Collision Attacks on Galois/Counter Mode (GCM)

John Preuß Mattsson

Ericsson Research, Sweden

Abstract. Advanced Encryption Standard in Galois/Counter Mode (AES-GCM) is the most widely used Authenticated Encryption with Associated Data (AEAD) algorithm in the world. In this paper, we analyze the use of GCM with all the Initialization Vector (IV) constructions and lengths approved by NIST SP 800-38D when encrypting multiple plaintexts with the same key. We derive attack complexities in both ciphertext-only and known-plaintext models, with or without nonce hiding, for collision attacks compromising integrity and confidentiality. Our analysis shows that GCM with random IVs provides less than 128 bits of security. When 96-bit IVs are used, as recommended by NIST, the security drops to less than 97 bits. Therefore, we strongly recommend NIST to forbid the use of GCM with 96-bit random nonces.

Keywords: Secret-key Cryptography · Block Ciphers · Cryptanalysis · Collision Attacks · Near Collisions · Nonce Hiding · AEAD · MAC · GCM · GMAC

1 Introduction

Galois/Counter Mode (GCM) is an Authenticated Encryption with Associated Data (AEAD) mode of operation, designed by McGrew and Viega [MV05] and standardized by the National Institute of Standards and Technology (NIST) in SP 800-38D [Dwo07]. GCM combines counter mode of encryption with Galois mode of authentication, which is a Wegman-Carter polynomial hash operating in the field $\mathrm{GF}(2^{128})$. Originally designed for block ciphers with a 128-bit block size, such as the Advanced Encryption Standard (AES) [AES23], but as shown in [CMP23] it can also be adapted for use with any stream cipher like SNOW 5G [EJMY21] or Rijndael-256-256 [DR03] in counter mode.

AES-GCM is the most widely used AEAD algorithm in the world, used in numerous security protocols, including TLS [Res18], QUIC [TT21], IPsec [VM05], MACsec [MAC18], and WiFi WPA3 [WPA24]. It is also supported by many cryptographic Application Programming Interfaces (APIs) such as PKCS #11 [PKC20], Oracle Java SE [JAV24], Microsoft Cryptography API [CNG21], W3C Web Cryptography API [W3C17], the Linux Kernel Crypto API [Lin], and Apple CryptoKit [App]. Its popularity is well-deserved due to its strong performance and proven security [MV04, IOM12]. GCM is online, fully parallelizable, and can be efficiently pipelined, making it highly effective in both hardware and software, especially on processors with dedicated instructions to accelerate AES and GHASH [Gue23].

Weaknesses in GCM have been discussed by several researchers, including Ferguson [Fer05], Joux [Ant06], Handschuh and Preneel [HP08], Iwata et al. [IOM12], Saarinen [Saa11], Procter and Cid [PC15], Mattsson and Westerlund [MW15], Abdelraheem et al. [ABBT15], Forler et al. [FLLW17], and Luykx and Preneel [LP18]. An extensive evaluation of GCM was conducted by Rogaway [Rog11]. It is well-known that reusing a counter value, known as a two-time pad, compromises confidentiality. Furthermore, Joux demonstrated that reusing a single Initialization Vector (IV) in GCM also breaks

E-mail: john.mattsson@ericsson.com (John Preuß Mattsson)

integrity [Ant06]. NIST has decided to revise NIST SP 800-38D [Ann24]. The proposed changes include removing support for authentication tags shorter than 96 bits, as suggested by [Rog11, MW15], and providing clearer guidance on IV constructions such as clarifying that the IV construction used in TLS 1.3 [Res18] is approved.

In this paper, we analyze the use of GCM with all the IV constructions and lengths approved in NIST SP 800-38D [Dwo07] when encrypting multiple plaintexts with the same key. We derive attack complexities in both ciphertext-only and known-plaintext models, considering different nonce hiding transforms [BNT19], for collision attacks compromising integrity and confidentiality. The confidentiality attacks enable the attacker to find a large number of colliding keystream blocks and are therefore significantly more severe than distinguishing attacks [IOM12]. Our analysis shows that GCM with random IVs is severely limited by the narrow 128-bit "block size" and therefore provides less than 128 bits of security. When 96-bit IVs are used, as recommended by NIST, the security drops to less than 97 bits. Users of AES-GCM expect 128, 192, or 256 bits of security. Therefore, we strongly recommend that NIST revise SP 800-38D to forbid the use of GCM with 96-bit random nonces. Nonce hiding [BNT19] requires collision attacks to be performed in a known-plaintext context rather than a ciphertext-only context. However, our analysis indicates that except for one examined IV construction combined with one examined nonce hiding transform, nonce hiding does not alter the attack complexity. To facilitate our analysis, we derive a new, simplified equation for near birthday collisions. The integrity attacks on GCM also apply to Galois Message Authentication Code (GMAC) [Dwo07]. Furthermore, many of the attacks are generic and affect other AEAD algorithms, such as Counter with Cipher Block Chaining-Message Authentication Code (CCM) [Dwo04] and ChaCha20-Poly1305 [NL18], when they are used with random nonces.

2 Collision Attacks on Galois Counter Mode (GCM)

In this section, we analyze GCM as specified in NIST SP 800-38D [Dwo07]. For simplicity, we assume the block cipher is AES [AES23], the only NIST-approved block cipher. Given an AES algorithm and key K, the authenticated encryption function takes three input strings: plaintext P, additional authenticated data A, and initialization vector IV. The output consists of ciphertext C and authentication tag T.

