Security Analysis of Image Observation on the Modified ECB Operations in Advanced Encryption Standard

Girish Dhoble and Manish Choudhary
Georgia Institute of Technology

ABSTRACT

This paper presents security analysis of a paper "Image Observation on the Modified ECB Operations in Advanced Encryption Standard"[1] by Huang et. al. which suggests modified ECB operations for image encryption by introducing some variants of initialization vectors. In the paper [1], the authors suggests seven different schemes out of which we found that two schemes were insecure under the notion of IND-CPA security and all schemes were insecure under the notion of IND-CCA security.

I. INTRODUCTION

Due to growth of multimedia application, more and more digital images are being stored on servers or sent over computer networks. Images are used in many fields such as medical science, military, social media and e-commerce. Medical imaging has enabled the doctors to look inside a patient, to learn more about neurobiology and human behaviours. At the same time, images are of great usage in military in form of maps and strategic plans. In the digital world, the privacy of sensitive image data like patients' health information or military plans is one of the most important concerns. Image encryption is one of the ways to achieve the objective of image data privacy. There have been many papers published in the field of image encryption, providing different cryptographic encryption schemes. The paper "Image Observation on the Modified ECB Operations in Advanced Encryption Standard"[1] by Huang et. al. suggests modified ECB operations for image encryption by introducing some variants of initialization vectors. The security analysis of this published paper [1] has been covered in our paper. The rest of the paper is divided in following sections. First, the work produced by the authors has been discussed, followed by the standard security definitions used for the analysis of the suggested schemes. Then, assumptions made in the paper, schemes suggested by the authors and their security analysis has been covered in order. Finally, the results, concluding remarks and some recommendations have been provided.

II. ABOUT AUTHORS' PAPER

The authors suggest that the use of 'Electronic Codebook Mode (ECB) based on AES block cipher' for image encryption is not secure. If the plain image contains some groups of identical colours, some kind of patterns may appear at the related areas in the encrypted cipher image. Therefore, the ECB mode with AES is not able to prevent the leakage of information from the encrypted image. To overcome this limitation, an approach is to make those identical colours different or to remove them. For making those inputs different before encryption, one of the following three number sequences can be added to the image inputs.

1. **Arithmetical sequence:** Sequential number from counter output i.e. ($\mathbf{T_j}$ =IV+j, for j=l to n) where IV is some random initialization vector. The sequence of random numbers {IV,IV+1, IV+2…} is either xored with the input plaintext or is directly given to encryption function and then xored with the plaintext. In both the cases, the output would be random.

2. **Arithmetical series:** Non-sequential number from the accumulator output i.e. (**Sj**=IV+ $\sum$ (j)). The arithmetical series of random numbers {IV, IV+1, IV+1+2…} is either xored with the input plaintext or is directly given to encryption function and then xored with the plaintext which produces random output.

3. **Random sequence:** Random numbers generated from cipher function itself can be xored to plaintext blocks.

By using different number sequences and some additional operations on ECB, the authors suggest seven different schemes which are, Counter Input Modification(CIM), Summation Input Modification(SIM), Random Input Modification(RIM), Counter Output Modification(COM), Summation Output Modification(SOM), Random Output Involved Pj (ROP) and Random Output not Involved Pj (RNP). All these schemes and their security analysis have been discussed in the later sections. After some entropy calculations, the authors conclude that there is no pattern appearing in the cipher images produced on encrypting the images with these modified ECB schemes.

III. STANDARD SECURITY DEFINITIONS

The security definitions used in this paper for the purpose of security analysis are as follows:

Definition 1**:** Let SE = (K,E,D) be a symmetric encryption scheme, and let A be an algorithm(adversary) that has access to an LR (Left-Right oracle) oracle. Considering two experiments, $Exp_{SE}^{ind\text{-}cpa\text{-}1}$(always encrypts the right message of the query) and $Exp_{SE}^{ind\text{-}cpa\text{-}0}$ (always encrypts the left message of the query), the advantage of any adversary can be defined as:

$$Adv_{SE}^{ind\text{-}cpa}(A) = Pr[Exp_{SE}^{ind\text{-}cpa\text{-}1}(A) = 1] - Pr[Exp_{SE}^{ind\text{-}cpa\text{-}0}(A) = 1]$$

The symmetric encryption scheme SE is indistinguishable under chosen-plaintext attacks (IND-CPA secure) if for any adversary A with "reasonable" resources, the $Adv_{SE}^{ind\text{-}cpa}$ is small (close to 0). The resources are defined by the number of queries, the total length of the queries and the time.

