Network Working Group   Stephen Kent, BBN Corp
Internet Draft   Randall Atkinson, @Home Network
draft-ietf-ipsec-esp-v2-05.txt   May 1998

IP Encapsulating Security Payload (ESP)

Status of This Memo

   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 6 months.
   Internet Drafts 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 "work in
   progress".

   This particular Internet Draft is a product of the IETF's IPsec
   working group. 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 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 "Security Architecture for the Internet Protocol" [KA97a],
hereafter referred to as the Security Architecture document).
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 the Security Architecture
document [KA97a].

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

ESP is used to provide confidentiality, data origin authentication,
connectionless integrity, an anti-replay service (a form of partial
sequence integrity), and limited traffic flow confidentiality. The
set of services provided depends on options selected at the time of
Security Association establishment and on the placement of the
implementation. Confidentiality may be selected independent of all
other services. However, use of confidentiality without
integrity/authentication (either in ESP or separately in AH) may
subject traffic to certain forms of active attacks that could
undermine the confidentiality service (see [Bel96]). Data origin
authentication and connectionless integrity are joint services
(hereafter referred to jointly as "authentication) and are offered as
an option in conjunction with (optional) confidentiality. The anti-
replay service may be selected only if data origin authentication is
selected, and its election is solely at the discretion of the
receiver. (Although the default calls for the sender to increment
the Sequence Number used for anti-replay, the service is effective
only if the receiver checks the Sequence Number.) Traffic flow
confidentiality requires selection of tunnel mode, and is most
effective if implemented at a security gateway, where traffic
aggregation may be able to mask true source-destination patterns.
Note that although both confidentiality and authentication are
optional, at least one of them MUST be selected.

It is assumed that the reader is familiar with the terms and concepts
described in the Security Architecture document. In particular, the
reader should be familiar with the definitions of security services
offered by ESP and AH, the concept of Security Associations, the ways

in which ESP can be used in conjunction with the Authentication
Header (AH), and the different key management options available for
ESP and AH. (With regard to the last topic, the current key
management options required for both AH and ESP are manual keying and
automated keying via IKE [HC98].)

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD,
SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this
document, are to be interpreted as described in RFC 2119 [Bra97].

2. Encapsulating Security Payload Packet Format

The protocol header (IPv4, IPv6, or Extension) immediately preceding the
ESP header will contain the value 50 in its Protocol (IPv4) or Next
Header (IPv6, Extension) field [STD-2].

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ ----
|               Security Parameters Index (SPI)                 | ^
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |Auth.
|                      Sequence Number                          | |Coverage
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ | -----
|                    Payload Data* (variable)                   | |   ^
~                                                               ~ |   |
|                                                               | |   |
+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |Confid.
|               |     Padding (0-255 bytes)                     | |Coverage*
+-+-+-+-+-+-+-+-+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |   |
|                               |  Pad Length   | Next Header   | v   v
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ -------
|                 Authentication Data (variable)                |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

      * If included in the Payload field, cryptographic synchronization
        data, e.g., an Initialization Vector (IV, see Section 2.3),
        usually is not encrypted per se, although it often is referred to
        as being part of the ciphertext.

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 Integrity Check Value (ICV,
see Section 2.7). 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 that, in combination with the
destination IP address and security protocol (ESP), uniquely
identifies the Security Association for this datagram. 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. It is ordinarily selected
by the destination system upon establishment of an SA (see the
Security Architecture document for more details). The SPI field is
mandatory.

The SPI value of zero (0) is reserved for local, implementation-
specific use and MUST NOT be sent on the wire. For example, a key
management implementation MAY use the zero SPI value to mean "No
Security Association Exists" during the period when the IPsec
implementation has requested that its key management entity establish
a new SA, but the SA has not yet been established.

