

AH may be applied alone, in combination with ESP, or even nested within another instance of itself. With these combinations, authentication can be provided between a pair of communicating hosts, between a pair of communicating firewalls, or between a host and a firewall.

2.2 ENCAPSULATING SECURITY PAYLOAD (ESP)

The IP Encapsulating Security Payload provides data confidentiality (encryption), connectionless (that is per-packet) integrity, data origin authentication, and protection against replay. ESP always provides data confidentiality, and can also optionally provide data origin authentication, data integrity checking, and replay protection. Comparing ESP to AH, one sees that only ESP provides encryption, while either can provide authentication, integrity checking, and replay protection.

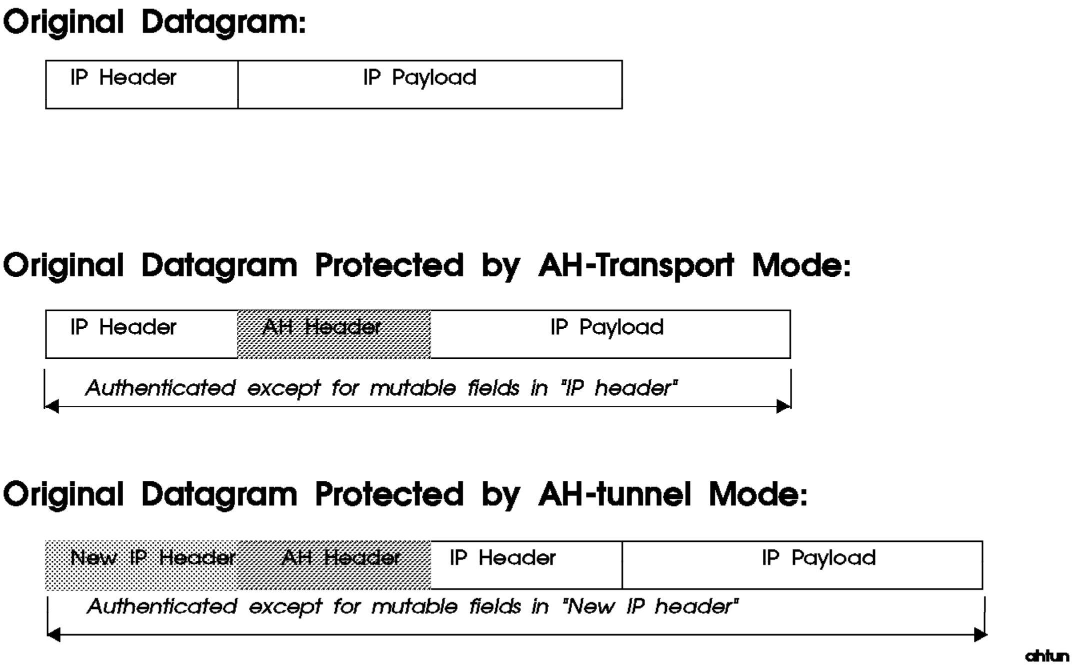

Original Datagram:
IP Header
IP Payload
Original Datagram Protected by AH-Transport Mode:
IP Header
AH Header
IP Payload
Authenticated except for mutable fields in "IP header"
Original Datagram Protected by AH-tunnel Mode:
New IP Header
AH Header
IP Header
IP Payload
Authenticated except for mutable fields in "New IP header"
ahtun

Figure 2. AH Tunnel and Transport Modes. *The AH protocol can be used in two modes: tunnel and transport.*

ESP′s encryption uses a symmetric shared key: that is, a shared key is used by both parties for encrypting and decrypting the data that is exchanged between them.

When ESP is used to provide authentication functions, it uses the same HMAC algorithms (HMAC-MD5 or HMAC-SHA) as are used by the AH protocol. However, the coverage is different, as shown in Figure 3:

- In transport mode, ESP′s authentication functions protect only the original IP payload, but not the original IP header. (Recall that in transport mode, AH protected both the original IP Header and the IP Payload.)
- In tunnel mode, ESP′s authentication protects the original IP Header and the IP Payload, but not the New IP Header. (Recall that in tunnel mode, AH protected the New IP Header, the original IP Header, and the IP Payload.)

ESP can be applied in either of two modes: *transport mode* or *tunnel mode* (see Figure 3):

- In transport mode, the datagram′s original IP header is retained. Only the payload of the original IP datagram and the ESP Trailer are encrypted. Note that the IP Header itself is neither authenticated nor encrypted. Hence, the addressing information in the outer header is visible to an attacker while the datagram is in transit.

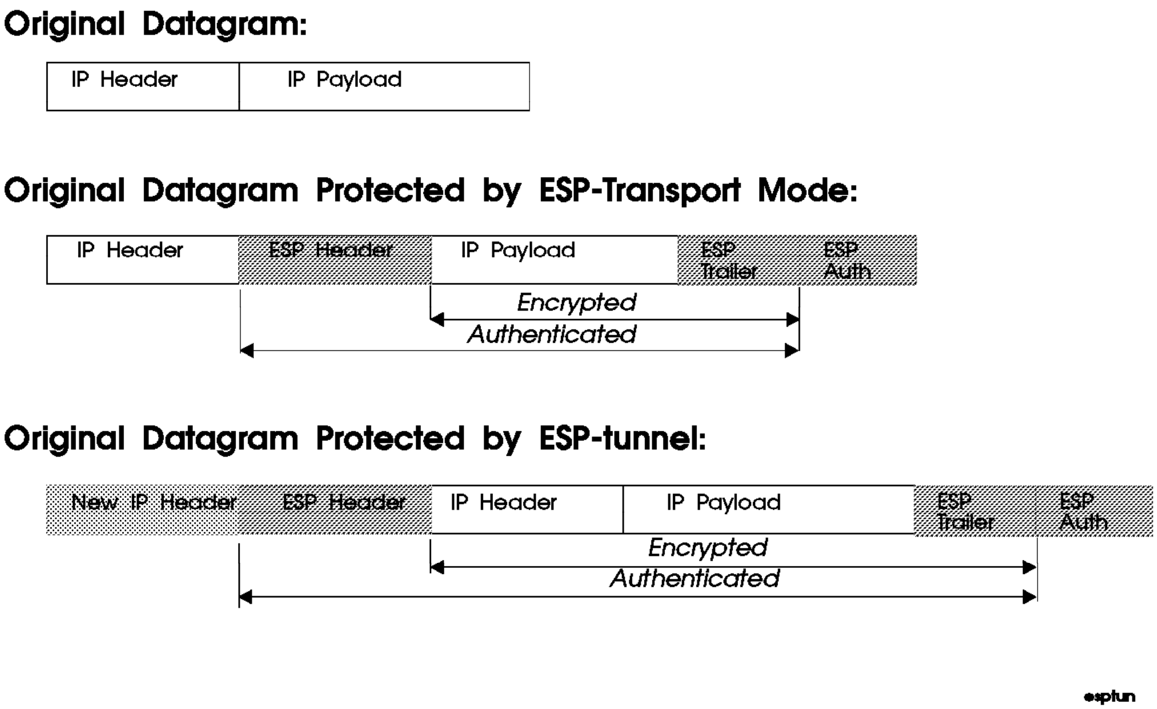


Figure 3. ESP Tunnel and Transport Modes. *Transport mode provides confidentiality for the payload of the original datagram, while tunnel mode provides confidentiality for both the header and the payload.*

- In tunnel mode, a new IP header is generated. The entire original IP datagram (both IP Header and IP Payload) and the ESP Trailer are encrypted. Because the original IP Header is encrypted, its contents are not visible to an attacker while it is in transit. A common use of ESP tunnel mode, therefore, is to hide internal address information while a datagram is ″tunneled″ between two firewalls, as described later in more detail for several scenarios in 6.

ESP may be applied alone, in combination with AH, or even nested within another instance of itself. With these combinations, authentication can be provided between a pair of communicating hosts, between a pair of communicating firewalls, or between a host and a firewall.

2.3 USE OF TRANSPORT AND TUNNEL MODES

We have seen that AH and ESP can be used in both transport mode or tunnel mode. IPSec′s tunnel mode is an encapsulation technique modeled after ″IP Encapsulation within IP″ (RFC 2003). The key points to keep in mind are:

- Transport mode is normally used between the end points of a connection. For example, if secure communication is desired along all elements of a path from a client to a server, the client and server would use IPSec′s transport mode.

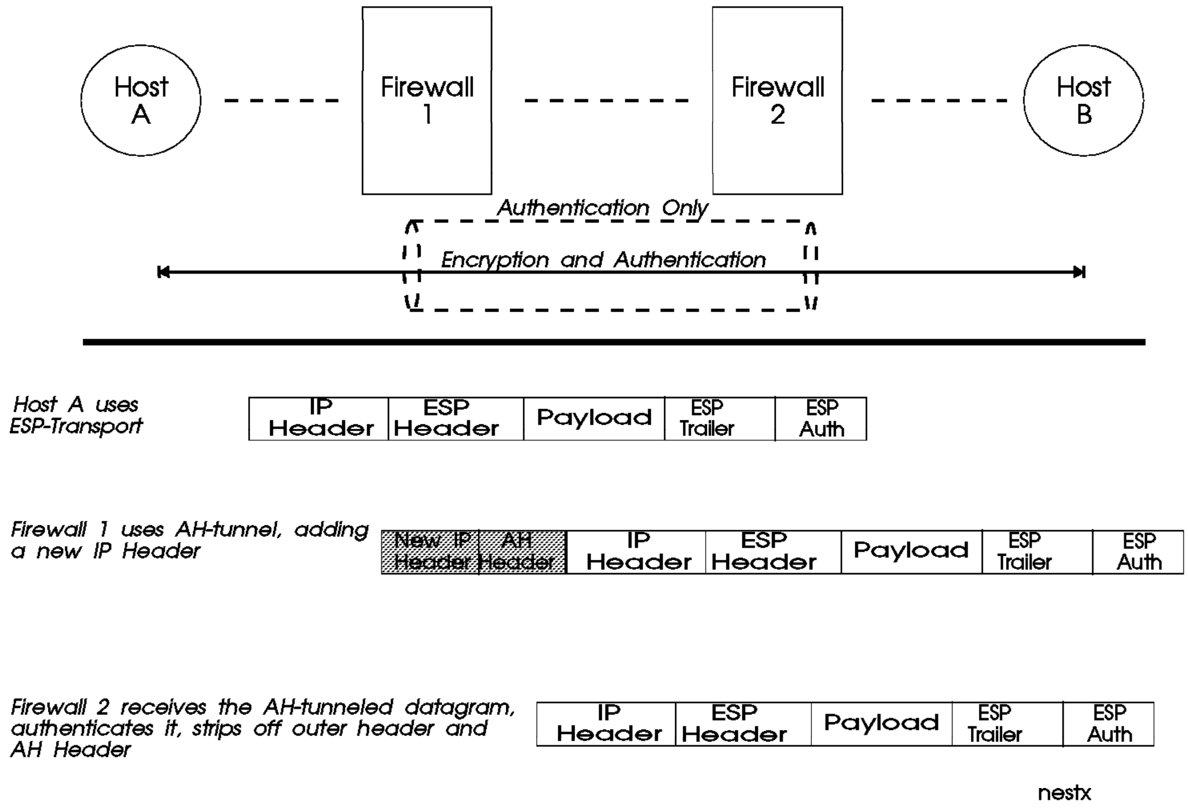


Figure 4. Nesting of IPSec Protocols. *Tunnel mode allows one IPSec protocol to be nested inside another.*

- Tunnel mode is normally used between two machine when at least one of the machines is not an end point of the connection. For example, if secure communication is desired between two firewalls that are located between a client and a server, the firewalls would use IPSec′s tunnel mode between themselves. Or if a remote host dialed in to its home network, it may want a secure path between itself and an entry gateway at its home network. Again, the remote host and the entry gateway would use IPSec′s tunnel mode in this situation.

Finally, we will see in the scenarios discussed in 6 that there are cases where it is desirable to use IPSec′s transport and tunnel modes simultaneously—a capability called ″nesting″ or ″bundling″. For example, a path between a client and a server might pass through two firewalls. In this case, the client and server would use IPSec′s transport mode, while the two firewalls would use IPSec′s tunnel mode. Between the firewalls, both modes would be active. For illustrative purposes, Figure 4 shows an example of how a composite datagram would be constructed when ESP is used between end points and AH is used between firewalls:

1. Host A uses ESP transport mode between itself and Host B.
2. When the datagram created by Host A arrives at firewall 1, firewall 1 then applies AH tunnel mode to this datagram. That is, it adds a New IP Header, specifying itself as the source and firewall 2 as the destination, and it adds the AH Header.
3. Since the outer header now has a destination of firewall 2, firewall 2 will process the incoming datagram. It will authenticate the inbound datagram, remove the outer header, and then remove the AH Header. At the end of this process, the original datagram launched by Host A will have been recovered.
4. The original datagram will then be routed onward to Host B, who will process the ESP protocol to extract the underlying cleartext payload.

Notice that while traveling between the two firewalls, the original datagram was encapsulated: that is, it was carried inside the payload of the datagram with the new outer header. Different names are often given to this process: it is ccalled *tunneling*, *nesting*, or *encapsulation*. All these terms are equivalent—they simply mean that an original IP datagram is being carried in the payload of another IP datagram. In theory, encapsulation can be applied iteratively, leading to nestings that are many levels deep. In practice, IPSec protocols require support for only two levels of nesting.

2.4 ISAKMP/OAKLEY

A Security Association (SA) contains all the relevant information that communicating systems need in order to execute the IPSec protocols, such as AH or ESP. For example, a Security Association will identify the cryptographic algorithm to be used, the keying information, the identities of the participating parties, etc. ISAKMP defines a standardized framework to support negotiation of Security Associations (SA), initial generation of all cryptographic keys, and subsequent refresh of these keys. Oakley is the mandatory key management protocol that is required to be used within the ISAKMP framework. ISAKMP supports automated negotiation of security associations, and automated generation and refresh of cryptographic keys. The ability to perform these functions with little or no manual configuration of machines will be a critical element as a VPN grows in size.

Secure exchange of keys is the most critical factor in establishing a secure communications environment—no matter how strong your authentication and encryption are, they are worthless if your key is compromised. Since the ISAKMP procedures deal with initializing the keys, they must be capable of running over links *where no security can be assumed to exist*—that is, they are used to ″bootstrap″ the IPSec protocols. Hence, the ISAKMP protocols use the most complex and processor-intensive operations in the IPSec protocol suite.

ISAKMP requires that all information exchanges must be both encrypted and authenticated: no one can eavesdrop on the keying material, and the keying material will be exchanged only among authenticated parties. In addition, the ISAKMP methods have been designed with the explicit goals of providing protection against several well-known exposures:

- Denial of Service: the messages are constructed with unique ″cookies″ that can be used to quickly identify and reject invalid messages without the need to execute processor-intensive cryptographic operations
- Man-in-the-Middle: protection is provided against the common attacks such as deletion of messages, modification of messages, reflecting messages back to the sender, replaying of old messages, and redirection of messages to unintended recipients
- Perfect Forward Secrecy: compromise of past keys provides no useful clues for breaking any other key, whether it occurred before or after the compromised key. That is, each refreshed key will be derived without any dependence on predecessor keys.

2.4.1 ISAKMP′s Two Phases

The robustness of any cryptography-based solution depends much more strongly on keeping the keys secret than it does on the actual details of the chosen cryptographic algorithms. Hence, the IPSec Working Group has prescribed a set of extremely robust ISAKMP/Oakley exchange protocols. It uses a 2-phase approach. In Phase 1, the cryptographic operations are the most processor-intensive, since this phase is designed to exchange a ″master secret″ securely, even in the case when there is absolutely no security protection in place as yet between the two machines. In contrast, Phase 2 exchanges are less complex, since they are used only after the security protection suite negotiated in Phase 1 has been activated.

