Network Working Group
Internet Draft
expires in six months

W A Simpson, Editor
[DayDreamer]

July 1997

IP Encapsulating Security Payload (ESP) for Implementors

draft-simpson-esp-v2-00.txt

Status of this Memo

Abstract

This document describes a confidentiality mechanism for IP datagrams. Payload headers are encapsulated within an opaque envelope. Under some circumstances, authentication and integrity are optionally provided for IP datagrams.

1. Introduction

The Encapsulating Security Payload (ESP) provides confidentiality for IP datagrams by encrypting the payload data to be protected. Depending on the user's security requirements, either an upper protocol-layer segment (such as TCP or UDP) is encrypted (transport-mode), or an entire IP datagram is encrypted and tunneled to the destination within another IP datagram (tunnel-mode).

To work properly without changing the entire Internet infrastructure (particularly non-participating routers), ESP is carried in a datagram with transparent IP headers. The information in these outer IP headers is used to route the protected datagram.

For more details about ESP in various network environments, see "Security Architecture for the Internet Protocol" [RFC-1825x]. That document provides important background for this specification, such as an explanation of Security Associations, forms of Security Association management, guidelines on transport and tunnel modes of operation, and interaction between ESP and the Authentication Header (AH) [RFC-1826x].

ESP may optionally provide authentication and integrity. Users desiring authentication and/or integrity without confidentiality should use AH instead of ESP.

ESP may also provide a form of anti-replay service, depending upon the selection of integrity. The enforcement of replay protection is solely at the discretion of the receiver.

Security services can be provided between:

- a pair of single user hosts,

- a single user host and a security firewall router, or

- a pair of firewall routers.

In the latter case, together with the tunnel mode of encapsulation, ESP may provide traffic flow confidentiality. Traffic aggregation at the firewalls may be able to mask source to destination patterns of the protected internal users.

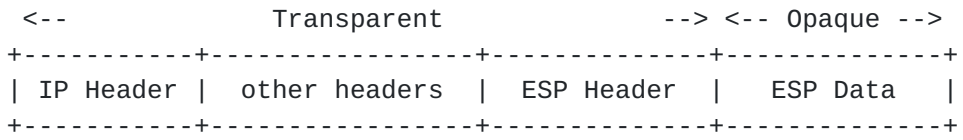
In this document, the key words "MAY", "MUST", "optional", "recommended", "required", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [RFC-2119].

1.1. Performance

Use of this specification will increase the protocol processing costs in participating systems, and will also increase the communications latency. The increased latency is primarily due to time required for encryption and decryption of each datagram. This can be very time intensive to implement.

1.2. Construction

ESP will always appear as the final payload header. The header immediately preceding the ESP Header will contain the value 50 (0x32) in its Next Header (Protocol) field.



The Encapsulating Security Payload has two components.

The transparent ESP Header consists of the unencrypted field(s) of the payload. The transparent field(s) of the unencrypted ESP Header inform the intended recipient how to properly process the opaque data.

The opaque ESP Data consists of protected fields for the ESP transform(s).