The AES key length can be 128, 192, or 256 bits, while the block size is always 128 bits, regardless of key size. The plaintext must be shorter than $2^{32} - 2$ 16-byte blocks. The IV length must be between 1 and $2^{61} - 1$ bytes, though NIST recommends that implementations restrict support to 96-bit IVs. NIST SP 800-38D specifies two IV constructions: one deterministic and one based on a Random Bit Generator (RBG). For IVs shorter than 96 bits, the deterministic construction must be used, while for IVs equal to or longer than 96 bits, either construction is permissible:

- In the deterministic construction, the IV is the concatenation of two fields: the fixed field identifying the device and the invocation field. For any given key, no two distinct devices shall share the same fixed field, and no two distinct sets of inputs to any single device shall share the same invocation field. Typically, the invocation field is an integer counter.

- In the RBG-based construction, the IV is the concatenation of two fields: the random field, which must be at least 96 bits long, and the free field, which has no specific requirements. For our analysis, we assume the free field is empty, meaning the length of the random field equals $|IV|$, the length of the initialization vector in bits. The random field must either 1) consist of the output from an approved RBG, or 2) be obtained by incrementing the random field of the previous IV modulo $2^{|IV|}$. The output string from the RBG is called a direct random string, and the random fields that result from applying the incrementing function are called its successors. We will refer to the two different options as the direct RBG-based construction and the successor RBG-based construction.

The deterministic construction guarantees that there are no IV collisions, while the RBG-based construction limits the use of state between invocations of GCM. The use of only direct random strings eliminates state within each device, while using one direct random string and its successors per device eliminates the need to sync fixed fields between devices. Unless an implementation exclusively uses 96-bit IVs generated by the deterministic construction, the number of invocations of the authenticated encryption function must not exceed 2^{32} for a given key.

The GCM authenticated encryption function is detailed in Section 7.1 of NIST SP 800-38D [Dwo07]. The steps relevant to our analysis are:

$H = \text{AES-ENC}(\ K,\ 0^{128}\ )$

If $|IV| = 96$, then $J_0 = IV \mid\mid 0^{31} \mid\mid 1$

If $|IV| \neq 96$, then $J_0 = \text{GHASH}(\ H,\ IV \mid\mid \ldots)$

$J_0 = F \mid\mid I$ where F is the leftmost 96 bits, and I is the rightmost 32 bits

$J_1 = F \mid\mid (I + 1) \mod 2^{32}$

$C_i = \text{AES-ENC}(\ K,\ J_{i+1}\ ) \oplus P_i$

$T = \text{AES-ENC}(\ K,\ J_0\ ) \oplus \ldots$

where "…" indicates data not relevant to our analysis. The steps assume a tag length of 128 bits and a plaintext length that is a multiple of 16 bytes. P_i and C_i denote the i-th block in the plaintext and ciphertext, respectively.

We analyze the security of AES-GCM across all approved IV constructions and lengths specified in NIST SP 800-38D [Dwo07] when encrypting multiple plaintexts with the same key K. Specifically, we derive concrete complexities of collision attacks finding collisions between initialization vectors IV or between counter values J in different AES-GCM invocations under the same key. Our attacks do not assume any flaws in the random bit generator or GHASH, remaining effective even if their behavior is indistinguishable from a truly random function. IV collision attacks on GCM were briefly mentioned in [PST23, PST24], but only in the context of ciphertext-only attacks involving cleartext 96-bit IVs composed of direct random strings.

It is evident that no collisions occur between counter values within a single invocation. In the following, we use the notation IV_k for the initialization vector in invocation k and J_{ik} for the counter value J_i in invocation k. A collision where $IV_k = IV_l$ implies $J_{0k} = J_{0l}$. A collision $J_{0k} = J_{0l}$ (where $k \neq l$) compromises both integrity and confidentiality. A collision $J_{ik} = J_{jl}$ (where $k \neq l$ and i and j are not both being 0) compromises confidentiality but not integrity.

2.1 Probabilities for Collisions and Near Collisions

Collisions. The probability of a collision among m uniformly distributed random integers between 0 and $N - 1$ is approximately given by

$$\approx \frac{(m-1)m}{2N}\ , \tag{1}$$

where the approximation is valid when $m^2 \ll N$. For $m \gg 1$, this simplifies to

$$\approx \frac{m^2}{2N}\ . \tag{2}$$

This is a well-known result from the birthday problem.

Near Collisions. A generalization of the birthday problem considers the probability of near collisions [AM70], specifically, the probability that at least two values are within a distance d of each other. Using the approximation $e^x \approx 1+x$ on equation (17) from [Tau09], the probability is approximately

$$\approx \frac{(2d+1)(m-1)m}{2N} \ , \tag{3}$$

where the approximation is valid when $dm^2 \ll N$. For $d \gg 1$ and $m \gg 1$, this simplifies to

$$\approx \frac{dm^2}{N} \ . \tag{4}$$

The approximations in equations (3) and (4) are also valid when the distance is calculated modulo N, i.e., the distance between a and b is $\min(|a-b|, N-|a-b|)$ instead of $|a-b|$.

