CHAPTER 9

Encapsulating Security Payload

In this chapter, we discuss the following topics:

- Encapsulating Security Payload (ESP) Format
- ESP Modes of Operation
  - ESP Transport Mode
  - ESP Tunnel Mode
- ESP Processing
  - Outbound Processing
  - Inbound Processing

As is the case with the authentication header (AH), the encapsulating security payload (ESP) is designed to improve the security of the Internet Protocol (IP). ESP provides data confidentiality, data origin authentication, connectionless integrity, antireplay service, and limited traffic flow confidentiality. In essence, ESP offers similar services to those provided by AH, plus two additional services: data confidentiality and a limited traffic flow confidentiality. The confidentiality service is afforded by the use of a cryptographic algorithm to encrypt relevant portions of the IP datagram. Traffic flow confidentiality is provided by the confidentiality service in tunnel mode; we discuss the mode of operation for ESP later in the chapter.

The cryptographic algorithms that ESP uses to encrypt datagrams are exclusively symmetric-key cryptosystems. Public-key cryptographic algorithms involve computationally intensive modular exponentiation of large integers of magnitude greater than 300 decimal digits, whereas symmetric-key ciphers utilize mainly primitive operations (exclusive-OR, bitwise AND, bit rotation, etc.) that are executed very efficiently in both hardware and software. As a result, symmetric-key cryptosystems give significantly greater encryption/decryption throughput compared to that of public-key cryptosystems.

ESP provides authentication service by the use of message authentication codes (MACs). MACs are similar to cryptographic hash functions except that a key is required to generate the message digest. For further detail about MACs, refer to Chapter 4.

The choice of encryption and authentication algorithms varies with different IPSec implementations; however, in order to ensure interoperability, the ESP specification RFC 2406 [KA98] stipulates mandatory algorithms that each implementation must support. At the time of writing, the mandatory-to-implement encryption algorithms were DES in CBC[1] mode and the NULL encryption algorithm, whereas the authentication algorithms were HMAC-MD5, HMAC-SHA-1, and the NULL authentication algorithm. The NULL encryption and authentication algorithms are options for no encryption and no authentication respectively. The NULL algorithm options are mandatory because ESP confidentiality and authentication services are optional. However, it is important to note that the NULL encryption and the NULL authentication algorithms cannot be utilized simultaneously; in other words, if ESP is employed, its confidentiality or its authentication or both must be utilized. DES is a weak encryption algorithm and is rarely used for VPN solutions. It is expected

[1]We discuss the mode of operation of block ciphers in Section 2.3.3.

that revisions of RFC 2406 will replace DES as a mandatory-to-implement algorithm with AES. Other common choices of encryption algorithms are CAST-128 and IDEA; we discussed these ciphers in Chapter 2.

ESP can be applied alone, in a nested fashion, or in combination with AH. In this chapter, we discuss the format of ESP packets, the modes of operation of ESP, and the processing of ESP packets.

9.1 ESP Packet Format

ESP packets consist of four fixed-length fields and three variable-length fields. Figure 9-1 shows the packet format for this protocol. A description of the fields follows:

- *Security parameter index (SPI):* The SPI is a 32-bit integer that is used in combination with the source or destination address and the IPSec protocol (ESP or AH) to uniquely identify the security association (SA) for the traffic to which a datagram belongs. A SA is an agreement between IPSec communicating peers on entities such as the encryption algorithm that will be used to provide ESP confidentiality service, the authentication algorithm, the cryptographic keys, the mode of operation of the IPSec protocol, and the lifetime of the SA; see Section 7.3 for further details on SA. The range of numbers from 1 to 255 is reserved by IANA for future use, and 0 is reserved for local and implementation-specific use. Therefore, the current valid SPI values are between 256 and $2^{32} - 1$. This field is similar to the AH SPI field.

