Full Key Extraction of SNOW-V Using ML-assisted Power SCA

Harshit Saurabh, Suparna Kundu, Samarth Shivakumar Titti, Anupam Golder, K K Soundra Pandian, Chaoyun Li, Angshuman Karmakar, and Debayan Das

***Abstract*—This paper proposes the first Side-Channel Analysis (SCA) attack with full key recovery on SNOW-V, a 5G mobile communication standardization candidate. Our preliminary analysis examines the SNOW-V architecture, revealing that the Linear Feedback Shift Register (LFSR) is the most susceptible point of attack. We then performed a Test Vector Leakage Assessment (TVLA) and Known-Key Correlation (KKC) to identify the leakage point. Subsequently, Correlational Power Analysis (CPA) attack is utilized to recover one key byte at a time. The correct secret key is then uniquely identified using Linear Discriminant Analysis (LDA). Additionally, we demonstrate how an incremental attack can be performed to recover all key bytes of SNOW-V. Finally, we integrated a Boolean masking countermeasure to secure SNOW-V implementation against SCA attacks.**

***Index Terms*—SNOW-V, Side Channel Analysis (SCA), Linear Feedback Shift Registers (LFSRs), Known-Key Correlation (KKC), Boolean masking, Countermeasures**

I. Introduction

THE upcoming 5G wireless networks promise high data rates, ultra-low latency, and enhanced Quality of Service. However, 5G networks significantly increase the demand for robust security and privacy measures.

In 2018, 3rd Generation Partnership Project (3GPP) tasked the European Telecommunications Standards Institute's Security Experts Group (ETSI SAGE) with creating new 256-bit cryptosystems for 5G networks [1], aiming for speeds exceeding 20 Gbps on both dedicated hardware and general-purpose CPUs, quantum-safe characteristics, and support for ultra-reliable low-latency communications with a 1ms latency budget [2]. Additionally, these systems must comply with the National Institute of Standards and Technology (NIST) recommendation for a classical 256-bit security level to withstand quantum computers. This initiative led to developing the stream cipher called SNOW-V [3].

This work was supported in part by the Cyber Security Karnataka (CySecK) initiative, Power Grid Centre of Excellence (PGCoE), and in part by the Department of Science & Technology (DST), India.

H. Saurabh, S. S. Titti, and D. Das are with the Indian Institute of Science, Bangalore, India 560012 (e-mail: {harshitsaura; samarthst; debayandas}@iisc.ac.in)

S. Kundu is with KU Leuven, Belgium (e-mail: suparna.kundu@esat.kuleuven.be)

A. Golder is with the Intel Labs, USA.

K K Soundra Pandian is with the Ministry of Electronics and Information Technology, India (e-mail: dr.pandian@meity.gov.in)

C. Li is with the University of Surrey, UK (e-mail: c.li@surrey.ac.uk)

A. Karmakar is with the Indian Institute of Technology, Kanpur, India (e-mail: angshuman@cse.iitk.ac.in)

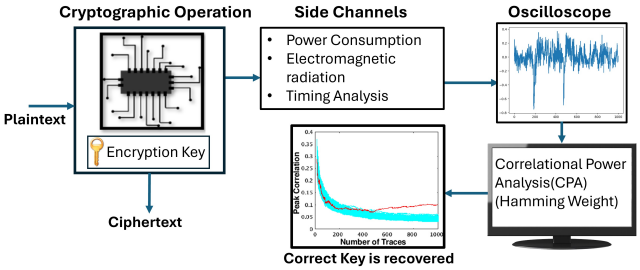

Fig. 1. Possible Side channel attacks on a Cryptographic device during encryption

SNOW-V, proposed by Ekdahl et al. [4], aims to replace SNOW 3G as the encryption standard for 5G systems, enhancing the algorithm from 128-bit to 256-bit security to meet 5G network requirements while addressing vulnerabilities for stronger cryptographic protection.

A. Motivation

Current cryptographic standards like SNOW 3G, designed for 3G networks, require adaptation for 5G systems. Transitioning to SNOW-V is necessary to meet increased security demands. In symmetric key cryptography, threats from quantum computers can be mitigated by doubling the length of the keys, since Grover's algorithm [5] provides only a quadratic speed-up against classical methods. To ensure robust future security, 3GPP and ETSI began standardizing 256-bit symmetric key algorithms in 2018, with SNOW-V as a candidate for 5G security.

Since SNOW-V is a lightweight stream cipher, it is ideal for securing air interface communications in the radio access network (RAN) between devices and base stations [6]. The resource-constrained lightweight IoT devices in the RAN are prime targets for side-channel attacks due to their high side-channel signal-to-noise ratio.

SNOW-V improves upon SNOW 3G by enhancing software throughput, addressing its predecessor's limitations. 3GPP recognized SNOW 3G's poor software performance and noted SNOW-V's similarity to 128-NxAx cryptosystems [7], facilitating hardware reuse in mobile devices. Moreover, ETSI SAGE anticipates that the 256-bit SNOW-V appears better suited for 256-bit security, which is also deemed more resistant to side-channel attacks than SNOW 3G [8], making the assessment of its SCA security essential before its deployment.

