Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 7, No. 3, 2024

Side-Channel Attacks Against the FESH Algorithm

Jiajun Fan *, Zhibo Du

School of Cybersecurity (Xin Gu Industrial College), Chengdu University of Information Technology, Chengdu Sichuan, 610225, China
*** Corresponding author:** Jiajun Fan (Email: muyuluqxl@gmail.com)

Abstract: The FESH algorithm is a block cipher algorithm based on finite field operations. Currently, no research has been conducted on its side-channel attack security. Therefore, this study proposes two methods to address this issue: a correlation power analysis attack method targeting the FESH algorithm, and a template attack method based on an improved TransNet model. The first method theoretically analyzed the vulnerabilities of the FESH algorithm and successfully obtained valid leaked information through a correlation power attack; The second method introduced BlurPool blurring and downsampling techniques, as well as normalization operations, which reduced the training parameters of the improved model by approximately 50%. Additionally, the validation was performed on both the FESH dataset and the desynchronized ASCAD public dataset, which provided evidence that the entropy estimates were significantly better than those of the original TransNet model. The experimental results highlight the importance of considering side-channel security when implementing the FESH algorithm.

Keywords: FESH Algorithm; Correlation Power Analysis Attack; Template Attack; TransNet Model; ASCAD.

1. Introduction

The Cryptography algorithms play an important role in network security by providing support and protection for data confidentiality, identity authentication, and access control. The security of cryptography algorithms not only has a significant impact on data protection but also affects trust relationships between users and systems, socio-economic costs, the formulation of laws and regulations, national security, and strategic interests. However, all cryptography algorithms used in modern cryptographic systems are based on computational security. After Paul introduced side-channel attacks [1], the security of cryptography not only requires attention to the security of algorithm design but also to the security of its implementation. Traditional encryption algorithms typically hide the relationship between the key and plaintext. Side-channel attacks, on the other hand, target cryptographic systems from the perspective of physical information leakage. By monitoring and analyzing the variations in physical signals, these attacks can infer the operation process of the encryption algorithm and related secret information. Attackers can obtain the key, plaintext, or other sensitive information by exploiting side-channel leakage differences such as processing time differences, power consumption variations, and different input conditions without directly cracking the encryption algorithm itself.

Side-channel attacks can be divided into two types: non-learning and learning. Non-learning methods, such as Differential Power Analysis (DPA) [2] and Correlation Power Analysis (CPA) [3], are based on information theory and statistical analysis and can provide interpretable attack results. Attackers can adjust their attack strategies to obtain useful information. Learning-based attack methods use feature learning to establish templates for attacks. Traditional template attacks [4] use multivariate Gaussian distribution to characterize power consumption features. With the development of neural networks, introducing neural networks to side-channel attacks breaks the linear correlation assumption between power consumption and Hamming distance in traditional side-channel attacks and has advantages such as non-linear modeling, adaptability, and generalization. In recent years, many studies have combined neural networks with side-channel attacks, using different models such as Least Squares Support Vector Machines (LS-SVM) [5], K-Nearest Neighbors (KNN) [6], CNN, VGG [7], etc., to establish templates or leakage models. In 2017, the attention[8] mechanism was used in machine translation tasks and achieved good results, becoming a new generation of deep learning architecture. In 2022, TransNet [9] introduced the long-term attention mechanism [10] to side-channel attacks. TransNet can reduce the guessing entropy to below 1 using only 400 records in multiple desynchronized datasets and has achieved significant results on several datasets.

FESH [11] is a new cryptographic algorithm that has performed well in the Chinese National Cryptographic Algorithm Design Competition and has entered the second round of evaluation for block algorithms. The main idea and strategy behind the design of the FESH algorithm are to meet the basic requirements of block cipher algorithms and efficiently implement them on different software and hardware platforms. The FESH algorithm has also undergone security and performance analyses, and the designers have conducted self-assessments of the cryptographic algorithm. For the security evaluation of the FESH algorithm, it is necessary to analyze the security of the cryptographic design from both the data and implementation aspects. Since the current evaluations are based on computational security, this paper focuses on the security of FESH algorithm from the perspective of side-channel attacks, specifically non-learning CPA attacks and learning-based template attacks using an improved TransNet model. The research on the side-channel security of the FESH algorithm can evaluate its resistance to side-channel attacks and identify potential vulnerabilities and weaknesses in the algorithm implementation. This research can provide guidance and suggestions for developers and system designers, helping them understand potential side-channel issues in algorithm implementation, ultimately improving the security of the algorithm, and contributing to the research of corresponding defense strategies and remediation measures, thus reducing the risk of side-channel attacks on systems. Additionally, this research is of great significance for protecting user privacy, improving product

security, ensuring the security and advancement of cryptographic products and systems, and enhancing the commercial cryptographic market system.

