Padding-based forgeries in the mode XOCB

Jean Liénardy*

Royal military Academy, Avenue de la renaissance 30, 1000, Bruxelles, Belgium

In this note, we identify a minor flaw in the design of the XOCB mode, presented at Eurocrypt '23. This vulnerability enables trivial tag forgeries and arises from the padding applied to messages. We examine the security proof and pinpoint the presence of the flaw within it. Furthermore, we propose a simple fix for this issue, drawing upon the features of OCB3, and discuss the implications of this modification on the proof of security.

1 Introduction

Authenticated encryption with associated data (AEAD) is a key component in modern cryptography, ensuring confidentiality and integrity of transmitted data. One widely-adopted AEAD mode is the Offset Codebook mode (OCB3), developed by Rogaway [1, 2]. While OCB3 has been remarkably efficient and enjoyed considerable success, its security is limited by the birthday-bound constraint.

In this article, we discuss a new mode of operation for AEAD called XOCB, which is a generalisation of OCB3 [3]. XOCB offers rate-1 computation as well as beyond-birthday-bound (BBB) security under the standard pseudorandom assumption of the internal block cipher, provided that the maximum block length is significantly smaller than the birthday bound.

However, our analysis reveals a weakness in the XOCB mode, undermining its security guarantees. In this article, we present a detailed description of the flaw and then discuss the proof of security of XOCB. We finally propose a fix for the flaw.

2 Description of the tag generation in XOCB

In this section, we provide an overview of the tag generation phase of the XOCB mode, which is the only portion of the design relevant to the current article. Throughout this note, E represents an n-bit block cipher, and E_K denotes the permutation obtained when the key K is fixed. The tag $T \in \{0,1\}^n$ is given by

$$T = \Gamma \oplus 3\Delta_3 \oplus E_K(2^m\Delta_1 \oplus \Delta_3) \oplus E_K(2^m\Delta_1 \oplus 2\Delta_3 \oplus \Sigma). \tag{1}$$

In this equation, $\Delta_1 = E_K(N\|00) \oplus E_K(N\|01)$ and $\Delta_3 = E_K(N\|00) \oplus E_K(N\|11)$ are masking values generated from the nonce $N \in \{0,1\}^{n-2}$. As for Γ, it is the result of the hash of the associated data and is not of interest for this note.

However, we are interested in Σ, which we refer to as the checksum and which contains information about the plaintext M. According to the design description, an input $M \in \{0,1\}^*$ is divided into $(M_1, \dots, M_m) \leftarrow M$ with $m = \lceil |M|/n \rceil$, where $|M_i| = n$ for $i = 1, \dots, m-1$ and

*Email address: jean.lienardy@mil.be

$0 < |M_m| \leq n$. The empty plaintext $M = \varepsilon$ is treated differently. This special case results in $(M_1) \leftarrow M$ (thus, with an effective value $m = 1$) with $M_1 = \varepsilon$, $|M_1| = 0$.

The checksum is defined as $\Sigma = \bigoplus_{i=1}^{m} \overline{M_i}$ where the sum runs over the plaintext blocks. In this equation,

$$\overline{M} = \begin{cases} M & \text{if } |M| = 0 \mod n, \\ \mathsf{pad}(M) = M \| 10^{n-(|M| \bmod n)-1} & \text{else} \end{cases}$$

Furthermore, according to the algorithm presented in Fig. 1 of [3] and the provided source code, the empty message $M = \varepsilon$ yields a checksum $\Sigma = 0^n$.

2.1 A trivial forgery

We observe that although the function pad is injective, as noted in [3], this property does not extend to the function that maps X to $\overline{X}$. Indeed, for every X such that $|X| < n$, X and $X' = \mathsf{pad}(X)$ satisfy $X \neq X'$ but $\overline{X} = \overline{X'}$.

Due to the non-injectiveness of $\overline{X}$, and consequently of Σ, we can easily deduce that the two plaintexts X and X' share the same tag under a given pair (N, A) for a given key K, which enables an adversary to forge a valid message.

As an example, consider the message $M = 0$. In this case, $\overline{M} = 0 \| 10^{126}$. An adversary may query the encryption oracle (N, A, M') with $M' = \overline{M} = 010^{126}$ and arbitrary (N, A). Given the output (C', T') of this query, one of the following is a correct forgery: $(N, A, C = 0, T = T')$ or $(N, A, C = 1, T = T')$. The correct forgery depends on which ciphertext C will be decrypted into $M = 0$ (this will occur since C is the result of XORing M with a value that does not depend on C).

