Improved SAFER Plus Algorithm Using the Odd-Squared New Mersenne Number Transform

Abdulmutalib A-Wahab Hussein, Lujain Sabah Abdulla and Mounir Taha Hamood
Department of Electrical Engineering, College of Engineering, Tikrit University, 34001 Tikrit, Iraq
abdulmuttalib.a.hussein@tu.edu.iq, lujainsabah@tu.edu.iq, m.t.hamood@tu.edu.iq

Keywords: Encryption, Symmetric, Asymmetric, SAFER+, O^2NMNT.

Abstract: Encryption secures data by transforming it into unreadable information (ciphertext) that can only be read by authorized receivers, who then convert the unclear data back into the original plaintext. Encryption can be classified into two types: symmetric and asymmetric. Symmetric encryption utilizes the same key for each of the sender and the receiver whereas two different keys are used in asymmetrical encryption. One of the encryption techniques is the SAFER+ algorithm. In this paper, a generalized type of the new Mersenne number transform (NMNT) called the odd-squared NMNT (O2NMNT) is used to generate a new matrix in the proposed SAFER+ algorithm to replace the basic matrix used in the standard SAFER+ algorithm. The results revel that in the proposed SAFER+ algorithm the number of attempts required to find the key used in the encryption process in extremely higher than the attempts needed in the standard SAFER+ algorithm this gives the suggested algorithm a higher encryption strength than the conventional algorithm.

1 INTRODUCTION

Data security has been increasingly important in recent years, particularly with current exchange networks, which include flaws that can be exploited to catastrophic effect [1]. Encryption is a means of preserving data so that it remains unmodified and secure at some point throughout the transfer from the sender to the intended receiver [2]. Cryptography is a rule or procedure that protects private or sensitive information from the public or other members. It is critical for ensuring data integrity, confidentiality, and user privacy [3]. An encryption algorithm can convert crucial data from plaintext to ciphertext. Decryption is the process of transforming ciphertext to plaintext with a decryption algorithm and a key [4]. The key is used for the encryption and decryption of certain data, controlling the transition between plaintext and ciphertext [5]. Data encryption /decryption process is shown in Figure 1.

There are two types of keys used in encryption /decryption process: symmetric and asymmetric [6]. The same key for encryption and decryption is utilized in symmetric key encryption technology, as illustrated in Figure 2 [7], which is the most efficient kind of encryption that presents one private key to cipher and decode data [8]. The key benefits of symmetric encryption systems are their reliability and speed [9].

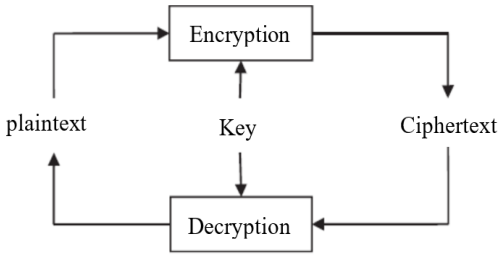

Figure 1: Encryption and decryption process.

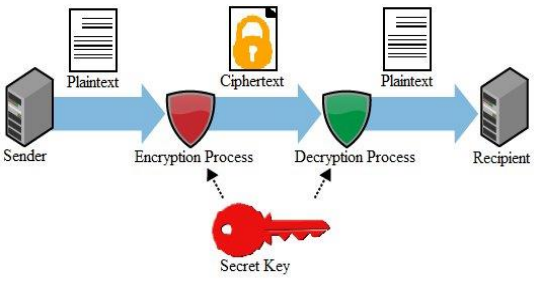

Figure 2: Symmetric key encryption.

Asymmetric encryption, often known as public-key encryption, is a relatively new technology when compared to symmetric encryption [10]. Asymmetric cryptography uses a pair of keys to encrypt basic textual content: a public key and a private key [11] as shown in Figure 3 [7]. Data is encrypted with a key that is publicly available and decrypted with an exclusive key [9]. Asymmetric encryption algorithms require not less than a 3000-bit key to attain the same level of privacy as 128-bit symmetric algorithms [12].

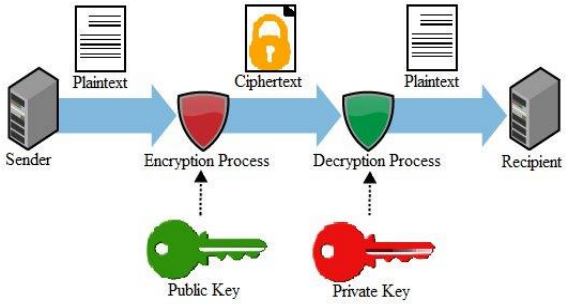

Figure 3: Asymmetric key encryption.

There are various security techniques for wireless network devices. Some popular algorithms are AES, DES, Triple DES, IDEA, SAFER+, and ECDH [13]. The SAFER+, which stands for (Secure and Fast Encryption Routine), algorithm is based on the current SAFER family of ciphers, which includes the SAFER K-64, SAFER K-128, SAFER SK-128 bit [14]. SAFER+ is a symmetric key standard in which the sender and receiver share the same key. Although SAFER+ is commonly used algorithm, it appears to have certain weaknesses [15] so that many authors proposed various improved algorithms to overcome these weaknesses.

A modified SAFER+ algorithm is proposed in [16] that uses the Fast Walsh Hadamard (FWH) transform and includes a rotation block for each round. The results reveal that the proposed SAFER+ approach has higher security than the previous algorithms. The effectiveness of the algorithm in [17] is determined by analyzing parameters such as encryption time, encryption frequency, data throughput, and security level. The improved SAFER+ method outperformed current algorithms when used in Bluetooth devices. In [18] the authors have made significant changes that has eliminated all the problems of the existing SAFER+ algorithm such as pseudo key collisions and differential cryptanalysis. The suggested algorithm accepts 16 bytes of plaintext and 16 bytes of key as input, then converts them to 20-bytes by generating the four-byte pseudo random number generator (PRNG). Using MATLAB to implement the SAFER+ standard in [15], the authors provide a promising possibility for the software implementation of encryption standards at a low cost and high performance. In [19], A MATLAB software implementation of the symmetric key block standard is used to encrypt and decrypt text, data, and images using the SAFER+ technique with a key of length 128. The outcomes provide good encryption/decryption performance through a series of novel approaches that facilitate the realization of non-linear functions employed by the SAFER+ algorithm.

