ERICSSON

Date: July 10, 2026

Ericsson AB
Group Function Technology
SE-164 80 Stockholm
SWEDEN

Comments on NIST SP 800-38D (GCM, GMAC, and wGCM)

Dear NIST,

Thanks for your continued efforts to produce well-written, user-friendly, and open-access security documents. We welcome NIST's plans to revise SP 800-38D and to standardize Rijndael-256 and wGCM. 3GPP and ETSI SAGE have already standardized Rijndael-256 for use in the MILENAGE-256 authentication and key-generation function [1]. Below, we provide feedback on the questions raised in [2], building on Ericsson's previous comments in [3–6].

In the comments below, we use the following notation:
- q, the number of encryption queries
- v, the number of decryption queries
- ℓ, the maximum number of authenticated blocks (plaintext + associated data)
- ℓ_P, the maximum number of plaintext blocks
- η, the length in bits of the random field in the RBG-based IV construction
- τ, the authentication tag length in bits

Questions on GCM:

- *"The revision to GCM will remove support for authentication tags whose lengths are less than 96 bits. Which of the remaining tag sizes (i.e., 96, 104, 112, 120, or 128 bits) are needed for GCM? In particular, would tag sizes of 96 bits and 128 bits be sufficient for applications?"*

  We are not aware of any applications using 104-, 112-, or 120-bit GCM tags. While 96- and 128-bit tags appear sufficient, we see no compelling reason to restrict the current 96–128-bit range. However, NIST should significantly reduce the maximum length of the associated data A. With $\ell = 2^{57}$ blocks of authenticated data, the forgery probability for 96-bit tags is $\approx 2^{-39}$, which is clearly not what users expect. NIST should also restrict the number of invocations for deterministic 96-bit IVs to ensure that the Bernstein bound factor [7]

$$\delta_{128} \lessapprox 1 + (q\ell_P + v)^2 / 2^{129} \quad ,$$

remains close to 1. The Bernstein bound factor δ depends on the total number of block cipher invocations σ, which we for simplicity upper-bound by $\sigma \lessapprox q \cdot \ell_P + v$. Furthermore, NIST should provide guidance on the enforcement of invocation limits in group-key environments, including one-to-many, many-to-one, and many-to-many communication scenarios.

New applications where byte efficiency and strong security matter should follow the example of 5G and 6G systems and adopt GCM with Strong Secure Tags (GCM-SST) [8], designed by ETSI SAGE and standardized by 3GPP in TS 35.240–35.248. GCM-SST mitigates several known vulnerabilities in GCM, significantly reducing integrity advantage bounds and the expected number of successful forgeries.

- *"GCM currently supports variable-length IVs by applying the internal hash function to generate a new, 96-bit IV value. Are variable-length IVs important for any applications?"*

  Longer IVs need to be discussed together with random IVs and Nonce-based Key Derivation (NKD). The observation that *"when the nonce length is restricted to 96 bits, GCM has better security bounds"* [3, 7] applies only to deterministic IVs. Random IVs shorter than the block size degrade both the privacy and authenticity bounds of GCM [7, 9] by introducing an additional $q^2 / 2^{\eta+1}$ term. NIST should not prohibit long IVs while continuing to permit random IVs. Nonce-based Key Derivation (NKD) appears to be a superior approach for addressing the same use case as random IVs, and NKD is already deployed in several large-scale systems [10–11]. We suggest that NIST mandate deterministic 96-bit IVs and specify NKD for all NIST-approved encryption modes. We note that NKD, when constructed by combining an approved KDF with an approved AEAD, is already approved, but would benefit from additional guidance.

