Network Working Group
Request for Comments: 4543
Category: Standards Track

D. McGrew
Cisco Systems, Inc.
J. Viega
McAfee, Inc.

May 2006

The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This memo describes the use of the Advanced Encryption Standard (AES) Galois Message Authentication Code (GMAC) as a mechanism to provide data origin authentication, but not confidentiality, within the IPsec Encapsulating Security Payload (ESP) and Authentication Header (AH). GMAC is based on the Galois/Counter Mode (GCM) of operation, and can be efficiently implemented in hardware for speeds of 10 gigabits per second and above, and is also well-suited to software implementations.

Table of Contents

1. Introduction

This document describes the use of AES-GMAC mode (AES-GMAC) as a mechanism for data origin authentication in ESP [RFC4303] and AH [RFC4302]. We refer to these methods as ENCR_NULL_AUTH_AES_GMAC and AUTH_AES_GMAC, respectively. ENCR_NULL_AUTH_AES_GMAC is a companion to the AES Galois/Counter Mode ESP [RFC4106], which provides authentication as well as confidentiality. ENCR_NULL_AUTH_AES_GMAC is intended for cases in which confidentiality is not desired. Like GCM, GMAC is efficient and secure, and is amenable to high-speed implementations in hardware. ENCR_NULL_AUTH_AES_GMAC and AUTH_AES_GMAC are designed so that the incremental cost of implementation, given an implementation is AES-GCM-ESP, is small.

This document does not cover implementation details of GCM or GMAC. Those details can be found in [GCM], along with test vectors.

1.1. Conventions Used in This Document

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. AES-GMAC

GMAC is a block cipher mode of operation providing data origin authentication. It is defined in terms of the GCM authenticated encryption operation as follows. The GCM authenticated encryption operation has four inputs: a secret key, an initialization vector (IV), a plaintext, and an input for additional authenticated data (AAD). It has two outputs, a ciphertext whose length is identical to the plaintext and an authentication tag. GMAC is the special case of GCM in which the plaintext has a length of zero. The (zero-length) ciphertext output is ignored, of course, so that the only output of the function is the Authentication Tag. In the following, we describe how the GMAC IV and AAD are formed from the ESP and AH fields, and how the ESP and AH packets are formed from the Authentication Tag.

Below we refer to the AES-GMAC IV input as a nonce, in order to distinguish it from the IV fields in the packets. The same nonce and key combination MUST NOT be used more than once, since reusing a nonce/key combination destroys the security guarantees of AES-GMAC.

Because of this restriction, it can be difficult to use this mode securely when using statically configured keys. For the sake of good security, implementations MUST use an automated key management system, such as the Internet Key Exchange (IKE) (either version two [RFC4306] or version one [RFC2409]), to ensure that this requirement is met.

3. The Use of AES-GMAC in ESP

The AES-GMAC algorithm for ESP is defined as an ESP "combined mode" algorithm (see Section 3.2.3 of [RFC4303]), rather than an ESP integrity algorithm. It is called ENCR_NULL_AUTH_AES_GMAC to highlight the fact that it performs no encryption and provides no confidentiality.

> Rationale: ESP makes no provision for integrity transforms to place an initialization vector within the Payload field; only encryption transforms are expected to use IVs. Defining GMAC as an encryption transform avoids this issue, and allows GMAC to benefit from the same pipelining as does GCM.

Like all ESP combined modes, it is registered in IKEv2 as an encryption transform, or "Type 1" transform. It MUST NOT be used in conjunction with any other ESP encryption transform (within a particular ESP encapsulation). If confidentiality is desired, then GCM ESP [RFC4106] SHOULD be used instead.

3.1. Initialization Vector

With ENCR_NULL_AUTH_AES_GMAC, an explicit Initialization Vector (IV) is included in the ESP Payload, at the outset of that field. The IV MUST be eight octets long. For a given key, the IV MUST NOT repeat. The most natural way to meet this requirement is to set the IV using a counter, but implementations are free to set the IV field in any way that guarantees uniqueness, such as a linear feedback shift register (LFSR). Note that the sender can use any IV generation method that meets the uniqueness requirement without coordinating with the receiver.