Definition 2: Let SE = (K,E,D) be a symmetric encryption scheme, and let A be an algorithm(adversary) that has access to an LR (Left-Right oracle) oracle and a decryption oracle. Considering two experiments, $Exp_{SE}^{ind\text{-}cca\text{-}1}$ and $Exp_{SE}^{ind\text{-}cca\text{-}0}$, the advantage of any adversary can be defined as:

$$Adv_{SE}^{ind\text{-}cca}(A) = Pr[Exp_{SE}^{ind\text{-}cca\text{-}1}(A) = 1] - Pr[Exp_{SE}^{ind\text{-}cca\text{-}0}(A) = 1]$$

The symmetric encryption scheme SE is indistinguishable under chosen-cipher text attacks (IND-CCA secure) if for any adversary A with "reasonable" resources, the $Adv_{SE}^{ind\text{-}cca}$ is small (close to 0) i.e. allowed to query message-pairs to the LR oracle and to query some cipher text (not output from the LR oracle) to the decryption oracle, it should be hard for adversary A to guess whether left or right message was encrypted. The resources are defined by the number of queries to both LR and decryption oracle, the total length of the queries and the time.

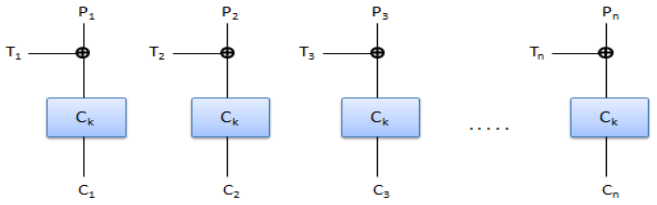
IV. ASSUMPTIONS

It has not been specified by the authors how the initialization vector is chosen. This can easily be argued that it itself introduces some weakness in the schemes which may lead to the adversaries having significant advantage in breaking the security of the encryption. For example, CBC with random initialization vector is IND-CPA secure but CBC with counter initialization vector is not IND-CPA secure. For, stronger security notion, it has been assumed that the initialization vector has been chosen at random so that other attack vectors can be explored.

V. ANALYSIS OF SCHEMES

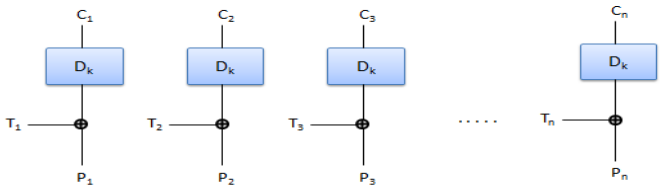
The seven schemes suggested by the authors in the paper [1] have been analysed in this paper under the notions of IND-CPA and IND-CCA security. C_k is encryption function, D_k is decryption function, P_j is j^{th} plaintext block, C_j is j^{th} cipher text block. Rest of the notations have been described in the Appendix.

1. Counter Input Modification (CIM)



$T_j = \{ IV, IV+1, IV+2, ..., IV + n-1\}$

Encryption: $C_j = C_k (T_j \oplus P_j)$



$T_j = \{ IV, IV+1, IV+2, ..., IV + n-1\}$

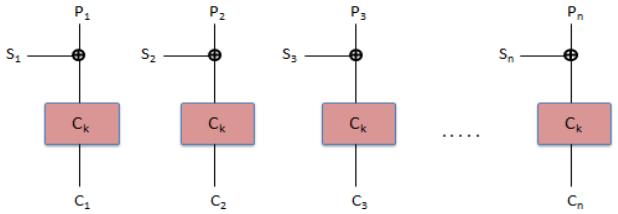
Decryption: $Pj = Dk (C_j) \oplus T_j$

Analysis of CIM

Proposition 1.1: The Counter Input Modification (CIM) Scheme is not secure under IND-CPA notion of security. *Proof of proposition 1.1 in Appendix A*

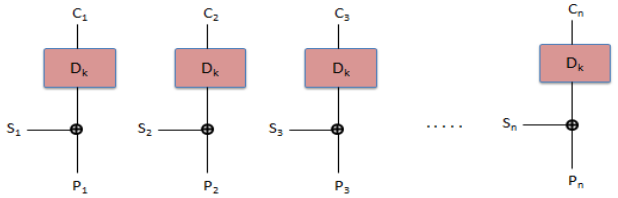
Proposition 1.2: The Counter Input Modification (CIM) scheme is not secure under the IND-CCA notion of security. *Proof of proposition 1.2 in Appendix A*

