How I learned to stop worrying and love AES-GCM

Quentin Gouchet

quentin.gouchet@atsec.com

ICMC 2017, Washington DC, May 18th

Agenda

High-level description of the GCM

AES-GCM operations

AES-GCM weakness

IG A.5, bullet 1)

Worst case assumptions

Probability of a collision of the {key,iv} pairs

IG A.5, bullet 2): random IV

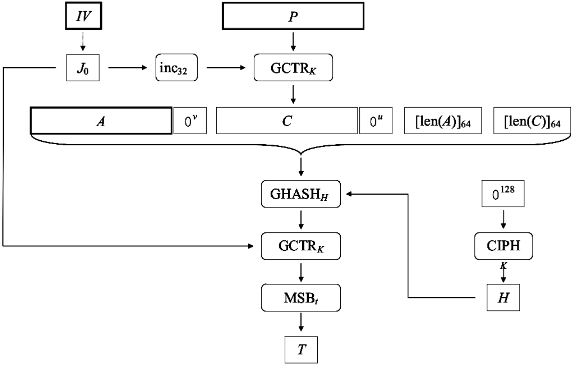
IG A.5, bullet 3): deterministic IV

CMVP letter: IG A.5 interpretation

Conclusion

Encryption and authentication (AEAD)

- AES-GCM uses the AES counter mode GCTR (AES-CTR).
- *CIPH* is a raw AES block encryption operation.
- The GHASH operation generates the tag *T*.
- The default *IV* length is 96 bits.



GHASH

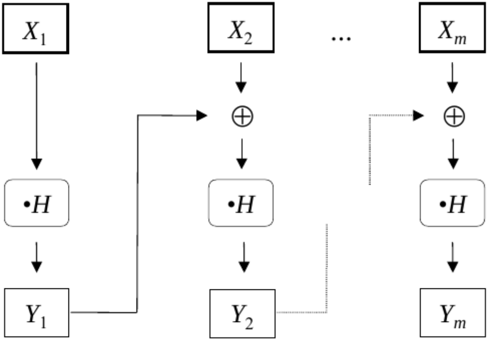


Figure 1: $\mathrm{GHASH}_H(X_1 \,\|\, X_2 \,\|\, \ldots \,\|\, X_m) = Y_m$.

AES-GCM weakness

- AES-CTR has known vulnerabilities.
- AES-GCM is based on AES-CTR.
- $H = CIPH(0^{128}, K)$
- The attacker can compute the GHASH key H.

AES-GCM weakness

- AES-CTR has known vulnerabilities.
- AES-GCM is based on AES-CTR.
- $H = CIPH(0^{128}, K)$
- The attacker can compute the GHASH key H.

$$GHASH(AAD_1 \| 0^{v_1} \| C_1 \| 0^{u_1} \| len(AAD_1) \| len(C_1), H) = G_1$$
$$GHASH(AAD_2 \| 0^{v_2} \| C_2 \| 0^{u_2} \| len(AAD_2) \| len(C_2), H) = G_2$$

AES-GCM weakness

- AES-CTR has known vulnerabilities.
- AES-GCM is based on AES-CTR.
- $H = CIPH(0^{128}, K)$
- The attacker can compute the GHASH key H.

$$GHASH(AAD_1 \| 0^{v_1} \| C_1 \| 0^{u_1} \| len(AAD_1) \| len(C_1), H) = G_1$$
$$GHASH(AAD_2 \| 0^{v_2} \| C_2 \| 0^{u_2} \| len(AAD_2) \| len(C_2), H) = G_2$$

$$T_1 = G_1 + GCTR_{AES}(IV \| 0^{32}, K)$$
$$T_2 = G_2 + GCTR_{AES}(IV \| 0^{32}, K)$$

AES-GCM weakness

- AES-CTR has known vulnerabilities.
- AES-GCM is based on AES-CTR.
- $H = CIPH(0^{128}, K)$
- The attacker can compute the GHASH key H.

$$GHASH(AAD_1 \| 0^{v_1} \| C_1 \| 0^{u_1} \| len(AAD_1) \| len(C_1), H) = G_1$$
$$GHASH(AAD_2 \| 0^{v_2} \| C_2 \| 0^{u_2} \| len(AAD_2) \| len(C_2), H) = G_2$$

$$T_1 = G_1 + GCTR_{AES}(IV \| 0^{32}, K)$$
$$T_2 = G_2 + GCTR_{AES}(IV \| 0^{32}, K)$$

Let P be a polynomial in H defined as:

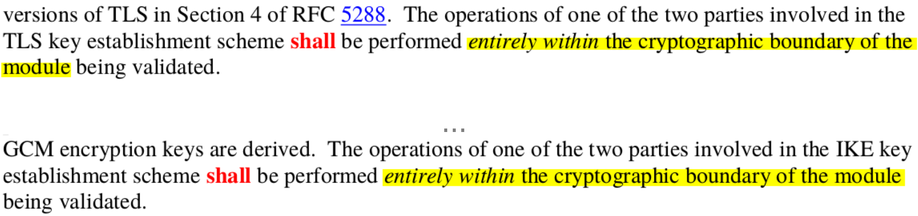
$$P(H) = T_1 + T_2 \tag{1}$$
$$= G_1 + G_2 \tag{2}$$

