Proposals for Standardization of Encryption Schemes

John Preuß Mattsson, Ben Smeets, Erik Thormarker
Ericsson

Abstract: NIST's standardized encryption modes have been extremely successful and important for securing data in transit and data at rest. NIST's current selection is however starting to show its age. NIST lacks an approved wide block tweakable cipher such as Adiantum, appropriate for length-preserving encryption, AEAD modes hardened against nonce misuse such as AES-SIV and AES-GCM-SIV, AEAD modes suitable for use with random nonces such as AEGIS-256, high-performance AEAD modes such as AEGIS, AEAD modes suitable for long plaintexts such as AEGIS, an alternative to AES to enable cryptographic agility, as well as one-pass AEAD modes suitable for short tags such as AES-GCM-SST. This paper suggests proposals for the upcoming work aiming to modernize the set of NIST standardized encryption modes.

Introduction

NIST's standardized encryption modes have been extremely successful and important for securing data in transit and data at rest. NIST's current selection is however starting to show its age. NIST lacks an approved wide block tweakable cipher such as Adiantum, appropriate for length-preserving encryption, AEAD modes hardened against nonce misuse such as AES-SIV and AES-GCM-SIV, AEAD modes suitable for use with random nonces such as AEGIS-256, high-performance AEAD modes such as AEGIS, AEAD modes suitable for long plaintexts such as AEGIS, an alternative to AES to enable cryptographic agility, as well as one-pass AEAD modes suitable for short tags such as AES-GCM-SST. This paper suggests proposals for the upcoming work aiming to modernize the set of NIST standardized encryption modes.

Based on the discussion in this paper we think NIST should standardize AES-SIV, AES-GCM-SIV, AES-GCM-SST, AEGIS, Rijndael with 256-bit blocks, a tweakable wide encryption scheme, and an AEAD mode based on Keccak. AEGIS alone provides many of the important properties missing from NIST's current set of standardized encryption modes.

Proposals for Standardization of New Encryption Schemes

Below are our proposals for standardization of new encryption schemes. The proposals are not in any particular order:

- **High-performance AEAD schemes.** While AES-GCM [1] has acceptable performance reaching around 35 Gbps on most modern commodity CPUs, new modes of operations of the AES round function significantly outperforms AES-GCM. AEGIS [2–4] can reach 350 Gbps

on CPUs with vector AES and AVX-512 instructions. AEGIS was a winner in the CEASAR competition and IETF is planning to publish AEGIS as an RFC. AEGIS has already been assigned code points for use in TLS, DTLS, and QUIC. Rocca-S [5] is another interesting algorithm, a benefit compared to AEGIS is that the amount of parallelism does not have to be negotiated and agreed upon between endpoints. As traffic volumes continue to grow and NIST's zero trust requirements include mandatory encryption of all data, high performance AEAD schemes are very important. We think NIST should standardize AEGIS.

- **Fully committing AEAD scheme.** It seems to be relatively common for systems to incorrectly assume that AEADs provide key commitment. AES-GCM [1] does e.g., not provide key commitment as an attacker can easily generate several keys that successfully decrypts the same ciphertext. Instead of just looking at key commitment, we think NIST should standardize a fully committing AEAD scheme. Many people likely expect AEAD schemes to be fully committing. In a fully committing AEAD scheme it is infeasible for an attacker to find two different (key, nonce, associated data)-tuples that successfully decrypt the same ciphertext [6]. The security level is half the tag length. AES-GCM can be modified at minimal cost to be fully committing [7]. Another fully committing AEAD scheme is AEGIS [2].

- **Key wrap mode with provable security.** AES-KW and AES-KWP [8] are acceptable but have several significant limitations. They have no security proofs, only support 64-bit tags, and do not support associated data. AES-KW only supports certain key lengths and AES-KWP has message expansion due to padding. AES-SIV [9–10] has a security proof, 128-bit tags, supports associated data, supports all key lengths, and does not require any padding. IETF is planning to add AES-SIV to the Hybrid Public Key Encryption (HPKE) [11]. We think NIST should standardize AES-SIV.

- **Nonce misuse resistant AEAD schemes.** Nonce reuse in AES-GCM [1] has catastrophic consequences as not only confidentiality but also integrity is lost. We think NIST should standardize a nonce misuse resistant AEAD scheme where nonce reuse only discloses whether the messages were equal or not. One nonce misuse resistant AEAD scheme is AES-GCM-SIV [12–13]. AES-GCM-SIV is supported in BoringSSL and benchmarks show that encryption runs at 70% the speed of AES-GCM and decryption is just as fast [14]. Another nonce misuse resistant scheme is AES-SIV [9]. We think NIST should standardize AES-GCM-SIV.