The objective of this paper is to improve the SAFER+ algorithm via generating a new matrix using the O2NMNT and replacing it with the matrix used in the standard algorithm.

2 THE ODD-SQUARED NEW MERSENNE NUMBER TRANSFORM

Generalized new Mersenne number transforms (GNMNTs) are being established to be substantial number theoretic transforms (NTTs) capable of reliably calculating correlations and convolutions [20]. Two new transforms, named the odd NMNT (ONMNT) and the odd-squared NMNT (O^2NMNT) are defined from the generalized new Mersenne number transform [21]. It should be stated that there is a special interest in using the O^2NMNT for encryption purposes. This method produces a matrix without zeros or ones, making it suitable for diffusion analysis. The O^2NMNT of an integer sequence $x(n)$ for transform length $(N = 2^m, 1 < m < p - 2)$ is defined as in the following equation:

$$X_{o^2}(k) = \left[\sum_{n=0}^{N-1} x(n)\beta\left(\frac{(2n+1)(2k+1)}{4}\right)\right] mod\ M_p$$
$$0 \le k \le N-1, \quad (1)$$

for $n \ge 0,\ k \le N-1$ be the elements of the O^2NMNT matrix, let:

$$M_{o^2}(nk) = \beta\left(\frac{(2n+1)(2k+1)}{4}\right). \quad (2)$$

According to (1), M_{o^2} can be expressed as:

$$M_{o^2} = \begin{matrix} \beta\left(\frac{1}{4}\right) & \beta\left(\frac{3}{4}\right) & \cdots & \beta\left(\frac{2N-1}{4}\right) \\ \beta\left(\frac{3}{4}\right) & \beta\left(\frac{9}{4}\right) & \cdots & \beta\left(\frac{3(2N-1)}{4}\right) \\ \beta\left(\frac{5}{4}\right) & \beta\left(\frac{15}{4}\right) & \cdots & \beta\left(\frac{5(2N-1)}{4}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \beta\left(\frac{2N-1}{4}\right) & \beta\left(\frac{3(2N-1)}{4}\right) & \cdots & \beta\left(\frac{(2N-1)(2N-1)}{4}\right) \end{matrix}, \quad (3)$$

where $mod\ M_p$ represents modulo M_p, $M_p = 2^p - 1$ is a Mersenne prime for $p = 2, 3, 5, 7, 13, 17, 19, \ldots \ldots$ and β is the transform kernel defined as [22]:

$$\beta(nk) = \beta_1(nk) + \beta_2(nk), \quad (4)$$

$$\beta_1(nk) = Re((\alpha_1 + j\alpha_2)^d)^{nk}\ mod\ Mp, \quad (5)$$

$$\beta_2(nk) = Im((\alpha_1 + j\alpha_2)^d)^{nk}\ mod\ Mp. \quad (6)$$

Where α_1, α_2, q and d are defined as:

$$\alpha_1 = \pm(2^q)\ mod\ \ Mp\ , \quad (7)$$

$$\alpha_2 = \pm(-3^q)\ mod\ \ Mp\ , \quad (8)$$

$$q = 2^{p-2}, \quad (9)$$

$$d = \frac{2^{p+1}}{N}. \quad (10)$$

The detailed descriptions and specifications of O^2NMNT are available in [22], which we did not want to repeat here.

3 THE STANDARD SAFER+ ALGORITHM

SAFER+ is a 128-bit block cryptography algorithm with three alternative key lengths: 128-bit, 192-bit, or 256-bit, with encryption/decryption rounds of 8, 12, and 16 respectively. The procedure starts with the key scheduling, which creates a set of 2R keys using the initial confidential key (K_1) shared by both of the sending and receiving sides. The 2R+1 keys set is used for cryptography rounds as two keys every round, starting with K_1 to K_{2R} and finishing with K_{2R+1} for the outcome of the encryption level [16]. The decryption process on the other side follows the opposite sequence, beginning with K_{2R+1} for the input decryption level and then two keys each round, as illustrated in Figure 4 [13].

4 KEY SCHEDULES OF SAFER+ STANDARD

The strength of a symmetrical key encryption algorithm is achieved by the structure of encryption /decryption, in addition to the complexity of producing the group of subkeys ($K_2 – K_{17}$) using key scheduling [20]. Figure 5 shows that various mathematical procedures are used to generate a total of seventeen 128-bit (16 bytes) keys from the symmetric key (K_1) while keeping in mind the randomness of the keys group [13]. The sixteen subkeys ($K_2 – K_{17}$) are generated in the following way. The user secret key serves as the first subkey (K_1) and is also stored into the first 16-byte locations of a 17-byte key register ($KB_1–KB_{16}$). The most recent byte position in the register (KB_{17}) will be filled with the bit-by-bit modulo 2 -sum of the 16 bytes of the user-selected key. Each byte in the key register is then rotated three-bit positions to the left. The second subkey (K_2) is then calculated as modulo 256-sum of the 16-byte bias word (B_2) with the bytes number $KB_2 – KB_{17}$ respectively. Each byte in the key register is then rotated leftwards by three-bit places. K_3 is obtained as the sum modulo 256 of the 16-byte bias word (B_3) with the bytes number $KB_3 – KB_{17}$, and KB_1 respectively.

The SAFER+ standard algorithm uses the B matrix to randomly generate the round subkeys [20], Table 1 shows the standard matrix B.

The SAFER+ round, which consists of two transformation layers, one nonlinear layer, and byte multiplication with matrix M is shown in Figure 6.

