Light-Weight Implementation Guidance (lwig)   D. Migault, Ed.
Internet-Draft   Ericsson
Intended status: Informational   T. Guggemos
Expires: September 14, 2017   LMU Munich
                                                         March 13, 2017

                              Minimal ESP
                    draft-mglt-lwig-minimal-esp-04.txt

Abstract

   This document describes a minimal version of the IP Encapsulation
   Security Payload (ESP) described in RFC 4303 which is part of the
   IPsec suite.

   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.

   This document does not update or modify RFC 4303, but provides a
   compact description of how to implement the minimal version of the
   protocol. If this document and RFC 4303 conflicts then RFC 4303 is
   the authoritative description.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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 inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on September 14, 2017.

Copyright Notice

Table of Contents

1. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

2. Introduction

ESP [RFC4303] is part of the IPsec suite protocol [RFC4301] . It 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.

Figure 1 describes an ESP Packet. Currently ESP is implemented in
the kernel of major multi purpose Operating Systems (OS). The ESP
and IPsec stack implemented is usually complete to fit multiple
purpose usage of these OS. Completeness of the IPsec stack as well
as multi purpose of these OS is often performed at the expense of
resources, or a lack of performance, and so devices especially

constraint devices like sensors have developed their own specific and
task specific OS. This document provides a minimal ESP
implementation guideline so these devices can implement ESP and
benefit from IPsec.