B. Contribution

This work introduces the first power SCA attack with full key recovery on the SNOW-V stream cipher operating on a

32-bit ARM Cortex-M4 microcontroller. To summarize, the main contributions of this research are:

- We present the first power SCA attack with full key recovery on the SNOW-V cipher, which combines a non-profiled CPA and a Machine Learning-based profiled attack model to recover the full key bytes of the SNOW-V cipher. The LDA model achieves more than 99% accuracy after training with 1K traces.
- Finally, we proposed a Boolean masking technique at the attack point, demonstrating the highest SCA resilience, preventing the unique recovery of the correct key even after 200K traces, and showing an improvement in minimum traces to disclosure (MTD) by $> 4000\times$.

II. Background and Related Work

A. Side-Channel Analysis (SCA) Attack

An SCA attack targets information unintentionally leaked during the cryptographic algorithm's execution. As shown in Fig. 1, these attacks involve analyzing side channels like power consumption, electromagnetic emissions, timing information, and other data related to the secret key.

Although the majority of published research on practical power or EM SCA focuses on block ciphers, there are relatively few studies presenting practical results of power SCA attacks on stream ciphers [9], [10]. Attacking most block ciphers usually requires targeting the first or last round. However, analyzing stream ciphers requires examining leaks across multiple rounds, making it more challenging. Additionally, when analyzing implementations of stream ciphers, particularly those that utilize feedback shift registers, it is common to integrate algebraic attacks with techniques from side-channel analysis to exploit potential vulnerabilities more effectively.

B. Architecture of SNOW-V

The SNOW-V cipher falls under the category of binary additive stream ciphers. It generates a keystream using two 256-bit Linear Feedback Shift Registers (LFSRs) and a Finite State Machine (FSM) [4].

The two LFSRs each consist of 16 elements from GF(2^{16}). LFSR A comprises elements a_{15},....,a_0 and generates a new a_{15} at each time step t according to the given expression

$a_{15}^{(t+1)} = b_0^{(t)} + \alpha a_0^{(t)} + a_1^{(t)} + \alpha^{-1} a_8^{(t)} \bmod g^A(\alpha)$.

Here, the generating polynomial $g^A(x) = x^{16}+x^{15}+x^{12}+x^{11}+x^8+x^3+x^2+x^1+1 \in GF(2)[x]$ and one of its roots α are used. Similarly, LFSR B consists of elements $b_{15}...b_0$ and uses the feedback function

$b_{15}^{(t+1)} = a_0^{(t)} + \beta b_0^{(t)} + b_3^{(t)} + \beta^{-1} b_8^{(t)} \bmod g^B(\beta)$.

Now β is a root of LFSR B's generating polynomial, which is expressed by $g^B(x) = x^{16}+x^{15}+x^{14}+x^{11}+x^8+x^6+x^5+x^1+1 \in GF(2)[x]$.

Each time we update the LFSR section, we clock LFSR-A and LFSR-B 8 times, meaning that 256 bits of the total 512-bit state are updated in a single step. Consequently, the two taps, T1 and T2, will have fresh values. Tap T1 is created by combining $(b_{15}, b_{14}, ..., b_8)$ into a 128-bit word, with b_8 as the least significant part. Similarly, Tap T2 is formed by combining $(a_7, a_6, ..., a_0)$ into a 128-bit word, with a_0 as the least significant part.

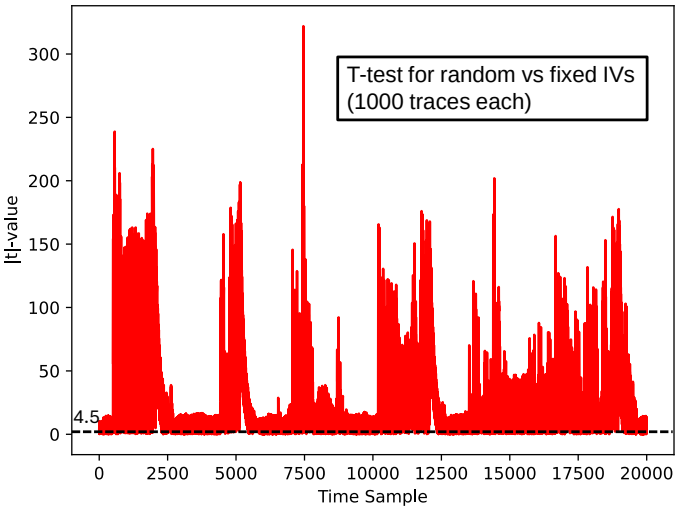

Fig. 2. Fixed-vs-random TVLA on the unprotected SNOW-V for 1K traces across time samples

The FSM includes three 128-bit registers (R1, R2, and R3), update, and output logic. The output logic generates 128-bit keystream outputs per FSM update. The LFSRs, which influence the update and output logic through taps T2 and T1, are updated eight times for each FSM update, ensuring the taps are fully refreshed each time. Both logic paths employ specific operators: First, the $\boxplus$ operator is defined as the parallel addition modulo 2^{32} of four 32-bit subwords. Second, the update logic uses two instances of the full AES round function, with the round keys C1 and C2 hardwired to 0^{128}.

C. Cryptanalysis of SNOW-V

SNOW-V has received much public evaluation after its publication, including guess-and-determine attacks [11], [12] and linear cryptanalysis [13], [14]. However, all these analyses are purely theoretical and cannot be implemented due to high time and memory complexities (both larger than 2^{240}). However, this work represents the first power side-channel attack (SCA) on SNOW-V with the objective of full key recovery.

