Security and IoT Applications of the Cryptosystem TinyJambu

Amparo Fúster-Sabater
Inst. of Physical and Information Technologies (ITEFI)
Consejo Superior Investigaciones Científicas (CSIC)
144, Serrano, 28006, Madrid, Spain
e-mail: amparo.fuster@csic.es

María Eugenia Pazo-Robles
Inst. of Physical and Information Technologies (ITEFI)
Consejo Superior Investigaciones Científicas (CSIC)
144, Serrano, 28006, Madrid, Spain
e-mail: eugepazorobles@gmail.com

***Abstract*—The cryptosystem TinyJambu is one of the ten finalists in the Lightweight Cryptography Standardization Project launched by the National Institute of Standards and Technology (NIST). In this work, we analyze the TinyJambu security from two different points of view: a) improving the best differential attack found in the literature and b) studying the randomness of its generated sequences. Possible applications of this cryptosystem in Internet of Things (IoT) environments are also considered.**

Keywords-lightweight cryptography; TinyJambu; IoT; stream cipher; randomness.

I. INTRODUCTION

In August 2018, NIST initiated a process to standardize lightweight cryptography algorithms to be deployed in constrained environments [1]. The cryptosystem TinyJambu [2] was one of the ten finalists, as well as the fastest among all the candidates submitted to this call.

Nowadays, IoT technology is being deployed to connect distinct devices of daily use. All these connections need security, i.e. cryptographic algorithms. Stream ciphers provide us with the best examples of cryptosystems to be applied in lightweight environments, e.g. IoT. In this work, we analyze TinyJambu as a stream cipher with particular attention to: (1) the best cryptanalytical attack found in the literature, and (2) a randomness study of the keystream sequences generated by the cryptosystem. After this analysis, it can be stated that TinyJambu might be used in different IoT applications, as those enumerated in Section III. The rest of the paper is structured as follows. In Section II, we discuss the security of TinyJambu. Section III presents some applications of TinyJambu in IoT scenarios. Conclusions and future work end the work in Section IV.

II. SECURITY ANALYSIS OF TINYJAMBU

TinyJambu is an Authentication and Encryption with Associated Data (AEAD) scheme with three different key sizes: 128, 192 and 256 bits. This cryptosystem is based on a keyed-permutation that provides both authentication and encryption. It uses a secret key permutation P_n in the form of Non-Linear Feedback Shift Register (NLFSR), made up of a 128-bit register and a feedback function, see Figure 1. The secret permutation is denoted by P_n where n represents the number of rounds, i.e. NLFSR shifts. In the original design - see [2] page 8, the introduction of the Nonce (message number) and Associated Data performs the permutation P_n with n = 384 rounds.

Next, we introduce several features concerning the cryptanalytical attacks found in the literature (specifically forgery attacks) against the cryptosystem TinyJambu.

A. Generalities of a Forgery Attack against TinyJambu

We denote by S_i and T_i $(i = 0,1,...,127)$ the binary contents for two slightly different initial states of TinyJambu. Thus, the initial differential is defined as:

$$\Delta S_i = S_i + T_i \quad (i = 0,1,..., 127), \qquad (1)$$

where the symbols "+" and "." represent the XOR and AND logic operations, respectively. After n rounds, both states have shifted according to the keyed permutation P_n. As the only nonlinear component per round in TinyJambu is the NAND logic operation (see Figure 1), then the differential of such an operation can be a good measure of how the differences propagate along n rounds. Since the NAND operation is the complementation of the AND operation, we can easily replace the NAND gate by an AND gate (omitting the 1) without affecting the result of this differential analysis and consider the differential $\Delta(S_{70+j} \,.\, S_{85+j}) = (S_{70+j} \,.\, S_{85+j}) + (T_{70+j} \,.\, T_{85+j})$ $(j = 0,1,..., n)$ as the propagation measure. Finally, we define an *active AND gate* as a differential of value:

$$\Delta(S_{70+j} \,.\, S_{85+j}) = 1, \qquad (2)$$

for any j in the range $(j = 0,1,..., n)$, as they propagate the actual differences that allow this cryptanalytical technique. In brief, we try to find differential trails that, after n rounds (in practice n = 384), minimize the number X of active AND gates, as a trail with score X can be satisfied with probability $p = 2^{-X}$. As long as the probability $p \geq 2^{-64}$, it allows one to launch an attack that breaks the 64-bit security claimed by the cryptosystem designers in [2]. This differential attack is called a forgery attack as we introduce a false Nonce that forces a particular initial differential, which in turn will allow us to obtain a number of differential trails with the minimum score X.