Linear layers perform byte-to-byte addition modulo 256 (add) and bit-by-bit addition (XOR) on the layer 128-bit input data block using one of the round keys. A byte transformation on exponential and logarithmic function base (45) modulo 257 is employed with the non-linear layer, defined for the byte B as [13]:

$$F(B) = \begin{cases} 45^B\ mod\ 257 & if\ B \neq 128 \\ 0 & if\ B = 128 \end{cases}, \quad (11)$$

$$G(y) = \begin{cases} log_{45}(y) & if\ y \neq 0 \\ 128 & if\ \ y = 0 \end{cases}. \quad (12)$$

The final transformation stage in each round is based on the predetermined invertible (16×16) matrix M [14]. The decryption procedure uses the inverse of the matrix M. The matrix M and its inverse M^{-1} are defined in (13) and (14) [20].

The decryption procedure starts with the input transformation layer operating on the ciphertext and the key K_{2R+1}, using subtraction (SUB) as the inverse operation of add and XOR in the output encryption layer. Two keys from the remaining 2R keys are utilized in descending order for rounds decryption. Every step of the decryption procedure involves an inverse operation of the one used during encryption process.

$$M = \begin{bmatrix} 2 & 2 & 1 & 1 & 16 & 8 & 2 & 1 & 4 & 2 & 4 & 2 & 1 & 1 & 4 & 4 \\ 1 & 1 & 1 & 1 & 8 & 4 & 2 & 1 & 2 & 1 & 4 & 2 & 1 & 1 & 2 & 2 \\ 1 & 1 & 4 & 4 & 2 & 1 & 4 & 2 & 4 & 2 & 16 & 8 & 2 & 2 & 1 & 1 \\ 1 & 1 & 2 & 2 & 2 & 1 & 2 & 1 & 4 & 2 & 8 & 4 & 1 & 1 & 1 & 1 \\ 4 & 4 & 2 & 1 & 4 & 2 & 4 & 2 & 16 & 8 & 1 & 1 & 1 & 1 & 2 & 2 \\ 2 & 2 & 2 & 1 & 2 & 1 & 4 & 2 & 8 & 4 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 4 & 2 & 4 & 2 & 16 & 8 & 2 & 1 & 2 & 2 & 4 & 4 & 1 & 1 \\ 1 & 1 & 2 & 1 & 4 & 2 & 8 & 4 & 2 & 1 & 1 & 1 & 2 & 2 & 1 & 1 \\ 2 & 1 & 16 & 8 & 1 & 1 & 2 & 2 & 1 & 1 & 4 & 4 & 4 & 2 & 4 & 2 \\ 2 & 1 & 8 & 4 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 4 & 2 & 2 & 1 \\ 4 & 2 & 4 & 2 & 4 & 4 & 1 & 1 & 2 & 2 & 1 & 1 & 16 & 8 & 2 & 1 \\ 2 & 1 & 4 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 1 & 1 & 8 & 4 & 2 & 1 \\ 4 & 2 & 2 & 2 & 1 & 1 & 4 & 4 & 1 & 1 & 4 & 2 & 2 & 1 & 16 & 8 \\ 4 & 2 & 1 & 1 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 1 & 8 & 4 \\ 16 & 8 & 1 & 1 & 2 & 2 & 1 & 1 & 4 & 4 & 2 & 1 & 4 & 2 & 4 & 2 \\ 8 & 4 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 1 & 2 & 1 & 4 & 2 \end{bmatrix}, \quad (13)$$

$$M^{-1} = \begin{bmatrix} 2 & -2 & 1 & -2 & 1 & -1 & 4 & -8 & 2 & -4 & 1 & -1 & 1 & -2 & 1 & -1 \\ -4 & 4 & -2 & 4 & -2 & 2 & -8 & 16 & -2 & 4 & -1 & 1 & -1 & 2 & -1 & 1 \\ 1 & -2 & 1 & -1 & 2 & -4 & 1 & -1 & 1 & -1 & 1 & -2 & 2 & -2 & 4 & -8 \\ -2 & 4 & -2 & 2 & -2 & 4 & -1 & 1 & -1 & 1 & -1 & 2 & -4 & 4 & -8 & 16 \\ 1 & -1 & 2 & -4 & 1 & -1 & 1 & -2 & 1 & -2 & 1 & -1 & 4 & -8 & 2 & -2 \\ -1 & 1 & -2 & 4 & -1 & 1 & -1 & 2 & -2 & 4 & -8 & 1 & -8 & 16 & -4 & 4 \\ 2 & -4 & 1 & -1 & 1 & -2 & 1 & -1 & 2 & -2 & 4 & -8 & 1 & -1 & 1 & -2 \\ -2 & 4 & -1 & 1 & -1 & 2 & -1 & 1 & -4 & 4 & -8 & 16 & -2 & 2 & -2 & 4 \\ 1 & -1 & 1 & -2 & 1 & -1 & 2 & -4 & 4 & -8 & 2 & -2 & 1 & -2 & 1 & -1 \\ -1 & 1 & -1 & 2 & -1 & 1 & -2 & 4 & -8 & 16 & -4 & 4 & -2 & 4 & -2 & 2 \\ 1 & -2 & 1 & -1 & 4 & -8 & 2 & -2 & 1 & -1 & 1 & -2 & 1 & -1 & 2 & -4 \\ -1 & 2 & -1 & 1 & -8 & 16 & -4 & 4 & -2 & 2 & -2 & 4 & -1 & 1 & -2 & 4 \\ 4 & -8 & 2 & -2 & 1 & -2 & 1 & -1 & 1 & -2 & 1 & -1 & 2 & -4 & 1 & -1 \\ -8 & 16 & -4 & 4 & -2 & 4 & -2 & 2 & -1 & 2 & -1 & 1 & -2 & 4 & -1 & 1 \\ 1 & -1 & 4 & -8 & 2 & -2 & 1 & -2 & 1 & -1 & 2 & -4 & 1 & -1 & 1 & -2 \\ -2 & 2 & -8 & 16 & -4 & 4 & -2 & 4 & -1 & 1 & -2 & 4 & -1 & 1 & -1 & 2 \end{bmatrix}. \quad (14)$$

