TinyJAMBU Update

Hongjun Wu and Tao Huang

Division of Mathematical Sciences
Nanyang Technological University
wuhongjun@gmail.com

30 September 2022

1 Overview of the Third-Party Analysis

To the best of our knowledge, we are aware of the following published third-party analysis on TinyJAMBU.

1.1 Differential analysis

In Year 2020, Saha et al. analyzed the security margin of the nonce and associated data of TinyJAMBU against the differential forgery attack [7]. In the TinyJAMBU design, our analysis shows that the differential forgery attack against nonce and associated data succeeds with probability at most 2^{-73}. In [7], it is shown that some NAND gates in the differential forgery attack are not independent, so the forgery attack against nonce and associated data succeeds with probability $2^{-70.64}$.

In the Round 3 submission of TinyJAMBU on 7 May 2021, we tweaked TinyJAMBU to provide large security margin against the forgery attack on the nonce and associated data. In the tweak, the 640-round permutation P_{640} is used to replace the 384-round P_{384}. The differential probability of P_{640} is at most 2^{-83} for any 32-bit input difference, the differential forgery attack on nonce and associated data can succeed with probability much smaller that 2^{-83} since the forgery attack requires 32-bit input difference and 32-bit output difference.

1.2 Linear analysis

There are two papers on the linear analysis of the permutation of TinyJAMBU.

In Year 2020, Saha et al. analyzed the linear bias of the permutation [7]. Their results on the linear bias of the permutation are close to the analysis results presented in the TinyJAMBU report.

In Year 2022, Li et al. extended Matsui's Algorithm 1 to linear hulls and applied it to analyze the last permutation of the tag generation of TinyJAMBU [6].

Their attacks use more than 2^{96} tags to recover 8 bits of the key of TinyJAMBU-128 (and 14 bits of the key of TinyJAMBU-192 and TinyJAMBU-256) if P_{640} is reduced to 387 rounds. So **TinyJAMBU has large security margin against this type of attacks**: 640 rounds of P_{640} is much larger than 387 rounds; and 2^{96} tags is much more than the 2^{47} tags allowed for TinyJAMBU for a single key.

In order to obtain more than 2^{96} tags in the attack, more than 2^{49} related keys are used, and generate 2^{47} tags from each key [6]. It is claimed in [6] that NIST's "recommendations for key generation allow keys with known XOR relations [BRD20, Section 6.3]". However, this claim is incorrect. In NIST's recommendation on key generation [2], it is stated that "Each component key shall be kept secret and shall not be used for any purpose other than the computation of a specific symmetric key K (i.e., a given component key shall not be used to generate more than one key)." It means that any two keys are never generated from the same component key, so any two keys are always independent from each other in NIST's recommended key generation.

1.3 Slide attack on the permutation

In Year 2022, Sibleyras et al. presented the slide attack on the permutation of TinyJAMBU at the NIST Lightweight Cryptography Workshop (it was later published at IWSEC 2022) [8, 9]. It is the standard slide attack on block cipher in which with more than 2^{64} different inputs, it is possible that an input at the first round would be identical to another input at the second round, and this slide property continues for the rest of the rounds, then the key used in the permutation can be recovered.

This slide attack does not affect the security of TinyJAMBU since the information of the 128-bit inputs and 128-bit outputs of the permutation are used in this slide attack, but only 32-bit inputs and 32-bit outputs of the permutation are known to the attacker in TinyJAMBU. We are fully aware of the slide property of the permuation of TinyJAMBU at the design stage, as stated in the security analysis of TinyJAMBU report. We believe that the mode of TinyJAMBU is strong enough to protect TinyJAMBU against the slide property of the permutation.

1.4 Cube attack on the permutation

In Year 2022, Teng et al. presented the cube distinguishing and key recovery attacks on round-reduced TinyJAMBU when the nonce is reused [10]. The distinguishing attack can be applied to 437 rounds of P_{1024}, and the key recovery attack can be applied to 428 rounds of P_{1024}.

In Year 2022, Dutta et al. showed in their preprint paper [5] that the cube distinguishing attacks can be applied to 476 rounds of P_{1024} when weak keys are used. Their cube key recovery attack can be applied to 440 rounds of P_{1024} with the help of MILP model.

Since the round numbers in the cube attacks are much smaller than the 1024 rounds of P_{1024}, **TinyJAMBU has large security margin against the cube attack**. At the design stage, we have analyzed the cube attack, and presented the analysis result in the TinyJABMU report.

1.5 Related-key attack on TinyJAMBU-192/256

In Year 2022, Dunkelman et al. presented the related-key forgery attack on TinyJAMBU-192/256 [4]. The time and data complexity of the forgery are 2^{32} using 2^{10} related-keys for the 256-bit key version, and 2^{42} using 2^{12} related-keys for the 192-bit key version. In these related-key forgery attacks, the key size is larger than the state size, so the difference in the state can be cancelled using the difference in the keys with high chance. **This related-key forgery attack does not apply to TinyJAMBU-128 since the key size is not larger than the state size. To defend against the related-key forgery attack on TinyJAMBU-192/256, our suggestion is to use independent keys in TinyJAMBU.**

Related-key key recovery attacks against the permutations are presented in [4]. These related-key key recovery attacks are not relevant to the security of TinyJAMBU since the information of the 128-bit inputs and 128-bit outputs are needed in the related-key recovery attacks, but only 32-bit inputs and 32-bit outputs of the permutation of TinyJAMBU are known to the attackers.

