

## IP Encapsulating Security Payload (ESP)

### Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### ABSTRACT

This document describes the IP Encapsulating Security Payload (ESP). ESP is a mechanism for providing integrity and confidentiality to IP datagrams. In some circumstances it can also provide authentication to IP datagrams. The mechanism works with both IPv4 and IPv6.

## 1. INTRODUCTION

ESP is a mechanism for providing integrity and confidentiality to IP datagrams. It may also provide authentication, depending on which algorithm and algorithm mode are used. Non-repudiation and protection from traffic analysis are not provided by ESP. The IP Authentication Header (AH) might provide non-repudiation if used with certain authentication algorithms [[Atk95b](#)]. The IP Authentication Header may be used in conjunction with ESP to provide authentication. Users desiring integrity and authentication without confidentiality should use the IP Authentication Header (AH) instead of ESP. This document assumes that the reader is familiar with the related document "IP Security Architecture", which defines the overall Internet-layer security architecture for IPv4 and IPv6 and provides important background for this specification [[Atk95a](#)].

### 1.1 Overview

The IP Encapsulating Security Payload (ESP) seeks to provide confidentiality and integrity by encrypting data to be protected and placing the encrypted data in the data portion of the IP Encapsulating Security Payload. Depending on the user's security requirements, this mechanism may be used to encrypt either a transport-layer segment (e.g., TCP, UDP, ICMP, IGMP) or an entire IP datagram. Encapsulating the protected data is necessary to provide confidentiality for the entire original datagram.

Use of this specification will increase the IP protocol processing costs in participating systems and will also increase the communications latency. The increased latency is primarily due to the encryption and decryption required for each IP datagram containing an Encapsulating Security Payload.

In Tunnel-mode ESP, the original IP datagram is placed in the encrypted portion of the Encapsulating Security Payload and that entire ESP frame is placed within a datagram having unencrypted IP headers. The information in the unencrypted IP headers is used to route the secure datagram from origin to destination. An unencrypted IP Routing Header might be included between the IP Header and the Encapsulating Security Payload.

In Transport-mode ESP, the ESP header is inserted into the IP datagram immediately prior to the transport-layer protocol header (e.g., TCP, UDP, or ICMP). In this mode bandwidth is conserved because there are no encrypted IP headers or IP options.

In the case of IP, an IP Authentication Header may be present as a header of an unencrypted IP packet, as a header after the IP header and before the ESP header in a Transport-mode ESP packet, and also as a header within the encrypted portion of a Tunnel-mode ESP packet. When AH is present both in the cleartext IP header and also inside a Tunnel-mode ESP header of a single packet, the unencrypted IPv6 Authentication Header is primarily used to provide protection for the contents of the unencrypted IP headers and the encrypted Authentication Header is used to provide authentication only for the encrypted IP packet. This is discussed in more detail later in this document.

The Encapsulating Security Payload is structured a bit differently than other IP payloads. The first component of the ESP payload consist of the unencrypted field(s) of the payload. The second component consists of encrypted data. The field(s) of the unencrypted ESP header inform the intended receiver how to properly decrypt and process the encrypted data. The encrypted data component includes protected fields for the security protocol and also the encrypted encapsulated IP datagram.

The concept of a "Security Association" is fundamental to ESP. It is described in detail in the companion document "Security Architecture for the Internet Protocol" which is incorporated here by reference [[Atk95a](#)]. Implementors should read that document before reading this one.

## 1.2 Requirements Terminology

In this document, the words that are used to define the significance of each particular requirement are usually capitalised. These words are:

- MUST

This word or the adjective "REQUIRED" means that the item is an absolute requirement of the specification.

- SHOULD

This word or the adjective "RECOMMENDED" means that there might exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before taking a different course.