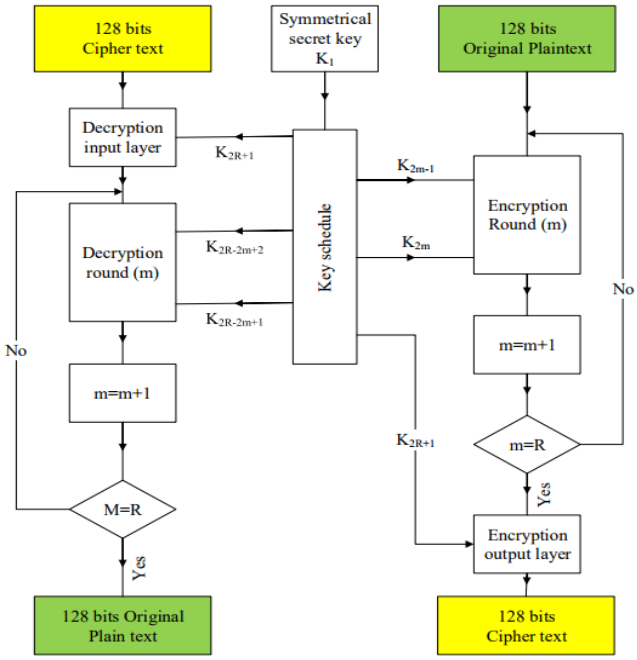

Figure 4: SAFER+ encryption/decryption process.

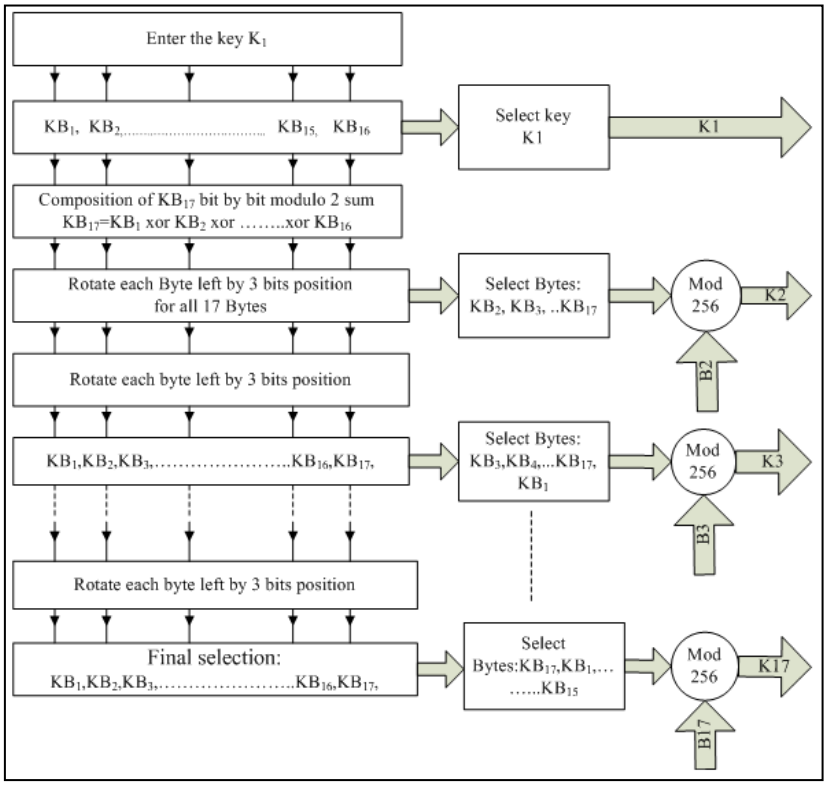

Figure 5: Subkeys Production using the standard B matrix.

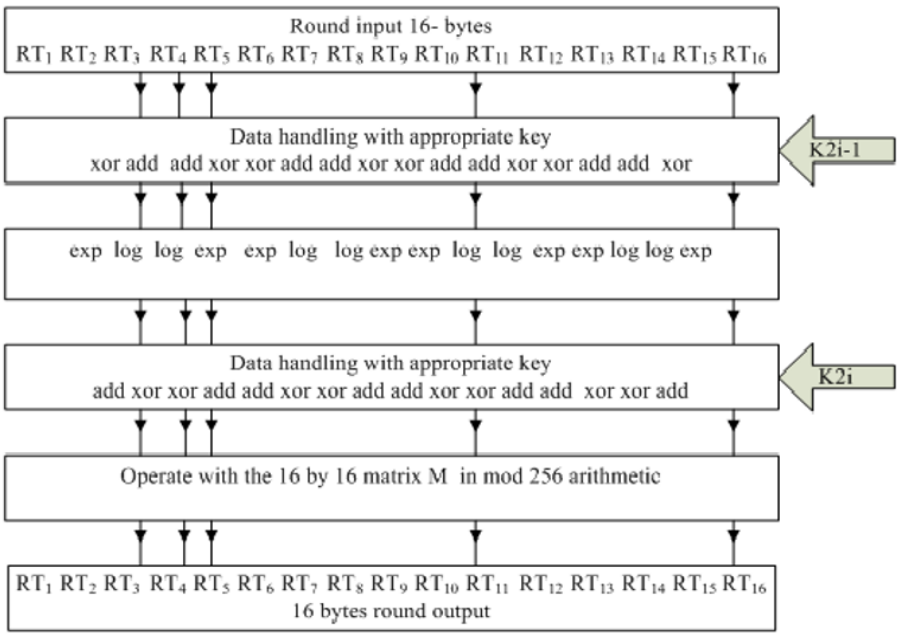

Figure 6: Encrypting round structure.

Table 1: The standard B matrix.

