Network Working Group
Internet Draft
draft-ietf-ipsec-new-esp-00.txt
Expire in six months

Stephen Kent, BBN Corp
Randall Atkinson, @Home Network
26 March 1997

IP Encapsulating Security Payload (ESP)

Status of This Memo

This particular Internet Draft is a product of the IETF's IPng and IPsec working groups. It is intended that a future version of this draft be submitted to the IPng Area Directors and the IESG for possible publication as a standards-track protocol.

Table of Contents

1. Introduction

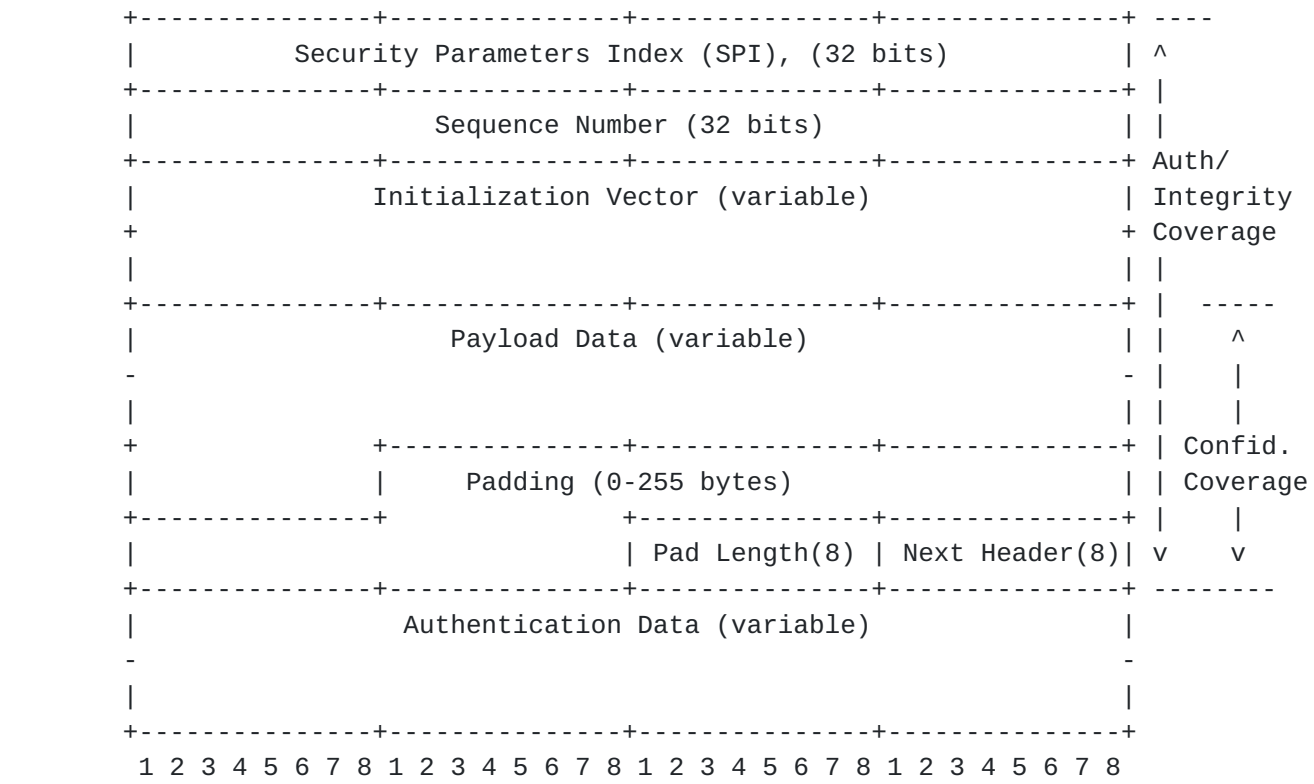
The Encapsulating Security Payload (ESP) header is designed to provide a mix of optional security services in IPv4 and IPv6. ESP may be applied alone, in combination with the IP Authentication Header (AH) [KA97b], or in a nested fashion, e.g., through the use of tunnel mode (see below). Security services can be provided between a pair of communicating hosts, between a pair of communicating security gateways, or between a security gateway and a host. For more details on how to use ESP and AH in various network environments, see "Security Architecture for the Internet Protocol" [KA97a].