B. Successive Security Evaluations of TinyJambu

The first security evaluation of TinyJambu was performed by its own designers in [2]. In fact, they computed a forgery attack probability of value $p = 2^{-80}$, which was far

Copyright (c) IARIA, 2024. ISBN: 978-1-68558-180-0

away from the 64-bit security. Consequently, they stated the immunity of TinyJambu regarding this kind of differential attack. Later, a new security evaluation with 384 rounds was reported in [3] where the authors introduced a method of finding differential trails by means of Mixed Integer Linear Programming. In fact, they introduced the concept of *correlated AND gate* as a correlation between successive active AND gates, see [3]. Indeed, if $(\Delta S_{70+j}, \Delta S_{85+j}, \Delta S_{100+j}) = (1,0,1)$ and $S_{85+j} = 1$, then $\Delta(S_{70+j} \,.\, S_{85+j})$ and $\Delta(S_{85+j} \,.\, S_{100+j})$ are jointly active AND gates, that is active gates separated by 15 rounds (the distance between the inputs to the NAND gate). As they count both correlated gates as a single active gate, they reduce the number X of active gates through n rounds and the success probability $p = 2^{-X}$ is consequently incremented. In this way, they found a differential trail with probability $p = 2^{-74}$, as well as other trails with higher probabilities - see the numerical values computed by Saha *et al.* [3] in Table 1. Later, summing up the number of trails multiplied by their corresponding probabilities, they computed a global differential probability of value $p = 2^{-70.68}$.

C. Our Contribution to the Security of TinyJambu

Making use of the correlated AND gate model developed in [3], we have searched for differential trails with a number of active AND gates satisfying $X < 74$ along $n = 384$ rounds. In order to accomplish this task, we have used Gurobi Optimizer [4], several programs written in Python language (Python 3.11 64-bits) and a desktop PC with a 13th Gen Intel® Core™ with 3.00 GHz, RAM 128GB with 24 cores and Microsoft Windows 11 Pro operating system. Proceeding in this way, we have ranged in a long interval of solutions provided by Gurobi and have found several differential trails with $X = 71$ active gates (84 actives gates and 13 correlated gates) - see the numerical values computed in Table 1 for the epigraph “This work” with X in the interval [71, …, 75]. Notice that, in our case, the number of trails is greater than the number computed in [3], as well as the number of active AND gates is lower than that of [3]. Combining both effects, we compute a global differential probability of value

$$p = 2^{-65.948}. \qquad (3)$$

In a more powerful computational scenario (we have just used a desktop PC), we could derive an even better differential probability to, in turn, break the claimed 64-bit security of the cryptosystem with 384 rounds.

D. Randomness Analysis of Keystream Sequences

TinyJambu is a stream cipher cryptosystem. Therefore, a randomness analysis of the keystream sequences generated by it must be performed. The length of our sequences is 2^{23} bits. We have used three kinds of tests: graphical tests, the Diehard battery of tests (see Figure 2) and the family of statistical tests Federal Information Processing Standards 140-2 (FIPS 140-2) developed by NIST. In our experiments, all the analyzed sequences pass satisfactorily the previous tests.

III. APPLICATIONS OF TINYJAMBU IN IOT SCENARIOS

As we have seen, TinyJambu with 384 rounds exhibits some security flaws. Consequently, the designers suggest to increase the number of rounds up to 640 in the Nonce introduction. Clearly, with 640 rounds, the number X of active gates increases and the success probability of a forgery attack will be dramatically reduced.