IG A.5 overview

- Bullet 1): the IV construction is according to the industry protocols IPsec (RFC4106) and TLS (RFC5282)
- Bullet 2): the IV is randomly generated
- Bullet 3): the IV is deterministically generated
- $IV = A||B$

	$A\|\|B$	A	B
IG A.5	IV	Fixed field	Invocation field
RFC5288	Nonce	Salt	IV
RFC4106	Nonce	Salt	IV

IG A.5, bullet 1)



versions of TLS in Section 4 of RFC 5288. The operations of one of the two parties involved in the TLS key establishment scheme **shall** be performed *entirely within* the cryptographic boundary of the module being validated.

…

GCM encryption keys are derived. The operations of one of the two parties involved in the IKE key establishment scheme **shall** be performed *entirely within* the cryptographic boundary of the module being validated.

IPsec basics

- The IKEv2 protocol is used.
- AES keys are uniformly distributed.
- The module is the “sender“ (only AES-GCM encryption is considered).
- There is an up-and-down set of $\{key, IV$, MAC key, etc.$\}$.
- The module handles the 64-bit invocation field of the AES-GCM IV.

Focus on the invocation field

- The first 32 bits can be considered to be the same
- In reality, these 32 bits come from a Diffie-Hellman key exchange or a pre-shared key

$$IV = 0^{32} || < \text{invocation field} >$$

Focus on the invocation field

- The first 32 bits can be considered to be the same
- In reality, these 32 bits come from a Diffie-Hellman key exchange or a pre-shared key

$$IV = 0^{32} || < \text{invocation field} >$$

(Ridiculous) assumptions for the worse case scenario

- Server running for y years
- 10 GB/s network
- SA key lifetime: 10s
- IPsec protocol (RFC4106)
- AES-GCM algorithm
- 10 million modules

Number of packets

- $\frac{2^{30}}{20 \cdot 8}$ packets per second (think about the smallest TCP/IP packet size in bits)
- The invocation field is a random number.
- It is deterministically incremented (field += 1, LFSR with primitive retro-action polynomial, etc.)

Number of packets

- $\frac{2^{30}}{20 \cdot 8}$ packets per second (think about the smallest TCP/IP packet size in bits)
- The invocation field is a random number.
- It is deterministically incremented (field += 1, LFSR with primitive retro-action polynomial, etc.)
- $\frac{2^{64} \cdot 20 \cdot 8}{2^{30}} = 2748779069440$ seconds $> 87,163$ years

Number of packets

- $\frac{2^{30}}{20 \cdot 8}$ packets per second (think about the smallest TCP/IP packet size in bits)
- The invocation field is a random number.
- It is deterministically incremented (field += 1, LFSR with primitive retro-action polynomial, etc.)
- $\frac{2^{64} \cdot 20 \cdot 8}{2^{30}} = 2748779069440$ seconds $> 87,163$ years
- The invocation field will not wrap.

Probability of repeating keys

- constant invocation field + constant fixed field = constant *IV*
- y years $\Rightarrow y \cdot 365 \cdot 24 \cdot 60 \cdot 6 \cdot 10^7 = 31536 \cdot 10^9 \cdot y$ AES keys
- $A = \{\text{AES key will repeat}\}$
- $\overline{A} = \{\text{AES keys will not repeat}\}$
- $\mathcal{P}(A) = 1 - \mathcal{P}(\overline{A})$

Probability of repeating keys

- constant invocation field + constant fixed field = constant IV
- y years $\Rightarrow y \cdot 365 \cdot 24 \cdot 60 \cdot 6 \cdot 10^7 = 31536 \cdot 10^9 \cdot y$ AES keys
- $A = \{$AES key will repeat$\}$
- $\overline{A} = \{$AES keys will not repeat$\}$
- $\mathcal{P}(A) = 1 - \mathcal{P}(\overline{A})$
- Let $F(y) = log_2(y) + log_2(31536) + 9 \cdot log_2(10)$.
- $\mathcal{P}(\overline{A}) = \prod_{i=0}^{2^{F(y)}} (1 - \frac{i}{2^{128}})$

Probability of repeating keys

- constant invocation field + constant fixed field = constant IV
- y years $\Rightarrow y \cdot 365 \cdot 24 \cdot 60 \cdot 6 \cdot 10^7 = 31536 \cdot 10^9 \cdot y$ AES keys
- $A = \{\text{AES key will repeat}\}$
- $\overline{A} = \{\text{AES keys will not repeat}\}$
- $\mathcal{P}(A) = 1 - \mathcal{P}(\overline{A})$
- Let $F(y) = log_2(y) + log_2(31536) + 9 \cdot log_2(10)$.
- $\mathcal{P}(\overline{A}) = \prod\limits_{i=0}^{2^{F(y)}} (1 - \frac{i}{2^{128}})$