2.2 Sequence Number

This unsigned 32-bit field contains a monotonically increasing
counter value (sequence number). It is mandatory and is always
present even if the receiver does not elect to enable the anti-replay
service for a specific SA. Processing of the Sequence Number field
is at the discretion of the receiver, i.e., the sender MUST always
transmit this field, but the receiver need not act upon it (see the
discussion of Sequence Number Verification in the "Inbound Packet
Processing" section below).

The sender's counter and the receiver's counter are initialized to 0
when an SA is established. (The first packet sent using a given SA
will have a Sequence Number of 1; see Section 3.3.3 for more details
on how the Sequence Number is generated.) If anti-replay is enabled
(the default), the transmitted Sequence Number must never be allowed
to cycle. Thus, the sender's counter and the receiver's counter MUST
be reset (by establishing a new SA and thus a new key) prior to the
transmission of the 2^32nd packet on an SA.

2.3 **Payload Data**

   Payload Data is a variable-length field containing data described by
   the Next Header field. The Payload Data field is mandatory and is an
   integral number of bytes in length. If the algorithm used to encrypt
   the payload requires cryptographic synchronization data, e.g., an
   Initialization Vector (IV), then this data MAY be carried explicitly
   in the Payload field. Any encryption algorithm that requires such
   explicit, per-packet synchronization data MUST indicate the length,
   any structure for such data, and the location of this data as part of
   an RFC specifying how the algorithm is used with ESP. If such
   synchronization data is implicit, the algorithm for deriving the data
   MUST be part of the RFC.

   Note that with regard to ensuring the alignment of the (real)
   ciphertext in the presence of an IV:
            o For some IV-based modes of operation, the receiver treats the
              IV as the start of the ciphertext, feeding it into the
              algorithm directly. In these modes, alignment of the start of
              the (real) ciphertext is not an issue at the receiver.
            o In some cases, the receiver reads the IV in separately from
              the ciphertext. In these cases, the algorithm specification
              MUST address how alignment of the (real) ciphertext is to be
              achieved.

2.4 **Padding (for Encryption)**

   Several factors require or motivate use of the Padding field.

            o If an encryption algorithm is employed that requires the
              plaintext to be a multiple of some number of bytes, e.g., the
              block size of a block cipher, the Padding field is used to
              fill the plaintext (consisting of the Payload Data, Pad Length
              and Next Header fields, as well as the Padding) to the size
              required by the algorithm.

            o Padding also may be required, irrespective of encryption
              algorithm requirements, to ensure that the resulting
              ciphertext terminates on a 4-byte boundary. Specifically, the
              Pad Length and Next Header fields must be right aligned within
              a 4-byte word, as illustrated in the ESP packet format figure
              above, to ensure that the Authentication Data field (if
              present) is aligned on a 4-byte boundary.

            o Padding beyond that required for the algorithm or alignment
              reasons cited above, may be used to conceal the actual length
              of the payload, in support of (partial) traffic flow

confidentiality. However, inclusion of such additional
padding has adverse bandwidth implications and thus its use
should be undertaken with care.

The sender MAY add 0-255 bytes of padding. Inclusion of the Padding
field in an ESP packet is optional, but all implementations MUST
support generation and consumption of padding.

>   a. For the purpose of ensuring that the bits to be encrypted are
>   a multiple of the algorithm's blocksize (first bullet above),
>   the padding computation applies to the Payload Data exclusive
>   of the IV, the Pad Length, and Next Header fields.
>
>   b. For the purposes of ensuring that the Authentication Data is
>   aligned on a 4-byte boundary (second bullet above), the
>   padding computation applies to the Payload Data inclusive of
>   the IV, the Pad Length, and Next Header fields.

If Padding bytes are needed but the encryption algorithm does not
specify the padding contents, then the following default processing
MUST be used. The Padding bytes are initialized with a series of
(unsigned, 1-byte) integer values. The first padding byte appended
to the plaintext is numbered 1, with subsequent padding bytes making
up a monotonically increasing sequence: 1, 2, 3, ... When this
padding scheme is employed, the receiver SHOULD inspect the Padding
field. (This scheme was selected because of its relative simplicity,
ease of implementation in hardware, and because it offers limited
protection against certain forms of "cut and paste" attacks in the
absence of other integrity measures, if the receiver checks the
padding values upon decryption.)