2. Introduction to Relevant Knowledge

2.1. FESH Algorithm

FESH is a block cipher algorithm based on finite field arithmetic. It has multiple versions, and each version of the FESH algorithm has different block lengths, key lengths, and specific rounds.

Table 1. FESH algorithm version and number of rounds

Algorithm version	Plaintext Block Length	Key Block Length	Number of Rounds
FESH-128-*	128 bit	128/192/256 bit	16/20/20
FESH-256-*	256 bit	256/384/512 bit	24/28/28

The encryption process of the FESH algorithm is based on word operations. It converts the input bit string into words and uses words as the basic unit for encryption. The FESH algorithm adopts the SPN structure, and the specific operations include S-box lookup, word operations, and round key addition. The bit-slice technique is also employed for parallel computation of the S-box. The pseudo code of the FESH algorithm is as follows:

Input: Plaintext (P), Key (K)

Output: Ciphertext (C)

a) RK = gen(K) // Key schedule function gen().

b) X[0] = P[0] xor RK[0] // Xor the plaintext and subkeys in the first round of grouping.

c) i = 0 // The i represents the current round number.

d) Y[i] = SubNibble(X[i]) // Apply S-box substitution to X[i] in this round.

e) Z[i] = MixWord(Y[i]) // Perform word mixing on Y[i] in this round.

f) X[i+1] = Z[i] xor RK[i+1] // Xor Z with the next round key.

g) i++; if i < N-1: go to step d // If the specified number of rounds is not reached, go back to step d).

h) C = X[n] // The XOR result after the final round is the ciphertext C.

2.2. Correlation Power Analysis Attack

Correlation Power Analysis (CPA) [3] is a side-channel attack method that utilizes a linear bit transformation model under the assumption of an ideal state:

$$W = aH(D,R) + b \tag{1}$$

$H(D,R)$ represents the mapping relationship of the number of bit flips from D to R, W represents the consumed power, a,b represents the correlation coefficient, and represents the bias. The correlation factor between the Hamming distance and the measured power can be derived from classical statistics using Pearson correlation. The correlation factor is used to assess and quantify the level of information leakage in side-channel attacks. Other common leakage metrics include Normalized Inter-Class Variance and Signal-Noise Ratio. In the equation, cov represents the covariance function, and ρ represents the standard deviation.

$$\rho_{WH} = \frac{\mathrm{cov}(W,H)}{\sigma_W \sigma_H} = \frac{a\sigma_H}{\sigma_W} = \frac{a\sigma_H}{\sqrt{a^2\sigma_H^2 + \sigma_b^2}} \tag{2}$$

Through analysis, it can be determined that $|\rho_{WH}| > |\rho_{WH'}|$. Therefore, the method of inferring sensitive data by iterating over the value space of leakage and obtaining the extreme value of the correlation factor is known as correlation power analysis (CPA) attack.

2.3. TransNet Neural Network

Hajra et al. introduced Transformer-XL into side-channel analysis [9]. Transformer-XL is a variant of the Transformer network that addresses the limitation of context length in learning long-term dependencies. The model proposes a segment mechanism, which improves the ability of the multi-head attention mechanism to capture information. For side-channel data, the segment mechanism uses relative positional encoding to better extract temporal features. Building upon this, TransNet improves upon the model and achieves state-of-the-art results on multiple public datasets. Specifically, in the encoding part based on Transformer-XL, TransNet reduces the use of the normalization layer and introduces an offset in the MultiHead Attention with TransformerXL, transforming it into Relational Positional MultiHead Attention. Additionally, TransNet adds an Addition layer after the PositionWiseFF module following the feed-forward network.