The ESP header is inserted after the IP header and before the upper layer protocol header (transport mode) or the encapsulated IP header (tunnel mode). These modes are described in more detail below.

ESP is used to provide confidentiality, data origin authentication, connectionless integrity, anti-replay service (a form of sequence integrity), and limited traffic flow confidentiality. The set of services provided depends on options selected at the time of Security Association establishment and the implementation placement. Confidentiality may be selected independent of all other services. Data origin authentication and connectionless integrity are joint services (hereafter referred to jointly as "authentication), independent of confidentiality. An anti-replay service may be selected only if data origin authentication is selected, but it is independent of confidentiality. Traffic flow confidentiality depends on confidentiality, and requires selection of tunnel mode.

It is assumed that the reader is familiar with the terms and concepts described in the document "Security Architecture for the Internet Protocol" [KA97a]. In particular, the reader should be familiar with the definitions of security services offered by ESP (and by AH), the concept of Security Associations, the different key management options available for ESP (and AH), and the ways in which ESP can be used in conjunction with the Authentication Header (AH).

2. Encapsulating Security Payload Packet Format



The following subsections define the fields in the header format. "Optional" means that the field is omitted if the option is not selected, i.e., it is present in neither the packet as transmitted nor as formatted for computation of an ICV. Whether or not an option is selected is defined as part of Security Association (SA) establishment. Thus the format of ESP packets for a given SA is fixed, for the duration of the SA. In contrast, "mandatory" fields are always present in the ESP packet format, for all SAs.

2.1 Security Parameters Index

The SPI is an arbitrary 32-bit value identifying the Security Association for this datagram (relative to the destination IP address contained in the IP header with which this security header is associated). The set of SPI values in the range 1 through 255 are reserved by the Internet Assigned Numbers Authority (IANA) for future use; a reserved SPI value will not normally be assigned by IANA unless the use of the assigned SPI value is specified in an RFC. A value of zero indicates that no Security Association exists. (Note that the SPI is similar to the SAID used in other security protocols. The name has been changed because the semantics used here are not exactly the same as those used in other security protocols.)

***Under what circumstances will a zero SPI be employed? Is this ***vestigial, or is there still a use for a zero SPI? Is it a SKIP ***support feature of some sort?

The SPI field is mandatory, and its value is ordinarily selected by the destination system upon establishment of an SA (see "Security Architecture for the Internet Protocol" for more details.)

2.2 Sequence Number

This unsigned 32-bit field contains a monotonically increasing counter value (sequence number). The counter is initialized to 1 when an SA is established. The Sequence Number must never be allowed to cycle; thus, it MUST be reset (by establishing a new SA and thus a new key) prior to the transmission of 2^32-1 packets on an SA.

The Sequence Number is optional. It is only included if the anti-replay service is selected as a security service for the SA. Since the anti-replay service requires selection of the authentication service as well, the Sequence Number MUST not be present in the absence of the Authentication Data field (described below.)

2.3 Initialization Vector

This is a variable-length field used only when an explicit IV is required by the selected encryption algorithm, mode or device. The length of the IV is dependent upon the choice of encryption algorithm, and is established during SA negotiation. The IV field is optional, but all implementations must be capable of generating and processing this field if they support algorithms or devices that require an explicit IV.

2.4 Payload Data

Payload Data is a variable-length field containing data described by the Next Header field. This field is an integral number of bytes in length. The Payload Data field is mandatory.

***We have a potential IPv6 alignment problem here, that may have ***been present for some time. Ignoring the presence or absence of an ***iv, the payload data will not be aligned on an 8-byte boundary if ***the Sequence Number is omitted. This may cause a problem for ***efficient crypto data transfer. If the IV is present, and the ***Sequence Number is omitted, the same problem arises, starting with ***the IV, unless the IV is of a compensating size. The decryption ***process can fix the problem for higher layer protocols, because the ***output buffer from decryption is usually distinct from the input

***buffer, but that still causes potential problems for transfer of
***data to the crypto module. Also, if encryption is not employed,
***this becomes a potential problem for authentication data being
***passed up. We could solve this by adding an optional alignment
***field to the ESP header, when required for IPv6. What do people think?

2.5 Padding (for Encryption)

If the confidentiality service has been selected, the Padding field is used to fill the Payload Data to a multiple of the blocksize required by the encryption algorithm. This blocksize requirement is a parameter of the algorithm negotiated during SA establishment.

