IPSECME   D. Migault
Internet-Draft   Ericsson
Intended status: Standards Track   T. Guggemos
Expires: December 28, 2018   LMU Munich
                                                                  Y. Nir
                                                                Dell EMC
                                                           June 26, 2018

    Implicit IV for Counter-based Ciphers in Encapsulating Security Payload
                                   (ESP)
                     draft-ietf-ipsecme-implicit-iv-05

Abstract

   Encapsulating Security Payload (ESP) sends an initialization vector
   (IV) or nonce in each packet. The size of IV depends on the applied
   transform, being usually 8 or 16 octets for the transforms defined by
   the time this document is written. Some algorithms such as AES-GCM,
   AES-CCM, AES-CTR and ChaCha20-Poly1305 require a unique nonce but do
   not require an unpredictable nonce. When using such algorithms the
   packet counter value can be used to generate a nonce. This avoids
   sending the nonce itself, and saves in the case of AES-GCM, AES-CCM,
   AES-CTR and ChaCha20-Poly1305 8 octets per packet. This document
   describes how to do this.

Status of This Memo

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

Counter-based AES modes of operation such as AES-CTR ([RFC3686]),
AES-CCM ([RFC4309]), and AES-GCM ([RFC4106]) require the
specification of an nonce for each ESP packet. The same applies for
ChaCha20-Poly1305 ([RFC7634]). Currently this nonce is sent in each
ESP packet ([RFC4303]). This practice is designated in this document
as "explicit nonce".

In some context, such as IoT, it may be preferable to avoid carrying
the extra bytes associated to the IV and instead generate it locally
on each peer. The local generation of the nonce is designated in
this document as "implicit IV".

The size of this nonce depends on the specific algorithm, but all of
the algorithms mentioned above take an 8-octet nonce.

This document defines how to compute the nonce locally when it is
implicit. It also specifies how peers agree with the Internet Key
Exchange version 2 (IKEv2 - [RFC7296]) on using an implicit IV versus
an explicit IV.

This document limits its scope to the algorithms mentioned above.
Other algorithms with similar properties may later be defined to use
this extension.

This document does not consider AES-CBC ([RFC3602]) as AES-CBC
requires the IV to be unpredictable. Deriving it directly from the
packet counter as described below is insecure as mentioned in
Security Consideration of [RFC3602] and has led to real world chosen
plain-text attack such as BEAST [BEAST].

3. Terminology

o   IoT: Internet of Things.

o   IV: Initialization Vector.

o   IIV: Implicit Initialization Vector.

o   Nonce: a fixed-size octet string used only once. This is similar
    to IV, except that in common usage there is no implication of non-
    predictability.

4. Implicit IV

With the algorithms listed in Section 2, the 8 byte nonce MUST NOT
repeat. The binding between a ESP packet and its nonce is provided
using the Sequence Number or the Extended Sequence Number. Figure 1
and Figure 2 represent the IV with a regular 4-byte Sequence Number
and with an 8-byte Extended Sequence Number respectively.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Zero                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
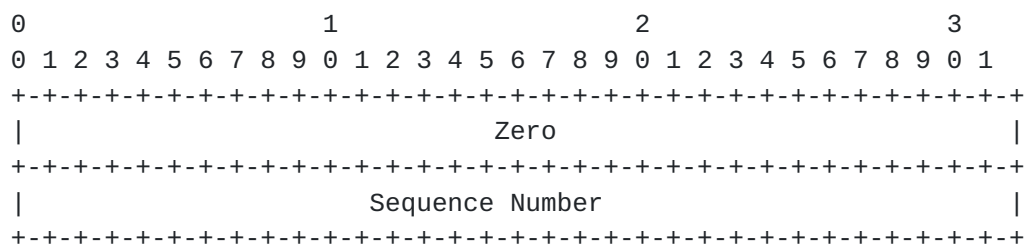
Figure 1: Implicit IV with a 4 byte Sequence Number

o   Sequence Number: the 4 byte Sequence Number carried in the ESP
    packet.

o Zero: a 4 byte array with all bits set to zero.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Extended                           |
|                       Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
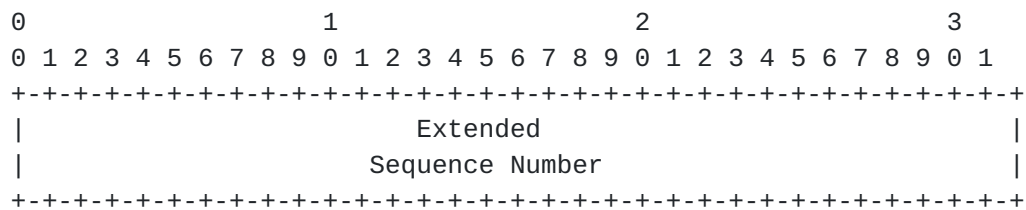
Figure 2: Implicit IV with an 8 byte Extended Sequence Number

o Extended Sequence Number: the 8 byte Extended Sequence Number of
   the Security Association. The 4 byte low order bytes are carried
   in the ESP packet.

5. Initiator Behavior

An initiator supporting this feature SHOULD propose implicit IV
algorithms in the Transform Type 1 (Encryption Algorithm)
Substructure of the Proposal Substructure inside the SA Payload. To
facilitate backward compatibility with non-supporting peers the
initiator SHOULD also include those same algorithms without Implicit
IV (IIV) as separate transforms.

6. Responder Behavior

The rules of SA Payload processing require that responder picks its
algorithms from the proposal sent by the initiator, thus this will
ensure that the responder will never send an SA payload containing
the IIV transform to an initiator that did not propose it.

