Enhanced Authenticated Encryption Scheme

Dr. Eng. / Jamal Abelfatah Morad Azzam
Research Center , SCA
Ismailia, Egypt.
e-mail:jamalazzam@yahoo.com

Abstract− Cryptography is a vital part in information handling. In this paper we introduce a new scheme for encryption and authentication of the encrypted message. A random number is generated in each encryption process. Both of the random number and the secret key is used to generate the subkeys, a different subkey for each data block. To increase the secrecy, double permutation processes are executed on data blocks in the form of mutation and crossover. Mutation process to be performed at an arbitrary bit number, and crossover is performed at another bit number. Encryption of each data block is dependent on the previous encrypted data blocks, the secret key, and the random number. Also, one way hash function is generated to ensure authenticity of the message. The scheme proves its strength against cryptanalysis.

Keywords − encryption ; decryption ; hash function ; secret key.

I. INTRODUCTION

Cryptography refers almost exclusively to encryption, it is the process of converting ordinary information (called plaintext) into unintelligible gibberish (called ciphertext). Decryption is the reverse, in other words, moving from the unintelligible ciphertext back to plaintext. A cipher in cryptosystems is a pair of algorithms that create the encryption and the reversing decryption [1][2]. The detailed operation of a cipher is controlled by the algorithm and the key. The objectives are the following items [3]-[12]:

- Privacy or confidentiality.
- Data integrity.
- Authentication.
- Non-repudiation.

There are two types of cryptosystems, one-key (or symmetric key), and two-key (asymmetric key) ciphers. In symmetric key ciphers, the encryption of a plaintext and the decryption of the corresponding ciphertext are performed using the same key. Until 1976 when Diffie and Hellman introduced public-key or two-key cryptography all ciphers were one-key systems [14]. Therefore one-key ciphers are also called conventional cryptosystems.

Conventional cryptosystems are widely used throughout the world today, and new systems are published frequently. There are two types of conventional cryptosystems: stream ciphers and block ciphers.

In stream ciphers, a long sequence of bits is generated from a short string of key bits, and it is then added bitwise modulo 2 to the plaintext to produce the ciphertext. In block ciphers the plaintext is divided into blocks of a fixed length, then they are encrypted into blocks of ciphertexts using the same key. Block ciphers can be divided into three groups: Substitution ciphers, Transposition ciphers and, Product ciphers.

Since its introduction in 1977, the Data Encryption Standard (DES) has become the most widely applied private key block cipher [15]. Recently, a hardware design to effectively break DES using exhaustive search was outlined by Wiener [16]. AES by its turn is subjected to different cryptanalysis that presumes its ability to break AES. [17]-[19]. So, there are a need to secure, and flexible block ciphers with immune encryption.

In this paper, a novel randomized scheme is proposed. It uses beside the secret key a random number. Both of them are implemented to generate different subkeys for different blocks of message. The secret key can be of any chosen size, provided key size modulo block size is zero. In each encryption process a new random number is generated, consequently, increases the resistance to cryptanalysis whatever it is based on differential [16], or linear [21]. A hash function is also used to ensure data authenticity. This scheme has significant strict avalanche criterion (SAC), and keeps cryptographic static and dynamic prosperities of the substitution permutation networks (SPNs).

The rest of this paper is organized as follows. Section II describes how to generate the subkeys, and the message authentication code (MAC), and the proposed algorithm. Section III Addresses the encryption process of a random number, and the MAC number, the mutation and crossover processes for message blocks, and encryption / decryption of data blocks. Section IV provides results of strict avalanche effect compared with other algorithms. Section V addresses an evaluation of the algorithm performance compared with other algorithms. Analysis of the algorithm is also introduced. Section VII summarizes the conclusions.

II. THE PROPOSED ALGORITH

The algorithm is a novel authenticated scheme that provides data confidentiality. By confidentiality we mean data encryption and data authentication / integrity. A secret key of 90 bits is used along with a randomly chosen number to encrypt the data. Range of the random number is 0 to 2^{90}-1. The random number is used with the secret key to generate subkeys for data blocks. The idea of generating these subkeys is as follows:

A group of nine coaxial disks are used to generate subkeys. These disks are represented in a physical form (for explanation purpose) in Fig. 1. Each disk is divided into 2^{10} slots. The slots are numbered 1, 2, 3, 1024; each slot carries a value 1, 2, to 1024. Initially, each slot carries a value equal its number e.g., slot number1 carries value 1, slot number 2 carries value 2 and so on., but after turning the disk with respect to a specific pointer P each slot carries a value that is different than its number e.g., If the disk is turned anticlockwise by 3 slots, then slot number 1 carries value 4, slot number 2 carries value 5 and so on, Fig. 2 and Fig. 3.

