IPSECME
Internet-Draft
Intended status: Standards Track
Expires: December 11, 2016

D. Migault, Ed.
Ericsson
T. Guggemos, Ed.
LMU Munich
Y. Nir
Check Point
June 9, 2016

Implicit IV for Counter-based Ciphers in IPsec
draft-mglt-ipsecme-implicit-iv-00.txt

Abstract

IPsec ESP sends an initialization vector (IV) or nonce in each packet, adding 8 or 16 octets. 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, saving 8 octets per packet. This document describes how to do this.

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 December 11, 2016.

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 ([RFC4104]) 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 compute 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 to negotiate this behavior within the Internet Key Exchange version 2 (IKEv2 - [RFC7296]).

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.

3. Terminology

- IoT: Internet of Things
- IV: Initialization Vector. Although security requirements vary, the common usage of this term implies that the value is unpredictable.
- 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.

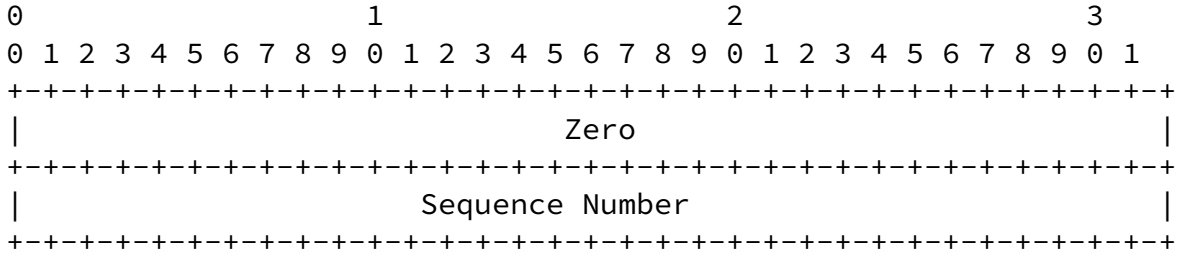

Figure 1: Implicit IV with a 4 byte Sequence Number

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

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

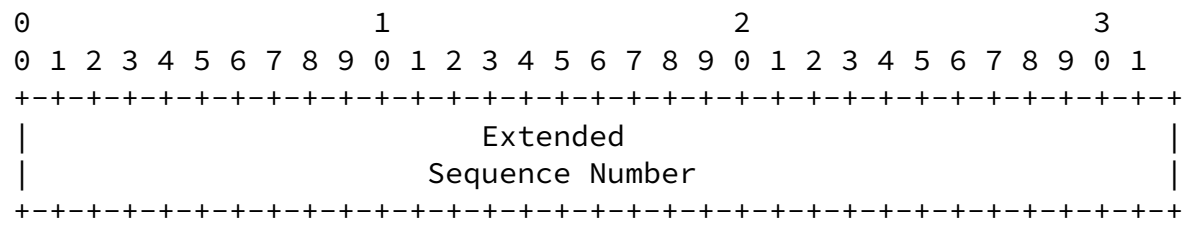

Figure 2: Implicit IV with an 8 byte Extended Sequence Number

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

5. Implicit IV Agreement

NOTE: THIS SECTION SHOWS SEVERAL WAYS TO DO THE SAME THING. OBVIOUSLY THIS DOCUMENT WILL NOT BE PUBLISHED LIKE THAT. WE EXPECT WG DISCUSSION TO PARE THIS DOWN TO JUST ONE WAY OF NEGOTIATING THE USE OF IMPLICIT IV (IIV).

Negotiation of the use of implicit IV can be done in 3 different ways:

- An IMPLICIT IV Transform Type. A proposal that contains this transform type requires (if accepted) that IPsec use the implicit IV and not include an explicit IV in the packets. To facilitate backward compatibility with non-supporting implementations the Initiator SHOULD add another proposal that does not include this transform type as well as cryptographic suite the Initiator does

not support the implicit IV.

- An IMPLICIT IV Transform ID. This doubles the number of ENCR transform IDs by creating an ENCR_AES_CCM_16_IIV for each ENCR_AES_CCM_16.

- An IMPLICIT IV Transform Attribute, which would be associated to a Transform Type ID, specifying the use of an implicit IV. marks a particular ENCR transform as using implicit IVs. To facilitate backward compatibility with non-supporting implementations the Initiator SHOULD add another ENCR transform for each algorithm so marked. In other words, for each ENCR_AES_CCM_16 with keylength=256 and IIV=1, there will need to be an ENCR_AES_CCM_16 with keylength=256 and no IIV attribute.

6. Initiator Behavior

An initiator supporting this feature SHOULD propose implicit IV for all relevant algorithms. To facilitate backward compatibility with non-supporting peers the initiator SHOULD also include those same algorithms without IIV. Depending on the method chosen in Section 5 this may require extra proposals or extra transforms.

7. Responder Behavior

An responder supporting this feature SHOULD accept implicit IV for all relevant algorithms. It MUST NOT accept implicit IV for algorithms not specified to be safe for IIV in this or subsequent documents. It SHOULD accept non-IIV proposals for all algorithms if IIV proposals were not included.

The rules of SA payload processing ensure that the responder will never send an SA payload containing the IIV indicator to an initiator that does not support IIV.

8. 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. 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.

9. IANA Considerations

TBD: The content of this section depends on our decisions about Section 5.

The following Transform Type is requested: IMPLICIT_IV Transform Type

Values associated to IMPLICIT Transform Type are:

False: 0 ===> Do we really need a negative value???

True: 1

10. 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>.

[RFC4104] Pana, M., Ed., Reyes, A., Barba, A., Moron, D., and M. Brunner, "Policy Core Extension Lightweight Directory Access Protocol Schema (PCELS)", RFC 4104, DOI 10.17487/RFC4104, June 2005, <http://www.rfc-editor.org/info/rfc4104>.

[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>.

[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, <http://www.rfc-editor.org/info/rfc7634>.

Appendix A. Document Change Log

Authors' Addresses

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

Email: daniel.migault@ericsson.com

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

Email: tobias.guggemos@gmail.com

Yoav Nir
Check Point Software Technologies Ltd.
5 Hasolelim st.
Tel Aviv 6789735
Israel

Email: ynir.ietf@gmail.com