Any encryption algorithm that requires Padding other than the default
described above, MUST define the Padding contents (e.g., zeros or
random data) and any required receiver processing of these Padding
bytes in an RFC specifying how the algorithm is used with ESP. In
such circumstances, the content of the Padding field will be
determined by the encryption algorithm and mode selected and defined
in the corresponding algorithm RFC. The relevant algorithm RFC MAY
specify that a receiver MUST inspect the Padding field or that a
receiver MUST inform senders of how the receiver will handle the
Padding field.

2.5 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 no Padding bytes are present. The Pad Length
   field is mandatory.

2.6 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.

2.7 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 is specified by the
   authentication function selected. The Authentication Data field is
   optional, and is included only if the authentication service has been
   selected for the SA in question. The authentication algorithm
   specification MUST specify the length of the ICV and the comparison
   rules and processing steps for validation.
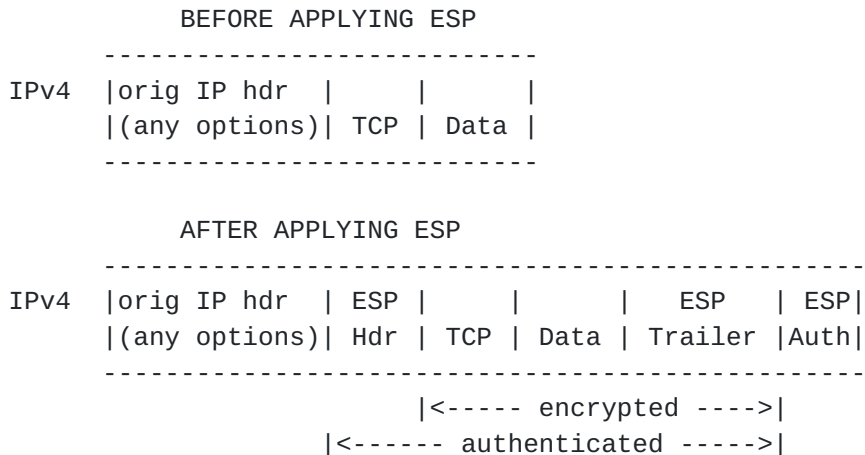
3. Encapsulating Security Protocol Processing

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, note that for "bump-in-the-stack" or "bump-in-the-
   wire" implementations, as defined in the Security Architecture
   document, inbound and outbound IP fragments may require an IPsec
   implementation to perform extra IP reassembly/fragmentation in order
   to both conform to this specification and provide transparent IPsec
   support. Special care is required to perform such operations within
   these implementations when multiple interfaces are in use.)

   In transport mode, ESP is inserted after the IP header and before an
   upper layer protocol, e.g., TCP, UDP, ICMP, etc. or before any other
   IPsec headers that have already been inserted. 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.)

```
                    BEFORE APPLYING ESP
              ----------------------------
      IPv4  |orig IP hdr  |     |      |
            |(any options)| TCP | Data |
              ----------------------------

                    AFTER APPLYING ESP
              ---------------------------------------------------
      IPv4  |orig IP hdr  | ESP |     |      |    ESP   | ESP|
            |(any options)| Hdr | TCP | Data | Trailer |Auth|
              ---------------------------------------------------
                                  |<----- encrypted ---->|
                            |<------ authenticated ----->|
```