2.2 Ciphertext-Only Collision Attacks ($\mathbf{IV_k = IV_l}$)

Deterministic Construction. When the deterministic construction is used, collisions between IVs do not occur. That is, $IV_k \neq IV_l$ when $k \neq l$, and $J_{ik} \neq J_{jl}$ when $i \neq j$ or $k \neq l$. Hence, ciphertext-only collision attacks are not feasible.

Direct RBG-Based Construction. Assuming the free field is empty, when using $n \gg 1$ direct random cleartext IVs with no successors under the same key, the probability of an IV collision, given by equation (2), is $\approx n^2/2^{|IV|+1}$. An IV collision $IV_k = IV_l$ where $k \neq l$ implies $J_{0k} = J_{0l}$, compromising both confidentiality and integrity. An attacker can detect collisions among n cleartext IVs with work $\approx n$ using a hash table. Thus, the time complexity of a collision attack is $\approx n/(n^2/2^{|IV|+1}) = 2^{|IV|+1}/n$, and the security is only $\approx |IV| + 1 - \log_2 n$. The memory and data complexities are $\mathcal{O}(n)$.

Successor RBG-Based Construction. Assume that $m \gg 1$ direct random string are used, each followed by $d \gg 1$ successors obtained by incrementing the random field of the previous IV modulo $2^{|IV|}$. The total number of IVs is $n \approx dm$. The probability that two IVs collide is given by the near-collision probability equation (4) and is $\approx dm^2/2^{|IV|}$. An attacker can, with high probability, detect collisions by hashing prefixes of the m direct random strings. The length s of the prefixes in blocks should be chosen so that $m^2 \ll 2^s \ll 2^{|IV|}/d$ and the work required is $\approx m$. If two prefixes collide, the attacker can check if any of the IVs collide with work $\mathcal{O}(1)$. Thus, the time complexity of a collision attack is $\approx m/(dm^2/2^{|IV|}) \approx 2^{|IV|}/n$, and the security is $\approx |IV| - \log_2 n$. The memory and data complexities are $\mathcal{O}(m)$.

2.3 Known-Plaintext Attacks ($\mathbf{J_{0k} = J_{0l}}$) when $\mathbf{|IV| \neq 96}$ bits

Since each IV_k is hashed to produce a 128-bit value J_{0k}, there might be collisions $J_{0k} = J_{0l}$ where $k \neq l$ even if $IV_k \neq IV_l$. Such collisions compromise both confidentiality and integrity.

Deterministic Construction. Since there are no collisions between IVs, the probability that $J_{0k} = J_{0l}$ for $k \neq l$ is $\approx n^2/2^{129}$. An attacker, assuming they have access to known plaintext, can find such a collision with work $\approx n$. If the attacker knows 16 bytes P_{ik} of each plaintext, they can identify collisions by hashing $P_{ik} \oplus C_{ik}$ from all invocations $k = 0 \ldots n-1$. If the first 16 bytes of the plaintexts P_{0k} contain a fixed header (refer to Section 3.4 of [Pre23]), the attacker can find a collision by hashing each C_{0k}. Consequently, the attack complexity is $\approx 2^{129}/n$, and the security against this attack is $\approx 129 - \log_2 n$.

Direct RBG-Based Construction. The probability that $J_{0k} = J_{0l}$ where $k \neq l$ is $\approx n^2/2^{|IV|+1} + n^2/2^{129}$. An attacker can find such a collision with work $\approx n$, assuming a fixed plaintext header. The attack complexity is therefore $\approx n/(n^2/2^{|IV|+1} + n^2/2^{129}) = (1/2^{|IV|+1} + 1/2^{129})^{-1}/n$. If $|IV| = 128$, the attack complexity is $\approx 2^{128}/n$. For $|IV| > 128$, the attack complexity is $\approx 2^{129}/n$. When $|IV| < 128$, the ciphertext-only attack described in Section 2.2 has lower complexity.

Successor RBG-Based Construction. The probability that $J_{0k} = J_{0l}$ where $k \neq l$ is $\approx dm^2/2^{|IV|} + n^2/2^{129}$. An attacker can find such a collision with work $\approx n$, assuming a fixed plaintext header. The attack complexity is therefore $\approx n/(dm^2/2^{|IV|} + n^2/2^{129}) = (1/2^{|IV|} + d/2^{129})^{-1}/m$. If $|IV| > 128$, the attack complexity is $\approx 2^{129}/n$. When $|IV| \leq 128$, the ciphertext-only attack described in Section 2.2 has lower complexity.

2.4 Known-Plaintext Attacks ($\mathbf{J_{ik} = J_{jl}}$) when $|\mathbf{IV}| \neq 96$ bits

A collision $J_{ik} = J_{jl}$ where $k \neq l$ and i and j are not both 0 does not break integrity but does compromise confidentiality.