The secret key is composed of 9 blocks: S_1, to S_9, Each key block is 10 bit long. The value of any key block can be chosen as a pointer value P e.g., suppose the secret key is S= 1000 1022 0300 0400 0500 0250 1023 0450 0333. P value can be S_9 i.e., 333. Each data block has a corresponding disk, that is data block number 1 corresponds to disk number 1, data block number 21 corresponds to disk number 21 mode 9 = 3. Each disk turns by a specific value that is different from other disks, and the slots values corresponding to P on each disk (after turning) are used to generate the subkeys as shown hereinafter.

The proposed algorithm can be divided into the following steps:

A. Dividing the Message into Blocks

The message is divided into blocks of 10 bit each. B_1, B_2, B_n., number of blocks in the message is n.

Also, the secret key of 90 bits has 9 key block,
S_1, S_2, …………………….,S_9.

Each data block has also a corresponding key block.

B. Generating a Random Number

In every encryption process choose a random positive integer number R of any value less than (2^{90}-1).

C. Compute Hash Function

Groups of data blocks are formed from nine block, i.e., GB_1, GB2,……, GB_m. Then, XOR ing GB_1 (first group of blocks) with the second GB_2, and the result is XOR ed with the third group of blocks GB_3, and so on till the message is finished

$$MAC = GB_1 \oplus GB_2 \oplus GB_3 \ldots\ldots \oplus GB_{n.} \qquad (1)$$

GB_1:

B_1	B_2	B_3	B4	B_5	B_6	B_7	B_8	B_9

GB_2:

B_{10}	B_{11}	B_{12}	B_{13}	B_{14}	B_{15}	B_{16}	B_{17}	B_{18}

GB_3:

B_{19}	B_{20}	B_{22}	B_{22}	B_{23}	B_{24}	B_{25}	B_{26}	B_{27}

D. Generation of Subkeys

Generation of subkeys is performed into two steps, computing the turning value of the disk that corresponds to the data block number, and then using the turning value with the pointer value to compute the subkey for this data block as follows:

a. Each data block has a non repetitive subkey which is different from all other subkeys. Random number R is used with the secret key to generate the subkeys for each data block. Number of subkeys to be generated equals number of data blocks n, i.e., for data block number (I) a corresponding disk J. Disk J turns by value TV_J :

$$TV_J = (I + R)^{S_J} \bmod (2^{10} -1) \qquad (2)$$

Where: J is disk number

I is data block number.

S_J is secret key block number.

$$\left.\begin{array}{ll} J = I & \text{for } I \leq 9 \\ J = I \text{ mode } 9 & \text{for } I > 9. \end{array}\right\} \qquad (3)$$

These two equations (2) and (3) produce different turning values for each disk in every encryption process. As a consequent, subkeys are not repeated, as an example: For the secret key is : S = 1000 1022 0300 0400 0500 0250 1023 0450 0333. Let the random number is 123456789, and let the pointer value be the value of S_9, i.e., P = 0333.

Disk number 1 will turn anticlockwise by value TV_1:

$TV_1 = (1+123456789)^{1000}$ mode $(2^{10}-1) = 397$.

Disk number 2 will turn anticlockwise by value

$TV_2 = (2+123456789)^{1022}$ mode $(2^{10}-1) = 16$.

And so on.

b. The subkey for data block number I is computed according to the turned value of its corresponding disk, and is given by:

$$SK_I = (P + TV_J) \text{ mode } (2^{10}-1) \tag{4}$$

For block number 1, its subkey is:

$SK_I = 397+ 0333= 730$ = 10110 11010.

Similarly, subkey for block number 2

$SK_2 = 16 + 0333 = 0349$ = 01010 11101

III. ENCRYPTION PROCESS

Given: plaintext B, secret key S, random number R, MAC, and computed subkeys SK_I.

We start by computing a function of R e.g., R~ using the secret key and the pointer value. Then we encrypt R~ by mutation and crossover with a fixed number X1. The same procedure is repeated with the MAC number except that we use R in computing the function of MAC e.g., MAC~.

Let the secret key S = S1 S2 S3 …..S9, and the plaintext divided into blocks e.g.,

B_1 = 1101101101, B_2=1011100101,
B_3 = 0101010111, $B_{4,}$ …………,B_n.

