SCIENCE CHINA
Information Sciences

• LETTER •

March 2018, Vol. 61 039103:1–039103:3
doi: 10.1007/s11432-016-9171-6

Protecting white-box cryptographic implementations with obfuscated round boundaries

Tao XU[1,2,3], Chuankun WU[4*], Feng LIU[1,2,5*] & Ruoxin ZHAO[1,2]

[1]*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093*, China;*
[2]*School of Cyber Security, University of Chinese Academy of Sciences, Beijing* 100049*, China;*
[3]*New United Group Co., Ltd., Changzhou* 213166*, China;*
[4]*School of Mathematics, Shandong University, Jinan* 250100*, China;*
[5]*School of Big Data and Computer Science, Guizhou Normal University, Guiyang* 550001*, China*

Received 28 September 2016/Revised 13 April 2017/Accepted 21 June 2017/Published online 25 August 2017

Citation Xu T, Wu C K, Liu F, et al. Protecting white-box cryptographic implementations with obfuscated round boundaries. Sci China Inf Sci, 2018, 61(3): 039103, doi: 10.1007/s11432-016-9171-6

Dear editor,

White-box cryptography aims to give a secure software implementation of the cryptographic algorithm running in an untrusted environment which is owned and controlled by an adversary. Its major goal is to protect the confidentiality of the secret key. Given the increasing demands for software-only applications, white-box cryptography has received a lot of attention from industry. By now, all publications of white-box cryptographic (WBC) implementations from public literatures can be divided into two groups: (i) implementations of existing block ciphers such as AES and DES; (ii) dedicated designs of block ciphers which are assumed to run in the white-box environment.

Chow et al. [1] introduced a practical technique of hiding secret keys into network-encoded lookup tables and designed a white-box implementation to protect AES codes. Their white-box AES implementation was broken [2]. Some efforts have been made to improve Chow et al.'s design and all failed. Apart from white-box AES, there are some white-box implementations of other ciphers, e.g., DES, SM4 and SHARK. Similar to AES, the public and constant confusion and diffusion operations of the original ciphers lead to the insecurity of their white-box implementations. Michiels et al. [3] proposed an algorithm to attack white-box implementations of SLT (substitution-linear transformation) ciphers and claimed that SLT ciphers are less suited for white-box implementations unless new white-box techniques are designed.

We revisit Chow et al.'s technique. By observing the known attacks, we notice that in all published white-box schemes of standard ciphers, the boundaries of each round are obvious. Thus no matter how complex the lookup tables in one round are, the attackers can merely use the input and output data of this round to extract some information of the secret key. To add difficulty to the attacks, the round boundaries should be obfuscated in some way. We focus on the SLT ciphers and extend Chow et al.'s method to derive a generic class of WBC implementations which are based on lookup table groups (LTGs).

White-box implementation of an SLT cipher. There is a general way to transform an SLT cipher into a white-box implementation by using Chow et al.'s technique. Each round of the SLT cipher is an n-bit to n-bit mapping and can be implemented as a network of lookup tables, which includes n/m m-bit to n-bit lookup tables and a layer of XOR tables, as shown in Figure 1(a). One XOR table takes two m-bit values as inputs and maps

* Corresponding author (email: 13520965063@163.com, liufeng@iie.ac.cn)
The authors declare that they have no conflict of interest.

