ERICSSON

Date: February 22, 2023

Ericsson AB
Group Function Technology
SE-164 80 Stockholm
SWEDEN

Comments on SP 800-132, Recommendation for Password-Based Key Derivation Part 1: Storage Applications

Dear NIST,

Thanks for your continuous efforts to produce well-written open-access security documents.

Please find below our comments on SP 800-132:

- We think SP 800-132 should be expanded with algorithms approved for hashing a password or passphrase for safe storage for authentication purposes. SP 800-63B [1] already refers to SP 800-132 for this use case.

- To mitigate against password cracking on GPUs, FPGAs, and ASICs we think SP 800-132 should be expanded with one or more NIST approved side-channel resistant memory-hard or cache-hard password-based key derivation function such as Argon2id [2], Balloon [3], or bscrypt [4]. Argon2id is the recommendation from the Open Worldwide Application Security Project (OWASP) [5] but does not use a NIST approved hash function. Balloon is suggested by SP 800-63B [1]. We think SP 800-132 should recommend the use of memory-hard or cache-hard algorithms instead of PBKDF2 [6] for all use cases.

- "*A minimum iteration count of 1,000 is recommended.*"

  This should be significantly increased. The current recommendation from OWASP [5] is to use an iteration count of 600,000 for PBKDF2 with SHA-256 and an iteration count of 1,300,000 for PBKDF2 with SHA-1.

- "*such as one of the modes that is defined in [3,4]*'

  In addition to AES-GCM and AES-CCM, we suggest to also add a reference to SP 800-38F [7] that defines AES-KW and AES-KWP.

- *"The following algorithm for the derivation of MKs from passwords is based on an algorithm specified in [6], where it was specified as PBKDF2 and used HMAC [1] with SHA-1 as a PRF. This Recommendation approves PBKDF2 as the PBKDF using HMAC with any approved hash function as the PRF."*

  RFC 2898 has been obsoleted by RFC 8018 [6] which introduces SHA-2 as an alternative.

The differences (if any) between the algorithm in Section 5.3 of SP 800-132 and the PBKDF2 algorithm specified in RFC 8018 should be described. We think NIST should remove the abbreviation PBKDF and only talk about PBKDF2. The abbreviation PBKDF is too associated with the PKCS #5 algorithms to be suitable as a general term.

- "*Dictionary attacks aim to recover passwords by trying the most-likely strings, such as the words in a dictionary. These attacks are less likely to succeed against passwords that include random combinations of upper/lowercase letters and numeric values. Such passwords can only be recovered using inefficient brute force attacks that try all possible passwords.*"

  We think this should be rewritten to better reflect modern methods for password cracking. Attackers typically generate likely passwords based on previous breach corpuses, details about the target, and sophisticated models on how humans create passwords.

- SP 800-132 should be aligned with SP 800-63B [1]:

  - SP 800-63B states that PBKDF2 as specified in SP 800-132 is a suitable algorithm for hashing a password for safe storage for authentication purposes. SP 800-132 states that PBKDF2 shall not be used for any other purposed than to generate data protection keys or intermediate keys to protect data protection keys. We suggest that the update to SP 800-132 approves the use of PBKDF2 for hashing a password or passphrase for safe storage for authentication purposes.

  - SP 800-63B states that the salt shall be at least 32 bits. SP 800-132 states that the randomly-generated portion of the salt shall be at least 128 bits.

  - In addition to a salt, SP 800-63B also recommends the use of a pepper, i.e., using an additional secret salt value known only to the verifier and stored separately (e.g., in an HSM). We think SP 800-132 should recommend peppering for all use cases. The pepper could be unlocked by multifactor authentication.

  - SP 800-63B requires that when processing requests to establish and change memorized secrets, the prospective secrets shall be compared against a list that contains values known to be commonly-used, expected, or compromised. We think SP 800-132 should require this as well. The U.S. Department of the Interior report [8] showed that between 20 and 40 percent of their passwords could be easily cracked.

- SP 800-132 and 800-63B should add some discussion of when the mechanisms in SP 800-132 and 800-63B are acceptable to use instead of a password-authenticated key exchange (PAKE) such as the augmented PAKE OPAQUE [9] and the balanced PAKE CPace [10] that are recommended by the Crypto Forum Research Group (CFRG). As PAKEs are a better solution in many situations it would be good to have NIST approved PAKEs.

Best Regards,
John Preuß Mattsson,
Expert Cryptographic Algorithms and Security Protocols

[1] SP 800-63B, “Digital Identity Guidelines Authentication and Lifecycle Management” https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63b.pdf

[2] RFC 9106, “Argon2 Memory-Hard Function for Password Hashing and Proof-of-Work Applications”
https://datatracker.ietf.org/doc/html/rfc9106

[3] Boneh et al., “Balloon Hashing: A Memory-Hard Function Providing Provable Protection Against Sequential Attacks”
https://eprint.iacr.org/2016/027

[4] Steve Thomas, “bscrypt a cache hard password hash/KDF”
https://github.com/Sc00bz/bscrypt

[5] OWASP, “Password Storage Cheat Sheet”
https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html

[6] RFC 8018, “PKCS #5: Password-Based Cryptography Specification Version 2.1”
https://datatracker.ietf.org/doc/html/rfc8018

[7] 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

[8] U.S. Department of the Interior, “P@s$w0rds at the U.S. Department of the Interior: Easily Cracked Passwords, Lack of Multifactor Authentication, and Other Failures Put Critical DOI Systems at Risk”
https://www.doioig.gov/sites/default/files/2021-migration/Final%20Inspection%20Report_DOI%20Password_Public.pdf

[9] IRTF, “The OPAQUE Asymmetric PAKE Protocol”
https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-opaque

[10] IRTF, “CPace, a balanced composable PAKE”
https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-cpace