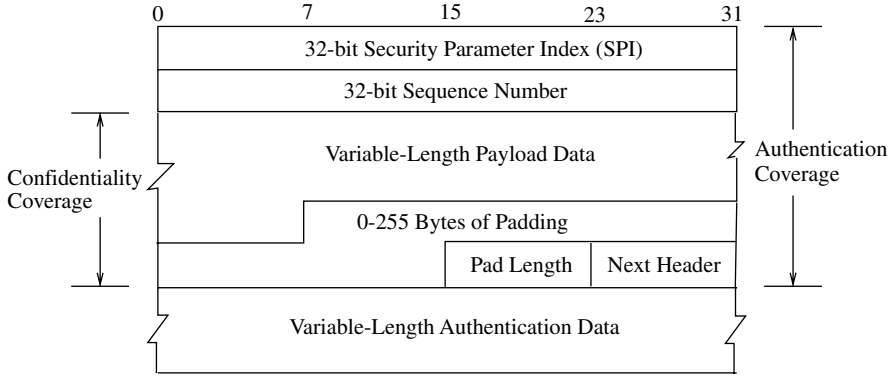

Figure 9-1
ESP Packet Format

- *Sequence number:* As is the case for the AH, this field contains a 32-bit unsigned integer that serves as a monotonically increasing counter. The sender's and receiver's sequence number counters are initialized to 0 when the SA is established. The sender consequently increases its sequence number by 1 for every packet it sends using a given SA. The sequence number is used to prevent intruders from capturing and resending previously transmitted datagrams. Sequence number values are not allowed to be recycled for a given SA; therefore, a new SA and consequently new keys must be negotiated prior to the transmission of 2^{32} packets on a given SA. It is mandatory that the sender transmit the sequence number to the receiver; however, the receiver can choose to disable the antireplay feature, and in so doing, ignore the sequence number field in datagrams associated with incoming traffic. If antireplay is enabled on the receiver host, it uses a sliding receiving window to detect duplicated packets. The specifics of a sliding window vary with different IPSec implementations. In general, the window size should be a minimum of 32 bits. The right edge of the window represents the highest validated sequence number value received on the given SA. Packets that have sequence numbers that are less than the left edge of the window should be rejected. Packets with sequence number values that are within the window should be checked against a list of received packets within the window. If the packets fall within the window, and they are new, or if the sequence number values of the packets are greater than the right edge of the sliding window and less than 2^{32}, the receiver node continues with the processing of the packet. If not, it drops the packet and audits the event.

- *Payload data:* This is a variable-length field that contains the actual payload data (that is, the ciphertext for the encrypted portion of the datagram) if the confidentiality service is utilized. This field is mandatory so it is present whether or not the SA in question requires the confidentiality service. If the encryption algorithm employed requires an initial vector (IV), the IV is transported in the payload data field, and the specification for the algorithm needs to specify the length of the IV and its location in the payload data field. We explained the use of the IV with block ciphers in Section 2.3.3; in brief, initial vectors are employed with block ciphers in certain modes of operation to ensure that the ciphertext resulting from plaintexts that are similar in the first

few bytes—for example, the header of IP datagrams—are different. The length of the payload data field in bits must be an integer multiple of 8.

- *Padding:* This field contains the padding bits—if any—that are utilized by the encryption algorithm, or that are used to align the pad length (see Figure 9-1) field so that it begins at the third byte within the 4-byte word. The length of this field can be between 0 and 255 bytes.
- *Pad length:* The pad length field is a 8-bit field that indicates the number of padding bytes in the padding field. The valid values for this field are integers between 0 and 255.
- *Next header:* This is an 8-bit field that identifies the type of data encapsulated in the payload. It may indicate an IPv6 extension header or a transport layer protocol. For example, a value of 6 indicates that the payload contains TCP data. IANA is the group that is responsible for assigning IP Protocol numbers. The IANA home page is http://www.iana.org.
- *Authentication data:* This is a variable-length field that contains the ICV, which, as indicated by Figure 9-1, is calculated over the length of the ESP packet minus the authentication data field. The actual length of this field depends on the authentication algorithm employed; for example, if HMAC-MD5 is utilized, the length of the authentication data field will be 128 bits, whereas if HMAC-SHA-1 or HMAC-RIPEMD-160 is used, it will be 160 bits. The authentication data field is optional, and it is included only if ESP authentication service is required for the given SA.