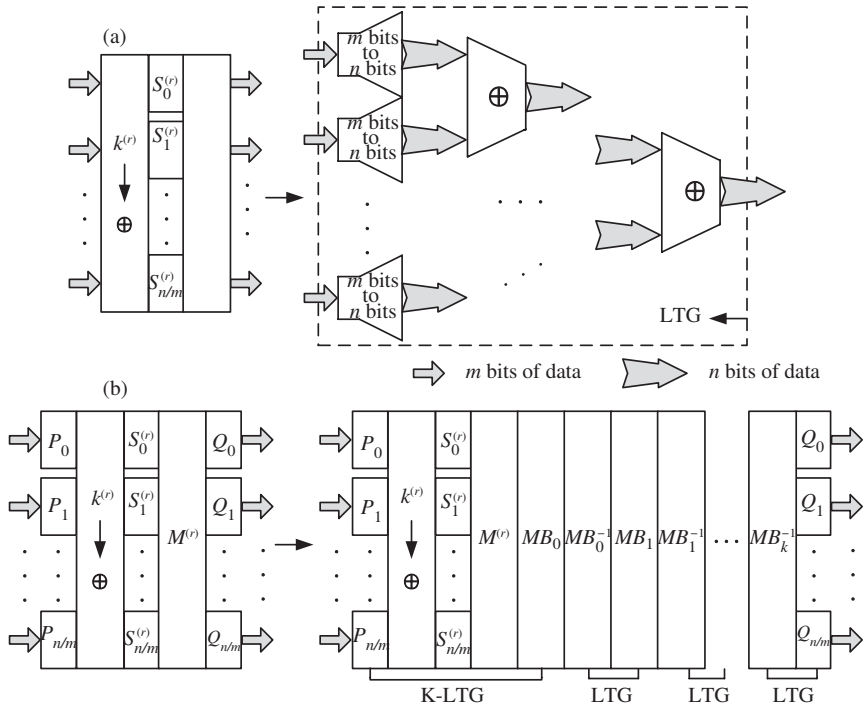

Figure 1 LTG and one obfuscated round of an SLT cipher. (a) Lookup tables of one LTG; (b) insert a random amount of mixing bijections into one round.

them into their m-bit XOR result. The network of lookup tables in Figure 1(a) is called a lookup table group (LTG).

Network-encodings are used to protect the secret key embedded in the tables. P_i and Q_j are input and output encodings which are randomly selected from $m \times m$ invertible matrices over $GF(2)$. The mixing bijection MB is inserted after the linear layer $M^{(r)}$ in a round. MB is randomly chosen from $n \times n$ invertible matrices over $GF(2)$. Similar to 4-bit to 4-bit protective bijections in Chow et al.'s scheme, sets of random nonlinear m-bit to m-bit bijections are implemented on the input and output of each table to complete the network-encodings finally. If only one MB is inserted into a round, there will be two LTGs in this round. The boundaries of each round are obvious. The correspondence between the input and output data is easy to determine. Then the known attacks are still valid.

Obfuscation of round boundaries. To obfuscate the round boundaries, we increase the number of the mixing bijection MB in each round with a random amount $k \geqslant 1$. As shown in Figure 1(b), each $n \times n$ matrix MB_i is followed by its inverse. One more bijection means one more LTG in this round. Actually ${MB_i}^{-1}$ could be randomly placed as long as it can cancel MB_i later, e.g. ${MB_0}^{-1} \circ {MB_1}^{-1} \circ MB_1 \circ MB_0$. Among all the LTGs, the ones that include the round keys are named K-LTGs. As a result, the final structure of the white-box implementation is a chain of LTGs. Each LTG has the same structure. Facing such a WBC implementation, it is difficult for an attacker to determine the boundaries of each round.

Security analysis. Before attacking a WBC implementation with obfuscated round boundaries, the attacker has to determine the boundaries of each round first. If the positions of two adjacent K-LTGs are located, the boundaries of one round will be determined. Suppose that there are R K-LTGs and x other LTGs. Each K-LTG is protected by at least one LTG. C is the number of tries to locate all K-LTGs. The work factor of locating K-LTGs with an exhaustive search method is $C = \mathcal{C}_{x-1}^{R}$, which could be used to evaluate the difficulty of the added attack. In some ciphers, the secret key can be compromised by attacking part of the K-LTGs, e.g., r K-LTGs, with a work factor being $C = \mathcal{C}_{x-(R-r)-1}^{r}$.

The above tells that locating K-LTGs is just a problem of permutations and combinations. In the case of $r = R$, with the increasing of x, the upper bound of $C = \mathcal{C}_{x-1}^{R}$ can be measured by $\mathcal{O}(x^R)$. The security level is improved polynomially, but not exponentially. In the case of $r < R$, it is even weaker. To obtain a practical security level, the number of LTGs should be increased largely. The code size of the implementation will be very big.