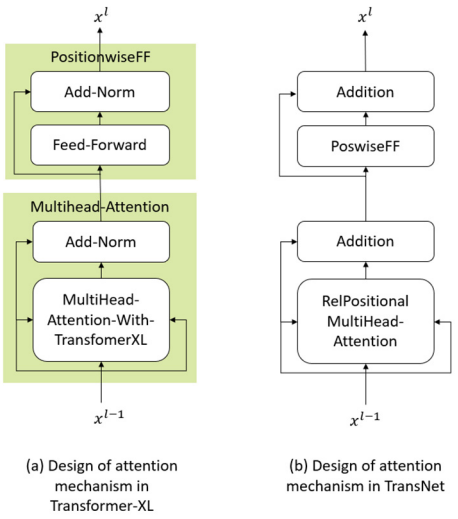

Figure 1. Adjustments made by TransNet on TransformerXL

2.4. BlurPool [12]

BlurPool is a convolutional neural network technique used for image processing. It is primarily used to reduce image resolution and computational complexity. BlurPool achieves this by blurring the image, thereby reducing the utilization of computational resources while preserving image information. Traditional pooling operations like max pooling or average pooling can help reduce the size of the image but may result in the loss of fine details in some cases. On the other hand, BlurPool applies a blur filter to the image before performing the pooling operation, allowing for the retention of more details while reducing the size. In side-channel analysis (SCA), CNNs are often employed to achieve translation invariance. For a function $\mathcal{F}$:

$$\mathcal{F}(X) = \mathcal{F}\left(\mathrm{Shift}_{\Delta h,\Delta w}(X)\right) \quad \forall(\Delta h,\Delta w) \tag{3}$$

From the research in this paper, it can be inferred that the equation mentioned above holds true only when the values of Δh and Δw are integer multiples of the translation amount. In such cases, it is referred to as periodic translation invariance.

BlurPool combines low-pass filtering with anti-aliasing to achieve its effect. It first applies a blur convolution kernel to blur the feature map and then downsamples it. The combination of blurring and downsampling operations allows for the reduction of resolution while retaining more detailed information.

3. Side-Channel Analysis of the FESH Algorithm

3.1. Correlation Power Analysis of the FESH Algorithm

3.1.1. Leakage Point Analysis of the FESH Algorithm

Analysis of the power consumption model of the FESH algorithm reveals potential attack and information leakage points in the S-box input, S-box output, and MixWord output. The leakage function during the attack is as follows:

$$d_r = f(p_i, k_j) \tag{4}$$

In the leakage function formula, p_i represents the i bit taken from the plaintext, k_j represents the j bit taken from the key, p_i and k_j the mapping relationship constructs the leakage function output as d_r. First, at the S-box input position, $X[0] = P\ xor\ RK[0]$ based on the analysis of the FESH-128-128 algorithm, each round of is 128 bits. After the word operation transformation, it generates four 32-bit values, namely x_0, x_1, x_2, x_3, P and RK. Similarly, the methods for obtaining the values have been elaborated in detail in Section 2.1 of this paper.

$$x_i = p_i \otimes rk_i \tag{5}$$

From the equation, it can be observed that the intermediate values x_i and rk_i have a strong correlation at the S-box input position. At the S-box output position, due to the parallel computation used in the S-box substitution of the FESH algorithm, it is possible to obtain:

$$\begin{aligned} T_1 &= x_2 \oplus (x_1 \vee x_0) \oplus x_3 \\ T_2 &= x_3 \oplus \{x_1 \wedge [x_0 \oplus (x_3 \vee x_2)]\} \end{aligned} \tag{6}$$

$$\begin{aligned} y_0 &= x_0 \oplus (x_3 \vee x_2) \oplus x_1 \\ y_1 &= x_1 \oplus T_1 \\ y_2 &= T_1 \oplus [(\sim y_0) \vee T_2] \\ y_3 &= T_2 \oplus [(\sim y_0) \vee T_2] \end{aligned} \tag{7}$$

y_i is determined by $x_{0\sim3}$. (Similarly, z_i for the word operation result Z, it is determined by $y_{0\sim3}$.) By analyzing the leakage of the S-box input, S-box output, and MixWord output using the correlation factor, the correlation between the intermediate sequence x_i, y_i, z_i and the power consumption curve can be calculated by substituting the plaintext sequence and the correct key. A higher correlation factor indicates a stronger linear relationship between sensitive information and the measurement values, which is a necessary condition for further attacks.