In summary:

1. **Phase I:** This set of negotiation establishes a ″master secret″ from which all cryptographic keys will subsequently be derived for protecting the users′ data traffic. In the most general case, public key cryptography is used to establish an ISAKMP security association between systems, and to establish the keys that will be used to protect the ISAKMP messages that will flow in the subsequent Phase 2 negotiations. Phase 1 is concerned only with establishing the protection suite for the ISAKMP messages themselves; but it does not establish any security associations or keys for protecting user data.

   Phase 1 operations need only be done infrequently, and a single Phase 1 exchange can be used to support multiple subsequent Phase 2 exchanges. As a rule of thumb, Phase 1 negotiations are executed once a day or maybe once a week, while Phase 2 negotiations are executed once every few minutes.

2. Phase II: a set of communicating systems negotiate the security associations and keys that will protect user data exchanges. Phase 2 ISAKMP messages are protected by the ISAKMP security association generated in Phase I. Phase 2 negotiations generally occur more frequently than Phase I: for example, a typical application of a Phase 2 negotiation is to refresh the cryptographic keys once every two to three minutes.

An illustration of the use of ISAKMP/Oakley to initially establish security associations and exchange keys between two systems is given in Appendix A.

3. NON-IPSEC COMPONENTS OF A VPN SOLUTION

The IPSec protocol suite provides security for the network layer of the protocol stack: that is, it provides security functions between pairs of machines that have verifiable network layer identities (e.g., IP address, fully qualified domain names, etc.). However, IPSec does not work in isolation: other protocols and functions can be employed to complement IPSec′s security functions, for example by providing finer granularity over the material to be protected. For example, efficient Certificate Management functions can make IPSec easier to deploy, and upper layer security functions, such as SSL (Secure Sockets Layer) can provide application level security in addition to IPSec′s network layer security. Or a centralized VPN Policy Directory that can be accessed with a protocol like LDAP (Lightweight Directory Access Protocol) can make it easier to configure systems correctly without tedious manual operations.

The following sections explain some complementary services and protocols as they relate to IPSec. Designers of VPNs should keep in mind that these techniques are complementary to IPSec, and in many cases can be used in conjunction with IPSec to provide very fine-grained protection of applications in cases where it is needed.

3.1 NETWORK ADDRESS TRANSLATION

Sometimes globally unique IP addresses are a scarce resource: for example, in Europe, it is especially hard to obtain a globally unique IPv4 address. Other times, a company simply wishes to keep secret the IP addresses of the machines in its intranet (an ″unlisted address″, similar in concept to an unlisted phone number). Both of these situations can be addressed with a function called Network Address Translation (NAT).

Network Address Translation (NAT) is usually implemented in a machine that resides at the boundary of a company′s intranet, at the point where there is a link to the public Internet. NAT sets up and maintains a mapping between internal IP addresses and external public (globally unique) IP addresses. Since the internal addresses are not advertised outside of the intranet, NAT can be used when they are private (globally ambiguous) addresses, or when they are public (globally unique) addresses that a company wishes to keep secret.

The weakness of NAT is that by definition the NAT-enabled machine will change some or all of the address information in an IP packet. When IPSec authentication is used, a packet whose address has been changed will always fail its integrity check under the AH protocol, since any change to any bit in the datagram will invalidate the Integrity Check Value that was generated by the source.

Within the IETF, there is a working group that is looking at the deployment issues surrounding NAT. This group has been advised by the Internet Engineering Steering Group (IESG) that the IETF will not endorse any deployment of NAT that would lead to less security that can be obtained when NAT is not used. Since NAT makes it impossible to authenticate a packet using IPSec′s AH protocol, NAT should be considered as a temporary measure at best, but should not be pursued as a long term solution to the addressing problem.

IPSec protocols offer some solutions to the addressing issues that were previously handled with NAT. We will see in later scenarios that there is no need for NAT when all the hosts that comprise a given Virtual Private Network use globally unique (public) IP addresses: address hiding can be achieved by IPSec′s tunnel mode. If a company uses private addresses within its intranet, IPSec′s tunnel mode can keep them from ever appearing in cleartext form in the public Internet.

3.2 PACKET FILTERING

Packet filtering is a technique that is commonly provided in many firewall products and in many routers. Packet filtering relies on having access to cleartext—that is, the contents of the IP datagram can not be encrypted or compressed. The machine examines the contents of an IP packet—typically the IP header and the TCP header, and sometimes even the contents of the TCP payload—looking for things like source addresses, destination addresses, protocol IDs, port numbers, etc. The firewall then applies a set of detailed filtering rules to this information to make a decision on whether to accept or reject the packet. There are various degrees of complexity in filtering: *stateless inspection* makes a decision on each packet individually, while *stateful inspection* makes a decision for a given packet based on both the packet itself and its history. For example, *history* for the TCP protocol could involve monitoring whether or not the TCP handshake messages occur in the correct order within an acceptable time interval.

The advantage of packet filtering is that it provides excellent granularity for making access control decisions. But this is also one of its weaknesses, since the granularity can only be achieved through the specification of elaborate, detailed filtering rules. Rules development tends to be a tedious, error-prone process. And, even if a robust set of rules is in place, they are vulnerable to relatively crude ″spoofing attacks″. As new attacks are discovered, firewall administrators end up on a treadmill: each attack must be countered with very specific new rules, but these new rules don′t offer protection against the next new attack.

The major drawback to packet filtering techniques is that they require access to cleartext, both in packet headers (for stateless inspection) and in the packet payloads (for stateful inspection). When encryption is applied, some or all of the information needed by the packet filters may no longer be available. For example,

- In transport mode, ESP will encrypt the payload of the IP datagram, thus precluding the use of stateful inspection techniques.
- In tunnel mode, ESP will encrypt the entire original datagram, both header and payload, thus precluding stateless or stateful inspection of the original datagram.

IPSec offers a way off the treadmill. Its AH protocol offers a cryptographically robust and spoof-proof way to enforce access control, and its HMAC algorithms are robust enough that they can not be broken by most hackers. The processor power and the time needed to break them are both prohibitively expensive.

In most IPSec-based VPNs, packet filtering will no longer be the principal method for enforcing access control. IPSec′s AH protocol, which is cryptographically robust, will fill that role. Both the number and the complexity of filtering rules will be greatly reduced, and they will be used for fine-tuning only after a packet has already been successfully authenticated by IPSec. And since IPSec′s authentication and encryption protocols can be applied simultaneously to a given packet, strong access control can be enforced even when the data itself is encrypted.

3.3 LAYER 2 TUNNELS

A remote access dial-up solution for mobile users is a very simple form of a Virtual Private Network, typically used to support dial-in access to a corporate network whose users are all company employees. To eliminate the long-distance charges that would occur if a remote user were to dial-in directly to a gateway on the home network, the IETF developed a tunneling protocol, Layer 2 Tunnel Protocol (L2TP). This protocol extends the span of the PPP connection: instead of beginning at the remote host and ending at a local ISP′s point of presence, the ″virtual PPP″ link now extends from the remote host all the way back to the corporate gateway: in effect, the remote host appears to be on the same subnet as the corporate gateway. Since the host and the gateway share the same PPP connection, they can take advantage of PPP′s ability to transport protocols other than just IP. For example, L2TP tunnels can be used to support remote LAN access as well as remote IP access.

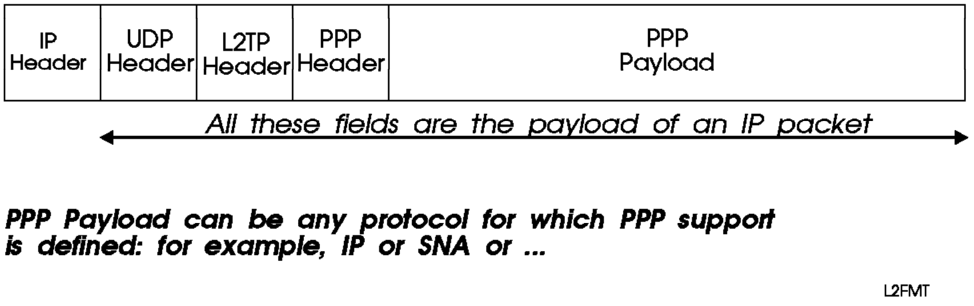

IP Header
UDP Header
L2TP Header
PPP Header
PPP Payload
All these fields are the payload of an IP packet
PPP Payload can be any protocol for which PPP support is defined: for example, IP or SNA or ...
L2FMT

Figure 5. IP Encapsulation of L2TP. *The packets of the Layer 2 Tunneling Protocol (L2TP), including PPP and its payload protocol, are encapsulated inside an IP packet for transmission across the Internet. Thus, IPSec can be used to protect them.*

Although L2TP provides cost-effective access, multi-protocol transport, and remote LAN access, it does not provide cryptographically robust security features. For example,

- Authentication is provided only for the identity of tunnel end points, but not for each individual packet that flows inside the tunnel.
- L2TP itself provides no facility to encrypt user data traffic.
- While the payload of the PPP packets can be encrypted, the PPP protocol suite does not provide mechanisms for automatic key generation or for automatic key refresh.

Realizing these shortcomings, the PPP Extensions Working Group of the IETF recently initiated a series of internet drafts to remedy these shortfalls. These drafts proposed to develop new IPSec-like protocols for use with PPP and L2TP. But since this work would have substantially duplicated the more mature work of the IPSec Working Group, the IETF took the position instead to support the use of the existing IPSec protocols to protect the data that flows through an L2TP tunnel. Rather than defining new methods, the most recent series of internet drafts simply outlines the best way to use IPSec in the various situations where an L2TP tunnel is likely to occur.

L2TP is actually another variation of an IP encapsulation protocol. As shown in Figure 5, an L2TP tunnel is created by encapsulating an L2TP frame inside a UDP packet, which in turn is encapsulated inside an IP packet whose source and destination addresses define the tunnel′s end points. Since the outer encapsulating protocol is IP, clearly IPSec protocols can be applied to this composite IP packet, thus protecting the data that flows within the L2TP tunnel. AH, ESP, and ISAKMP protocols can all be applied in a straightforward way.

In summary, layer 2 tunnel protocols are an excellent way of providing cost effective remote access; and when used in conjunction with IPSec, they are an excellent technique for providing *secure* remote access. However, without complementary use of IPSec, an L2TP tunnel alone does not furnish adequate security for the solutions that we discuss later in this paper. But in section 6.2, we will look at examples of how IPSec and L2TP can be used together to provide secure solutions.

3.4 APPLICATION GATEWAYS

Many firewalls provide *Application Gateways*. This technique requires the firewall to be aware of those applications that it will permit to flow across the boundary of a corporate intranet. The user connects to the firewall, which terminates the application. Then, the firewall launches another copy of the same application, running it between itself and the external destination. The firewall then provides synchronization between the internal application (user-to-firewall) and the external application (firewall-to-destination).

Neither network-layer nor application-layer security techniques is the best choice for all situations. There will be tradeoffs. Network-layer security protects the information created by upper-layer protocols, but it requires that IPSec be implemented in the communications stack. With network-layer security, there is no need to modify existing upper-layer applications. On the other hand, if security features are already imbedded within a given application, then the data *for that specific application* will be protected while it is in transit, even in the absence of network-layer security: that is, security functions must be imbedded on a per-application basis.

There are still other considerations:

- Network-layer security gives ″blanket protection″, but this may not be as fine-grained as would be desired for a given application.
- Network-layer security does not provide protection once the datagram has arrived at its destination host: that is, it is vulnerable to attack within the upper layers of the protocol stack at the destination machine.
- Application-layer security can protect the information that has been generated within the upper layers of the stack, but it offers no protection against several common network-layer attacks while the datagram is in transit. For example, a datagram in transit would be vulnerable to ″spoofing attacks″ against its source or destination address.

Many cases can occur, each of which needs to be examined on its own merit. It may be desirable to employ a mix of both network-layer security techniques and application-layer techniques to achieve the desired overall level of protection. For example, you cpould use an upper layer mechanism such as Secure Sockets Layer (SSL) to encrypt upper layer data. SSL could then be supplemented with IPSec′s

AH protocol at the network layer to provide per-packet data origin authentication and protection against spoofing attacks.

3.5 SECURE SOCKETS LAYER

Secure Socket Layer (SSL) is an upper-layer mechanism commonly used by Web browser clients and servers to provide peer authentication and encryption of application data. SSL mandates that the server authenticate itself to the client via a certificate-based technique. Authentication of client to the server is optional in SSL version 3, but is not commonly used in practice. SSL involves a handshake phase, where certificates are exchanged, session keys are generated, and encryption algorithms are agreed to. After the handshake phase, user data will be exchanged securely without the need for the application to be explicitly modified, other than to invoke the SSL services before actual data transfer begins. SSL is an end-to-end protocol, and therefore will be implemented in the machines at the end points of a given path (typically the client and the server), but it is not implemented in the intermediate machines along a given path (such as routers or firewalls).

3.6 SECURE MAIL

S-MIME (Secure Multipurpose Internet Mail Extension) can be thought of as a very specific SSL-like protocol: that is, S-MIME is an application-level security construct, but its use is limited to protecting e-mail via encryption and digital signatures. It relies on public key technology, and uses X.509 certificates to establish the identifies of the communicating parties.

S-MIME will be implemented in the communicating end systems; it is not used by intermediate routers or firewalls.

3.7 CERTIFICATE MANAGEMENT

Almost all of the currently popular security protocols begin by using public key cryptography to support a handshake that generates one or more master secrets. The master secrets are then used to generate the algorithm-specific keys needed by the authentication and encryption algorithms that will protect the user′s data. Since an authenticated exchange of master key information is the base on which all further security protocols depend, there is a need for supplementary protocols and procedures to issue, distribute, and verify the *certificates* that will bind each user to its public key. Public-key cryptographic operations are generally much more computationally intensive than symmetric key operations, but they offer tremendous advantages of scaling in large networks:

- When shared secrets are used, a box must *securely* store a secret key for each user that it wishes to communicate with. When public key cryptography is used, a given box needs to keep secret only a single key—its own private key. All other keys are public knowledge, and they do not need to be protected while they reside in local storage.

- Since public keys are freely available to the general public, there is no need to between communicating peers. Instead public keys can be retrieved and locally cached when needed, as a local implementation option. And unlike shared secrets, public keys need not be encrypted—after all, the keys are public information.

Within the IPSec protocols, public key certificates will be used only in ISAKMP Phase 1 for the authentication of the initial handshakes that set up a shared master key between the communicating entities. It is in this stage of the negotiations that the certificates offered by one host to the other should be verified.

The certificate format will be X.500, and certificates must be available to any host that needs them, preferably in a *Public Key Database* (such as Seecure DNS) that is accessible via a simple protocol such as LDAP (Lightweight Directory Access Protocol). The database must be authoritative, and it must be possible for the requesting hosts to verify its authenticity. For example, the directory itself could have a well-known public key which can be used by a host when it makes inquiries to the directory.

There must also be a *Certificate Authority Server* from which a host or a firewall can obtain an authoritative certificate. In fact, there may be a hierarchy of Certificate Authorities within a company (for example, a campus authority, a national authority, an international authority, etc.), and there will be a chain of trust among them. The intra-company CAs in turn will need to be recognized as authoritative by external public CAs.

