Public Comments on SP 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC

Comment period: August 6, 2021 – October 1, 2021

On August 6, 2021, NIST's Crypto Publication Review Board requested public comments for the review of SP 800-38D, *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*. The public comment period closed on October 1, 2021.

More details about this review are available at NIST's Crypto Publication Review Project site.

LIST OF COMMENTS

1. Center for Cybersecurity Standards (CCSS), National Security Agency, June 6 and October 1, 2021

[Received June 6, 2021]

GCM PIECE	OBSERVATIONS	REFERENCES	WITH CURRENT COMPUTING ABILITIES	SUGGESTIONS
NONCE/IV	Reusing the nonce/IV allows adversary to learn the key, which is referred to as the Forbidden Attack Example: Major companies caught reusing nonce. VISA is example.	Forbidden Attack paper: JOUX - Authentication Failures in NIST version of GCM VISA attack: GitHub - nonce-disrespect/nonce-disrespect: Nonce-Disrespecting Adversaries: Practical Forgery Attacks on GCM in TLS	This is a well-known issue. NIST discourages nonce reuse. The Birthday Attack is the best way to attack GCM if IVs do not repeat.	AES-GCM-SIV addresses the problem of nonce reuse. This mode is not as fast as GCM because the nonce misuse resistance property requires two passes over the data. Generate a new 96-bit nonce for each message using a cryptographically strong PRNG. Re-key at reasonably regular intervals, where "reasonably regular" is defined by how much data and how many messages are being encrypted.

	Variable lengths of the nonce: “when the nonce length is restricted to 96 bits, GCM has better security bounds than a general case of variable length nonces”; default length of IV is 96 bits	Breaking and Repairing GCM Security Proofs: https://eprint.iacr.org/2012/438.pdf		Experts recommend a fixed IV length
	If the IV is all zeros, the value of the hash key can be learned. This hash key is the universal hash function underlying the MAC scheme. An adversary can compute the keyed hash of any ciphertext once the key is known.		This was only true in NIST’s original draft of 800-38D. Even if the IV’s 96 bits are all zeros, with the counter = 31 zeros and 1 one, this is not the case. Could have a hash key = 128 zeros.	
H KEY	Weak key can lead to message forgeries	Security analysis of GCM for communication: https://onlinelibrary.wiley.com/doi/pdf/10.1002/sec.798	Most attacks on the H key are expensive and not realistic.	
		Key-Recovery Attacks on Universal Hash Function based MAC Algorithms: https://www.iacr.org/archive/crypto2008/51570145/51570145.pdf	The exception is the MAC algorithm. (Birthday attack also mentioned in this paper, section 3.3.)	

	Cycling attack - Bad values of the internal H key, which can be pre-calculated for specific AEAD key values, can negatively impact security	Cycling attack paper: https://eprint.iacr.org/2011/202.pdf		
LOOK-UP TABLES	Caching used to speed up use of GCM but can lead to cache leakage	Faster and Timing-Attack Resistant AES-GCM: https://eprint.iacr.org/2009/129.pdf		Nothing to report at this time.
TAG	Short tag can lead to adversary producing message forgeries. Example: If the tag is 32 bits, then after 2^{16} (65,536) forgery attempts and 2^{16} encryptions of chosen plaintext, a forged ciphertext can be produced. Forgeries can be created quickly when enough forgeries have been found.	GCM Update: https://csrc.nist.gov/csrc/media/projects/block-cipher-techniques/documents/bcm/comments/cwc-gcm/gcm-update.pdf		Experts recommend 128-bit tag
COUNTER	Counter wrapping, or integer overflow, because counter is 32 bits		Counter should not overflow	
TOTAL PLAINTEXT MESSAGE LENGTH 68GB	Not a vulnerability but might be a drawback			

IF INCLUDING AES	Timing Attack - successful extraction of a complete AES key from a network server on another computer	Cache-timing attacks on AES paper by Bernstein: https://cr.yp.to/antiforgery/cachetiming-20050414.pdf	The full 10, 12, or 14 rounds of AES, depending on key length, have not been successfully attacked to date.

[Received Oct. 1, 2021]

Comments for SP 800-38D, *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*

We reviewed this NIST Draft and appreciate this effort and the opportunity to provide the below comments from our SMEs for your consideration:

Higher data rates have made it difficult to manage IVs in certain applications to avoid re-use. Because there are serious consequences to re-using an IV in GCM, there is a serious need for a new mode that mitigates IV re-use.