For each field of the ESP packet represented in Figure 1 this
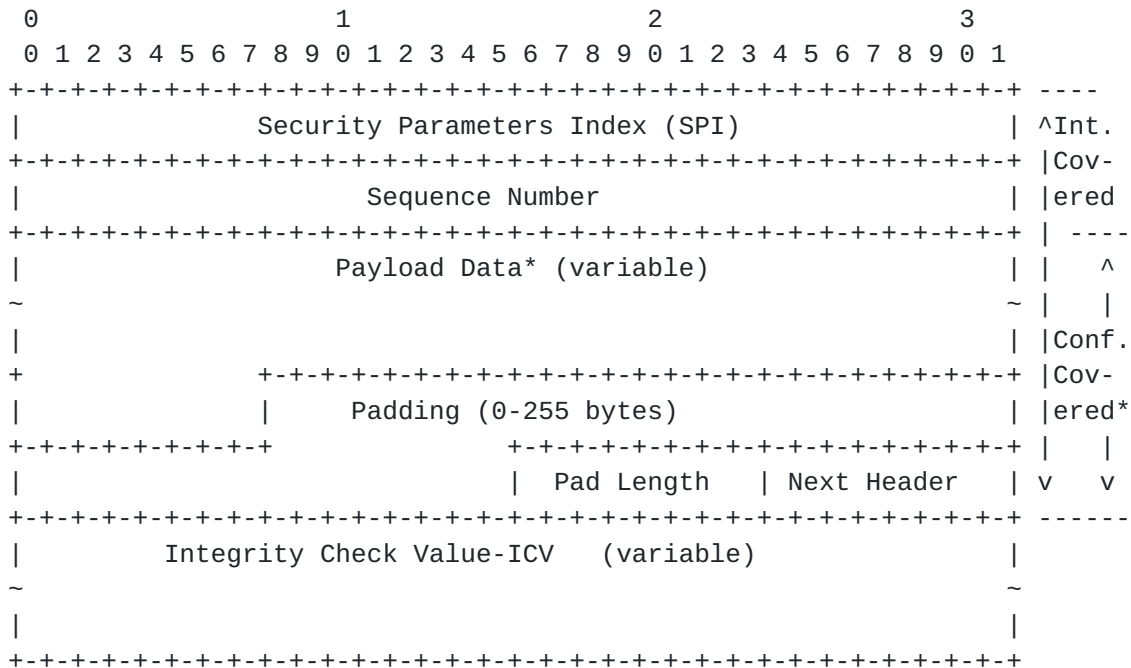document provides recommendations and guidance for minimal
implementations.

```
 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)                 | ^Int.
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |Cov-
|                      Sequence Number                          | |ered
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ | ----
|                    Payload Data* (variable)                   | |   ^
~                                                               ~ |   |
|                                                               | |Conf.
+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |Cov-
|               |     Padding (0-255 bytes)                      | |ered*
+-+-+-+-+-+-+-+-+               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |   |
|                               | Pad Length   | Next Header    | v   v
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ ------
|         Integrity Check Value-ICV   (variable)               |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: ESP Packet Description

3. Security Parameter Index (SPI) (32 bit)

According to the [RFC4303], the SPI is a mandatory 32 bits field and
is not allowed to be removed.

The SPI has a local significance to index the Security Association.
From [RFC4301] section 4.1, nodes supporting only unicast
communications can index their SA only using the SPI. On the other
hand, nodes supporting multicast communications must also use the IP
addresses and thus SA lookup needs to be performed using the longest
match.

When a node is receiving a lot of inbound session, it is RECOMMENDED
to randomly generate the SPI to index each inbound session.
Typically the random generation of the SPI reduces the probability of
SPI assigned to index inbound session associated to two different
remote nodes. When a collision occurs, SPI is expected to be

generated as long as the resulting SPI is not assigned. When SPI are
uniquely assigned, it can clearly index the SA.

However, for some constraint nodes, generating a random SPI may
consume to much resource, in which case SPI can be generated using
predictable functions or even a fix value. In fact, the SPI does not
need to the SPI does not need to be random.

When a constraint node uses a fix value as a SPI, it is RECOMMENDED
the constraint node has as many SPI values as ESP session per host,
and that lookup includes the IP addresses.

Note that SPI value is used only for inbound traffic, as such the SPI
negotiated with IKEv2 [RFC7296] or [RFC7815] by a peer, is the value
used by the remote peer when its sends traffic.

The use of a limited number of fix SPI also come with security or
privacy drawbacks. Typically, a passive attacker may derive
information such as the number of constraint devices connecting the
remote peer, and in conjunction with data rate, the attacker may
eventually determine the application the constraint device is
associated to. In addition, if the fix value SPI is fixed by a
manufacturer or by some software application, the SPI may leak in an
obvious way the type of sensor, the application involved or the model
of the constraint device. As a result, the use of a unpredictable
SPI is preferred to provide better privacy.

As far as security is concerned, revealing the type of application or
model of the constraint device could be used to identify the
vulnerabilities the constraint device is subject to. This is
especially sensitive for constraint device where patches or software
updates will be challenging to operate. As a result, these devices
may remain vulnerable for relatively long period. In addition,
predictable SPI enable an attacker to forge packets with a valid SPI.
Such packet will not be rejected due to an SPI mismatch, but instead
after the signature check which requires more resource and thus make
DoS more efficient, especially for devices powered by batteries.

Values 0-255 SHOULD NOT be used. Values 1-255 are reserved and 0 is
only allowed to be used internal and it MUST NOT be send on the wire.

[RFC4303] mentions :

  -   "The SPI is an arbitrary 32-bit value that is used by a receiver
      to identify the SA to which an incoming packet is bound. The SPI
      field is mandatory. [...]"

-    "For a unicast SA, the SPI can be used by itself to specify an
     SA, or it may be used in conjunction with the IPsec protocol type
     (in this case ESP). Because the SPI value is generated by the
     receiver for a unicast SA, whether the value is sufficient to
     identify an SA by itself or whether it must be used in conjunction
     with the IPsec protocol value is a local matter. This mechanism
     for mapping inbound traffic to unicast SAs MUST be supported by
     all ESP implementations."

4. Sequence Number(SN) (32 bit)

   According to [RFC4303], the sequence number is a mandatory 32 bits
   field in the packet.

   The SN is set by the sender so the receiver can implement anti-replay
   protection. The SN is derived from any strictly increasing function
   that guarantees: if packet B is sent after packet A, then SN of
   packet B is strictly greater then the SN of packet A.

   In IoT, constraint devices are expected to establish communication
   with specific devices, like a specific gateway, or nodes similar to
   them. As a result, the sender may know whereas the receiver