A. Encrypt the Random Number R

In this regard, we use two fixed large numbers known in sending and receiving algorithms $X_{1,}$ X_2. Also, we do not use R or MAC or S themselves, but functions of them.

$$ER = R^{\sim} \text{ (MuCr) } X_1 \tag{5}$$

Where: MuCr stands for (Mutated and Crossed over) as explained for data blocks hereinafter.

$R^{\sim} = (R \pm X_2) \oplus SR$.
$SR_i = p^{10}$ mode $(S' \oplus i)$. i= 1, 2, ……………………9.
$S' = S_1 \oplus S_2 \oplus S_3 \oplus$ ….. ,S_9
$X_{1,}$ X_2 are two large numbers.

B. Encrypt Message Authentication Code MAC

$$EMAC = MAC^{\sim} \text{ (MuCr) } X1 \tag{6}$$

$MAC^{\sim} = (MAC \pm X_2) \oplus SM$.
$SM_i = p^R$ mode $(S' \oplus i)$. i= 1, 2, ……………………9.

These two encrypted values ER and EMAC form the first 180 bits of the encrypted message. See Fig. 4.

C. Mutation:

For each block of data apply mutation process at a specific arbitrary bit number (say bit number 4):

B_1= 1101 ‖ 101101, B_2=1011 ‖ 100101 and

B_3 = 0101 ‖ 010111 would be :

B_1'= 1011011101, B_2' = 1001011011, B_3' = 0101110101.

D. Crossover:

At another arbitrary bit number (say bit number 6), B_1' will be crossed over with B_2', B_2' with B3', and so on till B_n' with B_1'as:

$B_1' = 101101 \| 1101$, $B_2' = 100101 \| \underline{1011}$,

$B_3' = 010111 \| | \underline{\underline{0101}}$

$B_1'' = 101101\underline{\underline{1011}}$, $B_2'' = 100101\underline{\underline{0101}}$

E. Blocks Encryption

Fig. 5 explains data blocks encryption. Each Block after mutation and crossover is encrypted by its subkey, so the encrypted block EB_I is computed as:

$$EB_I = EB_{(I-1)} \oplus B_I'' \oplus SK_I \tag{7}$$

ISBN: 978-1-61208-427-5

e.g., $EB_1 = B_1'' \oplus SK_1$

$= 10110\ 11011 \oplus 01110\ 01100 = 11000\ 10111.$

Also, $EB_2 = EB_1 \oplus B_2'' \oplus SK_{2.}$

$= 11000\ 10111 \oplus 10010\ 10101 \oplus 01010\ 11101$

$= 00000\ 11111$ ………..etc.

F. Summery

The algorithm block diagram is shown in Fig. 6.

A message of m bit, and a 90 bit security key $S = S_1\ S_2\ S_3\ S_4\ S_5\ S_6\ S_7\ S_8\ S_9$ can be encrypted as follows:

a- Choose a positive integer random number R;

$0 < R < 2^{90}$.

b- Divide the message into 10 bit blocks, each nine blocks form a group of 90 bit. Get the hash message authenticated code (MAC) by XOR ing groups as explained in (1).

c- Compute $R^{\sim}$ as function of R, and $MAC^{\sim}$ as a function of MAC. Then, encrypt $R^{\sim}$ and $MAC^{\sim}$ by XOR ing them with the X1, X 2 as explained in (5) and (6).

d- Generate a turning value for each disk as shown in (2) and (3).

e- Compute subkeys for each data block by (4) using the turning value of the corresponding desk.

f- Apply mutation and crossover operations on each block at an arbitrary bit number for each operation.

g- Encrypt each data block B_I of the information message using previous blocks and its subkey as in (7).

G. Decryption

The algorithm reads ER first to decrypt the random number R; $R = (R^{\sim} \pm X_2) \oplus S^{\sim}$. Getting R the algorithm decrypts EMAC to check the authenticity of the message (after being decrypted). Then the procedure goes on, in reverse order.

IV. EXAMPLE

Strict avalanche criteria(SAC) is a desirable property of cryptographic algorithms. That is if an input plaintext is changed slightly (e.g., flipping a single bit) the enciphered output changes significantly (may be more than half the output bits flip).

Example: The input plaintext is " DISASTER". Flipping one bit from the plaintext, we get "DISCSTER", (one flipping A (01000001) to C (01000011)). The Key used is “SRIRAMSR".

DISASTER encrypted message is 00111,11011 10001,10100 10101,01000 10100,10011 11011,01001 01001,11011 11011,00111

DISCSTER encrypted message is 01010,00110 00100,01011 01100,11100 11110,10000 00100,10111 01101,01110 11111,10010

Number of flipped bits in the encrypted message is 42 bit (out of 65 bits of the original message).

