Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 3, No. 1, 2023

Template attack based on uBlock cipher algorithm

Jiyuan Xin*, Zhibo Du

School of Cybersecurity, Chengdu University of Information Technology, Chengdu, 610200, China
* **Corresponding author**: Jiyuan Xin (Email: 872739360@qq.com)

Abstract: The uBlock cipher algorithm is a family of lightweight block cipher algorithms, which was proposed by Wu Wenling, Zhang Lei and others in 2019. Its block length and key length support 128 and 256 bits. It has certain flexibility in different environments to meet the application of algorithms in different software and hardware conditions. At present, the research of side channel attack against uBlock cipher algorithm is mainly differential fault analysis. In order to explore the application of template attack against uBlock cipher algorithm analysis attack, a template attack method against uBlock cipher algorithm is proposed. When constructing the template, the S-box output of uBlock cipher algorithm is selected as the attack point of energy analysis, and the Hamming weight is used as the analysis model to build the template about S-box output.

Keywords: Side channel attack; Template attack; uBlock algorithm.

1. Introduction

Cryptographic technology is the foundation and support of cyberspace security. From the physical equipment at the bottom to the system application at the top, cryptographic technology plays a security role in cyberspace security, such as confidentiality, integrity, authenticity and non-repudiation. The implementation of cryptographic algorithms in cryptography depends on a physical device platform, namely the cryptographic chip. The process of attackers recovering sensitive data or key information by using the side-channel information released by the cryptographic chip during its actual operation is called side-channel attack. The side channel analysis of cryptographic algorithm is an attack method to recover sensitive information such as key from the energy consumption, electromagnetic and other side information during the execution of cryptographic algorithm by the device. It is based on the fact that the instantaneous energy consumption, electromagnetic and other side information of cryptographic device depends on the data processed and the operation performed by the device.

The uBlock cipher algorithm is a lightweight block cipher algorithm proposed by Wu Wenling and others in the National Cryptographic Algorithm Design Competition. The block length and key length support 128 and 256 bits. The overall structure, S-box, diffusion matrix, key expansion and other designs of the uBlock algorithm reflect the balance of security, implementation efficiency and adaptability everywhere Encounter attack and other group cipher analysis methods have sufficient security redundancy. The uBlock algorithm adapts to various software and hardware platforms, takes full account of the computing resources of modern microprocessors, and can be efficiently implemented using SSE, AVX2 and other instruction sets. The hardware implementation is simple and effective. It can be implemented at high speed to ensure the safe application of high-performance environments, and can also be implemented in lightweight to meet the security requirements of resource-constrained environments.

According to the structural characteristics of uBlock cipher algorithm and the principle of template attack, this paper analyzes the feasibility of template attack against uBlock cipher algorithm, proposes and designs a template attack against uBlock cipher algorithm based on the analysis of the information leakage point of uBlock cipher algorithm, and explores the feasibility of template attack method applied to uBlock cipher algorithm. Through the actual attack experiment against the smart card of uBlock cipher algorithm, The effectiveness of the attack method is verified.

2. Introduction to uBlock algorithm

2.1. uBlock encryption algorithm

The uBlock block encryption algorithm is a symmetric encryption algorithm. The block length and key length support 128 and 256 bits. Therefore, there are three versions of the uBlock encryption algorithm, which are recorded as uBlock-128/128, uBlock-128/256, and uBlock-256/256. Their iteration rounds are 16, 24 and 24, respectively. The encryption round transformation diagram is shown in Figure 1.