   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
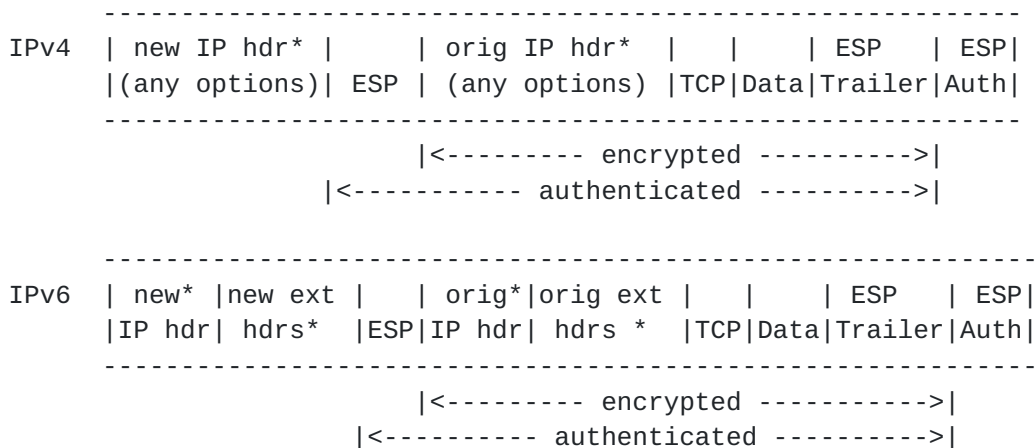   illustrates ESP transport mode positioning for a typical IPv6 packet.
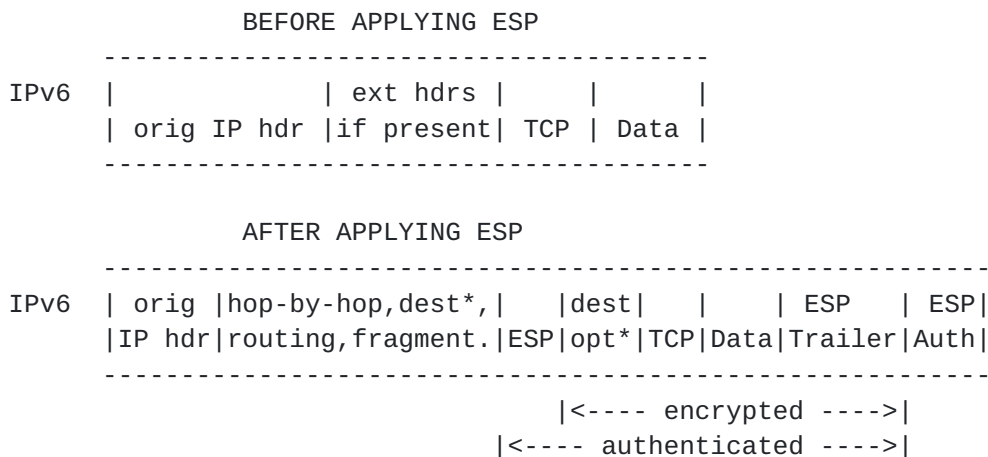
```
                    BEFORE APPLYING ESP
              -----------------------------------------
      IPv6  |               | ext hdrs |     |      |
            | orig IP hdr |if present| TCP | Data |
              -----------------------------------------

                    AFTER APPLYING ESP
              -----------------------------------------------------------
      IPv6  | orig |hop-by-hop,dest*,|   |dest|   |    | ESP   | ESP|
            |IP hdr|routing,fragment.|ESP|opt*|TCP|Data|Trailer|Auth|
              -----------------------------------------------------------
                                        |<---- encrypted ---->|
                                   |<---- authenticated ---->|

              * = if present, could be before ESP, after ESP, or both
```

   ESP and AH headers can be combined in a variety of modes. The IPsec
   Architecture document describes the combinations of security
   associations that must be supported.

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.

```
              ----------------------------------------------------------
     IPv4   | new IP hdr* |     | orig IP hdr* |   |    | ESP   | ESP|
            |(any options)| ESP | (any options) |TCP|Data|Trailer|Auth|
              ----------------------------------------------------------
                              |<--------- encrypted ---------->|
                        |<----------- authenticated ---------->|


              -----------------------------------------------------------
     IPv6   | new* |new ext |    | orig*|orig ext |   |    | ESP   | ESP|
            |IP hdr| hdrs*  |ESP|IP hdr| hdrs *  |TCP|Data|Trailer|Auth|
              -----------------------------------------------------------
                              |<--------- encrypted ----------->|
                        |<--------- authenticated --------->|
```

        * = if present, construction of outer IP hdr/extensions and
            modification of inner IP hdr/extensions is discussed
            below.