- **AEAD modes suitable for short tags**. 32-bit tags are standard in most radio link layers including 5G [15], 64-bit tags are very common in transport and application layers of the Internet of Things, and 32-, 64-, and 80-bit tags are common in media-encryption applications. Audio packets are small, numerous, and ephemeral, so on the one hand, they are very sensitive in percentage terms to crypto overhead, and on the other hand, forgery of individual packets is not a big concern. Due to its weaknesses, GCM is typically not used with short tags. The result is decreased performance from larger than needed tags [16], or decreased performance from using much slower constructions such as AES-CTR combined with HMAC [17–18]. Short tags are also useful to protect packets transporting a signed payload such as a firmware update. Galois Counter Mode with Secure Short Tags (GCM-SST) makes small theoretical proven changes to GCM to enable forgery probabilities close to ideal. We think NIST should standardize AES-GCM-SST.

- **AEAD schemes with better confidentiality.** The confidentiality of AES-GCM and AES-CCM are significantly limited by the 128-bit block size of AES and the birthday bound. The birthday bound means that the confidentiality advantage for an attacker is $\lesssim \sigma^2/2^{129}$, where σ is the number of encrypted 128-bit chunks. This means that in practical applications the confidentiality is far below 128-bit security even if frequent rekeying is mandated such as in TLS 1.3. As shown by the Sweet32 attack [19], distinguishing attacks on block ciphers can be practically exploitable. AEGIS has a much better confidentiality advantage of $\lesssim 2^{-128}$ [20]. Another simple way to get much stronger confidentiality would be to standardize Rijndael with 256-bit blocks. A 256-bit block cipher in normal modes of operation has a confidentiality advantage of $\lesssim \sigma^2/2^{259}$, where σ is the number of encrypted 128-bit chunks.

- **AEAD schemes suitable for use with random nonces.** AES-GCM [1] is not suitable for use with random nonces. If r random nonces are used with the same key, the collision probability for AES-GCM is $\approx r^2 / 2^{97}$ where a collision breaks both confidentiality and integrity. As an attacker can test r nonces for collisions with work r, the security of AES-GCM with random nonces is only $\approx 2^{97} / r$. We think NIST should standardize a AEAD mode suitable for use with random nonces. Such a scheme could either have large nonces or be nonce misuse resistant. One algorithm with large nonces based on the AES round function is AEGIS-256 [2] which uses a 256-bit nonce. With a 256-bit nonce, the security with random nonces is $\approx 2^{257} / r$. If NIST standardized Rijndael with 256-bit blocks, common modes of operation would accept 224-bit nonces instead of just 96 bits. Other suitable algorithms are AES-SIV and AES-GCM-SIV which are designed to be used with random nonces.

- **AEAD modes suitable for long plaintexts.** AES-GCM [1] only supports encryption of plaintexts shorter than 64 GiB and AES-CCM [21] with $q = 3$ only supports encryption of plaintexts shorter than 16 MiB. While this limit can be overcome by splitting up the plaintext into smaller parts, NIST should have an approved mode supporting longer plaintexts. AEGIS [2] supports plaintexts of up to 2 EiB (2^{31} GiB) which is enough for all current use cases. We think NIST should standardize AEGIS.

- **Tweakable wide encryption**. NIST has specified several non-authenticated encryption modes (ECB, CBC, CFB, OFB, CTR, XTS) in [21–22]. ECB and CBC causes message expansion and the only reason to ever use a non-authenticated encryption mode is if message expansion cannot be accepted. ECB and XTS offers very weak confidentiality even against passive attackers. All of the modes have very limited error propagation, an attacker flipping 1 bit in the ciphertext only affects 1–129 bits in the plaintext. The gold standard for encryption without message expansion is tweakable wide encryption. Such a scheme works like a Strong Pseudo Random Permutation (SPRP) with a block size equal to the message size. A NIST approved tweakable wide encryption scheme could replace all the modes specified in [21–22] and would significantly improve the confidentiality and security against data manipulation in many applications. We have not compared all suggested constructions for tweakable wide encryption, but we find the HBSH (hash, block cipher, stream cipher, hash) construction in Adiantum [23] compelling. A version of HBSH using NIST approved primitives could use GHASH, POLYVAL, Ascon, or Keccak as the hash function, AES as the block cipher, and AES-CTR, Ascon, or Keccak as the stream cipher. Adiantum is included in the Linux kernel since version 5.0 and in Android since version 10.

- **An alternative to AES to enable cryptographic agility.** Cryptographic agility is the ability to switch between cryptographic primitives without the need to modify or replace the surrounding infrastructure. The importance of cryptographic agility has been emphasized by several US agencies [24–26]. A necessity for cryptographic agility is to have a cryptographic primitive to switch to. With the deprecation of Triple DES, NIST does not have a standardized alternative to AES to be used in the event that AES would be broken. Ascon is not recommended as a general replacement for AES and standardizing new algorithms takes many years. NIST has previously discussed standardization of an AEAD mode based on Keccak [27–28]. We think NIST should standardize an AEAD mode of Keccak to enable cryptographic agility.