3.8 SECURITY POLICY SERVER

A VPN is comprised of a collection of client hosts, server hosts, firewalls, and routers. IPSec protocols allow the level of protection to be customized on a segment-by-segment basis along an end-to-end path through the nesting of security protocols (for example, AH-transport inside ESP-tunnel, ESP-transport inside ESP-tunnel, etc.). Sometimes an intranet may be considered to be a ″trusted″ environment (e.g., in the Branch Office scenario), but at other times may be considered to be ″untrusted″ (e.g., in the Supplier Network scenario). Sometimes an intranet may be trusted the company that owns it, but will be be untrusted by an business partner that needs to access a server inside inside it. Business needs dictate the details of a VPN configuration, so the number of IPSec-compliant variations is immense.

Hence, there will be a need to centralize a company′s *VPN Policy Database (VPN-DB)*. The VPN-DB will describe the characteristics of the IPSec protocols that will be used to create the company′s VPNs. These descriptions will be in a standardized format (called a *schema*) that can be used by the Lightweight Directory Access Protocol (LDAP). The standardized representation guarantees that IPSec-capable boxes from different vendors will all interpret the information identically.

A typical VPN Policy Database will contain information about: which algorithms are to be used; what the key refresh times should be; which path segments will use transport mode and which will use tunnel mode; how security associations will be bundled (nested); what auditing and logging procedures will be used; etc. The VPN Policy Database will also provide a checking function to validate that the policies are consistent across all members of the VPN.

The information in the VPN Policy Database will be stored in an easily accessible place, such as a Secure DNS server. From there, it can be either distributed automatically to the individual IPSec-capable boxes, or an IPSec-capable box can query the database for its information using the LDAP protocol. Once it learns the VPN policy information, an IPSec-capable box will then be able tostart the ISAKMP/Oakley negotiations for setting up the necessary security associations with its peers.

4. A TYPICAL END-TO-END PATH

To understand how IPSec can be used to provide end-to-end security, we will look at the elements along an end-to-end path. While not all the elements may appear in a given path, some of them will appear in each VPN configuration that we will examine in this paper. As shown in Figure 6, a path might include a first-hop dial-in connection to an Internet Service Provider (ISP), who in turn uses the backbone public Internet to carry the user′s traffic back to a gateway at the perimeter of the corporate network. Then, the traffic eventually flows within an intranet to its ultimate destination. As we also see in Figure 6, inter-company communication can create a path that includes two separate intranets (for example, Company A′s and Company B′s).

For discussion purposes in this paper, we will refer to these elements as outlined below:

- **Dial-in Segment:** In today′s environment, remote access has become a necessity. Both work-at-home and on-the-road employees want convenient and secure dial-in access to their company′s networks; and sometimes they even need to communicate with hosts located inside another company′s network. We will refer to both work-at-home and on-the-road users as *remote users*. This segment extends from a remote user′s machine to an access box provided by the ISP. The protocols and procedures used on this link are specified by the

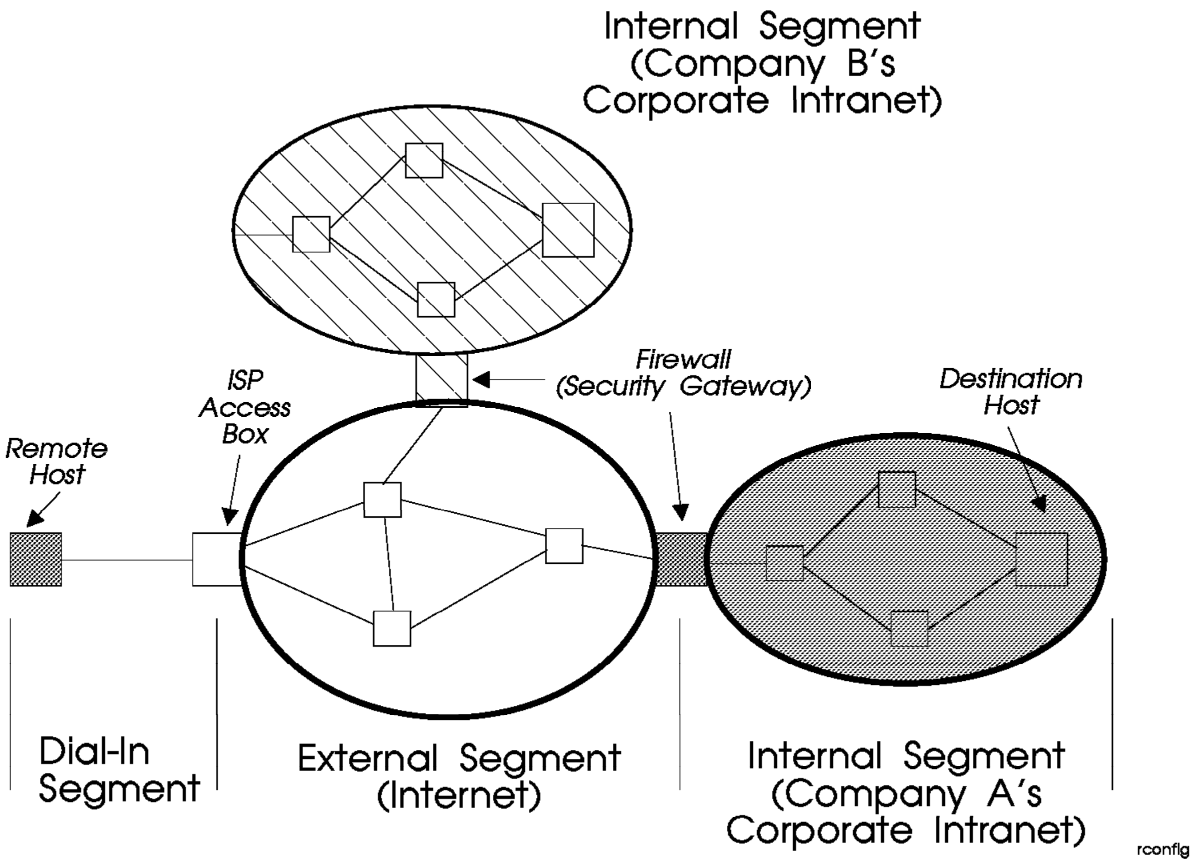


Figure 6. Typical Elements in an End-to-End Path. *End-to-end traffic will usually flow over some mix of three basic segments: a dial-in segment, an external segment (Internet), and an internal segment (Intranet).*

Internet Service Provider. Today, most ISPs support the PPP (Point-to-Point Protocol) suite of protocols on this segment.

- **External Network (Internet):** The public Internet is becoming a popular backbone for interconnecting *intranets*. Internet Service Providers (ISPs) today often offer cheaper rates for ″as needed″ access to the Internet backbone than they offer for dedicated leased lines. The Internet is not owned or operated by any single entity, but is a collection of distinct routing domains, each operated by a different authority. The unifying factor is the standardized IP communications protocols defined by the IETF (Internet Engineering Task Force). The IP (Internet Protocol) suite of protocols will route data traffic at the network layer over a path that may span several ISPs′ routing domains. Since IP is a connectionless technology, each user datagram could potentially follow a different path. And in fact, traffic from several different companies could all flow simultaneously through a given backbone router in the Internet. For example, a datagram that originated in Company A′s intranet and a datagram that originated in Company B′s intranet could both flow through a common router located somewhere in the Internet. A company′s traffic on the Internet can no longer be considered to be isolated from the outside world, as it would have been on a dedicated private network, since flows from different VPNs will be intermixed on the Internet backbone.

- **Internal Network (Intranet):** This segment appears at an end point of the communications path. It is under the control of the corporation, who typically operates and manages it. Traditionally, almost all traffic flowing within a corporate network was generated by the corporation′s employees; very little traffic entered or exited the corporate network; and the protocols in the intranet were proprietary.

  Today, IP is becoming a popular protocol for use within corporate intranets, and data traffic enters and exits the corporate intranet regularly—consider Web browsers, ftp, or telnet applications. In today′s world of e-business, there are emerging requirements for external suppliers and business partners to have access to data stored on another company′s internal servers. Since traffic flowing within an intranet at any given time may have been generated by several different companies, today it may no longer be possible to categorize a given intranet as ″trusted″ or ″untrusted″. A company may consider its own intranets to be ″trusted″, but at the same time its business partners may consider it to be ″untrusted″. In this environment, a VPN designer may need to provide network security functions both on the intranet segments and on the Internet segment.

As shown in Figure 6, there are four classes of machines that occur along the path:

- Remote hosts (dial-up)
- Fixed hosts (sources and destinations, or clients and servers)
- ISP Access box
- firewall/security Gateway/routers

Protocols in these machines are used to provide address assignment, tunneling, and IP security. Viable security solutions can be constructed by deploying IPSec in some combination of remote hosts, firewalls, routers, and fixed hosts. But since each company should be responsible for its own security, there is no requirement for the ISP boxes or the routers in the Internet backbone to support IPSec.

5. SECURITY CONSIDERATIONS FOR VPNS

The use of VPNs raises several security concerns beyond those that were present in traditional corporate networks. For example, an end-to-end data path might contain:

- Several machines not under control of the corporation (e.g., the ISP access box in the Dial-In Segment and the routers within the Internet)

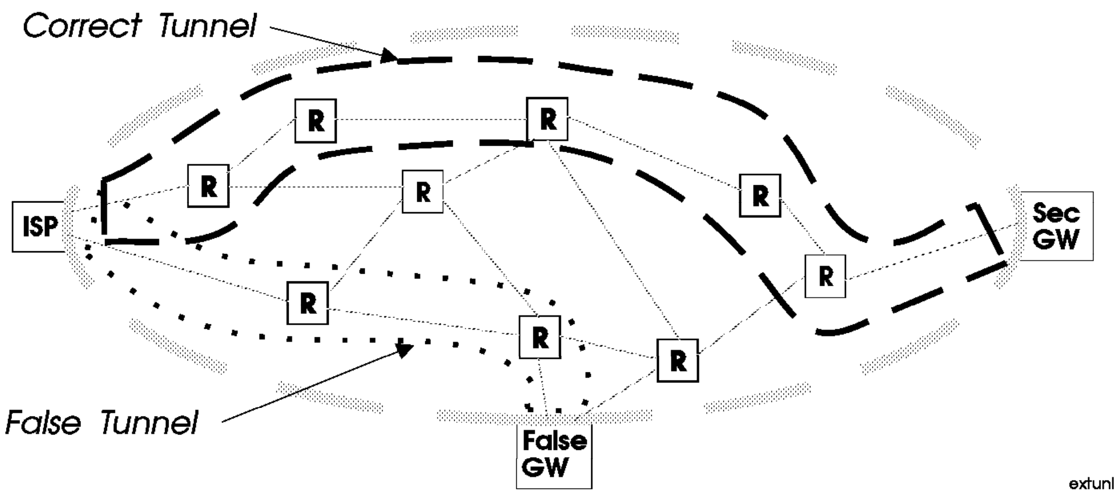


Figure 7. Exposures in the External (Internet) Segment. *Cleartext data can be examined by routers along the path, or false tunnels can be built.*

- A security gateway (firewall or router) that is located at the boundary between the Internal Segment and the External Segment
- An internal segment (Intranet) that contains hosts and routers. Some could could be malicious, and some will carry a mix of intra-company and inter-company traffic
- An external segment (Internet) that carries traffic not only from your company′s network but also from other sources

In this heterogeneous environment, there are many opportunities to eavesdrop, to change a datagram′s contents, to mount denial-of-service attacks, or to alter a datagram′s destination address, as outlined in the following sections. IPSec provides the tools to counter these threats.

5.1.1 Exposures in a Dial-In Segment

The Dial-In Segment in Figure 6 delivers a user′s data traffic directly to an Internet Service Provider (ISP). If the data is in cleartext, then it is very easy for the ISP to examine sensitive user data, or for an attacker to eavesdrop on the data as it travels over the dial-in link.

Link-layer encryption between the remote host and the ISP can protect against passive eavesdropping, but it does not protect against a malicious ISP. Since the ISP can decrypt the user′s data stream, sensitive data is still available to the ISP in cleartext format.

5.1.2 Exposures in the Internet

In some remote-access scenarios, an ISP builds a tunnel to extend the reach of the PPP connection so that its end points will be the access box and the Security Gateway. If the tunneling protocol does not incorporate robust security features, a malicious ISP could easily build a tunnel that terminates somewhere other than at the correct Security Gateway (see Figure 7). Thus, user′s data could be delivered via a false tunnel to a malicious impostor gateway where it could be examined or even altered.

There are also dangers as the datagram travels within the tunnel. As illustrated in Figure 7, user datagrams pass through routers in the Internet as they travel along a path toward the tunnel end point. If the datagrams are in cleartext, any of these routers could easily examine or modify the datagram, and passive attackers could eavesdrop on any of the links along the path.

Link-by-link encryption at each hop in the Internet backbone can thwart eavesdroppers, but does not protect the user′s data from a malicious router, since each router along the path would be capable of decrypting the user′s data stream. Nor does link-by-link encryption protect against false tunnels, since the false tunnel endpoint would have access to cleartext data.

As we noted in section 3.3, even popular tunneling protocols such as Layer 2 Tunneling Protocol (L2TP) do not provide robust security, so the IETF has recommended that the tunnel traffic should be protected with the IPSec protocols.

5.1.3 Exposures in a Security Gateway

The Security Gateway (firewall) shown in Figure 6 also creates security exposures. Its main purpose is to enforce an access control policy: that is, to accept only the desired inbound traffic, to reject undesired inbound traffic, and to prevent internally generated traffic from indiscriminately leaving the corporate network. The Security Gateway is under the control of the corporate network, but an internal attacker still has an opportunity to examine any traffic that the gateway decrypts and then forward into the intranet in cleartext form.

Non-cryptographic authentication provides some protection against unwanted traffic entering or leaving the network. Common techniques are passwords, packet filtering, and network address translation. However, these can easily be defeated by a variety of well known attacks, such as address spoofing, and new attacks are being developed regularly. Each time a new packet filter is designed to thwart a known attack, the hackers will devise a new attack, which in turn demands that a new filter rule be generated.

Because the cryptography-based authentication techniques require a long time to break, even with powerful computers, it becomes prohibitively expensive, both in time and in computer power, for a hacker to attempt to attack them. Hence, companies can deploy them with the confidence that they will provide robust protection against a hacker′s attacks.

Link-by-link encryption does not prevent an intermediate box along the path from monitoring, altering, or rerouting valid traffic, since each intermediate box will have access to the cleartext form of all messages. Even host-to-gateway encryption suffers from the same weakness: the gateway still has access to cleartext.

5.1.4 Exposures in an Intranet

Although there is a popular belief that the security breaches will occur in the public Internet, there have been studies showing that many of the attacks actually arise internally. Unless every host, Gateway, and router within the Intranet of Figure 6 can be fully trusted, it is possible for a malicious employee to modify an internal box, making it possible to monitor, alter, or reroute datagrams that flow within the corporate network. When data from several different networks flows within the intranet—for example, in the case where the VPN interconnects a manufacturer′s intranet with the intranets of several suppliers—threats within the intranet need to be guarded against. Even if Company A trusts that its own intranet is secure, the external supplier or business partner whose traffic must flow through Company A′s intranet may not trust it—after all, the partner′s data is at risk if Company A′s intranet is in fact compromised in any fashion.

6. IPSEC-BASED SOLUTIONS

There are security exposures everywhere along an end-to-end path: on the dial-up link, in an ISP′s access box, in the Internet, in the firewall or router, and even in the home corporate intranet. Previously, security solutions were developed to address just a subset of these exposures, but there was no framework that could protect against all these exposures using a single approach. IPSec is the first approach that offers a comprehensive, consistent approach: it can provide end-to-end protection as well as segment-by-segment protection.