Avalanche effect = 42 * 100 / 65 = 64.6%.

The same example was carried out by other algorithms [24], and the results are shown in Table I.

TABLE I COMPARISON OF AVALANCH EFFECT

Encryption Technique	No. of flipped bits	%
Playfair Cipher	4	6.25
Vigenere Cipher	2	3.13
Caesar Cipher	1	1.56
DES	35	54.68
Blowfish	19	28.71
The Proposed Technique	42	64.6

V. ALGORITHM EVALUATION

A comparative evaluation of the algorithm is presented in this section. Two main properties are discussed here : performance, and randomness.

A. Performance

The algorithm runs on a 3.2 GH PC with different lengths messages, the concluded speed is 5.19 cycle per byte (587 MiB/s) for the encryption process, which is very fast compared to different algorithms shown in the Table II benchmark, [33].

TABLE II ALGORITHM SPEED COMPARISON

Algorithm	MiB/Second	Cycles Per Byte
AES/GCM (2K tables)	102	17.2
AES/GCM (64K tables)	108	16.1
AES/CCM	61	28.6
AES/EAX	61	28.8
CRC32	253	6.9
Adler32	920	1.9
MD5	255	6.8
DES/CTR	32	54.7
DES-XEX3/CTR	29	60.6
DES-EDE3/CTR	13	134.5
PROPOSED ALGORITHM	587	5.19

Number of overhead bits is fixed and irrelevant to the message length (double the key size) i.e., 180 bits only for MAC and random number).

B. Analysis

Although the algorithm does not depend on substitution permutation networks (SPNs), it keeps its cryptographic static and dynamic prosperities. The algorithm strict avalanche effect causes 65 % of bits in average to be flipped in the enciphered text for one bit flipped in plaintext as shown in the Table 1. Strict avalanche criterion is a measure of a cipher's randomness [23]. This ensures its resistance to statistical, clustering, linear, and differential cryptanalysis.

The algorithm provides randomized encryption, so that when encrypting the same message several times, it will produce different ciphertexts each time. Key size can be of any chosen number. Consequently, block size can be larger, e.g., key size of 1024 bit with block 128 or 256 or 512. The larger the key size the larger range for the random number. $0 < R < 2^{key\text{-}size.}$.

The algorithm has two different arbitrary bit numbers (from 1 to block_size -1) one for mutation and the other for crossover process.

Also, number of mutation then crossover rounds can be increased to any chosen number.

The number of brute force trials of an n bit message is: {TV = 9 (I) × 2^{90} (R) × 2^{90} (S) × 2 (Turning direction)} ×{SK= 2^{10} (P) }×{10^{10} (10 possible bit number for mutation, and 10 bit number for crossover for each one)}. This equals 2.82×10^{68} or 8.95×10^{51} years (assuming 10^{10} decryption process per second).

In a known message attack, if the attacker knows both plain and ciphered messages completely, he cannot get MAC or R. Both are not send, but functions of them. To get the function of R, i.e., $R^{\sim}$, the attacker has to make reverse mutation and crossover in 10^{10} operations, (nine possible mutation bit numbers, and for each one nine possible crossover bit number). While guessing R from $R^{\sim}$ and X_2, there are 2^{90} possible changes in $R^{\sim}$. So, the attacker needs $(10^{10})^{90}$ i.e., 3.1×10^{883} years (assuming 10^{10} operation per second). A similar number of operations to be executed to get MAC from MAC~.

The algorithm time and space complexity is O(n), i.e., the required time and space for encryption/decryption increase linearly with message length. However attacker algorithm is NP complete.

VI. CONCLUSIONS

In this paper, we introduced a novel immune scheme for block cipher randomized encryption. The scheme shows high strength of confidentiality even for known message attack. A hash message authentication code is also used to ensure message authenticity. In the proposed scheme, both the key length and the block size can be of any chosen size, provided key length modulo block size is zero.

The algorithm has a high degree of randomness; its strict avalanche effect is very significant and surpasses a lot of the famous algorithms. This proves its immunity to cryptanalysis. Knowing the algorithm, the plaintext and the ciphertext does not reveal useful information for the attacker to crack the key or the random number, since in every run a new random number and new subkeys are generated, consequently different ciphertexts for the same plaintext.

The algorithm has a strong strict avalanche criterion (SAC) (65% in average). Also, it keeps the cryptographic static properties of substitution permutation networks (SPNs) of completeness, nonlinearity. In the same time the algorithm provides perfect security defined by Shannon [22], and every bit in the information message is encrypted using a different subkey.