If encryption has not been selected, the Padding field is used to align the Next Header field so that the last bit of that field ends on a 32-bit boundary.

The Padding bytes SHOULD be initialized with random data and they are transmitted. The transmitter can add 0-255 bytes of padding. Padding beyond that required for encryption algorithm blocksize alignment may be used to conceal the actual length of the payload, in support of traffic flow confidentiality. However, inclusion of such additional padding has adverse bandwidth implications and thus its use should be undertaken with care. The Padding field is optional, but all implementations MUST support generation and consumption of padding.

2.6 Pad Length

The Pad Length field indicates the number of pad bytes immediately preceding it. The range of valid values is 0-255, where a value of zero indicates that the byte immediately preceding the pad length field is the last byte of the payload. The Pad Length field is mandatory.

2.7 Next Header

The Next Header is an 8-bit field that identifies the type of data contained in the Payload Data field, e.g., an extension header in IPv6 or an upper layer protocol identifier. The value of this field is chosen from the set of IP Protocol Numbers defined in the most recent "Assigned Numbers" [STD-2] RFC from the Internet Assigned Numbers Authority (IANA). The Next Header field is mandatory.

2.8 Authentication Data

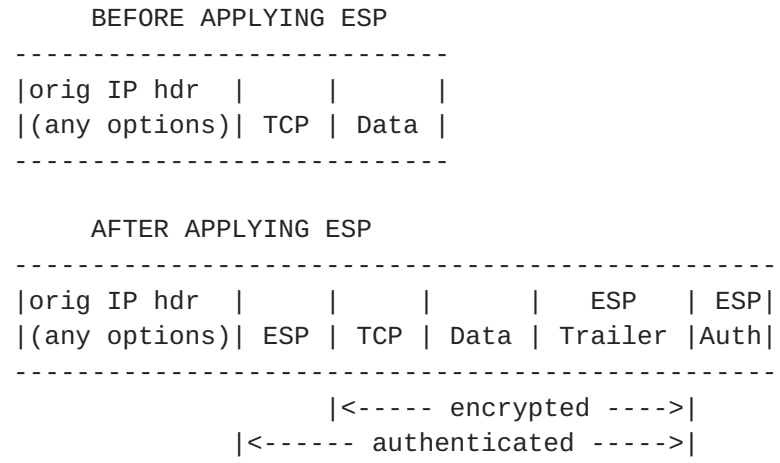
The Authentication Data is a variable-length field containing an Integrity Check Value (ICV) computed over the ESP packet minus the

Authentication Data. The length of the field depends upon the authentication function selected. The mandatory-to-implement authentication algorithms, HMAC with MD5 or SHA-1, both yield 96-bit ICVs because of the truncation convention adopted for use in IPsec. The Authentication Data field is optional.

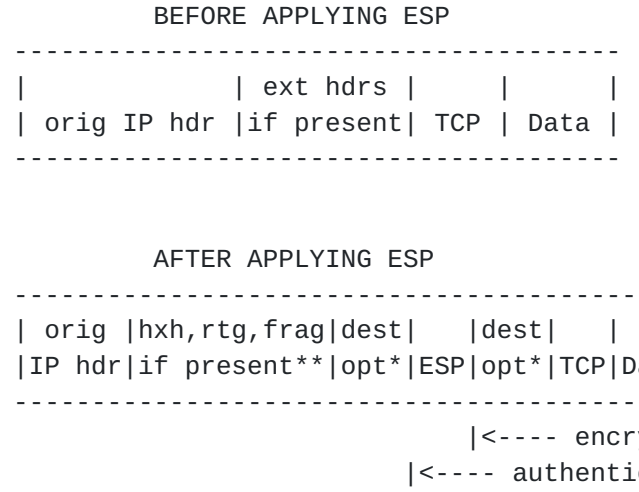
3. Encapsulating Security Protocol Processing