3.2. Nonce Format

The nonce passed to the AES-GMAC authentication algorithm has the following layout:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Salt                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Initialization Vector                     |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Nonce Format

The components of the nonce are as follows:

Salt
   The salt field is a four-octet value that is assigned at the beginning of the security association, and then remains constant for the life of the security association. The salt SHOULD be unpredictable (i.e., chosen at random) before it is selected, but need not be secret. We describe how to set the salt for a Security Association established via the Internet Key Exchange in Section 5.4.

Initialization Vector
   The IV field is described in Section 3.1.

3.3. AAD Construction

Data integrity and data origin authentication are provided for the SPI, (Extended) Sequence Number, Authenticated Payload, Padding, Pad Length, and Next Header fields. This is done by including those fields in the AES-GMAC Additional Authenticated Data (AAD) field. Two formats of the AAD are defined: one for 32-bit sequence numbers, and one for 64-bit extended sequence numbers. The format with 32-bit sequence numbers is shown in Figure 2, and the format with 64-bit extended sequence numbers is shown in Figure 3.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                               SPI                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     32-bit Sequence Number                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   ~                Authenticated Payload (variable)               ~
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Padding (0-255 bytes)                      |
   +                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                               |  Pad Length   | Next Header   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: AAD Format with 32-bit Sequence Number

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                               SPI                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                 64-bit Extended Sequence Number               |
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   ~                Authenticated Payload (variable)               ~
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Padding (0-255 bytes)                      |
   +                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                               |  Pad Length   | Next Header   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: AAD Format with 64-bit Extended Sequence Number

The use of 32-bit sequence numbers vs. 64-bit extended sequence numbers is determined by the security association (SA) management protocol that is used to create the SA. For IKEv2 [RFC4306] this is negotiated via Transform Type 5, and the default for ESP is to use 64-bit extended sequence numbers in the absence of negotiation (e.g., see Section 2.2.1 of [RFC4303]).

3.4. Integrity Check Value (ICV)

The ICV consists solely of the AES-GMAC Authentication Tag. The Authentication Tag MUST NOT be truncated, so the length of the ICV is 16 octets.

3.5. Differences with AES-GCM-ESP

In this section, we highlight the differences between this specification and AES-GCM-ESP [RFC4106]. The essential difference is that in this document, the AAD consists of the SPI, Sequence Number, and ESP Payload, and the AES-GCM plaintext is zero-length, while in AES-GCM-ESP, the AAD consists only of the SPI and Sequence Number, and the AES-GCM plaintext consists of the ESP Payload. These differences are illustrated in Figure 4. This figure shows the case in which the Extended Sequence Number option is not used. When that option is exercised, the Sequence Number field in the figure would be replaced with the Extended Sequence Number.

Importantly, ENCR_NULL_AUTH_AES_GMAC is *not* equivalent to AES-GCM-ESP with encryption "turned off". However, the ICV computations performed in both cases are similar because of the structure of the GHASH function [GCM].

```
                     +-> +-----------------------+ <-+
      AES-GCM-ESP    |   |          SPI          |   |
          AAD -------+   +-----------------------+   |
                     |   |    Sequence Number    |   |
                     +-> +-----------------------+   |
                         |    Authentication     |   |
                         |          IV           |   |
                  +->+-> +-----------------------+   +
      AES-GCM-ESP |      |                       |   |
       Plaintext -+      ~       ESP Payload     ~   |
                  |      |                       |   |
                  |      +-----------+-----+-----+   |
                  |      | Padding   |  PL | NH  |   |
                  +----> +-----------+-----+-----+ <-+
                                                     |
                       ENCR_NULL_AUTH_AES_GMAC AAD --+
```

Figure 4: Differences between ENCR_NULL_AUTH_AES_GMAC and AES-GCM-ESP

3.6. Packet Expansion

The IV adds an additional eight octets to the packet and the ICV adds an additional 16 octets. These are the only sources of packet expansion, other than the 10-13 bytes taken up by the ESP SPI, Sequence Number, Padding, Pad Length, and Next Header fields (if the minimal amount of padding is used).

