Security analysis of lightweight cryptographic algorithms

Dilyana Dimitrova
Department of Information Technologies
Nikola Vaptsarov Naval Academy
Varna, Bulgaria
di.dimitrova@naval-acad.bg

Ivaylo Dimitrov
Engineering Department
Blu11 Ltd.
Varna, Bulgaria
ivailo.dimitrov@blu11.com

***Abstract*. The paper examines three lightweight cryptographic algorithms - SKINNY, ForkAE, and Romulus. The research focuses on evaluating their security against various cryptographic attacks. Methods used: theoretical analysis and summary. Results indicate that all three algorithms exhibit strong security properties against common cryptographic attacks. SKINNY stands out for its security even with few encryption rounds, while the presence of SKINNY as a building block in the other two ciphers - ForkAE and Romulus makes them at least as secure as SKINNY.**

***Keywords:* lightweight cryptographic algorithms, lightweight cryptography, security analysis**

I. Introduction

In the modern world, the use of small IoT (Internet of Things) devices is becoming increasingly common, aiming to simplify our everyday life. While these devices are useful, their widespread adoption, coupled with the increased risk of cyberattacks, is leading to a growing number of vulnerable devices that are not properly protected against attacks. The weaknesses of IoT devices place a significant risk to both user's health and the protection of their personal data. Therefore, the way this information is protected is crucial, including what security and encryption methods are used when transmitting data from the device to the service-providing servers, as well as how the user's personal information is stored. To make IoT devices more secure, appropriate cryptographic algorithms should be used.

When using IoT devices, conventional cryptographic methods such as the symmetric cryptographic algorithm AES, hashing functions like SHA-256, MD5, as well as other cryptographic security methods such as RSA or ECC (Elliptic Curve Cryptography), do not perform optimally on systems with limited computational power and memory capacity because they occupy too much physical space and processor power, consequently consuming too much power, which is unacceptable for devices with limited capabilities [1], [2]. One of the biggest security threats associated with IoT devices is that even the simplest data collection devices (sensors and measuring modules) can be vulnerable to cyberattacks. Due to their small size and specific applications, most IoT devices do not have the computational power and capabilities of a server installation or even a personal computer. Therefore, special requirements and limitations related to size, consumption, and data processing speed are introduced for lightweight cryptography [3].

II. Materials and Methods

The paper involved the examination of three lightweight cryptographic algorithms: SKINNY, ForkAE, and Romulus. The study aimed to evaluate the security of these algorithms against various cryptographic attacks. Data collection for the study involved gathering information from existing literature sources, including research papers, conference proceedings, and technical documents related to the selected lightweight cryptographic algorithms. Theoretical analysis and summaries were made based on the information obtained from these sources. Theoretical analysis was performed to assess the security properties of the selected cryptographic algorithms. This involved studying the structure of the selected ciphers, key generation methods, encryption and decryption processes, and susceptibility to common cryptographic attacks such as differential and linear cryptanalysis. A summary of the findings from the theoretical analysis was compiled to provide insights into the security of each cipher. Emphasis was placed assessing the overall robustness of the selected cryptographic algorithms against potential attacks.

In the existing literature sources on the topic, analyses of the security of various lightweight cryptographic algorithms have been conducted. In the study [4] a differential cryptanalysis of the lightweight ciphers SIMON and SIMECK is presented, using nested tree search-based methods, to find high probability differential characteristics for the ciphers. The study [5] provides a comprehensive analysis of 101 existing lightweight algorithms, emphasizing the importance of incorporating secure design components such as substitution and permutation functions to ensure robust security in IoT devices. Selection of lightweight cryptographic algorithms

Print ISSN 1691-5402
Online ISSN 2256-070X
https://doi.org/10.17770/etr2024vol4.8233

for analysis might be done using the Analytical Hierarchy Process [6], [7], [8].

The absence of security studies comparing ciphers SKINNY, ForkAE and Romulus following the literature review underscores the relevance of the issue.

III. Results and discussion

A. *Selected ciphers*

Lightweight cryptographic algorithms could be divided into four main types of primitives - block ciphers, stream ciphers, hash functions, and cryptographic algorithms using elliptic curves as it is shown on figure 1. The factors by which each of them can be analysed include the size of the blocks used, the size of the key used, the number of executable rounds, their structure itself, security against different attacks etc.