2. Summation Input Modification (SIM)



$S_j = \{ IV, IV+1, IV+ 1+ 2, IV +1 + 2 +3, ..., IV + 1 + . . + n-1\}$

Encryption: $C_j = C_k (S_j \oplus P_j)$



$S_j = \{ IV, IV+1, IV+ 1+ 2, IV +1 + 2 +3, ..., IV + 1 + . . + n-1\}$

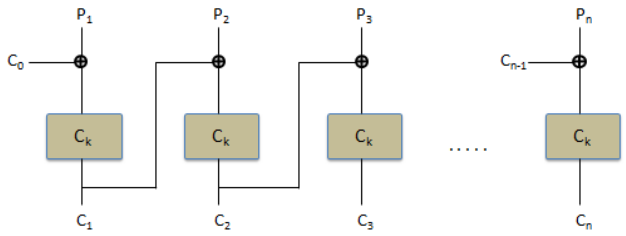
Decryption: $Pj = Dk (Cj) \oplus Sj$

Analysis of SIM

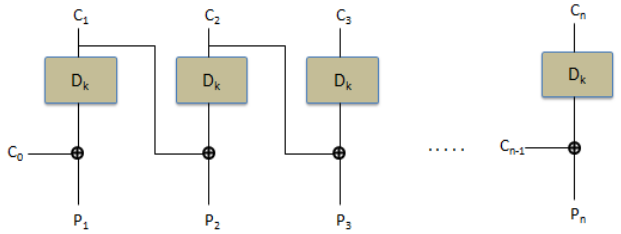
*Proposition 2.1***:** The 'Summation Input Modification' scheme SE is not secure under the IND-CPA notion of security. *Proof of proposition 2.1 in Appendix B*

Proposition 2.2: The 'Summation Input Modification' scheme SE is not secure under the IND-CCA notion of security. *Proof of proposition 2.2 in Appendix B*

3. Random Input Modification



Encryption: $C_j = C_k (\mathbf{C_{j-1}} \oplus P_j)$



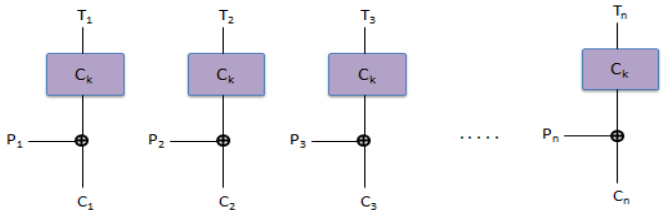
Decryption: $Pj = Dk (C_j) \oplus C_{j-1}$

Analysis of RIM

Proposition 3.1: The Random Input Modification (RIM) scheme is secure under IND-CPA notion of security if the underlying block cipher is PRF secure. *Proof of proposition 3.1 in Appendix C*

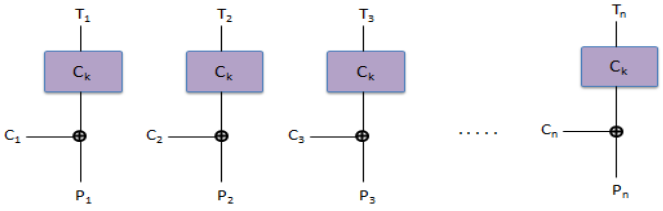
Proposition 3.2: The Random Input Modification (RIM) scheme is not secure under the IND-CCA notion of security. *Proof of proposition 3.2 in Appendix C*

4. Counter Output Modification (COM)



$T_j = \{ IV, IV+1, IV+2, ..., IV + n-1\}$

Encryption: $C_i = C_k (T_i) \oplus P_i$



$T_j = \{ IV, IV+1, IV+2, ..., IV + n-1\}$

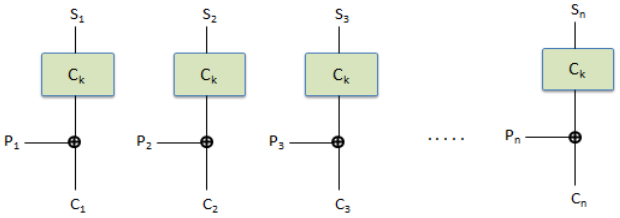
Decryption: $Pj = C_k (T_j) \oplus C_j$

Analysis of COM

Proposition 4.1: The Counter Output Modification (COM) scheme is secure under IND-CPA notion of security if the underlying block cipher is PRF secure. *Proof of proposition 4.1 in Appendix D*