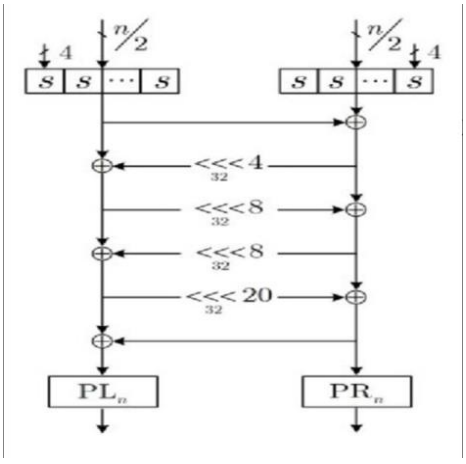

Figure 1. The U-Block encryption algorithm wheel transformation diagram

Enter n-bit plaintext X and round key RK0, RK1, RKr, output n-bit ciphertext Y. Encryption algorithm E is as follows:

$$\text{uBlock.Enc}\left(X, \text{RK}^0, \text{RK}^1, \cdots, \text{RK}^r\right)$$

$$X_0 \| X_1 \leftarrow X$$

$$\text{for } i = 0 \text{ to } r-1 \text{ do}$$

$$\text{RK}_0^i \| \text{RK}_1^i \leftarrow \text{RK}^i$$

$$X_0 \leftarrow S_n\left(X_0 \oplus \text{RK}_0^i\right)$$

$$X_1 \leftarrow S_n\left(X_1 \oplus \text{RK}_1^i\right)$$

$$X_1 \leftarrow X_1 \oplus X_0$$

$$X_0 \leftarrow X_0 \oplus \left(X_1 \underset{32}{\lll} 4\right)$$

$$X_1 \leftarrow X_1 \oplus \left(X_0 \underset{32}{\lll} 8\right)$$

$$X_0 \leftarrow X_0 \oplus \left(X_1 \underset{32}{\lll} 8\right)$$

$$X_1 \leftarrow X_1 \oplus \left(X_0 \underset{32}{\lll} 20\right)$$

$$X_0 \leftarrow X_0 \oplus X_1$$

$$X_0 \leftarrow \text{PL}_n\left(X_0\right)$$

$$X_1 \leftarrow \text{PR}_n\left(X_1\right)$$

$$\leftarrow \text{RK}^r \oplus \left(X_0 \| X_1\right)$$

It is the juxtaposition of n/8 4-bit S-boxes. The S-boxes and permutations used in the encryption process are shown in Table 1-3 below:

3. Template attack

Template attacks exploit the fact that energy consumption depends on the data being processed by the device. The template attack uses multivariate normal distribution to characterize the energy trace, thus capturing the key information leaked by the energy trace. In the process of describing the template, in order to obtain the side channel information output by the attacked device, the attacker needs a device of the same type to output, so the template attack is a selective plaintext attack. The template attack is divided into two stages: template characterization and key recovery. In the template characterization stage, the attacker calculates the characteristics of the energy traces corresponding to different keys according to the obtained side channel information, such as the mean vector and covariance matrix; In the key recovery phase, the attacker matches the energy trace with the template to obtain the corresponding matching probability. Generally, the key corresponding to the template with the maximum matching probability is the correct key.

The classic template attack uses multivariate Gaussian probability distribution to model the energy consumption noise of operation. The training phase includes selecting points of interest, vectorization of energy traces, and calculating template parameters.

(1) Select points of interest: for K operations, collect K training energy trace sets, each of which contains energy traces. Calculate the mean energy trace of each energy trace set. Calculate the difference of mean energy trace:

$$\delta = \sum_{i<j} M_i - M_j$$

And select the point with the largest difference value as the point of interest.

(2) And select the point with the largest difference value as the point of interest. Energy trace vectorization: for any energy trace t in the energy trace set Si, its noise vector X is:

$$x = \left(t[P_1] - M_i[P_n], \cdots, t[P_n] - M_i[P_n]\right)$$

(3) Calculate the template parameters: the multivariate Gaussian density distribution of the template Ti of the energy trace set Si is:

$$p(x \mid T_i) = \frac{1}{\sqrt{(2\pi)^n \mid \Sigma_i \mid}} \exp\left(-\frac{1}{2} x^T \Sigma_i^{-1} x\right)$$

Including:

$$\Sigma_i[j,k] = \frac{1}{n-1} x_j^T x_k, \quad x_j, x_k \in S_i$$

In the attack phase, the classic template attack uses an attack trace t to determine the template to which t belongs by the maximum likelihood method.

