Stream:	Internet Engineering Task Force (IETF)		
RFC:	8750		
Category:	Standards Track		
Published:	March 2020		
ISSN:	2070-1721		
Authors:	D. Migault	T. Guggemos	Y. Nir
	Ericsson	*LMU Munich*	*Dell Technologies*

RFC 8750
Implicit Initialization Vector (IV) for Counter-Based Ciphers in Encapsulating Security Payload (ESP)

Abstract

Encapsulating Security Payload (ESP) sends an initialization vector (IV) in each packet. The size of the IV depends on the applied transform and is usually 8 or 16 octets for the transforms defined at the time this document was written. When used with IPsec, some algorithms, such as AES-GCM, AES-CCM, and ChaCha20-Poly1305, take the IV to generate a nonce that is used as an input parameter for encrypting and decrypting. This IV must be unique but can be predictable. 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 8 octets per packet in the case of AES-GCM, AES-CCM, and ChaCha20-Poly1305. This document describes how to do this.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8750.

Copyright Notice

Table of Contents

1. Introduction

Counter-based AES modes of operation such as AES-CCM [RFC4309] and AES-GCM [RFC4106] require the specification of a nonce for each ESP packet. The same applies for ChaCha20-Poly1305 [RFC7634]. Currently, this nonce is generated thanks to the initialization vector (IV) provided in each ESP packet [RFC4303]. This practice is designated in this document as "explicit IV".

In some contexts, such as the Internet of Things (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 mechanisms.

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 Section 6 of [RFC3602], and has led to real-world chosen plaintext attacks such as BEAST [BEAST].

This document does not consider AES-CTR [RFC3686], as it focuses on the recommended Authenticated Encryption with Associated Data (AEAD) suites provided in [RFC8221].

2. 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 in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

IoT: Internet of Things

IV: Initialization Vector

IIV: Implicit Initialization Vector

Nonce: A fixed-size octet string used only once. In this document, the IV is used to generate the nonce input for the encryption/decryption.

4. Implicit IV

With the algorithms listed in Section 1, 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. Figures 1 and 2 represent the IV with a regular 4-byte Sequence Number and an 8-byte Extended Sequence Number, respectively.

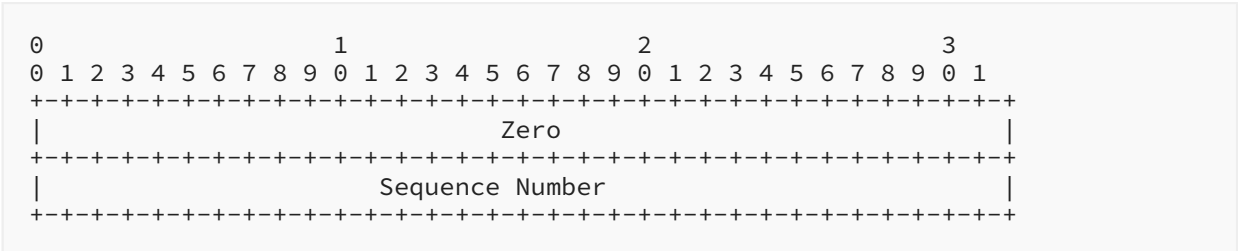

Figure 1: Implicit IV with a 4-Byte Sequence Number

Sequence Number:
: The 4-byte Sequence Number carried in the ESP packet.

Zero:
: A 4-byte array with all bits set to zero.

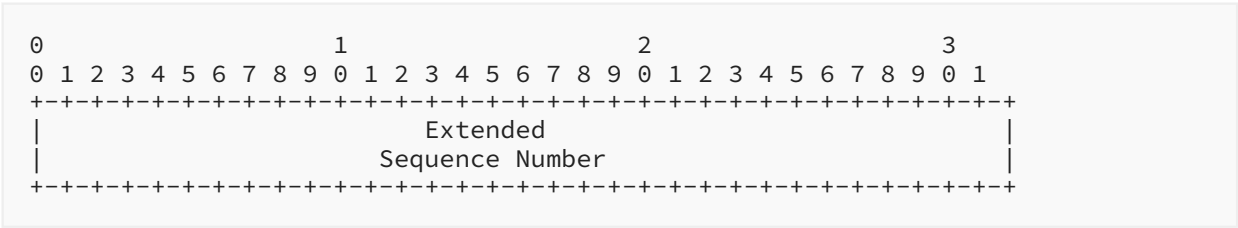

Figure 2: Implicit IV with an 8-Byte Extended Sequence Number

Extended Sequence Number:
: The 8-byte Extended Sequence Number of the Security Association. The four 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 (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 the responder pick its algorithms from the proposal sent by the initiator, thus ensuring 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 does not use an Extended Sequence Number and prior to the transmission of the 2^64th packet for an SA that does use an Extended Sequence Number. This prevents Sequence Number overlaps for the mundane point-to-point case. Multicast as described in [RFC5374], [RFC6407], and [G-IKEv2] is a prominent example in which 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 the Sequence Number from overlapping 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 for this is that IKEv2 messages don't contain a unique per-message value that can be used for IV generation. The Message-ID field in the IKEv2 header is similar to the SN field in the 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

IANA has updated the "Internet Key Exchange Version 2 (IKEv2) Parameters" registry [RFC7296] by adding the following new code points to the "Transform Type 1 - Encryption Algorithm Transform IDs" subregistry under the "Transform Type Values" registry [IANA]:

Number	Name	ESP Reference	IKEv2 Reference
29	ENCR_AES_CCM_8_IIV	RFC 8750	Not allowed
30	ENCR_AES_GCM_16_IIV	RFC 8750	Not allowed
31	ENCR_CHACHA20_POLY1305_IIV	RFC 8750	Not allowed

Table 1: Additions to "Transform Type 1 - Encryption Algorithm Transform IDs" Registry

9. References

9.1. 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>.

9.2. Informative References

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

[G-IKEv2] Weis, B. and V. Smyslov, "Group Key Management using IKEv2", Work in Progress, Internet-Draft, draft-ietf-ipsecme-g-ikev2-00, 8 January 2020, <https://tools.ietf.org/html/draft-ietf-ipsecme-g-ikev2-00>.

[IANA] IANA, "Internet Key Exchange Version 2 (IKEv2) Parameters", <https://www.iana.org/assignments/ikev2-parameters>.

Acknowledgements

We would like to thank Valery Smyslov, Éric Vyncke, Alexey Melnikov, Adam Roach, and Magnus Nyström (security directorate) as well as our three Security ADs -- Eric Rescorla, Benjamin Kaduk, and Roman Danyliw -- for their valuable comments. We also would like to thank David Schinazi for his implementation as well as Tero Kivinen and David Waltermire (the IPSECME Chairs) for moving this work forward.

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
Germany
Email: guggemos@nm.ifi.lmu.de
URI: http://mnm-team.org/~guggemos

Yoav Nir
Dell Technologies
9 Andrei Sakharov St
Haifa 3190500
Israel
Email: ynir.ietf@gmail.com