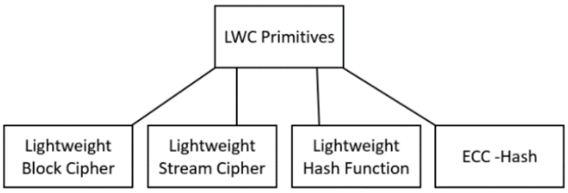

Fig.1. Division of lightweight cryptographic primitives.

The analysed lightweight cryptographic algorithms in the paper are SKINNY, ForkAE, and Romulus.

SKINNY is a lightweight SPN block cipher that uses substitution blocks (S-boxes) [10], as it is shown on fig. 2 [11], a greatly simplified new model for the diffusion layer, and a lightweight method for key generation. The cipher is based on the Tweakey structure, which uses so-called tweakey values [9], [10] as the input key to the cipher, rather than, as in traditional symmetric cryptographic algorithms, a secret key. Essentially, the secret key and the tweakey make no difference in the execution of the cryptographic algorithm.

Representatives of the SKINNY cipher family are SKINNY-AEAD and SKINNY-HASH, which respectively represent an encryption algorithm and a hash function. There are different versions of SKINNY, distinguished by the size of the used data block and the length of the tweakey value. The implementation of SKINNY in an AEAD scheme can be done with both SKINNY-128-256 and SKINNY-128-384 [10], [11]. Both ciphers use data blocks with a size of 128 bits, and the main difference lies in the tweakey value used for the key, with either 256 or 384 bits, respectively.

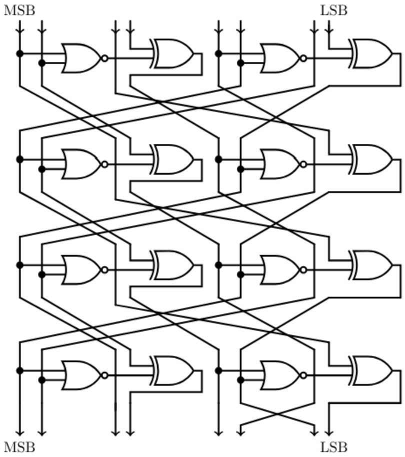

Fig.2. 8-bit S-box construction [11].

ForkAE [12] is a family of lightweight cryptographic algorithms designed to meet the construction of authenticated encryption with associated data (AEAD) ciphers. Unlike SKINNY, ForkAE is tightly optimized for processing short messages. This ensures good performance, security, and simplicity of operation. The cipher's specialization in short messages makes it a suitable candidate for a wide range of lightweight and IoT applications, including wireless sensors, and IoT devices that require very low energy consumption. In addition to these, short messages find applications in critical communication domains of 5G networks and protocols like Bluetooth, where the maximum packet size is 47 bytes.

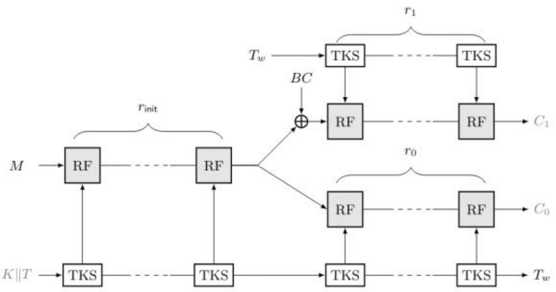

Fig.3. The structure of ForkSkinny, where TKS is round tweakey schedule function and RF is round function [13].

ForkAE is based on a combination of several well-analysed elements [12]. The building block of the cipher is Forkcipher. The standalone use of Forkcipher does not meet the necessary security requirements of NIST. Therefore, to achieve better results, the cipher is combined with another block cipher - SKINNY. This improves efficiency and throughput, as well as revealing new software advantages for applications and better hardware implementations. The combination of the two ciphers is called ForkSKINNY which is shown on fig. 3 [13]. In addition to performance advantages, it provides better results in the field of cryptographic security, as it achieves resistance against a wider range of cryptographic attacks, especially against more modern cryptanalytic techniques.