1.6 Randomness testing of permutation

In Year 2022, Bellini and Huang presented the randomness testing of the finalists at the NIST Lightweight Cryptography Workshop [3]. Based on the randomness testing results, the authors claim that the round numbers of TinyJAMBU are chosen more aggressive comparing to some other candidates. However, **this claim is inaccurate**. In the randomness testing, the authors tested only the permutation of TinyJAMBU, so the 128-bit inputs and 128-bit outputs of the permutation are used in the randomness testing. However, for the 128-bit permutation of TinyJAMBU, the message input is 32-bit, and the keystream output is 32-bit, so **the randomness of TinyJAMBU is much better than the randomness of the permutation**.

2 New Implementations

In Year 2022, the GMU team presented the side-channel resistant implementation of Elephant, TinyJAMBU and Xoodyak at the NIST Lightweight Cryptography Workshop [1]. The first order protected design of TinyJAMBU occupies 1267 LUTs on Xilinx Artix-7 FPGA, more than four times smaller than the implementation of Elephant and Xoodyak. GMU team presented that TinyJaMBU is the most resource and power efficient among these three candidates.

3 Platforms and metrics in which JAMBU performs better than current NIST standards

3.1 Comparing with AES

At LWC 2019, Yu and Aagaard provided the smallest AES cipher core on STMicro's 65nm ASIC which requires an area of 1960 GE [13]. For TinyJAMBU-128, the cipher core requires as small as **1610 GE** when the key is a fixed value and eight rounds are implemented in parallel on UMC Technology 180nm ASIC. It shows that **TinyJABMU could be implemented smaller than AES on ASIC.** (Note that the AES cipher core performs only encryption, while TinyJAMBU performs both encryption and authentication).

3.2 Comparing with AES-GCM

We compare TinyJAMBU with AES-GCM:

1. TinyJAMBU can be implemented in hardware much smaller than AES-GCM. We stated above that TinyJAMBU can be implemented even smaller than AES. AES-GCM is much more expensive to implement than AES since AES-GCM requires 384 extra bits comparing with AES.

2. TinyJAMBU has strong authentication security when nonce is reused. The authentication security of AES-GCM fails completely if nonce is reused.

3. TinyJAMBU has better encryption security than AES-GCM when nonce is reused. The encryption of TinyJAMBU is similar to Cipher Feedback mode; while the encryption of AES-GCM is counter mode.

4. TinyJAMBU can be simply used as Message Authentication Code (MAC) algorithm without using nonce. But AES-GCM still requires nonce when it is used as MAC algorithm.

4 Target applications and use cases for which the candidate is optimized

TinyJAMBU is especially optimized when a key is already available in a lightweight cryptographic hardware device (for example, when there is a fixed secret key in the hardware device). The TinyJAMBU-128 cipher core is only 1610 GE on UMC Technology 180nm ASIC for this type of application. For these lightweight devices, TinyJAMBU is significantly smaller in hardware than all the other candidates.

TinyJAMBU is optimized for the general hardware lightweight cryptographic applications. The state size of TinyJAMBU is only 128 bits, and the round function of TinyJAMBU is very simple.

TinyJAMBU is optimized for the use case of nonce misuse (defense in depth). When nonce is misused, the secret key in TinyJAMBU remains secure, and TinyJAMBU is strong against forgery attack. (But TinyJAMBU does not provide strong encryption security when nonce is misused since the encryption security of TinyJAMBU is similar to that of Cipher Feedback mode.)

It is straightforward to use TinyJAMBU as an MAC algorithm: treat the whole message as associated data, and nonce is not needed.

5 TinyJAMBU Security

The security of TinyJAMBU has the following features:

1. Strong security of the mode against forgery attack
   We used security proof and concrete attack to analyze the security of the mode against forgery attack when nonce is reused. These two different approaches give almost identical forgery advantage ($2^{-16} + 2^{-17} = 2^{-15.5}$) when 2^{48} adaptively chosen message blocks are used.
   The security proof and the concrete attack are provided in Section 6.3 and Section 7.1.2 of the TinyJAMBU report [12], respectively.

2. Strong security of the permutation against differential and linear analysis
   The differential and linear analysis of the permutation are analyzed using the MILP tool Gurobi optimizer, so the analysis results given in the report are obtained in an accurate way.

3. Strong security of the permutation against cube attack
   The message block of the permutation is small (32 bits), so the resistance against cube attack can be analyzed easily on a computer.

