A Block Cipher Mode of Operation with Two Keys

Yi-Li Huang, Fang-Yie Leu, Jung-Chun Liu, Jing-Hao Yang

A Block Cipher Mode of Operation with Two Keys

Yi-Li Huang, Fang-Yie Leu, Jung-Chun Liu, Jing-Hao Yang

Department of Computer Science, TungHai University, Taiwan
{yifung, leufy, jcliu, g01350036}@thu.edu.tw

Abstract. In this paper, we propose a novel block cipher mode of operation (BCMO for short), named Output Protection Chain (OPC for short), which as a symmetric encryption structure is different from other existing BCMOs in that it employs two keys, rather than one key, to protect the output of the mode. The security threats of chosen-plaintext attacks on three existing common BCMOs, including the Cipher Feedback mode (CFB), the Output Feedback mode (OFB), and the Counter mode (CTR), are also analyzed. After that, we explain why the OPC mode (or simply the OPC) can effectively avoid chosen-plaintext attacks, and why its security level is higher than those of CFB, OFB, and CTR.

***Keywords*:** Block cipher, Cipher Feedback mode, Output Feedback mode, Counter mode, Output Protection Chain mode, chosen-plaintext attack

1 Introduction

When standard block cipher algorithms, like Data Encryption Standard (DES), Triple Data Encryption Algorithm (3DES), and Advanced Encryption Standard (AES), are used to encrypt a plaintext block, the size of the block should be the same as the length of the encrypting key (or called the ciphering block) L. If the size exceeds L, we have to divide the plaintext block into sub-blocks. Each is L in length. Several BCMOs defined by National Institute of Standards and Technology (NIST) have been widely adopted by different block cipher techniques [2]. Through the use of these BCMOs, these techniques can be then applied to many applications.

Generally, the standard BCMOs include the cipher Feedback mode (CFB for short), the Output Feedback mode (OFB for short) and the Counter mode (CTR for short), the characteristics of which are that they use only one key to encrypt multiple plaintext blocks, and the efficiencies of their block cipher algorithms are high [1]. Currently, different types of attacks on these BCMOs have been developed [3] [4], meaning the BCMOs have their own security problems. Therefore, in this study, we propose a novel BCMO, named Output Protection Chain (OPC for short), to solve the existing BCMOs' security problems. Two different structures of the OPC, named OPC-1 and OPC-2, have been developed to enhance the security levels of BCMOs. We will describe the two structures later.

2 Block Cipher Modes of Operation

Before describing operations of the CFB, OFB, and CTR, we first define the parameters used by them.

P_i : The i^{th} plaintext block to be encrypted, $1 \le i \le n$.

C_i : The i^{th} ciphertext block, $1 \le i \le n$.

Block Cipher Encryption (BCE) unit: According to [2], the standard BCE units are AES-128, AES-192, and AES-256. The function of a BCE unit is denoted by $E(I_P, K)$, in which the key K and the input I_p are used to encrypt a given plaintext block.

K: The block cipher key [2].

O_i: The output block produced by invoking the $E(I_p, K)$, $1 \le i \le n$.

cr: The counter, which is an input of the BCE unit of the CTR.

IV: Initialization Vector (IV for short), a random value employed by the CFB and OFB since they need an additional initial input.

The general rule in CFB is that $E(C_{i-1}, K)$ receives C_{i-1} and K as its inputs to generate O_i which is then XORed with P_i to produce C_i, $1 \le i \le n$, where $C_0 = IV$. The process can be formulated as follows.

$$C_i = P_i \oplus E(C_{i-1}, K) = P_i \oplus O_i \quad (1)$$

The encryption operations of the OFB are similar to those of the CFB. The difference is the inputs of $E(I_P, K)$. In the OFB, O_{i-1}, rather than C_{i-1}, is fed back to the BCE unit to generate O_i, $1 \le i \le n$, where $O_0 = IV$. It can be formulated as follows.

$$C_i = P_i \oplus E(O_{i-1}, K) = P_i \oplus O_i \quad (2)$$

The CTR encryption replaces the feedback operation employed by the CFB and OFB with a counter cr as one of the inputs of the BCE unit to generate O_i. The value of the counter used to generate O_i is $cr + i - 1$ where cr is the value adopted to produce O_i , $1 \le i \le n$. The formulas utilized to encrypt plaintext blocks of the CTR are as follows.

$$C_i = P_i \oplus E(cr + i - 1, K) \quad (3)$$

3 The Output Protection Chain (The OPC)

In this section, we describe how to encrypt plaintext blocks and decrypt ciphertext blocks in the proposed OPC structures, i.e., OPC-1 and OPC-2. We first define those parameters and functions invoked by the OPCs.