3.2 Algorithms

The mandatory-to-implement algorithms are described in Section 5,
"Conformance Requirements". Other algorithms MAY be supported. Note
that although both confidentiality and authentication are optional,
at least one of these services MUST be selected hence both algorithms
MUST NOT be simultaneously NULL.

3.2.1 Encryption Algorithms

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 any data
required to allow the receiver to establish cryptographic
synchronization for decryption. This data may be carried explicitly
in the payload field, e.g., as an IV (as described above), or the
data may be derived from the packet header. Since ESP makes

   provision for padding of the plaintext, encryption algorithms
   employed with ESP may exhibit either block or stream mode
   characteristics. Note that since encryption (confidentiality) is
   optional, this algorithm may be "NULL".

3.2.2 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 appropriate, though performance and space
   considerations currently preclude use of such algorithms. Note that
   since authentication is optional, this algorithm may be "NULL".

3.3 Outbound Packet Processing

   In transport mode, the sender 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 Security Architecture document. If there is more than one IPsec
   header/extension required by security policy, the order of the
   application of the security headers MUST be defined by security
   policy.

3.3.1 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 Security Architecture document.

3.3.2 Packet Encryption

   In this section, we speak in terms of encryption always being applied
   because of the formatting implications. This is done with the
   understanding that "no confidentiality" is offered by using the NULL
   encryption algorithm. Accordingly, the sender:
           1. encapsulates (into the ESP Payload field):
                   - for transport mode -- just the original upper layer
                     protocol information.

                - for tunnel mode -- the entire original IP datagram.
      2. adds any necessary padding.
      3. encrypts the result (Payload Data, Padding, Pad Length, and
         Next Header) using the key, encryption algorithm, algorithm
         mode indicated by the SA and cryptographic synchronization
         data (if any).
                - If explicit cryptographic synchronization data, e.g.,
                  an IV, is indicated, it is input to the encryption
                  algorithm per the algorithm specification and placed
                  in the Payload field.
                - If implicit cryptographic synchronication data, e.g.,
                  an IV, is indicated, it is constructed and input to
                  the encryption algorithm as per the algorithm
                  specification.

The exact steps for constructing the outer IP header depend on the
mode (transport or tunnel) and are described in the Security
Architecture document.

If authentication is 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.

3.3.3 Sequence Number Generation

The sender's counter is initialized to 0 when an SA is established.
The sender increments the Sequence Number for this SA and inserts the
new value into the Sequence Number field. Thus the first packet sent
using a given SA will have a Sequence Number of 1.

If anti-replay is enabled (the default), the sender checks to ensure
that the counter has not cycled before inserting the new value in the
Sequence Number field. In other words, the sender MUST NOT send a
packet on an SA if doing so would cause the Sequence Number to cycle.
An attempt to transmit a packet that would result in Sequence Number
overflow is an auditable event. (Note that this approach to Sequence
Number management does not require use of modular arithmetic.)

The sender assumes anti-replay is enabled as a default, unless
otherwise notified by the receiver (see 3.4.3). Thus, if the counter

   has cycled, the sender will set up a new SA and key (unless the SA
   was configured with manual key management).

   If anti-replay has been disabled, the sender does not need to monitor
   or reset the counter, e.g., in the case of manual key management (see
   Section 5).

3.3.4 Integrity Check Value Calculation

   If authentication is selected for the SA, the sender computes the ICV
   over the ESP packet minus the Authentication Data. Thus the SPI,
   Sequence Number, Payload Data, Padding (if present), Pad Length, and
   Next Header are all encompassed by the ICV computation. Note that
   the last 4 fields will be in ciphertext form, since encryption is
   performed prior to authentication.