1. IV Limits

In section 8.3, recall that the total number of invocations of the authenticated encryption function with a given key is limited to no more than 2^{32}, unless one uses only 96-bit IVs that are generated using the deterministic construction. This limit can be unnecessarily restrictive for m-bit IVs with m > 96. For example, consider 128-bit IVs generated using the RBG-based construction in section 8.2.2, and where the random field has length r = 128 bits and the free field has length 0 bits (i.e., the random field is the entire IV, as recommended). Then limiting the total number of invocations (and hence, the number of IVs) to no more than 2^{32} implies that the probability of a repeated IV is about $2^{63-128} = 2^{-65}$. This is far smaller than the mandated 2^{-32} upper bound on the probability of a repeated IV stated at the beginning of section 8, and many additional 128-bit IVs could be safely generated before nearing the upper bound.

One way to partially mitigate this limitation in SP 800-38D is **to set an alternative limit for those implementations that fix a single IV length for all invocations with a given key**. As such, in the discussion below we assume only one IV length is allowed for a given key.

IVs generated using the RBG-based construction

According to section 8.2.2, for IVs generated using the RBG-based construction, the random field (and hence, the IV itself) must have length r ≥ 96 bits, and for any given IV length, the value r must be fixed for the life of the key. Thus, for IVs generated using the RBG-based construction and where only one IV length is allowed per key, one option is to limit the total number of IVs with a given key to 2^s, where

$s = \min((r-31)/2, 48.5)$.

The table below shows the proposed value of the limit, 2^s, for various values of r.

r	96	104	112	120	≥ 128
2^s	$2^{32.5}$	$2^{36.5}$	$2^{40.5}$	$2^{44.5}$	$2^{48.5}$

First, this limit ensures that the probability of a repeated IV is $\mathrm{Bin}(2^s,2)/2^r \leq 2^{r-32}/2^r = 2^{-32}$, as desired. Here Bin(n,k) = n!/(k!(n-k)!) denotes the binomial coefficient associated with n and k. Furthermore, for IVs with r > 128, this limit ensures that if the IVs are distinct, the probability of a repeated 128-bit J_0 value is $\mathrm{Bin}(2^s,2)/2^{128} \leq 2^{96}/2^{128} = 2^{-32}$. J_0 is the 128-bit pre-counter block obtained by applying GHASH to the IV; it can be considered the "effective IV" for IVs whose length is not 96. While J_0 collisions may be more difficult to detect than IV collisions (since J_0 is secret), it may still be desirable to bound the probability of a repeated J_0 value. Note that if the IV length is fixed for a given key, is not 96, and is ≤ 128 (in which case r ≤ 128 as well), then due to the details of the GHASH function, there cannot be a J_0 collision for distinct IVs unless the hash key H = 0, which occurs with probability 2^{-128}.

IVs generated using the deterministic construction

For IVs generated using the deterministic construction and where only one IV length m is allowed per key, one option is to impose no IV limit if m ≤ 128 (other than the constraints in section 8.3 on the invocation and fixed fields of the IV), and to limit the total number of IVs with a given key to $2^{48.5}$ if m > 128. For the deterministic construction IVs cannot repeat. Furthermore, due to the details of the GHASH function, if the fixed IV length m ≤ 128 (and is not 96), then there cannot be a J_0 collision unless the hash key H = 0. If m > 128, the limit ensures that the probability of a repeated 128-bit J_0 value is $\text{Bin}(2^{48.5},2)/2^{128} = 2^{-32}$.

Remark: We note that the proposed limits above are meant solely to provide upper bounds on the probability of a repeated IV or J_0 value. They do not consider adversaries whose goal is to distinguish the output of GCM from the output of a random function under the IND-CPA model (indistinguishability under chosen-plaintext attack). While such goals may be achievable even with the IV limits proposed (including the current limit of 2^{32} in SP 800-38D), distinguishing attacks are less concerning than repeated IVs.