III. Attack Methodology

A. Initial Findings

Our initial analysis of the SNOW-V architecture identifies the LFSR as a vulnerable point of attack due to its storage of keys and IVs during the initialization phase.

During initialization, all key bytes are stored in the LFSR, while the FSM uses zero values for constants C1 and C2, making the AES component irrelevant for attack as it only serves to randomize the sequence.

According to the initialization phase mentioned in the specification [4]

$$\begin{aligned}(a_{15}, a_{14}, ..., a_8) &\leftarrow (k_7, k_6, ..., k_0)\\ (a_7, a_6, ..., a_0) &\leftarrow (iv_7, iv_6, ..., iv_0)\\ (b_{15}, b_{14}, ..., b_8) &\leftarrow (k_{15}, k_{14}, ..., k_8)\\ (b_7, b_6, ..., b_0) &\leftarrow (0, 0, ..., 0)\end{aligned}$$

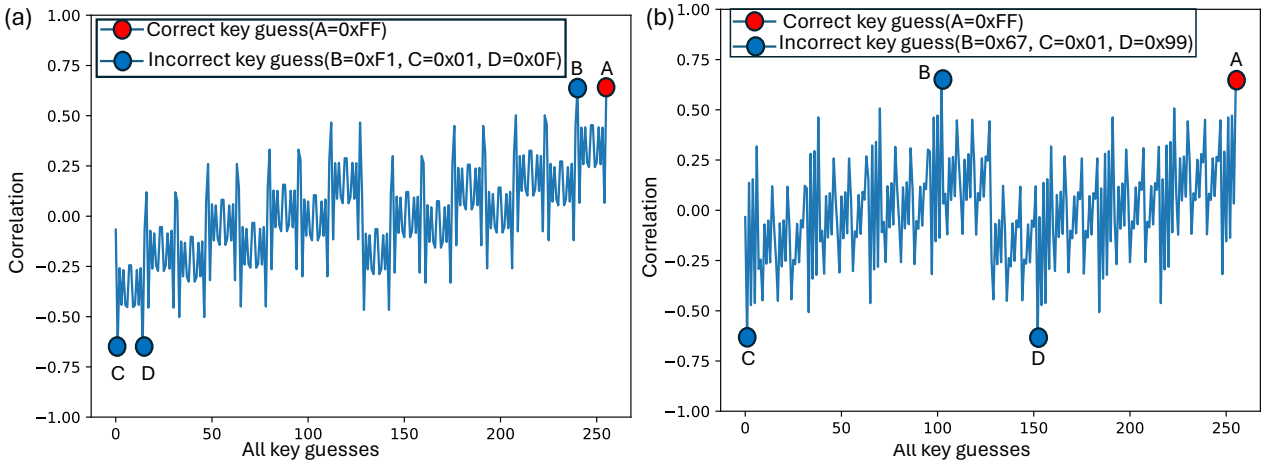

Fig. 3. (a) Simulated CPA on LSB of A[8] i.e., A[8][7:0] lower 8 bits of A[8] (b) Simulated CPA on MSB of A[8] i.e., A[8][15:8] upper 8 bits of A[8]

where the *secret key* $K = (k_{15}, k_{14}, ..., k_1, k_0)$, the IV = $(iv_7, iv_6, ..., iv_1, iv_0)$, and each of k_i, iv_j, $0 \leq i \leq 15, 0 \leq j \leq 7$, is a 16-bit vector.

These equations indicate that the LFSR, especially where the key is utilized, can be the main target for the attack.

According to [4], the parameter 'u' is defined as:

$$u = mul\_x(A[0], 0x990f) \oplus A[1] \oplus mul\_x\_inv(A[8], 0xcc87) \oplus B[0] \quad (1)$$

By examining the values of u and v across the eight iterations in the $lfsr\_update()$ function [4], we can progressively recover all 32 key bytes, starting from $A[8]$ which contains the first two bytes of the secret key.

From the equation of 'u' and $mul\_x\_inv()$ function [4], it is clear that during the u computation, information about the LSB of the key byte under attack A[8] ($A[8][0]$) is lost due to the 1-bit right shift within the $mul\_x\_inv()$ function. This affects the CPA attack, leading to multiple ghost peaks.

B. Proposed Side Channel Attack on SNOW-V

The primary target of the proposed Side Channel Attack on SNOW-V is the LFSR within the SNOW-V architecture.

Test Vector Leakage Assessment (TVLA) was performed on SNOW-V using both fixed and random sets of 1,000 traces to detect data-dependent side-channel leakage, assessing time points with $|t|$-values exceeding 4.5, as depicted in Fig. 2.

After fixing the potential attack point, we conducted a Known-Key Correlation (KKC) analysis to validate our attack model. This involved fixing the key and varying the IV to assess the Hamming Weight (HW) of all 16 bits of u. We then simplified the model to an 8-bit version to deduce the initial key byte. We opted for this approach because it reduced complexity from 2^{16} to 2^8 while maintaining a favorable signal-to-noise ratio (SNR).

After completing the KKC analysis, we applied CPA to target the u and v operations within the $lfsr\_update()$ function for extracting the secret key byte. To streamline the process, we first conducted CPA on simulated traces before presenting the measurement results.