   implements anti-replay protection or not. Even though the sender may
   know the receiver does not implement anti replay protection, the
   sender MUST implement a always increasing function to generate the
   SN.

   Usually, SN is generated by incrementing a counter for each packet
   sent. A constraint device may avoid maintaining this context. If
   the device has a clock, it may use the time indicated by the clock
   has a SN. This guarantees a strictly increasing function, and avoid
   storing any additional values or context related to the SN. When the
   use of a clock is considered, one should take care that packets
   associated to a given SA are not sent with the same time value.

   [RFC4303] mentions :

   -    "This unsigned 32-bit field contains a counter value that
        increases by one for each packet sent, i.e., a per-SA packet
        sequence number. For a unicast SA or a single-sender multicast
        SA, the sender MUST increment this field for every transmitted
        packet. Sharing an SA among multiple senders is permitted, though
        generally not recommended. [...] The field is mandatory and MUST
        always be present even if the receiver does not elect to enable
        the anti-replay service for a specific SA."

5. Padding

   The purpose of padding is to respect the 32 byte alignment of ESP.
   Although the 32 byte alignment is mandatory, it may be performed
   either by the encryption algorithm or by ESP. AES in CBC mode
   [RFC3602] performs padding, while AES in CTR [RFC3686], GCM [RFC4106]
   or CCM [RFC4309] mode do not consider Padding. As a result, when
   such algorithms are used, Padding must be done by ESP. ESP defines
   that padding bytes MUST be generated by a succession of unsigned
   bytes starting with 1, 2, 3 with the last byte set to Pad Length,
   where Pad Length designates the length of the padding bytes.
   Checking the padding structure is not mandatory, so the constraint
   device may not proceed to such checks, however, in order to
   interoperate with existing ESP implementations, it MUST build the
   padding bytes as recommended by ESP.

   [RFC4303] mentions :

   -    "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.)"

6. Next Header (8 bit)

   According to [RFC4303], the Next Header is a mandatory 8 bits field
   in the packet. In some cases, devices are dedicated to a single
   application or a single transport protocol, in which case, the Next
   Header has a fix value.

   [RFC4303] mentions :

   -    "The Next Header is a mandatory, 8-bit field that identifies the
        type of data contained in the Payload Data field, e.g., an IPv4 or
        IPv6 packet, or a next layer header and data. [...] the protocol
        value 59 (which means "no next header") MUST be used to designate
        a "dummy" packet. A transmitter MUST be capable of generating
        dummy packets marked with this value in the next protocol field,
        and a receiver MUST be prepared to discard such packets, without
        indicating an error."

7. ICV

   The ICV is an optional value with variable length. Unless the
   crypto-suite provides authentication without the use of the ICV
   field, the ICV field is used to host the authentication part of the
   packet.

   As detailed in Section 8 we recommend to use authentication, the ICV
   field is expected to be present that is to say with a size different
   from zero. This makes it a mandatory field which size is defined by
   the security recommendations only.

   [RFC4303] mentions :

   -    "The Integrity Check Value is a variable-length field computed
      over the ESP header, Payload, and ESP trailer fields. Implicit
      ESP trailer fields (integrity padding and high-order ESN bits, if
      applicable) are included in the ICV computation. The ICV field is
      optional. It is present only if the integrity service is selected
      and is provided by either a separate integrity algorithm or a
      combined mode algorithm that uses an ICV. The length of the field
      is specified by the integrity algorithm selected and associated
      with the SA. The integrity algorithm specification MUST specify
      the length of the ICV and the comparison rules and processing
      steps for validation."

8. Cryptographic Suites

   The cryptographic suites implemented are an important component of
   ESP. The recommended suites to use are expect to evolve over time
   and implementer SHOULD follow the recommendations provided by
   [I-D.ietf-ipsecme-rfc7321bis] and updates. Recommendations are
   provided for standard nodes as well as constraint nodes.

   This section lists some of the criteria that may be considered. The
   list is not expected to be exhaustive and may also evolve overtime.
   As a result, the list is provided as indicative:

   - Security :   Security is the criteria that should be considered
         first when a selection of cipher suites is performed. The
         security of cipher suites is expected to evolve over time, and
         it is of primary importance to follow up-to-date security
         guidances and recommendations. The chosen cipher suites MUST
         NOT be known vulnerable or weak (see
         [I-D.ietf-ipsecme-rfc7321bis] for outdated ciphers). ESP can
         be used to authenticate only or to encrypt the communication.
         In the later case, authenticated encryption must always be
         considered [I-D.ietf-ipsecme-rfc7321bis].

