RC5-CBC: Best Matching Combination of Block Cipher and Model of Operation for Implementing ORAM Schemes

Changqi Sun, Rui Li

College of Computer Science and Network Security, Dongguan University of Technology Dongguan 523808, Guangdong, China
changqischolar@qq.com, ruili@dgut.edu.cn

Abstract. The encryption and decryption time of semantically secure encryption techniques have a great influence on the total end-to-end delay of an ORAM scheme. However, to the best of our knowledge, there is no work to study the impacts of combinations of block cipher and common model of operation on implementing semantically secure encryption techniques by comparing and analysing their execution time. This paper examines the running time of commonly used combinations of block cipher and model of operation to find the most matchable one for an ORAM scheme. We first selected the candidates of symmetric block encryption algorithms based on the existing block ciphers included in the LibTomCrypt, a popular cryptography toolkit. We second designed a comprehensive experiment in terms of prominent application scenarios of ORAM. We finally analysed the experimental results and found that the encryption algorithm of RC5-CBC outperforms others. In the end, we conclude that the RC5-CBC is the most matchable combination of block cipher and model of operation for implementing an ORAM scheme.

Keywords- ORAM, Block Cipher, Model of Operation, Encryption and Decryption Time

1. Introduction

Leveraging cloud platforms to store and manage their sensitive big data and IT service infrastructures is increasingly becoming the best choice for enterprises and companies[1]. Because the cloud service could largely deduce their operating cost and increase their operating efficiency. Nevertheless, the interaction between a client and a server inevitably leakages data's sensitive information to adversaries e.g., servers. Specifically, the data's positions accessed by a client (access pattern) severely threaten client's privacy especially in cloud scenario[2]. The classic privacy-preserving tool encryption algorithm cannot hide data's access pattern. Therefore, Goldreich and Ostrovsky proposed the Oblivious Random Access Machine (ORAM) to hide the access pattern[3]. But ORAM has been considered a theoretical concept not a practical one, due to its overwhelming end-to-end accessing delay[4]. In other words, decreasing the access delay plays a vital role in making ORAMs practical.

Most of ORAM schemes[5]–[10] need semantically secure encryption techniques to cater for hiding access pattern, so that two of ciphertexts of the same plaintext is different and no related. The combination of a block cipher and one of block cipher models of operation (e.g., Counter Model) is used to implementing semantically secure encryption technique. Therefore, the problem is to find the most matchable combination for implementing ORAM scheme.

The prior work related with ORAM directly advised to use combination AES-CTR to implement semantically secure encryption technique with no reason, e.g., Hoang et al. used the AES-CTR to implement the encryption and decryption of block data in Path-ORAM[2]. The works about pure comparisons of combinations of block cipher and model of operation did not consider the requirements of ORAM for symmetric encryption techniques. For example, Almuhammadi et al. compared the block cipher model of operation Electronic Code Book (ECB) that not satisfies the requirement of semantic security[11].

In this paper, we compare the running time of adequately secure combinations of block cipher and model of operation to find the most matchable one for an ORAM scheme. We first selected the candidates of symmetric block encryption algorithms based on the existing block ciphers included in the LibTomCrypt, a popular cryptography toolkit. In addition, we picked out the block cipher models of operation according to the information learned from prior works and requirements of semantic security. We second designed a comprehensive experiment in terms of prominent application scenarios of ORAM, e.g., cloud computing. We finally analyzed the experimental results and found that the combination of RC5-CBC outperforms others.

Our contributions are summarized as follows. (1) we, in this paper, find the best matchable combination of block cipher and model of operation for implementing ORAM schemes. (2) we conduct a comprehensive experiment to compare the performance of common combinations of block cipher and model of operation.

2. Related Work

Prior works in need of semantically secure encryption techniques. The tree-based ORAM schemes [3]–[5], [10], [12] point that they need the semantically secure encryption techniques to ensure the twice ciphertexts for same plaintext are not related. But they do not refer to the implementation ways. The ORAM schemes[2], [13] that do the comparison experiments directly use AES-CTR as the implementation way of semantically secure encryption techniques without any explanations.