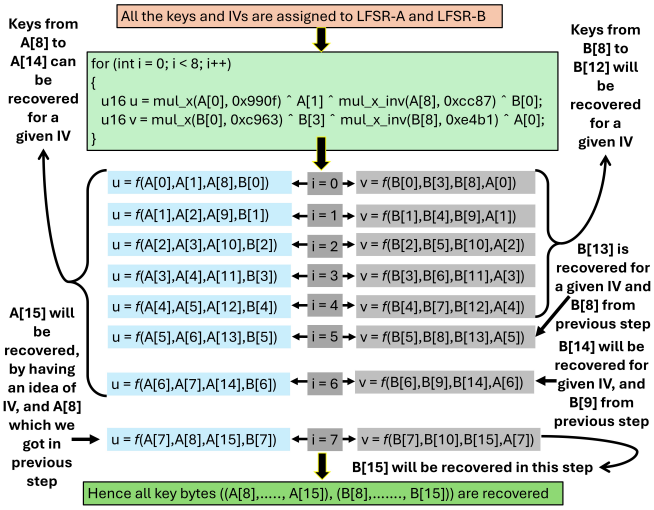

Fig. 4. A flowchart to recover all the key bytes of SNOW-V

In the simulated CPA (Fig. 3), we begin by computing the 16-bit u using the $lfsr\_update()$ function and calculating its Hamming Weight (HW), which serves as the basis for our simulated traces. With our 8-bit key hypothesis, we vary the key byte under attack $A[8]$ from 0 to 255 and compute the corresponding hypothetical 8-bit u.

In Fig. 3 (a), CPA was conducted on the lower 8-bits of $A[8]$, specifically A[8][7:0]. The analysis revealed four key guesses with the highest correlation: two positive peaks (A, B) and two negative peaks (C, D). Among these peaks, one corresponds to the correct key, while the remaining three are ghost peaks. Similarly, in Fig. 3 (b), CPA was performed on the MSB of $A[8]$, A[8][15:8] to identify the correct key guess. A comprehensive case study analyzing ghost peaks is demonstrated in [15], which provides an in-depth analysis of this phenomenon. The study delves into the detailed examination and characterization of ghost peaks observed during CPA, which led to the successful extraction of the entire key.

Among the four identified peaks (Fig 3), one corresponds to the correct key, while the remaining three are ghost peaks. The 32-bit ARM Cortex-M4 microcontroller pre-charges its data bus to zero, resulting in CPA showing positive peaks for the

correct key byte. Consequently, we can ignore the two negative ghost peaks (C and D). However, accurately determining the correct key byte among the two positive peaks (A and B) remains challenging.

To differentiate between A and B, we applied Linear Discriminant Analysis (LDA) by modeling each trace based on the LSB of A[8]. We used A[8][8] for the upper 8 bits and A[8][0] for the lower 8 bits of A[8]. In recent SCA attacks, LDA has been used successfully in profiled SCA [16], [17].

In the LDA model, traces are initially gathered and labeled as 0 or 1 depending on the LSB of $A[8]$. This labeled dataset is then employed to train the LDA model, which is subsequently evaluated on fresh data to assess its accuracy. This methodology aims to eliminate the false peak (B) observed after CPA, enabling the precise identification of the correct secret key byte (A).

C. Incremental Attack to Recover All Key Bytes

We can progressively recover all correct key guesses by employing an incremental attack approach. This method allows us to systematically and methodically extract each key byte, ensuring that the entire key is accurately reconstructed over each iteration.

Fig. 4 outlines the detailed steps required to retrieve all potential key guesses. Following these steps, we can incrementally refine our attack to recover the entire set of correct key values accurately. From Fig. 4, it is evident that after the fourth iteration (i.e., for $i \geq 5$), there will be an XOR operation between two 16-bit words of the same keys. One of these keys will already be known from the previous attack steps (i.e., during $i \leq 4$). Therefore, the XOR operation between the two 16-bit words, with one value already known, will yield the other unknown value.

In the initial iteration of the LFSR update function ($i = 0$), as detailed in Fig. 4, the calculations for u and v involve performing XOR operations on both the initialization vector (IV) and the key. Specifically, these operations target the values in $A[8]$ and $B[8]$, respectively. Similarly, for iterations $i = 1, 2, 3, 4$, the equations for u and v require performing XOR operations that involve both the IV and the keys located in specific positions within the arrays. These positions are $A[9]$, $A[10]$, $A[11]$, and $A[12]$ for the LFSR-A and $B[9]$, $B[10]$, $B[11]$ and $B[12]$ for the LFSR-B.

In iteration $i = 5$, the u equation involves XORing the IV with the key in $A[13]$. Meanwhile, the v equation XORs the IV with two 16-bit words: one from the known key in $B[8]$ (from i = 0) and the other from $B[13]$, making the unknown key in $B[13]$ recoverable in this iteration. Similarly, for $i = 6$, the u equation involves a straightforward XOR operation between the IV and the key in $A[14]$. However, the v equation XORs the IV with two 16-bit words of the same key: one from the known key in $B[9]$ (identified in the second iteration, i = 1) and the other from $B[14]$. This allows us to recover the unknown key in $B[14]$.