Deterministic Construction. For plaintexts of length $\ell \geq 2^{31}$ blocks or larger, the probability of at least two different counter values colliding is $\approx n^2/2^{97}$. If $J_{ik} = J_{jl}$, it is likely that $J_{(i+1)k} = J_{(j+1)l}$, where the addition is modulo 2^{32}. This results in the keystreams $P \oplus C$ in invocations k and l being partially identical. The work for an attacker to find such a collision is $\approx n \cdot 2^{31}$, as they only need to test the first $\approx 2^{31}$ blocks. The attack complexity is $\approx n \cdot 2^{31}/(n^2/2^{97}) = 2^{128}/n$. An assumption in this scenario can be that the plaintexts consist of approximately 2^{32} blocks, with the attacker knowing the first half but not the second half. For plaintexts of length $\ell < 2^{31}$ blocks, the probability of at least two different counter values colliding is $\approx (n^2/2^{97})(2\ell/2^{32}) = \ell n^2/2^{128}$. The work required is $\approx \ell n$, and the attack complexity is $\approx \ell n/(\ell n^2/2^{128}) = 2^{128}/n$. An assumption in this scenario can be that the attacker knows most, but not all, of each plaintext.

RBG-Based Constructions. The analysis is the same as for the deterministic construction. However, for $|IV| < 128$, the ciphertext-only attacks described in Section 2.2 has lower complexity.

2.5 Nonce Hiding Transforms

Cleartext nonces can compromise privacy by enabling tracking and identification of both the sender and receiver. They can also reveal information to an attacker who has compromised the pseudorandom number generator. As discussed in Section 2.2, cleartext nonces can be exploited for ciphertext-only collision attacks, compromising both integrity and confidentiality. Bellare et al. [BNT19] provide a theoretical treatment of nonce-hiding AEADs and propose several nonce-hiding transformations.

HN1 Transform. In the HN1 (Hiding Nonce One) transform [BNT19], employed in DTLS 1.3 [RTM22] and QUIC [TT21], the encrypted initialization vector transmitted over the network is $IV \oplus$ AES-ENC(K_2, C_0), where K and K_2 can be derived from the same secret. Assuming a fixed plaintext header, the attacker can detect IV collisions by hashing the encrypted IVs. For the successor RBG-based construction, collisions can still be detected by hashing prefixes of the m direct random strings, see Section 2.2. The work required is $\approx m$, and the prefix collision probability is $\approx m^2/2^{s+1}$, where $m^2 \ll 2^s$. If two prefixes collide, the attacker can then check whether any of the $\approx 2d$ IVs associated with these prefixes also collide by hashing them. The average work remains $\approx m$. Therefore, the HN1 transform does not alter the attack complexities, but it requires the attack to be conducted in a known-plaintext model instead of a ciphertext-only model.

HN2 Transform. In the HN2 transform [BNT19], the encrypted initialization vector transmitted over the network is AES-ENC(K_2, $IV \mid\mid x$), where x is a prefix of C_0. When the HN2 transform is employed, the most effective collision attack appears to involve hashing all the encrypted IVs, which requires work $\approx n$. For the successor RBG-based construction this increases the attack complexity to $\approx n/(dm^2/2^{|IV|}) \approx 2^{|IV|}/m$. Depending on the parameters, the complexity $\approx 2^{|IV|}/m$ may be lower or higher than the complexities of other attacks in the known-plaintext model described in Sections 2.3 and 2.4.

HN3 Transform. In the HN3 transform [BNT19], the initialization vector used in GCM and transmitted over the network is PRF(K_2, IV), where PRF is a pseudorandom function family. This effectively converts the deterministic and successor RBG-based constructions into the direct RBG-based construction. Consequently, the attack complexity aligns with that of the direct RBG-based construction described in Sections 2.3 and 2.4.

2.6 Summary

The security of GCM against collision attacks is summarized in Tables 1, 2, and 3. Table 1 summarizes the complexity of collision attacks in the ciphertext-only model that compromise integrity and confidentiality. Table 2 shows the complexity of collision attacks in the known-plaintext model that compromise integrity and confidentiality. Finally, Table 3 details the complexity of collision attacks in the known-plaintext model that compromise confidentiality. For certain parameters, the attacks in Table 3 are slightly more effective than the attacks in Table 2 for an attacker focused solely on compromising confidentiality.

Table 1: Complexity of ciphertext-only collision attacks ($IV_k = IV_l$) breaking integrity and confidentiality. $1 \ll n \leq 2^{32}$ is the number of cleartext IVs.

	$\lvert IV\rvert < 96$	$\lvert IV\rvert \geq 96$
Deterministic	∞	∞
RBG Direct	N/A	$2^{\lvert IV\rvert+1}/n$
RBG Successor	N/A	$2^{\lvert IV\rvert}/n$

Table 2: Complexity against known-plaintext collision attacks ($J_{0k} = J_{0l}$) breaking integrity and confidentiality. $1 \ll n \leq 2^{32}$ is the number of IVs. $m \leq n$ is the number of direct random strings.

	$\lvert IV\rvert < 96$	$\lvert IV\rvert = 96$	$96 > \lvert IV\rvert < 128$	$\lvert IV\rvert = 128$	$\lvert IV\rvert > 128$
Deterministic	$2^{129}/n$	∞	$2^{129}/n$	$2^{129}/n$	$2^{129}/n$
RBG Direct	N/A	$2^{97}/n$	$2^{\lvert IV\rvert+1}/n$	$2^{128}/n$	$2^{129}/n$
RBG Successor	N/A	$2^{96}/n$	$2^{\lvert IV\rvert}/n$	$2^{128}/n$	$2^{129}/n$
RBG Successor HN2	N/A	$2^{96}/m$	$\min(\ 2^{\lvert IV\rvert}/m,\ 2^{129}/n\ )$		

Table 3: Complexity against known-plaintext collision attacks ($J_{ik} = J_{jl}$) breaking confidentiality. $1 \ll n \leq 2^{32}$ is the number of IVs. $m \leq n$ is the number of direct random strings.