The security of the cipher depends mostly on round function, so its proper design and use are crucial stages in the design of the specific cipher. The same operations are used to modify the data as SKINNY (SubCells, AddConstants, AddRoundTweakey, ShiftRows, MixColumns), with the difference that during the "AddConstants" operation, certain changes have been made to the operation's structure. This is because by design, the ForkSKINNY cipher has more rounds than SKINNY, which means that applying the original operation cannot provide the necessary number of unique constants for all rotations of the function. This leads to the repetition of some constant values and can therefore be a vulnerability and weak point in the cipher. The change made by the cryptographers who designed ForkSKINNY to avoid this potential problem is that they increased the length of the constant itself. In SKINNY, the constant has a length of 6 bits, while in ForkSKINNY, it has been changed to 7 bits. This allows the generation of a larger number of unique constant values needed for most rounds in the cipher.

The most significant difference between ForkSKINNY and SKINNY is the presence of an additional step in message processing. This operation is unique to ForkSKINNY, as it is linked to the design and structure of the cipher itself. This step is called forking and is used in generating the two cipher blocks in Forkcipher.

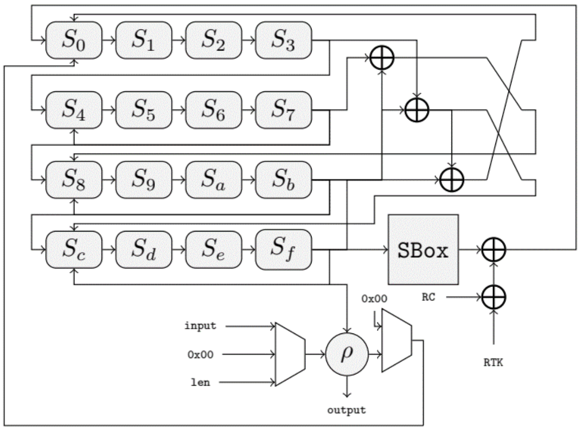

Fig.4. Serial State Update Function used in Romulus [14].

Romulus offers three AEAD schemes - Romulus-N, Romulus-M, and Romulus-T, as well as a hash function - Romulus-H. The cipher is specialized in processing serial data as it is shown on fig. 4 [14]. The first variant of the cipher is oriented towards nonce-based authenticated encryption (NAE). In cryptography, the term "nonce" represents a random number that can only be used once in cryptographic communication. Typically, this number is randomly generated or pseudo-randomly generated, with its primary purpose being to ensure that old communication sessions cannot be reused by an attacker in authenticated encryption. The second variant is applicable for "nonce misuse-resistant" authenticated encryption (MRAE) [14]. The third variant limits the potential for physical data leakage through side-channel attacks. At its core, the Romulus cipher is based on the structure for tweakable block ciphers. The main goal of the cipher is to optimize the NAR/MRAE schemes in such a way that they are applicable in constrained IoT devices.

Like ForkAE, the Romulus team also chose to use the SKINNY lightweight block cipher as the main building block in constructing their cipher. This allows them to inherit all the strengths of the SKINNY cipher, including all the advancements in cryptographic security achieved by SKINNY. Since SKINNY offers various versions and variants of the used blocks and tweakey values, the Romulus team has opted for only one variant, which features 128-bit blocks and tweakey values of 2n and 3n lengths, meaning 256 and 384-bit values for the tweak.

B. Cryptographic security

The security of block ciphers, whether they are Feistel ciphers or ciphers based on substitution and permutation, has been extensively studied. However, when the attacker is allowed to have access to encryption or decryption with different keys for the same message, then he can establish various relationships between encryption and decryption operations without knowing the actual message. Many ciphers lose their security and robustness precisely under such attacks. Numerous ciphers considered secure have been compromised by related-key/related-tweak attacks.

The family of block ciphers SKINNY is designed to be resistant to related-key attacks [15], [16]. In this type of attack, the attacker can observe the behaviour of the cipher under different keys without needing to know the initial value of the key used but known mathematical dependencies in its structure [16].

The behaviour of SKINNY against the most well-known attacks in cryptography - differential and linear attacks [15], [17], can be demonstrated by calculating the smallest number of pairs of plaintext and ciphertext for the smallest possible number of active substitution blocks. An active substitution block is defined as any block with a non-zero input difference. Attacks based on differential cryptanalysis exclusively work by detecting differences between input and output data when subjected to some alteration.

Unlike the standard single-key model where the round tweakeys are constant values and cannot be changed, thus not affecting the activity model, in the related-tweakey model, the attacker can change some of the states of the tweakey matrices. SKINNY can have 3 tweakey input matrices depending on which version of the cipher is being applied, thus there are three attack variants on the tweakey matrices. Only one of the matrices (TK1) may be changed, both at the same time (TK1, TK2), or all three (TK1, TK2, TK3).