In the final iteration ($i = 7$), the u equation involves XORing the IV with two 16-bit words: one from the known key in $A[8]$ (determined in the first iteration, i = 0) and the other from $A[15]$. Because $A[8]$ is already known, this iteration enables us to discover the previously unknown key in $A[15]$. Similarly, for the v equation, XOR operations are conducted between the IV and two 16-bit words of the same key: one from $B[10]$ and the other from $B[15]$. Since $B[10]$ was determined in the third iteration (i = 2), this allows us to now identify the previously unknown key in $B[15]$. Thus, the final unknown key can be accurately recovered through these operations by leveraging the known value from an earlier iteration.

Upon completing all iterations, we successfully retrieved all the key bytes located in both LFSR-A (ranging from $A[8]$ to $A[15]$) and LFSR-B (spanning from $B[8]$ to $B[15]$). As a result, the incremental attack proves to be highly effective in systematically recovering every key byte. This comprehensive approach ensures that each key byte is accurately identified, demonstrating the robustness and efficiency of the incremental attack method.

```
main()
{
//Assigning iv1, iv2, key1 and key2
A1 = f(iv1, key1);
A2 = f(iv2, key2);
B1 = f(key1, 0);
B2 = f(key2, 0);
//LFSR-A = A1 ^ A2
//LFSR-B = B1 ^ B2

keystream_mask(); //Calling this function once
}
void keystream_mask(void)
{
 fsm_update_mask();
 lfsr_update_mask(); //Function to be attacked
}
void lfsr_update_mask(void)
{
 //Two Boolean shares of sensitive variable u as (u1
    ,u2).
 (p1,p2) = mul_x_mask((A1[0], A2[0]), 0x990f);
 (q1,q2) = SecXor((p1,p2), (A1[1], A2[1]));
 (r1,r2) = mul_x_inv_mask((A1[8], A2[8]), 0xcc87);
 (s1,s2) = SecXor((r1,r2), (B1[0], B2[0]));
 (u1,u2) = SecXor((q1,q2), (s1,s2));
}
//mul_x_mask() checks for MSB, if MSB is 1, then it
    does a left shift by 1 and XOR with 0x990f, else
     it does only a left shift by 1.
//mul_x_inv_mask() checks for LSB, if LSB is 1, then
     it does a right shift by 1 and XOR with 0xcc87,
     else it does only a right shift by 1.
//SecXor function performs a simple XOR operation on
     its two input parameters.
//Attacker can attack either of  u 1   or u2, which
    will not  r e v e a l  any secret key.
```

Listing 1. Pseudo code for masked implementation of SNOW-V

D. Proposed masking on unprotected SNOW-V

Masking [18] is a provable secure countermeasure to protect cryptographic implementations from passive side-channel attacks, such as CPA. This technique involves masking the sensitive data, i.e., splitting the sensitive data into multiple shares and performing the operations on these shares independently. By using random masks that change with each execution, masking mitigates the correlation between side-