On the other hand, due to the lightness of the components, TinyJambu unifies in a single algorithm speed and simplicity, what means very low energy consumption, as well as less hardware involved. In spite of the security issue found for 348 rounds, TinyJambu is a very fast and ductile algorithm that allows easily to increase the number of rounds up to 640, with the corresponding increment in the security level. It is important to notice that many IoT sensors are not managed nor are equipped with security mechanisms. It is in these situations where the use of TinyJambu with 640 rounds is recommended. As examples of TinyJambu applications (see Figure 3) we can enumerate, among others:

- Any sort of wearable devices (fitness tracker, smartwatches, wearable blood pressure measuring devices, etc);
- Environmental sensors: humidity, temperature, smart agriculture (good environmental conditions);
- Smart cities (air quality, parking planification);
- Industry 4.0 (automation of industrial processes);
- Tracking for truck fleets; etc.

In general, TinyJambu can be used in any kind of application where the security levels were not very demanding. Nevertheless, the use of TinyJambu for critical infrastructures is not recommended.

IV. CONCLUSIONS

Although TinyJambu with 384 rounds exhibits clearly security flaws, the updated version with 640 rounds seems to be immune to differential attacks, in particular forgery attacks. This is the reason why this new version of TinyJambu in conjunction with good performances (good relationship throughput/area, speed in encryption/decryption process and low energy consumption) allow one to recommend this cryptosystem for its deployment in IoT applications with no high security.

The study of the relationship between the number of rounds and the minimum number of active AND gates, as well as the possible implementation of TinyJambu in the frame of the Message Queuing Telemetry Transport (MQTT) protocol (designed for connections among devices with resource constraints or limited bandwidth, such as in IoT) are some of our priorities for a near future work.

TABLE I. COMPARISON BETWEEN RESULTS

Saha et al.							
Probability	2^{-74}	2^{-75}	2^{-76}	2^{-77}	2^{-78}	2^{-79}	2^{-80}
#Trails	1	5	9	14	20	24	30
This work							
Probability	2^{-71}	2^{-72}	2^{-73}	2^{-74}	2^{-75}	2^{-76}	2^{-77}
#Trails	9	24	27	28	18	14	22

ACKNOWLEDGMENT

This work is part of the R+D+i grant P2QProMeTe (PID 2020-112586RB-I00) funded by MCIU/AEI/10.13039/501100011033. It is also funded by University of Málaga (Spain) through network “BIOMED-SEC”, reference D5-2022-04.

REFERENCES

[1] National Institute of Standards and Technology (NIST). *Lightweight Cryptography (LWC) Standardization Project*, 2019. Available from: https://csrc.nist.gov/projects/lightweight-cryptography [retrieved: May, 2024]

[2] H. Wu and T. Huang, “TinyJAMBU: A Family of Lightweight Authenticated Encryption Algorithms, ” The NIST Lightweight Cryptography (LWC) Standardization Project, 2020. Available from: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/TinyJAMBU-spec-round2.pdf [retrived: May, 2024]

[3] D. Saha, Y. Sasaki, D. Shi, F. Sibleyras, S. Sun, and Y. Zhang, “On the Security Margin of TinyJAMBU with Refined Differential and Linear Cryptanalysis, ” IACR Trans. on Symmetric Cryptology, vol. 3, pp. 152-174, March 2020, doi:10.13154/tosc.v2020.i3.152-174

[4] Gurobi Optimizer. https://www.gurobi.com/academia/academic-program-and-licenses/ [retrived: May, 2024]

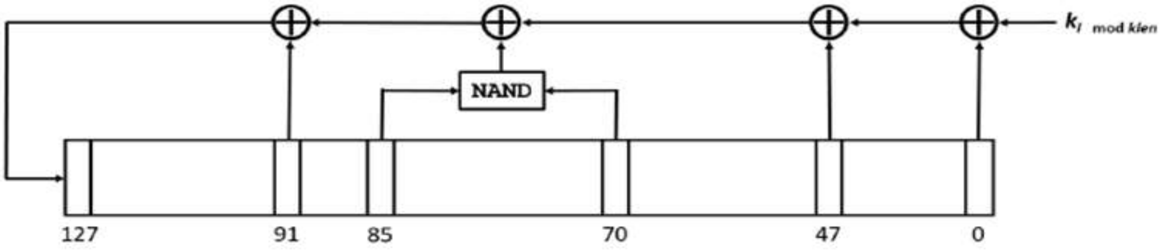

Figure 1. General scheme of the cryptosystem TinyJambu.