70	151	177	186	163	183	16	10	197	55	179	201	90	40	172	100
236	171	170	198	103	149	88	13	248	154	246	110	102	220	5	61
138	195	216	137	106	233	54	73	67	191	235	212	150	155	104	160
93	87	146	31	213	113	92	187	34	193	190	123	188	153	99	148
42	97	184	52	50	25	253	251	23	64	230	81	29	65	68	143
221	4	128	222	231	49	214	127	1	162	247	57	218	111	35	202
58	208	28	209	48	62	18	161	205	15	224	168	175	130	89	44
125	173	178	239	194	135	206	117	6	19	2	144	79	46	114	51
192	141	207	169	129	226	196	39	47	108	122	159	82	225	21	56
252	32	66	199	8	228	9	85	94	140	20	118	96	255	223	215
250	11	33	0	26	249	166	185	232	158	98	76	217	145	80	210
24	180	7	132	234	91	164	200	14	203	72	105	75	78	156	53
69	77	84	229	37	60	12	74	139	63	204	167	219	107	174	244
45	243	124	109	157	181	38	116	242	147	83	176	240	17	237	131
182	3	22	115	59	30	142	112	189	134	27	71	126	36	86	241
136	70	151	177	186	163	183	16	10	197	55	179	201	90	40	172

5 THE PROPOSED IMPROVEMENT ON SAFER+ ALGORITHM

In the present research, improvements will be made to the process of producing subkeys by replacing the standard matrix B with a matrix generated using the O^2NMNT approach as shown in Figure 7, which will be referred to as the proposed matrix $\widehat{B}$ as illustrated in Table 2. The O^2NMNT was used and preferred over Fast Walsh-Hadamard or DFT because it can used to generate a matrix with element values ranging from 0 to 255, which is not available in the two methods mentioned, which increases the strength of the keys.

The values of the elements of matrix $\widehat{B}$ depend on the constants in the special equations of the O^2NMNT, which means that other matrices can be generated. The original text, the subkeys, the encrypted text, and the text after decryption when using the B matrix are shown in Table 3, while Table 4 illustrates the same procedure using the $\widehat{B}$ matrix. The original SAFER+ encryption algorithm relies, for its strength, on the master key only, from which the rest of the subkeys are generated, as matrix B is known within the encryption algorithm, while in the proposed method, replacing the matrix B with another generated matrix $\widehat{B}$ that is unknown within the encryption algorithm, which leads to increase the strength of the encryption algorithm and make it depend on the master key in addition to its reliance on the $\widehat{B}$ matrix as well.

The O^2NMNT technique can be used to produce a number of different matrices, any of which can be utilized instead of the matrix B used in the standard SAFER+ encryption algorithm. Because these matrices are unknown within the encryption process, they are treated as an additional key which increasing the strength of the code in the suggested encryption approach. To make it possible for hackers to obtain the utilized key in the standard SAFER+ encryption method, the number of possibilities will be 2^{128} because it depends only on the length of the master key (128 bits) from which the sub-keys will be generated, whereas in the proposed algorithm, the master key and sub-keys (17 keys of 128 bits length) are unknown because the matrix $\widehat{B}$ is only available to the two encryption parties (sender and receiver), so the number of possibilities will be $2^{128*17}=2^{2176}$. The number of attempts is extremely high, indicating the strength of the encryption in the suggested method. Encryption and decryption process by the standard and proposed methods take the same time because the operations in both algorithms are the same. The only difference is the time required to generate the matrix $\widehat{B}$, which is stored in a Table 4 and used in the encryption algorithm.

The standard matrix B and the generated matrix $\hat{B}$ are used to encrypt/decrypt a randomly chosen RGB digital image shown in Figure 8. The results of the encryption/decryption process are presented in Figure 9 and Figure 10. The results illustrate that the digital image is successfully encrypted/decrypted by using the two matrices but the advantage of using the

generated matrix $\hat{B}$ lies on the strongness of the generated keys.

When using the proposed method, we notice that the original text (PTX) was recovered after decryption, as shown in Table 4, which means that this method is effective, as the image after decryption is completely identical to the original image.

To measure the randomness of the keys generated by the standard and proposed algorithms, the NIST-STS test and P-values are obtained as illustrated in Table 5. The data is considered random if the P value is greater than 0.01, so the values obtained from the test meet this condition. The results also show that the proposed method is more random than the standard method.

Table 2: The proposed $\hat{B}$ matrix.

15	56	106	105	105	106	56	15	83	26	14	36	91	113	101	44
56	105	15	14	113	112	22	71	101	36	83	106	106	83	36	101
106	15	91	71	71	91	15	106	14	44	22	101	26	105	83	113
105	14	71	44	83	56	113	22	91	106	26	112	112	26	106	91
105	113	71	83	83	71	113	105	91	21	26	15	112	101	106	36
106	112	91	56	71	36	15	21	14	83	22	26	26	22	83	14
56	22	15	113	113	15	22	56	101	91	83	21	106	44	36	26
15	71	106	22	105	21	56	112	83	101	14	91	91	14	101	83
83	101	14	91	91	14	101	83	112	56	21	105	22	106	71	15
26	36	44	106	21	83	91	101	56	22	15	113	113	15	22	56
14	83	22	26	26	22	83	14	21	15	36	71	56	91	112	106
36	106	101	112	15	26	21	91	105	113	71	83	83	71	113	105
91	106	26	112	112	26	106	91	22	113	56	83	44	71	14	105
113	83	105	26	101	22	44	14	106	15	91	71	71	91	15	106
101	36	83	106	106	83	36	101	71	22	112	113	14	15	105	56
44	101	113	91	36	14	26	83	15	56	106	105	105	106	56	15

Table 3: The encryption / decryption process using the standard B matrix.