   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, (after the
   Next Header field) 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. Note that MD5 and SHA-1 are
   viewed as having a 1-byte blocksize because of their internal padding
   conventions.

3.3.5 Fragmentation

   If necessary, fragmentation is performed after ESP processing within
   an IPsec implementation. Thus, transport mode ESP is applied only to
   whole IP datagrams (not to IP fragments). An IP packet to which ESP
   has been applied may itself be fragmented by routers en route, and
   such fragments must be reassembled prior to ESP processing at a
   receiver. In tunnel mode, ESP is applied to an IP packet, the
   payload of which may be a fragmented IP packet. For example, a
   security gateway or a "bump-in-the-stack" or "bump-in-the-wire" IPsec
   implementation (as defined in the Security Architecture document) may
   apply tunnel mode ESP to such fragments.

   NOTE: For transport mode -- As mentioned at the beginning of Section
   3.1, bump-in-the-stack and bump-in-the-wire implementations may have
   to first reassemble a packet fragmented by the local IP layer, then
   apply IPsec, and then fragment the resulting packet.

   NOTE: For IPv6 -- For bump-in-the-stack and bump-in-the-wire
   implementations, it will be necessary to walk through all the
   extension headers to determine if there is a fragmentation header and
   hence that the packet needs reassembling prior to IPsec processing.

3.4 Inbound Packet Processing

3.4.1 Reassembly

   If required, reassembly is performed prior to ESP processing. If a
   packet offered to ESP for processing appears to be an IP fragment,
   i.e., the OFFSET field is non-zero or the MORE FRAGMENTS flag is set,
   the receiver MUST discard the packet; this is an auditable event. The
   audit log entry for this event SHOULD include the SPI value,
   date/time received, Source Address, Destination Address, Sequence
   Number, and (in IPv6) the Flow ID.

   NOTE: For packet reassembly, the current IPv4 spec does NOT require
   either the zero'ing of the OFFSET field or the clearing of the MORE
   FRAGMENTS flag. In order for a reassembled packet to be processed by
   IPsec (as opposed to discarded as an apparent fragment), the IP code
   must do these two things after it reassembles a packet.

3.4.2 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, security protocol (ESP), and the SPI. (This
   process is described in more detail in the Security Architecture
   document.) The SA indicates whether the Sequence Number field will
   be checked, whether the Authentication Data field 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; this is an auditable event. The audit log entry for this
   event SHOULD include the SPI value, date/time received, Source
   Address, Destination Address, Sequence Number, and (in IPv6) the
   cleartext Flow ID.

3.4.3 Sequence Number Verification

   All ESP implementations MUST support the anti-replay service, though
   its use may be enabled or disabled by the receiver on a per-SA basis.

This service MUST NOT be enabled unless the authentication service
also is enabled for the SA, since otherwise the Sequence Number field
has not been integrity protected. (Note that there are no provisions
for managing transmitted Sequence Number values among multiple
senders directing traffic to a single SA (irrespective of whether the
destination address is unicast, broadcast, or multicast). Thus the
anti-replay service SHOULD NOT be used in a multi-sender environment
that employs a single SA.)

If the receiver does not enable anti-replay for an SA, no inbound
checks are performed on the Sequence Number. However, from the
perspective of the sender, the default is to assume that anti-replay
is enabled at the receiver. To avoid having the sender do
unnecessary sequence number monitoring and SA setup (see section
3.3.3), if an SA establishment protocol such as IKE is employed, the
receiver SHOULD notify the sender, during SA establishment, if the
receiver will not provide anti-replay protection.

