ERICSSON

Date: October 29, 2024

Ericsson AB
Group Function Technology
SE-164 80 Stockholm
SWEDEN

Comments on SP 800-131A Rev. 3 (Initial Public Draft)

Dear NIST,

Thanks for your continuous efforts to produce well-written, user-friendly, and open-access security documents. We welcome NIST's plans to revise SP 800-131A, including the retirement of outdated algorithms like ECB, DSA, SHA-1, and 224-bit hash functions.

- We are not aware of any current or past deployments utilizing 224-bit hash functions.

- We welcome the statement that AES-128 will remain secure for the foreseeable future. This effectively reflects the current state of knowledge, summarized in e.g., [1]. We hope NIST will use the same formulation in other documents.

  Given the existence of questions like Q1 in [2], we think NIST needs to provide an explicit statement also for other algorithms than AES. Readers should understand that all algorithms with a security strength of at least 128 bits, such as SHA-256, SHA3-256, HMAC-SHA-256, KMAC128, and Ascon will remain secure for the foreseeable future. Appendix A is not clear enough.

- We suggest that NIST disallow the use of ECB for all use cases and update NIST specifications where "ECB mode" remains acceptable. For example, one sentence in NIST SP 800-73pt2-5 could be revised to state: "The 16-byte IV SHALL be generated by encrypting the encryption counter with SK_{ENC} using the AES Cipher() function" with a reference to FIPS 197. We do not consider the application of the AES Cipher() function to a single block as a mode of operation.

  We have encountered individuals who mistakenly believe that ECB is safe to use for everything because QUIC and DTLS 1.3 use "ECB". This misconception is highly dangerous, and we fully support NIST's position that using ECB for protecting data constitutes a severe security vulnerability [3].

- The document specifies that encryption using TDEA is disallowed, while decryption is allowed for legacy use. To enhance clarity, we recommend explicitly stating that the use of TDEA for confidentiality protection of data in storage is prohibited, as the encryption may have happened in the past. Data still requiring confidentiality protection must be re-encrypted using AES. Similar considerations apply to stored data with RSA-1024 and SHA-1 signatures, which might need to be re-signed. We think NIST should provide guidance on reprotection of stored data. Reprotection will be necessary also in the transition to quantum-resistant algorithms.

- *"RSA: RSA keys are generated with respect to a modulus n, which is used to determine the security strength that can be provided by a digital signature. The RSA algorithm for digital signatures is specified in [RFC 8017], and guidance for use is provided in FIPS 186."*

  There are several RSA algorithms for digital signatures and FIPS 186-5 provides more than guidance. We suggest:

  "RSA-based Digital Signatures (RSASSA-PKCS1-v1_5 and RSASSA-PSS): RSA keys are generated with respect to a modulus n, which is used to determine the security strength that can be provided by a digital signature. RSASSA-PKCS1-v1_5 and RSASSA-PSS are specified in [RFC 8017], and further requirements are provided in FIPS 186."

- *"ECDSA and EdDSA signature generation providing at least 128 bits of security is acceptable. These signatures shall be generated using elliptic curves and private keys such that len(n) ≥ 256"*

  This is inconsistent and should be changed to make it clear that Ed25519 is acceptable. SP 800-186 correctly states that Edwards25519, with len(n) ≈ 252, offers a security strength of 128 bits. When considering the actual number of low-level operations, Ed25519 provides stronger security than AES-128 against classical computers. One solution would be to remove all mentions of len(n), as it is overly technical and unnecessary, given that SP 800-186 already lists the security strength of all relevant curves.

- We recommend that NIST allow key agreement using Curve25519 and Curve448. Currently, Curve25519 is used in the vast majority of TLS, DTLS, QUIC, and SSH connections, and this is expected to continue after the transition to quantum-resistant cryptography, as X25519MLKEM768 is anticipated to dominate future implementations. Ericsson is planning to transition as much as possible to Curve25519 and Curve448 during this shift.

  While there is nothing inherently wrong with NIST P-curves or Brainpool, Curve25519 and Curve448 offer superior performance, encoding efficiency, and robustness. A significant issue in consumer and industry products is the lack of public-key validation in some implementations. Given the widespread deployment and superior properties, it would make a lot of sense for NIST to approve Curve25519 and Curve448 for key agreement.

John Preuß Mattsson,
Expert Cryptographic Algorithms and Security Protocols

[1] IETF Statement on Quantum Safe Cryptographic Protocol Inventory
https://datatracker.ietf.org/liaison/1942/

[2] 3GPP Statement on PQC Migration
https://www.3gpp.org/ftp/tsg_sa/WG3_Security/TSGS3_118_Hyderabad/docs/S3-244307.zip

[3] NIST, "Announcement of Proposal to Revise Special Publication 800-38A"
https://csrc.nist.gov/news/2022/proposal-to-revise-sp-800-38a