	$\lvert IV\rvert < 96$	$\lvert IV\rvert = 96$	$96 < \lvert IV\rvert < 128$	$\lvert IV\rvert \geq 128$
Deterministic	$2^{128}/n$	∞	$2^{128}/n$	$2^{128}/n$
RBG Direct	N/A	$2^{97}/n$	$2^{\lvert IV\rvert+1}/n$	$2^{128}/n$
RBG Successor	N/A	$2^{96}/n$	$2^{\lvert IV\rvert}/n$	$2^{128}/n$
RBG Successor HN2	N/A	$2^{96}/m$	$\min\left(2^{\lvert IV\rvert}/m,\ 2^{128}/n \right)$	

3 Analysis of Algorithm and Protocol Specifications

Section 8 of NIST SP 800-38D [Dwo07] states the following regarding IV "uniqueness":

> *"The probability that the authenticated encryption function ever will be invoked with the same IV and the same key on two (or more) distinct sets of input data shall be no greater than 2^{-32}."*

> *"The total number of invocations of the authenticated encryption function shall not exceed 2^{32}, including all IV lengths and all instances of the authenticated encryption function with the given key."*

NIST does not provide a motivation for the probability limit. Expressing requirement as probabilities has several issues. First, it assumes that users know birthday probability formulas (2) and (4) and can calculate that with e.g., direct random strings, a probability of 2^{-32} corresponds to $\approx 2^{(|IV|-31)/2}$ invocations. Additionally, achieving a probability of 2^{-32} is actually impossible with 2^{32} truly random 96-bit IVs. Our analysis shows that with a collision probability of 2^{-33}, a ciphertext-only attack compromising both integrity and confidentiality requires only complexity 2^{64}. Moreover, probability is not directly related to attack complexity; it only establishes a lower bound on security. This makes it unclear what security level NIST intended the requirement to provide. Furthermore, as Rogaway states Section 12.4.10 of [Rog11]:

> *"the exposition in the NIST spec seems to kind of "fall apart" in Sections 8 and 9, and in Appendix C. These sections stray from the goal of defining GCM, and make multiple incorrect or inscrutable statements. Here are some examples. Page 18 :* ***The probability that the authenticated encryption function ever will be invoked with the same IV and the same key on two (or more) sets of input data shall be no greater than $\mathbf{2^{-32}}$*** *(here and later in this paragraph, imperatives are preserved in their original bold font). The probabilistic demand excludes use of almost all cryptographic PRGs (including those standardized by NIST), where no such guarantee is known."*

Theoretically, using a cryptographic pseudorandom generator (PRG) for generating a large number of non-colliding IVs is the wrong approach. Instead, a pseudorandom function family (PRF) should be utilized. While a PRG ensures that a single output appears random, a PRF guarantees that all outputs appear random. The Double-Nonce-Derive-Key-GCM (DNDK-GCM) construction [Gue24] effectively uses a PRF.

3.1 Protocols and Other Algorithms

Many IETF protocols use the NIST-standardized version of GCM [Dwo07] with a deterministic construction and an IV length of 96 bits and do therefore not suffer from collision attacks. The exceptions are JOSE [Jon15] and COSE [Sch22], which may use random IVs, IPsec [VM05], which uses the pre-standardized version of GCM [MV05], and CMS [Hou07], which may use all IVs constructions and lengths allowed by NIST.

The collision attacks on GCM compromising integrity also apply to GMAC, which is also standardized in NIST SP 800-38D [Dwo07]. The ciphertext-only collision attacks compromising confidentiality listed in Table 1 also apply to CCM [32] and ChaCha20-Poly1305 [33] if used with random nonces. CCM with random nonces would be particularly problematic as it can be used with 7–13 byte nonces. In SP 800-38C NIST states that "The nonce is not required to be random", suggesting that AES-CCM with random nonces is NIST-approved. Unlike for GCM, NIST does not mandate any specific nonce constructions, maximum collision probabilities, or maximum number of invocations. The security of AES-CCM with random nonces would be $\approx |IV| + 1 - \log_2 n$ where $|IV|$ can be as low as 56 and n can be as large as $\approx 2^{59}$. SP 800-38C only restricts the number of block cipher invocations:

> *"The total number of invocations of the block cipher algorithm during the lifetime of the key shall be limited to* 2^{61}*."*

As stated in Section 11.9 of [Rog11], Rogaway and Fergusson suggest that "The nonce is not required to be random" should be interpreted as the nonce need not be unpredictable. It is likely this was NIST's intention. However, we do not believe this is how the statement will be understood by developers and users.

4 Conclusions and Recommendations

Without counter value collisions $J_{0k} = J_{0l}$ where $k \neq l$, the security against forgeries in GCM and GMAC is $\approx 2^{129}/\ell$ where ℓ is the plaintext length in blocks. For short plaintexts, the forgery probability is $\approx 2^{128}$, and for maximum length plaintexts the forgery probability is $\approx 2^{97}$. As shown in Table 1 and 2, the RBG-based construction significantly lowers security against forgeries. The attack model is practically serious, as it can be executed by passively observing communications, performing calculations offline, and if successful, allowing any number of forgeries with a success probability of 1.