9.2 ESP Modes

As is the case with AH, the location of the ESP in the packet depends on the mode of operation of ESP. There are two modes of operations: *transport mode* and *tunnel mode*.

9.2.1 ESP Transport Mode

In transport mode, ESP is inserted after the IP header and any options it contains but before any transport layer protocol, or before any IPSec pro-

tocol that has already been applied. So, for IPv4 in transport mode, ESP is inserted after the variable-length options field. Figure 9-2 shows the position of ESP in transport mode relative to other header fields. In this diagram, the ESP header field consists of the SPI and sequence number fields, whereas the ESP trailer field consists of the padding, pad length, and next header fields. The portions of the datagram that are encrypted or authenticated are indicated in the diagram. If confidentiality service is required, the SPI and the sequence number fields are not encrypted because the receiver node utilizes these fields to identify the SA that should be used to process the datagram and to identify replayed packets if antireplay is enabled, respectively. Similarly, the authentication data field, if present, is not encrypted because if a given SA requires ESP authentication service, the destination host uses this field to verify the integrity of the datagram before it is processed.

For IPv6 datagrams, ESP is inserted after the hop-by-hop, routing, and fragmentation extension headers; the destination options extension header can be placed before or after the ESP header. If the destination option header is to be processed by the first destination that appears in the IPv6 destination address field, plus subsequent destinations listed in the routing header, it should should be placed before ESP. However, if it is to be processed only by the destination node, it can be placed after ESP. Figure 9-3 illustrates the position of the ESP relative to the other IPv6 extension headers for the transport mode of operation.

It is important to note that for ESP authentication service, unlike that for AH, the entire IP datagram is not authenticated; consequently, ESP