End-to-end data encryption is only part of the solution. IPSec also offers end-to-end authentication and integrity checking. Its *transport* and *tunnel* modes allow nesting of IPSec protocols within one another. *Transport mode* provides end-to-end protection along a complete path, while *tunnel mode* provides protection on a single segment of the path. The tunnel and transport capabilities are especially important, since they allow a network operator to deploy security functions along each

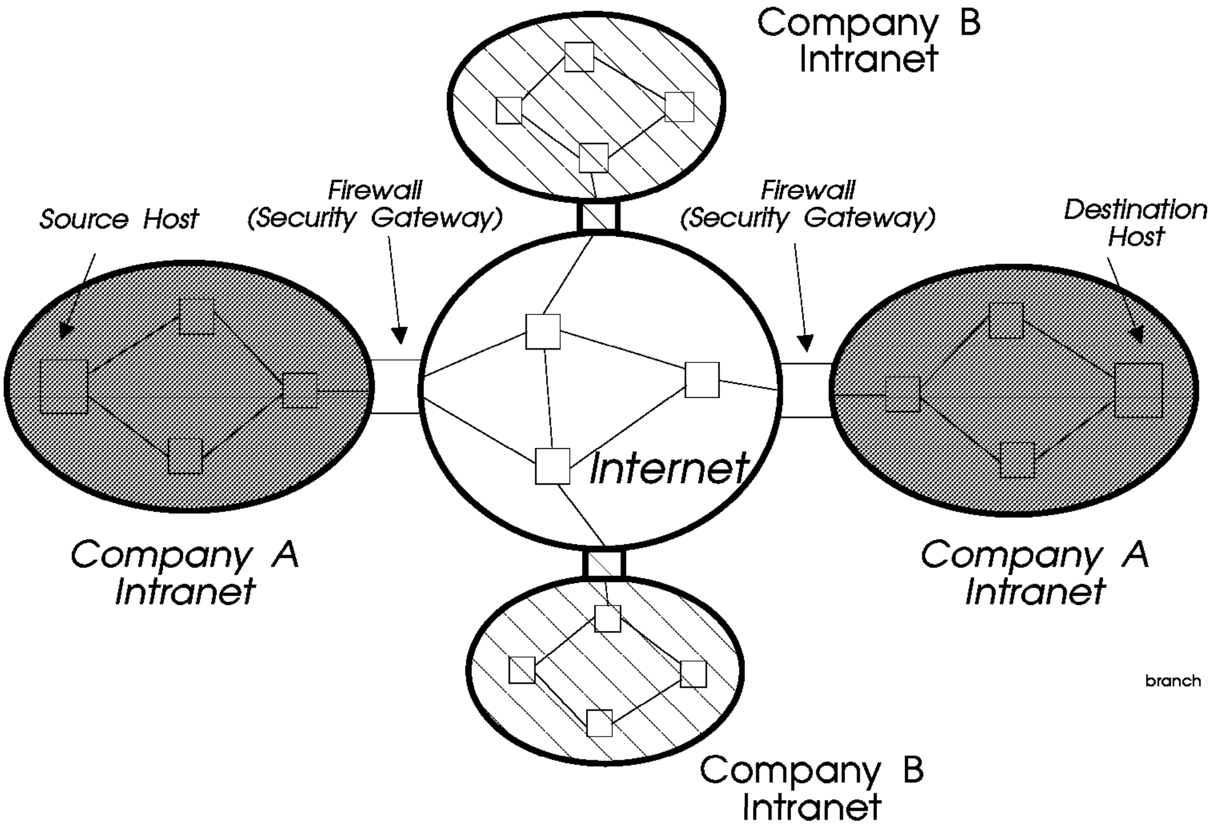


Figure 8. Branch Office Interconnect Configuration. *The intranets in several locations are interconnected using the Internet as a backbone network.*

segment of a path in the most cost effective manner for his particular application and network configuration.

IPSec′s authentication functions will also play a key role, because they provide cryptographically strong access control by Security Gateways, client machines, and server machines. And finally, we will see that the IPSec protocols can also protect the topology and address information that will be exchanged by a VPN′s routing protocols.

6.1 BRANCH OFFICE INTERCONNECT

Consider a company that was running its own private network, using its own routers, bridges, and private lines. If the company had campuses at geographically dispersed sites, it may prove more economical to break the corporate network into pieces (the ″intranets″), add a firewall to control traffic flow across the intranet/Internet boundary, and then procure service from one or more ISPs to interconnect the intranets over the Internet backbone. Figure 8 illustrates this configuration: it shows two companies, A and B, that have each created two intranets and then interconnected them over the Internet. For this example, we will assume that Company A just wants to enable communication between its intranets, but does not necessarily want to communicate with entities outside of Company A.

Let us consider how Company A could construct a Virtual Private Network for interconnecting its intranets securely.

6.1.1 Authenticating Backbone Traffic

As shown in Figure 8, the Internet will be carrying traffic not just from Company A′s VPN, but also from other VPNs, such as Company B′s. Company A′s firewall must admit only traffic from Company A, and must reject incoming traffic that originates from all other sources. Deploying IPSec′s authentication protocols in Company A′s firewalls (or IPSec-enabled routers) at the intranet boundary will accomplish these goals. IPSec′s authentication techniques are cryptographically strong, so they provide significantly better protection against ″address spoofing″ attacks than would reliance on conventional, non-cryptographic filtering techniques. In this scenario, cryptographic authentication using HMAC will be the first line of defense. Having established that the traffic has come from somewhere within Company A′s network, non-cryptographic filtering can then be used as the second line of defense to provide more granular access control, if desired.

6.1.2 Data Confidentiality

It should be obvious that Company A will want to keep its data confidential (i.e., encrypted) while it is in transit across the public Internet. But it is not always clear if the data should also be protected when it flows within its own intranets. If the company had not previously encrypted its internal traffic when it used a monolithic private network, it may not see value in encrypting it when it flows within its intranets.

If a company does not believe that it is subject to internally generated attacks, the simplest solution will be to encrypt and authenticate traffic flowing between firewalls, and make no security-related changes to the end systems themselves. This has the advantage of much fewer security associations to manage: two per firewall for bidirectional data flow, compared to two per host for host-to-host encryption. But it has the disadvantage that traffic is exposed to relatively simple attacks while it flows in the intranet. Since authentication is also needed between firewalls, the simplest Branch Office VPN will use ESP with authentication between the two firewalls. And since the firewall-to-firewall segment is only a portion of the end-to-end path, we will use ESP in its tunnel mode.

But let′s also consider how to configure a solution between Branch Offices that can protect you against threats both in the Internet and in your company′s intranet as well. Here we would want to encrypt the data along its entire path, so we would use the ESP protocol in transport mode between the source and destination hosts. And even within a single company, access to some databases (such as a personnel database) may be restricted to only a few specific users. So, we may want to use end-to-end authentication as well as end-to-end encryption. This can be handled by using the ESP protocol in transport mode, with its optional authentication feature activated.

Since we′re using end-to-end encryption, we do not need an additional layer of encryption between the firewalls, but we still want firewall-to-firewall authentication to assure that the VPN does not admit foreign traffic into the company intranet. Hence we will use the AH in tunnel mode between the two firewalls.

Since we have simultaneously deployed ESP between the end points and AH between the firewalls, we have made use of nested (or bundled) security associations:

- A security association is needed between the source and destination hosts to support end-to-end encryption with authentication
- Another security association is needed between the two firewalls to support firewall-to-firewall authentication
- The end-to-end SA is nested inside the firewall-to-firewall SA

Section 2.3 described the concepts of nesting in general terms. The formats of the datagrams for the example above with nested IPSec protocols are the ones illustrated in Figure 4.

6.1.3 Addressing Issues

In the Branch Office Scenario, we assumed that Company A previously had a traditional network in place, where its various intranets were interconnected over private facilities, such as leased lines or frame relay. We also assumed that Company A has already developed an address plan for its network. Since the network was self-contained and the backbone used only private facilities, Company A could have used either globally ambiguous (private) IP addresses (i.e., of the form ″10.x.y.z″) or globally unique (public) addresses obtained from the NIC (Network Information Center).

Since assignment of public IP addresses is coordinated through a global authority, they are unambiguous. Public addresses are ″routable″ everywhere.

However,since private address assignments are assigned locally without coordination by a global authority, they are ambiguous: the same private address could be assigned to different machnes independently by several different companies. Therefore, a private address is ″routable″ only within its own private network, but not on the public Internet.

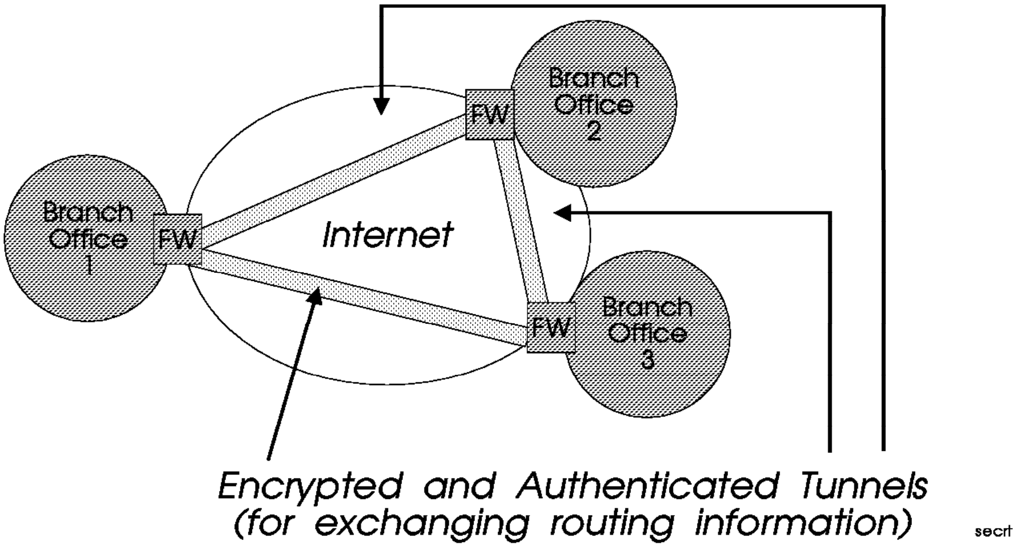

Branch Office 1
FW
Branch Office 2
FW
Branch Office 3
FW
Internet
Encrypted and Authenticated Tunnels (for exchanging routing information)
secrt

Figure 9. Exchanging Routing Information Securely. *IPSec′s ESP protocol between a set of firewall-routers can authenticate and encrypt routing information.*

In summary:

1. If Company A had used public addresses in its network, they can continue to be used without change in the VPN environment. If it desired to hide them while the datagram is in transit over the Internet, an ESP tunnel can be used between firewalls, as noted in section 2.2.
2. If Company A used private IP addresses in its network, they can also continue to be used on all subnets that have no physical connection to the public Internet. But for those subnets that do connect to the public Internet—typically the exit links at the boundary of the intranet—a public IP address must be used.

ESP tunnel mode between firewalls handles both situations. The ESP tunnel′s ″New IP Header″ will use the global addresses of the two firewalls, allowing datagrams to be routed over the Interent between the two firewalls (or routers). The header of the original (inner) IP datagram will use the IP addresses assigned for use in the intranet; since these addresses will be hidden from view by ESP′s encryption protocol, they can be either publicly or privately assigned.

6.1.4 Routing Issues

Because a VPN is in fact a *network*, all but the smallest VPNs will typically need to deploy an IP routing protocol between the gateway machines (firewalls or routers) at the boundaries of the company′s intranets. Routing protocols typically exchange information that will describe the topology of the VPN—that is, the topology updates will describe the IP addresses that are reachable within each intranet that participates in the VPN. IPSec can be used to both encrypt and authenticate the routing information, thus hiding topological details of the intranet as they are exchanged across the public Internet.

The corporate network operator can incorporate conventional IP routing protocols into the firewalls, and then use IPSec′s ESP protocol to encrypt and authenticate the exchange of routing information among the firewalls. Figure 9 illustrates this concept schematically for a sample configuration that consists of three branch offices of a given company that need to communicate among themselves via the public Internet.

When an IPSec tunnel is established between a pair of firewalls, they appear to be logically adjacent to one another, even though there may be several routers along the actual physical path. Each pair of virtually adjacent firewall/Routers will set up a security association between themselves, using ESP tunnel mode to provide both encryption and authentication. The routing information that is exchanged will then be hidden from view because it will have been encrypted.

Because the set of firewalls participate in a common routing protocol, they will know the correct firewall for reaching any given destination host within the intranets. Hence, data traffic arriving at an exit firewall can be sent via an ESP-tunnel, using its authentication option, and can then be authenticated by the entry firewall that protects the remote branch office′s Intranet.

Thus, IPSec makes it possible to exchange routing information and user data between branch offices over the Internet while preserving the confidentiality of both user data and intranet topology information. Because routing information (e.g., IP addresses) is visible only to other members of the corporate network, this scheme can be used regardless of whether addresses used in the interior of an intranet are globally unique or privately assigned. To be more specific, since the intranet addresses are carried within encrypted routing update messages, they are neither visible to, nor used by, any of the routers in the Internet. Therefore, if Company A′s intranets use a self-consistent addressing scheme, either public or private, Network Address Translation is not needed for intranet addresses. Encryption already hides the interior addresses, and all backbone routing is based only on the public IP addresses of the boundary firewalls. Finally, depending on the sophistication of the routing algorithms, it may also be possible to support redundant entry/exit points into a corporate network.

6.1.5 Summary: Branch Office Interconnection

This application replaces existing private lines or leased lines in a corporate network, and uses the public Internet as the backbone for interconnecting a company′s branch offices[3]. This solution does not mandate any changes in the clients (PCs or servers) unless it is desired to protect against internal attacks as well as external ones.

The design features are:

- Client machines (hosts and servers) need not support IPSec if the intranets are considered to be trusted and secure. This minimizes the migration issues of moving to a VPN approach and maintains the pre-existing host-to-host security policies and procedures of the original network. IPSec support will be required only at the intranet boundaries—that is, in the VPN-firewall[4] boxes. And there will be no security-related protocols required in any of the routers, bridges, or switches that are located either in the interior of the intranets or in the public backbone Internet.
- VPN-firewalls situated at the perimeter of each branch office intranet implement the basic firewall functions (e.g., packet filtering) and also support IPSec protocols to build secure and authenticated tunnels between all VPN-firewalls of the branch office networks that comprise the VPN.

  User data traffic will be both authenticated and encrypted. Any inbound traffic that can not be authenticated by the VPN-firewall will not be delivered into the intranet. Authentication will be cryptographically based, using the authentication option for IPSec′s ESP protocol. Note that since ESP can provide encryption, authentication, replay protection, and integrity checking, there is no requirement for the VPN-firewall to support IPSec′s AH protocol if the intranets are considered to be ″trusted″.
- Routing control messages will be exchanged among the set of VPN-firewalls, and these messages will also be encrypted and authenticated using IPSec procedures.
- If the number of VPN-firewalls in the initial VPN deployment is small, key distribution and security association definition can be handled by manual methods. But as the VPN′s size grows to encompass more and more intranets, the automated ISAKMP/Oakley procedures will rapidly become a necessity.
- Security associations will be set up among the set of VPN-firewalls. Since the source and destination hosts (i.e., clients and servers) are not required to

[3] This solution is not limited only to branch offices, but can also be be applied between any collection of a company′s geographically dispersed sites, such as labs, manufacturing plants, warehouses, etc.

