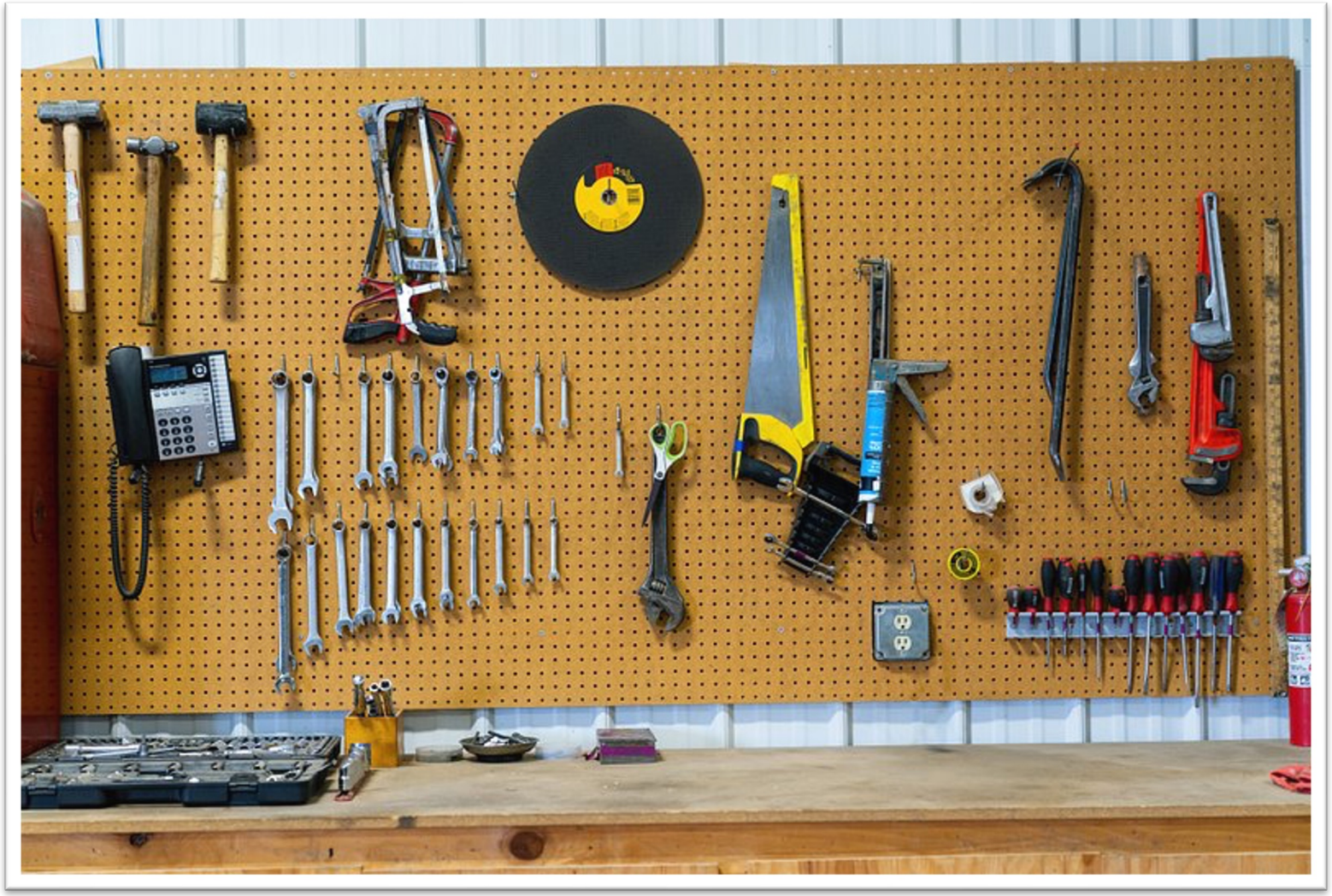
Proposals for Standardization of Encryption Schemes

John Preuß Mattsson, Ben Smeets, Erik Thormarker
Ericsson

The Need for a Well-Assorted Encryption Toolbox

- NIST's standardized encryption schemes have achieved remarkable global success and play a crucial role in securing data in transit and data at rest.
- The current selection of tools is however starting to show its age. Some schemes are broken, some are no longer state-of-art, and some tools are missing.
- We propose updates designed to bring NIST standardized encryption schemes up to date and in line with modern requirements and use cases.



Proposals for Standardization of Encryption Schemes

High-performance AEAD schemes. The need for speed.