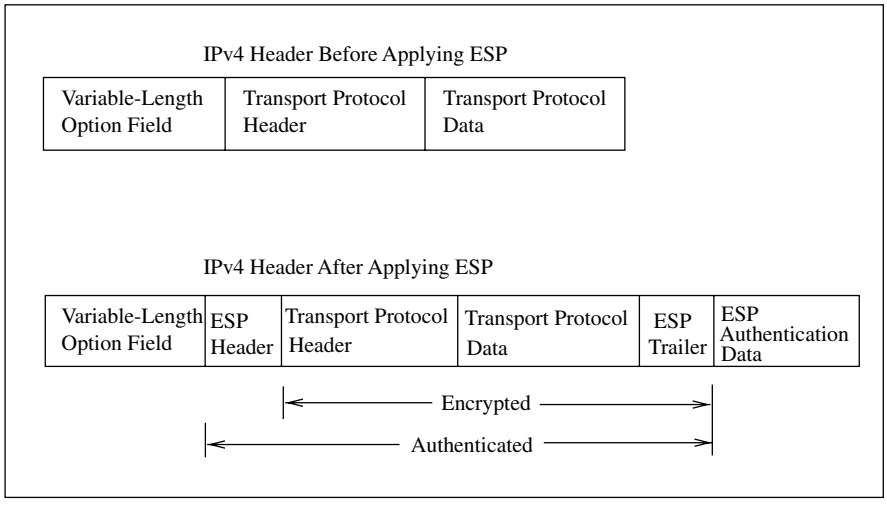

Figure 9-2
ESP Relative to Other IPv4 Header Fields in Transport Mode

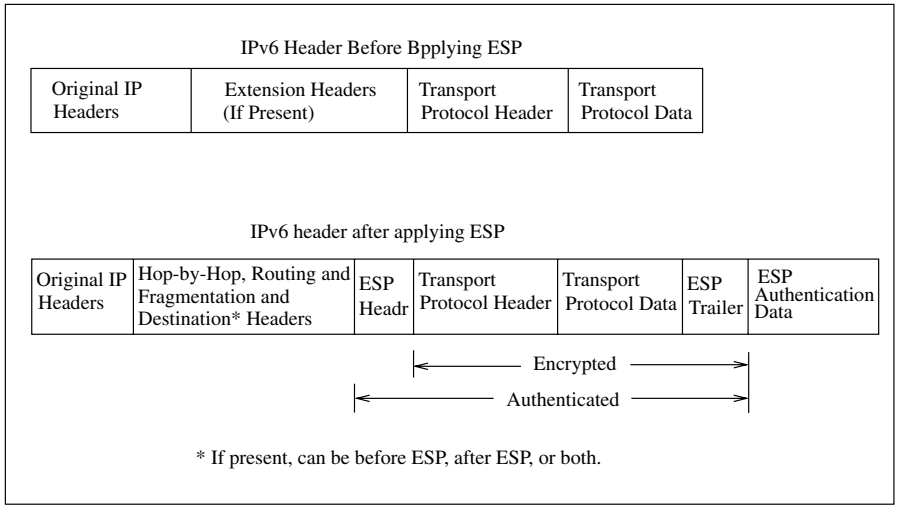

Figure 9-3
ESP Relative to Other IPv6 Extension Headers in Transport Mode

transport mode does not suffer from any of the limitations discussed in Section 8.2.1. The communication between hosts with ambiguous (or private) IP addresses (via the Internet) or that between hosts behind secure gateways can be secured with ESP authentication service because the source and destination IP address fields and other fields in the IP header are not authenticated. Therefore, NAT and security gateways can change relevant IP header fields in a datagram and, provided that the header checksum[2] is recomputed correctly after the modification and none of the ESP header fields are modified, the destination node will successfully authenticate the datagram.

This degree of flexibility that the ESP authentication service offers, however, accounts for its weakness. Apart from the ESP header, any of the IP header fields can be modified while a datagram is in transit from the source node to the destination and, provided that the header checksum is recomputed correctly after the modification (if the SA in question only uses ESP authentication and confidentiality services), the destination host will not detect the modification. ESP transport mode authentication service, therefore, offers less security than that provided by the AH transport mode. Hence, if a high degree of security is needed and the communicating peers have public IP addresses, AH authentication service should be utilized instead of or in conjunction with ESP transport mode authentication service.

[2]We discussed the computation of the IP header checksum in Chapter 7; see this chapter for further details.

It is noteworthy to mention that ESP in transport mode does not offer any traffic flow confidentiality service since the source and destination IP address fields are not encrypted.

9.2.2 ESP Tunnel Mode

In tunnel mode, ESP is inserted before the original IP header, and a new IP header is inserted in front of the ESP header. This is illustrated diagrammatically for IPv4 in Figure 9-4. For IPv6 datagrams, in addition to the new IP header, the extension headers present in the original IPv6 datagram are also inserted in front of the ESP header. Figure 9-5 illustrates this for IPv6 datagrams.

The inner IP header carries the true source (the node that generated the packet) and the final destination address. The outer source and destination IP header fields can carry the source and destination nodes' security gateways, respectively. Consequently, the source addresses in the inner and outer IP headers may be different. The same holds for the destination address.