Summary and Conclusions

NIST's standardized encryption modes has been extremely successful but the current selection is starting to show its age. Based on the above discussion we think NIST should standardize AES-SIV, AES-GCM-SIV, AES-GCM-SST, AEGIS, Rijndael with 256-bit blocks, a tweakable wide encryption scheme, and an AEAD mode based on Keccak. AEGIS alone provides many of the important properties missing from NIST's current set of standardized encryption modes.

References

[1] NIST SP 800-38D, "Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC"
https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-38d.pdf

[2] IETF, "The AEGIS Family of Authenticated Encryption Algorithms"
https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-aegis-aead

[3] Wu, Preneel, "AEGIS: A Fast Authenticated Encryption Algorithm (v1.1)"
https://competitions.cr.yp.to/round3/aegisv11.pdf

[4] Denis, "Adding more parallelism to the AEGIS authenticated encryption algorithms"
https://eprint.iacr.org/2023/523.pdf

[5] Nakano, Fukushima, Isobe, "Encryption algorithm Rocca-S"
https://datatracker.ietf.org/doc/html/draft-nakano-rocca-s

[6] Chan, Rogaway, "On Committing Authenticated-Encryption"
https://eprint.iacr.org/2022/1260.pdf

[7] Bellare, Hoang, "Efficient Schemes for Committing Authenticated Encryption"
https://eprint.iacr.org/2022/268.pdf

[8] NIST SP 800-38F, “Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping”
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf

[9] IETF RFC 5297, “Synthetic Initialization Vector (SIV) Authenticated Encryption Using the Advanced Encryption Standard (AES)”
https://www.rfc-editor.org/rfc/rfc5297

[10] Rogaway, Shrimpton, “Deterministic Authenticated-Encryption A Provable-Security Treatment of the Key-Wrap Problem”
https://eprint.iacr.org/2006/221.pdf

[11] IETF, “Deterministic Nonce-less Hybrid Public Key Encryption”
https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-dnhpke

[12] IETF RFC 8452, “AES-GCM-SIV: Nonce Misuse-Resistant Authenticated Encryption”
https://www.rfc-editor.org/rfc/rfc8452

[13] Gueron, Langley, Lindell, “AES-GCM-SIV: Specification and Analysis”
https://eprint.iacr.org/2017/168.pdf

[14] Denis, “BoringSSL AEADs comparison”
https://raw.githubusercontent.com/jedisct1/openssl-family-bench/master/img/boring-aeads.png

[15] 3GPP TS 33.501, “Security architecture and procedures for 5G System”
https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3169

[16] IETF, “Media Over QUIC”
https://datatracker.ietf.org/wg/moq/about/

[17] Baugher, McGrew, Näslund, Carrara, Norrman, "The Secure Real-time Transport Protocol (SRTP)",
https://www.rfc-editor.org/rfc/rfc3711

[18] Omara, Uberti, Murillo, Barnes, Fablet, "Secure Frame (SFrame)"
https://datatracker.ietf.org/doc/html/draft-ietf-sframe-enc

[19] Bhargavan, Leurent, “On the Practical (In-)Security of 64-bit Block Ciphers”
https://sweet32.info/SWEET32_CCS16.pdf

[20] Eichlseder, Nageler, Primas, “Analyzing the Linear Keystream Biases in AEGIS”
https://tosc.iacr.org/index.php/ToSC/article/view/8468/8034

[21] NIST SP 800-38A, "Recommendation for Block Cipher Modes of Operation: Methods and Techniques"
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf

[22] NIST SP 800-38E, "Recommendation for Block Cipher Modes of Operation: the XTS-AES Mode for Confidentiality on Storage Devices"
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38e.pdf

[23] Crowley, Bigger, "Adiantum: length-preserving encryption for entry-level processors"
https://eprint.iacr.org/2018/720.pdf

[24] Homeland Security, "Cryptographic Agility"
https://www.dhs.gov/sites/default/files/2022-05/22_0512_plcy_2966-01_cryptographic-agility-infographic.pdf

[25] The White House, "National Security Memorandum on Promoting United States Leadership in Quantum Computing While Mitigating Risks to Vulnerable Cryptographic Systems"
https://www.whitehouse.gov/briefing-room/statements-releases/2022/05/04/national-security-memorandum-on-promoting-united-states-leadership-in-quantum-computing-while-mitigating-risks-to-vulnerable-cryptographic-systems/

[26] NISTIR 8105, "Report on Post-Quantum Cryptography"
https://nvlpubs.nist.gov/nistpubs/ir/2016/nist.ir.8105.pdf

[27] FIPS 202, "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions"
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[28] NIST SP 800-185, "SHA-3 Derived Functions: cSHAKE, KMAC, TupleHash and ParallelHash"
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-185.pdf