PTX	69	110	99	114	121	112	116	105	111	110	32	105	115	32	116	104
K1	35	137	230	138	205	192	244	74	99	162	26	212	206	114	33	252
K2	146	206	5	40	169	94	98	37	218	7	89	63	237	49	147	60
K3	165	77	29	246	164	39	48	181	126	207	169	10	174	27	203	5
K4	159	94	89	114	254	175	123	125	236	92	207	22	143	209	174	179
K5	57	99	225	195	11	155	253	8	14	232	208	74	109	203	251	2
K6	138	219	221	229	131	38	103	98	80	208	100	222	174	5	183	212
K7	176	45	13	104	79	132	17	72	133	149	99	197	0	10	77	1
K8	131	60	112	20	202	23	96	197	108	114	68	217	139	211	18	68
K9	224	79	204	195	144	249	239	113	33	54	139	118	217	251	50	39
K10	213	93	117	31	20	235	171	255	72	184	177	243	192	231	188	138
K11	130	85	245	99	80	35	207	29	192	69	182	233	144	60	113	175
K12	163	168	5	66	19	47	236	204	181	179	253	205	194	37	22	23
K13	4	219	25	83	155	141	60	54	182	167	84	184	239	132	198	214
K14	126	221	210	114	182	0	127	143	113	159	70	204	140	188	187	94
K15	177	230	232	249	195	80	80	171	245	102	124	61	122	121	64	190
K16	85	102	122	164	23	111	71	136	91	207	135	155	193	190	47	63
K17	163	105	32	151	68	112	119	4	84	40	217	205	157	40	154	205
CTX	112	1	252	246	168	241	77	148	170	189	118	161	134	163	94	44
DOTX	69	110	99	114	121	112	116	105	111	110	32	105	115	32	116	104

Table 4: The encryption / decryption process using the proposed $\hat{B}$ matrix.

PTX	69	110	99	114	121	112	116	105	111	110	32	105	115	32	116	104
K1	35	137	230	138	205	192	244	74	99	162	26	212	206	114	33	252
K2	91	111	190	215	111	17	138	42	104	234	180	154	238	122	76	4
K3	241	11	130	62	174	2	238	239	235	89	6	6	178	146	234	45
K4	127	170	220	48	219	33	84	158	183	201	250	167	19	159	153	132
K5	69	26	150	208	137	98	18	99	71	145	44	63	33	76	2	201
K6	201	235	108	4	164	84	219	208	148	165	152	156	1	41	221	105
K7	61	153	232	194	175	119	74	222	146	70	130	166	64	177	125	69
K8	129	130	99	180	11	232	100	92	4	190	183	70	70	125	221	50
K9	114	233	132	234	55	135	89	108	110	136	151	65	229	219	37	71
K10	104	53	180	209	238	23	76	43	137	132	76	189	132	112	238	97
K11	160	89	223	6	93	146	33	45	154	207	177	228	161	76	168	16
K12	183	240	250	92	19	76	153	33	226	36	191	200	33	239	54	175
K13	16	145	119	63	192	76	173	201	17	77	83	162	247	125	155	10
K14	148	250	152	253	1	222	221	160	252	209	178	120	221	152	27	211
K15	245	70	213	166	139	177	86	69	109	226	132	212	209	195	98	165
K16	4	135	183	155	70	164	221	125	229	95	220	197	81	169	66	134
K17	71	136	250	65	174	219	218	71	89	155	12	131	61	56	170	48
CTX	201	43	140	164	13	14	125	127	58	100	158	89	58	181	137	103
DOTX	69	110	99	114	121	112	116	105	111	110	32	105	115	32	116	104

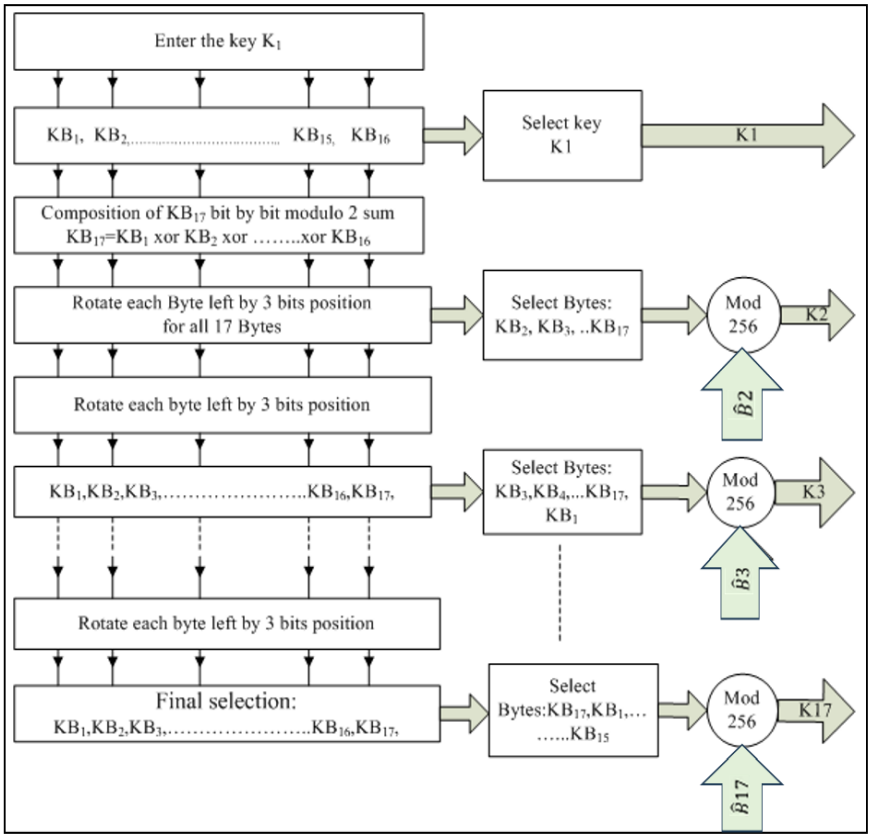

Figure 7: Subkeys production using the proposed $\hat{B}$ matrix.

Table 5: The results of NIST-STS test for the standard and the proposed algorithms.