The security of the second algorithm – ForkSKINNY, to a certain extent, is based on the security of the SKINNY cipher because it is one of the main building blocks in the overall construction of the ForkAE family of block ciphers [13]. In this regard, all arguments related to the security of SKINNY are applicable here as well. If it is assumed that the attacker has access to the plaintext and at the same time knows the ciphertext of the first block (C_0), this type of attack is equivalent to breaking SKINNY with parameters equal to r_{init} + r_1 - round. However, since ForkSKINNY has known structural differences from the original SKINNY and the cipher has an additional forking step on the messages, analysing the security of ForkSKINNY requires an analysis of the so-called reconstructive attacks. This type of attacks are applicable in situations where the attacker has access to both blocks of ciphertexts and can generate values for one block from the other, and vice versa [18]. This difference in ForkSKINNY is due to the construction of the cipher because the two cipher blocks are interrelated. Reconstructive attacks focus on the middle rounds of the cipher, when operations switch from decryption to encryption.

The last representative of the selected lightweight ciphers – Romulus [19], provides two modes of operation: nonce-respecting (NR) and nonce-misusing (NM). For each of them, there is a proposed value up to which the security of the encrypted data is guaranteed. The security analysis is based on the number of queries made and the total number of processed message blocks. The proposed results guarantee that the cipher is considered secure up to these values and exceeding them could compromise and break the cryptographic algorithm [19]. Table 1 presents the assumed values up to which the Romulus algorithm is considered secure. The numbers in the table represent the effort required by the attacker in terms of data complexity to break the cipher, calculated by taking the logarithm at base 2.

TABLE 1 THE ASSUMED SECURITY VALUES FOR ROMULUS

	Romulus-N	Romulus-M
NR-Priv	128	128
NR-Auth	128	128
NM-Priv	–	64 ~ 128
NM-Auth	–	64 ~ 128

Meet-in-the-Middle attack

One way to determine the security of a cipher against Meet-in-the-Middle attacks is to examine the diffusion of the cipher. The diffusion [20] of a cipher represents the number of rounds d, required for any input bit to influence all other bits of the cipher's internal state (IS) matrix. In other words, the change in one input bit leads to changes in all other bits in the IS matrix. When the key length is equal to the block length and the entire key is used in each round, then for a cipher with diffusion equal to d, it means that each output bit after that d round is an expression dependent on all other key bits.

In Meet-in-the-Middle attacks, SKINNY provides very good security [15]. To determine its security level, three important characteristics are considered: partial-matching, initial structure, and splice-and-cut. Each characteristic has a limit at which it may work. For SKINNY, the partial-matching characteristic succeeds up to the 10th round, the initial structure is successful up to the 7th round, and splice-and-cut has been calculated to work up to the 5th round. By combining all characteristics, the number of rounds required for the cipher to withstand Meet-in-the-Middle attacks is obtained. The result is 22 rounds, but SKINNY's capability to operate beyond these 22 rounds, usually 48 or 56, provides significant resilience to this type of attack.

On the other hand, for ForkAE [13], it should be noted that only half of the tweakey value is used in each round, and the forking step has a lower diffusion value, which adds additional rounds to the mandatory 22 provided by SKINNY. Thus, the rounds required to break the cipher using a Meet-in-the-Middle attack become even more than 22, indicating that the ForkAE cipher can also be considered resilient to Meet-in-the-Middle attacks.

For the security of the Romulus cipher, specific data regarding its resilience against Meet-in-the-Middle attacks are currently not available. However, considering that Romulus also utilizes SKINNY as its primary building block in its construction, it can be assumed that Romulus is also resilient to Meet-in-the-Middle attacks, at least up to the minimum 22 rounds provided by SKINNY, which are assumed to make the cipher secure.

Impossible Differential Attack

In Impossible Differential attacks, two values (α, β) are considered, determining that for all possible keys, two messages with an XOR difference equal to α cannot produce other two messages differing by β after a certain number of encryption rounds (r) [21]. To discover the key, the attacker adds several rounds before and after r, then makes assumptions about some key bits, checking if the values for α and β are confirmed. If so, the assumption about the key is wrong because it leads to an impossible situation (two different keys having the same α and β values). After a certain number of repetitions, the total number of keys becomes small enough to apply a brute force attack on the key by trying all possible key variants.