Prior works about efficiency of combinations of block cipher and model of operation. Almuhammadi et al. only compare the efficiency of combinations of AES and models of operation (ECB, CBC, CFB, OFB, and CTR)[11]. Raigoza et al. compare and analyze the efficiency of DES-ECB, 3DES-ECB, AES-ECB, RC2-ECB, and Blowfish-ECB[14]. Nidhi et al. compare the efficiency of combinations of block ciphers (AES, DES, Blowfish) and models of operations(ECB, CBC, OFB, and CFB)[15]. Li et al. only consider the efficiency of combinations of AES and models of operation(ECB and CBC)[16].

3. Preliminary

Oblivious Random Access Machine (ORAM) is a cryptography primitive that can hide data's access pattern from the server[3]. A typical ORAM scheme consists of two components initialization and access. The initialization part is to prepare the processed data and corresponding metadata. The access part needs to make the client's access oblivious from the server. Therefore, it will first fetch some data encrypted data blocks from the server and then rewrite the blocks back to the server. To protect the fetched blocks' information about their positions, the re-encrypted block must be no related with the original block ciphertext. Therefore, the combination of block cipher and model of operation is indispensable to the implementation of ORAM schemes. In addition, most of ORAM schemes needs encrypting and decrypting poly-logarithmic blocks for each client's access. Therefore, the encryption and decryption time of combinations of block cipher and model of operation is an essential component of ORAM's end-to-end delay.

We next introduce the block ciphers used in our experiment. One of assumptions in constructing an ORAM scheme is that adversaries play honest-but-curious manners, which means that players always send the original messages but try to learn some sensitive information from the visible data. Therefore, we can say all the mainstream block ciphers satisfy the security requirements of constructing ORAM schemes. We leverage the LibTomCrypt as a filter to help us obtain the block cipher list in our

experiment. In the end, we chose the Data Encryption Standard (DES)[17], Triple Data Encryption Standard (3DES)[18], Rivest Cipher2 (RC2)[19], Rivest Cipher5 (RC5)[20], Blowfish[21], and Advanced Encryption Standard (AES)[22].

The above block ciphers only take data block with fixed size as its input. But in real-world scenario, the block size usually exceeds the block ciphers' standard size of input, which the model of operation was created to address. To our knowledge, the common models of operation include Electronic Code Book (ECB)[23], Cipher Block Chaining (CBC)[11], Cipher Feedback (CFB)[24], Output Feedback(OFB) [11], and Counter (CTR) [11]. We remove the ECB from the candidate list due to its lack of initial vector. In other words, the model of operation ECB cannot generate different ciphertext for same plaintext with fixed key value.

4. The Combination of Block Cipher (RC5) and Model of Operation (CBC)

RC5 is a simple block cipher mode [20]. Ron Rivest invented it and then RSA Laboratories theoretically analyzed and verified. A new feature of RC5 is the heavy use of data-dependent rotations. RC5 has variables word length, variable number of rounds, and variable length keys. There are only three operations in RC5: XOR, Addition, and Rotation. Rotation is a constant time operation on most processors, variable rotation is a non-linear function. The rotation operation depends on the key and data. In 2000, the WAP Forum specified RSA Security's RC5^{TM} encryption algorithm for its Wireless Transport Level Security Specification (WTLS). The block cipher RC5 is the only symmetric encryption algorithm designated by the WAP Forum for the WTLS environment.

Electronic Code Book (ECB) is the simplest block cipher mode of operation, which encrypts each plaintext data block individually [23]. The biggest shortage of ECB is that the same ciphertext will be obtained after encrypting the same plaintext. Therefore, the data patterns encrypted in ECB cannot be completely hidden, making its usage scenarios limited. Currently, ECB is the simplest and most efficient mode of operation. The encryption and decryption process of ECB is described by the following two expressions:

$$C_i = E_k(P_i) \quad (1)$$

$$P_i = D_k(C_i) \quad (2)$$

where P_i and C_i represent the i_{th} block plaintext and ciphertext, respectively. $E_k()$ and $D_k()$ denote the encryption and decryption function interface of a symmetric encryption, respectively.

