Network Working Group P Metzger
Internet Draft W A Simpson
expires in six months January 1995

IPv4 Encapsulating Security Payload (4ESP)
draft-metzger-esp-00.txt

Status of this Memo

This document is a submission to the IP Security Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the ipsec@ans.net mailing list.

Distribution of this memo is unlimited.

This document is an Internet-Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months, and may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material, or to cite them other than as a ``working draft'' or ``work in progress.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the internet-drafts Shadow Directories on:

ftp.is.co.za (Africa)
nic.nordu.net (Europe)
ds.internic.net (US East Coast)
ftp.isi.edu (US West Coast)
munnari.oz.au (Pacific Rim)

Abstract

This document describes a privacy mechanism for IPv4, encapsulating transport headers within an opaque envelope.

1. Introduction

The Encapsulating Security Payload (ESP) seeks to provide integrity and confidentiality to IP datagrams. It may also provide authentication, depending on which algorithm and algorithm mode are used.

Users desiring integrity and authentication without confidentiality should use the Authentication Header (AH) instead of ESP.

This document assumes that the reader is familiar with the related document "IPv4 Security Overview" [RAsa], which defines the overall security plan for IPv4, and provides important background for this specification.

1.1. Overview

The Encapsulating Security Payload (ESP) provides confidentiality and integrity by encrypting the data to be protected. Depending on the user's security requirements, only a transport-layer segment (such as UDP or TCP) is encrypted, or the entire IP datagram may be encrypted and tunneled to the destination.

In order for ESP to work properly without changing the entire Internet infrastructure (particularly non-participating routers), the payload is placed within a datagram having unencrypted IP headers. The information in the unencrypted IP headers is used to route the secure datagram.

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 containing an Encapsulating Security Payload. Encapsulating the protected data can be very expensive to implement.

1.2. Key Management

Key management is an important part of the IP security architecture. A scalable standard Internet key management protocol is needed to make widespread use of ESP practical.

However, in order to facilitate early adoption, manual key management is the only key management technique required by this specification.

The only coupling between key management and ESP is the Security Association Identifier (SAID), which is described in more detail later. This 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, key management is specified in a separate specification [TBD].

> Nota Bene: It is intended that the key management mechanisms being developed in other IETF Working Groups will be useful for both IPv4 and IPv6.

1.3. Security Associations

The key management mechanism is used to negotiate a number of parameters for each Security Association between the communicating parties. This includes the key(s) used to encrypt and decrypt the opaque portion of the ESP payload, the sensitivity level (such as Secret or Unclassified) of the user data in the ESP payload, and the particular transform to be used.

The key management implementation usually maintains a table containing the several parameters for each current Security Association. The ESP implementation needs to access that security parameter table to determine how to process each datagram.

The Security Association Identifier (SAID) is assigned by the entity controlling the Destination IP address of the Security Association. The selection mechanism used for the SAID is implementation dependent.

> A given Destination is not necessarily in control of the negotiation process. In the case of multicast groups, a single node or cooperating subset of the multicast group may work on behalf of the entire group to set up a Security Association.

A session between two nodes will normally have two SAID values (one in each direction). The nodes use the combination of SAID and IP Destination to distinguish the correct association.

Senders to a multicast group may share a common Security Association, if all communications are authenticated using the same security configuration parameters. In this case, the receiver only knows that the message came from a node knowing the Security Association for the

group, and cannot authenticate which member of the group sent the datagram when symmetric algorithms are in use.

Multicast groups may also use a separate Security Association value for each Source. If each sender is keyed separately and asymmetric algorithms are used, data origin authentication is also provided.