If the receiver has enabled the anti-replay service for this SA, the
receive packet counter for the SA MUST be initialized to zero when
the SA is established. For each received packet, the receiver MUST
verify that the packet contains a Sequence Number that does not
duplicate the Sequence Number of any other packets received during
the life of this SA. This SHOULD be the first ESP 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.) A MINIMUM window size of 32 MUST be supported; but a
window size of 64 is preferred and SHOULD be employed as the default.
Another window size (larger than the MINIMUM) MAY be chosen by the
receiver. (The receiver does NOT notify the sender of the window
size.)

The "right" edge of the window represents the highest, validated
Sequence Number 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 the Security Architecture document.

If the received packet falls within the window and is new, or if the
packet is to the right of the window, then the receiver proceeds to
ICV verification. If the ICV validation fails, the receiver MUST

   discard the received IP datagram as invalid; this is an auditable
   event. The audit log entry for this event SHOULD include the SPI
   value, date/time received, Source Address, Destination Address, the
   Sequence Number, and (in IPv6) the Flow ID. The receive window is
   updated only if the ICV verification succeeds.

   DISCUSSION:

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

3.4.4 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; this is an auditable
   event. The log data SHOULD include the SPI value, date/time
   received, Source Address, Destination Address, the Sequence Number,
   and (in IPv6) the cleartext Flow ID.

   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 saved value, using the comparison rules defined by the
      algorithm specification. (For example, if a digital signature and
      one-way hash are used for the ICV computation, the matching
      process is more complex.)

3.4.5 Packet Decryption

   As in section 3.3.2, "Packet Encryption", we speak here in terms of
   encryption always being applied because of the formatting
   implications. This is done with the understanding that "no

   confidentiality" is offered by using the NULL encryption algorithm.
   Accordingly, the receiver:

         1. decrypts the ESP Payload Data, Padding, Pad Length, and Next
            Header using the key, encryption algorithm, algorithm mode,
            and cryptographic synchronization data (if any), indicated by
            the SA.
                  - If explicit cryptographic synchronization data, e.g.,
                    an IV, is indicated, it is taken from the Payload
                    field and input to the decryption algorithm as per the
                    algorithm specification.
                  - If implicit cryptographic synchronization data, e.g.,
                    an IV, is indicated, a local version of the IV is
                    constructed and input to the decryption algorithm as
                    per the algorithm specification.
         2. processes any padding as specified in the encryption
            algorithm specification. If the default padding scheme (see
            [Section 2.4](#)) has been employed, the receiver SHOULD inspect
            the Padding field before removing the padding prior to
            passing the decrypted data to the next layer.
         3. reconstructs the original IP datagram from:
                  - for transport mode -- original IP header plus the
                    original upper layer protocol information in the ESP
                    Payload field
                  - for tunnel mode -- tunnel IP header + the entire IP
                    datagram in the ESP Payload field.

   The exact steps for reconstructing the original datagram depend on
   the mode (transport or tunnel) and are described in the Security
   Architecture document. At a minimum, in an IPv6 context, the
   receiver SHOULD ensure that the decrypted data is 8-byte aligned, to
   facilitate processing by the protocol identified in the Next Header
   field.

   If authentication has been selected, verification and decryption MAY
   be performed serially or in parallel. If performed serially, then
   ICV verification SHOULD be performed first. If performed in
   parallel, verification MUST be completed before the decrypted packet
   is passed on for further processing. 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. Note:
   If the receiver performs decryption in parallel with authentication,
   care must be taken to avoid possible race conditions with regard to
   packet access and reconstruction of the decrypted packet.

   Note that there are several ways in which the decryption can "fail":

a. The selected SA may not be correct -- The SA may be
   mis-selected due to tampering with the SPI, destination
   address, or IPsec protocol type fields. Such errors, if they
   map the packet to another extant SA, will be
   indistinguishable from a corrupted packet, (case c).
   Tampering with the SPI can be detected by use of
   authentication. However, an SA mismatch might still occur
   due to tampering with the IP Destination Address or the IPsec
   protocol type field.

b. The pad length or pad values could be erroneous -- Bad pad
   lengths or pad values can be detected irrespective of the use
   of authentication.