- New modes of operations of the AES round function significantly outperforms AES-GCM.
- AEGIS can reach 350 Gbps on CPUs with vector AES and AVX-512 instructions. AEGIS was the winner in the CEASAR competition and has already been assigned code points for use in TLS, DTLS, and QUIC.
- A benefit with Rocca-S compared to AEGIS is that the amount of parallelism does not need to be agreed between the 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. Incorrectly assuming AEADs provide key commitment.

- Many people likely believe that AEAD schemes are fully committing.
- AES-GCM can easily be modified to be fully committing. Another fully committing AEAD scheme is AEGIS.
- We think NIST should standardize a fully committing AEAD.

Proposals for Standardization of Encryption Schemes

Key wrap mode with provable security. AES-KW and AES-KWP have several significant limitations.

- AES-KW and AES-KWP 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 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).
- We think NIST should standardize AES-SIV.

Nonce misuse resistant AEAD schemes. Nonce reuse in AES-GCM has catastrophic consequences.

- Nonce reuse in AES-GCM 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.
- The nonce misuse resistant AES-GCM-SIV is supported in BoringSSL and benchmarks show that encryption runs at 70% the speed of AES-GCM and that decryption is just as fast.
- We think NIST should standardize AES-GCM-SIV

Proposals for Standardization of Encryption Schemes

AEAD schemes suitable for use with random nonces. AES-GCM is not suitable for use with random nonces.

- If r random 96-bit 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 an AEAD mode suitable for use with random nonces. Such a scheme could either have large nonces or be nonce misuse resistant. With n-bit nonces the security is $\approx 2^{n+1}/r$.
- AEGIS-256 uses a 256-bit nonce. And if NIST standardized Rijndael with 256-bit blocks, common modes of operation would accept 224-bit nonces instead of just 96 bits.

An alternative to AES to enable cryptographic agility. The need for a backup algorithm.

- The importance of cryptographic agility has been emphasized by several US agencies.
- A necessity for cryptographic agility is to have a cryptographic primitive to switch to.
- With the deprecation of 3DES, NIST have no alternative to AES in the event that AES would be broken. Ascon is not recommended as a general replacement for AES and standardizing a new algorithm takes many years.
- We think NIST should standardize an AEAD mode of Keccak to enable cryptographic agility.

Proposals for Standardization of Encryption Schemes

AEAD schemes with better confidentiality. The confidentiality of AES-GCM/CCM is far below 128-bit security.

- 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.
- As shown by the Sweet32 attack, distinguishing attacks on block ciphers can be practically exploitable.
- AEGIS has a much better confidentiality advantage of $\lesssim 1/2^{128}$.
- Rijndael with 256-bit blocks 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.
- We think NIST should standardize encryption schemes with better confidentiality.

AEAD modes suitable for long plaintexts. AES-GCM only supports encryption of plaintexts shorter than 64 GiB.

- AES-CCM with $q = 3$ only supports encryption of plaintexts shorter than 16 MiB
- AEGIS supports plaintexts of up to 2 EiB (2^{31} GiB) which is enough for all current use cases.
- We think NIST should standardize AEGIS.

Proposals for Standardization of Encryption Schemes

Tweakable wide encryption. ECB, CBC, CFB, OFB, CTR, XTS have several significant limitations.

- ECB and CBC have message expansion and the only reason to ever use non-authenticated is if message expansion cannot be accepted. ECB and XTS offers very weak confidentiality even against passive attackers.
- All of the NIST IND-CPA modes have very limited error propagation, an attacker flipping 1 bit in the ciphertext only affects 1–129 bits in the plaintext.
- A NIST approved tweakable wide encryption scheme could potentially replace all of NIST's non-authenticated encryption modes and would significantly improve the confidentiality and security against data manipulation in many applications. We think NIST should standardize a tweakable wide encryption scheme.
- Adiantum/HBSH is included in the Linux kernel since version 5.0 and in Android since version 10. Performance is better than ECB on many platforms.
- 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.

AEAD modes suitable for short tags. 32-, 64-, and 80-bit tags are common in many use cases.

- See Galois Counter Mode with Secure Short Tags (GCM-SST) presentation.
- We think NIST should standardize a fast scheme with secure short tags.

Summary

— NIST lacks a wide block cipher appropriate for length-preserving encryption, AEAD modes hardened against nonce misuse, AEAD modes suitable for use with random nonces, high-performance AEAD modes, AEAD modes suitable for long plaintexts, one-pass AEAD modes suitable for short tags, and an alternative to AES to enable cryptographic agility.

— 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.**

Summary by Nick Youngson CC BY-SA 3.0 Alpha Stock Images

https://www.ericsson.com/en/security