Dedicated WBC implementations. Biryukov et al. [4] proposed several schemes based on the **ASASA** structure with affine (A) and nonlinear

(S) layers interleaved. Bogdanov et al. [5] proposed another family of dedicated white-box block ciphers **SPACE** including several variants with different code sizes from 3.84 KB to 51.5 GB. Their basic approach is to build a scalable lookup table from a well-studied standard block cipher by constraining the plain text and truncating the cipher text. In a Feistel-type construction, this table is reused in each round as an obfuscator. Later, they improved **SPACE** to **SPNbox** [6].

In the above designs, the notion of space hardness is introduced which is used to evaluate the difficulty of code lifting by the size of the codes. Code lifting means that an adversary can lift the whole WBC implementation as an equivalent large key to copy the functionality. Large size of incompressible codes, which is beyond an attacker's processing capacity, can mitigate code lifting. Our scheme follows the notion of space hardness and could be regarded as a WBC mechanism with high level of space hardness to design new dedicated WBC implementation. For example, combing with the method in [7], some new WBC implementation will be designed.

Discussion. The prime security goal of a WBC implementation is unbreakability. However, in real applications, some other goals need to be considered: one-wayness, incompressibility and traceability. A more detailed description can be found in Appendix A. By now, there are not standard specifications for white-box cryptography. Each published WBC implementation satisfies a subset of these goals. Delerablée et al. [8] formalized the security goals. They separately considered various security goals and attack models, and obtained distinct security definitions by pairing a particular goal with a particular attack model. As an improvement to Chow et al.'s technique, Our method could satisfy these security goals under certain conditions.

Experiment. We applied our scheme to AES and evaluated the added difficulty when using known attack methods. Also, we tested the running efficiency. For detailed results of the experiment, please refer to Appendix B.

Conclusion. We proposed an LTG-based method to transform an SLT cipher into a white-box implementation with obfuscated round boundaries. Our method forces an attacker to analyze the implementation as a whole and increases the difficulty of applying known attacks. In our method, a constant number of K-LTGs are randomly distributed among a chain of LTGs. The total number of LTGs in this chain is random and can be selected according to the actual computing resource and the requirement of security level. In recent years, some dedicated designs have been proposed. The notion of space hardness is adopted to evaluate the ability against code lifting attack. Our method, which can be regarded as a WBC strategy with high level space hardness and a variable scale, could be used to improve these dedicated schemes or design new dedicated WBC implementations.

Acknowledgements This work was supported by National Key R&D Program of China (Grant No. 2016YFB0800100), CAS Strategic Priority Research Program (Grant No. XDA06010701), and National Natural Science Foundation of China (Grant No. 61671448).

Supporting information Appendixes A and B. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Chow S, Eisen P, Johnson H, et al. White-box cryptography and an AES implementation. In: Proceedings of International Workshop on Selected Areas in Cryptography. Berlin: Springer, 2003. 250–270
2 Billet O, Gilbert H, Ech-Chatbi C. Cryptanalysis of a white box AES implementation. In: Proceedings of International Workshop on Selected Areas in Cryptography. Berlin: Springer, 2005. 227–240
3 Michiels W, Gorissen P, Hollmann H D L. Cryptanalysis of a generic class of white-box implementations. In: Proceedings of International Workshop on Selected Areas in Cryptography. Berlin: Springer, 2009. 414–428
4 Biryukov A, Bouillaguet C, Khovratovich D. Cryptographic schemes based on the ASASA structure: black-box, white-box, and public-key. In: Proceedings of International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2014. 63–84
5 Bogdanov A, Isobe T. White-box cryptography revisited: space-hard ciphers. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2015. 1058–1069
6 Bogdanov A, Isobe T, Tischhauser E. Towards practical whitebox cryptography: optimizing efficiency and space hardness. In: Proceedings of International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2016. 126–158
7 Xu T, Liu F, Wu C K. A white-box AES-like implementation based on key-dependent substitution-linear transformations. Multimed Tools Appl, 2017, doi: 10.1007/s11042-017-4562-8
8 Delerablée C, Lepoint T, Paillier P, et al. White-box security notions for symmetric encryption schemes. In: Proceedings of International Conference on Selected Areas in Cryptography. Berlin: Springer, 2013. 247–264