c. The encrypted ESP packet could be corrupted -- This can be
   detected if authentication is selected for the SA.,

In case (a) or (c), 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]. Auditing

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. For the most
part, the granularity of auditing is a local matter. However,
several auditable events are identified in this specification and for
each of these events a minimum set of information that SHOULD be
included in an audit log is defined. Additional information also MAY
be included in the audit log for each of these events, and additional
events, not explicitly called out in this specification, also MAY
result in audit log entries. There is no requirement for the
receiver to transmit any message to the purported sender in response
to the detection of an auditable event, because of the potential to
induce denial of service via such action.

[5]. 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 document. If the key used to compute an ICV is manually

distributed, correct provision of the anti-replay service would
require correct maintenance of the counter state at the sender, until
the key is replaced, and there likely would be no automated recovery
provision if counter overflow were imminent. Thus a compliant
implementation SHOULD NOT provide this service in conjunction with
SAs that are manually keyed. A compliant ESP implementation MUST
support the following mandatory-to-implement algorithms:

            - DES in CBC mode [MD97]
            - HMAC with MD5 [MG97a]
            - HMAC with SHA-1 [MG97b]
            - NULL Authentication algorithm
            - NULL Encryption algorithm

Since ESP encryption and authentication are optional, support for the
2 "NULL" algorithms is required to maintain consistency with the way
these services are negotiated. NOTE that while authentication and
encryption can each be "NULL", they MUST NOT both be "NULL".

6. Security Considerations

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

7. Differences from RFC 1827

This document differs from RFC 1827 [ATK95] in several significant
ways. The major difference is that, this document attempts to
specify a complete framework and context for ESP, whereas RFC 1827
provided a "shell" that was completed through the definition of
transforms. The combinatorial growth of transforms motivated the
reformulation of the ESP specification as a more complete document,
with options for security services that may be offered in the context
of ESP. Thus, fields previously defined in transform documents are
now part of this base ESP specification. For example, the fields
necessary to support authentication (and anti-replay) are now defined
here, even though the provision of this service is an option. The
fields used to support padding for encryption, and for next protocol
identification, are now defined here as well. Packet processing
consistent with the definition of these fields also is included in
the document.

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 3 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 Karen Seo for providing
   extensive help in the review, editing, background research, and
   coordination for this version of the specification. The authors
   would also 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, Norman Shulman, William Simpson and Nina
   Yuan.

References

   [ATK95]   R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC
             1827, August 1997.

   [Bel96]   Steven M. Bellovin, "Problem Areas for the IP Security
             Protocols", Proceedings of the Sixth Usenix Unix Security
             Symposium, July, 1996.

   [Bra97]   S. Bradner, "Key words for use in RFCs to Indicate
             Requirement Level," RFC-2119, March 1997.

   [HC98]   D. Harkins & D. Carrel, "The Internet Key Exchange (IKE)",
             Internet Draft, February 1998.

   [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.

   [KA97a]   Steve Kent, Randall Atkinson, "Security Architecture for
             the Internet Protocol", Internet Draft, March 1998.

   [KA97b]   Steve Kent, Randall Atkinson, "IP Authentication Header",
             Internet Draft, March 1998.

   [MD97]   C. Madson & N. Doraswamy, "The ESP DES-CBC Cipher Algorithm
             With Explicit IV", Internet Draft, February 1998.

   [MG97a]   C. Madson & R. Glenn, "The Use of HMAC-MD5-96 within ESP
             and AH", Internet Draft, February 1998.

   [MG97b]   C. Madson & R. Glenn, "The Use of HMAC-SHA-1-96 within ESP
             and AH", Internet Draft, February 1998.

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

   [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
   E-mail: kent@bbn.com
   Telephone: +1 (617) 873-3988

   Randall Atkinson
   @Home Network
   425 Broadway,
   Redwood City, CA 94063
   USA
   E-mail: rja@corp.home.net
   Telephone: +1 (415) 569-5000