Public Comments on SP 800-107 Rev. 1, Recommendation for Applications Using Approved Hash Algorithms

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

On August 6, 2021, NIST's Crypto Publication Review Board initiated a review of SP 800-107 Rev. 1, *Recommendation for Applications Using Approved Hash Algorithms* (August 2012). This document includes the public comments received during the comment period from August 6, 2021 to October 1, 2021.

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

LIST OF COMMENTS

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

As FIPS 140-2 will be retired as of September 21, 2021. Reference to it should be replaced by references to FIPS 140-3.

Comments on SP 800-107r1:

A number of other revisions should be included in the document:

- FIPS 186-3 was updated to FIPS 186-4 (with revision 5 in draft form, potentially coming out soon)
- SP 800-56A was updated to Revision 3
- SP 800-56B was updated to Revision 2
- SP 800-56C was updated to Revision 2
- SP 800-57 Part 1 was updated to Revision 5
- SP 800-90A was updated to Revision 1
- SP 800-131A was updated to Revision 2
- SP 800-133 was updated to Revision 2
- SP 800-135 was updated to Revision 1

In Section 4.1, page 7, discussion on security properties and usage of SHA-1, updated and complete reference [1] could be used to point out that the security proof for HMAC does not rely on collision resistance of the underlying PRF.

Footnote 4 on page 14 discusses exclusion of impractical collision attacks from this document. Researchers since found existing generic attacks similarly impractical [2, 3] and this information can be included in a similar footnote.

References:

[1] Bellare, M. New Proofs for NMAC and HMAC: Security without Collision Resistance. J Cryptol 28, 844–878 (2015). https://doi.org/10.1007/s00145-014-9185-x

[2] Peyrin T., Sasaki Y., Wang L. (2012) Generic Related-Key Attacks for HMAC. In: Wang X., Sako K. (eds) Advances in Cryptology – ASIACRYPT 2012. ASIACRYPT 2012. Lecture Notes in Computer Science, vol 7658. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34961-4_35

[3] Guo J., Peyrin T., Sasaki Y., Wang L. (2014) Updates on Generic Attacks against HMAC and NMAC. In: Garay J.A., Gennaro R. (eds) Advances in Cryptology – CRYPTO 2014. CRYPTO 2014.

Lecture Notes in Computer Science, vol 8616. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-44371-2_8

2. Comments from 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 800-107 Rev. 1.

Best Regards,

John Preuß Mattsson,

Senior Specialist, Ericsson

ERICSSON

Date: September 30, 2021

Comments on SP 800-107 Rev. 1: Recommendation for Applications Using Approved Hash Algorithms

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 800-107 Rev. 1:

- "FIPS 180-4"

  I assume this will be updated to [180-4] and [FIPS PUB 202]. I think the updated document should describe SHAKE and KMAC [800-185] in the same way as it discusses SHA-2 and HMAC. The SHAKE functions are used quite a lot (X.509, EdDSA, COSE, etc.) while the fixed-length SHA-3 hash algorithms seem to see limited practical use. Long-term I think NIST should consider referring to SHAKE as variable-length hash functions. Right now, the terminology is a bit confusing. NIST states that the variable-length KMAC is a keyed hash function but insists that SHAKE is not a hash function.

- It would be good if table 1 also listed security against length extension attacks. The low resistance against length extensions in many of the SHA-2 variants is not very nice and might come as a surprise to people using SHA-2.

- "A commonly acceptable length for the MacTag is 64 bits; MacTags with lengths shorter than 64 bits are discouraged."

  This is still a good general recommendation that does **not** require an update.

  The DTLS 1.3 draft [1] has recently forbidden 64-bit tags based on the single key integrity advantage. This measure is of theoretic interest but is not a good measure for security protocols where each connection has many keys and communication between two parties can use many connections. The process used in DTLS 1.3 leads to misleading results like that frequent rekeying the ideal MAC increases security [2].

  While using only 128-bit tags might be fine for many non-constrained systems, using 64-bit tags make perfect sense in constrained IoT. To break 64-bit security against online brute force an attacker would on average have to send 4.3 billion messages per second for 68 years, which is totally infeasible in constrained IoT radio technologies.

[1] https://datatracker.ietf.org/doc/html/draft-ietf-tls-dtls13

[2] https://datatracker.ietf.org/meeting/110/materials/slides-110-saag-analysis-of-usage-limits-of-aead-algorithms-00

Best Regards,
John Preuß Mattsson,
Senior Specialist, Ericsson