3.1 ESP Header Location

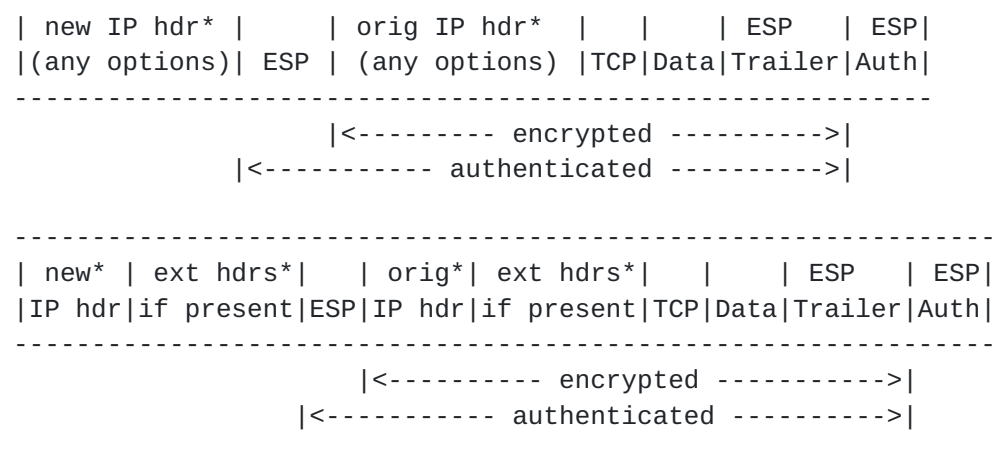
Like AH, ESP may be employed in two ways: transport mode or tunnel mode. The former mode is applicable only to host implementations and provides protection for upper layer protocols, but not the IP header. In this mode, ESP is inserted after the IP header and before an upper layer protocol, e.g., TCP, UDP, ICMP, etc. In the context of IPv4, this translates to placing ESP after the IP header (and any options that it contains), but before the upper layer protocol. (Note that the term "transport" mode should not be misconstrued as restricting its use to TCP and UDP. For example, an ICMP message MAY be sent using either "transport" mode or "tunnel" mode.) The following diagram illustrates ESP transport mode positioning for a typical IPv4 packet, on a "before and after" basis. (The "ESP trailer" encompasses any Padding, plus the Pad Length, and Next Header fields.)



In the IPv6 context, ESP is viewed as an end-to-end payload, and thus should appear after hop-by-hop, routing, and fragmentation extension headers. The destination options extension header(s) could appear either before or after the ESP header depending on the semantics desired. However, since ESP protects only fields after the ESP header, it generally may be desirable to place the destination options header(s) after the ESP header. The following diagram illustrates ESP transport mode positioning for a typical IPv6 packet.



Tunnel mode ESP may be employed in either hosts or security gateways. When ESP is implemented in a security gateway (to protect subscriber transit traffic), tunnel mode must be used. In tunnel mode, the "inner" IP header carries the ultimate source and destination addresses, while an "outer" IP header may contain distinct IP addresses, e.g., addresses of security gateways. In tunnel mode, ESP protects the entire inner IP packet, including the entire inner IP header. The position of ESP in tunnel mode, relative to the outer IP header, is the same as for ESP in transport mode. The following diagram illustrates ESP tunnel mode positioning for typical IPv4 and IPv6 packets.



3.2 Outbound Packet Processing

In transport mode, the transmitter encapsulates the upper layer protocol information in the ESP header/trailer, and retains the specified IP header (and any IP extension headers in the IPv6 context). In tunnel mode, the outer and inner IP header/extensions can be inter-related in a variety of ways. The construction of the outer IP header/extensions during the encapsulation process is described in the document, "Security Architecture for the Internet Protocol".

3.2.1 Security Association Lookup

ESP is applied to an outbound packet only after an IPsec implementation determines that the packet is associated with an SA that calls for ESP processing. The process of determining what, if any, IPsec processing is applied to outbound traffic is described in the document, "Security Architecture for the Internet Protocol".

3.2.2 Anti-replay Service

If the anti-replay service has been selected for this SA, the transmitter increments the Sequence Number for this SA, checks to ensure that the counter has not cycled, and inserts the new value into the Sequence Number field. A transmitter MUST NOT send a packet on an SA if doing so would cause the Sequence Number to cycle.

As mentioned in section 2.2, the anti-replay service requires the selection of the authentication services thus the Sequence Number field MUST NOT be present in the absence of the Authentication Data field (described below.)