- MAY

This word or the adjective "OPTIONAL" means that this item is truly optional. One vendor might choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

## 2. KEY MANAGEMENT

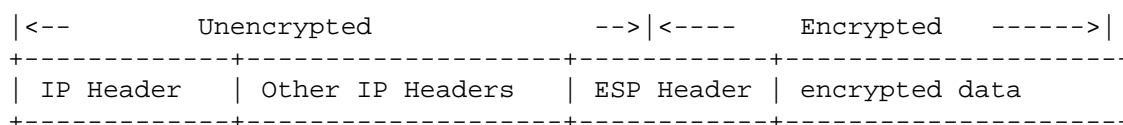
Key management is an important part of the IP security architecture. However, a specific key management protocol is not included in this specification because of a long history in the public literature of subtle flaws in key management algorithms and protocols. IP tries to decouple the key management mechanisms from the security protocol mechanisms. The only coupling between the key management protocol and the security protocol is with the Security Parameter Index (SPI), which is described in more detail below. This decoupling permits several different key management mechanisms to be used. More importantly, it permits the key management protocol to be changed or corrected without unduly impacting the security protocol implementations. Thus, a key management protocol for IP is not specified within this memo. The IP Security Architecture describes key management in more detail and specifies the key management requirements for IP. Those key management requirements are incorporated here by reference [Atk95a].

The key management mechanism is used to negotiate a number of parameters for each security association, including not only the keys but other information (e.g., the cryptographic algorithms and modes,

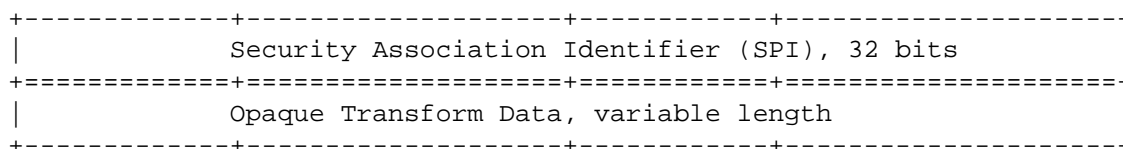
security classification level, if any) used by the communicating parties. The key management protocol implementation usually creates and maintains a logical table containing the several parameters for each current security association. An ESP implementation normally needs to read that security parameter table to determine how to process each datagram containing an ESP (e.g., which algorithm/mode and key to use).

### 3. ENCAPSULATING SECURITY PAYLOAD SYNTAX

The Encapsulating Security Payload (ESP) may appear anywhere after the IP header and before the final transport-layer protocol. The Internet Assigned Numbers Authority has assigned Protocol Number 50 to ESP [STD-2]. The header immediately preceding an ESP header will always contain the value 50 in its Next Header (IPv6) or Protocol (IPv4) field. ESP consists of an unencrypted header followed by encrypted data. The encrypted data includes both the protected ESP header fields and the protected user data, which is either an entire IP datagram or an upper-layer protocol frame (e.g., TCP or UDP). A high-level diagram of a secure IP datagram follows.



A more detailed diagram of the ESP Header follows below.



Encryption and authentication algorithms, and the precise format of the Opaque Transform Data associated with them are known as "transforms". The ESP format is designed to support new transforms in the future to support new or additional cryptographic algorithms. The transforms are specified by themselves rather than in the main body of this specification. The mandatory transform for use with IP is defined in a separate document [KMS95]. Other optional transforms exist in other separate specifications and additional transforms might be defined in the future.

### 3.1 Fields of the Encapsulating Security Payload

The SPI is a 32-bit pseudo-random value identifying the security association for this datagram. If no security association has been established, the value of the SPI field shall be 0x00000000. An 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.

The set of SPI values in the range 0x00000001 through 0x000000FF are reserved to the Internet Assigned Numbers Authority (IANA) for future use. A reserved SPI value will not normally be assigned by IANA unless the use of that particular assigned SPI value is openly specified in an RFC.

The SPI is the only mandatory transform-independent field. Particular transforms may have other fields unique to the transform. Transforms are not specified in this document.

### 3.2 Security Labeling with ESP

The encrypted IP datagram need not and does not normally contain any explicit Security Label because the SPI indicates the sensitivity level. This is an improvement over the current practices with IPv4 where an explicit Sensitivity Label is normally used with Compartmented Mode Workstations and other systems requiring Security Labels [Ken91] [DIA]. In some situations, users MAY choose to carry explicit labels (for example, IPSO labels as defined by RFC-1108 might be used with IPv4) in addition to using the implicit labels provided by ESP. Explicit label options could be defined for use with IPv6 (e.g., using the IPv6 End-to-End Options Header or the IPv6 Hop-by-Hop Options Header). Implementations MAY support explicit labels in addition to implicit labels, but implementations are not required to support explicit labels. Implementations of ESP in systems claiming to provide multi-level security MUST support implicit labels.