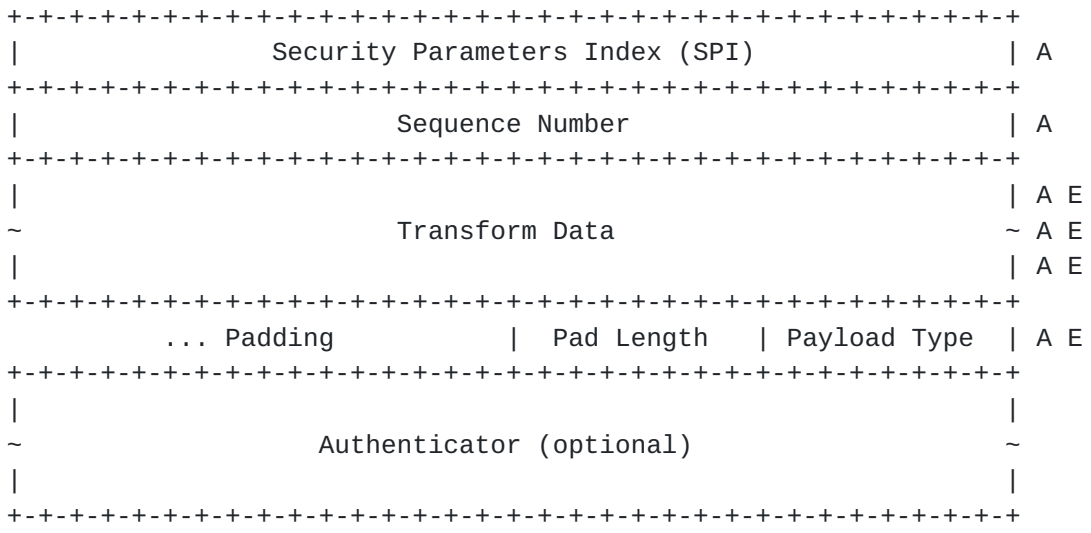
1.3. Transforms

Cipher and authenticator algorithms, and the precise format of opaque ESP data associated with them, are known as "ESP transforms". It is intended that the ESP format should be sufficiently general to permit the specification of new transforms as new cryptographic algorithms are developed.

The parameter description requirements for companion transform documents are described in [RFC-ffff].

When the authenticator transform requires a separate key, that key is generated after any cipher keys.

2. Field Format



All fields are transmitted in network order (most significant byte first).

Fields that may be authenticated are designated by a trailing A.

Fields that may be encrypted are designated by a trailing E.

2.1. Security Parameters Index (SPI)

The SPI is a 32-bit (4 byte) unsigned value identifying the Security Association parameters for the ESP transform. The value is relative to the IP Destination in the preceding IP Header of the datagram.

The use of this value is orthogonal to usage of similar values by other related security protocols, such as the Authentication Header (AH). That is, the same value MAY be used by multiple protocols to concurrently indicate different Security Association parameters.

The value zero (0) indicates that no Security Association has been established, and is primarily used for testing.

Values in the range 1 through 255 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 specification is openly available and documented in the RFC publication series.

Values in the range 256 through 65535 are recommended for manual configuration.

Remaining values are utilized by automated Security Association management. These values are recommended to be generated by a cryptographically random method, to protect against replay attacks and traffic analysis.

This field is mandatory and transparent. That is, the field is always present, and the value is not concealed by encryption.

Rationale:

Even when many SPIs are part of the same Security Association, there is no numerical relation between SPIs in different directions, and no requirement that SPIs be defined in pairs or other multiples.

It should be understood that the same SPI value for different protocols will indicate different parameters. For example, one might indicate a cipher algorithm, and another an authenticator algorithm.

The requirement for SPI orthogonality between protocols arises whenever the values are assigned for multiple protocols in the same transaction by automated or manual configuration using a single common value, or the values are assigned independently by an outside agent (such as a Key Distribution Center). Orthogonality may also be desirable in future multicast implementations.

A small range of values for manual configuration is utilized to promote ease of configuration and interoperability. Experience has shown that large random numbers are not easily remembered and checked by human operators.

This does not preclude automated configuration from utilizing unused values in the reserved and/or recommended manual configuration ranges during operation.

2.2. Sequence Number

The Sequence Number is a 32-bit (4 byte) unsigned counter. This field protects against replay attacks, and may also be used for synchronization by stream or block-chaining ciphers.

Manual configuration can only detect replay of consecutive duplicate Seqeunce Numbers, and during short runs of Sequence Numbers within the round trip time for the parties. The limited anti-replay security of the sequence of datagrams depends upon the unpredictability of the values. When configured manually, the first value sent SHOULD

be a random number.

Long term replay prevention requires automated configuration. When configured via an automated Security Association management protocol, the first value sent is 1, unless otherwise negotiated.

Thereafter, the value is monotonically increased for each datagram sent. A replacement SPI SHOULD be established before the value rolls over and repeats.

This field is mandatory and transparent. That is, the field is always present, and the value is not concealed by encryption.

Although sending this field is mandatory, verification of the sequence of values is at the discretion of the receiver. When integrity checking is available, either through the optional Authenticator field or an external Authentication Header (AH), the implementation SHOULD NOT accept duplicate values. This may be achieved by accepting only those datagrams that contain a different value than previously received, or by maintaining a small window of acceptable values. See Operational Considerations.

Rationale:

> To prevent replay of payloads by substitution of a fresh Sequence Number, anti-replay must be coupled with authentication and/or integrity.
>
> Less than 2**32 datagrams are sent under any single key when anti-replay protection is enforced.