[4] The boxes at the boundaries could be IPSec-enabled firewalls or IPSec-enabled routers. The term ″VPN-firewall″ is a shorthand notation to cover both cases.

support IPSec, no security associations need to be set up between hosts, and no keys need to be assigned to them. In the future, if even stronger security is desired for host-to-host communications, then clients and servers will need to support the IPSec protocols.

- The IP addresses assigned for use in the intranets can be used ″as is″, regardless of whether they were assigned from a public or a private address space. Only the subnets of the VPN-firewalls that attach to the Internet backbone are required to use globally unique IP addresses.
- Packet filtering rules, if any, that were used in the pre-VPN network should be installed on the VPN-firewall to control traffic that enters the branch office intranet. They can be used as a second line of defense, after the packets have been authenticated by IPSec′s AH protocol.
- If end-to-end IPSec functions are deployed between hosts, then new packet filtering rules will be needed in the firewalls to recognize the IPSec AH or ESP Headers.
- Explicit packet filtering rules to protect the branch office intranet from traffic that originated outside the VPN are not required because IPSec′s cryptographic authentication techniques will provide this protection. In particular, IPSec provides robust protection against spoofing attacks.

6.2 INTERCONNECTING DIFFERENT COMPANIES′ INTRANETS

Consider a situation where a manufacturing company needs to communicate regularly with its suppliers—for example to facilitate just-in-time delivery of parts, to settle invoices among themselves, or for any number of other reasons. There are two issues to consider:

- **Access Control:** While it may be a business necessity for a supplier to have access to some of Company A′s internal resources (such as databases), there will also be valid business reasons to prevent the supplier from having access to all of Company A′s databases.
- **Data Confidentiality:** Clearly the data should be hidden from general view while it is in transit over the public Internet. But there may be even more stringent requirements. Company A may consider its own intranet to be ″trusted″, but its suppliers may not. For example, a supplier may want to insure that its sensitive data, while traveling through Company A′s intranet, is hidden until it reaches its final destination—for example, the supplier may be worried that an unscrupulous eavesdropper inside Company A may try to intercept the data and sell it to a competitor. And Company A may have the same concerns about its data as it travels though the supplier′s intranet. Thus, it will not be unusual for each party to treat the other′s intranet as ″untrusted″.

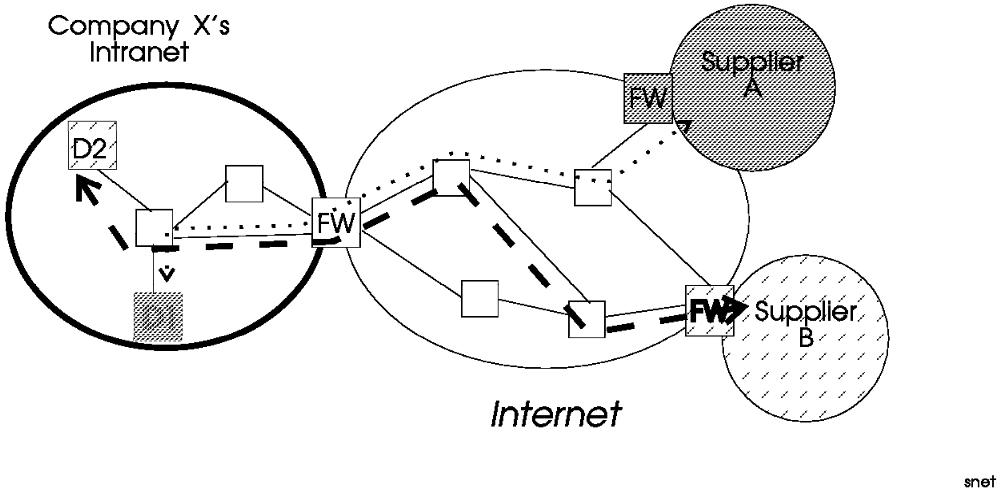


Figure 10. A Typical Supplier Configuration. *Traffic from Suppliers A and B can be intermixed both within the Internet and within Company X's intranet.*

This scenario is an extension of the multiple branch office scenario. Here we have multiple supplier Intranets that need to access a common corporate network over the Internet. Each supplier is allowed access to only a limited set of destinations within the corporate network. Even though traffic from the different suppliers flows over common data links in both the public Internet and in the destination intranet, the VPN must be constructed to guarantee that no traffic from a given supplier will be visible to any other supplier or to/any system other than its intended destination. See Figure 10, where we illustrate how the two data paths, the dashed and dotted lines, can flow through several common boxes. In this example, Supplier A can talk only to Destination 1 and Supplier B can talk only to destination 2.

IPSec provides a secure solution in this scenario, but it will be more complex for the branch office scenario outlined in section 6.1. The extra complexity arises from the following factors:

- There can be multiple suppliers who need to communicate with the manufacturer. Hence, it may be necessary to insure that Supplier A can never see any other supplier′s data in cleartext form, either in the Internet or in the manufacturer′s intranet.
- If the manufacturer and the suppliers, or some subset of them, use private addressing in their respective intranets, then it is possible that ″routing collisions″ can occur if the same private address has been assigned to multiple hosts. To avoid this possibility, the members of the VPN must either use public

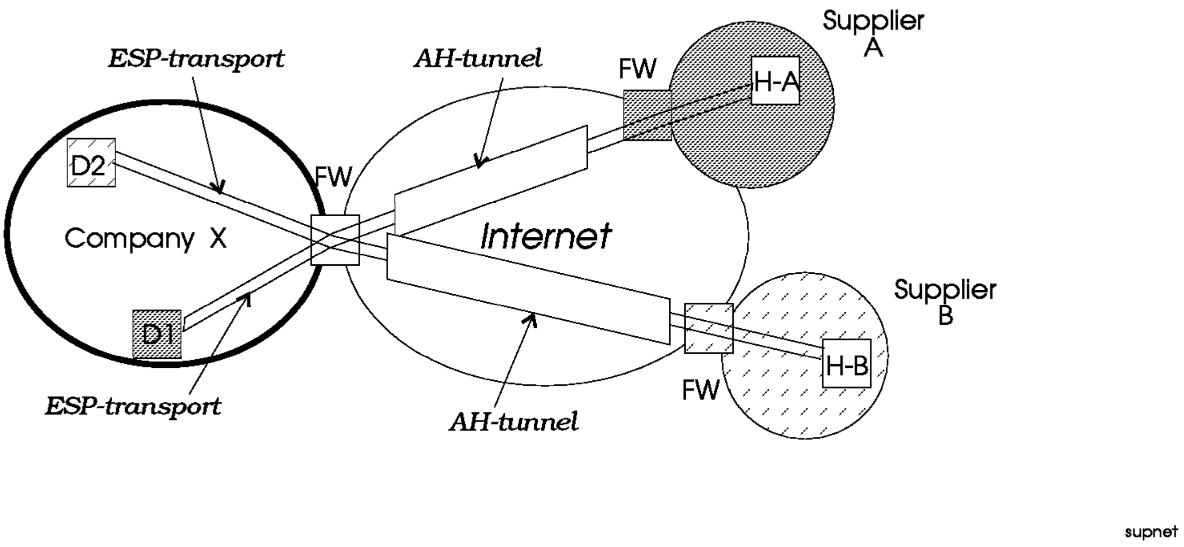


Figure 11. A Typical Supplier Configuration. *Systems in Supplier intranet A can communicate with Destination 1, and systems in Supplier intranet B can communicate with Destination 2.*

IP addresses in their intranets, coordinate the assignment of private IP addresses among the systems participating in the VPN, or adopt some sort of Network Address Translation strategy.

- Because security coverage extends from host-to-host (client-to-server) rather than just from firewall-to-firewall, there will be many more Security Associations to be negotiated, and many more keys to be securely distributed and refreshed, as compared to the Branch Office scenario. Hence, the automated secure functions of ISAKMP/Oakley will become a necessity.
- Because security coverage extends from host-to-host, IPSec functions will need to be supported in clients, servers, and firewalls.

6.2.1 Authenticating and Encrypting Supplier Traffic

As shown in Figure 11, the VPN-firewall that guards the entry to Company X′s intranet must accept traffic from both Supplier A and Supplier B. This can be accomplished by using IPSec′s AH protocol. There will be one tunnel between the firewall of Company X and Supplier A; and another between the firewall of Company X and Supplier B. The AH protocol will be used in tunnel mode, providing cryptographically strong access control.

But as we have noted, there is a need for even finer-grained authentication: namely, each source to its intended destination. For example, in Figure 10, we need to assure that destination D1 will accept traffic only from Host A and not from Host B. To achieve data confidentiality, we will use end-to-end encryption between each host and its intended destination server (for example, from Host A to destination D1). IPSec protocols provide the means to accomplish this by using *bundled security associations*, which make use of both tunnel and transport modes of operation simultaneously.

To handle the host-to-host authentication and encryption requirements, we will establish a security association between each client machine and its server. The protocol will be ESP with authentication, and the mode of operation will be ″transport″, since this is an end-to-end security association.

Next, we will establish a different IPSec security association between the firewalls that protect Company X′s intranet and the supplier′s intranet. This SA applies over only part of the complete path, so it will use the AH protocol in tunnel mode. Between firewalls, ESP security association will be nested inside the AH security association. Figure 12 illustrates the structure of the datagram that flows between firewalls.

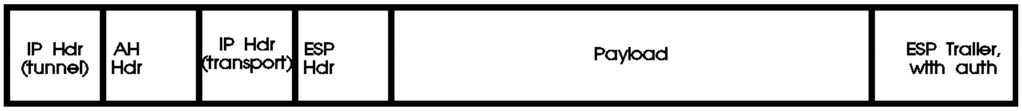
Note that IPSec protocols enforce two levels of authentication: firewall-to-firewall and client-to-server. The firewall-to-firewall authentication prevents denial of service attacks by making sure that only traffic from legitimate suppliers can enter Company X′s intranet; the host-to-host authentication assures that the destination will accept traffic only from its intended partner machines.

This considerably exacerbates scaling issues. Unlike the Branch Office case where Security Associations were established only between VPN-firewalls, it is now necessary to establish two Security Associations per client, and then ″bundle″ them together. Each security association will require its own set of cryptographic keys. This scenario illustrates the need for automated ISAKMP-based methods, both for negotiating multiple bundled security associations and for distributing the associated keys.

6.2.2 Addressing Issues

In Figure 11 there are tunnels between Supplier A and Company X, and also between Supplier B and Company X. But there is no tunnel between Supplier A and Supplier B. For routing purposes, Supplier A and Company X will run a mutually acceptable routing protocol over their tunnel, and Company X and Supplier B will also independently run their own routing protocol. Because each tunnel has its own security association, routing data for Supplier A can be kept secret from Supplier B, and vice versa. As in the case of the Branch Office Interconnection, each security association will use IPSec′s ESP protocol to both encrypt and authenticate the routing updates.

Unlike the Branch Office case, where we could assume that a consistent addressing plan had been applied across all the company′s intranets, in this configuration it is very likely that Company X and each of its suppliers have administered their own



NOTES:

1. Outer (tunnel) header uses Firewall addresses for source and destination.

2. Inner (transport) header has client's address as source and server's address as destination.

Figure 12. A Typical Supplier Scenario Datagram. *An inner datagram is nested inside an outer datagram to support two distinct bundled security associations: client-to-server and firewall-to-firewall.*

addressing plan independently of one another. For example, it would be possible that Supplier A and Supplier B both used private (globally ambiguous) IP addresses in their networks, and it would be possible for some or all of their addresses to overlap. In this case, conventional IP routing protocols will not be able to resolve these ambiguities. Hence, we will make the assumption that the IP addresses of all systems—both in the home intranet and in the suppliers′ intranets—have been assigned so that they are non-overlapping: that is, we will assume that when private IP addresses are used, there will be coordination between the communicating intranets.

6.2.3 Packet Filtering and Proxies

In this configuration, we have seen that there is a requirement for end-to-end encryption. This can cause problems for conventional packet filtering techniques, since the TCP header is part of the encrypted payload field and is no longer visible to the VPN-firewalls. Another area that needs to be addressed is the nesting of IPSec protocols. This means that the VPN-firewall must be able to handle IP packets where the ″next protocol″ field might indicate AH or ESP. It may also mean that packet filters will need to operate on both ″inner″ and ″outer″ IP address information, in cases where tunnel mode is used.

This area needs more study. The effectiveness of packet filtering will be significantly reduced, since unencrypted upper layer data is no longer available for examination by the VPN-firewall. As the cryptographic techniques become used more widely for end-to-end protection, more and more access control decisions in a firewall will be handled via the AH protocol, and conventional packet filtering will become less and less useful. However, at the final destination host, where cleartext data is once again available, packet filtering will still continue to play a useful role for providing finer-grained access level control within the destination host itself.

6.2.4 Summary: Inter-company Interconnection

This application uses the public Internet to connect a company and its suppliers together. It requires upgrades to existing client and server machines, since they must now support the IPSec protocol suite. It requires enhancements to conventional packet filtering techniques, since some headers from upper layer protocols may no longer be decryptable at the VPN-firewall. And finally, it makes use of IPSec′s nesting capabilities. The major elements of complexity, compared to the Branch Office case, are summarized below:

- Client machines (hosts and servers) must support IPSec′s ESP protocol, both for encryption and for authentication.
- The number of machines that need to participate in the IPSec protocols has increased significantly. Security associations will need to be set up both end-to-end and firewall-to-firewall.
- For very small configurations, manual key distribution and manual configuration of security associations may be possible, but for any medium to large sized configuration, support for ISAKMP/Oakley in clients, servers, and VPN-firewalls will rapidly become a necessity.
- New packet filtering rules will need to be developed to accommodate a) encrypted upper layer payloads, and b) pairs of inner and outer cleartext headers that arise when IPSec protocols are nested within one another. It remains to be seen if firewall filtering rules in the presence of end-to-end encryption will continue to serve a useful purpose. In the long term, filtering′s importance will probably diminish as cryptographically-based access control techniques become more widely used.

6.3 DIAL-IN REMOTE ACCESS

Figure 13 illustrates a common remote access configuration that includes a remote host, a dial-up link, an ISP access box, the Internet, a firewall, and a destination host inside the corporate intranet. The remote user dials in to a local ISP, using whatever procedures and protocols the ISP requires. For example, many ISPs require the remote host to use the PPP protocol, and to identify itself by an account name and a password. At this point, the remote host typically receives a dynamically-assigned IP address from the ISP[5].

[5] Sometimes the corporate firewall could assign the dynamic IP address, particularly when a protocol like L2TP has been used to extend the span of a PPP connection.