4. The Use of AES-GMAC in AH

In AUTH_AES_GMAC, the AH Authentication Data field consists of the IV and the Authentication Tag, as shown in Figure 5. Unlike the usual AH case, the Authentication Data field contains both an input to the authentication algorithm (the IV) and the output of the authentication algorithm (the tag). No padding is required in the Authentication Data field, because its length is a multiple of 64 bits.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     Initialization Vector (IV)                |
   |                             (8 octets)                        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   |              Integrity Check Value (ICV) (16 octets)          |
   |                                                               |
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: The AUTH_AES_GMAC Authentication Data Format

The IV is as described in Section 3.1. The Integrity Check Value (ICV) is as described in Section 3.4.

The GMAC Nonce input is formed as described in Section 3.2. The GMAC AAD input consists of the authenticated data as defined in Section 3.1 of [RFC4302]. These values are provided as to that algorithm, along with the secret key, and the resulting authentication tag given as output is used to form the ICV.

5. IKE Conventions

This section describes the conventions used to generate keying material and salt values for use with ENCR_NULL_AUTH_AES_GMAC and AUTH_AES_GMAC using the Internet Key Exchange (IKE) versions one [RFC2409] and two [RFC4306].

5.1. Phase 1 Identifier

This document does not specify the conventions for using AES-GMAC for IKE Phase 1 negotiations. For AES-GMAC to be used in this manner, a separate specification would be needed, and an Encryption Algorithm Identifier would need to be assigned. Implementations SHOULD use an IKE Phase 1 cipher that is at least as strong as AES-GMAC. The use of AES-CBC [RFC3602] with the same AES key size as used by ENCR_NULL_AUTH_AES_GMAC or AUTH_AES_GMAC is RECOMMENDED.

5.2. Phase 2 Identifier

For IKE Phase 2 negotiations, IANA has assigned identifiers as described in Section 9.

5.3. Key Length Attribute

AES-GMAC can be used with any of the three AES key lengths. The way that the key length is indicated is different for AH and ESP.

For AH, each key length has its own separate integrity transform identifier and algorithm name (Section 9). The IKE Key Length attribute MUST NOT be used with these identifiers. This transform MUST NOT be used with ESP.

For ESP, there is a single encryption transform identifier (which represents the combined transform) (Section 9). The IKE Key Length attribute MUST be used with each use of this identifier to indicate the key length. The Key Length attribute MUST have a value of 128, 192, or 256.

5.4. Keying Material and Salt Values

IKE makes use of a pseudo-random function (PRF) to derive keying material. The PRF is used iteratively to derive keying material of arbitrary size, called KEYMAT. Keying material is extracted from the output string without regard to boundaries.

The size of the KEYMAT for the ENCR_NULL_AUTH_AES_GMAC and AUTH_AES_GMAC MUST be four octets longer than is needed for the associated AES key. The keying material is used as follows:

ENCR_NULL_AUTH_AES_GMAC with a 128-bit key and AUTH_AES_128_GMAC
   The KEYMAT requested for each AES-GMAC key is 20 octets. The first 16 octets are the 128-bit AES key, and the remaining four octets are used as the salt value in the nonce.

ENCR_NULL_AUTH_AES_GMAC with a 192-bit key and AUTH_AES_192_GMAC
   The KEYMAT requested for each AES-GMAC key is 28 octets. The first 24 octets are the 192-bit AES key, and the remaining four octets are used as the salt value in the nonce.

ENCR_NULL_AUTH_AES_GMAC with a 256-bit key and AUTH_AES_256_GMAC
   The KEYMAT requested for each AES-GMAC key is 36 octets. The first 32 octets are the 256-bit AES key, and the remaining four octets are used as the salt value in the nonce.

6. Test Vectors

Appendix B of [GCM] provides test vectors that will assist implementers with AES-GMAC.

7. Security Considerations

Since the authentication coverage is different between AES-GCM-ESP and this specification (see Figure 4), it is worth pointing out that both specifications are secure. In ENCR_NULL_AUTH_AES_GMAC, the IV is not included in either the plaintext or the additional authenticated data. This does not adversely affect security, because the IV field only provides an input to the GMAC IV, which is not required to be authenticated (see [GCM]). In AUTH_AES_GMAC, the IV is included in the additional authenticated data. This fact has no adverse effect on security; it follows from the property that GMAC is secure even against attacks in which the adversary can manipulate both the IV and the message. Even an adversary with these powerful capabilities cannot forge an authentication tag for any message (other than one that was submitted to the chosen-message oracle). Since such an adversary could easily choose messages that contain the IVs with which they correspond, there are no security problems with the inclusion of the IV in the AAD.