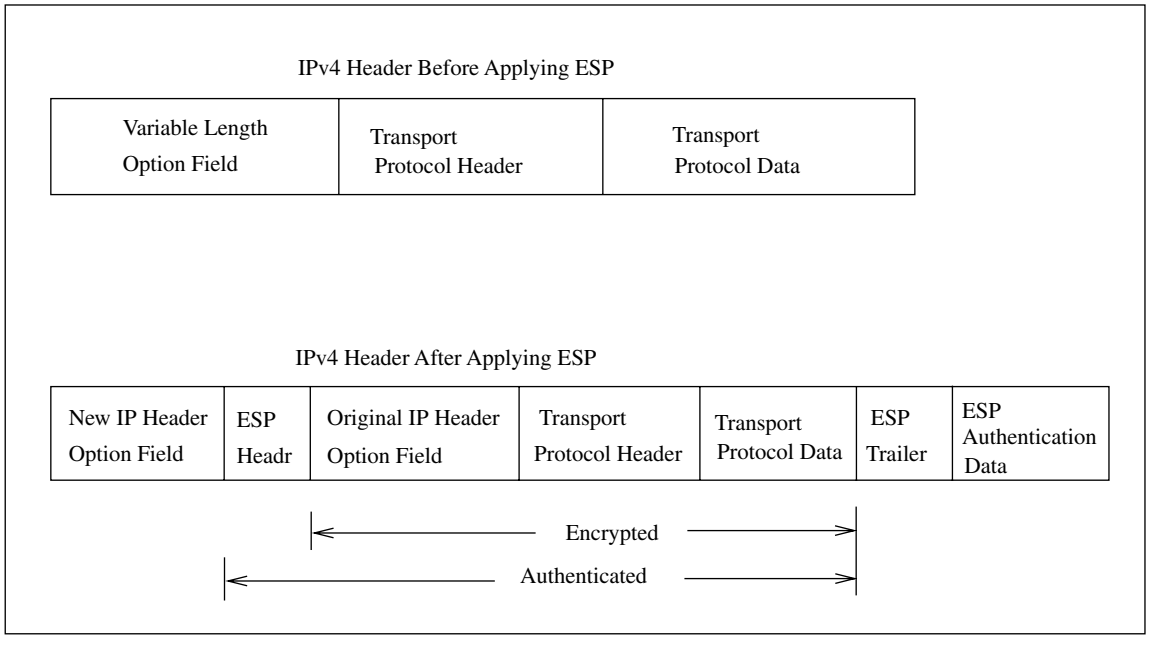

Figure 9-4 *ESP Relative to Other IPv4 Header Fields in Tunnel Mode*

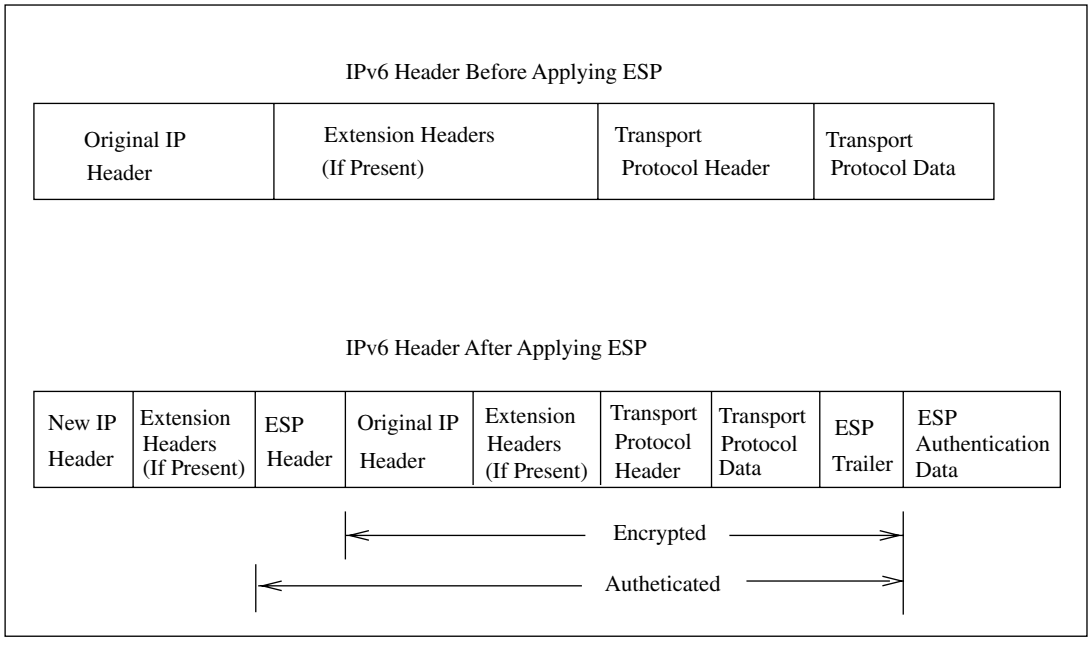

Figure 9-5 ESP Relative to Other IPv6 Extension Headers in Tunnel Mode

As is the case for transport mode, the ESP header field consists of the SPI and sequence number fields, whereas the ESP trailer field consists of the padding, pad length, and next header fields. The portions of the IPv4 and IPv6 datagrams that are encrypted or authenticated are indicated in Figures 9-4 and 9-5, respectively. The ESP header and trailer fields are not encrypted because these fields contain information the destination nodes need to identify the SA for the traffic that the datagrams are a part of or information that might be necessary to process the datagrams prior to them being encrypted.

Note that for the tunnel mode authentication and confidentiality services the entire inner IP header is authenticated and encrypted. However, the outer IP header is neither authenticated nor encrypted. It is not encrypted because routers need the information in it to route the packet. It is not authenticated because if it were, it would suffer from the same limitations, discussed in Sections 8.2.1 and 8.2.2, that are associated with AH authentication service. That is, if the outer IP header were authenticated, ESP tunnel mode authentication service, like that of AH, would not be able to authenticate end-to-end traffic between hosts that are behind

NAT or security gateways that modify the source or destination address fields of packets they forward. The modifications that gateways make to the packets would cause the authentication to fail at the receiver host.