Cipher Block Chaining (CBC) was designed to address the problem of obtaining the same ciphertext after encrypting the same plaintext in ECB. It reduces the possibility of repeated patterns in the ciphertext [17]. In CBC mode, before a block of plaintext is encrypted, it needs to be XORed with the previous block of ciphertext. The first plaintext block is XORed with the initialization vector (IV).

$$C_i = E_k(P_i \oplus C_{i-1}), \text{with } C_0 = IV \quad (3)$$

$$P_i = D_k(C_i) \oplus C_{i-1}, \text{with } C_0 = IV \quad (4)$$

where $\oplus$ denotes the XOR operation.

From the above description of block cipher RC5 and model of operation CBC, we can see that the combination of them could encrypt any plaintext with any size. In addition, the RC5-CBC also achieve the property of semantic encryption, which means that the ciphertexts generated by the same plaintext via RC5-CBC are no related.

5. Experimental Evaluation

5.1. Evaluation Methodology and Metrics

Evaluation Methodology. We first selected the block ciphers used in our experiment according to the block cipher list in the LibTomCrypt, a popular cryptography toolkit. To make the experimental results more convincing, we chose all the block ciphers in the LibTomCrypt. We second picked out the block cipher models of operation in terms of whether the scheme exists initial vector (IV). Therefore, the model of operation of CBC, CFB, OFB and CTR became our experimental objects. To obtain the best performance of combination of block cipher and model of operation, we, above all,

compared the combinations of each block cipher and four kinds of model of operations and gained the most matchable model of operation for each block cipher. Then, we made a comparison for combinations gained above and found the final combination with best performance from execution time perspective.

The software and hardware configuration. The operating system for our experiments is WSL-based Ubuntu 20.4. We implemented the combination of block cipher and model of operation in C++ language, and the required external library LibTomCrypt, and we disclose the code of this experiment for readers to check and improve. The device of computer used in this experiment is Intel(R) Core (TM) i7-9700 CPU @ 3.00GHz and 16GB RAM.

Evaluation Metrics. What we first need to do is to decide the size of block. According to the block size used in [2], we selected the block size in our experimental ranging from $2KB$ to $512KB$ growing exponentially. More importantly, we chose the encryption and decryption time as an evaluation metric measuring the execution efficiency of all combinations of block cipher and model of operation in our experiment.

5.2. Experimental Results and Analysis

Experimental results show that the RC5-CBC outperforms other combinations of block cipher and model of operation. Figures 1, 2 and 3 show the encryption and decryption time of combinations of each block cipher and its most matchable model of operation with block size ranging from $2KB$ to $512KB$. It is easy for us to find that for all block sizes the execution time of RC5-CBC is less than other combinations. As shown in Figure 3, when the block size is $256KB$, the encryption and decryption time of RC5-CBC is 1398 microseconds, and the encryption and decryption time of the other combinations AES-CBC, Blowfish-CBC, DES-CBC, 3DES-CBC and RC2-CBC is 2356, 2918, 5124, 14558 and 6302 microseconds, respectively.

Experimental results show that model of operation CBC performs better than other models of operation for most of block sizes. As shown in Figure 1, when block size is equal to $8KB$, for all block ciphers CBC is the only model of operation appearing in figure. On the contrast, when face $4KB$ block size, we find that for block ciphers of DES and RC5, the model of operation OFB is better than others. In general, model of operation CBC is the prior choice for any block size and block cipher.