Without counter value collisions $J_{ik} = J_{jl}$ where $k \neq l$, the best attacks on AES-GCM confidentiality are distinguishing attacks based on the birthday bound. With counter value collisions, collision attacks finding colliding parts of keystream (two-time pad) becomes possible. As shown in Table 1 and 3, the security of the RBG-based construction significantly lowers security even when the number of IVs, n, is small.

We strongly recommend that NIST disallow the use of the RGB-based construction when $|IV| < 128$, as it significantly lowers the security against forgeries for all plaintext lengths. Additionally, NIST should consider disallowing the RGB-construction when $|IV| \geq 128$ as it significantly lowers the security against forgeries for short plaintext lengths. We also advise NIST to disallow the use of the deterministic construction when $|IV| \neq 96$, as it lowers security and there is no reason to ever use it. NIST should ensure that all remaining options achieve security strength of 128 bits [KEY20] and clearly describe the security strength category [KEM23] of each option.

If NIST intends to continue allowing the RBG-based construction, given the potential use cases for AES-GCM with random IVs, we recommend that NIST mandate that the random field is at least 17 bytes and clearly state the security level against collision attacks. While GCM with non-96-bit IVs has other theoretical weaknesses [ABBT15], to our knowledge, none are remotely comparable to ciphertext-only attacks that break integrity and confidentiality with complexity $2^{96}/n$.

If the RBG-based construction is kept, NIST should replace the probability-based IV requirement with an explicit requirement that is easy to understand for developers and users. This requirement should clearly specify the number of authenticated encryption invocations with the same key for different lengths of the random field. NIST should also give examples of PRGs or PRFs that can be used for generating a large number of non-colliding IVs. A better solution than using the RBG-based construction is likely deriving a new key K for each random nonce as suggested in DNDK-GCM [Gue24].

We strongly recommend that NIST and IETF explicitly disallow the use of the random nonces in AES-CCM and ChaCha20-Poly1305. Additionally, we suggest that NIST update the terminology in SP 800-38D to use "nonce" instead of "IV", as "nonce" is now the established term for the AEAD input parameter [McG08], while "IV" commonly refers to one of the fields used to construct the nonce [Res18, VM05]. Updating SP 800-38D to use the term "nonce" will align it with SP 800-38C. We recommend IETF to update the use of GCM in IPsec [VM05] to refer to the standardized version of GCM [Dwo07].

The security of AES-GCM is severely limited by the narrow 128-bit block size in AES and 128-bit digest size GHASH. Future AEAD schemes should use 256-bit keys and 256-bit nonces. Shorter nonces could be acceptable for misuse-resistant AEs (MRAE) [RS06] as nonce collisions only lowers the security to DAE (deterministic authenticated encryption). Robust AE (RAE) [HKR14] are especially attractive as they combine misuse-resistance with reforgeability resilience. However, as interfaces should be designed to minimize user demands and mitigate the consequences of human errors [Gui16], users ideally should not have to handle nonces. Consequently, we believe that future standardized authenticated encryption interfaces should not require nonces as input. One such interface is AERO [MF14, Min15], which not only manages nonces but also provides replay protection and nonce hiding.

References

[ABBT15] Mohamed Ahmed Abdelraheem, Peter Beelen, Andrey Bogdanov, and Elmar Tischhauser. Twisted Polynomials and Forgery Attacks on GCM. Cryptology ePrint Archive, Paper 2015/1224, 2015. https://eprint.iacr.org/2015/1224.

[AES23] Advanced Encryption Standard (AES). National Institute of Standards and Technology, NIST FIPS PUB 197, May 2023. doi:10.6028/NIST.FIPS.197-upd1.

[AM70] Morton Abramson and WOJ Moser. More birthday surprises. *The American Mathematical Monthly*, 77(8):856–858, 1970. https://www.vartang.com/wp-content/uploads/2013/03/More-Birthday-Surprises.pdf.

[Ann24] NIST to Revise Special Publication 800-38D | Galois/Counter Mode (GCM) and GMAC Block Cipher Modes. National Institute of Standards and Technology, March 2024. https://csrc.nist.gov/News/2024/nist-to-revise-sp-80038d-gcm-and-gmac-modes.

[Ant06] Joux Antoine. Authentication failures in NIST version of GCM. Comment to NIST, 2006. https://csrc.nist.gov/csrc/media/projects/block-cipher-techniques/documents/bcm/comments/cwc-gcm/ferguson2.pdf.

[App] Apple CryptoKit. Apple. https://developer.apple.com/documentation/cryptokit/.

[BNT19] Mihir Bellare, Ruth Ng, and Björn Tackmann. Nonces are Noticed: AEAD Revisited. Cryptology ePrint Archive, Paper 2019/624, 2019. https://eprint.iacr.org/2019/624.

[CMP23] Matthew Campagna, Alexander Maximov, and John Preuß Mattsson. Galois Counter Mode with Secure Short Tags (GCM-SST). The Third NIST Workshop on Block Cipher Modes of Operation 2023, October 2023. https://csrc.nist.gov/csrc/media/Events/2023/third-workshop-on-block-cipher-modes-of-operation/documents/accepted-papers/Galois%20Counter%20Mode%20with%20Secure%20Short%20Tags.pdf.