Questions on wGCM:

- *"Would a limit of 2^{64} wGCM invocations per key be sufficient?"*

  A limit of 2^{64} invocations would be sufficient for all use cases. However, 2^{64} invocations cannot be combined with message lengths of 2^{64} blocks without incurring security degradation due to the Bernstein bound factor [7]

$$\delta_{256} \lessapprox 1 + (q\ell_P + v)^2 / 2^{257} \quad .$$

  To maintain $\delta_{256} \approx 1$, which should be a strict requirement, one option is to require that

$$q\ell_P + v \ll 2^{128} \quad .$$

  Furthermore, a single key should not be used for such a large number of invocations. To align with zero-trust principles and minimize the impact of key compromise, rekeying should occur well before reaching 2^{64} invocations. Even in historical systems where keys were distributed on paper, high-security practice commonly employed a cryptoperiod of one day [12]. Modern guidance recommends rekeying via ephemeral asymmetric key exchange providing Forward Secrecy (FS) and Post-Compromise Security (PCS) after 1 hour or 2^{25}–2^{32} 256-bit blocks [13–15].

While we believe NIST should increase the invocation and plaintext-length limits relative to GCM, we are cautious about increasing the limit on the total number of blocks encrypted under a single key. Even if cryptographic bounds permit larger volumes of encrypted data, NIST should recommend much stricter limits to reduce the consequences of key compromise.

- *"What authentication tag sizes should be specified for wGCM (e.g., 128, 192, or 256 bits) to support larger data limits than GCM? Is more than one tag size necessary?"*

  Rather than asking what tag lengths should be specified, the relevant question is what level of security the tags should provide. Let $E(F)$ denote the expected number of successful forgeries. Assuming a sufficiently large key size such that brute-force key-recovery attacks can be neglected, a strong integrity mechanism should satisfy

$$E(F) \approx v/2^{\tau} \quad ,$$

  for all allowed values of q, v, ℓ, and τ. We recommend that all future NIST standards, including wGCM, target this robust security definition. Users, developers, and standards authors typically expect that $E(F) \approx v/2^{\tau}$.

  We believe $E(F) \approx v/2^{128}$ provides sufficient security for all use cases, as it likely falls below the probability of a verification incorrectly succeeding due to environmental phenomena, such as cosmic rays and thermal noise. Shorter tags with $E(F) \approx v/2^{\tau}$ are, however, needed in many use cases. The fact that both IETF [16] and 3GPP [8] have recently standardized new AEADs demonstrates that NIST's existing AEADs do not meet all industry requirements.

- *"NIST proposes 192 bits as the default IV length for wGCM, devoting the remaining 64 bits of the IV block within wGCM to the message block counter for the GCTR encryption. Is there a need to support longer IVs in wGCM by applying the internal hash function, as in GCM?"*

  We believe wGCM should support a single fixed IV length smaller than the block size. A 192-bit IV appears appropriate. However, message lengths of 2^{64} blocks cannot be combined with 2^{64} invocations due to the birthday bound and zero-trust principles; see our answer on the invocation limit.

- *"The deterministic IV construction for GCM defines a fixed field and an invocation field. For wGCM, NIST proposes to recommend a fixed field of 96 bits and an invocation field of 96 bits to promote interoperability. Are other options needed?"*

  The length of the invocation field should be aligned with the invocation limit and CPU-efficient data sizes. A 128-bit fixed field may be useful for larger identities and contexts. We suggest NIST recommend a 128-bit fixed field and a 64-bit invocation field.

- *"The random IV construction for GCM defines a random field and a free field. For wGCM, if the number of invocations per key were limited to 2^{64}, and if 192 bits were specified for the random field, then the probability of an IV collision would be limited to 2^{-64}. Is the free field useful/necessary for wGCM?"*

Rather than discussing collision probabilities, NIST should discuss security strength. Random IVs shorter than the block size degrade both the privacy and authenticity bounds by introducing an additional q^2 / $2^{\eta+1}$ term. If the free field is used to identify the device, this term becomes q_i^2 / $2^{\eta+1}$, where q_i is the number of encryption queries using free field i. Although the success probability of the attack in the single-user setting is at most 2^{-64}, the corresponding multi-user attack has a success probability that approaches 1 as the number of users grows. We believe that random 192-bit IVs are acceptable provided their security properties are accurately described, although Nonce-based Key Derivation (NKD) appears to be a superior approach.

- *"Are these two 192-bit IV constructions sufficiently flexible, or should NIST consider adding other, comparably secure constructions?"*

  We think the IV constructions for wGCM should be recommendations. We do not see the need to describe any other IV constructions but we suggest NIST specifies NKD for wGCM.