3.1.2. CPA Attack Method on the FESH Algorithm

The steps for performing a CPA attack on the FESH algorithm are as follows:

a) Power Trace Acquisition: For the implemented FESH encryption algorithm, randomly select plaintext sequences P and input them into the encryption chip. The measurement device will generate power consumption samples T based on the measured power traces over time.

b) Leakage Point Analysis: Substitute the plaintext sequence P and the key value K into the encryption algorithm to obtain the intermediate value sequence x_i, y_i, z_i for the first round. Calculate the correlation factor between the intermediate value sequence and using the leakage metric to identify the leakage point in the power consumption samples.

c) Guessing Key Correlation Coefficients: Traverse the key space to obtain a guessed key K'. Calculate the intermediate value H' based on the leakage point obtained from the power consumption samples using the plaintext sequence P and K'. This results in $\rho_{WH'}$.

d) Obtaining the Key: Sort the guessed correlation coefficients in descending order. The corresponding key sequence is the ranking of the possible keys obtained through the CPA attack. The comparison and analysis of the experimental results will be discussed in detail in Section 4.3.

3.2. Design and Improvement of the TransNet Model

3.2.1. Adjustments to the TransNet Network

Improvements were made to the TransNet model while retaining its multi-head attention mechanism design. The adjustments made to the model are as follows:

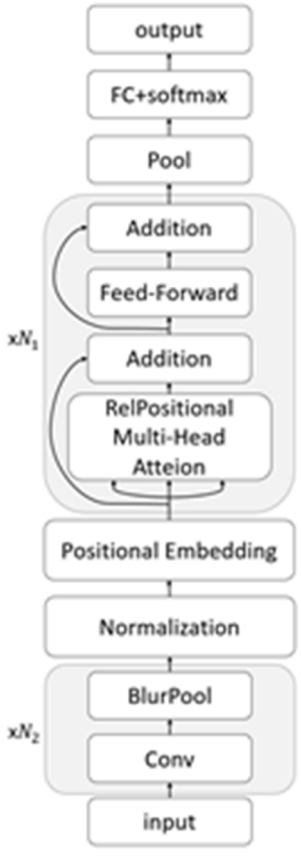

Figure 2. Schematic diagram of an improved TransNet model structure

Firstly, in the convolutional layer, the AvgPool operation was replaced with BlurPool. TransNet demonstrated shift invariance on multiple datasets, which is an important property for handling asynchronous trajectories. The improved model considers this as an expression of periodic translation invariance and adopts BlurPool to provide more detailed trajectory information to the self-attention

mechanism.

Secondly, the improved model incorporates data normalization. While TransNet inputs data into a CNN, convolutional operations alone cannot replace feature normalization. Modeling attacks like Side-Channel Analysis (SCA) can be viewed as a temporal classification task, and the effectiveness of normalization techniques has been demonstrated in SCA literature through multiple examples.

Lastly, the improved model adds a positional embedding layer and modifies the PoswiseFF to Feed-Forward, building upon the TransNet model. The positional embedding layer provides position information, and the modification in PoswiseFF enhances the feed-forward operation.

3.2.2. Template Attack Method Based on the Improved TransNet Model

The improved TransNet model is used for training in this paper, and the main steps of the attack are as follows:

a) Label Generation: Based on the collected power consumption samples of the FESH encryption algorithm, a desynchronization strategy is chosen (with maximum offsets of 0, 50, or 100). The power traces T_i and corresponding plaintexts P_i are used to generate labels L_i based on the leakage point positions. The dataset is then divided into training, validation, and test sets (where i represents each sample in the dataset).

b) Template Construction: Using the improved TransNet model with the hyperparameters of the original model, the hyperparameters of the BlurPool layer are randomly initialized, and zero-mean normalization is added after the convolution operation. The model is trained and the template is established on the training and validation sets.

c) Prediction and Attack: Predicted value rankings V_{ij} (where j represents the predicted value sequence for the $i-th$ sample, ranging from 0 to 255) are generated on the test set. Multiple rankings are averaged to perform the attack, resulting in the likelihood rankings K_{ij} for the corresponding keys. The position r of the correct key k in the ranking is obtained.

d) Results Presentation: During the attack, the length of the trace S_l is traversed (where l represents a randomly sampled length from the test set, typically ranging from 0 to 300). r_l is calculated and the results are presented. The comparison and analysis of the experimental results will be discussed in detail in Section 4.4.1.

e) Hyperparameter Optimization: Grid search is used to optimize and select the hyperparameters of the model in order to achieve better attack performance. The comparison and analysis of the hyperparameters will be discussed in detail in Section 4.4.2.