As a generalisation, for any M_m with $|M_m| < n$, the query $(N, A, M' = M_1 \| \ldots \| M_{m-1} \| \overline{M_m})$ giving $(C' = C'_1 \| \ldots \| C'_m, T')$ allows one to make the forgery $(N, A, C = C'_1 \| \ldots \| C'_{m-1} \| C_m, T = T')$ with $|C_m| = |M_m|$ which is valid with probability $2^{-|X_m|} > 2^{-n}$, where the probability arises from the decryption of C_m into M_m.

Using only messages with lengths that are multiples of n, a second type of forgery is possible. Given the output (C, T) of the query $(N, A, M = 0^n)$, the following ciphertext-tag pair is valid: (ε, T). This is due to both messages 0^n and ε yielding the checksum $\Sigma = 0^n$.

In both cases, the adversary has a significant advantage (close to one) while having made only one encryption query and one or two decryption queries. These observations thus contradict the security claim of [3].

3 Proof of security

n the paper [3], the XOCB mode is accompanied with a proof of security. This proof is constructed using Patarin's H technique [4] and mirror theory [5].

The proof of the security claim is based on three lemmas. In particular, the third lemma (called Lemma 6) ensures that no "bad event" badC occurs due to collisions in the inputs of π, an idealised blockcipher during the tag generation.

We are specifically interested in the badC_3 event, which occurs (following [3] with $\alpha = 0$) if

there exists a decryption query i and an encryption query j in the final transcript such that

$$2^{m^{(i)}}\Delta_1^{(i)} \oplus \Delta_3^{(i)} = 2^{m^{(j)}}\Delta_1^{(j)} \oplus \Delta_3^{(j)}, \tag{2}$$

$$2^{m^{(i)}}\Delta_1^{(i)} \oplus 2\Delta_3^{(i)} \oplus \Sigma^{(i)} = 2^{m^{(j)}}\Delta_1^{(j)} \oplus 2\Delta_3^{(j)} \oplus \Sigma^{(j)}, \tag{3}$$

$$T^{(i)} \oplus 3\Delta_3^{(i)} \oplus \Gamma^{(i)} = T^{(j)} \oplus 3\Delta_3^{(j)} \oplus \Gamma^{(j)}. \tag{4}$$

In this equation, the superscript indicates that the blocks (masks Δ_1 and Δ_3, tags T, hash value of the associated data Γ, and checksum $\Sigma = \bigoplus_{k=1}^{m} \overline{M_k}$) correspond to either query i or j.

We focus on the case where $N^{(i)} = N^{(j)}$ and $A^{(i)} = A^{(j)}$, resulting in $\Delta_{1/3}^{(i)} = \Delta_{1/3}^{(j)}$ and $\Gamma^{(i)} = \Gamma^{(j)}$. Under these conditions, the $\mathsf{badC3}$ event requires:

$$2^{m^{(i)}}\Delta_1^{(i)} = 2^{m^{(j)}}\Delta_1^{(j)}, \quad \Sigma^{(i)} = \Sigma^{(j)} \quad \text{and } T^{(i)} = T^{(j)}.$$

The first condition implies that the messages have the same length $m^{(i)} = m^{(j)}$. If the queried messages are different, there must exist a $\beta \leq m^{(j)}$ such that $M_\beta^{(i)} \neq M_\beta^{(j)}$. (The case $\beta > m^{(j)}$ considered in [3] can be excluded due to the length condition).

We examine the first subcase where the last blocks $\beta = m^{(i)}$ differ, and $|M_\beta^{(i)}| < n$ is a partial block. In this scenario, we can have $\Sigma^{(i)} = \Sigma^{(j)}$ by choosing $M_\beta^{(i)} \neq M_\beta^{(j)}$ such that $\overline{M_\beta^{(i)}} = M_\beta^{(j)}$, thus contradicting the statement that this sub-event has a probability of 0. Another option is to take $\beta = 1$, $M_\beta^{(i)} = \varepsilon$, and $M_\beta^{(j)} = 0^n$. Consequently, the probability of badC_3 is higher than anticipated, and the security claim is not met.

Our findings show that even "provably secure schemes" may contain errors in their proof of security. Similar situations have occurred with OCB2 [6], GCM [7], recently with OCB3 [8], and numerous other examples (see [6] for more instances). We believe this paper highlights once again the value of thoroughness in crafting and reviewing security proofs to ensure the robustness of cryptographic designs.

4 A fix using features from OCB

In this section, we give a fix to the above-mentioned weakness using features taken from the design of OCB. Indeed, in each generation of OCB, this problem is handled differently.