   - Interoperability :   Interoperability considers the cipher suites
        shared with the other nodes. Note that it is not because a
        cipher suite is widely deployed that is secured. As a result,
        security SHOULD NOT be weaken for interoperability.
        [I-D.ietf-ipsecme-rfc7321bis] and successors consider the life
        cycle of cipher suites sufficiently long to provide
        interoperability. Constraint devices may have limited
        interoperability requirements which makes possible to reduces
        the number of cipher suites to implement.

   - Power Consumption and Cipher Suite Complexity :   Complexity of the
        cipher suite or the energy associated to it are especially
        considered when devices have limited resources or are using
        some batteries, in which case the battery determines the life
        of the device. The choice of a cryptographic function may
        consider re-using specific libraries or to take advantage of
        hardware acceleration provided by the device. For example if
        the device benefits from AES hardware modules and uses AES-CTR,
        it may prefer AUTH_AES-XCBC for its authentication. In
        addition, some devices may also embed radio modules with
        hardware acceleration for AES-CCM, in which case, this mode may
        be preferred.

   - Power Consumption and Bandwidth Consumption :   Similarly to the
        cipher suite complexity, reducing the payload sent, may
        significantly reduce the energy consumption of the device. As
        a result, cipher suites with low overhead may be considered.
        To reduce the overall payload size one may for example, one MAY
        consider:

        a Use of counter-based ciphers without fixed block length
             (e.g. AES-CTR, or ChaCha20-Poly1305)

        b Use of ciphers with capability of using implicit IVs
             [I-D.mglt-ipsecme-implicit-iv]

        c Use of ciphers recommended for IoT
             [I-D.ietf-ipsecme-rfc7321bis].

        d Avoid Padding by sending payload data which are aligned to
             the cipher block length -2 for the ESP trailer.

9. IANA Considerations

   There are no IANA consideration for this document.

10. Security Considerations

   Security considerations are those of [RFC4303].

11. Acknowledgment

12. References

12.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC3602] Frankel, S., Glenn, R., and S. Kelly, "The AES-CBC Cipher
              Algorithm and Its Use with IPsec", RFC 3602,
              DOI 10.17487/RFC3602, September 2003,
              <http://www.rfc-editor.org/info/rfc3602>.

   [RFC3686] Housley, R., "Using Advanced Encryption Standard (AES)
              Counter Mode With IPsec Encapsulating Security Payload
              (ESP)", RFC 3686, DOI 10.17487/RFC3686, January 2004,
              <http://www.rfc-editor.org/info/rfc3686>.

   [RFC4106] Viega, J. and D. McGrew, "The Use of Galois/Counter Mode
              (GCM) in IPsec Encapsulating Security Payload (ESP)",
              RFC 4106, DOI 10.17487/RFC4106, June 2005,
              <http://www.rfc-editor.org/info/rfc4106>.

   [RFC4301] Kent, S. and K. Seo, "Security Architecture for the
              Internet Protocol", RFC 4301, DOI 10.17487/RFC4301,
              December 2005, <http://www.rfc-editor.org/info/rfc4301>.

   [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)",
              RFC 4303, DOI 10.17487/RFC4303, December 2005,
              <http://www.rfc-editor.org/info/rfc4303>.

   [RFC4309] Housley, R., "Using Advanced Encryption Standard (AES) CCM
              Mode with IPsec Encapsulating Security Payload (ESP)",
              RFC 4309, DOI 10.17487/RFC4309, December 2005,
              <http://www.rfc-editor.org/info/rfc4309>.

   [RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T.
              Kivinen, "Internet Key Exchange Protocol Version 2
              (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October
              2014, <http://www.rfc-editor.org/info/rfc7296>.

   [RFC7815]   Kivinen, T., "Minimal Internet Key Exchange Version 2
               (IKEv2) Initiator Implementation", RFC 7815,
               DOI 10.17487/RFC7815, March 2016,
               <http://www.rfc-editor.org/info/rfc7815>.

12.2. Informative References

   [I-D.ietf-ipsecme-rfc7321bis]
               Migault, D., Mattsson, J., Wouters, P., Nir, Y., and T.
               Kivinen, "Cryptographic Algorithm Implementation
               Requirements and Usage Guidance for Encapsulating Security
               Payload (ESP) and Authentication Header (AH)", draft-ietf-
               ipsecme-rfc7321bis-05 (work in progress), February 2017.

   [I-D.mglt-ipsecme-implicit-iv]
               Migault, D., Guggemos, T., and Y. Nir, "Implicit IV for
               Counter-based Ciphers in IPsec", draft-mglt-ipsecme-
               implicit-iv-02 (work in progress), November 2016.

Appendix A. Document Change Log

   [RFC Editor: This section is to be removed before publication]

   -00: First version published.

   -01: Clarified description

   -02: Clarified description

Authors' Addresses

   Daniel Migault (editor)
   Ericsson
   8400 boulevard Decarie
   Montreal, QC H4P 2N2
   Canada

   Email: daniel.migault@ericsson.com

   Tobias Guggemos
   LMU Munich
   MNM-Team
   Oettingenstr. 67
   80538 Munich, Bavaria
   Germany

   Email: guggemos@mnm-team.org