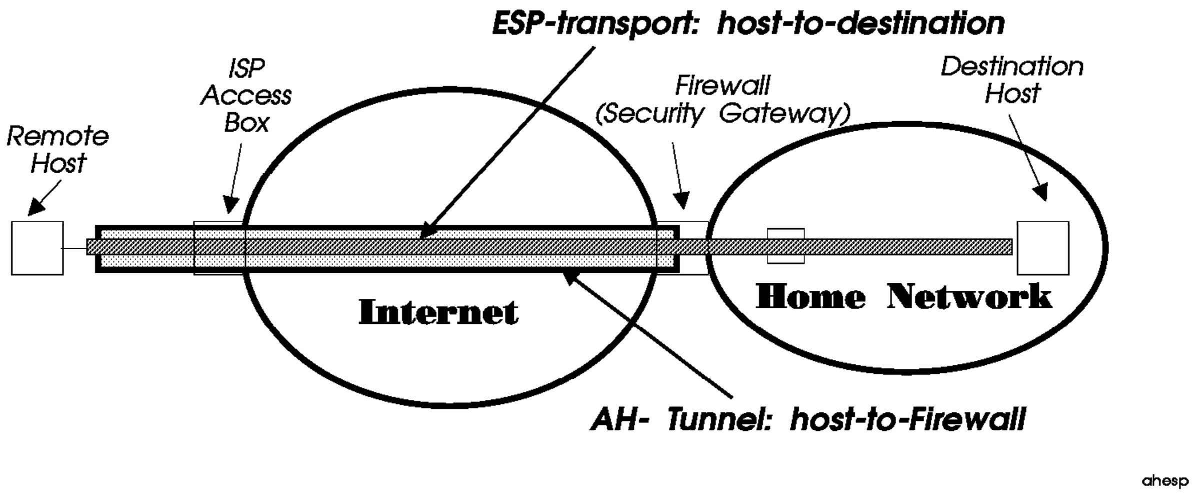

ESP-transport: host-to-destination
ISP Access Box
Remote Host
Firewall (Security Gateway)
Destination Host
Internet
Home Network
AH- Tunnel: host-to-Firewall
ahesp

Figure 13. Providing Segment-specific Protection. *Traffic from the remote host is authenticated by the firewall and decrypted by the destination.*

Just as in the other scenarios that we have examined, the remote host will be responsible to set up the needed security associations with its peers. Since the remote host′s dynamically assigned IP address is not known in advance, it is not possible to pre-configure security associations for the firewall or the destination host to use when they send packets to the remote host. (Remember that the *destination address* is one of the items that an IPSec-enabled source uses to select the security association that it will use for sending traffic to the remote host.)

But the ISAKMP protocol offers a solution even when the remote host′s IP address is not known in advance. ISAKMP allows a remote host to identify itself by a ″permanent″ identifier, such as a name or an e-mail address. The ISAKMP Phase 1 exchanges will then authenticate the remote host′s permanent identity using public key cryptography:

- Certificates create a binding between the permanent identifier and a public key. Therefore, ISAKMP′s certificate-based Phase 1 message exchanges can authenticate the remote host′s permanent identify.
- Since the ISAKMP messages themselves are carried within IP datagram, the ISAKMP partner (e.g., a firewall or destination host) can associate the remote host′s dynamic IP address with its authenticated permanent identity.

See Appendix A for a detailed discussion of how ISAKMP/Oakley exchanges authenticate the remote host to its peer and set up the security associations dictated by its corporate VPN policy.

The critical element in the remote access scenario is the use of ISAKMP/Oakley to identify the remote host by name, rather than by its dynamically assigned IP address. Once the remote host′s identity has been authenticated and the mapping to its dynamically assigned IP address has been ascertained, the remainder of the processes are the same as we have described for the other scenarios. For example, if the corporate intranet is considered to be trusted, then the remote host needs to establish a single SA between itself and the firewall. But if the corporate intranet is considered to be untrusted, then it may be necessary for the remote host to set up two SAs: one between itself and the firewall, and a second between itself and the destination host.

Recall that a single ISAKMP Phase 1 negotiation can protect several subsequent Phase 2 negotiations. Phase 1 ISAKMP negotiations use computationally intensive public key cryptographic operations, while Phase 2 negotiations use the less computationally intensive symmetric key cryptographic operations. Hence, the heavy computational load only occurs in Phase I, which will only be executed once when the dial-up connection is first initiated.

The principal points that pertain to the remote access case are:

- The remote host′s dynamically assigned address is the one that is placed in the IP header of all ISAKMP messages.
- The remote host′s permanent identifier (such as an e-mail address, for example) is the quantity that is placed in the ID field of the ISAKMP Phase 1 messages.
- The remote host′s certificate used in the ISAKMP exchange must be associated with the remote host′s permanent identifier.
- In traffic-bearing datagrams, the remote host′s dynamically assigned IP address will be used. This is necessary since the destination IP address that appears in the datagram′s IP header is used in conjunction with the SPI and protocol type to identify the relevant IPSec security association for processing the inbound datagram.

6.3.1 Layer 2 Tunneling Protocols and IPSec

The remote access configuration described in section 6.3 allowed a remote host that was running the IP protocol to communicate securely with a peer in its home corporate intranet. IPSec protocols were applied in a straightforward manner, and the remote host dialed in to a local ISP, thus avoiding expensive long distance charges. But, all of this was accomplished without using any layer 2 tunneling protocol. When the IPSec is used to secure IP traffic, any L2TP tunnels are redundant, and only add additional layers of protocol processing. This begs the question of why protocols like L2TP are popular for remote access.

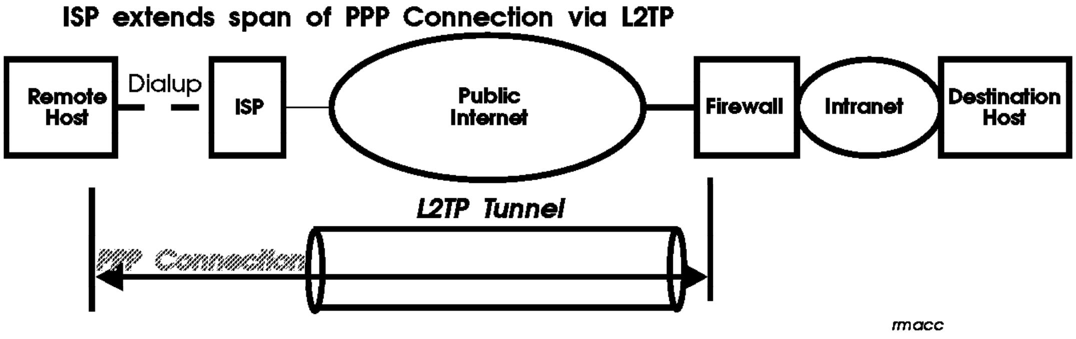


Figure 14. Remote Access Possibilities. *An ISP may or may not build a Layer 2 Tunnel back to the firewall.*

The answer is their ability to support multiple protocols other than IP. The PPP can carry payload protocols other than IP. L2TP, in turn, encapsulates the PPP packet. Since L2TP encapsulates PPP packets, L2TP inherits PPP′s ability to carry multiple protocols.

Let′s look briefly at the two ways that its proponents envision L2TP to be used: either the ISP builds an L2TP tunnel between itself and the corporate firewall, or the remote host builds an L2TP tunnel between itself and the corporate firewall. These situations are illustrated in Figure 14 and Figure 15.

Recall that L2TP uses IP encapsulation to extend the PPP connection so that the ″virtual link″ spans the Internet between the tunnel end points. As described in 3.3 and illustrated in Figure 5, when L2TP tunneling is used, the outer header of the resulting datagram will always be an IP header: that is, one can treat everything behind the IP header shown in Figure 5 as the payload of an IP packet. Hence, it follows that the tunnel end points (whose addresses are carried in the outer IP header) can protect the L2TP tunnel, including the PPP payload, using IPSec technologies.

One key point to remember is that if the PPP payload carries IP, then the L2TP tunnel is not really needed in the first place, so any additional IPSec functions used to protect the L2TP tunnel are unnecessary overhead. But when non-IP protocols are carried in the PPP payload, then IPSec is the method of choice for securing the L2TP tunnel[6].

[6] See for example the internet draft ″Securing L2TP using IPSec″ (draft-ietf-pppext-l2tp-security.00.txt) for a discussion of L2TP′s security weaknesses and how IPSec can mitigate them.

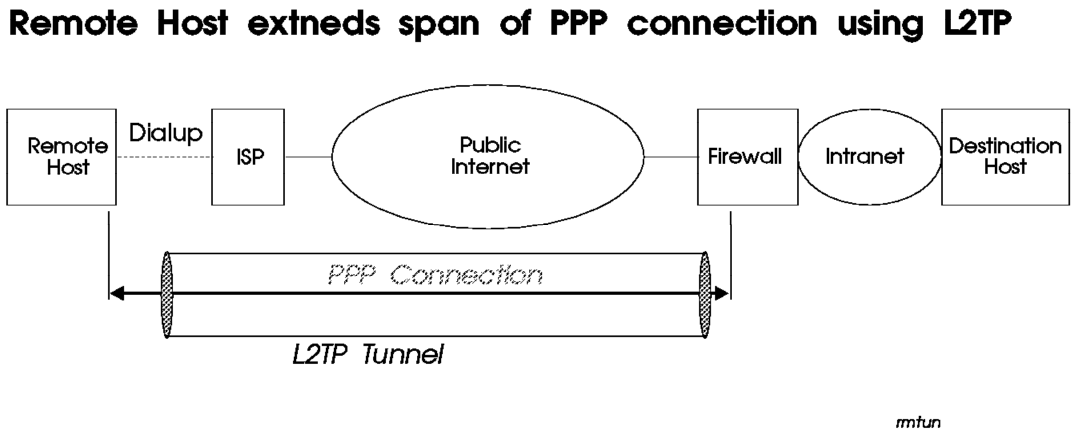


Figure 15. Remote Host as Endpoint of L2TP Tunnel. *A remote host that implements L2TP can use IPSec to protect any protocol that can be carried within a PPP packet.*

The second key point to remember is that when L2TP is used to take advantage of its multiprotocol capability, both boxes at the ends of the PPP connection (which has been extended by L2TP) must support whatever non-IP protocol is being tunneled. And both must also support IP, since IP is the encapsulating protocol that carries the L2TP packets over the Internet.

7. CONCLUSIONS

The IPSec suite of protocols provides a flexible, robust method for building secure networks. By judiciously using AH-tunnel, AH-transport, ESP-tunnel, and ESP-transport, network operators can accommodate a variety of common configurations. And inclusion of routing functionality in VPN-firewalls will allow dynamic, routing-responsive VPNs to be implemented with the IPSec protocol suite.

ISAKMP/Oakley can provide a secure, automatic method for negotiating security associations and distributing cryptographic keys, even over insecure links. And IPSec can provide robust security for other popular protocols, such as L2TP, which themselves provide only minimal protection.

Constructing a complete, easy-to- deploy, easy-to-manage VPN solution may also include other complementary protocols and procedures (e.g., certificate management, use of upper layer security protocols, etc.) in the overall VPN solution.

In summary, IPSec provides an extremely flexible, powerful framework from which to tailor the best security solution for your particular needs. This paper discussed three simple scenarios to illustrate the versatility of IPSec, but many other configurations are possible. For example, there may be supplier scenarios where the supplier′s host needs to connect to a manufacturing company′s intranet using dial-up facilities. By mixing and matching IPSec capabilities and elements from the three example scenarios, you can design a VPN that can meet almost any foreseeable e-business need.

This paper has only skimmed the surface. IPSec is a powerful security framework, and many variations are possible. We attempted to highlight the major technical aspects of IPSec, but to discuss all possible details would have required a book, rather than a white paper. The interested reader is referred to the Bibliography to locate the most current technical specifications for the IPSec protocols.

APPENDIX A. INITIALIZING SAS WITH ISAKMP/OAKLEY

This section outlines how ISAKMP/Oakley protocols initially establish security associations and exchange keys between two systems that wish to communicate securely. To provide a concrete example, we will describe a message sequence with the following characteristics:

- ISAKMP messages themselves will be carried as a UDP payload
- ISAKMP Phase 1 exchanges will be authenticated with digital signatures based on certificates obtained from a valid certificate authority.
- Parties participating in the ISAKMP Phase 1 exchanges will be identified by user-based certificates (i.e., by name) rather than by host-based certificates (i.e., by IP addresses). This is a more general solution, since it can be used even when a host receives a dynamically assigned IP address.
- ISAKMP Phase 2 exchanges will be used to negotiate the protection of user traffic with the ESP protocol, making use of its optional authentication function.

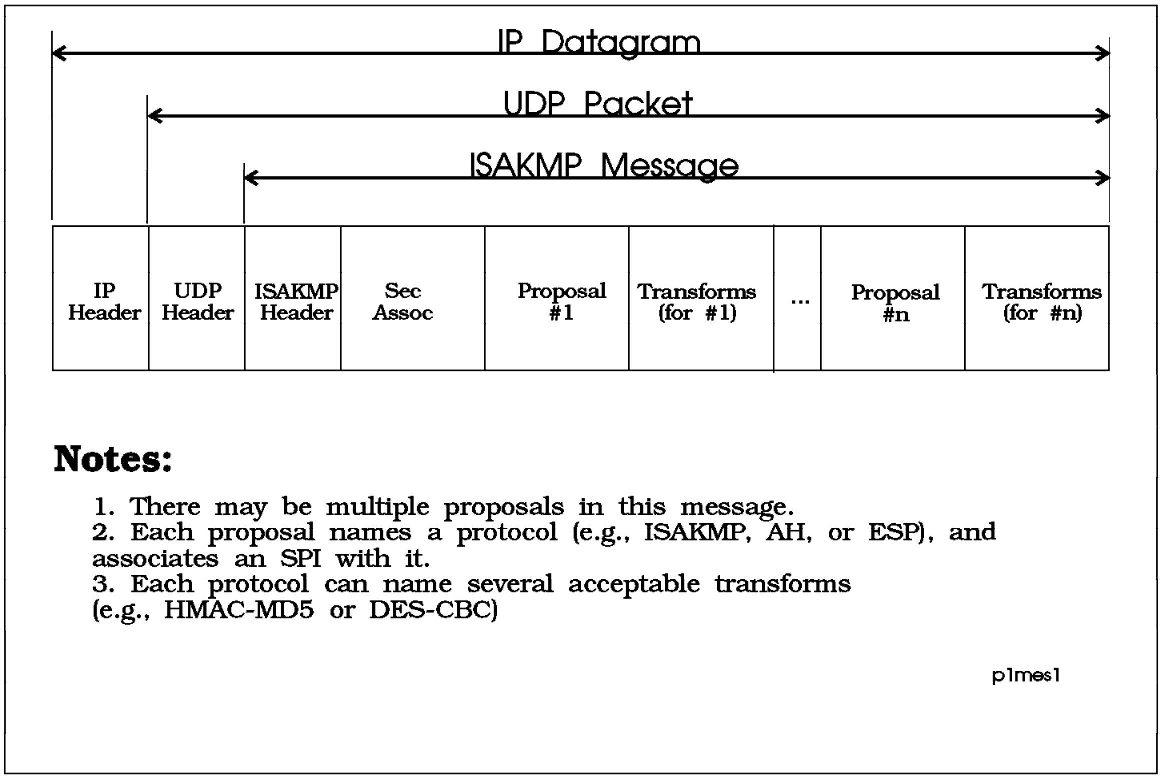
There are other message sequences possible within the ISAKMP framework, but they are not illustrated in this appendix. The interested reader is referred to the Bibliography for technical details.

In the remainder of this discussion, we will assume that the parties involved are named *Host-A* and *Host-B*. Host-A will be the initiator of the ISAKMP Phase 1 exchanges, and Host-B will be the responder. If needed for clarity, subscripts A or B will be used to identify the source of various fields in the message exchanges.

A.1 PHASE 1: SETTING UP THE ISAKMP SA

The security associations that protect the ISAKMP messages themselves are set up during the Phase 1 exchanges. Since we are starting ″cold″, the Phase 1 exchanges will use the ISAKMP Identity Protect exchange (also known as Oakley Main Mode). Six messages are needed to complete the exchange:

- Messages 1 and 2 negotiate the characteristics of the security associations. Messages 1 and 2 flow in the clear for the initial Phase 1 exchange, and they are unauthenticated.
- Messages 3 and 4 exchange nonces and also execute a Diffie-Hellman exchange to establish a master key (SKEYID). Messages 3 and 4 flow in the clear for the initial Phase 1 exchange, and they are unauthenticated.
- Messages 5 and 6 exchange digital signatures, and optionally the pertinent user-based certificates for the purpose of mutually authenticating the parties′ identities. The payloads of Messages 5 and 6 are protected by the encryption algorithm and keying material established with messages 1 through 4.