- *"The natural generalization of the GCM hash function (i.e., GHASH) for wGCM would specify a suitable polynomial of degree 256 to define a hash function (called wide-GHASH) on 256-bit blocks. The output can be truncated as needed to obtain the appropriate tag size. Should NIST consider a different option for the hash function component of wGCM? For example, two instances of GHASH using independent 128-bit keys could be concatenated and truncated as needed. Since this option (called concat-GHASH) operates on 128-bit blocks, it would be more efficient than wide-GHASH. However, concat-GHASH cannot provide as much generic security assurance as wide-GHASH and would require much more restrictive usage bounds (i.e., smaller limits on total data, the message length, and invocations per key) for any given security target."*

  The authentication security of GCM is already far from ideal. For 128-bit tags, assuming $\delta \approx 1$, the integrity advantage and expected number of successful forgeries are

  $$\mathrm{Adv} \approx v\ell/2^{128} \quad \text{and} \quad E(F) \approx v^2\ell/2^{129} \ ,$$

  which, for $\ell = 2^{57}$ becomes

  $$\mathrm{Adv} \approx v/2^{71} \quad \text{and} \quad E(F) \approx v^2/2^{72} \ ,$$

  which is significantly worse than the expected

  $$E(F) \approx v/2^{128} \ .$$

  Our understanding is that concat-GHASH would have

  $$\mathrm{Adv} \lessapprox v\ell^2/2^{\tau} \quad \text{and} \quad E(F) \lessapprox v^2\ell^2/2^{\tau+1} \ ,$$

  weakening security with an additional ℓ factor compared to GCM. We do not believe NIST should deviate even further from $E(F) \approx v/2^{\tau}$. We think it was a mistake that the two vulnerabilities pointed out by Ferguson [17] were not mitigated in the final standard as suggested by Nyberg, Gilbert, and Robshaw [18]. As a result, GCM was not state-of-the-art even at the time SP 800-38D was published. We are not aware of any public rationale from NIST for not incorporating the recommendations in [18]. One possible explanation is compatibility with existing pre-standard implementations. In our view, this led to long-term drawbacks: standardized IPsec [19] still, two decades later, relies on non-standardized variants of GCM, and it remains unclear to what extent deployed implementations conform to all requirements of SP 800-38D.

wGCM does not have any pre-standard implementations and NIST should address as many vulnerabilities as possible. The $\approx v/2^{\tau}$ security of [8] in protocols with replay protection has now been proven by Inoue et al. [20] and Naito et al. [21]. Security protocols with replay protection represent the dominant use case for GCM and wGCM. Deriving fresh authentication keys for each nonce provides not only reforgeability resistance but also nonce-misuse resilience [22].

We agree with NIST that wGCM should use a binary field. However, we do not believe that NIST should consider concat-GHASH, and we do not recommend using either big-endian GHASH or a polynomial of degree 256. Better performance and strong security can be achieved using a trinomial with a lower degree. A detailed study of efficient multiplication and reduction techniques using optimal polynomials over binary fields was carried out by Maximov et al. [23]. Depending on the targeted security level and limits on plaintext and associated data lengths, suitable high-performance choices include, for example, $x^{225} + x^{128} + 1$, $x^{191} + x^{120} + 1$, $x^{185} + x^{144} + 1$, and $x^{159} + x^{128} + 1$. For authenticated data lengths up to $\ell = 2^{63}$ blocks, a 191-degree polynomial can provide 128-bit integrity, while a 159-degree polynomial can provide 96-bit integrity. Note that $E(F) \approx v/2^{96}$ is significantly stronger than any of NIST's currently approved AEAD schemes.

We suggest that wGCM incorporate little-endian optimizations, use fixed-length deterministic nonces, derive two fresh authentication keys per nonce [8, 18], avoid concat-GHASH, adopt a wide-POLYVAL construction based on a high-performance trinomial of sufficient degree, and choose usage limits and tag truncation to ensure that

$$E(F) \approx v/2^{\tau}$$

for all allowed values of q, v, ℓ, and τ.