The security of SKINNY [15] against this type of attack is evaluated at a maximum of 11 rounds of encryption, beyond which the cipher is considered to be broken if a truncated attack type is used. After these 11 rounds, it is assumed that the key information is lost, and the attack becomes ineffective. If the attacker has access to the relationship between different tweakey values (related-tweakey), the security of SKINNY increases with each used tweakey value. Accordingly, SKINNY can use 3 tweakey value matrices, and depending on the number used, the following security values against Impossible Differential attacks are determined: TK1 (128 bits) - 12 rounds, TK2 (256 bits) - 14 rounds, and for TK3 (384 bits) - 16 rounds.

On the other hand, in ForkAE [13], it is necessary to determine the security during the forking operation because the remaining cipher structure is like SKINNY. However, if the security of the cipher is viewed only during the forking operation, it is like that of SKINNY. It has been calculated that a truncated differential attack is not possible after the 12th round of ForkAE, making it as secure as SKINNY at least.

The security of Romulus depends entirely on the cryptographic security obtained from SKINNY since Romulus does not make significant changes to the design of its underlying cryptographic primitive - SKINNY.

IV. Conclusions

SKINNY is a lightweight Substitution-Permutation Network (SPN) block cipher based on the tweakey structure, which specializes in processing messages in a parallel manner. ForkAE is a family of lightweight block cryptographic algorithms that are closely optimized for processing short messages. Romulus is a block cipher specialized in processing serial data. Each cipher is specialized in a specific direction and offers different modes of operation, authentication methods, and possibilities for software and hardware implementation.

The family of block ciphers SKINNY is designed to be resistant to related-key attacks. SKINNY also demonstrates good resistance against differential and linear attacks. The security of ForkSKINNY to a certain extent is based on the security of the SKINNY cipher. However, its construction has known structural differences from the original SKINNY, making it susceptible to reconstructive attacks. Every cipher has proposed secure values which guarantee that the cipher is considered secure up to these values and exceeding them could compromise and break the cryptographic algorithm.

The security of a cipher against Meet-in-the-Middle attacks can be determined by examining its diffusion. SKINNY provides very good security against Meet-in-the-Middle attacks, with a required number of rounds of 22. The rounds required to break ForkAE with this type of attack are more than this of SKINNY. The Romulus resilience against Meet-in-the-Middle attacks is currently unknown, but since it uses SKINNY as its primary building block, it can be assumed to be resilient up to the minimum 22 rounds required by SKINNY.

The security of SKINNY against Impossible Differential attacks is evaluated up to 11 rounds of encryption, beyond which the cipher is considered to be broken. ForkAE is at least as secure as SKINNY against this type of attack. Romulus's security depends on SKINNY's cryptographic security.

After analyzing the selected lightweight cryptographic algorithms, it can be concluded that the security the ciphers provide against well-known attacks such as differential and linear cryptanalysis, as well as attacks like Meet-in-the-Middle Attack and Impossible Differential Attack, meets current security requirements. The SKINNY cipher offers good security even with a small number of rounds used, and its ability to use a significantly larger number of rounds in its encryption function makes it resistant against the most well-known cryptographic attacks. The presence of SKINNY as a building block in the other two ciphers, ForkAE and Romulus, also makes them at least as secure as SKINNY.

The report is in implementation of the National Scientific Program "Security and Defense", adopted with RMS No. 731/21.10.2021, and financed by the Ministry of Education and Science of the Republic of Bulgaria according to Agreement No. D01-74/19.05.2022.

References

[1] W. Easttom, *Modern Cryptography: Applied Mathematics for Encryption and Information Security*. Springer, 2021.

[2] M. Banday, Cryptographic Security Solutions for the Internet of Things. IGI Global, 2019.

[3] NIST, "Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process" [Online]. Available from: https://csrc.nist.gov/csrc/media/Projects/lightweight-cryptography/documents/final-lwc-submission-requirements-august2018.pdf.

[4] A. D. Dwivedi and G. Srivastava, "Security analysis of lightweight IoT encryption algorithms: SIMON and SIMECK," Internet Things, vol. 21, p. 100677. ISSN 2542-6605, 2023. doi:10.1016/J.IOT.2022.100677.