3.2.3 Packet Encryption

3.2.3.1 Scope of Encryption

In transport mode, if encryption has been selected, the transmitter encapsulates the original upper layer protocol information into the ESP payload field, adds any necessary padding, and encrypts the result (Payload Data, Padding, Pad Length, and Next Header) using the key, encryption algorithm, and algorithm mode indicated by the SA. In tunnel mode, the transmitter encapsulates and encrypts the the entire original IP datagram (plus the Padding, Pad Length, and Next Header).

If both encryption and authentication have been selected, encryption is performed first, before the authentication and the encryption does

not encompass the Authentication Data field. This order of processing facilitates rapid detection and rejection of replayed or bogus packets by the receiver, prior to decrypting the packet, hence potentially reducing the impact of denial of service attacks. It also allows for the possibility of parallel processing of packets at the receiver, i.e., decryption can take place in parallel with authentication. Note that since the Authentication Data is not protected by encryption, a keyed authentication algorithm must be employed to compute the ICV.

3.2.3.2 Encryption Algorithms

If confidentiality is selected, the encryption algorithm employed is specified by the SA. ESP is designed for use with symmetric encryption algorithms. Because IP packets may arrive out of order, each packet must carry either an explicit Initialization Vector (IV) that allows the receiver to establish cryptographic synchronization for decryption, or data derived from the packet header must suffice to generate an IV at the receiver. Since ESP makes provision for padding of the plaintext, encryption algorithms employed with ESP may exhibit either block or stream mode characteristics.

At the time of writing, one mandatory-to-implement encryption algorithm and mode has been defined for ESP. It is based on the Data Encryption Standard (DES) [NIST77] in Cipher Block Chaining Mode [NIST80]. Details of use of this mode are contained in [need a new I-D with DES-CBC and implicit IV generation, but no overlap with this document].

3.2.4 Integrity Check Value Calculation

3.2.4.1 Scope of Authentication Protection

If authentication is selected for the SA, the transmitter computes the ICV over the ESP packet minus the Authentication Data. Thus the SPI, Sequence Number (if present), Initialization Vector (if present), Payload Data, Padding (if present), Pad Length, and Next Header are all encompassed by the ICV computation. If encryption has been selected, the last 4 fields will be in ciphertext form.

3.2.4.2 Authentication Padding

For some authentication algorithms, the byte string over which the ICV computation is performed must be a multiple of a blocksize specified by the algorithm. If the length of this byte string does not match the blocksize requirements for the algorithm, implicit padding MUST be appended to the end of the ESP packet, prior to ICV

computation. The padding octets MUST have a value of zero. The blocksize (and hence the length of the padding) is specified by the algorithm specification. This padding is not transmitted with the packet.

3.2.4.3 Authentication Algorithms

The authentication algorithm employed for the ICV computation is specified by the SA. For point-to-point communication, suitable authentication algorithms include keyed Message Authentication Codes (MACs) based on symmetric encryption algorithms (e.g., DES) or on one-way hash functions (e.g., MD5 or SHA-1). For multicast communication, one-way hash algorithms combined with asymmetric signature algorithms are suitable. As of this writing, the mandatory-to-implement authentication algorithms are based on the former class, i.e., HMAC [KBC97] with SHA-1 [SHA] or HMAC with MD5 [Riv92]. The output of the HMAC computation is truncated to (the leftmost) 96 bits. Other algorithms, possibly with different ICV lengths, MAY be supported.

3.2.5 Fragmentation

If necessary, fragmentation is performed after ESP processing within an IPsec implementation. However, an IP packet to which ESP has been applied may itself be fragmented by routers en route, including security gateways that may apply AH or ESP (tunnel mode) to the already-protected packet or fragments.

3.3 Inbound Packet Processing

3.3.1 Pre-ESP Processing Overview

If required, reassembly is performed prior to ESP processing.

3.3.2 Security Association Lookup

Upon receipt of a (reassembled) packet containing an ESP Header, the receiver determines the appropriate (unidirectional) SA, based on the destination IP address and the SPI. (This process is described in more detail in the document, "Security Architecture for the Internet Protocol".) The SA indicates whether the Sequence Number, Initialization Vector, and Authentication Data fields should be present, and it will specify the algorithms and keys to be employed for decryption and ICV computations (if applicable).