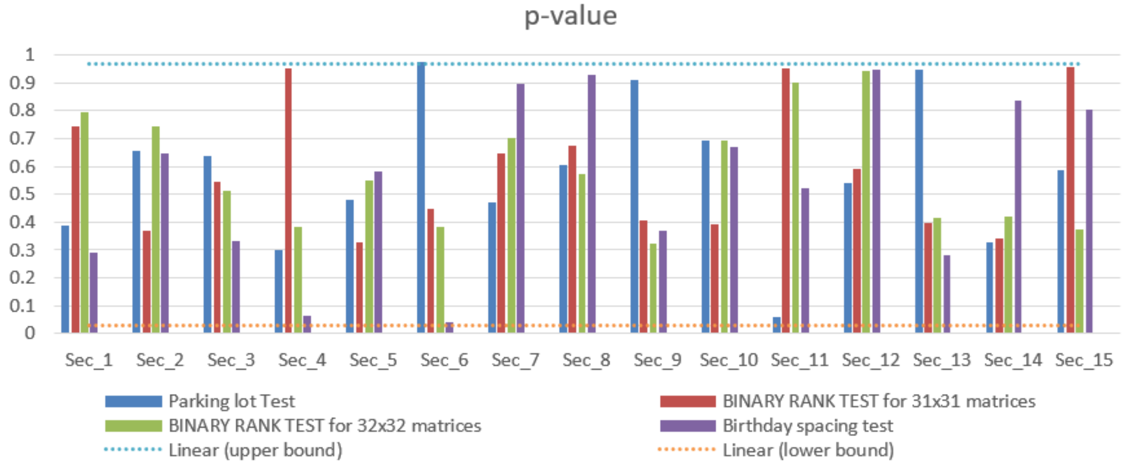

Figure 2. Results of four DIEHARD tests applied to 15 TinyJambu sequences (sequence_1 - sequence_15).

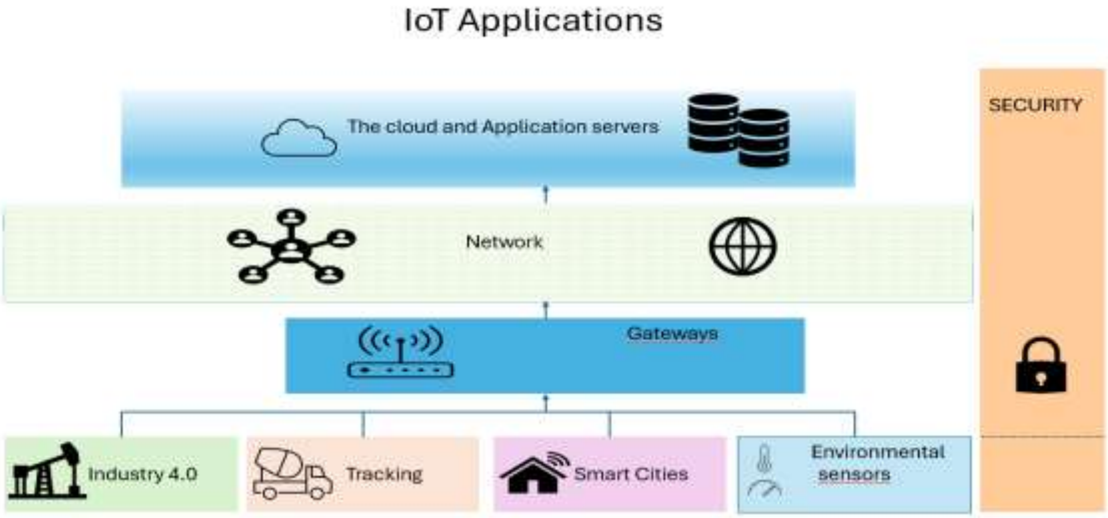

Figure 3. IoT applications.