GMAC is provably secure against adversaries that can adaptively choose plaintexts, ICVs and the AAD field, under standard cryptographic assumptions (roughly, that the output of the underlying cipher under a randomly chosen key is indistinguishable from a randomly selected output). Essentially, this means that, if used within its intended parameters, a break of GMAC implies a break of the underlying block cipher. The proof of security is available in [GCMP].

The most important security consideration is that the IV never repeats for a given key. In part, this is handled by disallowing the use of AES-GMAC when using statically configured keys, as discussed in Section 2.

When IKE is used to establish fresh keys between two peer entities, separate keys are established for the two traffic flows. If a different mechanism is used to establish fresh keys, one that establishes only a single key to protect packets, then there is a high probability that the peers will select the same IV values for some packets. Thus, to avoid counter block collisions, ESP or AH implementations that permit use of the same key for protecting packets with the same peer MUST ensure that the two peers assign different salt values to the security association (SA).

The other consideration is that, as with any block cipher mode of operation, the security of all data protected under a given security association decreases slightly with each message.

To protect against this problem, implementations MUST generate a fresh key before processing 2^64 blocks of data with a given key. Note that it is impossible to reach this limit when using 32-bit Sequence Numbers.

Note that, for each message, GMAC calls the block cipher only once.

8. Design Rationale

This specification was designed to be as similar to AES-GCM-ESP [RFC4106] as possible. We re-use the design and implementation experience from that specification. We include all three AES key sizes since AES-GCM-ESP supports all of those sizes, and the larger key sizes provide future users with more high-security options.

9. IANA Considerations

IANA has assigned the following IKEv2 parameters. For the use of AES GMAC in AH, the following integrity (type 3) transform identifiers have been assigned:

"9" for AUTH_AES_128_GMAC

"10" for AUTH_AES_192_GMAC

"11" for AUTH_AES_256_GMAC

For the use of AES-GMAC in ESP, the following encryption (type 1) transform identifier has been assigned:

"21" for ENCR_NULL_AUTH_AES_GMAC

10. Acknowledgements

Our discussions with Fabio Maino and David Black significantly improved this specification, and Tero Kivinen provided us with useful comments. Steve Kent provided guidance on ESP interactions. This work is closely modeled after AES-GCM, which itself is closely modeled after Russ Housley’s AES-CCM transform [RFC4309]. Additionally, the GCM mode of operation was originally conceived as an improvement to the CWC mode [CWC] in which Doug Whiting and Yoshi Kohno participated. We express our thanks to Fabio, David, Tero, Steve, Russ, Doug, and Yoshi.

11. References

11.1. Normative References

[GCM] McGrew, D. and J. Viega, "The Galois/Counter Mode of Operation (GCM)", Submission to NIST. http://csrc.nist.gov/CryptoToolkit/modes/proposedmodes/gcm/gcm-spec.pdf, January 2004.

[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.

11.2. Informative References

[CWC] Kohno, T., Viega, J., and D. Whiting, "CWC: A high-performance conventional authenticated encryption mode", Fast Software Encryption. http://eprint.iacr.org/2003/106.pdf, February 2004.

[GCMP] McGrew, D. and J. Viega, "The Security and Performance of the Galois/Counter Mode (GCM)", Proceedings of INDOCRYPT ’04, http://eprint.iacr.org/2004/193, December 2004.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC4106] Viega, J. and D. McGrew, "The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP)", RFC 4106, June 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

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

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[RFC4309] Housley, R., "Using Advanced Encryption Standard (AES) CCM Mode with IPsec Encapsulating Security Payload (ESP)", RFC 4309, December 2005.

Authors’ Addresses

David A. McGrew
Cisco Systems, Inc.
510 McCarthy Blvd.
Milpitas, CA 95035
US

Phone: (408) 525 8651
EMail: mcgrew@cisco.com
URI: http://www.mindspring.com/˜dmcgrew/dam.htm

John Viega
McAfee, Inc.
1145 Herndon Parkway, Suite 500
Herndon, VA 20170

EMail: viega@list.org

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).