If no valid Security Association exists for this session (for example, the receiver has no key), the receiver MUST discard the

packet and the failure MUST be recorded in an audit log. The log entry MUST include the SPI value, date/time, Source Address, Destination Address, and (in IPv6) the cleartext Flow ID. The log entry MAY also include other identifying data. There is no requirement for the receiver to transmit any message to the purported transmitter in response to receipt of such packets (because of the potential to induce denial of service via such actions).

3.3.3 Anti-replay Service

If the anti-replay service has been selected for this SA, the receiver MUST verify that the packet contains a Sequence Number value that does not duplicate the Sequence Number of any other packet received during the life of this SA. This SHOULD be the first AH check applied to a packet after it has been matched to an SA, to speed rejection of duplicate packets.

Duplicates are rejected through the use of a sliding receive window. (How the window is implemented is a local matter, but the following text describes the functionality that the implementation must exhibit.) The default window size is 32 and all AH implementations MUST support this window size. A larger window size MAY be established during SA negotiation. If a larger window size is negotiated it MUST be a multiple of 32.

The "right" edge of the window represents the highest, validated Reply Protection value received on this SA. Packets that contain Sequence Numbers lower than the "left" edge of the window are rejected. Packets falling within the window are checked against a list of received packets within the window. An efficient means for performing this check, based on the use of a bit mask, is described in [KA97a].

If the received packet falls within the window, then the receiver proceeds to ICV verification. If the ICV validation fails, the receiver MUST discard the received IP datagram as invalid and MUST record the authentication failure in an audit log. If such a failure occurs, the log entry MUST include the SPI value, date/time received, Sending Address, Destination Address, and (in IPv6) Flow ID. The log data MAY also include other information about the failed packet. The window is updated only if the ICV verification succeeds.

DISCUSSION:

> Note that if the packet is either inside the window and new, or outside the window on the "right" side, the receiver MUST authenticate the Sequence Number before updating the Sequence

Number window data.

3.3.4 Integrity Check Value Verification

If authentication has been selected, the receiver computes the ICV over the ESP packet minus the Authentication Data using the specified authentication algorithm and verifies that it is the same as the ICV included in the Authentication Data field of the packet. Details of the computation are provided below.

If the computed and received ICV's match, then the datagram is valid, and it is accepted. If the test fails, then the receiver MUST discard the received IP datagram as invalid and MUST record the authentication failure in an audit log. The log data MUST include the SPI value, date/time received, Source Address, Destination Address, and (in IPv6) the clear-text Flow ID. The log data MAY also include other information about the failed packet.

DISCUSSION:

Begin by removing and saving the ICV value (Authentication Data field). Next check the overall length of the ESP packet minus the Authentication Data. If implicit padding is required, based on the blocksize of the authentication algorithm, append zero-filled bytes to the end of the ESP packet directly after the Next Header field. Perform the ICV computation and compare the result with the received value. (If a digital signature and one-way hash are used for the ICV computation, the matching process is more complex and will be described in the algorithm specification.)

3.3.5 Packet Decryption

If data confidentiality was selected, the receiver decrypts the ESP Payload Data, Padding, Pad Length, and Next Header using the session key that has been established for this traffic. If an explicit IV is present, it is input to the decryption algorithm as per the algorithm specification. If an implicit IV is employed, a local version of the IV is constructed and input to the decryption algorithm as per the algorithm specification. (Decryption may take place in parallel with authentication, but care must be taken to avoid possible race conditions with regard to packet access and reconstruction of the decrypted packet.)

After decryption, the original IP datagram is reconstructed and processed per the normal IP protocol specification. The exact steps for reconstructing the original datagram depend on the mode (tunnel

vs transport) and are described in the document, "Security Architecture for the Internet Protocol."

Note that there are two ways in which the decryption can "fail". The selected SA may not be correct or the encrypted ESP packet could be corrupted. (The latter case would be detected if authentication is selected for the SA, as would tampering with the SPI. However, an SA mismatch might still occur due to tampering with the IP Destination Address.) In either case, the erroneous result of the decryption operation (an invalid IP datagram or transport-layer frame) will not necessarily be detected by IPsec, and is the responsibility of later protocol processing.