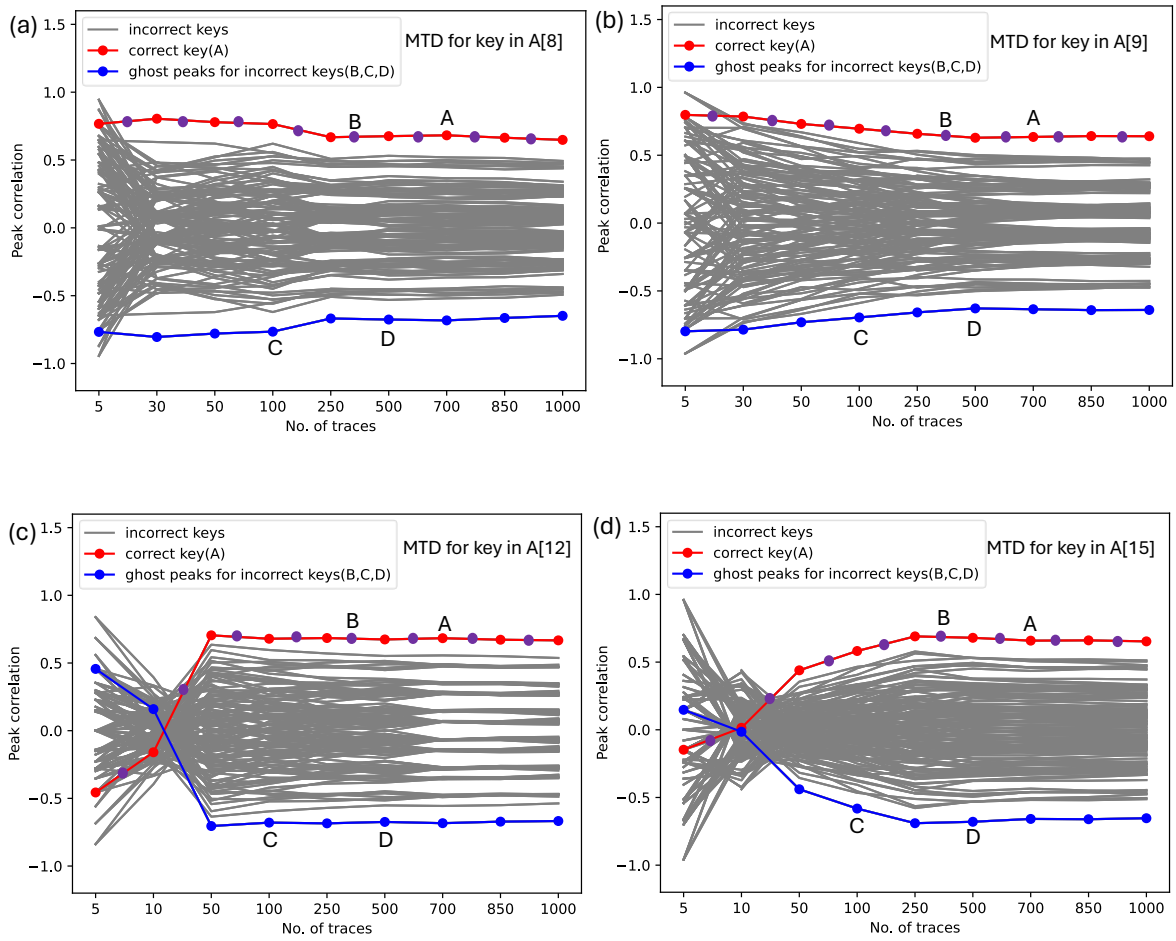

Fig. 5. MTD plot (measured) for the unprotected SNOW-V on lower 8 bits of (a) Key byte in A[8] (b) Key byte in A[9] (c) Key byte in A[12] (d) Key byte in A[15]

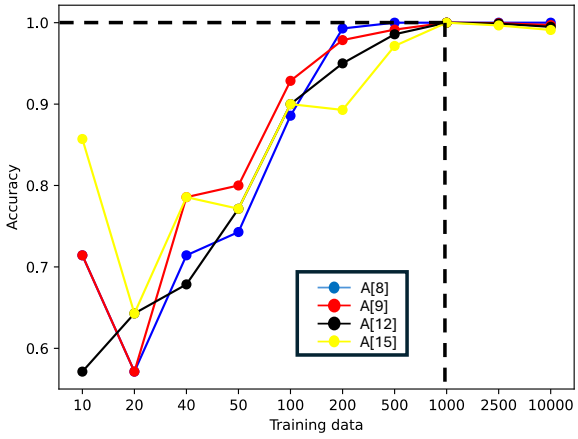

Fig. 6. Training accuracy of the LDA model for identifying the LSB of some key bytes under attack. It achieves more than 99% accuracy after training with 1K traces

channel leakage and sensitive information, thereby enhancing the security of cryptographic algorithms.

Boolean masking is a robust countermeasure employed to safeguard SNOW-V implementations against SCA. It involves partitioning sensitive variables like x, derived from interactions with the secret key, into two shares ($x1$ and $x2$, satisfying $x1 \oplus x2 = x$). Subsequently, cryptographic operations are executed independently on these shares to prevent leakage of sensitive information through side channels. Integration of this masking countermeasure for the entire algorithm increases the runtime of the entire cryptographic algorithm by $> 2\times$. However, for 5G ciphers like SNOW-V, speed is very crucial. Therefore, we have masked the most critical operations, which we were able to exploit during our attack. Hence, first-order masking has been implemented on the $lfsr\_update()$ function to secure SNOW-V with less performance overhead.

Listing 1 illustrates the pseudo-code for the masked implementation of SNOW-V. Initially, IVs (i.e., $iv1$ and $iv2$) and keys (i.e., $key1$ and $key2$) are generated. These values are then assigned to A1, A2, B1, and B2, as described in Section III.

In this implementation, the LFSR masks are designed such that the XOR operation between A1 and A2 yields LFSR-A, and similarly, the XOR operation between B1 and B2 results in LFSR-B. This ensures that each LFSR value is split into two masked components, with the IV splitting into two shares, $iv1$, and $iv2$, and the key splitting into $key1$ and $key2$.

The $lfsr\_update()$ function, which serves as our attack

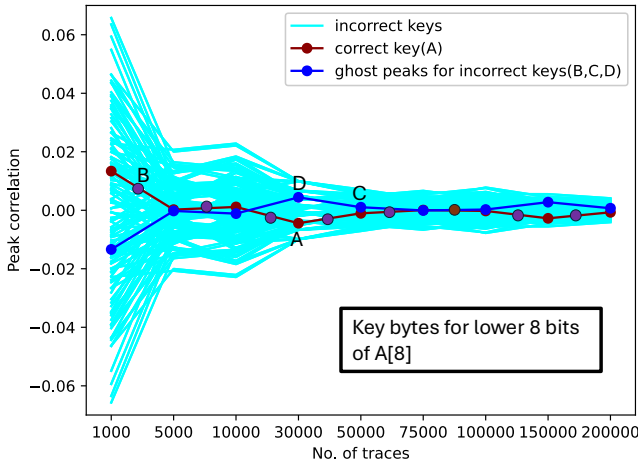

Fig. 7. The measured CPA results on the masked SNOW-V implementation indicate that the secret key byte of A[8] could not be extracted even after analyzing $200K$ traces

target, now comprises two equations, $u1$ and $u2$. These equations are constructed such that XOR-ing $u1$ and $u2$ results in u. Consequently, the attacker cannot directly target u since it is split into $u1$ and $u2$, and u itself is not exposed. While the attacker might attempt to target either $u1$ or $u2$, this approach would not reveal any part of the secret key.

The proposed implementation with Boolean masking takes 17 ms when initializing the SNOW-V core with a key and IV, compared to 11 ms for the unprotected version. This results in a performance overhead increase of approximately $\sim 55\%$.