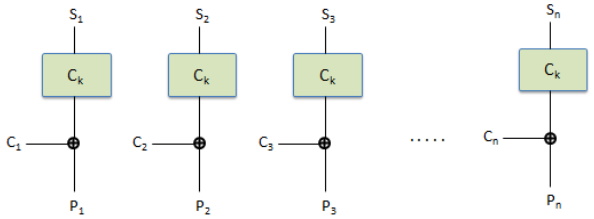
Proposition 4.2: The Counter Output Modification (COM) scheme is not secure under the IND-CCA notion of security. P*roof of proposition 4.2 in Appendix D*

5. Summation Output Modification (SOM)



$S_j = \{ IV, IV+1, IV+ 1+ 2, IV +1 + 2 +3, ..., IV + 1 + . . + n-1\}$

Encryption: $C_j = C_k (\mathbf{S_j}) \oplus P_j$



$S_j = \{ IV, IV+1, IV+ 1+ 2, IV +1 + 2 +3, ..., IV + 1 + . . + n-1\}$

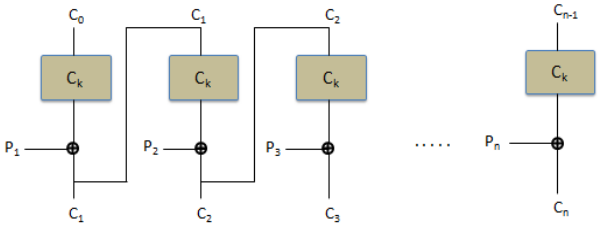
Decryption: $Pj = C_k (S_j) \oplus C_j$

Analysis of SOM

Proposition 5.1: The Summation Output Modification (SOM) scheme is secure under IND-CPA notion of security if the underlying block cipher is PRF secure. *Proof of proposition 5.1 in Appendix E*

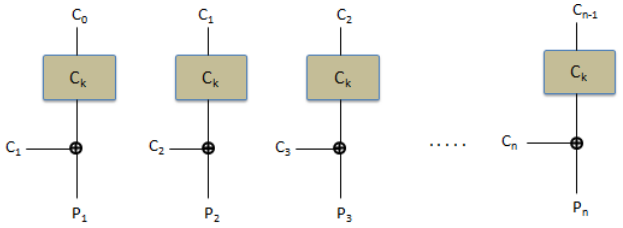
Proposition 5.2: The Summation Output Modification (SOM) scheme is not secure under the IND-CCA notion of security. *Proof of proposition 5.2 in Appendix E*

6. Random Output Involved P_j (ROP)



$C_0 = IV$ where IV is chosen randomly

Encryption: $C_j = C_k (\mathbf{C_{j-1}}) \oplus P_j$



C_0 = IV where IV is chosen randomly

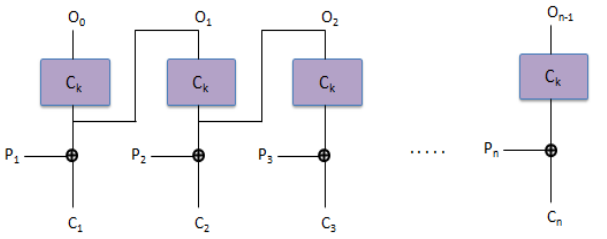
Decryption: $Pj = C_k(C_{j-1}) \oplus C_j$

Analysis of ROP

Proposition 6.1: The Random Output Involved P_j (ROP) scheme is secure under IND-CPA notion of security if the underlying block cipher is PRF secure. *Proof of proposition 6.1 in Appendix F*

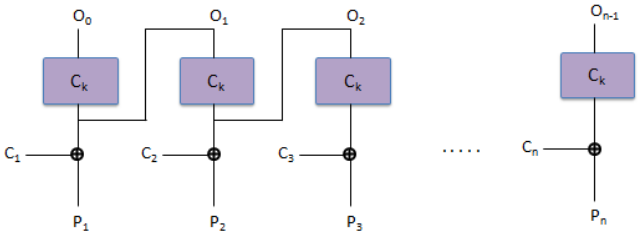
Proposition 6.2: The Random Output Involved P_j (ROP) scheme is not secure under the IND-CCA notion of security. *Proof of proposition 6.2 in Appendix F*

7. Random Output not Involved P_j (RNP)



O_0 = IV where IV is chosen randomly

Encryption: $C_j = C_k(\boldsymbol{O_{j-1}}) \oplus P_j$



O_0 = IV where IV is chosen randomly

Decryption: $Pj = C_k(O_{j-1}) \oplus C_j$

Analysis of RNP

Proposition 7.1: The Random Output not Involved P_j (RNP) scheme is secure under IND-CPA notion of security if the underlying block cipher is PRF secure. *Proof of proposition 6.1 in Appendix F*

Proposition 7.2: The Random Output not Involved P_j (RNP) scheme is not secure under the IND-CCA notion of security. *Proof of proposition 6.2 in Appendix F*