2.3. Transform Data

An implementation will use the SPI to determine the Transform Data contents and use. It retains the same general format for all datagrams of any given IP Destination and SPI. The length of this field is variable, but is always an integral number of bytes.

If the transform requires a separate synchronization field, then such data SHOULD be carried at the beginning of this field.

This field is optional and opaque. That is, the field is not interpretable without knowledge of the Security Association parameters. Refer to each Security Transform specification for more information regarding the contents of this field.

2.4. Padding

If a cipher algorithm requires the plaintext to be a multiple of some number of bytes (such as the block size of a block cipher), the Padding field is used to fill the plaintext to the size required by the algorithm. All implementations MUST support generation and consumption of such padding.

In addition, padding may be used to conceal the actual length of the plaintext. However, inclusion of such additional padding has adverse bandwidth implications and thus its use should be undertaken with care.

Finally, when the Authenticator field is present, padding also may be required to ensure that the resulting ciphertext terminates on a 32-bit (4 byte) boundary.

Prior to encryption, this field is filled with a series of integer values, to align the Pad Length and Payload Type fields at the end of the required boundary (measured from the beginning of the Transform Data). By default, Self-Describing-Padding is used [RFC-1790]. Each byte of padding contains the index of that byte. The first pad byte contains the value one (1). The final pad byte indicates the number of pad bytes to remove. For example, three pad bytes would contain the values 1, 2, and 3.

After decryption, this field MAY be examined for a valid series of integer values. Verification of the sequence of values is at the discretion of the receiver.

This field is optional and opaque. That is, the value (when present) is set prior to encryption, and is examined only after decryption.

Rationale:

> The default padding values were selected for simplicity, ease of implementation in hardware, and compatibility with other internet security efforts.

2.5. Pad Length

The Pad Length (1 byte) indicates the amount of Padding immediately preceding it. The value does not include the Pad Length and Payload Type fields.

The range of valid values is 0 through 255. A value of zero indicates that no Padding is present.

This field is mandatory and opaque. That is, the value is set prior to encryption, and is examined only after decryption.

Rationale:

> This extra field is appended to Self-Describing-Padding for compatibility with earlier implementations that used random padding values.

2.6. Payload Type

The Payload Type (1 byte) indicates the contents of the Transform Data field, using the IP Next Header (Protocol) value. Up-to-date values of the IP Next Header (Protocol) are specified in the most recent "Assigned Numbers" [RFC-1700].

For example, when encrypting an entire IP datagram (Tunnel-Mode), this field will contain the value 4, indicating IP-in-IP encapsulation.

This field is mandatory and opaque. That is, the value is set prior to encryption, and is examined only after decryption.

2.7. Authenticator

When the ESP data is not otherwise validated (externally using AH or internally by the transform algorithm itself), it is recommended (but not required) that an Integrity Check Value (ICV) be provided here. The ICV is computed over the ESP data after encryption, beginning with the SPI and ending with the Payload Type. A keyed algorithm must be employed to compute the ICV. The length of the field depends upon the authenticator transform selected.

For some authenticator transforms, the bytes over which the computation is performed must be a multiple of a block size specified by the algorithm, or the block must be "strengthed" by appending a length. Implicit padding is appended to the end of the ESP packet, prior to Authenticator computation. The size and form of the padding is specified by the transform specification. This implicit padding is not transmitted with the datagram.

This field is optional and opaque. That is, the field (when present) is not concealed by encryption, but is not interpretable without knowledge of the Security Association parameters.

Rationale:

The order of processing (authentication "outside" encryption) facilitates rapid detection of bogus datagrams by the receiver prior to decrypting, potentially reducing the impact of denial of service attacks. It also allows for the possibility of parallel processing at the receiver; decryption can take place in parallel with authentication, but care must be taken to avoid race conditions for packet access and reconstruction of the decrypted packet.

3. Processing

The Security Parameters Index (SPI) is the only coupling between ESP and Security Association management mechanisms. This permits different management mechanisms to be used concurrently. More importantly, it permits any automated management protocol to be changed or corrected without unduly impacting the security protocol implementations. In order to facilitate early adoption, manual configuration is the only mechanism required by this specification.