4. Experimental Results and Analysis

4.1. Experimental Dataset

4.1.1. ASCAD Dataset [15]

ASCAD is the first publicly available dataset designated for evaluating deep learning techniques in side-channel attacks. The dataset consists of multiple databases, and the one commonly discussed in the literature is the dataset collected on an 8-bit ATmega8515 smart card platform with a working frequency of 4MHz. This dataset involves the collection of electromagnetic signals for a key-fixed first-order Boolean Masking AES-128 algorithm, with the leakage model focused on the first round S-box operation

$$Y^{(i)}(k^*) = Sbox[P_3^{(i)} \oplus k^*] \tag{8}$$

This database contains 60,000 traces collected during the acquisition process, with 50,000 traces used for training and 10,000 traces used for evaluating the performance of the trained model. Additionally, desynchronized datasets with maximum offsets of 50 or 100 can be generated using the provided code for training and validating the model's generalization performance.

4.1.2. FESH Dataset

Based on the leakage point analysis of the FESH algorithm, a dataset of 200,000 samples suitable for deep learning was collected. This dataset is based on the FESH-128-128 encryption algorithm executed on an STM32F103 microcontroller operating at a working frequency of 72MHz. Instructions were sent from a computer device to the microcontroller, along with random plaintexts, and an electromagnetic probe was used to collect the leakage signals of the encryption process on the microcontroller to an oscilloscope. Finally, the trace information was forwarded to a computer for storage. From the collected traces, 800 sample points after the key generation were located through alignment filtering and power analysis. Additionally, a desynchronized dataset with a maximum offset of 50 or 100 was generated to train and validate the model's generalization performance.

4.2. Experimental Metrics

4.2.1. Top-n Key Success Rate

For CPA attacks, the results obtained are the rankings of the correlation coefficients for the guessed keys. Given a set of guessed keys, the correlation coefficients are ranked in descending order, denoted as K', where $K' = \{k_0, k_1, \cdots, k_{255}\}$ represents the guessed key (8 bits). If the true key is k^*, then there will definitely exist a value k_i such that $k^* = k_i$ the correct prediction is in $i-th$ the position. If, $i < n$ it is considered a correct prediction, while if $i >= n$,it is considered an incorrect prediction.

$$Top-n = \frac{C}{T} \tag{9}$$

Top-n represents the success rate, where C represents the number of correctly predicted samples, and T represents the total number of samples. For the FESH-128-128 encryption algorithm, the total number of samples is equal to the length of the guessed key (128 bits) divided by 8 bits, which equals 16.

4.2.2. Guessing Entropy

For neural networks, guessing entropy is commonly used as a performance metric. Guessing entropy first appeared in side-channel analysis, where Standaert et al. theoretically proposed its impact based on time complexity, memory complexity, encryption algorithms, and attack algorithms [13]. In neural network side-channel analysis, only the learned prediction results are used as the calculation metric. In multiple literature achieving state-of-the-art results, the approach is to randomly select curves for each trace length in the range of [0, 300] and perform 100 attack iterations to obtain the guessing results.

4.3. CPA Attack Results of the FESH Algorithm

By taking the intermediate sequences of S-box input

(sboxin), S-box output (sboxout), and word mix (mixword), and calculating them with the 0th bit of the power traces for every 5k traces, the results based on the correlation factor as the leakage indicator can be obtained.