Notes:

1. There may be multiple proposals in this message.
2. Each proposal names a protocol (e.g., ISAKMP, AH, or ESP), and associates an SPI with it.
3. Each protocol can name several acceptable transforms (e.g., HMAC-MD5 or DES-CBC)

Figure 16. Message 1 of an ISAKMP Phase 1 Exchange. *The initiator sends an unauthenticatible cleartext message that proposes one or more Protection Suites from which the responder can choose.*

A.1.1 ISAKMP Security Associations (Phase 1, Message 1)

Since Host-A is the initiating party, it will construct a cleartext ISAKMP message (Message 1) and send it to Host-B. The ISAKMP message itself is carried as the payload of a UDP packet, which in turn is carried as the payload of a normal IP datagram (see Figure 16).

The source and destination addresses to be placed in the IP header are those of Host-A (initiator) and Host-B (responder), respectively. The UDP header will identify that the destination port is 500, which has been assigned for use by the ISAKMP protocol. The payload of the UDP packet carries the ISAKMP message itself.

In Message 1, Host-A, the initiator, proposes a set of one or more proposed Protection Suites for consideration by Host-B, the responder. Hence, the ISAKMP Message contains at least the following fields in its payload:

- The ISAKMP Header in Message 1 will indicate an exchange type of ″Main Mode″, and will contain a Message ID of ″0″. Host-A will set the ″Responder

Cookie″ field to ″0″, and will fill in a random value of its choice for the ″Initiator Cookie″, which we will denote as ″Cookie-A″.

- The Security Association field identifies the Domain of Interpretation (DOI). Since the hosts plan to run IPSec protocols between themselves, the DOI is simply ″IP″.
- Host-A′s Proposal Payload will specify the protocol ″PROTO_ISAKMP″ and will set the SPI value to ″0″.

  Note: For ISAKMP Phase 1 messages, the actual SPI field within the Proposal Payload is not used to identify the ISAKMP Security Association. During Phase 1, the ISAKMP SA is identified instead by the pair of values <Initiator Cookie, Responder Cookie>, both of which must be non-zero values. Since the Responder Cookie has not yet been generated by Host-B, the ISAKMP SA is not yet unambiguously identified.

- The Transform Payload will specify KEY_OAKLEY. For the KEY_OAKLEY transform, Host-A must also specify the relevant attributes: namely, the authentication method to be used, the pseudo-random function to be used, and the encryption algorithm to be used. Host-A will specify: authentication using digital signatures, a pseudo-random function of HMAC-MD5, and an encryption algorithm of DES-CBC.

A.1.2 ISAKMP Security Associations (Phase 1, Message 2)

In Message 1, Host-A proposes one or more candidate protection suites to be used to protect the ISAKMP exchanges. Host-B uses Message 2 to indicate which one, if any, it will support. Note that in our example, Host-A proposed just a single option, so Host-B merely needs to acknowledge that the proposal is acceptable.

The message contents will be as follows:

- The source and destination addresses to be placed in the IP header are those of Host-B (responder) and Host-A (initiator), respectively. The UDP header will identify that the destination port is 500, which has been assigned for use by the ISAKMP protocol. The payload of the UDP packet carries the ISAKMP message itself.
- The ISAKMP Header in Message 2 will indicate an exchange type of ″Main Mode″, and will contain a Message ID of ″0″. Host-B will set the ″Responder Cookie″ field to a random value, which we will call ″Cookie-B″, and will copy into the ″Initiator Cookie″ field the value that was received in the ″Cookie-A″ field of Message 1. The value pair <Cookie-A, Cookie-B> will serve as the SPI for the ISAKMP Security Association.
- The Security Association field identifies the Domain of Interpretation (DOI). Since the hosts plan to run IPSec protocols between themselves, the DOI is simply ″IP″.

- Host-B′s Proposal Payload will specify the protocol ″PROTO_ISAKMP″ and will set the SPI value to ″0″.

  Note: For ISAKMP Phase 1 messages, the actual SPI field within the Proposal Payload is not used to identify the ISAKMP Security Association. During Phase 1, the ISAKMP SA is identified instead by the pair of values <Initiator Cookie, Responder Cookie>, both of which must be non-zero values.

- The Transform Payload will specify KEY_OAKLEY. For the KEY_OAKLEY transform, the attributes that were accepted from the proposal offered by Host-A are copied into the appropriate fields. That is, Host-B will confirm that it will use: authentication using digital signatures, a pseudo-random function of HMAC-MD5, and an encryption algorithm of DES-CBC.

Properties of ISAKMP SA Established

At this point, the properties of the ISAKMP Security Association have been agreed to by Host-A and Host-B. The identity of the ISAKMP SA has been set equal to the pair <Cookie-A, Cookie-B>. However, the identities of the parties claiming to be Host-A and Host-B have not yet been authoritatively verified.

A.1.3 ISAKMP Security Associations (Phase 1, Message 3)

The third message of the Phase 1 ISAKMP exchange begins the exchange of the information from which the cryptographic keys will eventually be derived (see Figure 17). All information is exchanged in the clear. None of the messages themselves carry the actual cryptographic keys. Instead, they carry inputs that will be used by Host-A and Host-B to derive the keys locally. The ISAKMP payload will be used to exchange two types of information:

- Diffie-Hellman public value: g^x from the initiator. The exponent ″x″ in the public value is the private value that must be kept secret.
- Nonce: N_i from the initiator. (*Nonce* is a fancy name for a value that is considered to be random according to some very strict mathematical guidelines.)

These values are carried in the Key Exchange Payload and the Nonce Payload.

A.1.4 ISAKMP Security Associations (Phase 1, Message 4)

After receiving a Diffie-Hellman public value and a nonce from Host-A, Host-B will respond by sending to Host-A its own Diffie-Hellman public value (g^y from the responder) and its nonce (N_r from the responder).

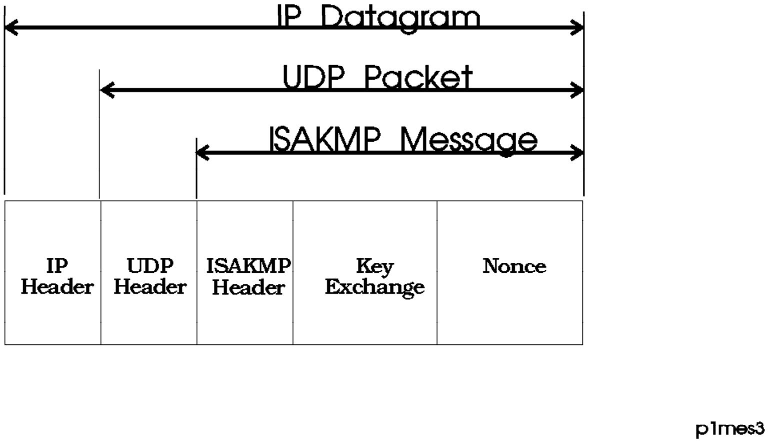


Figure 17. Message 3 of an ISAKMP Phase 1 Exchange. *The initiator sends an unauthenticatible cleartext message that carries a nonce and a Diffie-Hellman public value.*

A.1.5 Generating the Keys (Phase 1)

At this point, each host knows the values of the two nonces (N_i and N_r). Each host also knows its own private Diffie-Hellman value (x and y) and also knows its partner′s public value (g^x or g^y). Hence each side can construct the composite value g^{xy}. And finally, each side knows the values of the Initiator Cookie and the Responder Cookie.

Given all these bits of information, each side can then independently compute identical values for the following quantities:

- SKEYID: This collection of bits is sometimes referred to as ″keying material″, since it provides the raw input from which actual cryptographic keys will be derived later in the process. It obtained by applying the agreed-to pseudorandom function (in our example, HMAC-MD5) to the known inputs:

  $$\text{SKEYID} = \text{HMAC-MD5}(N_i, N_r, g^{xy})$$

- Having computed the value SKEYID, each side then proceeds to generate two cryptographic keys and some additional keying material:
  - SKEYID_d is keying material that will be subsequently used in Phase 2 to derive the keys that will be used in non-ISAKMP SAs for protecting user traffic:

    $$\text{SKEYID\_d} = \text{HMAC-MD5}(\text{SKEYID}, g^{xy}, \text{CookieA}, \text{CookieB}, 0)$$

- SKEYID_a is the key used for authenticating ISAKMP messages:

  SKEYID_a = HMAC-MD5(SKEYID, SKEYID_d, g^{xy}, CookieA, CookieB, 1)

- SKEYID_e is the key used for encrypting ISAKMP exchanges:
  SKEYID_e = HMAC-MD5(SKEYID, SKEYID_a, g^{xy}, CookieA, CookieB, 2)

Keys are Available

At this point in the protocol, both Host-A and Host-B have derived identical authentication and encryption keys that they will use to protect the ISAKMP exchanges. And they have also derived identical keying material from which they will derive keys to protect user data during Phase 2 of the ISAKMP negotiations. However, at this point, the two parties′ identities still have not been authenticated to one another.

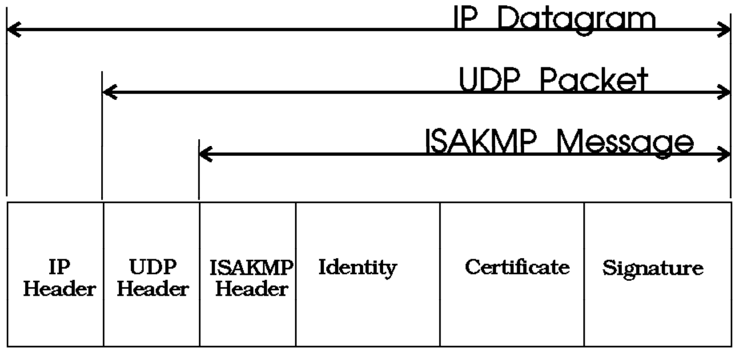
A.1.6 ISAKMP Security Associations (Phase 1, Message 5)

At this point in the Phase 1 flows, the two hosts will exchange identity information with each other, using the Digital Signature Algorithm to authenticate themselves. As shown in Figure 18, the ISAKMP message will carry an Identity Payload, a Signature Payload, and an optional Certificate Payload. Host-A uses Message 5 to send information to Host-B that will allow Host-B to authenticate Host-A.

When an actual certificate is present in the Certificate Payload field, the receiver can use the information directly, after verifying that it has been signed with a valid signature of a trusted Certificate Authority. If there is no certificate in the message, then it is the responsibility of the receiver to obtain a certificate using some implementation method. For example, it may send a query to a trusted Certificate Authority using a protocol such as LDAP, or it may query a secure DNS server, or it may maintain a secure local cache that maps previously used certificates to their respective ″ID″ values, or it may send an ISAKMP Certificate Request message to its peer, who must then immediately send its certificate to the requester. The method for obtaining a certificate is a local option, and is not defined as part of ISAKMP/Oakley. In particular, it is a local responsibility of the receiver to check that the certificate in question is still valid and has not been revoked.

There are several points to bear in mind:

- At this point in the process, all ISAKMP payloads, whether in Phase 1 or Phase 2, are now encrypted, using the encryption algorithm (negotiated in Messages 1 and 2) and the keys (derived from the information in Messages 3 and 4). The ISAKMP header itself, however, is still transmitted in the clear.



Notes:

1. The "Identity" does not necessarily bear any relationship to source IP address
2. However, the "Certificate" must apply to the "Identity"
3. Including a "Certificate" is optional.

Figure 18. Message 5 of an ISAKMP Phase 1 Exchange. *The initiator sends an encrypted, digitally signed message to identify itself to the responder.*

- In Phase 1, IPSec′s ESP protocol is not used: that is, *there is no ESP header.* The recipient uses the Encryption Bit in the Flags field of the ISAKMP header to determine if encryption has been applied to the message. The pair of values <CookieA, CookieB>—which serve as an ″SPI″ for Phase 1 exchanges—provide a pointer to the correct algorithm and key to be used to decrypt the message.
- The Digital Signature is not applied to the ISAKMP message itself. Instead, it is applied to a hash of information that is available to both Host-A and Host-B.
- The identity carried in the Identity Payload does not necessarily bear any relationship to the source IP address; however, the identity carried in the Identity Payload must be the identity to which the Certificate applies.

Since the pseudo-random function for the ISAKMP SA is HMAC-MD5 in our example, Host-A (the initiator) will generate the following hash function, sign it using the Digital Signature algorithm, and then place the result in the Signature Payload field:

$$\text{HASH\_I} = \text{HMAC-MD5}(\text{SKEYID}, g^x, g^y, \text{CookieA}, \text{CookieB}, \text{SA}_p, \text{ID}_A)$$

where ID_A is Host-A′s identity information that was transmitted in the Identity Payload of this message, and SA_p is the entire body of the SA payload that was sent by Host-A in Message 1, including all proposals and all transforms proposed by Host-A. The cookies, public Diffie-Hellman values, and SKEYID were explicitly carried in Messages 1 through 4, or were derived from their contents.

A.1.7 ISAKMP Security Associations (Phase 1, Message 6)

After receiving Message 5 from Host-A, Host-B will verify the identity of Host-A by validating the digital signature. If the signature is valid, then Host-B will send Message 6 to Host-A to allow Host-A to verify the identity of Host-B.

The structure of Message 6 is the same as that of Message 5, with the obvious changes that the Identity Payload and the Certificate Payload now pertain to Host-B. A less obvious difference is that the hash that is signed by Host-B is different from the one previously signed by Host-A:

$$\text{HASH\_R} = \text{HMAC-MD5}(\text{SKEYID}, g^y, g^x, \text{CookieB}, \text{CookieA}, SA_p, ID_B)$$

Notice that the order in which Diffie-Hellman public values and the cookies appear has been changed, and the final term now is the Identity Payload that Host-B has included in Message 6.

Phase-1 is Complete

When Host-A receives Message 6 and verifies the digital signature, the Phase 1 exchanges are then complete. At this point, each participant has authenticated itself to its peer, both have agreed on the characteristics of the ISAKMP Security Association, and both have derived the same set of keys (or keying material).

A.1.8 Miscellaneous Phase 1 Facts

There are several miscellaneous facts worth noting:

1. The Phase 1 message flows above pertain to the case where the ISAKMP/Oakley messages will be authenticated by the Digital Signature Standard. There are other permissible ways to authenticate ISAKMP messages. For example, pre-shared keys or public key encryption could also be used. Regardless of the specific authentication mechanism that is used, there will be six messages exchanged for ISAKMP Main Mode. However, the content of the individual messages will differ, depending on the authentication method. The key calculation formula for SKEYID will also differ depending on the authentication method. The key calculation formulas and outlines of the message contents can be found in

Section 5 of ″draft-ietf-ipsec-isakmp-oakley-04.txt″ (*Resolution of ISAKMP with Oakley*).

2. Although ISAKMP/Oakley exchanges make use of both encryption and authentication, *they do not use either IPSec′s ESP or AH protocol*. ISAKMP exchanges are protected with application-layer security mechanisms, not with Network layer security mechanisms.