The Security Association management mechanism specifies a number of parameters for each Security Association between the communicating parties. Manually configurable parameters are summarized in "Operational Considerations".

These Security Association determinations are described in more detail in the Security Architecture document [RFC-1825].

3.1. Outgoing

ESP is applied to an outgoing datagram only after an implementation determines that an existing Security Association calls for ESP processing, based on the IP Destination. If not, then the Security Association management mechanism is used to establish the SPI for this communication session prior to the use of ESP.

The indicated SPI begins the ESP Header.

3.1.1. Compression

When configured, perform data compression of the payload. If expansion occurs, the outgoing SPI may be changed to a value that indicates unexpanded data.

3.1.2. Sequence Number

Check to ensure that the Sequence Number for this SPI has not overflowed. If the implementation detects that 2**32 datagrams have already been sent, the datagram is discarded, and an auditable event is indicated.

Otherwise, insert the next value into the Sequence Number field.

3.1.3. Padding

Append zero or more bytes of padding to the plaintext, as required by the transform.

Append a Pad Length byte containing the number of padding bytes just added.

For example, if the plaintext length is 41, and the block size is 64-bits (8 bytes), padding is needed to make its modulo 8 length equal to 6, leaving 2 bytes for the Pad Length and Payload Type.

The padding values are 1, 2, 3, 4, 5, and the following Pad Length is 5.

3.1.4. Payload Type

Append a Payload Type byte containing the IP Next Header (Protocol) value which identifies the protocol header that begins the payload.

3.1.5. Encryption

Provide an Initialization Variable (IV) of the form indicated by the cipher transform specification.

Encrypt the plaintext, Padding, Pad Length and Payload Type, producing a ciphertext in the form indicated by the cipher transform specification.

3.1.6. Authentication

When configured, calculate and append the optional Authenticator in the form indicated by the authenticator transform specification.

3.1.7. Completion

Construct an appropriate IP datagram for the target Destination.

The IP Total/Payload Length is adjusted to reflect the length of the encrypted payload, with the SPI, Sequence Number, and optional Authenticator.

3.2. Incoming

Upon receipt of a (reassembled) incoming datagram containing an ESP Header, the implementation determines the appropriate Security Association, based on the IP Destination and the SPI. If the SPI is invalid, then the datagram is discarded, and the "Bad SPI" error is indicated.

The SPI field is used as an index into the local Security Association table to find the negotiated parameters and key(s).

3.2.1. Sequence Number

When replay checking is enabled, ensure that the Sequence Number is in the appropriate window for this SPI, and that it has not been previously received.

If the Sequence Number appears to be valid, then the implementation proceeds to authentication. The receive window is updated only when authentication, decryption and decompression succeed.

If the Sequence Number is invalid, then the datagram is discarded, and the "Authentication Failed" error is indicated.

3.2.2. Authentication

When present, remove and verify the optional Authenticator. If the Authenticator is invalid, then the datagram is discarded, and the "Authentication Failed" error is indicated.

3.2.3. Decryption

If the length of the data to be decrypted is not an integral multiple of the transform block size, then the datagram is discarded, and the "Decryption Failed" error is indicated.

Provide an Initialization Variable (IV) of the form indicated by the cipher transform specification.

Decrypt the ciphertext, producing a plaintext, Padding, Pad Length and Payload Type, in the form indicated by the cipher transform specification.

3.2.4. Payload Type

The Payload Type is removed and examined. If it is unrecognized, then the datagram is discarded, and the "Decryption Failed" error is indicated.

3.2.5. Padding

The Pad Length is removed and examined. If pad checking is configured, and the padding bytes are not the correct values for the Pad Length, then the datagram is discarded, and the "Decryption Failed" error is indicated.

The specified number of padding bytes are removed from the end of the decrypted payload.

3.2.6. Decompression

As indicated by the configured SPI, perform decompression of the plaintext. If it is invalid, then the datagram is discarded, and the "Decompression Failed" error is indicated.

3.2.7. Completion

The IP Total/Payload Length is adjusted to reflect the length of the resulting payload, without the SPI, Sequence Number, Padding, Pad Length, Payload Type, and optional Authenticator.

The IP Header(s) and the remaining portion of the payload are passed to the protocol processing routine specified by the Payload Type field.

3.3. Error Procedures

When an error is indicated by this specification, an ICMP error message of that type is transmitted [RFC-xxxx].