It is important to note that ESP tunnel mode authentication and confidentiality services offer more security than those of ESP transport mode since the former, unlike the latter, authenticate and encrypt the original IP header. However, tunnel mode services utilize more bandwidth than transport mode services because an extra IP header is inserted on datagrams that are protected by tunnel mode services. Therefore, if bandwidth utilization is a big concern, transport mode services might be more suitable.

Although ESP tunnel mode authentication theoretically does not offer as much security as either AH transport or tunnel mode authentication—since it does not authenticate the outer IP header—the security it offers is nonetheless adequate because it is the information in the inner IP header that is used to process the packet.

It is also noteworthy to mention that ESP tunnel mode confidentiality service, particularly when implemented on security gateways, provides confidentiality service for the traffic flow, in that the inner IP header—which contains the IP address from which the packet originated—is encrypted.

9.3 ESP Processing

Let us now look at what is involved in processing packets that are protected with ESP service. The details of processing procedures may vary with different implementations of IPSec; however, in general, the procedure is as outlined below.

9.3.1 Outbound Processing

When an IPSec implementation receives an outbound packet, it uses the relevant selectors (destination IP address and port, transport protocol, etc.) to search the security policy database (SPD) and ascertain what policy is applicable to the traffic. If IPSec processing is required and a SA or SA bundle has already been established, the SPD entry that matches the selectors in the packet will point to the appropriate SA in the security association database (SAD). If a SA has not been established, the IPSec implementation will employ the IKE (Internet key exchange) pro-

tocol to negotiate a SA and link it to the SPD entry. The SA is then used to process the packet as follows:

