Public Comments on Special Publication (SP) 800-38A, *Recommendation for Block Cipher Modes of Operation: Methods and Techniques*

Comment period: March 24, 2022 - April 25, 2022

On March 24, 2022, NIST's Crypto Publication Review Board announced a proposal to revise Special Publication (SP) 800-38A, *Recommendation for Block Cipher Modes of Operation: Methods and Techniques*, and to incorporate 800-38A Addendum, *Three Variants of Ciphertext Stealing for CBC Mode* into the revision.

The public comments that NIST received on the proposal are collected below. More information about this review is available from NIST's Crypto Publication Review Project site.

LIST OF COMMENTS

1. Danny Niu, March 27, 2022

While it's clear **why** we need to revise SP-800-38A, I think setting a goal (for what we want the revised version to provide) is equally important.

It is clear from industry practice, that all encryption must be accompanied with authentication, and this makes it clear that confidentiality-only modes are of little use.

Specifying confidentiality-only modes have only 1 practical use:
to serve as reference for building block for authenticated encryption modes. I believe such a reference should contain:

1. A definition for each mode,
2. Performance/efficiency characteristics, 3. Requirements for nonce/IV, and thus therefore:
4. Recommended usage.

I've left "security properties" out of this list because it's not meaningful to talk about encryption without authentication.
For "recommended usage", it should make it clear that all confidentiality-only modes are not to be used directly for encrypting data, but that all of them must be incorperated into an authenticated encryption construct. The factors influencing the choice of usage recommendation should focus mostly on balancing point 2 and 3.

2. Ashley R. Thomas, April 3, 2022

Hi NIST Crypto Review Board,

Regarding your request for comments on potential revisions to SP 800-38A, this email is a request that block cipher mode guidance generally, including any guidance within revisions to SP 800-38A, clarify whether or not an "initial counter block" used with Counter Mode should be kept secret. I did not see such clarifications in SP 800-38A, where subsequent Internet searches seemed to clarify that the exchange of an "initial counter block" can be handled similarly to that of an Initialization Vector (IV). Since SP 800-38A does a nice job of clarifying the non-secret nature of an otherwise properly generated IV, clarification on secrecy and related exchanges/handling of initial counter block values would be a nice fit for SP 800-38A, if not NIST Counter Mode guidance generally.

Kindest regards,
Ashley R. Thomas

3. Aleksandr V. Tereschenko (Intel Product Assurance and Security Cryptology Team), April 25, 2022

Intel Product Assurance and Security (IPAS) Cryptology team comments on SP 800-38A Update Decision Proposal

In general, IPAS Cryptology team supports revision of SP 800-38A as described in the Announcement [1]. Given that high-level description it is hard to comment in more detail and we will be glad to review any specific Drafts that come out of that update if that kind of review is planned by NIST.

One particular item we'd like to mention though is Rogaway's critique of the nonce-based CBC/CFB IV generation method described in the fourth paragraph of Appendix C (called "the first method" there). As Rogaway shows in his work [2], page 37, this approach to generating the IV is not CPA-secure and therefore should probably be removed or reworked.

[1] https://csrc.nist.gov/News/2022/proposal-to-revise-sp-800-38a

[2] Rogaway, P. Evaluation of Some Blockcipher Modes of Operation 2011. Available at https://web.cs.ucdavis.edu/~rogaway/papers/modes.pdf

---

Alexander T., on behalf of the team (corresponding author)

4. Neils Ferguson, Mike Grimm (Microsoft Corporation), April 25, 2022

Comments to NIST on AES ECB mode

NIST requested public comments on the existing block cipher mode standard SP800-38A. These are the comments of Microsoft. We understand that NIST is considering changing the SP800-38A standards on block cipher modes, particularly in limiting the uses of ECB.

We wholeheartedly agree that ECB is a bad block cipher mode for message encryption. ECB for message encryption has been banned for use in our products starting at our first internal crypto standard back in 2005. However, it is widely used in many other situations where it is perfectly secure, and the best way of doing things. Here are some examples:

- AES-GCM has the best performance of all encryption algorithms, but it requires a unique nonce. One structure used by quite several applications is to generate a fresh AES 256-bit key for each message, encrypt the fresh key in ECB with the channel key, and use the fresh key in AES-GCM to encrypt the actual message (with a fixed nonce). This eliminates the nonce requirement at the cost of 32 bytes ciphertext (the encrypted key) and about 400 clock cycles. In this case, the weakness of ECB has no effect on the overall message encryption security.
- A closely related option is to not encrypt a key for each message, but to pick a random 32-byte string, make it part of the ciphertext, encrypt the string with ECB and use that as the AES-GCM key. This is different from the above scenario in that there is never any decryption, only encryption. ECB mode is used to derive a secret from a shared random value, similar to how AES-CTR-DRBG uses AES.
- There are several applications that implement CTR-mode encryption by creating a buffer of counter values, encrypting it with ECB, and then xorring that into plaintext. This is used by any application whose CTR-mode specification uses a block increment rule that is not implemented inside the crypto libraries.

In short, we have forty years of code development that uses ECB in many, many places. Usage of straight ECB for message encryption has long been banned, but the kind of scenarios listed above are still widely used.

We recommend that NIST be very careful in the wording around the allowed usage of ECB. Banning ECB from actual data encryption is an excellent idea, but if the ban is worded too broadly so that it would cover uses like the examples given above, it would make many currently compliant systems non-compliant. This would cause a large amount of upheaval and cost with zero security benefit.