In both OCB1 and OCB2, the checksum is given by $\bigoplus_{i=1}^{m-1} M_i \oplus C_m\|0^*$. It thus contains some information on the length of the last message as part of the last cipher block given by $C_m \oplus M_m = \mathsf{msb}_{|M_m|}(E_K(\mathsf{len}(M_m) \oplus \tilde{\Delta}))$, therefore preventing the weakness presented here.

The improvement of OCB3 is to reduce the latency of mode by eliminating the need to wait for the ciphertext before computing the final encryption required for the tag. Recast using the notation of [3], the checksum is $\Sigma^{\mathsf{OCB3}} = \bigoplus_{i=1}^{m} \overline{M_i}$. To circumvent the attack presented here, OCB3 design requires to make a difference between the two cases $|M_m| = n$ and $|M_m| < n$. This is done by updating the value of Δ according to $\Delta \leftarrow \Delta \oplus L_*$ *only if* $|M_m| < n$. This ensures that the tag, given in the absence of AD by $T = E_K(\Sigma^{\mathsf{OCB3}} \oplus \Delta)$ will not be subject to the identified security weakness.

In light of this, the most immediate way to fix the XOCB mode would be to update the input value of E_K depending on the length of the message. Following the OCB3 design, we propose the following modification:

$$Q = E_K(0^n \oplus 2^m\Delta_1 \oplus \Delta_3) \oplus 2^m\Delta_1 \oplus \Delta_3 \tag{5}$$

$$T = \Gamma \oplus Q \oplus \begin{cases} E_K(\bigoplus_{i=1}^{m} \overline{M_i} \oplus 2^m\Delta_1 \oplus 2\Delta_3) \oplus 2^m\Delta_1 \oplus 2\Delta_3 & \text{if } |M_m| = n \\ E_K(\bigoplus_{i=1}^{m} \overline{M_i} \oplus 2^m\Delta_1 \oplus 4\Delta_3) \oplus 2^m\Delta_1 \oplus 4\Delta_3 & \text{if } |M_m| < n \end{cases} \tag{6}$$

The presence of $2\Delta_3$ or $4\Delta_3$, depending on the message length will be sufficient to prevent the small weakness of section 2 and the randomness of the newly used value ($4\Delta_3$) will most certainly keep the security claims unchanged. However, a comprehensive re-evaluation of the security proof is necessary to ensure that the modified mode maintains its security guarantees. The later is beyond the scope of this note.

5 Conclusion

In this brief note, we have identified a flaw in the design of the recently proposed XOCB mode. This weakness arises from an issue in the padding applied to messages during the tag generation phase. We highlighted the error made in the security proof and suggested a fix for the mode, drawing inspiration from the design of OCB3. We recommend carrying out a comprehensive security analysis to ascertain the effectiveness of the proposed fix and assess its impact on the overall security of the XOCB mode. This paper serves as a reminder of the importance of the meticulous writing and examination of security proofs, as these are crucial to the reliability of cryptographic schemes.

References

[1] T. Krovetz, P. Rogaway, The software performance of authenticated-encryption modes, in: International Workshop on Fast Software Encryption, Springer, 2011, pp. 306–327.

[2] T. Krovetz, P. Rogaway, The Design and Evolution of OCB, Journal of Cryptology 34 (4) (2021) 1–32.

[3] Z. Bao, S. Hwang, A. Inoue, B. Lee, J. Lee, K. Minematsu, XOCB: Beyond-Birthday-Bound Secure Authenticated Encryption Mode with Rate-One Computation (Full Version), IACR Cryptology ePrint Archive, Paper 2023/253 (2023), `https://eprint.iacr.org/2023/253`

[4] J. Patarin, The "coefficients H" technique. In Selected Areas in Cryptography: 15th International Workshop, SAC 2008, Springer (2009), 328–345.

[5] J. Patarin, Mirror Theory and Cryptography. IACR Cryptology ePrint Archive, Report 2016/702 (2016), `http://eprint.iacr.org/2016/702`

[6] A. Inoue, T. Iwata, K. Minematsu, B. Poettering, Cryptanalysis of OCB2: attacks on authenticity and confidentiality, Journal of Cryptology 33 (4) (2020) 1871–1913.

[7] Breaking and repairing GCM security proofs T. Iwata, K. Ohashi, K. MinematsuK, Breaking and repairing GCM security proofs, in Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, Springer, pp. 31–49.

[8] J. Liénardy, F. Lafitte, A weakness in OCB3 used with short nonces allowing for a break of authenticity and confidentiality, *in publication in* Information Processing Letters (2023)