IV. Experimental results

The proposed SNOW-V stream cipher architecture was analyzed using CPA to determine the correct key bytes. The CPA results showed ghost peaks for incorrect keys due to the LSB shift of A[8]. The MTD for CPA on measured SNOW-V traces shows that the correct key is recovered with < 50 traces for most of the key bytes (Fig. 5).

The MTD plot (Fig. 5) demonstrates that, in the case of positive correlation, there is an overlap between one incorrect key (B) and the correct key (A). To uniquely identify the correct key byte, we initially gathered power traces by varying both the key and the IV. Each trace was modeled based on the LSB of the key byte under attack, specifically focusing on bits such as A[8], A[9], A[12], and A[15]. The dataset containing 10,000 traces was divided into 70% for training and 30% for testing. The trained LDA model successfully predicts the LSB of the key byte under attack with more than 99% accuracy after training with 10K traces (Fig. 6).

Fig. 7 shows the measured CPA results on the masked SNOW-V implementation. The results clearly show that the correct key present in the lower 8 bits of A[8], i.e., A[8][7:0], could not be determined even after analyzing with 200K traces, indicating $> 4000\times$ SCA security improvement.

V. Conclusion

In summary, this paper introduces the first power SCA on the 5G standard candidate SNOW-V, successfully demonstrating that by employing a combined CPA and LDA attack on the SNOW-V implementation running on a 32-bit ARM Cortex-M4 microcontroller, the secret key can be fully recovered. During CPA, the LSB for the byte under attack remains undetermined due to an intermediate operation ($mul\_x\_inv()$ within the $lfsr\_update()$). To address this, an LDA model is trained to predict the LSB based on the branching condition in the $mul\_x\_inv()$ function, achieving more than 99% accuracy with 10,000 training traces.

Additionally, we demonstrate the effectiveness of an incremental attack technique that systematically recovers all key bytes of SNOW-V, showcasing how each iteration progressively reveals the complete secret key. Overall, using our proposed attack strategy, most of the secret key bytes were successfully recovered in fewer than 50 traces, demonstrating the effectiveness and efficiency of the combined CPA and LDA approach in breaking the SNOW-V implementation. Finally, we demonstrated the efficacy of boolean masking, showing a $> 4000\times$ MTD improvement over the unprotected SNOW-V.

References

[1] J. Yang and T. Johansson, "An overview of cryptographic primitives for possible use in 5G and beyond," *Sci. China Inf. Sci.*, vol. 63, p. 220301, Nov. 2020.

[2] A. Caforio, F. Balli, and S. Banik, "Melting SNOW-V: improved lightweight architectures," *J Cryptogr Eng*, vol. 12, pp. 53–73, Apr. 2022.

[3] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and Systems Aspects; Study on the support of 256-bit algorithms for 5G (Release 16). 3GPP TR 33.841 V0.7.0)," tech. rep., 2018.

[4] P. Ekdahl, T. Johansson, A. Maximov, and J. Yang, "A new SNOW stream cipher called SNOW-V," *IACR Transactions on Symmetric Cryptology*, pp. 1–42, Sept. 2019.

[5] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing*, STOC '96, (New York, NY, USA), pp. 212–219, Association for Computing Machinery, July 1996.

[6] R. D. Koninck, *Enhancing 5G Security: A Comparison of 256-bit Symmetric-key Cryptosystems on FPGA*. PhD thesis, KU Leuven, 2023. Bart Preneel (promotor).

[7] 3GPP, "3rd generation partnership project; technical specification group services and system aspects; security architecture and procedures for 5g system (release 18). 3gpp ts 33.501 v18.0.0."," tech. rep., 2020.

[8] ETSI SAGE, "256-bit algorithms based on SNOW 3G or SNOW V (S3-211407)," tech. rep., 2021.

[9] S. Kumar, V. A. Dasu, A. Baksi, S. Sarkar, D. Jap, J. Breier, and S. Bhasin, "Side Channel Attack On Stream Ciphers: A Three-Step Approach To State/Key Recovery," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 166–191, Feb. 2022.

[10] D. Strobel, "Side Channel Analysis Attacks on Stream Ciphers," Master's thesis, Ruhr-Universität Bochum, 2009.

[11] L. Jiao, Y. Li, and Y. Hao, "A guess-and-determine attack on SNOW-V stream cipher," *Comput. J.*, vol. 63, no. 12, pp. 1789–1812, 2020.

[12] J. Yang, T. Johansson, and A. Maximov, "Improved guess-and-determine and distinguishing attacks on SNOW-V," *IACR Trans. Symmetric Cryptol.*, vol. 2021, no. 3, pp. 54–83, 2021.

[13] Z. Zhou, D. Feng, and B. Zhang, "Efficient and extensive search for precise linear approximations with high correlations of full SNOW-V," *Des. Codes Cryptogr.*, vol. 90, no. 10, pp. 2449–2479, 2022.

[14] Z. Shi, C. Jin, J. Zhang, T. Cui, L. Ding, and Y. Jin, "A correlation attack on full SNOW-V and snow-vi," in *Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III* (O. Dunkelman and S. Dziembowski, eds.), vol. 13277 of *Lecture Notes in Computer Science*, pp. 34–56, Springer, 2022.

[15] H. Saurabh, A. Golder, S. S. Titti, S. Kundu, C. Li, A. Karmakar, and D. Das, "SNOW-SCA: ML-assisted Side-Channel Attack on SNOW-V," 2024. HOST 2024.