4. Conformance Requirements

Implementations that claim conformance or compliance with this specification MUST implement the ESP syntax and processing described here and MUST comply with all requirements of the "Security Architecture for the Internet Protocol." Note that support for manual key distribution is required, but its use is inconsistent with anti-replay service, and thus a compliant implementation must not negotiate this service in conjunction with SAs that are manually keyed. A compliant ESP implementation MUST support the following mandatory-to-implement algorithms (specified in [KBC97] and in [need a new I-D with DES-CBC and implicit IV generation, but no overlap with this document].

- DES in CBC mode
- HMAC with MD5
- HMAC with SHA-1

5. Security Considerations

Security is central to the design of this protocol, and this security considerations permeate the specification. Additional security-relevant aspects of using IPsec protocol are discussed in the document, "Security Architecture for the Internet Protocol".

Acknowledgements

Many of the concepts embodied in this specification were derived from or influenced by the US Government's SP3 security protocol, ISO/IEC's NLSP, or from the proposed swIPe security protocol. [SDNS89, ISO92 IB93].

For over 2 years, this document has evolved through multiple versions

and iterations. During this time, many people have contributed significant ideas and energy to the process and the documents themselves. The authors would like to thank the members of the IPSEC and IPng working groups, with special mention of the efforts of (in alphabetic order): Steve Bellovin, Steve Deering, Phil Karn, Perry Metzger, David Mihelcic, Hilarie Orman, and William Simpson. In addition, Charlie Lynn, Karen Seo, and Nina Yuan provided extensive help in the review and editing of this version of the specification.

References

[Bel89] Steven M. Bellovin, "Security Problems in the TCP/IP Protocol Suite", ACM Computer Communications Review, Vol. 19, No. 2, March 1989.

[CERT95] Computer Emergency Response Team (CERT), "IP Spoofing Attacks and Hijacked Terminal Connections", CA-95:01, January 1995. Available via anonymous ftp from info.cert.org.

[DH95] Steve Deering & Robert Hinden, Internet Protocol Version 6 (Ipv6) Specification, RFC 1883, December 1995.

[IB93] John Ioannidis & Matt Blaze, "Architecture and Implementation of Network-layer Security Under Unix", Proceedings of the USENIX Security Symposium, Santa Clara, CA, October 1993.

[ISO92] ISO/IEC JTC1/SC6, Network Layer Security Protocol, ISO-IEC DIS 11577, International Standards Organisation, Geneva, Switzerland, 29 November 1992.

[KBC97] Hugo Krawczyk, Mihir Bellare, and Ran Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC-2104, February 1997.

[Ken91] Steve Kent, "US DoD Security Options for the Internet Protocol (IPSO)", RFC-1108, November 1991.

[KA97a] Steve Kent, Randall Atkinson, "Security Architecture for the Internet Protocol", Internet Draft, ?? 1997.

[KA97b] Steve Kent, Randall Atkinson, "IP Authentication Header", Internet Draft, March 1997.

[MS95] Perry Metzger & W.A. Simpson, "The ESP DES-CBC Transform", RFC-1829, August 1995.

[NIST77] US National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 46, January 1977.

[NIST80] US National Bureau of Standards, "DES Modes of Operation" Federal Information Processing Standard (FIPS) Publication 81, December 1980.

[NIST81] US National Bureau of Standards, "Guidelines for Implementing and Using the Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 74, April 1981.

[NIST88] US National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 46-1, January 1988.

[Riv92] Ronald Rivest, MD5 Digest Algorithm, RFC-1321, April 1992.

[SHA] NIST, FIPS PUB 180-1: Secure Hash Standard, April 1995

[STD-2] J. Reynolds and J. Postel, "Assigned Numbers", STD-2, 20 October 1994.

[Sch94] Bruce Schneier, Applied Cryptography, John Wiley & Sons, New York, NY, 1994. ISBN 0-471-59756-2

[SDNS89] SDNS Secure Data Network System, Security Protocol 3, SP3, Document SDN.301, Revision 1.5, 15 May 1989, as published in NIST Publication NIST-IR-90-4250, February 1990.

Disclaimer

The views and specification here are those of the authors and are not necessarily those of their employers. The authors and their employers specifically disclaim responsibility for any problems arising from correct or incorrect implementation or use of this specification.

Author Information

Stephen Kent
BBN Corporation
70 Fawcett Street
Cambridge, MA 02140
USA
Telephone: +1 (617) 873-3988

Randall Atkinson <rja@inet.org>
@Home Network
385 Ravendale Drive
Mountain View, CA 94043
USA