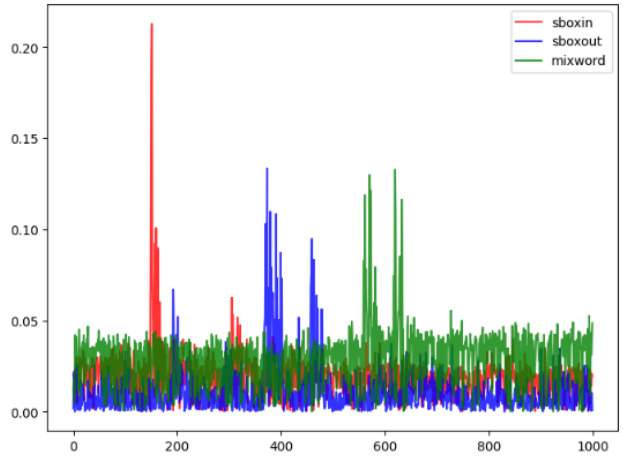

Figure 3. Analysis of FESH algorithm leakage point

According to the analysis of the correlation factors, it can be observed that there is leakage to varying degrees in all three selected leakage points, with a higher linear correlation observed in the S-box input.

For the CPA attack on the FESH algorithm, experiments were conducted with different numbers of power traces to predict the 16-bit 8-bit ciphertext and obtain the results.

Table 2. CPA attack results against FESH algorithm

stage	2k		5k		1w	
	Top-1	Top-4	Top-1	Top-4	Top-1	Top-4
sboxin	1.0	1.0	1.0	1.0	1.0	1.0
sboxout	0.25	0.5	0.188	0.75	0.188	0.75
mixword	0.0	0.06	0.0	0.0	0.0	0.06

By obtaining the success rates for the first four ranks, it can be concluded that if a random plaintext is used for side-channel attacks on the FESH algorithm, the key value can be obtained through a CPA attack on the leakage point of the S-box input. The S-box output can reveal a significant portion of the key in the first four ranks, reducing the difficulty of algebraic analysis in the results after conducting a CPA attack. However, no useful information can be obtained from the CPA attack on the word mix output.

4.4. Results of Template Attack Based on Improved TransNet Model

4.4.1. Experimental Comparison

In this study, a template attack was performed using the improved TransNet model on the FESH dataset, and the results were compared with the CNN model [14] and the original TransNet model. After establishing the attack models, a grid search method was used to optimize the hyperparameters, and the results were obtained by averaging multiple attacks. The improved TransNet model reduced the training parameters from 656,640 to 299,072, reducing the computational complexity of the model and making it more efficient. This reduction also mitigated the risk of overfitting and allowed the model to learn more discriminative feature representations. It can be clearly observed from the graph that the improved TransNet model outperforms the original TransNet model on the FESH dataset, and achieves consistent results with the CNN model on the synchronized dataset.

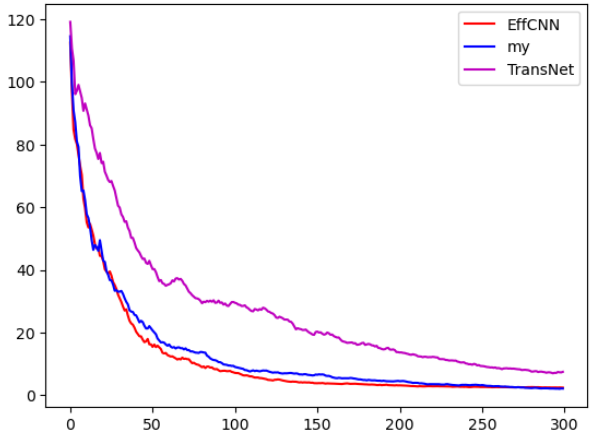

Figure 4. The Attack Results of FESH Algorithm on Synchronous Datasets

In order to validate the generalization performance of the model on the desynchronized dataset, training was conducted using the ASCAD dataset with maximum offsets of 0, 50, and 100. It was found that the improved TransNet model consistently outperformed the original TransNet model on all these datasets.