(1) Calculate the likelihood ratio of t to template i: first, convert t to the vector in template i:

$$y = \left(t[P_1] - M_i[P_n], \cdots, t[P_n] - M_i[P_n]\right)$$

Likelihood rate is:

$$p(y \mid T_i) = \frac{1}{\sqrt{(2\pi)^n \mid \Sigma_i \mid}} \exp\left(-\frac{1}{2} y^T \Sigma_i^{-1} y\right)$$

(2) Take the template of maximum likelihood rate as the matching template of energy trace t.

$$\hat{T} = \underset{T_i}{argmax}\, p\left(y_i \mid T_i\right)$$

(3) The key is calculated according to the corresponding operation of the matched template.

As mentioned earlier, operation O actually represents the information it discloses. According to the relationship between O's information (such as the Hamming weight output by SBOX) and the key, one or more sub-keys can be calculated from O. If there are multiple sub-keys, brute force attack is required.

4. Template attack against uBlock cipher algorithm

There are many information leakage points in uBlock cipher algorithm. The commonly used information leakage points are S-box input, S-box output, P-box input, P-box output, etc. For the model attack research of uBlock cipher algorithm, the selected information leakage point is S-box output, and the energy model is Hamming weight model. The template attack method for uBlock cipher algorithm is as follows:

(1) Collect energy curve. Under the same acquisition conditions, the energy curve t corresponding to the encryption operation of known key K and known plaintext X is collected, and the number of curves is n.

(2) Data preprocessing. In order to improve the signal-to-noise ratio of the energy curve, filtering and static alignment are used to preprocess the collected energy curve.

(3) Leakage analysis. There are two forms of attack model leakage range: one is the leakage area, which means that there is leakage in the middle value of the range from a certain sample location to a certain sample location. This method is

mostly used in the training of neural network models. The other is the point of interest (that is, the disclosure point), which is mostly used for TA attacks (traditional template attacks). The purpose of leakage analysis is to find the leakage area or sample points (POIs) of the target value in the energy consumption curve.

(4) Train the attack model. The goal of the training model is to use the training data set to learn the probabilities of various values of the predicted target median based on energy consumption.

(5) Check the attack model. The checking model is a preparation before attacking the master key, and its purpose is to evaluate the number of attack traces required when attacking the master key.

(6) Attack master key. A batch of candidate master keys will be obtained during the attack. These master keys are used by the encryption algorithm to encrypt known plaintext, and the results are compared with known ciphertext. In case of equality, the master key has been attacked.

5. Experiment of template attack against uBlock cipher algorithm

For the template attack experiment of the uBlock cryptographic algorithm, the key selected for the template construction is 6CECC67F287D083DEB8766F0738B36CF, the clear text is 0123456789ABCDEFFEDCBA9876543210, the experimental condition is the Inspector side channel attack platform, the test object is the uBlock cryptographic algorithm smart card, the attacked key is 128 bits, A total of 50000 energy trace curves were collected in this experiment, of which 45000 were used as training sets and 5000 were used as verification sets.and the collected energy curve is shown in Figure 2.

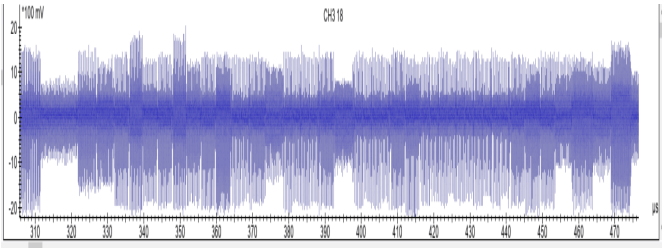

Figure 2. UBlock cryptographic algorithm smart card energy curve

5.1. Data preprocessing

Use the filter tool provided by the Inspector software to perform low-pass filtering on the energy curve. The filter parameter is 30. Perform static alignment on the filtered data. The aligned curve is shown in Figure 3.