[CNG21] Cryptography API: Next Generation. Microsoft, July 2021. https://learn.microsoft.com/en-us/windows/win32/seccng/cng-portal.

[DR03] Joan Daemen and Vincent Rijmen. The Rijndael Block Cipher. Submission to NIST Advanced Encryption Standard Process, April 2003. https://csrc.nist.gov/csrc/media/projects/cryptographic-standards-and-guidelines/documents/aes-development/rijndael-ammended.pdf.

[Dwo04] Morris Dworkin. Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality. National Institute of Standards and Technology, NIST SP 800-38D, May 2004. doi:10.6028/NIST.SP.800-38C.

[Dwo07] Morris Dworkin. Recommendations for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. National Institute of Standards and Technology, NIST SP 800-38C, November 2007. doi:10.6028/NIST.SP.800-38D.

[EJMY21] Patrik Ekdahl, Thomas Johansson, Alexander Maximov, and Jing Yang. SNOW-Vi: an extreme performance variant of SNOW-V for lower grade CPUs. Cryptology ePrint Archive, Paper 2021/236, 2021. https://eprint.iacr.org/2021/236. doi:10.1145/3448300.3467829.

[Fer05] Niels Ferguson. Authentication weaknesses in GCM. Comment to NIST, May 2005. https://csrc.nist.gov/csrc/media/projects/block-cipher-techniques/documents/bcm/comments/cwc-gcm/ferguson2.pdf.

[FLLW17] Christian Forler, Eik List, Stefan Lucks, and Jakob Wenzel. Reforgeability of Authenticated Encryption Schemes. Cryptology ePrint Archive, Paper 2017/332, 2017. https://eprint.iacr.org/2017/332.

[Gue23] Shay Gueron. Constructions based on the AES round and polynomial multiplication that are efficient on modern processor architectures. The Third NIST Workshop on Block Cipher Modes of Operation 2023, October 2023. https://csrc.nist.gov/csrc/media/Presentations/2023/constructions-based-on-the-aes-round/images-media/sess-5-gueron-bcm-workshop-2023.pdf.

[Gue24] Shay Gueron. Double-nonce-derive-key-gcm (dndk-gcm) general design paradigms and application. NIST Workshop on the Requirements for an Accordion Cipher Mode 2024, June 2024. https://csrc.nist.gov/csrc/media/Presentations/2024/double-nonce-derive-key-gcm-dndk-gcm/images-media/sess-6-gueron-acm-workshop-2024.pdf.

[Gui16] NIST Cryptographic Standards and Guidelines Development Process. National Institute of Standards and Technology, NISTIR 7977, March 2016. doi:10.6028/NIST.IR.7977.

[HKR14] Viet Tung Hoang, Ted Krovetz, and Phillip Rogaway. Robust authenticated-encryption: AEZ and the problem that it solves. Cryptology ePrint Archive, Paper 2014/793, 2014. https://eprint.iacr.org/2014/793.

[Hou07] Russ Housley. Using AES-CCM and AES-GCM Authenticated Encryption in the Cryptographic Message Syntax (CMS). RFC 5084, November 2007. URL: https://www.rfc-editor.org/info/rfc5084, doi:10.17487/RFC5084.

[HP08] Helena Handschuh and Bart Preneel. Key-recovery attacks on universal hash function based MAC algorithms. In David A. Wagner, editor, *Advances in Cryptology - CRYPTO 2008, 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceedings*, volume 5157 of *Lecture Notes in Computer Science*, pages 144–161. Springer, 2008. doi:10.1007/978-3-540-85174-5_9.

[IOM12] Tetsu Iwata, Keisuke Ohashi, and Kazuhiko Minematsu. Breaking and repairing GCM security proofs. Cryptology ePrint Archive, Paper 2012/438, 2012. https://eprint.iacr.org/2012/438.

[JAV24] Java Platform, Standard Edition & Java Development Kit Version 22 API Specification. Oracle, 2024. https://docs.oracle.com/en/java/javase/22/docs/api/index.html.

[Jon15] Michael B. Jones. JSON Web Algorithms (JWA). RFC 7518, May 2015. doi:10.17487/RFC7518.

[KEM23] Module-lattice-based key-encapsulation mechanism standard. National Institute of Standards and Technology, NIST FIPS 203 (Initial Public Draft), December 2023. doi:10.6028/NIST.FIPS.203.ipd.

[KEY20] Recommendation for key management: Part 1 – general. National Institute of Standards and Technology, NIST SP 800-57 Part 1 Revision 5, May 2020. doi:10.6028/NIST.SP.800-57pt1r5.

[Lin] Crypto API. The Linux Kernel. https://www.kernel.org/doc/html/v6.9/crypto/index.html.

[LP18] Atul Luykx and Bart Preneel. Optimal Forgeries Against Polynomial-Based MACs and GCM. Cryptology ePrint Archive, Paper 2018/166, 2018. https://eprint.iacr.org/2018/166.

[MAC18] Media Access Control (MAC) Security. Institute of Electrical and Electronics Engineers, IEEE 802.1AE-2018, September 2018. doi:10.1109/IEEESTD.2018.8585421.

[McG08] David McGrew. An Interface and Algorithms for Authenticated Encryption. RFC 5116, January 2008. doi:10.17487/RFC5116.