1. *Generate or increment sequence number:* The sequence number is used to prevent replay of previously transmitted packets. When a new SA is established, the sender initializes its sequence number counter to zero. For each packet that the sender transmits, it increases the sequence number by 1 and inserts the resulting value of the sequence number counter into the sequence number field of the ESP packet.
2. *Encryption of the packet:* If the traffic requires confidentiality services, the SA will specify the encryption algorithm to be used. The available choices of encryption algorithms depend on the IPSec implementation; however, as mentioned previously, only symmetric-key cryptosystems are currently used because of the slow execution speed of public-key cryptosystems compared to symmetric-key ciphers. When the encryption algorithm requires an initial vector (IV), as is the case with DES in CBC,[3] the IV is carried in the first few bytes of the payload data field. When encryption service is required, the packet must be encrypted before the calculation of the integrity check value (ICV).
3. *Calculate the integrity check value:* If the SA for the packet stipulates that ESP authentication service should be applied, the ICV is calculated using the values in all the fields of the ESP header except the authentication data field, which will ultimately store the computed ICV. The SA for the packet specifies the message authentication code (MAC) algorithm that should be used to generate the ICV. The available choices of authentication algorithms vary with different IPSec implementations. However, for interoperability, the ESP specification stipulated that all implementations must support HMAC-MD5 and HMAC-SHA-1. The authentication algorithms require cryptographic keys to generate the ICV. The IKE protocol is responsible for negotiating and establishing necessary cryptographic keys and other SA parameters. We discuss IKE in later chapters.
4. *Fragmentation:* If fragmentation is required, the maximum transfer unit (MTU) of the path from the packet's source to its destination is discovered by suitable means[4] The packet is then broken into the appropriate sizes and sent to the destination node.

[3]We discussed the modes of block ciphers in Section 2.3.3.
[4]For details on path MTU discovery, see RFC 1191.

9.3.2 Inbound Processing

When a packet arrives at an IPSec host or security gateway, if the more fragment (MF) bit is set this is an indication that there are other fragments that are yet to arrive. The IPSec application waits until a fragment arrives with a sequence number that is similar to the previous ones, and has the MF bit not set. It then reassembles the IP fragments and performs the following steps:

1. It uses the destination IP address and IPSec protocol in the IP header (outer IP header if in tunnel mode) and the SPI in the ESP header to look up the SA for the packet in the inbound SAD. If the lookup fails, it drops the packet and audits the event.
2. It uses the SA found in step 1 to process the ESP packet. This involves first checking to determine whether the selectors in the IP headers (inner header if tunnel mode) match those in the SA. If the selectors do not match, the application drops the packet and audits the event. If the selectors match, the IPSec application keeps track of the SA and the order in which it is applied relative to the others, and continues to do steps 1 and 2 until it encounters a transport layer protocol—for IPv4 datagrams, or a non-IPSec extension header—for IPv6 datagrams.
3. It uses the selectors in the packet to find a policy in the inbound SPD whose selectors match those of the packet.
4. It checks whether the SAs found in steps 1 and 2 match the policy specified in step 3. If the check fails, it repeats steps 4 and 5 until all policy entries have been checked or until the check succeeds.
5. If anti-replay is enabled, it uses the anti-replay window of the SA—as discussed earlier in the chapter—to determine if the packet is a replay. If the packet is a replay, it drops the packet and audits the event.
6. If the SA stipulates that authentication service is required, the authentication algorithm and the private key specified by the SA bundle are used to calculate the ICV for the packet and compare it with the value stored in the ESP authentication data field. If the two values differ, the packet is discarded and the event audited.
7. If the SA indicates the confidentiality service is required, the cryptographic algorithm and the key that the SA specifies are utilized to decrypt the packet. Decryption processes are, in general, quite CPU- and memory-intensive. If the IPSec system is allowed to

perform unnecessary decryption or encryption of packets, then the system will be vulnerable to denial of service attack. Consequently, when decryption or encryption is required, this service is applied only after the packet has been successfully authenticated.

At the end of these steps, if the packet have not been discarded, it is then passed to the transport layer protocol or is forwarded to the node indicated in the destination IP address field.