VI. SUMMARY OF OUR RESULTS

Scheme	IND-CPA secure?	IND-CCA secure?
CIM	NO	NO
SIM	NO	NO
RIM	YES	NO
COM	YES	NO
SOM	YES	NO
ROP	YES	NO
RNP	YES	NO

VII. CONCLUSION

Image encryption is important for the security of confidential image data. ECB cannot be used for this purpose as it is not able to prevent cipher image leakage. The seven encryption modes suggested are modified ECB modes using IV variants and some additional properties. These schemes add some number sequence to make the input to the cipher function different and thus, are successful in providing high entropy and chucking out the patterns occurring in the encrypted image. But, this doesn't guarantee the security of encryption schemes unless they have been evaluated as per the standard security definitions. As analysed in the above section, the two modes CIM and SIM don't provide IND-CPA security and thus, it is recommended not to use these encryption modes to provide confidentiality. The other 5 modes are definitely IND-CPA secure with condition for some modes that the initialization vector should be chosen at random. It is advised not to use RIM if counter initialization vector is supposed to be used. COM is an efficient encryption mode and ensures IND-CPA security even with counter initialization vector and thus, recommended for use.

VIII. ACKNOWLEDGEMENTS

We would like to thank our mentor, Professor A. Boldyreva, for her guidance and support in learning the various standard security notions and the approach to analyse the cryptographic security of different practically implemented schemes and protocols, under the standard security definitions.

We are also grateful to Sara Krehbiel for her continuous support and feedback.

IX. REFERENCES

[1] C. Huang et. al. Image Observation on the Modified ECB Operations in Advanced Encryption Standard. IEEE. 2011.

[2] M. Bellare, P. Rogway. Symmetric Encryption. http://cseweb.ucsd.edu/~mihir/cse207/w-se.pdf

[3] A. Boldyreva. Symmetric encryption, encryption modes and security notions. Applied Cryptography. http://www.cc.gatech.edu/~aboldyre/teaching/F11cs6260/symenc1.pdf

Appendix

NOTATIONS

The notation followed is the same as the notation followed by the authors in the paper [1].

1. *AES* is the block cipher, with input length l, used in all of the schemes discussed later.

2. *Encryption Function* (C_k) is the encryption algorithm for encrypting one block.

3. *Decryption Function* (D_k) is the decryption algorithm

4. The images can be encoded in form of bit-strings which can be divided in n equal blocks with each block having length l.

5. P_j represents one *plain text block* for j = 1 to n

6. C_j represents one *cipher text block* for j = 1 to n

7. IV has been used to denote initialization vector.

8. T_j represents the IV, which is a part of arithmetic sequence, for the block j for j= 1 to n

9. S_j represents the IV, which is a part of arithmetic series, for the block j for j= 1 to n

10. C_0 and O_0 represent the IV in cases where IV is part of a random sequence.

Appendix A: Counter Input Modification (CIM)

Proof of proposition 1.1

Consider an adversary A who is given an $C_k(LR(.,.,b))$ oracle who can query q number of (M0, M1) message pairs to the oracle, where the length of M_0 is equal to the length of M_1. The adversary has to determine whether left message or right message is being encrypted. The adversary A works as follows: A queries message pair $(0^l0^l0^l, 0^l0^{l-1}10^l)$ to the LR oracle where each message is of three block length. A gets back C'⟵ <IV||C> and parses C as C[1]||C[2]||C[3] where C[i] is of length l.

If (C[1] = C[2]) OR (C[2]=C[3]) : A returns 1 else returns 0.

It can be claimed that A has advantage which is

$Adv_{CIM}^{ind\text{-}cpa}(A) = Pr[Exp_{CIM}^{ind\text{-}cpa\text{-}1}(A) = 1] - Pr[Exp_{CIM}^{ind\text{-}cpa\text{-}0}(A) = 1] = 1\text{-}0 = 1$

A would always succeed because whenever right message (M_1) is chosen and even IV is selected the first condition would be true i.e. C[1]=C[2]. Also, whenever odd IV is selected for right message (M_1) the second condition would be true i.e. C[2]=C[3] . Therefore, A returns 1 indicating that right message is encrypted when either of these becomes true else A would return 0 indicating that left message is encrypted.

Therefore, CIM is not secure as A has advantage one with reasonable amount of resources i.e. using 2 queries, message length 3l and time to compare maximum 2l bit.

Proof of Proposition 1.2: The theorem states that the IND-CCA security implies IND-CPA security. Another interpretation would be that if the scheme is not IND-CPA secure, it can't be IND-CCA secure. Thus, CIM is not IND-CCA secure as it is not IND-CPA secure.