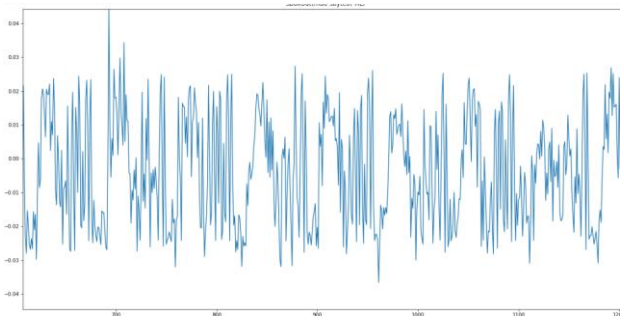

Figure 3. Curve after filtering and static alignment

5.2. Leakage analysis

The information disclosure point selected for the template attack research of uBlock cryptographic algorithm is S-box output, and the energy model is Hamming weight model. The collected energy curve is analyzed for leakage. Through the leakage analysis results, obvious leakage is found as shown in Figure 4.

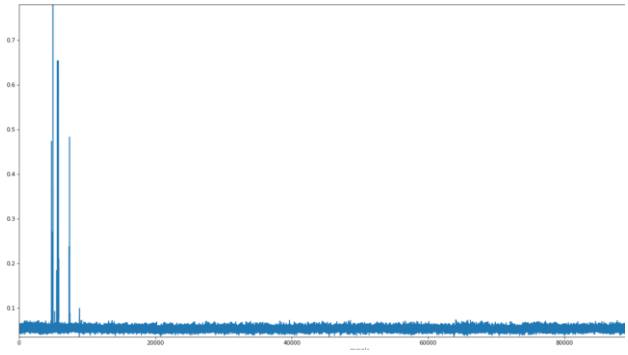

Figure 4. Energy curve after leakage analysis

5.3. Training attack model

The goal of the training model is to use the training data set to learn the probabilities of various values of the predicted target median based on energy consumption. Both TA, SVM, RF or neural network models are used to predict the probability distribution of the median based on energy consumption.

It should be noted that the training data set needs to be loaded from the training set file when training the model. At this time, the energy consumption only loads the samples of the leakage range class obtained in the previous leakage analysis, or the samples on the POIs.

5.4. Check attack model

After the model training, the attack model needs to be checked. A parameter called guess entropy will be obtained during the model training. Guess entropy refers to the average rank of the sub-key in the candidate sub-key queue when attacking a sub-key. When attacking a subkey of a sbyte, we need to use the model to calculate the score of various guessed subkeys, that is, according to the energy consumption vector, use the trained model to "predict" the probability of guessing the middle value (usually its logarithm), and use it as the score of guessed subkeys. According to the score of each guess subkey, they are sorted in descending order. The highest score is ranked at 0, the second is ranked at 1, and so on. In the training process, in order to verify the quality of the model, an independent verification set data is used to evaluate the model multiple times. The average value of the correct sub-key ranking in the multiple evaluations is the guess entropy, which is used as the score of the model. It can be seen that the smaller the guess entropy (model score), the better the quality of the model. Guess the minimum entropy is 0. The result of checking the attack model is shown in Figure 5.In addition to guessing entropy, the check results also include first-order to fourth-order success rates. The first-order success rate represents the percentage of correct sub-key ranking first. The second order success rate represents the attack proportion of the correct sub-key ranking in the top two. Third, fourth and so on.