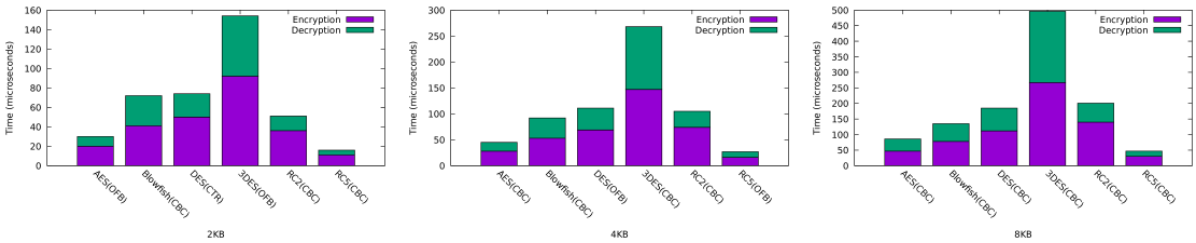

Figure 1 Encryption and decryption time with block size 2KB, 4KB and 8KB

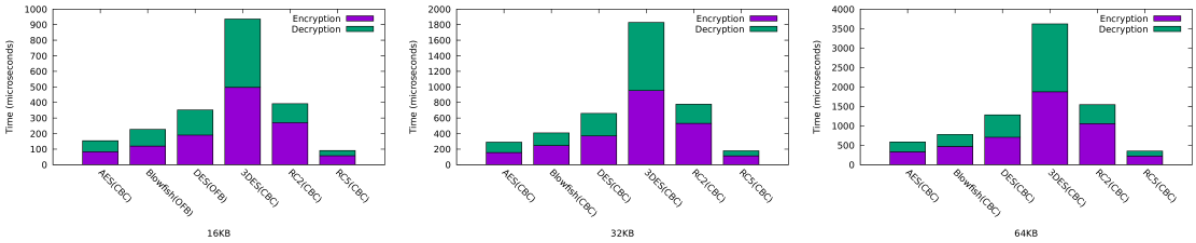

Figure 2 Encryption and decryption time with block size 16KB, 32KB and 64KB

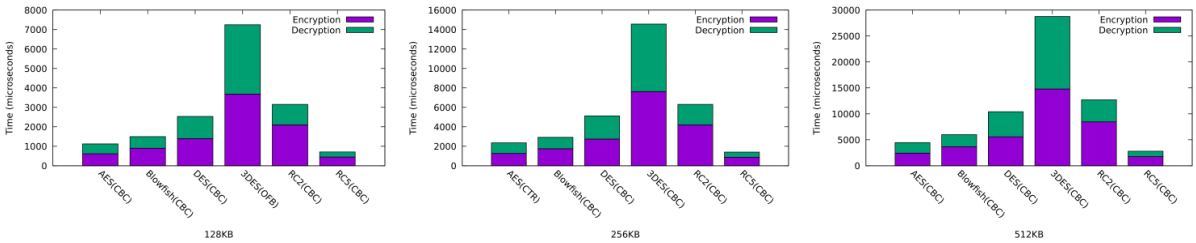

Figure 3 Encryption and decryption time with block size 128KB, 256KB and 512KB

6. Conclusion

In this paper, we first selected the block ciphers used in our experiment based on the block cipher list in LibTomCrypt. We second chose the block cipher models of operation according to the requirements of semantically secure encryption techniques. We third compared and analysed the encryption and decryption efficiency of combinations of each block cipher and four models of operation, CBC, CFB, CTR and OFB. Finally, we compared the block ciphers with their most matchable model of operation. For example, when the block size is 512KB, we compared the performance of AES-CBC, Blowfish-CBC, DES-CBC, 3DES-CBC, RC2-CBC and RC5-CBC. And we find that the combination of block cipher and model of operation RC5-CBC is the optimal choice for implementing ORAM schemes.

7. References

[1] Chang, D. Xie, and F. Li, “Oblivious RAM: a dissection and experimental evaluation,” Proc. VLDB Endow., vol. 9, no. 12, pp. 1113–1124, Aug. 2016

[2] T. Hoang, A. A. Yavuz, and J. Guajardo, “A Multi-server ORAM Framework with Constant Client Bandwidth Blowup,” ACM Trans. Priv. Secur., vol. 23, no. 1, p. 1:1-1:35, 2020

[3] O. Goldreich and R. Ostrovsky, “Software protection and simulation on oblivious RAMs,” J. ACM, vol. 43, no. 3, pp. 431–473, May 1996

[4] Z. Liu et al., “Eurus: Towards an Efficient Searchable Symmetric Encryption With Size Pattern Protection,” IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 3, pp. 2023–2037, 2022

[5] E. Stefanov et al., “A Retrospective on Path ORAM,” IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 39, no. 8, pp. 1572–1576, Aug. 2020

[6] L. Ren et al., “Constants count: practical improvements to oblivious RAM,” in Proceedings of the 24th USENIX Conference on Security Symposium, USA, pp. 415–430, 2015