## 4. ENCAPSULATING SECURITY PROTOCOL PROCESSING

This section describes the steps taken when ESP is in use between two communicating parties. Multicast is different from unicast only in the area of key management (See the definition of the SPI, above, for more detail on this). There are two modes of use for ESP. The first mode, which is called "Tunnel-mode", encapsulates an entire IP datagram inside ESP. The second mode, which is called "Transport-Mode", encapsulates a transport-layer (e.g., UDP, TCP) frame inside ESP. The term "Transport-mode" must not be misconstrued as restricting its use to TCP and UDP. For example, an ICMP message MAY

be sent either using the "Transport-mode" or the "Tunnel-mode" depending upon circumstance. ESP processing occurs prior to IP fragmentation on output and after IP reassembly on input. This section describes protocol processing for each of these two modes.

#### 4.1 ESP in Tunnel-mode

In Tunnel-mode ESP, the ESP header follows all of the end-to-end headers (e.g., Authentication Header, if present in cleartext) and immediately precedes an tunnelled IP datagram.

The sender takes the original IP datagram, encapsulates it into the ESP, uses at least the sending userid and Destination Address as data to locate the correct Security Association, and then applies the appropriate encryption transform. If host-oriented keying is in use, then all sending userids on a given system will have the same Security Association for a given Destination Address. If no key has been established, then the key management mechanism is used to establish an encryption key for this communications session prior to the use of ESP. The (now encrypted) ESP is then encapsulated in a cleartext IP datagram as the last payload. If strict red/black separation is being enforced, then the addressing and other information in the cleartext IP headers and optional payloads MAY be different from the values contained in the (now encrypted and encapsulated) original datagram.

The receiver strips off the cleartext IP header and cleartext optional IP payloads (if any) and discards them. It then uses the combination of Destination Address and SPI value to locate the correct session key to use for this packet. It then decrypts the ESP using the session key that was just located for this packet.

If no valid Security Association exists for this session (for example, the receiver has no key), the receiver MUST discard the encrypted ESP and the failure MUST be recorded in the system log or audit log. This system log or audit log entry SHOULD include the SPI value, date/time, cleartext Sending Address, cleartext Destination Address, and the cleartext Flow ID. The log entry MAY also include other identifying data. The receiver might not wish to react by immediately informing the sender of this failure because of the strong potential for easy-to-exploit denial of service attacks.

If decryption succeeds, the original IP datagram is then removed from the (now decrypted) ESP. This original IP datagram is then processed as per the normal IP protocol specification. In the case of system claiming to provide multilevel security (for example, a B1 or Compartmented Mode Workstation) additional appropriate mandatory access controls MUST be applied based on the security level of the

receiving process and the security level associated with this Security Association. If those mandatory access controls fail, then the packet SHOULD be discarded and the failure SHOULD be logged using implementation-specific procedures.

#### 4.2 ESP in Transport-mode

In Transport-mode ESP, the ESP header follows the end-to-end headers (e.g., Authentication Header) and immediately precedes a transport-layer (e.g., UDP, TCP, ICMP) header.

The sender takes the original transport-layer (e.g., UDP, TCP, ICMP) frame, encapsulates it into the ESP, uses at least the sending userid and Destination Address to locate the appropriate Security Association, and then applies the appropriate encryption transform. If host-oriented keying is in use, then all sending userids on a given system will have the same Security Association for a given Destination Address. If no key has been established, then the key management mechanism is used to establish a encryption key for this communications session prior to the encryption. The (now encrypted) ESP is then encapsulated as the last payload of a cleartext IP datagram.

The receiver processes the cleartext IP header and cleartext optional IP headers (if any) and temporarily stores pertinent information (e.g., source and destination addresses, Flow ID, Routing Header). It then decrypts the ESP using the session key that has been established for this traffic, using the combination of the destination address and the packet's Security Association Identifier (SPI) to locate the correct key.

