                   Diet-ESP: Generating compressed IV and SN
                  draft-mglt-ipsecme-diet-esp-iv-generation-00.txt

Abstract

   Diet-ESP describes how to compress the various ESP fields, thanks to
   the Diet-ESP Context. This document describes how the IV fields that
   belong to the encrypted payload can be compressed.

   The document describes the extensions of the the Diet-ESP Context as
   well as the compression protocol.

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 January 4, 2015.

the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

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

Diet-ESP [I-D.mglt-ipsecme-diet-esp] describes how to compress ESP
fields. Fields are compressed according to a Diet-ESP Context.
Diet-ESP has been described as a specific ROHC [RFC5795] framework
that has no IR, IR-DYN nor any feed back ROHC message. It works in
the Unidirectional mode of operation (U mode), and all necessary
parameters are transmitted via the Diet-ESP Context that is
negotiated between the two peers. As a result Diet-ESP defines a
very specific and simplified ROHC framework which makes possible to
implement Diet-ESP without implementing the whole ROHC.

In fact, Diet-ESP avoids ROHC complexity as a lot of parameters have
already been negotiated with IKEv2 [RFC5996].

The Initialization Vector (IV) is defined as a input for AES
encryption and decryption. In order to provide the appropriated IV
value AES-CBC [RFC3602] and AES-CTR [RFC3686] sends the IV in each IP
packet as shown in figure Figure 1. In fact the output of AES-CTR
and AES-CBC outputs a payload where the encrypted data is appended to
the IV.

The IV MUST have to properties 1) they MUST be unpredictable by someone observing the network, then 2) the IV MUST be unique. The size of the IV differs depending on the encryption algorithm. AES-CTR has an 8 byte IV and AES-CBC a 16 byte IV.

This document defines a way to avoid sending the IV in each packet. Instead peers agree on a suite of pseudo random bytes. This makes the IV predictable by both peers only, and random to the rest of the world. As the IV can be derived by both peers, it may be removed completely from each IP packet. Another way is to only provide the LSB of the generated IV so receiver can better identify the appropriated IV used for decryption.