In addition, the implementation SHOULD record the event in a statistics counter, and SHOULD generate an audit log containing date and time, IP Source, IP Destination, IP Flow Label (when present), SPI, Sequence Number, and/or the entire contents of the discarded datagram.

Not all systems that implement ESP will implement auditing. However, if ESP is incorporated into a system that supports auditing, then the ESP implementation MUST also support auditing, and MUST allow a system administrator to enable or disable auditing for ESP.

Additional events not explicitly called out in this specification MAY result in audit log entries.

Operational Considerations

This specification provides only a few manually configurable parameters:

SPI
: Manually configured SPIs are limited in range to aid operations. Automated SPIs are pseudo-randomly distributed throughout the remaining 2**32 values.

  Default: 0 (none). Range: 256 to 65,535.

SPI LifeTime (SPILT)
: Manually configured LifeTimes are generally measured in days. Automated LifeTimes are specified in seconds.

  Default: 32 days (2,764,800 seconds). Maximum: 182 days (15,724,800 seconds).

Replay Window
: Some earlier implementations use pseudo-random values in the present Sequence Number field. This check must only be used with those peers that are known to have implemented this feature.

  Default: 0 (checking off). Range: 32 to 256.

Pad Values
: New implementations use verifiable values. Some earlier

implementations used random values. This check must only be used with those peers that are known to have implemented this feature.

Also, some operations desire additional padding to inhibit traffic analysis. This feature can be combined with verifiable values to provide limited integrity checking.

The value zero (0) indicates that no checking is done, and the range of padding values is defined by the default required for the cipher algorithm.

Default: 0 (checking off). Range: 3 to 255, defined per cipher.

Encryption
Each cipher document will describe the keying material needed.

Default: DES-CBC with derived IV.

Compression
Default: none.

Authentication
Each authenticator document will describe the keying material needed.

Default: none.

Each party configures a list of known SPIs and symmetric secret-keys.

In addition, each party configures local policy that determines what access (if any) is granted to the holder of a particular SPI. For example, a party might allow FTP, but prohibit Telnet. Such considerations are outside the scope of this document.

Security Considerations

This specification is principly concerned with 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 environment.

Users need to understand that the quality of the security provided by this specification depends completely on the strength of whichever cryptographic algorithm that has been implemented, the correctness of that algorithm's implementation, the security of the key management mechanism and its implementation, the strength of the key [CN94], and upon the correctness of the 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. Implementers are encouraged to use high assurance development techniques to develop all of the security relevant parts of their products.

Note that it is possible, when some cryptographic algorithms are employed without an authentication mechanism, for a third party to alter the cleartext of a message, even though that party does not possess the key. It is important that applications requiring both confidentiality and authentication select transforms that prevents this.

The padding bytes have a predictable value. They provide a small measure of tamper detection on their own block and the previous block in CBC mode. This makes it somewhat harder to perform appending attacks in the absence of other integrity protection. Also, this avoids a possible large covert channel (although the amount of padding itself could provide a covert channel), and aids in mechanical certification of the implementation. This small amount of potential known plaintext does not create any problems for modern ciphers.

This mechanism alone does not provide complete immunity from traffic analysis. Users seeking further protection from traffic analysis might consider the use of appropriate link encryption. These details are outside the scope of this specification.

Acknowledgements

This document benefited greatly from earlier work done by Randall Atkinson, Perry Metzger, William Simpson, and Phil Karn, for the SIP, SIPP, and IPv6 Working Groups.

Many of the concepts here are derived from the swIPe security protocol [IBK93a, IBK93b], or were influenced through community diffusion of knowledge regarding the US Government's SP3 security protocol specification [SDN.301], and the ISO/IEC's NLSP specification [ISO-11577].

Steve Bellovin, Steve Deering, Steve Kent, Dave Mihelcic, and Hilarie Orman provided useful critiques of earlier versions of this document.

Contacts

Comments about this document should be discussed on the ipsec@tis.com mailing list.

Questions about this document can also be directed to:

William Allen Simpson
DayDreamer
Computer Systems Consulting Services
1384 Fontaine
Madison Heights, Michigan 48071

wsimpson@UMich.edu
wsimpson@GreenDragon.com (preferred)
bsimpson@MorningStar.com