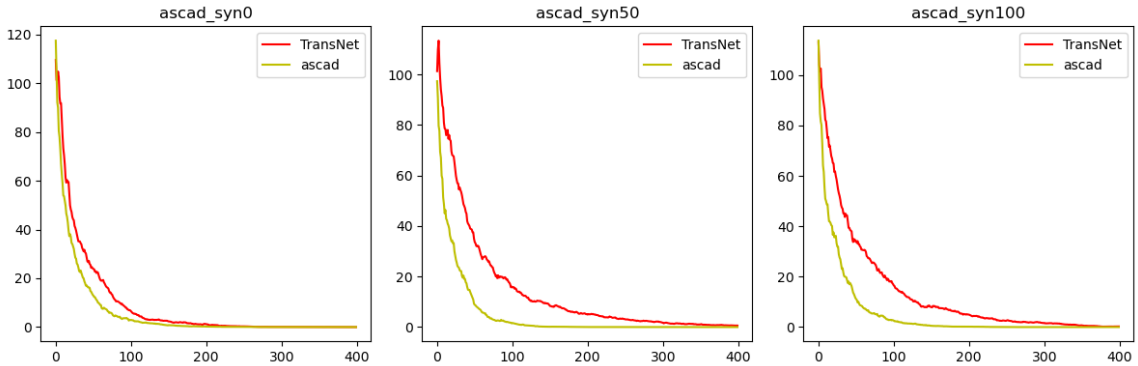

Figure 5. Training Results of FESH Dataset with Different Offsets

4.4.2. Hyperparameter Optimization

Hyperparameter optimization is an important step in training machine learning models. It aims to optimize the performance and generalization capabilities of the model by adjusting its hyperparameters. The goal of hyperparameter optimization is to find the best combination of hyperparameters that allows the model to perform well on the

training set and generalize well on unseen data. This process typically involves trying out different combinations of hyperparameters to evaluate the model's performance and selecting the best-performing combination as the final choice. In this project, grid search was used to explore the impact of kernel size, pool size, and dmodel size on the training process.

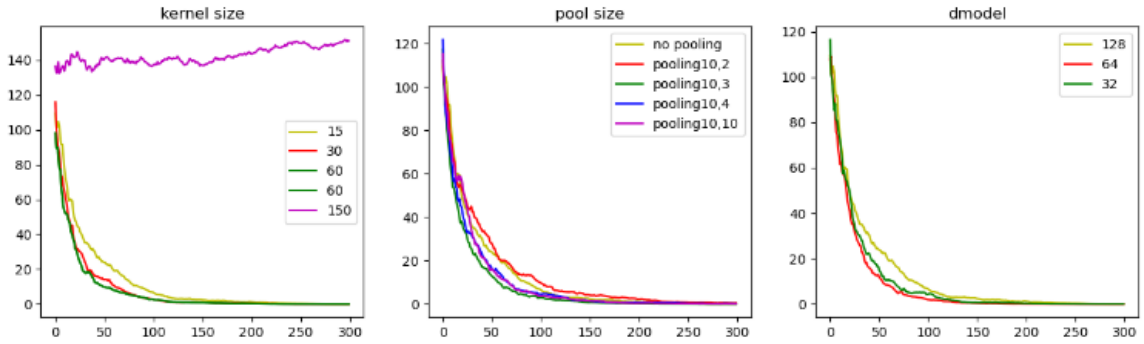

Figure 6. The influence of hyperparameters on guessing entropy

Finally, the hyperparameter values used in this study were as follows:

Table 3. Hyperparameter Values

Name	Vlaue	Name	Value
N1 layer	2	untie r	True
N2 layer	1	smooth pos emb	False
d model	64	untie pos emb	True
n head	2	init	Normal
d head	32	init std	0.02
d inner	256	max learning rate	2.5e-4
dropout	0.05	(gradient) clip	0.25
dropatt	0.05	min lr ratio	0.004
conv kernel size	60	warmup steps	0
pool size	10	batch size	256
pool stride	3	train steps	30000
clamp len	700		

5. Conclusion

The main objective of this study was to perform side-channel attacks on the FESH algorithm by collecting electromagnetic power consumption and generating a dataset for research purposes. Firstly, in terms of non-learning side-channel attacks, the correlation power analysis attack method successfully obtained the key at the S-box input position of the FESH algorithm and obtained effective leakage values at the S-box output position. Secondly, in terms of learning side-channel attacks, template attack methods based on the improved TransNet model were experimented on both the FESH dataset and the ASCAD dataset. The results showed that this model outperformed the traditional TransNet model on both synchronized and desynchronized datasets. This indicates that incorporating BurlPool and data normalization into the TransNet network can yield better results.

In conclusion, although the FESH algorithm performs well against algebraic analysis, it does not achieve satisfactory results in resisting side-channel analysis attacks to some extent. Additionally, the improved TransNet model still does not significantly outperform the CNN model on the desynchronized dataset. While self-attention mechanisms outperform convolution operations on more datasets, the improved TransNet model has higher training costs. Further breakthroughs are expected for self-attention mechanisms in the field of side-channel attacks.