2. Short Tags

We suggest that NIST consider removing the option for 32-bit and/or 64-bit tags, with the understanding that disallowing these tag sizes would only be applicable to new implementations of GCM. Guidelines for using 32-bit and 64-bit tags are provided in Appendix C, and it is suggested there that the guidelines should be straightforward for “knowledgeable security professionals”. While that may be true, those without such a background may still attempt to implement GCM with short tags, raising the risk of vulnerable implementations. Furthermore, a recent trend to put less responsibility on implementers of cryptographic protocols and designs seems to be gaining momentum. For example, this can be seen with the emergence of the notion of nonce-misuse resistance in several recent block cipher modes. To this end, removing the option for short tags (and the nontrivial guidelines that need to be followed when using short tags) would support this growing paradigm. Finally, the Secure Real-time Transport Protocol (SRTP) is mentioned in Appendix C as an example of a protocol where short tags may be appropriate. While that may have been considered true at the time of the publication of SP 800-38D in 2007 (and perhaps is still considered true by some), note that RFC 7714 (which discusses the use of AES-GCM in SRTP) mandates 128-bit tags in section 13.2. In particular, the authors of the RFC and the working group believe that the risks associated with truncated tags in SRTP are too high. Thus, we believe NIST should at least consider the possibility of removing the options for 32-bit and/or 64-bit tags in new implementations of GCM.

2. Canadian Centre for Cyber Security (CCCS), September 1, 2021

As FIPS 140-2 will be retired as of September 21, 2021, reference [3] in SP 800-38D should be updated to FIPS 140-3. Several mention of FIPS 140-2 in Section 9, page 22, should be also updated to call on FIPS 140-3. Similarly, reference [9] in SP 800-38D should be replaced with the corresponding document "Implementation Guidance for FIPS 140-3and the Cryptographic Module Validation Program", https://csrc.nist.gov/CSRC/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/FIPS%20140-3%20IG.pdf

In [1], the authors show that McGrew and Viega's original security proofs were flawed. Fortunately, in this paper the proofs are repaired and the new security bounds are established. As a result, the reference to security at the end of Introduction (Section 3) should be updated to include Iwata et at. Furthermore, the new security bound is better for 96-bit IVs and, supported by the fact that many applications already use this fixed IV length, it would be reasonable to recommend it as default.

For certain weak authentication key classes, the security of the algorithm break down [2, 3, 4, 6]. In the literature, there are available criteria to test for weak keys and, while it might not be possible to always check for weak keys due to efficiency issues, the standard should warn of their existence and hence the vulnerability to cycling attacks.

In view of new key recovery attacks, in particular [5], the use of short tags should be discouraged except potentially except in exceptional circumstances, such as in constrained devices or environments which may necessitate shorter tags. Note that while Appendix C lists potential applications and uses SRTP as an example, this protocol itself now mandates the use of 128-bit tags [7].

Additional comments on Appendix C:

- Appendix C, page 28, item 2: please include a clarification (if available) as to why AAD cannot contain additional information user would like to include to be authenticated.
- Appendix C, page 28, last paragraph regarding SRTP is not correct as this protocol violates the stated conditions 1 and 2. See [5, 7] for details. Furthermore, as pointed out above, SRTP no longer allows the use of short tags altogether in GCM mode.
- Appendix C, page 29, tables 1 and 2: the “Maximal permitted number of invocations of the authenticated decryption function” is not specified for message lengths exceeding 2^10/2^25 bytes for 32/64-bit tags. Specify if longer lengths for the ciphertext and AAD are supported or not.

Section 8: as pointed out in [8] (page 133), it is not clear how the requirements for 2^(-32) probability can be controlled or enforced. Moreover, only section 8.3 brings up a numerical requirement to support this, while the "with high probability" statement in 8.1 is not specific.

Section 8.3: as per clarification from McGrew reported in [8] (page 133), the bold text should read "The total number of invocations of the authenticated encryption function shall not exceed 2^32, including invocations with all IV lengths and all instances of the authenticated encryption function with the given key. "

References:

[1] Iwata T., Ohashi K., Minematsu K. (2012) Breaking and Repairing GCM Security Proofs. In: Safavi-Naini R., Canetti R. (eds) Advances in Cryptology – CRYPTO 2012. CRYPTO 2012. Lecture Notes in Computer Science, vol 7417. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-32009-5_3

[2] Saarinen M. Cycling Attacks on GCM, GHASH and Other Polynomial MACs and Hashes. In Anne Canteaut, editor, FSE, volume 7549 ofLecture Notes in Computer Science, pages 216–225. Springer, 2012 https://eprint.iacr.org/2011/202.pdf