[16] J. Danial, D. Das, A. Golder, S. Ghosh, A. Raychowdhury, and S. Sen, "Em-x-dl: Efficient cross-device deep learning side-channel attack with noisy em signatures," *J. Emerg. Technol. Comput. Syst.*, vol. 18, sep 2021.
[17] O. Choudary and M. G. Kuhn, "Efficient Template Attacks," in *Smart Card Research and Advanced Applications* (A. Francillon and P. Rohatgi, eds.), Lecture Notes in Computer Science, (Cham), pp. 253–270, Springer International Publishing, 2014.
[18] S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi, "Towards sound approaches to counteract power-analysis attacks," in *Advances in Cryptology — CRYPTO' 99* (M. Wiener, ed.), (Berlin, Heidelberg), pp. 398–412, Springer Berlin Heidelberg, 1999.

Harshit Saurabh is currently pursuing PhD at the Department of Electronic Systems Engineering (DESE), Indian Insitute of Science (IISc), Bangalore. He completed his Bachelor of Technology in Electronics and Communication Engineering from National Institute of Science and Technology (NIST), Odisha in 2023. His research interest includes hardware security.

Suparna Kundu is pursuing her PhD at COSIC, the Department of Electrical Engineering (ESAT) from Katholieke Universiteit Leuven (KU Leuven), Belgium. Before that, she completed her Master of Technology in Cryptology and Security in 2020 from the Indian Statistical Institute (ISI), Kolkata, India. Her research interests include post-quantum cryptography, side-channel attacks, and its countermeasures.

Samarth Shivakumar Titti is currently pursuing his PhD at the Department of Electronic Systems Engineering (DESE) at Indian Insitute of Science (IISc), Bangalore. He has completed his Bachelors of Engineering with honors in Electronics and Communication from Ramaiah Institute of Technology, Bangalore in 2023. His research interests include mixed signal IC design, biomedical circuits and hardware security.

Anupam Golder received the M.S. and Ph.D. degree in Electrical and Computer engineering from Georgia Institute of Technology in 2020 and 2023, respectively. Prior to that, he received the B.Sc. degree in Electrical and Electronic Engineering from Bangladesh University of Engineering and Technology in 2015.

Since May 2023, Anupam Golder has been a research scientist at Intel Labs. His research interests include hardware acceleration of cryptographic schemes and physical side-channel vulnerability analysis of such implementations.

K K Soundra Pandian received the Ph.D. degree from IIT Patna and worked as a postdoctoral fellow at New York University.

He is currently serving as a Scientist in the Ministry of Electronics and Information Technology, Office of CCA, Government of India. Prior to this, he worked as a Research Scientist at the Indian Institute of Information Technology Design and Manufacturing Jabalpur, as a Research Scientist (Grade II) at the Indian Institute of Technology (IIT) Kanpur, and as a Project Engineer at the Center for Electronics and Design and Technology of India, Ministry of Communications and Information Technology. He has authored/co-authored more than 40 peer-reviewed conferences and journals, including 5 patents.

Chaoyun Li received the PhD degree in electrical engineering from COSIC, Department of Electrical Engineering (ESAT), KU Leuven, Belgium in February 2020. Prior to that he received the B.S. and M.S. degrees in mathematics from Hubei University, Wuhan, China, in 2012 and 2015, respectively.

Since March 2023, Chaoyun Li has been a lecturer of the Surrey Centre for Cyber Security at University of Surrey, UK. He has worked as a postdoctoral researcher at KU Leuven from 2020 to 2023, which was funded by an FWO postdoc fellowship. His research interests include cryptography, security and privacy.

Angshuman Karmakar received the B.E. degree in computer science and engineering from Jadavpur University, Kolkata, India, in 2010 the M.Tech. degree in computer science and engineering from the Indian Institute of Technology, Kharagpur, India, in 2012 and the Ph.D. degree from Katholieke Universiteit Leuven (KU Leuven), Belgium, in 2020 for his dissertation titled "Design and Implementation Aspects of Post-Quantum Cryptography." He is one of the primary designers of the post-quantum Saber KEM scheme which is one of the finalists in the NIST's post-quantum standardization procedure. He is currently working as an assistant professor at the Indian Institute of Technology, Kanpur, in India, and as a free researcher at COSIC, KU Leuven, Belgium. Earlier he was an FWO Post-Doctoral Fellow with the COSIC Research Group, KU Leuven. His research interest spans different aspects of lattice-based post-quantum cryptography and computation on encrypted data.

Debayan Das received his PhD and MS in Electrical and Computer Engineering from Purdue University, USA, in 2021 and his Bachelor of Electronics and Telecommunication Engineering degree from Jadavpur University, India, in 2015.

He is an Assistant Professor with the Department of Electronic Systems Engineering (DESE) at the Indian Institute of Science (IISc), Bangalore. He has worked as a Security Researcher at Intel, USA, during 2021-22 and as a Research Scientist in the Intel Labs, USA, during 2022-23. Before his Ph.D., he worked as an Analog Design Engineer at a startup based in India. His research interests include mixed-signal IC design, biomedical circuits, and hardware security. He has authored/co-authored more than 60 peer-reviewed conferences and journals, including 2 book chapters and 3 US patents.