Implementing different random number of large range ($2^{\mathbf{key\_size}}$) in every encryption process makes it impossible to be cracked. For the chosen length (90 bit), brute force attack needs 3. 7 X 10^{16} Years to crack that key, (assuming 10^{10} check per second).

The random number and MAC number are safely distributed between sender and receiver in a new procedure. The hash function used ensures the authentication and integrity of the message. Any bit change will be detected.

As an additional feature of the scheme some of its parameters are selective and not fixed: Key size – Block size – Bit number to perform mutation – Bit number to perform crossover – Disk number to be used as a pointer.

A good feature of the algorithm is that, its time and space complexity is O (n). So, the required memory resources and computation time are not increased in a large scale with the increase of message length. In the same time the attacking algorithm is NP complete.

REFRENCES

[1] D. Kahn, " The code breakers", ISBN 0-684-83130-9 New York, Macmillan ,1967,

[2] "Cryptology (definition)", http:// www.marriam-webster.com/dictionary/cryptology, Merriam-Webster's Collegiate Dictionary (11th edition Ed.).Merriam- dictionary / cryptology, retrieved 2015-03-25.

[3] H. Beker and F. Piper, "Cipher Systems: The Protection of Communications", John Wiley & Sons, New York, 1982.

[4] D. W. Davies and W.L. Price, " Security for Computer Networks", John Wiley & Sons, New York, 2nd edition, 1989.

[5] D.E. Denning, "Cryptography and Data Security", Addison-Wesley, Reading, Massachusetts, Reprinted with corrections1983.

[6] W. Diffie and M. Hellman, “New directions in cryptography”, IEEE Transactions on Information Theory, 22,644–654,1976.

[7] W. Diffie, “The first ten years of public key cryptology”, G.J. Simmons, editor, Contemporary Cryptology: The Science of Information Integrity, 135–175, IEEE press,1992.

[8] W. Diffie and M. E. Hellman, “Multiuser cryptographic techniques”, Proceedings of AFIPS National Computer Conference, 109–112, 1976.

[9] D. Kahn, "The Code breakers", Macmillan Publishing Company, New York, 1967.

[10] A.G. Konheim, "Cryptography, A Primer", John Wiley & Sons, New York, 1981.

[11] G. J. Simmons, editor, “Contemporary cryptology: An

introduction”, Contemporary Cryptology: The Science of Information Integrity, 1–39, IEEE Press, 1992.

[12] R. Merkle, " Secrecy, Authentication, and Public Key Systems", UMI Research Press, Ann Arbor, Michigan, 1979.

[13] Boritz, J. "IS Practitioners' Views on Core Concepts of Information Integrity". International Journal of Accounting, Information Systems. Elsevier, August 2011.

[14] W. Diffie and M. Hellman, “New directions in cryptography”. IEEE Trans. On Information Theory, IT-22(6), 1976.

[15] National_ Bureau_ of_ Standards,“ Data Encryption Standard (DES),” federal Information Processing Standard Publication FIPS PUB 46-3,U.S. dept. of commerce/national institute of standards and technology, 1977.

[16] I. Ben-Aroya and E. Biham, " Differential cryptanalysis of Lucifer". In D.R. Stinson, editor, Advances in Cryptology: CRYPTO'93, LNCS 773, 1993.

[17] D. Warren,"1. AES seems weak. 2. Linear time securecryptography",http://www.researchgate.net/publication/220335792_1._AES_seems_weak._2._Linear_time_secure_cryptography, IACR Cryptology ePrint Archive 01/ 2007, retrieved 2015-03-01.

[18] Bruce Schneier, "Another New AESAttack" https://www.schneier.com/blog/archives/2009/07/another_new_aes.html, retrieved 2015-03-03.

[19] E. Biham and A. Shamir, " Differential Cryptanalysis of DES-like Cryptosystems", Journal of Cryptology, vol. 4, no. 1, pp 3-72,1991.

[20] Niels Ferguson e.al. "Improved Crypt analysis of Rijndael " https:// www.schneier.com/ paper-rijndael.pdf Counterpane Internet Security, Inc., 3031 Tisch Way Suite 100PE, San Jose,CA 95128, retrieved 2015-03-01.

[21] L. Brown, J. Pieprzyk, and J. Seberry, " LOKI - a cryptographic primitive for authentication and secrecy applications". In J. Seberry and J. Pieprzyk, editors, Advances in Cryptology: AusCrypt'90, LNCS 453,. Springer Verlag, 1990.