3. ISAKMP messages are sent using UDP—there is no guaranteed delivery for them.

4. The only way to identify that an ISAKMP message is part of a Phase 1 flow rather than a Phase 2 flow is to check the Message ID field in the ISAKMP Header. For Phase 1 flows, it must be ′0′, and (although not explicitly stated in the ISAKMP documents), but for Phase 2 flows it must be non-zero.

A.2 PHASE 2: ISAKMP/OAKLEY QUICK MODE

After having completed the ISAKMP/Oakley Phase 1 negotiation process to set up the ISAKMP Security Association, Host-A′s next step is to initiate the ISAKMP/Oakley Phase 2 message exchanges to define the security associations and keys that will be used to protect IP datagrams exchanged between the pair of users. (In the Internet drafts, these are referred to somewhat obtusely as ″non-ISAKMP SAs″). Since the purpose of the Phase 1 negotiations was to agree on how to protect ISAKMP messages, all ISAKMP Phase 2 payloads, but not the ISAKMP header itself, must be encrypted using the algorithm agreed to by the Phase 1 negotiations.

The Oakley Quick Mode will be used for the Phase 2 negotiations for non-ISAKMP SAs. When Oakley Quick Mode is used in Phase 2, authentication is achieved via the use of several cryptographically-based hash functions. The input to the hash functions comes partly from Phase 1 information (SKEYID) and partly from information exchanged in Phase 2. Phase 2 authentication is based on certificates, but the Phase 2 process itself does not use certificates directly. Instead, it uses the SKEYID-a material from Phase 1, which itself was authenticated via certificates.

Quick Mode comes in two forms:

- Without a Key Exchange attribute, Quick Mode can be used to refresh the cryptographic keys, but does not provide the property of Perfect Forward Secrecy (PFS).
- With a Key Exchange attribute, Quick Mode can be used to refresh the cryptographic keys in a way that provides Perfect Forward Secrecy. This is accomplished by including an exchange of public Diffie-Hellman values within messages 1 and 2.

Note: PFS apparently is a property that is very much desired by cryptography experts, but strangely enough, the specs treat PFS as ″optional″. They mandate that a system must be capable of handling the Key Exchange field when it is present in a Quick Mode message, but do not require a system to include the field within the message.

An overview of the three messages within Quick Mode is given below. In our example, we will have Host-A propose two protection suites, and then illustrate how Host-B chooses just one of them:

1. Proposal 1 will offer protection via ESP with DES-CBC for encryption and HMAC-MD5 for authentication
2. Proposal 2 will offer protection via AH with HMAC-MD5 for authentication.

A.2.1 Non-ISAKMP Security Associations (Phase 2, Message 1)

Message 1 of a Quick Mode Exchange allows Host-A to authenticate itself, to select a nonce, to propose security association(s) to Host-B, to execute an exchange of public Diffie-Hellman values, and to indicate if it is acting on its own behalf or as a proxy negotiator for another entity. An overview of the format of Message 1 is shown in Figure 19.

Note: Inclusion of a key exchange field is optional. However, when Perfect Forward Secrecy is desired, it must be present.

Since we have assumed that Host-A and Host-B are each acting on their own behalf, the user identity fields illustrated in Figure 19 will not be present. The message will consist of

- ISAKMP Header: will indicate an exchange type of ″Quick Mode″, will include a non-zero Message-ID chosen by Host-A, will include the initiator and responder cookie values chosen in Phase 1 (i.e., Cookie-A and Cookie-B), and will turn on the Encryption Flag to indicate that the payloads of the ISAKMP message are encrypted according to the algorithm and key negotiated during Phase 1.

- HASH_1: A Hash Payload must immediately follow the ISAKMP header. HASH_1 uses the pseudo-random function that was negotiated during the Phase 1 exchanges; in our example, this is HMAC-MD5. Thus,

  $$\text{HASH\_1} = \text{HMAC-MD5}(\text{SKEYID\_a}, \text{M-ID}, \text{SA}, N_i, \text{KE})$$

  where

  - SKEYID_a was derived from the Phase 1 exchanges
  - M-ID is the Message-ID of this message
  - SA is the Security Association payload carried in this message, including all proposals that were offered

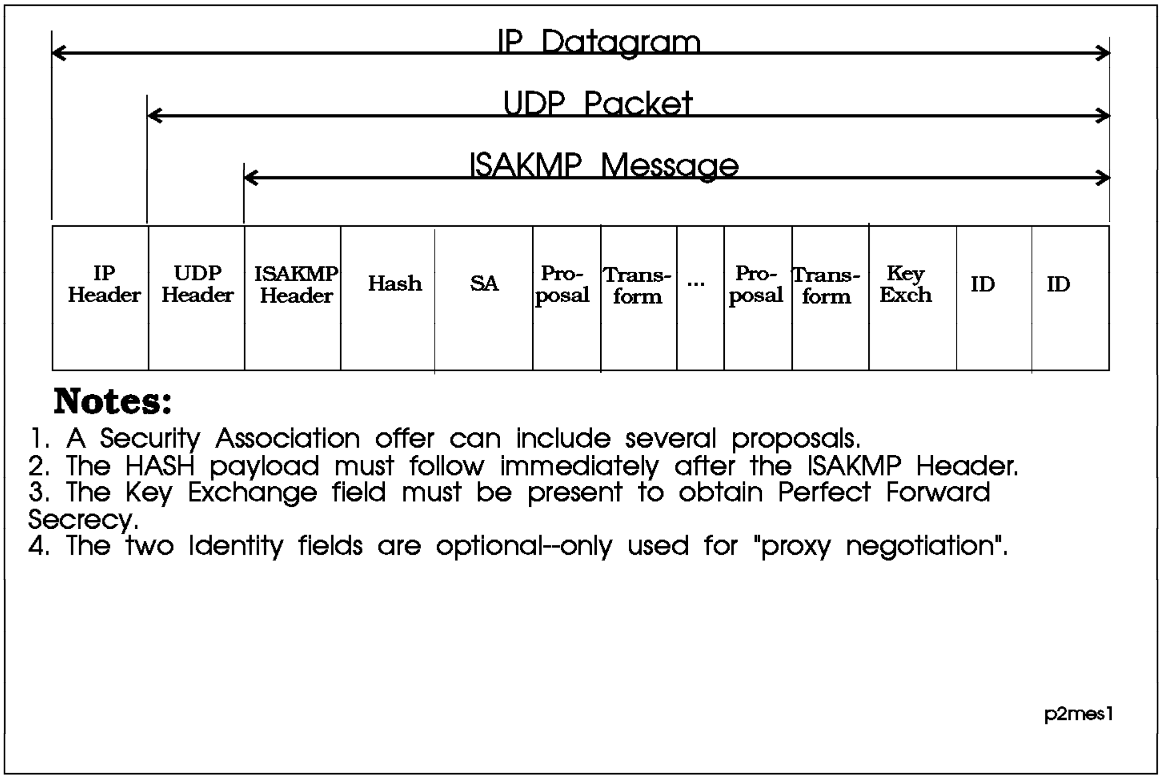

IP Datagram
UDP Packet
ISAKMP Message
IP Header
UDP Header
ISAKMP Header
Hash
SA
Pro-posal
Trans-form
...
Pro-posal
Trans-form
Key Exch
ID
ID
Notes:
1. A Security Association offer can include several proposals.
2. The HASH payload must follow immediately after the ISAKMP Header.
3. The Key Exchange field must be present to obtain Perfect Forward Secrecy.
4. The two Identity fields are optional--only used for "proxy negotiation".
p2mes1

Figure 19. Message 1 of an ISAKMP Phase 2 Quick Mode Exchange. *The initiator offers one or more security associations to the responder.*

– KE is the public Diffie-Hellman value carried in this message. This quantity is chosen by Host-A, and is denoted as g_{qm}^{x}. Note that this is not the same quantity as g^{x} that was used in the Phase 1 exchanges.

- Security Association Payload: indicate ″IP″ as the Domain of Interpretation
- Proposal, Transform Pairs: There will be two of these pairs:

1. First Protection Suite: The first Proposal Payload will be numbered 1, will identify ESP as the protocol to be used, and will include an SPI value that is randomly chosen by Host-A for use with the ESP protocol. The Proposal Payload will be followed by a single Transform Payload that indicates ESP_DES as the algorithm.

   Note: In the current ″Domain of Interpretation″ document, the code point ″ESP_DES″ calls for both encryption with DES-CBC and authentication with HMAC-MD5.

2. Second Protection Suite: The second Proposal Payload will be numbered 2, will identify AH as the protocol to be used, and will include an SPI value that

is randomly chosen by Host-A for use with the AH protocol. The Proposal Payload will be followed by a single Transform payloads that names HMAC-MD5 as the algorithm.

- Nonce: this contains the nonce N_i that was chosen randomly by Host-A
- KE: This is the Key Exchange Payload that will carry the public Diffie-Hellman value chosen by Host-A, g_{qm}^{x}.

  There is also a field called ″Group″, that indicates the prime number and generator used in the Diffie-Hellman exchange. The default group is ″Modular Exponentiation″, using a 768 bit prime number and a generator value of 2 (see ″Resolution of ISAKMP with Oakley″, section 5.7.1). I assume that if the default is acceptable, no ″Group″ payload is needed in the message; if a different ″Group″ is desired, then it must be explicitly carried in the message.

A.2.2 Non-ISAKMP Security Associations (Phase 2, Message 2)

After Host-B receives Message 1 from Host-A and successfully authenticates it using HASH_1, it constructs a reply, Message 2, to be sent back to Host-A. The Message-ID of the reply will be the same one that Host-A used in Message 1. Host-B will choose new values for the following:

- Nonce Payload now carries N_r, a random value chosen by Host-B
- Key Exchange Payload now carries Host-B′s public Diffie-Hellman value, g_{qm}^{y}.
- HASH Payload now carries the value HASH_2, which is defined as:

$$\text{HASH\_2} = \text{HMAC-MD5}(\text{SKEYID\_a}, N_i, \text{M-ID}, \text{SA}, N_r, \text{KE})$$

- Security Payload only describes the single chosen proposal and its associated transforms, not all of the protection suites offered by Host-A. In this example, Host-B will select Proposal #1, which will use ESP_DES (encryption plus authentication) as the protocol. Host-B also chooses an SPI value for the ESP_DES, the selected protocol; Host-B′s SPI does not depend in any way on the SPI that Host-A assigned to that protocol when it offered the proposal. That is, it is not necessary that SPI_A be the same as SPI_B; it is only necessary that they each be non-zero and that they each be randomly chosen.

Keys for non-ISAKMP Can be Derived

At this point, Host-A and Host-B have exchanged nonces and public Diffie-Hellman values. Each one can use this in conjunction with other information to derive a pair of keys, one for each direction of transmission.

A.2.3 Generating the non-ISAKMP Keys

Using the nonces, public Diffie-Hellman values, SPIs, protocol code points exchanged in Messages 1 and 2 of Phase 2 Quick Mode, and the SKEYID value from Phase 1, each host now has enough information to derive two sets of keying material.

- For data generated by Host-A and received by Host-B, the keying material is:
  KEYMAT_{AB} = HMAC-MD5(SKEYID, g_{qm}^{xy}, protocol, SPI_B, N_i, N_r)
- For data generated by Host-B and received by Host-A, the keying material is:
  KEYMAT_{BA} = HMAC-MD5(SKEYID, g_{qm}^{xy}, protocol, SPI_A, N_i, N_r)

Note:

> In this example, Host-A needs to derive four keys:
>
> 1. Key for generating the Integrity Check Value for transmitted datagrams
> 2. Key for validating the Integrity Check Value of received datagrams
> 3. Key for encrypting transmitted datagrams
> 4. Key for decrypting received datagrams
>
> Likewise, Host-B needs to derive the ″mirror image″ of the same four keys: for example, the key that Host-B uses to encrypt its outbound messages is the same key that Host_A uses to decrypt its inbound messages, etc.

A.2.4 Non-ISAKMP Security Associations (Phase 2, Message 3)

At this point, Host-A and Host-B have exchanged all the information necessary for them to derive the necessary keying material. The third message in the Quick Mode exchange is used by Host-A to prove its liveness, which it does by producing a hash function that covers the Message-ID and both nonces that were exchanged in Messages 1 and 2. Message 3 consists only of the ISAKMP Header and a Hash Payload that carries:

HASH_3 = HMAC-MD5(SKEYID_a, 0, M-ID, N_i, N_r)

When Host-B receives this message and verifies the hash, then both systems can begin to use the negotiated security protocols to protect their user data streams.

APPENDIX B. NEGOTIATING MULTIPLE SECURITY ASSOCIATIONS

The example covered in Sections A.2.1 through A.2.4 illustrated a case where a single non-ISAKMP security association was negotiated by means of a Quick Mode message exchange. However, it is also possible to negotiate multiple security associations, each with its own set of keying material, within a single 3-message Quick Mode exchange.

The message formats are very similar to the previously illustrated ones, so only the differences will be highlighted below:

- Message 1 will carry multiple Security Association Payloads, each offering a range of protection suites.
- HASH_1 will cover the entire set of all offered Security Associations carried in Message 1: that is, each Security Association and all of its offered proposals are included.

Caution to Reader

I did not find this definition in the ″Resolution of ISAKMP with Oakley″ document. There, ″SA″ is defined as "..an SA negotiation payload with one or more proposals.." When multiple SAs are being negotiated with a single instance of Quick Mode, I have assumed that the ″SA″ is really the set of all offered SAs.

- In Message 2, for each offered SA, Host-B will select a single protection suite. That is, if ″n″ SAs are open for negotiation, then Host-B will choose ″n″ protection suites, one from each proposal.
- As was the case for HASH_1, HASH_2 will now cover the entire set of all offered Security Associations carried in Message 1: that is, each Security Association and all of its offered proposals are included.
- After Messages 1 and 2 have been exchanged, then Host-A and Host-B will be able to generate the keying material for each of the accepted protection suites, using the same formulas as in section A.2.3, applied individually for each accepted SA. Even though the nonces and the public Diffie-Hellman values are the same for all selected suites, the keying material derived for each selected protection suite will be different because each proposal will have a different SPI.
- Because multiple Security Associations have been negotiated, it is a matter of local choice as to which one is used to protect a given datagram. A receiving system must be capable of processing a datagram that is protected by any SA that has been negotiated. That is, it would be legal for a given source host to send two consecutive datagrams to a destination system, where each datagram was protected by a different SA.

APPENDIX C. BIBLIOGRAPHY

These are the most recent IETF documents that pertain to IPSec:

- *Security Architecture for the Internet Protocol*, draft-ietf-ipsec-arch-sec-02.txt
- *The Internet IP Security Domain of Interpretation for ISAKMP*, draft-ietf-ipsec-ipsec-dos-06.txt
- *Internet Security Association and Key Management Protocol (ISAKMP)*, draft-ietf-ipsec-isakmp-08.ps
- *The Resolution of ISAKMP with Oakley*, draft-ietf-ipsec-isakmp-oakley-05.txt
- *IP Authenticatio Header*, draft-ietf-ipsec-auth-header-03.txt
- *IP Encapsulating Security Payload (ESP)*, draft-ietf-ipsec-esp-v2-02.txt
- *The Oakley Key Determination Protocol*, draft-ietf-ipsec-oakley-02.txt
- *IP Security Documnet Roadmap*, draft-ietf-ipsec-doc-roadmap-02.txt
- *The Use of HMAC-MD5-96 within ESP and AH*, draft-ietf-ipsec-auth-hmac-md5-96-01.txt
- *The Use of HMAC-SHA-1-96 within ESP and AH*, draft-ietf-ipsec-auth-hamc-sha196-01.txt
- *Securing L2TP using IPSec*, draft-ietf-pppext-l2tp-security-00.txt