John Preuß Mattsson,
Expert Cryptographic Algorithms and Security Protocols, Ericsson
On behalf of the Ericsson Cryptography Team

References

[1] Specification of the MILENAGE-256 algorithm set
https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4244

[2] Second Pre-Draft Call for Comments: GCM and GMAC
https://csrc.nist.gov/pubs/sp/800/38/d/r1/2prd

[3] 2021 public comments on NIST SP 800-38D (GCM and GMAC)
https://csrc.nist.gov/csrc/media/Projects/crypto-publication-review-project/documents/initial-comments/sp800-38d-initial-public-comments-2021.pdf

[4] 2023 public comments on NIST SP 800-38D (GCM and GMAC)
https://csrc.nist.gov/csrc/media/Projects/crypto-publication-review-project/documents/decision-proposal-comments/sp800-38d-decision-proposal-comments-2023.pdf

[5] 2025 public comments on NIST SP 800-38D (GCM and GMAC)
https://csrc.nist.gov/files/pubs/sp/800/38/d/r1/upd/iprd/docs/sp800-38d-pre-draft-public-comments.pdf

[6] 2025 public comments on NIST SP 800-197A (Cryptographic Accordions)
https://csrc.nist.gov/csrc/media/Events/2024/accordion-cipher-mode-workshop-2024/documents/papers/comments-on-NIST-reqs-accordion-cipher.pdf

[7] Breaking and Repairing GCM Security Proofs
https://eprint.iacr.org/2012/438.pdf

[8] Galois Counter Mode with Strong Secure Tags (GCM-SST)
https://www.ietf.org/archive/id/draft-mattsson-cfrg-aes-gcm-sst-21.html

[9] GCM Security Bounds Reconsidered
https://eprint.iacr.org/2015/214.pdf

[10] Public comments on FIPS 197 (AES)
https://csrc.nist.gov/csrc/media/Projects/crypto-publication-review-project/documents/initial-comments/fips-197-initial-public-comments-2021.pdf

[11] Double-Nonce-Derive-Key-GCM (DNDK-GCM) General design paradigms and application
https://csrc.nist.gov/csrc/media/Presentations/2024/double-nonce-derive-key-gcm-dndk-gcm/images-media/sess-6-gueron-acm-workshop-2024.pdf

[12] Military Cryptanalysis Part II
https://www.nsa.gov/portals/75/documents/news-features/declassified-documents/friedman-documents/publications/FOLDER_257/41751589079090.pdf

[13] The Secure Shell (SSH) Transport Layer Protocol
https://www.rfc-editor.org/rfc/rfc4253.html

[14] Recommendations for securing networks with IPsec
https://messervices.cyber.gouv.fr/documents-guides/NT_IPsec_EN.pdf

[15] Continuous Updating and Ratcheting for Rekeying Encrypted Network (CURRENT)
https://datatracker.ietf.org/doc/bofreq-housley-continuous-updating-and-ratcheting-for-rekeying-encrypted-network-transport/

[16] Secure Frame (SFrame): Lightweight Authenticated Encryption for Real-Time Media
https://www.rfc-editor.org/rfc/rfc9605.html

[17] Authentication weaknesses in GCM
https://csrc.nist.gov/csrc/media/projects/block-cipher-techniques/documents/bcm/comments/cwc-gcm/ferguson2.pdf

[18] Galois MAC with forgery probability close to ideal
https://csrc.nist.gov/CSRC/media/Projects/Block-Cipher-Techniques/documents/BCM/Comments/general-comments/papers/Nyberg_Gilbert_and_Robshaw.pdf

[19] The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP)
https://www.rfc-editor.org/rfc/rfc4106.html

[20] Generic Security of GCM-SST
https://eprint.iacr.org/2024/1928.pdf

[21] The Multi-user Security of GCM-SST and Further Enhancements
https://link.springer.com/chapter/10.1007/978-3-032-08124-7_3

[22] Boosting Authenticated Encryption Robustness with Minimal Modifications
https://link.springer.com/chapter/10.1007/978-3-319-63697-9_1

[23] On Fast Multiplication in Binary Finite Fields and Optimal Primitive Polynomials over GF(2)
https://eprint.iacr.org/2017/889.pdf