NIST Test	P-Value for Matrix B	P-Value for Matrix $\widehat{B}$
Frequency (Monobit)	0.4123	0.4772
Block Frequency	0.3465	0.3856
Cumulative Sums (Forward)	0.5218	0.5491
Cumulative Sums (Reverse)	0.4379	0.4897
Runs	0.1986	0.1594
Longest Run of Ones	0.6271	0.6948
Rank	0.7124	0.6382
Discrete Fourier Transform	0.3140	0.3109
Non-overlapping Template	0.4827	0.5083
Overlapping Template	0.2439	0.2185
Universal	0.5068	0.4870
Approximate Entropy	0.3652	0.3398
Serial Test	0.3983	0.4260
Linear Complexity	0.5715	0.5526

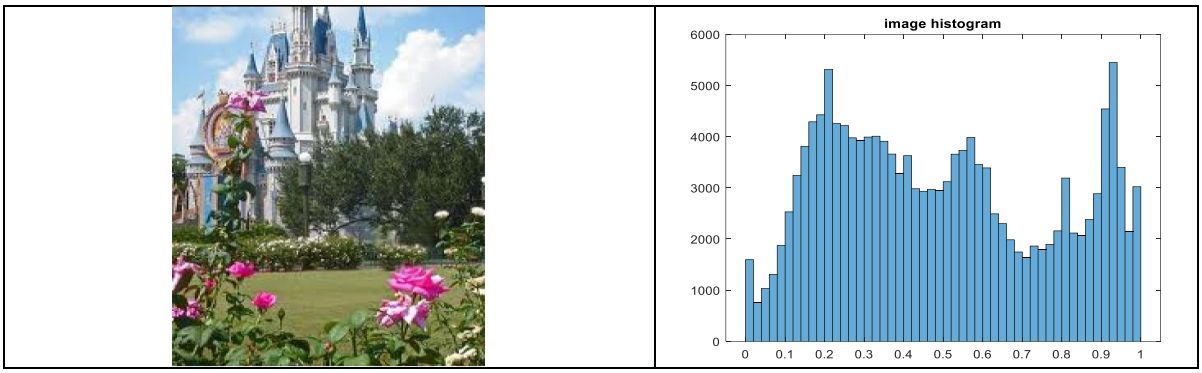

Figure 8: The original image with its histogram.

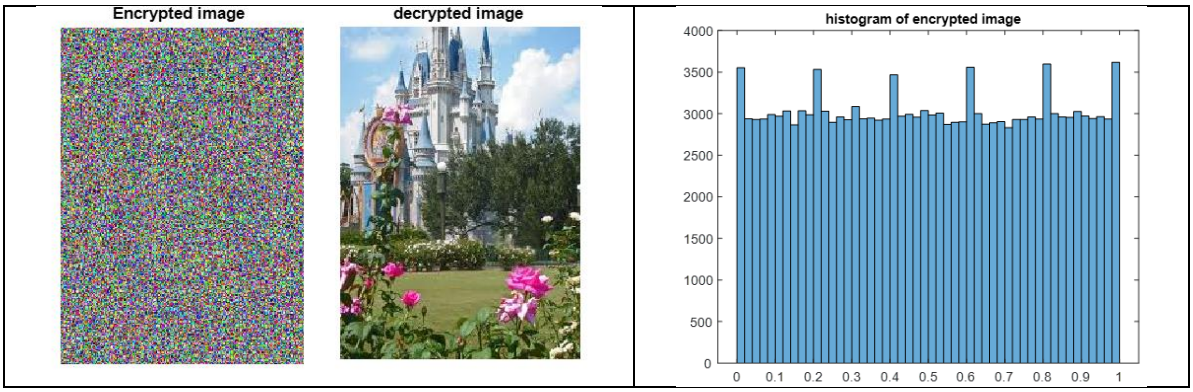

Figure 9: The encryption/decryption process using the standard B matrix.

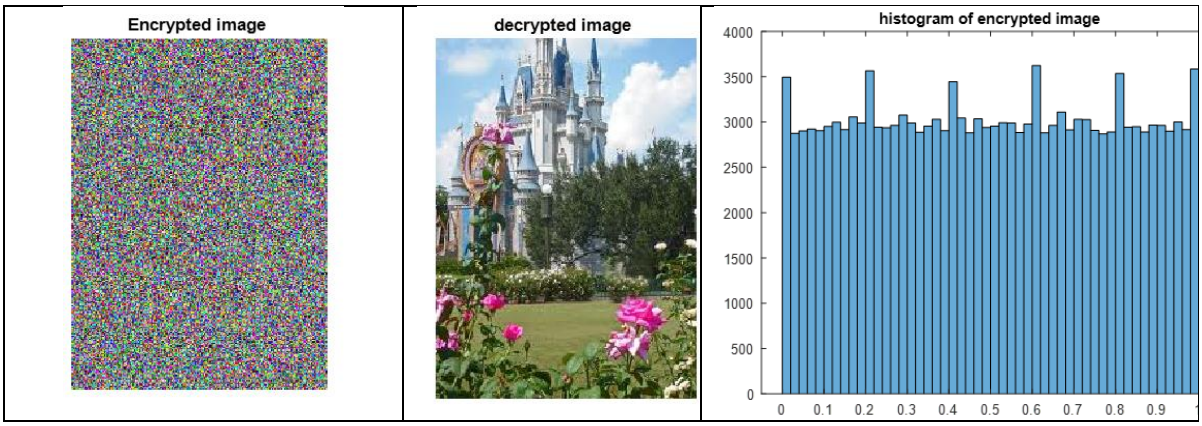

Figure 10: The encryption/decryption process using the proposed $\widehat{B}$ matrix.

6 CONCLUSIONS

This work proposes an improved SAFER+ encryption technique that replaces the original matrix B which is used in the standard SAFER+ algorithm with another matrix $\widehat{B}$. The matrix $\widehat{B}$ has been generated by using the odd-squared new Mersenne number transform (O^2NMNT). A large variety of different matrices can be generated and used in the encryption process. The number of keys used by the suggested encryption method is significantly greater than that generated and used by the standard algorithm, resulting in an extremely high number of attempts for acquiring the keys, and leads to increase the proposed encryption algorithm's strength against hacking. As a result, the difficulty of brute-force or exhaustive key search attacks is greatly amplified, enhancing the algorithm's resilience against hacking attempts. By integrating the O2NMNT-based matrix generation, the improved SAFER+ not only maintains the strong cryptographic foundations of the original algorithm but also achieves higher complexity and unpredictability, making it more robust for secure communications and sensitive data protection in modern cryptographic applications.