The definitions of P_i, C_i, BCE units, $E(I_p, K)$ and O_i, $1 \le i \le n$, are the same as those defined above. New parameters, operations and functions are defined below.

Key1: The block cipher key, the role of which is the same as K defined above.

$D(I_p, K)$: Function of the block decipher, in which the key K and an input I_P are used to decrypt a plaintext block from its ciphertext block.

G_i: The data block produced by $O_i \oplus P_i$, $1 \le i \le n$.

Key2: A key with the length the same as that of O_i. It is used to encrypt G_1 in the OPC-1, and O_1 in the OPC-2.

$+_2$: a binary adder, which is a logical operator defined in [5].

$-_2$: The Inverse operation of $+_2$.

3.1 The OPC-1

As shown in Fig.1, the general rule of the OPC-1 is that Key1 and G_{i-1} are input to the BCE unit to generate O_i, which is XORed with P_i to produce G_i. G_i is then binary-added with O_{i-1} to generate C_i, $1 \le i \le n$. The formulas derived are as follows.

$$C_i = [E(G_{i-1}, Key1) \oplus P_i] +_2 O_{i-1} = G_i +_2 O_{i-1} \quad (4)$$

where $G_0 = IV$ and $O_0 = Key2$. The decryption process as shown in Fig. 2 can be formulated as follows.

$$P_i = O_i \oplus G_i = E(G_{i-1}, Key1) \oplus (C_i -_2 O_{i-1}) \quad (5)$$

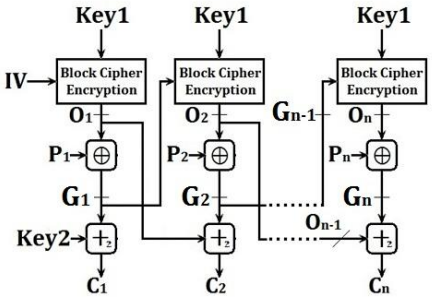

Fig. 1. The OPC-1 encryption

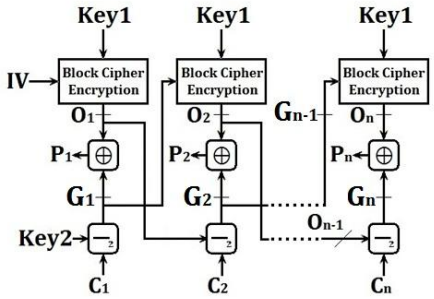

Fig. 2. The OPC-1 decryption

3.2 The OPC-2

The encryption process of the OPC-2 is shown in Fig. 3. The general rule is that P_i and Key1 are input to the BCE unit to generate O_i, which is XORed with O_{i-1} to generate C_i,$1 \le i \le n$. It can be formulated as follows.

$$C_i = O_i \oplus O_{i-1} = E(P_i, Key1) \oplus O_{i-1} \quad (6)$$

where $O_0 = Key2$. The decryption structure of the OPC-2 as shown in Fig. 4 is as follows. To decrypt C_i, one needs O_{i-1} to calculate O_i because $O_i = C_i \oplus O_{i-1}$, $1 \le i \le n$. P_i can be obtained by invoking the following formulas.

$$P_i = D(C_i \oplus O_{i-1}, Key1) = D(O_i, Key1) \quad (7)$$

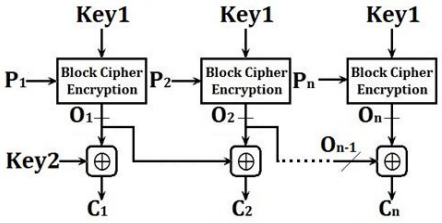

Fig. 3. The OPC-2 encryption

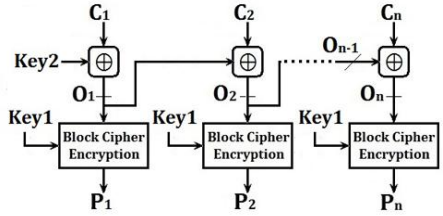

Fig. 4. The OPC-2 decryption

4 Security Analysis

The advantage of using BCMOs is that these BCMOs can enhance security of a single block's encryption. Even if the block cipher (e.g. DES) has been cracked, in order to improve the security level of a security system, one can apply the DES as the BCE unit to the BCMOs. We will analyze the security of BCMOs mentioned above in the following subsections.