	1	2	3	4	5	6	7	8	9	10
一阶成功率	0.5	0.92	0.98	0.96	0.94	1.0	1.0	1.0	1.0	1.0
二阶成功率	0.82	0.98	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
三阶成功率	0.94	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
四阶成功率	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
猜测值	0.74	0.1	0.02	0.04	0.06	0.0	0.0	0.0	0.0	0.0

Figure 5. Inspection result chart

5.5. Attack master key

After training all the attack models, you can attack the master key. During the attack, a batch of candidate master keys will be obtained. These master keys are used by the encryption algorithm to encrypt the known plaintext, and the results are compared with the known ciphertext. In case of equality, the master key has been attacked. The master key from this experiment attack is the same as the master key set at the beginning, and the attack is successful. The result is shown in Figure 6.

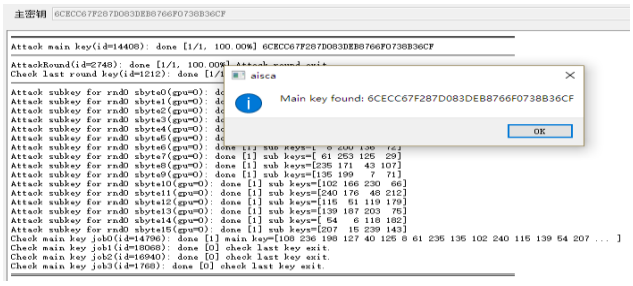

Figure 6. Result of attack on master key

6. Conclusion

Research on template attack of uBlock cryptographic algorithm, propose a template attack method against uBlock cryptographic algorithm, and explore and practice the application of this attack method against uBlock cryptographic algorithm smart card. This attack method not only has practical application significance for the security research of uBlock cryptographic algorithm products, but also has important reference significance for the template attack security research of other cryptographic algorithms.

References

[1] P. Kocher, J. Jaffe, and B. Jun, “Differential power analysis,” in Advances in Cryptology—CRYPTO’99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19, 1999, pp. 388–397.

[2] Wu Wenling, Zhang Lei, Zheng Yafei, Li Lingchen, and others, "Block cipher uBlock," Journal of Cryptography,vol. 6, no. 6, pp. 690–703, 2019.

[3] Guo Dongxin, Chen Kaiyan, Zhang Yang, Xie Fangfang, and Zhang Xiaoyu, "Overview of research on template attacks against cryptographic chips", no. 12, pp. 79–83, 2018.

[4] Li Peizhi, Yan Yingjian, and Duan Erpeng, "Research on DES cryptographic chip template attack technology," PhD Thesis, 2013.

[5] Kuang Xiaoyun, Huang Kaitian, Lan Tian, Du Zhibo, and Wu Zhen, "Template attack against SM4 cryptographic algorithm," Journal of Chengdu University of Information Engineering, 2021

[6] S. Chari, J. R. Rao, and P. Rohatgi, “Template attacks,” in Cryptographic Hardware and Embedded Systems-CHES 2002: 4th International Workshop Redwood Shores, CA, USA, August 13–15, 2002 Revised Papers 4, 2003, pp. 13–28.

[7] C. Rechberger and E. Oswald, “Practical template attacks,” in Information Security Applications: 5th International Workshop, WISA 2004, Jeju Island, Korea, August 23-25, 2004, Revised Selected Papers 5, 2005, pp. 440–456.

[8] O. Choudary and M. G. Kuhn, “Efficient template attacks,” in Smart Card Research and Advanced Applications: 12th International Conference, CARDIS 2013, Berlin, Germany, November 27-29, 2013. Revised Selected Papers 12, 2014, pp. 253–270.

[9] O. Choudary and M. G. Kuhn, “Template attacks on different devices,” in Constructive Side-Channel Analysis and Secure Design: 5th International Workshop, COSADE 2014, Paris, France, April 13-15, 2014. Revised Selected Papers 5, 2014, pp. 179–198.

[10] Du Zhibo, Sun Yuanhua, and Wang Yi, "Multi-point joint energy analysis attack against AES cryptographic algorithm," Journal on Communications, 2016.