7. Security Consideration

Nonce generation for these algorithms has not been explicitly
defined. It has been left to the implementation as long as certain
security requirements are met. Typically, for AES-GCM, AES-CCM, AES-
CTR and ChaCha20-Poly1305, the IV is not allowed being repeated for
one particular key. This document provides an explicit and normative
way to generate IVs. The mechanism described in this document meets
the IV security requirements of all relevant algorithms.

As the IV must not repeat for one SA when Counter-Mode ciphers are
used, Implicit IV as described in this document MUST NOT be used in
setups with the chance that the Sequence Number overlaps for one SA.
Multicast as described in [RFC5374], [RFC6407] and
[I-D.yeung-g-ikev2] is a prominent example, where many senders share
one secret and thus one SA. As such, Implicit IV may only be used
with Multicast if some mechanisms are employed that prevent Sequence

Number to overlap for one SA, otherwise Implicit IV MUST NOT be used with Multicast.

This document defines three new encryption transforms that use implicit IV. Unlike most encryption transforms defined to date, which can be used for both ESP and IKEv2, these transforms are defined for ESP only and cannot be used in IKEv2. The reason is that IKEv2 messages don't contain unique per-message value, that can be used for IV generation. The Message-ID field in IKEv2 header is somewhat counterpart of SN field in ESP header, but recent IKEv2 extensions ([RFC6311], [RFC7383]) do allow it to repeat, so there is no an easy way to derive unique IV from IKEv2 header fields.

8. IANA Considerations

This section assigns new code points to the recommended AEAD suites provided in [RFC8221], thus the new Transform Type 1 - Encryption Algorithm Transform IDs [IANA] are as defined below:

- ENCR_AES_CCM_8_IIV: 29

- ENCR_AES_GCM_16_IIV: 30

- ENCR_CHACHA20_POLY1305_IIV: 31

These algorithms should be added with this document as ESP Reference and "Not Allowed" for IKEv2 Reference.

9. Acknowledgements

We would like to thanks people Valery Smyslov for their valuable comments, David Schinazi for its implementation, as well as the ipseceme chairs Tero Kivinen and David Waltermire for moving this work forward.

10. References

10.1. 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, <https://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, <https://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,
              <https://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,
              <https://www.rfc-editor.org/info/rfc4106>.

   [RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)",
              RFC 4303, DOI 10.17487/RFC4303, December 2005,
              <https://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,
              <https://www.rfc-editor.org/info/rfc4309>.

   [RFC5374] Weis, B., Gross, G., and D. Ignjatic, "Multicast
              Extensions to the Security Architecture for the Internet
              Protocol", RFC 5374, DOI 10.17487/RFC5374, November 2008,
              <https://www.rfc-editor.org/info/rfc5374>.

   [RFC6311] Singh, R., Ed., Kalyani, G., Nir, Y., Sheffer, Y., and D.
              Zhang, "Protocol Support for High Availability of IKEv2/
              IPsec", RFC 6311, DOI 10.17487/RFC6311, July 2011,
              <https://www.rfc-editor.org/info/rfc6311>.

   [RFC6407] Weis, B., Rowles, S., and T. Hardjono, "The Group Domain
              of Interpretation", RFC 6407, DOI 10.17487/RFC6407,
              October 2011, <https://www.rfc-editor.org/info/rfc6407>.

   [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, <https://www.rfc-editor.org/info/rfc7296>.

   [RFC7383] Smyslov, V., "Internet Key Exchange Protocol Version 2
              (IKEv2) Message Fragmentation", RFC 7383,
              DOI 10.17487/RFC7383, November 2014,
              <https://www.rfc-editor.org/info/rfc7383>.

   [RFC7634] Nir, Y., "ChaCha20, Poly1305, and Their Use in the
              Internet Key Exchange Protocol (IKE) and IPsec", RFC 7634,
              DOI 10.17487/RFC7634, August 2015,
              <https://www.rfc-editor.org/info/rfc7634>.

[RFC8221] Wouters, P., Migault, D., Mattsson, J., Nir, Y., and T.
           Kivinen, "Cryptographic Algorithm Implementation
           Requirements and Usage Guidance for Encapsulating Security
           Payload (ESP) and Authentication Header (AH)", RFC 8221,
           DOI 10.17487/RFC8221, October 2017,
           <https://www.rfc-editor.org/info/rfc8221>.

10.2. Informational References

[BEAST]   Thai, T. and J. Juliano, "Here Come The xor Ninjas", ,
           May 2011, <https://www.researchgate.net/
           publication/266529975_Here_Come_The_Ninjas>.

[I-D.yeung-g-ikev2]
           Weis, B., Nir, Y., and V. Smyslov, "Group Key Management
           using IKEv2", draft-yeung-g-ikev2-13 (work in progress),
           March 2018.

[IANA]   "IANA IKEv2 Parameter - Type 1 - Encryption Algorithm
           Transform IDs", <https://www.iana.org/assignments/ikev2-
           parameters/ikev2-parameters.xhtml#ikev2-parameters-5>.

Authors' Addresses

   Daniel Migault
   Ericsson
   8275 Trans Canada Route
   Saint Laurent, QC H4S 0B6
   Canada

   Email: daniel.migault@ericsson.com

   Tobias Guggemos
   LMU Munich
   Oettingenstr. 67
   80538 Munich, Bavaria
   Germany

   Email: guggemos@mnm-team.org
   URI: http://mnm-team.org/~guggemos

Yoav Nir
Dell EMC
9 Andrei Sakharov St
Haifa 3190500
Israel

Email: ynir.ietf@gmail.com