Appendix B: Summation Input Modification (SIM)

Proof of Proposition 2.1**:** To show the insecurity of this scheme, an adversary A can be constructed. The adversary is given an LR oracle $C_k(LR(.,.,b))$ and can query q number of (M_0, M_1) message pairs, where the length of M_0 is equal to the length of M_1, to the oracle. The adversary has to determine whether left message or right message is being encrypted. The adversary A works as follows. A queries message pair $(0^l||0^l||0^l, 0^{l-1}1||0^l||0^l)$ to the LR oracle where each message is of three block length. A gets back C'⟵ <IV||C> and parses C as C[1]||C[2]||C[3] where C[i] is of length l. If (C[1] = C[2]) A returns 1 else A repeats the same query one more time. For second query, if (C[1] = C[2]) A returns 1 else A returns 0.

It can be shown that the advantage of adversary is

$Adv_{SIM}^{ind\text{-}cpa}(A) = Pr[Exp_{SIM}^{ind\text{-}cpa\text{-}1}(A) = 1] - Pr[Exp_{SIM}^{ind\text{-}cpa\text{-}0}(A) = 1] = 1\text{-}0 = 1$.

A would always succeed because he can definitely predict whether left or right message is getting encrypted. If M_1 is chosen for encryption and the IV chosen is even, $(0^{l-1}1 \oplus IV)$ adds 1 to the IV effectively. 2 blocks being input to C_k are equal and thus, C[1] = C[2]. In case M_0 is chosen, these two cipher blocks would be different. As IV is chosen randomly from a range, the probability of IV being even is equal to the probability of IV being odd. Thus, IV should be even for at least one of the queries otherwise A can query the same message pair for some more times.

Therefore, SIM is not secure as A has advantage one with reasonable amount of resources i.e. maximum 2 queries, message length 6l and time to compare 2l bits.

Proof of Proposition 2.2: The theorem states that the IND-CCA security implies IND-CPA security. Another interpretation would be that if the scheme is not IND-CPA secure, it can't be IND-CCA secure. Thus, SIM is not IND-CCA secure as it is not IND-CPA secure.

Appendix C: Random Input Modification (RIM)

Proof of proposition 3.1

This is proved using the method of reduction. There exists an adversary B who needs to distinguish whether it is given an oracle access to a truly random function or a real function (instance of encryption function C). B will use A's ability to break the RIM scheme. B will run A as a subroutine, simulating the IND-CPA experiment for it. B will answer A's oracle queries using its own oracle. Finally, if A wins, B will win.

Adversary B has a oracle with function f(.). Adversary B selects a bit b at random. Adversary B runs adversary A as its subroutine. Whenever A makes an oracle query with message pair M_0 and M_1 , B selects a message M_b based on bit b. B performs following operation on message M_b:

B selects an Initialization Vector (IV) of length L at random. B divides M_b into n blocks each of length L bits. B performs $f(C_{i-1} \oplus P_i)$ for each block where C_0 is randomly selected Initialization Vector (IV). Resultant cipher would be

$IV||f(C_0 \oplus P_0)||f(C_1 \oplus P_1)||.....|| f(C_1 \oplus P_1)$.

A returns a bit b'. if (b'=b) return 1 else return 0.

B simulates A perfectly. Hence, whenever A wins, B wins.

Advantage: Random Input Modification scheme is similar to the scheme Cipher Block Chaining with random Initialization Vector (IV). So, its advantage is same as that of CBC with random initialization vector. Hence, its advantage is given as:

$Adv_{RIM}^{ind\text{-}cpa}(A) <= 2.\ Adv_C^{prf}(B) + \sigma^2/2^L$ where σ is number of oracle queries by A, L is length of each block. For proof refer section 4.8 of [2]

Proof of Proposition 3.2: To show that the scheme is IND-CCA insecure, an adversary A can be constructed. A is given an LR oracle $C_k(LR(.,.,b))$ and can query q number of (M_0, M_1) message pairs, where the length of M_0 is equal to the length of M_1, to the oracle. A is also given a decryption oracle $D_k(.)$ and can query it just with a constraint that the cipher text to the decryption oracle should not be an output by the LR-oracle. The adversary has to determine whether left or right message is being encrypted. The adversary A works as follows. A sends the query $(M_0, M_1) \longleftarrow (0^l, 1^l)$ to the LR-oracle and gets back $C' \longleftarrow <IV \parallel C>$. A computes $IV' \longleftarrow IV \oplus 1^l$, sends $<IV' \parallel C>$ to the $D_k(.)$ oracle and gets back M'. If $(M' = M_0)$, A returns 1 else A returns 0.