4.1 Security of the CFB

To launch a chosen-plaintext attack, an attacker first inputs n different plaintext blocks, denoted by $P = \{P_1, P_2, \dots, P_n\}$, to acquire a set of n ciphertext blocks, denoted by $C = \{C_1, C_2, \dots, C_n\}$, where P_i is the i^{th} block of P, and C_i is the i^{th} block of C, $1 \le i \le n$. In the CFB, O_i can be derived from P_i and C_i since $O_i = P_i \oplus C_i$. If n is huge, the attacker can then collect sufficient $\langle C_{i-1}, O_i \rangle$ pairs, as the input and output of the BCE unit when encrypting P_i, to analyze the value of the key K.

4.2 Security of the OFB

For the OFB, we analyze its security based on two cases, one is that the IV can be chosen by users, and the other is cannot be chosen.

4.2.1 Attack on IV able to be chosen. In the OFB, O_i, $1 \le i \le n$, is only determined by IV and K. If IV can be chosen by users, the attacker can select the IV the same as the one chosen by a user, i.e., the victim, to encrypt their chosen-plaintext and calculate O_i by using $P_i \oplus C_i$.

Since K and the encryption algorithm of the BCE unit when encrypting different plaintext blocks are themselves the same, that means once the chosen IVs for encrypting two plaintexts are the same. When O_is of the BCE unit are acquired, the attacker can use an illegally intercepted C_i to search the corresponding O_i from all its collected $\langle C_i, O_i \rangle$ pairs to derive P_i without requiring breaking the key K of the BCE unit, since $P_i = C_i \oplus O_i$.

4.2.2 Attack on IV unable to be chosen. If the IV cannot be chosen, the security level of the OFB is higher. But it still faces the same security problem of the CFB. Like that in attacking the CFB, the attacker can first input a long plaintext, $P = \{P_1, P_2, \dots, P_n\}$, to acquire the corresponding ciphertext, $C = \{C_1, C_2, \dots, C_n\}$, so as to generate a set of $O = \{O_1, O_2, \dots, O_n\}$ since $O_i = P_i \oplus C_i$.

If n is huge enough, the attacker can then collect sufficient $\langle O_{i-1}, O_i \rangle$ pairs to analyze the key K of its BCE unit. After that, when the attacker eavesdrops the messages delivered between the sender and receiver, and retrieves the IV, he/she can generate O to decrypt the intercepted C so as to obtain P since $P = C \oplus O$.

4.3 Security of the CTR

Like that in the OFB, the CTR can also be divided into two cases, i.e., a user can and cannot choose the value of cr.

4.3.1 Attack on cr able to be chosen. In the CTR encryption, O_is are determined

only by cr and K in which cr is an incremental integer (will be transformed into a bit string) and K is a fixed key. The general rule is that given a chosen cr, $E(cr + i - 1, K)$ receives $cr + i - 1$ and K, $1 \le i \le n$, as its inputs to generate a set of output $O = \{O_1, O_2, \ldots, O_n\}$, $n \gg 1$, without requiring inputting any plaintext. On the other hand, for the chosen cr, the attacker can choose a plaintext $P = \{P_1, P_2, \ldots, P_n\}$ for the CTR to generate a ciphertext $C = \{C_1, C_2, \ldots, C_n\}$. After that, for each i, $1 \le i \le n$, $O_i = C_i \oplus P_i$. Then the attacker can acquire O.

If users can choose cr, then the attacker can also choose a cr the same as that of a user, i.e., the victim, to obtain O corresponding to this cr by using the above process. Now the attacker can decrypt the plaintext block P_i from the ciphertext block C_i intercepted from the user by using $P_i = C_i \oplus O_i$.

4.3.2 Attack on cr unable to be chosen. If the cr cannot be chosen, the attacker is still able to know the cr. Because cr is delivered together with C_is to the receiver with cr unencrypted [6], the attacker can obtain n input blocks, i.e., $I_p = \{cr, cr + 1, \ldots, cr + i - 1\}$, $n \gg 1$, from those messages carrying crs and C_is.

On the other hand, the attacker can input a set of plaintext blocks $P = \{P_1, P_2, \ldots, P_n\}$ to the CTR to obtain the corresponding ciphertext blocks, denoted by $C = \{C_1, C_2, \ldots, C_n\}$. He/she can then acquire a set of outputs of the BCE unit, denoted by $O = \{O_1, O_2, \ldots, O_n\}$, since $O_i = P_i \oplus C_i$, $1 \le i \le n$. As a result, the attacker can analyze the key K used by the BCE unit after collecting a large number of $\langle cr + i - 1, O_i \rangle$ pairs.