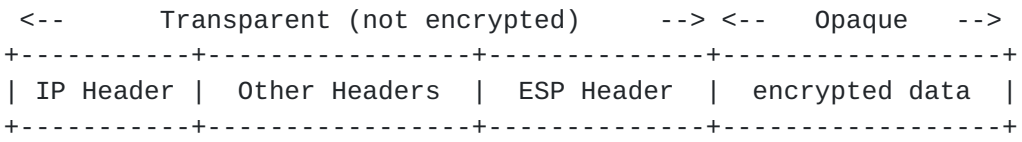
1.4. Transforms

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

Each SAID value indicates the encryption algorithm and mode used, the block size (if any) of the encryption algorithm, the authentication algorithm being used (if separate from the encryption algorithm), the block size (if any) of the authentication algorithm, and the presence/absence and size of a cryptographic synchronization or initialization vector field. These transforms are described in companion documents.

2. Payload Format

The Encapsulating Security Payload (ESP) may appear anywhere after the IP header. The header immediately preceding the ESP header will always contain the value <TBD> in its Next Header (Protocol) field.



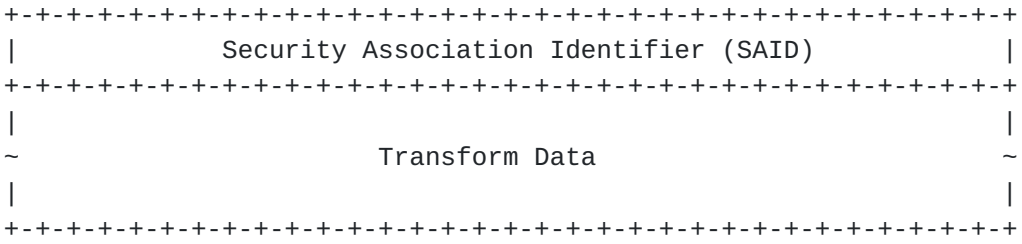
The Encapsulating Security Payload has two components.

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 receiver how to properly decrypt and process the encrypted data.

The opaque ESP component consists of encrypted data. The encrypted data includes protected fields for the ESP transform, and also the encapsulated IP datagram.

2.1. Header Fields

A more detailed diagram of the ESP Header follows:



Security Association Identifier (SAID)

A value identifying the Security Association for this datagram. If no Security Association has been established, the value of this field is zero.

SAID values in the range 0xFFFFFFF1 through 0xFFFFFFFF are reserved for future use.

Transform Data

The length of this field is variable, but is always at least 32-bits.

An implementation will normally use the SAID to determine the field's size and use. It retains the same format for all datagrams of any given SAID and IP Destination.

Refer to each Security Transform specification for more information regarding the contents of this field.

3. Payload Processing

This chapter describes the steps taken when ESP is in use between two communicating parties. There are two modes of use for ESP.

The first mode, which is herein called "IP-mode", encapsulates an entire IP datagram inside ESP.

The second mode, which is herein called "Transport-Mode", encapsulates a transport-layer segment (such as UDP or TCP) inside ESP.

In either case, the sender first determines if a Security Association has been established with the target receiver. If not, then the key management mechanism is used to establish the SAID for this communications session prior to the encryption.

If cleartext datagram If a SAID is received which is not valid for a particular Destination,

then the datagram is discarded, and an appropriate ICMP message is returned. The failure SHOULD be recorded in the system or audit log, including the cleartext values for the SAID, date/time, Source, Destination, and other identifying information.

Multicast is different from unicast only in the area of key management.

3.1. IP-mode

The sender takes the entire original IP datagram, applies the encryption algorithm using the appropriate key for the receiving

party, and encapsulates the result within an ESP header. Next, ESP is sent as the final payload of a cleartext IP datagram.

This mode is used to send encrypted ICMP or IGMP messages. Such messages are often specific to the IP addressing and routing information.

> If strict red/black separation is being enforced, then the addressing and other information in the cleartext IP headers and payloads might be different from the values contained in the (now encrypted and encapsulated) original datagram.

The receiver processes the cleartext IP header and other intervening headers (if any). It then decrypts the ESP using the session key that has been established for this SAID.