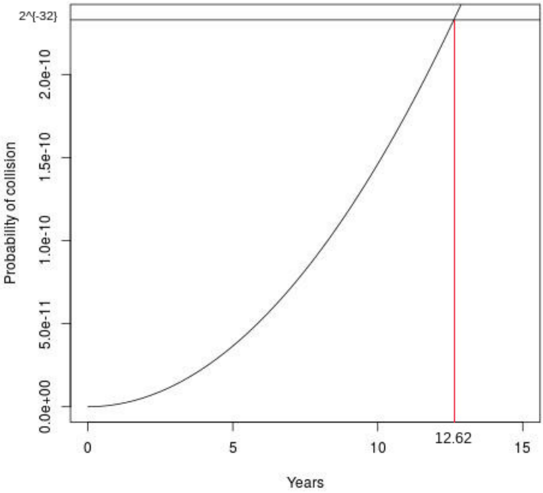
$$e^x \approx 1 + x \Rightarrow \mathcal{P}(\overline{A}) \approx e^{-\frac{1}{2^{128}} \sum\limits_{i=0}^{2^{F(y)}} i} \tag{3}$$

$$= e^{\frac{2^{F(y)}(1-2^{F(y)})}{2^{129}}} \tag{4}$$

$$\approx e^{-\frac{2^{2 \cdot F(y)}}{2^{129}}} \tag{5}$$

Probability of repeating keys

$$\begin{aligned}
\mathcal{P}(A) &= 1 - \mathcal{P}(\overline{A}) && (6)\\
&\approx 1 - e^{-\frac{2^{2\cdot F(y)}}{2^{129}}} && (7)\\
&= 1 - e^{-2^{2\cdot F(y)-129}} && (8)\\
&\Rightarrow \text{compliant with SP 800-38D if } < 2^{-32} && (9)\\
&\Rightarrow \text{compliant with SP 800-38D if } y < 12.62 && (10)
\end{aligned}$$


2^{-32}
2.0e-10
1.5e-10
1.0e-10
5.0e-11
0.0e+00
Probability of collision
0
5
10
12.62
15
Years

TLS

- 32 bits of the IV are also derived from the key exchanged and a PRF
- 64 bits set by the protocol (packet number or session ID)
- The previous calculations still apply.

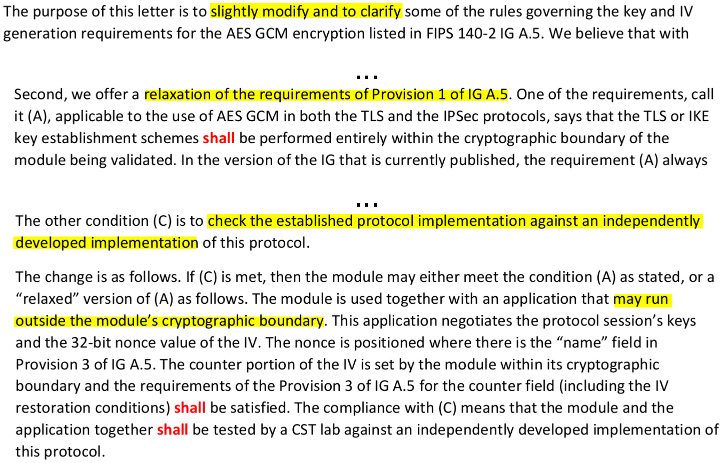
IG A.5, bullet 2): random *IV*

$$\mathcal{P}(\overline{A}) = \prod_{i=0}^{2^{F(y)}} (1 - \frac{i}{2^{128+96}}) \tag{11}$$

IG A.5, bullet 3): deterministic *IV*

- Bullets 1) and 3) cases in IG A.5 are not disjoint.
- Bullet 1) is a special case of bullet 3).
- Bullet 3) allows the first 32 bits to be externally generated.
- Bullet 1) is more restrictive that bullet 3).

How to Interpret Various Provisions in IG A.5



The purpose of this letter is to slightly modify and to clarify some of the rules governing the key and IV generation requirements for the AES GCM encryption listed in FIPS 140-2 IG A.5. We believe that with

...

Second, we offer a relaxation of the requirements of Provision 1 of IG A.5. One of the requirements, call it (A), applicable to the use of AES GCM in both the TLS and the IPSec protocols, says that the TLS or IKE key establishment schemes **shall** be performed entirely within the cryptographic boundary of the module being validated. In the version of the IG that is currently published, the requirement (A) always

...

The other condition (C) is to check the established protocol implementation against an independently developed implementation of this protocol.

The change is as follows. If (C) is met, then the module may either meet the condition (A) as stated, or a "relaxed" version of (A) as follows. The module is used together with an application that may run outside the module's cryptographic boundary. This application negotiates the protocol session's keys and the 32-bit nonce value of the IV. The nonce is positioned where there is the "name" field in Provision 3 of IG A.5. The counter portion of the IV is set by the module within its cryptographic boundary and the requirements of the Provision 3 of IG A.5 for the counter field (including the IV restoration conditions) **shall** be satisfied. The compliance with (C) means that the module and the application together **shall** be tested by a CST lab against an independently developed implementation of this protocol.

IG A.5

- Thank you to CMVP !!!
- IG A.5 will be updated soon

ANY
QUESTIONS
?