It can be shown that advantage of adversary is

$Adv_{RIM}^{ind\text{-}cca}(A) = Pr[Exp_{RIM}^{ind\text{-}cca\text{-}1}(A) = 1] - Pr[Exp_{RIM}^{ind\text{-}cca\text{-}0}(A) = 1] = 1\text{-}0 = 1$.

A would always succeed because he can definitely predict whether left or right message is getting encrypted. If M_0 is chosen for encryption, C would be $C_k(IV \oplus 0^l)$ i.e. $C_k(IV)$. In the decryption oracle, the decryption block outputs $D_k(C) = D_k(C_k(IV)) = IV$ which is xored with IV' and thus the decryption oracle returns 1^l which is not equal to M_0 leading A to return 0. If M_1 is chosen for encryption, C would be $C_k(IV \oplus 1^l) = C_k(comp(IV))$ i.e. encryption of compliment of IV. In the decryption oracle, the decryption block outputs $D_k(C) = D_k(C_k(comp(IV)) = comp(IV)$ which is xored with IV' i.e. compliment of IV resulting in 0^l. Thus, decryption oracle returns 0^l which is equal to M_0 and thus, A returns 1.

Therefore, RIM is not IND-CCA secure as A has advantage one with reasonable amount of resources i.e. 1 query to LR-oracle with message length l-bits, 1 query to decryption oracle with message length 2l-bits, and time taken to compare l bits.

Appendix D: Counter Output Modification (COM)

Proof of proposition 4.1

This is proved using the method of reduction. There exists an adversary B who needs to distinguish whether it is given an oracle access to a truly random function or a real function (instance of encryption function C). B will use A's ability to break the COM scheme. B will run A as a subroutine, simulating the IND-CPA experiment for it. B will answer A's oracle queries using its own oracle. Finally, if A wins, B will win.

Adversary B has a oracle with function f(.). Adversary B selects a bit b at random. Adversary B runs adversary A as its subroutine. Whenever A makes an oracle query with message pair M_0 and M_1, B selects a message M_b based on bit b. B performs following operation on message M_b:

B selects an Initialization Vector (IV) of length L at random. B divides M_b into n blocks each of length L bits. B performs $f(T_i) \oplus P_i$ for each block where T_0 is randomly selected Initialization Vector (IV) and T = {IV, IV+1, IV+2, . .}. Resultant cipher would be

$IV||f(T_0) \oplus P_0||f(T_1) \oplus P_1||.....|| f(T_1) \oplus P_1$.

A returns a bit b'. if (b'=b) return 1 else return 0.

B simulates A perfectly. Hence, whenever A wins, B wins.

Advantage: Advantage is given as:

Claim: $Adv_{COM}^{ind\text{-}cpa}(A) <= 2.\ Adv_C^{prf}(B)$

Proof:

$Adv_C^{prf}(B) = Pr[Exp_C^{prf}(B) = 1] - Pr[Exp_C^{prf}(B) = 1]$

$= Pr[Exp_{C\text{-}real}^{ind\text{-}cpa\text{-}cg}(A) = 1] - [Exp_{C\text{-}random}^{ind\text{-}cpa\text{-}cg}$

$(A) = 1]$

$= \{1/2 + ½.Adv_{C\text{-}real}^{ind\text{-}cpa}(A) \} - \{1/2 +$

$½.Adv_{C\text{-}random}^{ind\text{-}cpa}(A) \}$

$= ½.Adv_{C\text{-}real}^{ind\text{-}cpa}(A) - ½.Adv_{C\text{-}random}^{ind\text{-}cpa}(A)$

$= ½.Adv_{COM}^{ind\text{-}cpa}(A)$

because $Adv_{C\text{-}random}^{ind\text{-}cpa}(A) = 0$ as all values obtained after encrypting using a random function are random and independent. So, it acts as One Time Pad which does not provide any advantage to the adversary.

Proof of Proposition 4.2: To show the insecurity of scheme SE, an adversary A can be constructed. A is given an LR oracle $C_k(LR(.,.,b))$ and can query q number of (M_0, M_1) message pairs, where the length of M_0 is equal to the length of M_1, to the oracle. A is also given a decryption oracle $D_k(.)$ and can send decryption queries to it with the constraint that the cipher text in the decryption query should not be output by the LR-oracle. The adversary has to determine whether left message or right message is being encrypted. The adversary A works as follows. A sends the query $(M_0, M_1) \leftarrow (0^l, 1^l)$ to the LR-oracle and gets back $C' \leftarrow <IV \parallel C>$. A computes $C'' \leftarrow C \oplus 1^l$, sends $<IV \parallel C''>$ to the $D_k(.)$ oracle and gets back M'. If $(M' = M_0)$, A returns 1 else A returns 0.