[22] C. E.Shannon, " Communication theory of secrecy systems", Bell System Technical J. 28, 656-715,1949.

[23] Heys and Tavers , "On Design of Secure Block Ciphers", Queen's 17 th symposium on Communications, Kingstone, Ontario, Canada, May 1994.

[24] Sriram Ramanujam and Marimuthu Karuppiah, "Designing an algorithm with high avalanche effect", IJCSNS International Journal of Computer Science and Network Security, VOL. 11 NO.1, January 2011.

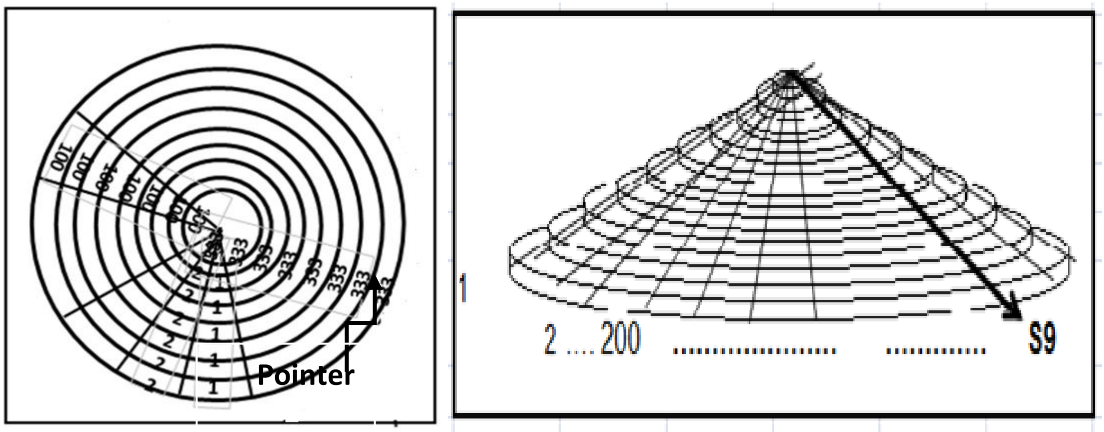

Figure 1 Physical Representation of Coaxial Disks

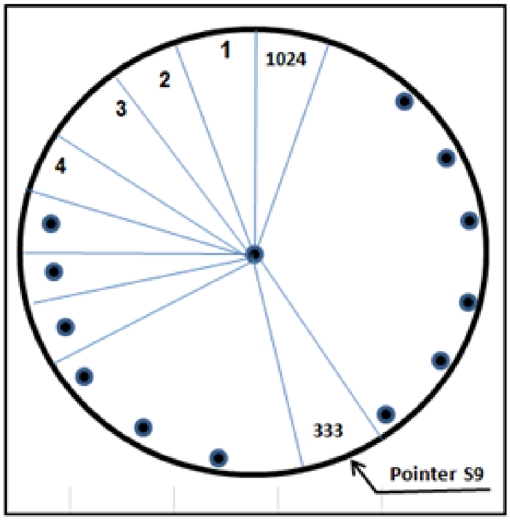

Figure 2 One Disk in Initial Position

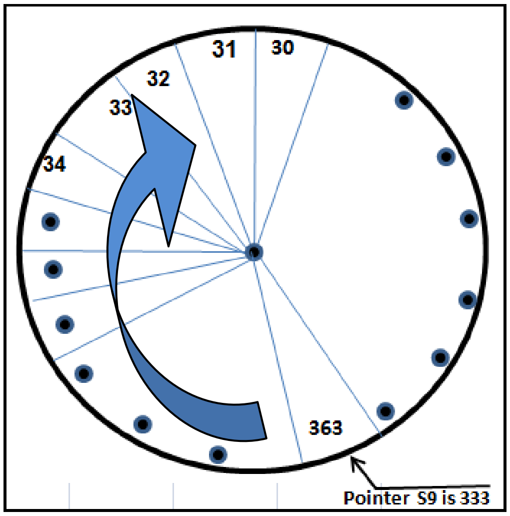

Figure 3 Same Disk after Turned 30 Slots

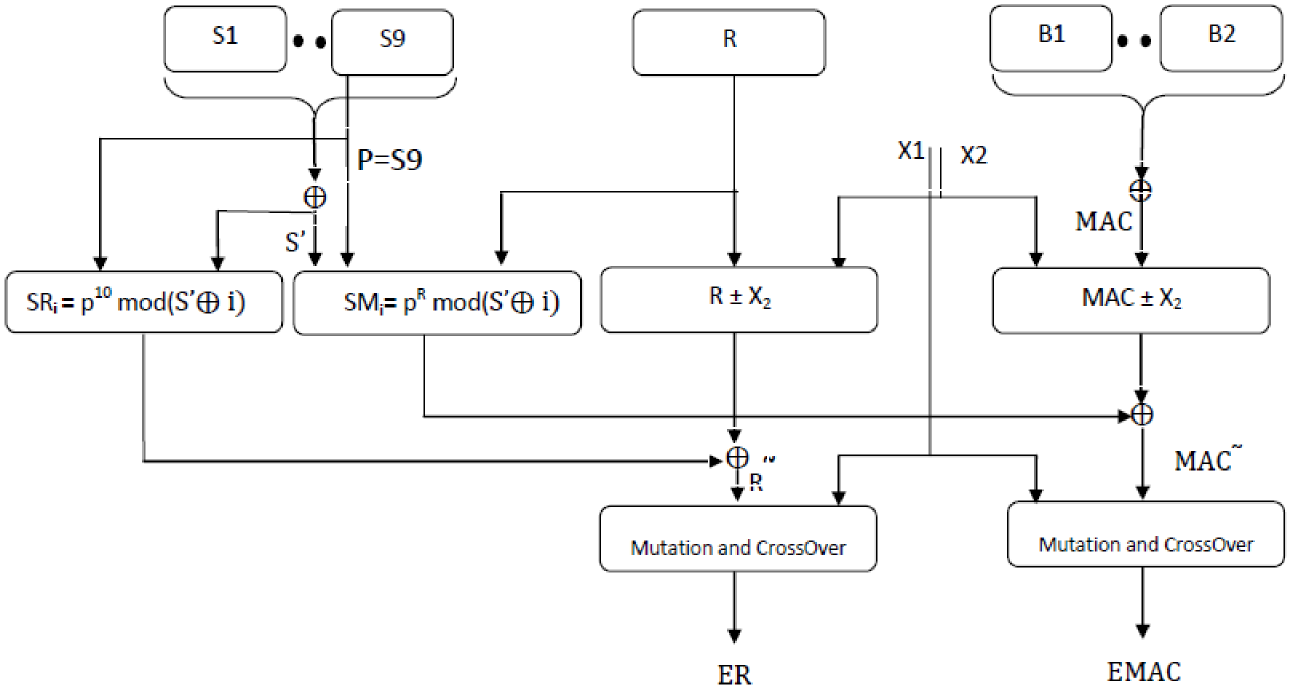

Figure 4 Encryption of Random and MAC Number.