If no key exists for this session or the attempt to decrypt fails, the encrypted ESP MUST be discarded and the failure MUST be recorded in the system log or audit log. If such a failure occurs, the recorded log data SHOULD include the SPI value, date/time received, clear-text Sending Address, clear-text Destination Address, and the Flow ID. The log data MAY also include other information about the failed packet. If decryption does not work properly for some reason, then the resulting data will not be parsable by the implementation's protocol engine. Hence, failed decryption is generally detectable.

If decryption succeeds, the original transport-layer (e.g., UDP, TCP, ICMP) frame is removed from the (now decrypted) ESP. The information from the cleartext IP header and the now decrypted transport-layer header is jointly used to determine which application the data should be sent to. The data is then sent along to the appropriate application as normally per IP protocol specification. In the case of a system claiming to provide multilevel security (for example, a

B1 or Compartmented Mode Workstation), additional Mandatory Access Controls MUST be applied based on the security level of the receiving process and the security level of the received packet's Security Association.

#### 4.3. Authentication

Some transforms provide authentication as well as confidentiality and integrity. When such a transform is not used, then the Authentication Header might be used in conjunction with the Encapsulating Security Payload. There are two different approaches to using the Authentication Header with ESP, depending on which data is to be authenticated. The location of the Authentication Header makes it clear which set of data is being authenticated.

In the first usage, the entire received datagram is authenticated, including both the encrypted and unencrypted portions, while only the data sent after the ESP Header is confidential. In this usage, the sender first applies ESP to the data being protected. Then the other plaintext IP headers are prepended to the ESP header and its now encrypted data. Finally, the IP Authentication Header is calculated over the resulting datagram according to the normal method. Upon receipt, the receiver first verifies the authenticity of the entire datagram using the normal IP Authentication Header process. Then if authentication succeeds, decryption using the normal IP ESP process occurs. If decryption is successful, then the resulting data is passed up to the upper layer.

If the authentication process were to be applied only to the data protected by Tunnel-mode ESP, then the IP Authentication Header would be placed normally within that protected datagram. However, if one were using Transport-mode ESP, then the IP Authentication Header would be placed before the ESP header and would be calculated across the entire IP datagram.

If the Authentication Header is encapsulated within a Tunnel-mode ESP header, and both headers have specific security classification levels associated with them, and the two security classification levels are not identical, then an error has occurred. That error SHOULD be recorded in the system log or audit log using the procedures described previously. It is not necessarily an error for an Authentication Header located outside of the ESP header to have a different security classification level than the ESP header's classification level. This might be valid because the cleartext IP headers might have a different classification level after the data has been encrypted using ESP.



## 5. CONFORMANCE REQUIREMENTS

Implementations that claim conformance or compliance with this specification MUST fully implement the header described here, MUST support manual key distribution with this header, MUST comply with all requirements of the "Security Architecture for the Internet Protocol" [Atk95a], and MUST support the use of DES CBC as specified in the companion document entitled "The ESP DES-CBC Transform" [KMS95]. Implementors MAY also implement other ESP transforms. Implementers should consult the most recent version of the "IAB Official Standards" RFC for further guidance on the status of this document.

## 6. SECURITY CONSIDERATIONS

This entire document discusses a security mechanism for use with IP. This mechanism is not a panacea, but it does provide an important component useful in creating a secure internetwork.

Cryptographic transforms for ESP which use a block-chaining algorithm and lack a strong integrity mechanism are vulnerable to a cut-and-paste attack described by Bellovin and should not be used unless the Authentication Header is always present with packets using that ESP transform [Bel95].

Users need to understand that the quality of the security provided by this specification depends completely on the strength of whichever encryption algorithm has been implemented, the correctness of that algorithm's implementation, upon the security of the key management mechanism and its implementation, the strength of the key [CN94] [Sch94, p233] and upon the correctness of the ESP and IP implementations in all of the participating systems.

If any of these assumptions do not hold, then little or no real security will be provided to the user. Use of high assurance development techniques is recommended for the IP Encapsulating Security Payload.

Users seeking protection from traffic analysis might consider the use of appropriate link encryption. Description and specification of link encryption is outside the scope of this note.