[5] A. A. Zakaria et al., "Systematic literature review: Trend analysis on the design of lightweight block cipher," J. King Saud Univ. Comput. Inf. Sci., vol. 35, no. 5, p. 101550. ISSN 1319-1578, 2023. doi:10.1016/J.JKSUCI.2023.04.003.

[6] V. Petrova, "The Hierarchical Decision Model of cybersecurity risk assessment" 12th National Conference with International Participation (ELECTRONICA), vol. 2021, 2021, pp. 1-4. doi:10.1109/ELECTRONICA52725.2021.9513722. 978-1-6654-4061-5.

[7] V. Petrova, "Using the Analytic Hierarchy Process for LMS selection": 20th International Conference on Computer Systems and Technologies. Ruse, Bulgaria: Pages, ISBN: 978-1-4503-7149-0, Jun. 2019, pp. 332-336. doi:10.1145/3345252.3345297.

[8] M. Sotirov and V. Petrova, "The Nine-Steps Gamification Process: Increasing Student Engagement in LMS," in *2023 International Conference Automatics and Informatics (ICAI)*, IEEE, 2023, pp. 496–501.

[9] J. Jean et al., "Tweaks and keys for block ciphers: The TWEAKEY framework" in Asiacrypt 2014. Lecture Notes in Computer Science, vol. 8874, P. Sarkar, T. Iwata, Eds. Berlin, Heidelberg: Springer, 2014, 274-288. doi:10.1007/978-3-662-45608-8_15.

[10] C. Beierle et al., "The SKINNY family of block ciphers and its low-latency variant MANTIS" in Crypto 2016. Lecture Notes in Computer Science, M. Robshaw, J. Katz, Eds., 2016, 123-153. doi:10.1007/978-3-662-53008-5_5(), vol 9815. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-53008-5_5.

[11] C. Beierle et al., "SKINNY-AEAD and SKINNY-Hash v1.1." Accessed: Dec. 11, 2019. [Online]. Available: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/SKINNY-spec-round2.pdf.

[12] A. Deprez et al., "Optimized software implementations for the lightweight encryption scheme ForkAE" in, Smart Card Research and Advanced Applications, P. Y. Liardet, N. Mentens, Eds., 2021, 68-83. doi:10.1007/978-3-030-68487-7_5 Smart Card Research and Advanced Applications. CARDIS, Lecture Notes in Computer Science, 2020(), vol 12609. Springer, Cham. https://doi.org/10.1007/978-3-030-68487-7_5.

[13] E. Andreeva, A. Deprez, J. Pittevils, A. Roy, A. Singh Bhati, and D. Vizár, "New Results and Insighs on ForkAE." Accessed: Apr. 17, 2024. [Online]. Available: https://csrc.nist.gov/CSRC/media/Events/lightweight-cryptography-workshop-2020/documents/papers/new-results-ForkAE-lwc2020.pdf.

[14] T. Iwata et al., "Romulus v1.2" [Online]. Available at: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/Romulus-spec-round2.pdf.

[15] C. Beierle et al., "SKINNY-AEAD and SKINNY-hash v1.1". Available at: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/SKINNY-spec-round2.pdf. NIST [Online].

[16] R. Ankele et al., "Related-Key Impossible-Differential Attack on Reduced-Round SKINNY." Accessed: Apr. 17, 2024. [Online]. Available: https://eprint.iacr.org/2016/1127.pdf.

[17] H. M. Heys, "A TUTORIAL ON LINEAR AND DIFFERENTIAL CRYPTANALYSIS," Cryptologia, vol. 26, no.

3, pp. 189–221, Jul. 2002, doi: https://doi.org/10.1080/0161-110291890885.

[18] K. G. Paterson et al., “Security against related randomness attacks via reconstructive extractors” in Lect. Notes Comput. Sci.. IMACC 2015, J. Groth, Ed. Cryptography and Coding, 2015(), vol 9496. Springer, Cham. https://doi.org/10.1007/978-3-319-27239-9_2.

[19] C. Guo et al., Final-Round Updates on Romulus, 2022.

[20] C. Shannon, “Diffusion and Confusion.” Available: https://www.nku.edu/~christensen/diffusionandconfusion.pdf.

[21] A. Biryukov, "Impossible Differential Attack," in Encyclopedia of Cryptography and Security, H.C.A. van Tilborg, Ed. Boston, MA: Springer, 2005, pp. 197. [Online]. Available: https://doi.org/10.1007/0-387-23483-7_197.