REFERENCES

[1] A. Ahmed, M. Naeem, and U. IJEACS, "Analysis of most common encryption algorithms," Int. J. Eng. Appl. Comput. Sci. (IJEACS), vol. 4, no. 2, pp. 1270–1275, 2022, doi: 10.24032/IJEACS/0402/003.

[2] B. Kaushik, V. Malik, and V. Saroha, "A review paper on data encryption and decryption," Int. J. Res. Sci. Innov. (IJRSI), 2023, doi: 10.22214/ijraset.2023.50101.

[3] R. Kumari, J. G. Pandey, and A. Karmakar, "An RTL implementation of the data encryption standard (DES)," arXiv preprint arXiv:2301.05530, 2023, doi: 10.48550/arXiv.2301.05530.

[4] S. M. Kareem and A. M. S. Rahma, "Development of data encryption standard algorithm based on magic square," Indonesian J. Elect. Eng. Comput. Sci., vol. 28, no. 1, pp. 297–305, Oct. 2022, doi: 10.11591/ijeecs.v28.i1.pp297-305.

[5] L. Gong, L. Zhang, W. Zhang, X. Li, X. Wang, and W. Pan, "The application of data encryption technology in computer network communication security," in AIP Conf. Proc., vol. 1834, no. 1, 2017, doi: 10.1063/1.4981623.

[6] M. E. Smid, "Development of the advanced encryption standard," J. Res. Natl. Inst. Stand. Technol., vol. 126, p. 126024, 2021, doi: 10.6028/jres.126.024.

[7] M. F. Mushtaq, S. Jamel, A. H. Disina, Z. A. Pindar, N. S. A. Shakir, and M. M. Deris, "A survey on the cryptographic encryption algorithms," Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 11, 2017, doi: 10.14569/IJACSA.2017.081141.

[8] A. Khompysh, D. Dyusenbayev, and M. Maxmet, "Development and analysis of symmetric encryption algorithm," Int. J. Elect. Comput. Eng. (IJECE), vol. 15, no. 2, pp. 1900–1911, 2025, doi: 10.11591/ijece.v15i2.pp1900-1911.

[9] B. E. H. H. Hamouda, "Comparative study of different cryptographic algorithms," J. Inf. Secur., vol. 11, no. 3, pp. 138–148, 2020.

[10] M. A. M. Abu-Faraj and Z. A. Alqadi, "Using highly secure data encryption method for text file cryptography," Int. J. Comput. Sci. Netw. Secur., vol. 21, no. 12, pp. 53–60, 2021, doi: 10.22937/IJCSNS.2021.21.12.8.

[11] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. 13th ACM Conf. Comput. Commun. Security, 2006, pp. 89–98.

[12] M. R. Parag and T. A. Setu, "Data encryption using image processing: A brief study," Aust. J. Eng. Innov. Technol., vol. 4, no. 3, pp. 45–51, 2022, doi: 10.34104/ajeit.022.045051.

[13] D. Tiwari, B. Mondal, and A. Singh, "Fast encryption scheme for secure transmission of e-healthcare images," Int. J. Image, Graphics Signal Process., vol. 15, no. 5, pp. 88–99, 2023, doi: 10.5815/ijigsp.2023.05.07.

[14] J. Zhao, M. Wang, J. Chen, and Y. Zheng, "New impossible differential attack on SAFER block cipher family," IEICE Trans. Fundam. Electron. Commun. Comput. Sci., vol. 98, no. 3, pp. 843–852, 2015, doi: 10.1587/transfun.E98.A.843.

[15] M. K. Mahmood, L. S. Abdulla, A. H. Mohsin, and H. A. Abdullah, "MATLAB implementation of 128-key length SAFER+ cipher system," Int. J. Eng. Res. Appl., vol. 7, no. 2, pp. 49–55, 2017, doi: 10.9790/9622-0702054955.

[16] D. Sharmila and R. Neelaveni, "A proposed SAFER plus security algorithm using fast Walsh–Hadamard transform for Bluetooth technology," Int. J. Wireless Mobile Netw., vol. 1, no. 2, pp. 80–88, 2009.

[17] B. Jagadeesh, D. Kishore, and R. V. Vijay Krishna, "Design of SAFER+ encryption algorithm for Bluetooth transmission," Int. J. Innov. Technol. Res., vol. 3, no. 1, pp. 1864–1867, Jan. 2015.

[18] M. Shrivastava and R. Sinha, "An improved version of SAFER+ algorithm," Int. J. Comput. Intell. Res., vol. 13, no. 6, pp. 1431–1440, 2017.

[19] M. F. Şahin, M. K. Mahmood, and I. Myderrizi, "Secure and fast encryption routine+: Evaluation by software application," Int. J. Eng. Technol. (IJET), vol. 6, no. 2, pp. 13–24, 2020, doi: 10.19072/ijet.755570.

[20] J. Hua, F. Liu, Z. Xu, F. Li, and D. Wang, "A fast realization of new Mersenne number transformation and its applications," Int. J. Circuit Theory Appl., vol. 47, no. 5, pp. 738–752, 2019, doi: 10.1002/cta.2614.

[21] Y. Al-Aali, M. T. Hamood, and S. Boussakta, "Radix-22 algorithm for the odd new Mersenne number transform (ONMNT)," Signals, vol. 4, no. 4, pp. 746–767, 2023, doi: 10.3390/signals4040041.

[22] L. S. Abdulla, A. Abdulmuttalib, W. Hussein, and M. T. Hamood, "Unified algorithms for generalized new Mersenne number transforms," TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 21, no. 6, pp. 1346–1355, 2023, doi: 10.12928/TELKOMNIKA.v21i6.25253.