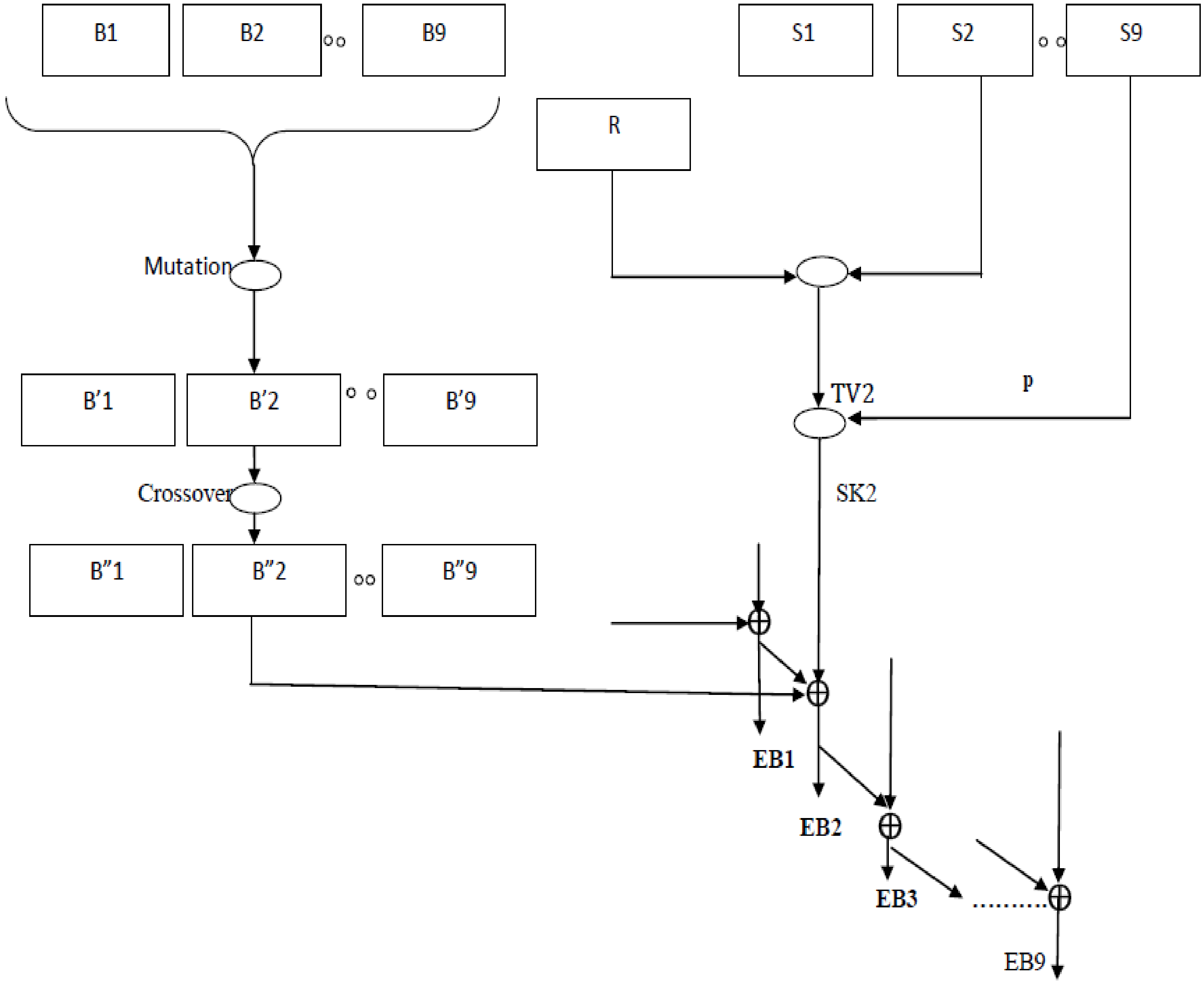

Figure 5 Encryption of Data Blocks

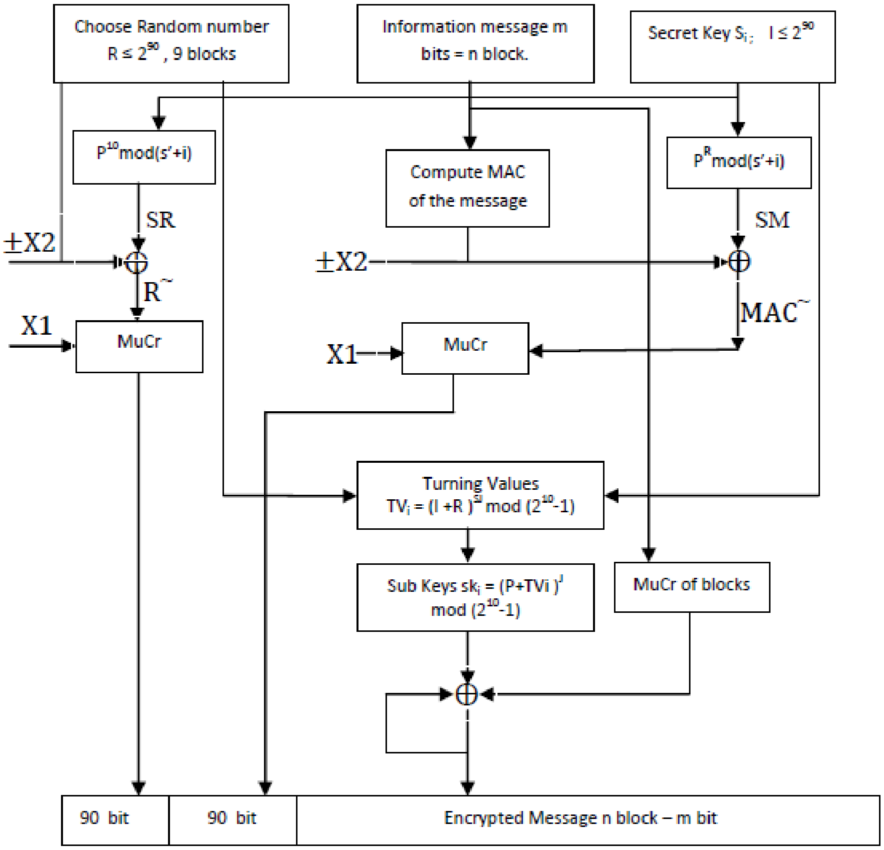

Figure 6 Encryption Scheme

ISBN: 978-1-61208-427-5