It can be shown that, $Adv_{COM}^{ind\text{-}cca}(A) = Pr[Exp_{COM}^{ind\text{-}cca\text{-}1}(A) = 1] - Pr[Exp_{COM}^{ind\text{-}cca\text{-}0}(A) = 1] = 1\text{-}0 = 1$.

A would always succeed because he can definitely predict whether left or right message is getting encrypted. If M_1 is chosen for encryption, C would be $comp(C_k(IV))$ i.e. compliment of encryption of IV. C'' would be $comp(comp(C_k(IV)))$ i.e. $C_k(IV)$. M' would be $C_k(IV) \oplus C_k(IV) = 0^l = M_0$ and thus, A returns 0. If M_0 is chosen for encryption, C would be $C_k(IV)$ and C'' would be $comp(C_k(IV))$. M' would be $C_k(IV) \oplus comp(C_k(IV)) = 1^l = M_1$ and thus A returns 0.

Therefore, COM is not IND-CCA secure as A has advantage one with reasonable amount of resources i.e. 1 query to LR-oracle with message length l-bits, 1 query to decryption oracle with message length 2l-bits, and time taken to compare l bits.

Appendix E: Summation Output Modification

Proof of proposition 5.1

The reduction proof of proposition is similar to the proof of proposition 4.1. The only difference is instead of T being an arithmetic sequence; T in this case is arithmetic series. Therefore, T = {IV, IV+1, IV+1+2, . . }. The remaining proof remains the same. The advantage and resources of the adversary are same as those of the adversary in the proof of proposition 4.1.

Proof of Proposition 5.2: The encryption and decryption schemes are same as those in the COM encryption mode for the first block of input. The adversary A can use the same attack as shown in the proof of proposition 4.2 as it uses 1-block inputs only. Therefore, the adversary can easily determine whether left or right message is getting encrypted. The $Adv_{SOM}^{ind\text{-}cca}(A)$ is 1 and the resources are same as those of adversary in proposition 4.2.

Appendix F: Random Output Involved P_j

Proof of Proposition 6.1: This is proved using the method of reduction. There exists an adversary B who needs to distinguish whether it is given an oracle access to a truly random function or a real function (instance of encryption function C). B will use A's ability to break the ROP scheme. B will run A as a subroutine, simulating the IND-CPA experiment for it. B will answer A's oracle queries using its own oracle. Finally, if A wins, B will win.

Adversary B has a oracle with function f(.). Adversary B selects a bit b at random. Adversary B runs adversary A as its subroutine. Whenever A makes an oracle query with message pair M_0 and M_1, B selects a message M_b based on bit b. B performs following operation on message M_b:

B selects an Initialization Vector (IV) of length L at random. Let C_0 be IV selected by adversary B. B divides M_b into n blocks each of length L bits. B performs $f(C_i) \; P_i$ for each block where C_0 is randomly selected Initialization Vector (IV). Resultant cipher would be

$IV \| f(C_0) \oplus P_0 \| f(C_1) \oplus P_1 \| ..... \| f(C_L) \oplus P_L$.

A returns a bit b'. if (b'=b) return 1 else return 0.

B simulates A perfectly. Hence, whenever A wins, B wins.

Proof of Proposition 6.2: The encryption and decryption schemes are same as those in the ROP encryption mode for the first block of input. The adversary A can use the same attack as shown in the proof of proposition 4.2 as it uses 1-block inputs only. Therefore, the adversary can easily determine whether left or right message is getting encrypted. The $Adv_{ROP}^{ind\text{-}cca}(A)$ is 1 and the resources are same as those of adversary in proposition 4.2.

Appendix G: Random Output not Involved P_j

Proof of proposition 7.1

The reduction proof of proposition 7.1 is similar to the proof of proposition 6.1. The only difference is instead of giving cipher as the input to next block, the output of the current encryption function is the input to next block. The remaining proof remains the same.

Proof of Proposition 7.2: The encryption and decryption schemes are same as those in the COM encryption mode for the first block of input. The adversary A can use the same attack as shown in the proof of proposition 4.2 as it uses 1-block inputs only. Therefore, the adversary can easily determine whether left or right message is getting encrypted. The $Adv_{RNP}^{ind\text{-}cca}(A)$ is 1 and the resources are same as those of adversary in proposition 4.2.