[MF14] David McGrew and John Foley. Authenticated Encryption with Replay prOtection (AERO). Internet-Draft draft-mcgrew-aero-01, Internet Engineering Task Force, February 2014. Work in Progress. URL: https://datatracker.ietf.org/doc/draft-mcgrew-aero/01/.

[Min15] Kazuhiko Minematsu. Authenticated Encryption without Tag Expansion (or, How to Accelerate AERO). *IACR Cryptol. ePrint Arch.*, 2015:738, 2015. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1aadcbfc8d1b529c9953fced088e9c1929f25eb0.

[MV04] David A. McGrew and John Viega. The Security and Performance of the Galois/Counter Mode of Operation (Full Version). *IACR Cryptol. ePrint Arch.*, page 193, 2004. URL: http://eprint.iacr.org/2004/193, doi:10.1007/978-3-540-30556-9_27.

[MV05] David A. McGrew and John Viega. The Galois/Counter Mode of Operation (GCM). Submission to NIST Modes of Operation Process, May 2005. https://csrc.nist.rip/groups/ST/toolkit/BCM/documents/proposedmodes/gcm/gcm-revised-spec.pdf.

[MW15] John Mattsson and Magnus Westerlund. Authentication key recovery on galois counter mode (GCM). Cryptology ePrint Archive, Paper 2015/477, 2015. https://eprint.iacr.org/2015/477. doi:10.1007/978-3-319-31517-1_7.

[NL18] Yoav Nir and Adam Langley. ChaCha20 and Poly1305 for IETF Protocols. RFC 8439, June 2018. doi:10.17487/RFC8439.

[PC15] Gordon Procter and Carlos Cid. On Weak Keys and Forgery Attacks Against Polynomial-Based MAC Schemes. *J. Cryptol.*, 28(4):769–795, 2015. doi:10.1007/S00145-014-9178-9.

[PKC20] Cryptographic Token Interface Current Mechanisms Specification Version 3.0. OASIS, June 2020. http://docs.oasis-open.org/pkcs11/pkcs11-curr/v3.0/pkcs11-curr-v3.0.pdf.

[Pre23] John Preuß Mattsson. Hidden Stream Ciphers and TMTO Attacks on TLS 1.3, DTLS 1.3, QUIC, and Signal. Cryptology ePrint Archive, Paper 2023/913, 2023. https://eprint.iacr.org/2023/913. doi:10.1007/978-981-99-7563-1_12.

[PST23] John Preuß Mattsson, Ben Smeets, and Erik Thormarker. Proposals for Standardization of Encryption Schemes. The Third NIST Workshop on Block Cipher Modes of Operation 2023, October 2023. https://csrc.nist.gov/csrc/media/Events/2023/third-workshop-on-block-cipher-modes-of-operation/documents/accepted-papers/Proposals%20for%20Standardization%20of%20Encryption%20Schemes%20Final.pdf.

[PST24] John Preuß Mattsson, Ben Smeets, and Erik Thormarker. Comments on NIST's Requirements for an Accordion Cipher. NIST Workshop on the Requirements for an Accordion Cipher Mode 2024, June 2024. https://csrc.nist.gov/csrc/media/Events/2024/accordion-cipher-mode-workshop-2024/documents/papers/comments-on-NIST-reqs-accordion-cipher.pdf.

[Res18] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, August 2018. doi:10.17487/RFC8446.

[Rog11] Phillip Rogaway. Evaluation of Some Blockcipher Modes of Operation. Evaluation carried out for the Cryptography Research and Evaluation Committees (CRYPTREC) for the Government of Japan, February 2011. https://web.cs.ucdavis.edu/~rogaway/papers/modes.pdf.

[RS06] Phillip Rogaway and Thomas Shrimpton. Deterministic authenticated-encryption: A provable-security treatment of the key-wrap problem. Cryptology ePrint Archive, Paper 2006/221, 2006. https://eprint.iacr.org/2006/221.

[RTM22] Eric Rescorla, Hannes Tschofenig, and Nagendra Modadugu. The Datagram Transport Layer Security (DTLS) Protocol Version 1.3. RFC 9147, April 2022. doi:10.17487/RFC9147.

[Saa11] Markku-Juhani O. Saarinen. Cycling Attacks on GCM, GHASH and Other Polynomial MACs and Hashes. Cryptology ePrint Archive, Paper 2011/202, 2011. https://eprint.iacr.org/2011/202.

[Sch22] Jim Schaad. CBOR Object Signing and Encryption (COSE): Initial Algorithms. RFC 9053, August 2022. doi:10.17487/RFC9053.

[Tau09] Hans J Tausch. Simplified birthday statistics and hamming edac. *IEEE Transactions on Nuclear Science*, 56(2):474–478, 2009. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4812293.

[TT21] Martin Thomson and Sean Turner. Using TLS to Secure QUIC. RFC 9001, May 2021. doi:10.17487/RFC9001.

[VM05] John Viega and David McGrew. The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP). RFC 4106, June 2005. doi:10.17487/RFC4106.

[W3C17] Web Cryptography API. W3C, January 2017. https://www.w3.org/TR/WebCryptoAPI/.

[WPA24] WPA3 Specification Version 3.3. Wi-Fi Alliance, February 2024. https://www.wi-fi.org/system/files/WPA3%20Specification%20v3.3.pdf.