Note that the ICV of standard ESP [RFC4303] and Diet-ESP ICV includes the whole IV. As a result, the IV MUST be restored prior to the ICV check.

```
 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)                 | ^
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
|                      Sequence Number                          | |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
|                                                               | |Int.
~               Initialization Vector (variable)                ~ |Cov-
|                                                               | |ered
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
|                                                               | |
~                 Encrypted Payload (variable)                  ~ |
|                                                               | v
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ ------
|                                                               |
~                Authentication Data (variable)                 ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
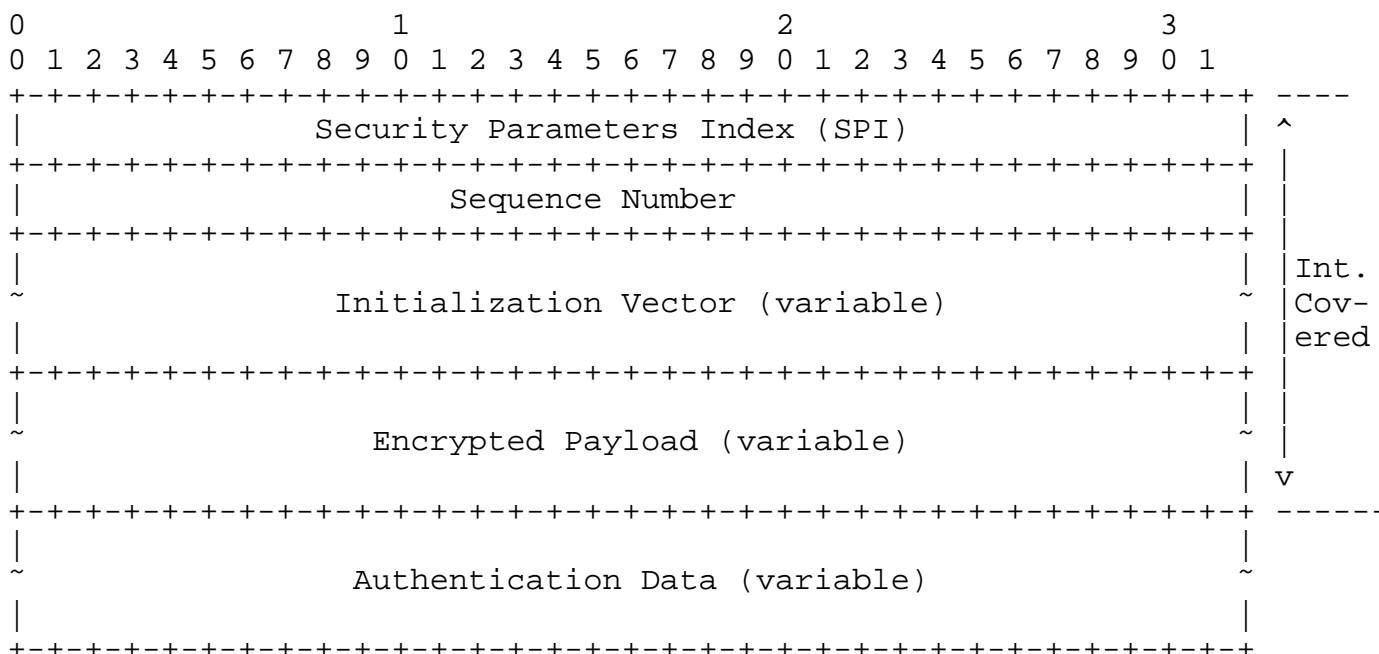
                   Figure 1: The IV in the ESP payload.

Section 4 describes the new parameters for the Diet-ESP Context. Section 5 describes how the Pseudo Random Function is derived, and Section 6 describes the protocol.

3. Terminology

   - IoT: Internet of Things

   - IV: Initialization Vector

- ICV: Integrity Check Value

- PRF: Pseudo Random Function

4. Diet-ESP context extension

   To enable the compression of the IV, the Diet-ESP context defined in
   [I-D.mglt-ipsecme-diet-esp] is extended with to values:

   IV_COMPRESSION:
      Defines if the IV is generated and compresses.

   IV_PRFT (optional):
      Defines the Pseudo Random Function Transform used for the Pseudo
      Random Function. Available IDs are defined in [1]
      Section Transform Type 2 - Pseudo random Function Transform IDs.
      Section 2.13 [RFC5996] defines how the PRF is derived. By default
      PRF_AES_128_CBC is the Pseudo Random Function Transform
      considered.

   IV_LSB:
      Defines the number of Least Significant Bytes of the IV carried by
      the payload.

5. Pseudo Random Function

   The Pseudo Random Function (PRF) is defined from the Pseudo Random
   Function Transform in Section 2.13 [RFC5996]. Unless specified
   otherwise PRF_AES128_XCBC [RFC4434] is the default Pseudo Random
   Function Transform.

   The PRF "prf+" described in Section 2.13 [RFC5996] takes two
   arguments designated as K and S. In this document K is the
   encryption key and S is the authentication key appended to the string
   "IV random generation". The string results in non null S value even
   if no integrity algorithms are negotiated.

6. Protocol Description

   IV generation and compression is performed only and only if
   IV_COMPRESSION is set. Otherwise, the IV is embedded into the packet
   and sent on the wire as described in [RFC4303].

   When IV_COMPRESSION is set, the PRD is defined as described in
   Section 5. On the sending part, the ICV or Diet-ESP ICV is computed,
   the IV is compressed to its LSB, before it is sent on the wire. On
   te receiver part, the IV is decompress prior to the ICV check, then
   decryption is performed with the decompressed IV.

7. IANA Considerations

   There are no IANA consideration for this document.

8. Security Considerations

9. Acknowledgment

   The current draft represents the work of Tobias Guggemos while his
   internship at Orange [GUGG14] .

   Diet-ESP is a joint work between Orange and Ludwig-Maximilians-
   Universitaet Munich. We thank Daniel Palomares and Carsten Bormann
   for their useful remarks, comments and guidance.

10. References

10.1. Normative References

   [I-D.mglt-ipsecme-diet-esp]
            Migault, D., Guggemos, T., and D. Palomares, "Diet-ESP: a
            flexible and compressed format for IPsec/ESP", draft-mglt-
            ipsecme-diet-esp-00 (work in progress), March 2014.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
            Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC3602] Frankel, S., Glenn, R., and S. Kelly, "The AES-CBC Cipher
            Algorithm and Its Use with IPsec", RFC 3602, September
            2003.

   [RFC3686] Housley, R., "Using Advanced Encryption Standard (AES)
            Counter Mode With IPsec Encapsulating Security Payload
            (ESP)", RFC 3686, January 2004.

   [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC
            4303, December 2005.

   [RFC4434] Hoffman, P., "The AES-XCBC-PRF-128 Algorithm for the
            Internet Key Exchange Protocol (IKE)", RFC 4434, February
            2006.

   [RFC5795] Sandlund, K., Pelletier, G., and L-E. Jonsson, "The RObust
            Header Compression (ROHC) Framework", RFC 5795, March
            2010.

   [RFC5996]   Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen,
               "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC
               5996, September 2010.

10.2. Informational References

   [GUGG14]   Guggemos, TG., "Diet-ESP: Applying IP-Layer Security in
               Constrained Environments (Masterthesis)", September 2014.

10.3. URIs

   [1] http://www.iana.org/assignments/ikev2-parameters/
       ikev2-parameters.xhtml#ikev2-parameters-6

Appendix A. Document Change Log

   [draft-mglt-ipsecme-diet-esp-IV-generation-00.txt]: First version
   published.

Authors' Addresses

   Daniel Migault (editor)
   Orange
   38 rue du General Leclerc
   92794 Issy-les-Moulineaux Cedex 9
   France

   Phone: +33 1 45 29 60 52
   Email: daniel.migault@orange.com

   Tobias Guggemos (editor)
   Orange / LMU Munich
   Am Osteroesch 9
   87637 Seeg, Bavaria
   Germany

   Email: tobias.guggemos@gmail.com