[7] L. Zhang, G. Zeng, Y. Chen, N. Cao, S.-M. Yiu, and Z. Liu, “OC-ORAM: Constant Bandwidth ORAM with Smaller Block Size using Oblivious Clear Algorithm:,” in Proceedings of the 16th International Joint Conference on e-Business and Telecommunications, Prague, Czech Republic, pp. 149–160, 2019

[8] S. Zahur et al., “Revisiting Square-Root ORAM: Efficient Random Access in Multi-party Computation,” in 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, pp. 218–234, May 2016

[9] I. Abraham, C. W. Fletcher, K. Nayak, B. Pinkas, and L. Ren, “Asymptotically Tight Bounds for Composing ORAM with PIR,” in Public-Key Cryptography - PKC 2017 - 20th IACR International Conference on Practice and Theory in Public-Key Cryptography, Amsterdam, The Netherlands, Proceedings, Part I, 2017, vol. 10174, pp. 91–120, March 28-31, 2017

[10] E. Shi, T.-H. H. Chan, E. Stefanov, and M. Li, “Oblivious RAM with O((logN)3) Worst-Case Cost,” in Advances in Cryptology – ASIACRYPT 2011, Berlin, Heidelberg, pp. 197–214, 2011

[11] S. Almuhammadi and I. Al-Hejri, “A comparative analysis of AES common modes of operation,” in 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1–4, Apr, 2017,

[12] E. Stefanov et al., “Path ORAM: an extremely simple oblivious RAM protocol,” in Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, New York, NY, USA, pp. 299–310, Nov, 2013
[13] Y. Gong, F. Gao, W. Li, H. Zhang, Z. Jin, and Q. Wen, “LPS-ORAM: Perfectly Secure Oblivious RAM with Logarithmic Bandwidth Overhead,” Security and Communication Networks, vol. 2022, pp. 1–12, Aug. 2022
[14] J. Raigoza and K. Jituri, “Evaluating Performance of Symmetric Encryption Algorithms,” in 2016 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 1378–1379, 2016
[15] N. Singhal and J. P. S. Raina, “Comparative Analysis of AES and RC4 Algorithms for Better Utilization,” International Journal of Computer Trends and Technology, vol. 1, no. 3, pp. 259–263, 2011
[16] Q. Li, C. Zhong, K. Zhao, X. Mei, and X. Chu, “Implementation and Analysis of AES Encryption on GPU,” in 2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems, pp. 843–848, Jun, 2012
[17] S. Jindal and M. Sharma, “Design and Implementation of Kerberos using DES Algorithm,” in Proceedings of the ACM Symposium on Women in Research 2016, New York, NY, USA, pp. 92–95, 2016
[18] P. Patil, P. Narayankar, Narayan D.G., and Meena S.M., “A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish,” Procedia Computer Science, vol. 78, pp. 617–624, Jan. 2016
[19] N. A. Sharma and M. Farik, “A Performance Test On Symmetric Encryption Algorithms - RC2 Vs Rijndael,” International Journal of Scientific & Technology Research, vol. 6, no. 07, p. 3, 2017
[20] T. Nie, Y. Li, and C. Song, “Performance Evaluation for CAST and RC5 Encryption Algorithms,” in 2010 International Conference on Computing, Control and Industrial Engineering, vol. 1, pp. 106–109, Jun, 2010
[21] R. R. Corpuz, B. D. Gerardo, and R. P. Medina, “Using a Modified Approach of Blowfish Algorithm for Data Security in Cloud Computing,” in Proceedings of the 6th International Conference on Information Technology: IoT and Smart City, New York, NY, USA, pp. 157–162, 2018
[22] C. Sanchez-Avila and R. Sanchez-Reillol, “The Rijndael block cipher (AES proposal) : a comparison with DES,” in Proceedings IEEE 35th Annual 2001 International Carnahan Conference on Security Technology (Cat. No.01CH37186), pp. 229–234, 2001
[23] P. Crescenzi and G. Innocenti, “Development of an ECB on Computer Networks Based on WWW Technologies, Resources and Usability Criteria,” in Proceedings of the International Conference on Computers in Education, USA, p. 1198, 2002
[24] T. Hwang and P. Gope, “RT-OCFB: Real-Time Based Optimized Cipher Feedback Mode,” Cryptologia, vol. 40, no. 1, pp. 1–14, 2016