4.4 Security of the OPC-1

In Fig. 1, we use Key2 to protect G_1 and produce C_1, where$G_1 = P_1 \oplus O_1$. After that, O_i, $i > 1$, as the new Key2 of the next encryption round, is used to encrypt G_{i+1} to generate C_{i+1}. The advantage is that, when a large number of chosen-plaintext is input, C_i collected by the attacker is the one encrypted by O_{i-1} or Key2 (when $i = 1$). So there is no way for the attacker to decrypt G_is by using an inverse operation on C_is without knowing Key2 beforehand.

Moreover, G_i is fed back to generate O_{i+1}. The purpose is to increase the complexity of solving Key1. Also, G_i is encrypted by O_{i-1} or Key2, resulting in the fact that it is hard for the attacker to analyze the relationship between P_i and C_i, $1 \le i \le n$.

The shortcoming of the OPC-1 is that the plaintext is not encrypted by the BCE unit so the possibility for the attacker to decrypt P_i from C_i is still high since the attacker does not need to decrypt the BCE unit.

4.5 Security of the OPC-2

Fig. 3 shows the OPC-2 encryption, in which Key2 is used to encrypt O_1 so as to produce C_1. Meanwhile, O_1 is also used to encrypt O_2, i.e., $C_i = O_i \oplus O_{i-1}$, $1 < i \le n$. As with the OPC-1, it is hard for the attacker to acquire O_i by decrypting C_i since O_i is encrypted by O_{i-1} or Key2, even he/she has collected a large number of ciphertext by inputting many chosen-plaintexts to the OPC-2.

If the attacker wishes to analyze the BCE unit of the OPC-2, he/she can input a long plaintext $P = \{P_1, P_2, \ldots, P_n\}$ of n plaintext blocks to the OPC-2 to generate the

corresponding ciphertext $C = \{C_1, C_2, \ldots, C_n\}$. But before generating the set of output $O = \{O_1, O_2, \ldots, O_n\}$ of the BCE unit, he/she still needs to know Key2 because $O_1 = C_1 \oplus \text{Key2}$, and $O_i = C_i \oplus O_{i-1}$, $1 < i \leq n$. In our design, all encryption and decryption steps are dependent, so it is impossible to acquire O_1 without knowing Key2. Moreover, P_i is also protected by the BCE unit. With this, OPC-2 effectively strengthens the security level of O_i. As a result, it is hard for the attacker to collect sufficient information to analyze Key1 of the BCE unit.

5 Conclusions and Future Studies

In this paper, we describe the security drawbacks of the standard BCMOs, and propose the OPCs to improve the security level of a block ciphering system by protecting the outputs of its BCE unit, i.e., O_is, without the need of preventing the attacker from collecting P_is, C_is and their relationship. The purpose is avoiding the security system from being attacked by known or chosen-plaintext/ciphertext attacks.

However, in the OPC-2, the BCE unit must be invertible, e.g., DES, 3-DES, or AES. Otherwise, the plaintext P_i cannot be reverted from O_i. Since the encryption speeds of non-invertible algorithms are often short, and their encryption keys are difficult to crack, if one replaces the BCE unit of the CFB, OFB, CTR or OPC-1 with a non-invertible algorithm, the security levels and the processing performance of these BCMOs will be then higher than before. Therefore, in the future, we will apply non-invertible algorithms to the OPC-1 so as to propose a new BCMO with the security at least the same as or higher than those of the two OPCs.

Acknowledgements:

This research was partially supported by TungHai University on GREENs project, and National Science Council, Taiwan, grants NSC 100-2221-E-029-018 and 101-2221-E-029-003-MY3.

6 Reference

1. W. Stallings, Cryptography and Network Security: Principles and Practice, Fifth Edition, Publisher: Prentice Hall, January, 2010.
2. National Institute of Standards and Technology, NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation Methods and Techniques, December, 2001.
3. H. Hudde, "Building Stream Ciphers from Block Ciphers and their Security," Seminararbeit Ruhr-Universität Bochum, February, 2009. http://imperia.rz.rub.de:9085/imperia/md/content/seminare/itsws08_09/hudde.pdf
4. D. Wang, D. Lin, and W. Wu, "Related-Mode Attacks on CTR Encryption Mode," International Journal of Network Security, Vol.4, No.3, PP.282–287, May 2007.
5. Y.F. Huang, F.Y. Leu, C.H. Chiu and I.L. Lin, "Improving Security Levels of IEEE802.16e Authentication by Involving Diffie-Hellman PKDS," Journal of Universal Computer Science, vol. 17, no.6, March 2011, pp. 891-911.
6. H. Lipmaa, P. Rogaway, D. Wagner, "Comments to NIST concerning AES Modes of Operations: CTR-Mode Encryption", 2000. http://csrc.nist.gov/