If user-oriented keying is not in use, then the algorithm in use should not be an algorithm vulnerable to any kind of Chosen Plaintext attack. Chosen Plaintext attacks on DES are described in [BS93] and [Mat94]. Use of user-oriented keying is recommended in order to preclude any sort of Chosen Plaintext attack and to generally make cryptanalysis more difficult. Implementations SHOULD support user-

oriented keying as is described in the IP Security Architecture [Atk95a].

#### ACKNOWLEDGEMENTS

This document benefited greatly from work done by Bill Simpson, Perry Metzger, and Phil Karn to make general the approach originally defined by the author for SIP, SIPP, and finally IPv6.

Many of the concepts here are derived from or were influenced by the US Government's SP3 security protocol specification, the ISO/IEC's NLSP specification, or from the proposed swIPe security protocol [SDNS89, ISO92a, IB93, IBK93, ISO92b]. The use of DES for confidentiality is closely modeled on the work done for the SNMPv2 [GM93]. Steve Bellovin, Steve Deering, Dave Mihelcic, and Hilarie Orman provided solid critiques of early versions of this memo.

#### REFERENCES

- [Atk95a] Atkinson, R., "Security Architecture for the Internet Protocol", RFC 1825, NRL, August 1995.
- [Atk95b] Atkinson, R., "IP Authentication Header", RFC 1826, NRL, August 1995.
- [Bel89] Steven M. Bellovin, "Security Problems in the TCP/IP Protocol Suite", ACM Computer Communications Review, Vol. 19, No. 2, March 1989.
- [Bel95] Steven M. Bellovin, Presentation at IP Security Working Group Meeting, Proceedings of the 32nd Internet Engineering Task Force, March 1995, Internet Engineering Task Force, Danvers, MA.
- [BS93] Eli Biham and Adi Shamir, "Differential Cryptanalysis of the Data Encryption Standard", Springer-Verlag, New York, NY, 1993.
- [CN94] John M. Carroll & Sri Nudiaty, "On Weak Keys and Weak Data: Foiling the Two Nemeses", Cryptologia, Vol. 18, No. 23, July 1994. pp. 253-280
- [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.

- [DIA] US Defense Intelligence Agency (DIA), "Compartmented Mode Workstation Specification", Technical Report DDS-2600-6243-87.
- [GM93] Galvin J., and K. McCloghrie, "Security Protocols for version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1446](#), Trusted Information Systems, Hughes LAN Systems, April 1993.
- [Hin94] Bob Hinden (Editor), Internet Protocol version 6 (IPv6) Specification, Work in Progress, October 1994.
- [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.
- [IBK93] John Ioannidis, Matt Blaze, & Phil Karn, "swIPE: Network-Layer Security for IP", presentation at the Spring 1993 IETF Meeting, Columbus, Ohio.
- [ISO92a] ISO/IEC JTC1/SC6, Network Layer Security Protocol, ISO-IEC DIS 11577, International Standards Organisation, Geneva, Switzerland, 29 November 1992.
- [ISO92b] ISO/IEC JTC1/SC6, Network Layer Security Protocol, ISO-IEC DIS 11577, [Section 13.4.1](#), page 33, International Standards Organisation, Geneva, Switzerland, 29 November 1992.
- [Ken91] Kent, S., "US DoD Security Options for the Internet Protocol", [RFC 1108](#), BBN Communications, November 1991.
- [KMS95] Karn, P., Metzger, P., and W. Simpson, "The ESP DES-CBC Transform", [RFC 1829](#), Qualcomm, Inc., Piermont, Daydreamer, August 1995.
- [Mat94] Matsui, M., "Linear Cryptanalysis method for DES Cipher", Proceedings of Eurocrypt '93, Berlin, Springer-Verlag, 1994.
- [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.
- [STD-2] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700, USC/Information Sciences Institute, 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 author and are not necessarily those of his employer. The Naval Research Laboratory has not passed judgement on the merits, if any, of this work. The author and his employer specifically disclaim responsibility for any problems arising from correct or incorrect implementation or use of this specification.

## AUTHOR'S ADDRESS

Randall Atkinson  
Information Technology Division  
Naval Research Laboratory  
Washington, DC 20375-5320  
USA

Phone: (202) 404-7090  
Fax: (202) 404-7942  
EMail: atkinson@itd.nrl.navy.mil