Acknowledgments

I would like to express my heartfelt gratitude to my parents for their nurturing and support, as well as to my advisor for their guidance and encouragement, which enabled me to successfully complete this research. I am also grateful to my colleagues in the laboratory for their cooperation and support. Their valuable suggestions and discussions have played a crucial role in our research endeavors.

References

[1] Paul C. Kocher, “Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems,” Annu. Int. Cryptol. Conf., 1996, doi: 10.1007/3-540-68697-5_9.

[2] Paul C. Kocher, Joshua M. Jaffe, and Benjamin Jun, “Differential Power Analysis,” Annu. Int. Cryptol. Conf., 1999, doi: 10.1007/3-540-48405-1_25.

[3] E. Brier, C. Clavier, and F. Olivier, “Correlation Power Analysis with a Leakage Model,” in Cryptographic Hardware and Embedded Systems - CHES 2004, vol. 3156, M. Joye and J.-J. Quisquater, Eds., in Lecture Notes in Computer Science, vol. 3156. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 16–29. doi: 10.1007/978-3-540-28632-5_2.

[4] Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi, “Template Attacks,” 2002.

[5] Gabriel Hospodar, Benedikt Gierlichs, Elke De Mulder, Ingrid Verbauwhede, and Joos Vandewalle, “Machine learning in side-channel analysis: a first study,” J. Cryptogr. Eng., 2011, doi: 10.1007/s13389-011-0023-x.

[6] Lukas Malina, Vaclav Zeman, Josef Martinasek, and Zdenek Martinasek, “K-Nearest Neighbors Algorithm in Profiling Power Analysis Attacks,” Radioengineering, 2016, doi: 10. 13164/ re.2016.0365.

[7] Matthew D. Zeiler and Rob Fergus, “Visualizing and Understanding Convolutional Networks,” Eur. Conf. Comput. Vis., 2014, doi: 10.1007/978-3-319-10590-1_53.

[8] A. Vaswani et al., “Attention is All you Need,” in Advances in Neural Information Processing Systems, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017. [Online]. Available: https://proceedings. neurips.cc/ paper_files/ paper/ 2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[9] Suvadeep Hajra, Sayandeep Saha, Manaar Alam, and Debdeep Mukhopadhyay, “TransNet: Shift Invariant Transformer Network for Side Channel Analysis (extended version),” 2022.

[10] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. Le, and R. Salakhutdinov, “Transformer-XL: Attentive Language Models beyond a Fixed-Length Context,” in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy: Association for Computational Linguistics, 2019, pp. 2978–2988. doi: 10.18653/v1/P19-1285.

[11] Jia keting; Dong xiaoyang; Wei zongming; Li zheng; Zhou haibo; Cong tianshuo;, "Block Cipher Algorithm FESH," Journal of Cryptography, no. 06 vo 6, pp. 713–726, 2019, doi: 10. 13868/j. cnki. jcr.000336.

[12] R. Zhang, “Making Convolutional Networks Shift-Invariant Again,” in Proceedings of the 36th International Conference on Machine Learning, K. Chaudhuri and R. Salakhutdinov, Eds., in Proceedings of Machine Learning Research, vol. 97. PMLR, Jun. 2019, pp. 7324–7334. [Online]. Available: https:// proceedings. mlr.press/v97/zhang19a.html.

[13] F.-X. Standaert, T. G. Malkin, and M. Yung, “A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks,” in Advances in Cryptology - EUROCRYPT 2009, vol. 5479, A. Joux, Ed., in Lecture Notes in Computer Science, vol. 5479. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 443–461. doi: 10.1007/978-3-642-01001-9_26.

[14] G. Zaid, L. Bossuet, A. Habrard, and A. Venelli, “Methodology for Efficient CNN Architectures in Profiling Attacks,” IACR Trans. Cryptogr. Hardw. Embed. Syst., pp. 1–36, Nov. 2019, doi: 10.46586/tches.v2020.i1.1-36.

[15] R. Benadjila, E. Prouff, R. Strullu, E. Cagli, and C. Dumas, “Deep learning for side-channel analysis and introduction to ASCAD database,” J. Cryptogr. Eng., vol. 10, no. 2, pp. 163–188, Jun. 2020, doi: 10.1007/s13389-019-00220-8.