IPSECME
Internet-Draft
Intended status: Standards Track
Expires: April 18, 2020
D. Migault
Ericsson
T. Guggemos
LMU Munich
Y. Nir
Dell EMC
October 16, 2019

Implicit IV for Counter-based Ciphers in Encapsulating Security Payload (ESP)

draft-ietf-ipsecme-implicit-iv-08

Abstract

Encapsulating Security Payload (ESP) sends an initialization vector (IV) 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 and ChaCha20-Poly1305 when used with IPsec, take the IV to generate a nonce that is used as an input parameter for encrypting and decrypting. These algorithms require a unique IV but do not require an unpredictable IV. As a result, the value provided in the ESP Sequence Number (SN) can be used instead to generate the nonce. This avoids sending the IV itself, and saves in the case of AES-GCM, AES-CCM 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", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described BCP 14 [RFC2119], [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Introduction

Counter-based AES modes of operation such as 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 generated thanks to the Initialize Vector (IV) provided in each ESP packet ([RFC4303]). This practice is designated in this document as "explicit IV".

In some contexts, 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 IV is designated in this document as "implicit IV".

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

This document defines how to compute the IV 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 similar mechanism.

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].

This document does not consider AES-CTR [RFC3686] as it focuses on the recommended AEAD suites provided in [RFC8221].

3. Terminology

- IoT: Internet of Things.

- IV: Initialization Vector.

- IIV: Implicit Initialization Vector.

- Nonce: a fixed-size octet string used only once. In our case, the nonce takes the IV as input and is provided as an input parameter for encryption/decryption.

4. Implicit IV

With the algorithms listed in Section 2, the 8-byte IV MUST NOT repeat for a given key. The binding between an ESP packet and its IV 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.

This document solely defines the IV generation of the algorithms defined in [RFC4106] for AES-GCM, [RFC4309] for AES-CCM and [RFC7634] for ChaCha20-Poly1305. All other aspects and parameters of those algorithms are unchanged, and are used as defined in their respective specifications.

5. IKEv2 Initiator Behavior

An initiator supporting this feature SHOULD propose implicit IV (IIV) algorithms in the Transform Type 1 (Encryption Algorithm) Substructure of the Proposal Substructure inside the Security Association Payload (SA Payload) in the IKEv2 Exchange. To facilitate backward compatibility with non-supporting peers the initiator SHOULD also include those same algorithms with explicit IV as separate transforms.

6. IKEv2 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 Considerations

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 and ChaCha20-Poly1305, the IV is not allowed to be 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. 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 for an SA that uses a non extended Sequence Number (respectively the 2^64nd packet for an SA that uses an Extended Sequence Number). This prevents sequence number overlaps for the mundane point-to-point case. 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 a unique per-message value that can be used for IV generation. The Message-ID field in IKEv2 header is similar to the SN field in ESP header, but recent IKEv2 extensions ([RFC6311], [RFC7383]) do allow it to repeat, so there is not an easy way to derive unique IV from IKEv2 header fields.

8. IANA Considerations

The IANA has assigned the following code points:

- 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 thank Valery Smyslov, Eric Vyncke, Magnus Nystrom (security directorate), as well as our three ADs Eric Rescorla, Benjamin Kaduk and Roman Danyliw for their valuable comments. We also would like to thank David Schinazi for its implementation, as well as the ipseceme chairs Tero Kivinen and David Waltermire for moving this work forward.

NOTE TO THE EDITOR Eric has a accent on E and Magnus has double points on o.

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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[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. and V. Smyslov, "Group Key Management using IKEv2", draft-yeung-g-ikev2-16 (work in progress), July 2019.

[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