[3] Handschuh H. and Preneel B. Key-Recovery Attacks on Universal HashFunction Based MAC Algorithms. In David Wagner, editor,CRYPTO, volume5157 ofLecture Notes in Computer Science, pages 144–161. Springer, 2008, https://www.esat.kuleuven.be/cosic/publications/article-1150.pdf

[4] Abdelraheem M.A., Beelen P., Bogdanov A., Tischhauser E. (2015) Twisted Polynomials and Forgery Attacks on GCM. In: Oswald E., Fischlin M. (eds) Advances in Cryptology -- EUROCRYPT 2015. EUROCRYPT 2015. Lecture Notes in Computer Science, vol 9056. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-46800-5_29

[5] Mattsson J., Westerlund M. Authentication Key Recovery on Galois/Counter Mode (GCM). In: Pointcheval D., Nitaj A., Rachidi T. (eds) Progress in Cryptology – AFRICACRYPT 2016. AFRICACRYPT 2016. Lecture Notes in Computer Science, vol 9646. Springer, Cham. https://doi.org/10.1007/978-3-319-31517-1_7

3. John Preuß Mattsson, Ericsson, September 30, 2021

Dear NIST,

Thanks for your continuous efforts to produce well-written open-access security documents. Please find attached our comments on SP 800-38D.

Best Regards,
John Preuß Mattsson,
Senior Specialist, Ericsson

Date: September 30, 2021

Comments on SP 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC

Dear NIST,

Thanks for your continuous efforts to produce well-written open-access security documents. FIPS 198-1, SP 800-22 Rev. 1a, SP 800-38D, SP 800-38E, and SP 800-107 Rev. 1 are all important documents that should be updated.

Please find below our comments on SP 800-38D:

- The “If len(IV) ≠ 96” where the IV is hashed does not seem to be used in practice, provide weaker security, and adds complexity. We suggest that this IV hashing alternative is deprecated.

- Many recent IETF protocols like TLS 1.3 [RFC8446], OSCORE [RFC8613], Encrypted Content-Encoding for HTTP [RFC8188] etc. does not adhere to the IV constructions in 800-38D. The update to 800-38D should allow for IV to be constructed as a 96-bit fixed random number XORed with the invocation field. Such a construction could optionally allow 128-bit randomness where the block counter is also XORed with the fixed random number instead of being concatenated.

- As stated in [1], several of the statements in Appendix C are not correct. SRTP does in general not meet the guidelines. The idea that an attacker does not get side-channel information about successful forgeries is almost always wrong and very dangerous. As GCM with short tags does not seem to be used, we would recommend to just remove Appendix C and forbid short, truncated tags.

  Note that a high-performance software friendly AEAD algorithm like AES-GCM with secure short tags (e.g., 64 bits) would be useful in wireless networks.

  [1] https://eprint.iacr.org/2015/477.pdf

Best Regards,
John Preuß Mattsson,
Senior Specialist, Ericsson

4. Salesforce Cryptographic Review Board, October 1, 2021

Comments from Joe Salowey, Matthew Schechtman, Prasad Peddada, Taher Elgamal and the members of the Salesforce Cryptographic Review Board.

This message is a response to the request for comments for the NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC.

1. Nonce Length - the current specification encourages the use of a 96-bit nonce “to promote interoperability, efficiency, and simplicity of design.” While these are good reasons to use 96-bit nonce, [IWATA-2012] shows that there are security reasons that a 96-bit nonce should be used. In particular, a longer nonce does not improve the security of AES-GCM. NIST should consider adding security to the list of reasons and consider stricter guidance on nonce length.
2. Nonce reuse - Currently, the AEAD modes CCM and GCM have significant failures when a nonce is repeated. This is something that implementations continue to have problems with. An AEAD mode that was tolerant of nonce reuse or not dependent on the use of a nonce could remove this failure mode.
3. Parameters - It would be beneficial to have an AEAD mode that eliminates the possibility of choosing insecure parameters.

[IWATA-2012] Iwata, T., Ohashi, K., and K. Minematsu, "Breaking and Repairing GCM Security Proofs", 1 August 2012, <https://eprint.iacr.org/2012/438.pdf>.