The original datagram is extracted from the (now decrypted) ESP. This datagram is then processed as if received normally. In the case of a B1 or Compartmented Mode Workstation, additional mandatory access controls are applied, as appropriate.

3.2. Transport-mode

The sender takes the original transport segment, applies the encryption algorithm using the appropriate key for the receiving party, and encapsulates the result within an ESP header. Next, ESP is sent as the final payload of a cleartext IP datagram.

The receiver processes the cleartext IP header and other invervening IP headers (if any), and temporarily stores pertinent information (such as Source and Destination). It then decrypts the ESP using the session key that has been established for this SAID.

The original transport header is extracted from the (now decrypted) ESP. The information from the cleartext IP header and the extracted transport header is jointly used to determine to which application the data belongs. In the case of a B1 or Compartmented Mode Workstation, additional mandatory access controls are applied, as appropriate.

3.3. Authentication

Some Transforms provide authentication as well as encryption. When such a mechanism is not in use, the Authentication Header [RAah]

might be used.

There are several different approaches, depending on which part of the 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. Next, any intervening IP headers are added before the ESP header. Finally, the 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 Authentication Header process. If authentication succeeds, decryption using the normal ESP process occurs. If decryption is successful, the resulting data is passed to the higher protocol layers.

If the authentication is to be applied only to the data protected by ESP, and the protected data is an entire IP datagram, then the Authentication Header is placed normally within that protected IP datagram.

If the authentication is to be applied to less than an entire IP datagram, then the Authentication Header is placed within the encrypted payload, immediately after the ESP protected header, and before any other header.

An Authentication Header may be present both preceding the ESP header, and also as a header within the encrypted ESP envelope. In such a case, the unencrypted 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 for the encapsulated datagram.

3.4. Other Headers

It is important that all routing information and other such internal headers be included within the encrypted datagram, even if the same information is in the unencrypted part of the datagram.

The receiving system MUST ignore all routing information in the unencrypted portion of the received datagram, and strictly rely on the routing information from the protected payload instead. If this

rule is not strictly adhered to, then the system will be vulnerable to various kinds of attacks, including source routing attacks.

The original datagram may contain an explicit Sensitivity Label, but the encrypted datagram need not include any Sensitivity Label. The SAID indicates the Sensitivity Label for the encrypted datagram.

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.

Users need to understand that the quality of the security provided by this specification depends completely on the strength of whichever encryption 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 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 Encapsulating Security Payload.

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 a transform that prevents this.

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

The original text of this specification was derived from work by Ran Atkinson for the SIP, SIPP, and IPv6 Working Groups.

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

Steve Bellovin, Steve Deering, and Dave Mihelcic provided useful critiques of earlier versions of this draft.

References

[CN94] John M. Carroll & Sri Nudiati, "On Weak Keys and Weak Data: Foiling the Two Nemeses", Cryptologia, Vol. 18 No. 23 pp. 253-280, July 1994.

[IBK93a] John Ioannidis, Matt Blaze, & Phil Karn, "swIPe: The IP Security Protocol", unpublished draft, 14 April 1993.

[IBK93b] John Ioannidis, Matt Blaze, & Phil Karn, "swIPe: Network-Layer Security for IP", Presentation at the Spring 1993 IETF Meeting, Columbus, Ohio.

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

[RAsa] Randall Atkinson, et alia, IPv6 Security Architecture, work in progress.

[RAah] Randall Atkinson, IPv6 Authentication Header, work in progress.

[SDN.301] 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.

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

[15] Dan McDonald, "Security Extensions to the IPv6 Sockets API", work in progress.

Author's Address

Questions about this memo can also be directed to:

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

Telephone: (DSN) 354-8590
Fax: (DSN) 354-7942
<atkinson@itd.nrl.navy.mil>

Perry Metzger
Piermont Information Systems Inc.
160 Cabrini Blvd., Suite #2
New York, NY 10033

perry@piermont.com

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

Bill.Simpson@um.cc.umich.edu
bsimpson@MorningStar.com

Table of Contents