4. Design an integrated authenticated encryption algorithm
   When we design TinyJAMBU, our approach is not to simply insert a strong block cipher into a strong authenticated encryption mode. In the TinyJAMBU mode, the message block size is only 32 bits, so we design a dedicated keyed permutation to reduce the redundant operation in the overall algorithm. For example, the mode of TinyJAMBU prevents the slide attack from being exploited, so we do not need to implement operations in the permutation to resist the slide attack.
   (In the CAESAR competition, we used block cipher SIMON (128-bit block size) in the design of JAMBU. SIMON is probably the most lightweight hardware block cipher. However, we realized that when a strong block cipher is used in JAMBU, it is impossible to design an optimized authenticated encryption algorithm since some operations of the block cipher are redundant in the overall JAMBU algorithm.)

References

[1] Abubakr Abdulgadir, Richard Haeussler, Sammy Lin, Jens-Peter Kaps, and Kris Gaj. Side-Channel Resistant Implementations of Three Finalists of the NIST Lightweight Cryptography Standardization Process: Elephant, TinyJAMBU, and Xoodyak. NIST Lightweight Cryptography Workshop, May 2022. Available at `https://csrc.nist.gov/csrc/media/Events/2022/lightweight-cryptography-workshop-2022/documents/papers/side-channel-resistant-implementations-of-three-finalists-of-the-nist-lwc-standardizat` `pdf` .

[2] Elaine Barker, Allen Roginsky, Richard Davis. Recommendation for Cryptographic Key Generation. NIST Special Publication 800-133, Revision 2, June 2020.

[3] Emanuele Bellini, Yun Ju Huang. Randomness Testing of the NIST Light Weight Cipher Finalist Candidates. NIST Lightweight Cryptography Workshop, May 2022. The presentation slides is available at `https://csrc.nist.gov/Presentations/2022/randomness-testing-of-the-nist-light-weight-cipher` .

[4] Orr Dunkelman, Eran Lambooij, Shibam Ghosh. Practical Related-Key Forgery Attacks on the Full TinyJAMBU-192/256. IACR ePrint. Available at `https://eprint.iacr.org/2022/1122` .

[5] Pranjal Dutta, Mahesh Sreekumar Rajasree, Santanu Sarkar. Weak-keys and key-recovery attack for TinyJAMBU. Preprint at Research Square. Available at `https://assets.researchsquare.com/files/rs-1646044/v1_covered.pdf?c=1652897480` .

[6] Muzhou Li, Nicky Mouha, Ling Sun and Meiqin Wang. IACR Transactions on Symmetric Cryptography., Vol. 2022, No. 2, pp. 161-200.

[7] Dhiman Saha, Yu Sasaki, Danping Shi, Ferdinand Sibleyras, Siwei Sun and Yingjie Zhang. On the Security Margin of TinyJAMBU with Refined Differential and Linear Cryptanalysis. IACR Transaction on Symmetric Cryptography, Vol. 2020, No. 3, pp. 152-174. Available at `https://eprint.iacr.org/2020/1045`

[8] Ferdinand Sibleyras, Yu Sasaki, Yosuke Todo, Akinori Hosoyamada, Kan Yasuda. Birthday-Bound Slide Attacks on TinyJAMBU's Keyed-Permutations for All Key Sizes. NIST Lightweight Cryptography Workshop, May 2022. Available at `https://csrc.nist.gov/Presentations/2022/birthday-bound-slide-attacks-on-tinyjambus-keyed-p` .

[9] Ferdinand Sibleyras, Yu Sasaki, Yosuke Todo, Akinori Hosoyamada, Kan Yasuda. Birthday-Bound Slide Attacks on TinyJAMBU's Keyed-Permutations for All Key Sizes. Advances in Information and Computer

Security: 17th International Workshop on Security, IWSEC 2022, pp. 107-127, Tokyo, Japan, August 2022.

[10] Wil Liam Teng, Iftekhar Salam, Wei-Chuen Yau, Josef Pieprzyk, and Raphaël C.-W. Phan. Cube attacks on round-reduced TinyJAMBU. Scientific Reports, 29 Mar 2022, 12(1):5317.

[11] Hongjun Wu, and Tao Huang. The JAMBU Lightweight Authentication Encryption Mode (v2.1). CAESAR Competition Round 3 Submission. `https://competitions.cr.yp.to/round3/jambuv21.pdf`

[12] Hongjun Wu, and Tao Huang. TinyJAMBU: A Family of Lightweight Authenticated Encryption Algorithms (Version 2). NIST Lightweight Cryptography Competition Submission. Available at `https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/finalist-round/updated-spec-doc/tinyjambu-spec-final.pdf` .

[13] Jenny W. Yu and Mark D. Aagaard. Benchmarking and Optimizing AES for Lightweight Cryptography on ASICs. NIST Lightweight Cryptography Workshop 2019. Available at `https://csrc.nist.gov/CSRC/media/Events/lightweight-cryptography-workshop-2019/documents/papers/benchmarking-and-optimizing-aes-for-lwc-on-asics-lwc2019.pdf`