An Improved Hybrid Encryption Scheme Based on the Sequence of Reduced Residue Systems

Muhammad A. H.[1], Ibrahim A. A.[2], Garba A. I.[2], Sarki M. N.[3], Mua'azu S. B.[3], Abubakar S. F.[3], Shehu S.[4], James T. O.[3] and Abubakar T. U.[5]

[1]Department of Science, Mathematics Unit, State Collage of Basic & Remedial Studies, Sokoto.

[2]Department of Mathematics, UsmanuDanfodio University Sokoto.

[3]Department of Mathematics, Abdullahi Fodio University of Science and Technology, Aliero.

[4]Department of Mathematics, Sokoto State University.

[5]Department of Mathematics, ShehuShagari College of Education, Sokoto.

DOI: https://dx.doi.org/10.51244/IJRSI.2025.1210000348

Received: 10 November 2025; Accepted: 16 November 2025; Published: 22 November 2025

ABSTRACT

The rapid growth of digital communication has created an urgent demand for advanced cryptographic techniques that ensure both security and efficiency. As reliance on digital platforms increases, so does the risk of cyber threats and data breaches. Robust cryptography is, therefore, essential to protect sensitive information and maintain trust in digital transactions. This research proposes an improved composed hybrid cryptosystem that integrates Transposition, Caesar, and Hill ciphers, followed sequentially by RSA encryption. The study examines how a hybrid of four ciphers can be attacked when treated as a composite function. To further enhance security, a sequence of Reduced Residue System (RRS) values was introduced to replace ASCII characters after the third cipher (Hill cipher), adding an additional layer of residue-based encryption before the final RSA stage. The findings demonstrate that the improved hybrid cryptosystem significantly enhances data security and key generation efficiency, adding a new level of complexity that makes it more challenging for attackers to guess or compute decryption keys.

Keywords: Hybrid, Encryption, Decryption, Attacks and RSA.

INTRODUCTION

Cryptography is the study of secure communication techniques that allow the sender and intended recipient of a message to view and understand its contents, (Ibrahim, *et al.*, 2021). In today's digital world, the need for robust data security has never been more critical as now. Information exchange proliferates across various platforms and networks, safeguarding sensitive data from unauthorized access, modification, or disclosure becomes paramount, (Shehu *et al*., 2023). Individual encryption methods, while effective, often face challenges related to computational efficiency, key management, and vulnerable to attacks, (Yao & Su, 2021). Hybrid encryption systems have emerged as a widely adopted solution to these challenges, (Manna, *et al.*, 2017). Recently, numerous research endeavours have focused on enhancing security using the hybrid cryptosystem. For instance, Rufa'i *et al*., (2020) integrated RSA, Shifting, and Hill ciphers to enhance security and robustness, while Hassan, Garko, *et al.*, (2023) combined Hill and Transposition ciphers to improve data protection. Despite these advancements, vulnerabilities remain in existing hybrid cryptosystems. This paper aims to enhance data security by improving a hybrid encryption system that sequentially integrates Transposition, Caesar, Hill, and RSA ciphers, using a sequence of Reduced Residue Systems (RRS) to replace ASCII characters.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

Integrating an RRS sequence into a hybrid cryptosystem enhances both data security and efficiency by combining the strengths of hybrid encryption and residue-based techniques. This modification addresses vulnerabilities found in the existing system and strengthens its resistance to potential attacks. The additional layer of complexity introduced by RRS makes it considerably more difficult for attackers to decipher ciphertexts.

LITERATURE REVIEW

The concept of hybrid cryptography has attracted considerable attention in recent years due to its ability to provide enhanced security features. Some of the most relevant works are:

Rufa'i et al. (2020) combined three ciphers—RSA, Shifting, and Hill—and represented them as bijective functions. The composition of these functions resulted in an improved hybrid cipher system with enhanced encryption performance.

Khan, Pradhan, & Chandavarkar (2021) proposed a hybrid algorithm based on the Guillou–Quisquater scheme and RSA. Their model aims to ensure data integrity through RSA-based key generation, while the Guillou–Quisquater component handles integrity and confidentiality.

Suhasini and Bushra (2021) introduced a three-level encryption technique designed to overcome the limitations of single-key encryption by combining the Advanced Encryption Standard (AES), Data Encryption Standard (DES), and RSA. This multi-tier approach provides greater resistance against brute-force and key-compromise attacks.

Saja, Zaynab, and Jaafer (2023) proposed a two-layer hybrid cryptosystem. In the first layer, plaintext is encrypted using a modified Playfair cipher to produce ciphertext, which is subsequently re-encrypted using RSA to generate the final ciphertext. This layered design enhances overall data confidentiality.

Prakash, Saeed, Rajan, Mohammad, and Ahmed (2023), in their paper presented a scheme combining RSA with a Simple Symmetric Key (SSK) algorithm.

Susmitha, Kumar, and Bulla (2023) demonstrated that a hybrid cryptographic approach combining AES and ElGamal enhances file security by integrating symmetric and asymmetric techniques. Their method ensures confidentiality and integrity for stored data and provides valuable insights into key management strategies.

Hassan et al. (2023) enhanced data security using a combination of Hill and Transposition ciphers. In their method, plaintext is first encrypted with the Hill cipher and then re-encrypted with the Transposition cipher, resulting in a more complex ciphertexts.

Ariffin, Wijonarko, Suwarno, and Kristianto (2024) combined the Unimodular Hill Cipher with RSA to produce a more secure text encryption system.

Despite these significant contributions, hybrid cryptosystems remain susceptible to certain forms of attack. Consequently, there is a need for further improvement. This study aims to advance the composed hybrid cryptosystem by incorporating a sequence of Reduced Residue Systems (RRS) to replace ASCII characters, thereby increasing security complexity.

METHODOLOGY

The methodology employed in this study incorporates several fundamental concepts of number theory, including the Greatest Common Divisor (GCD), Euler's Totient Function $\varphi(N)$, Congruence, and the concept of the Reduced Residue System (RRS), as well as the American Standard Code for Information Interchange (ASCII) table. Furthermore, it integrates four cryptographic techniques: Transposition cipher, Caesar cipher, Hill cipher, and RSA algorithm, alongside the method of composing multiple ciphers.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

ASCII Table

The complete 7-bit ASCII table and corresponding decimal equivalents are presented in table 1.

Table 1: ASCII Table (Google Search, 2024)

Dec	Chr	Dec	Chr	Dec	Chr	Dec	Chr	Dec	Chr
0	NUL	26	SUB	52	4	78	N	104	h
1	SOH	27	ESC	53	5	79	O	105	i
2	STX	28	FS	54	6	80	P	106	j
3	ETX	29	GS	55	7	81	Q	107	k
4	EOT	30	RS	56	8	82	R	108	l
5	ENQ	31	US	57	9	83	S	109	m
6	ACK	32		58	:	84	T	110	n
7	BEL	33	!	59	;	85	U	111	o
8	BS	34	"	60	<	86	V	112	p
9	HT	35	#	61	=	87	W	113	q
10	LF	36	$	62	>	88	X	114	r
11	VT	37	%	63	?	89	Y	115	s
12	FF	38	&	64	@	90	Z	116	t
13	CR	39	'	65	A	91	[	117	u
14	SO	40	(	66	B	92	\	118	v
15	SI	41	)	67	C	93	]	119	w
16	DLE	42	*	68	D	94	^	120	x
17	DC1	43	+	69	E	95	_	121	y
18	DC2	44	,	70	F	96	`	122	z
19	DC3	45	-	71	G	97	a	123	{
20	DC4	46	.	72	H	98	b	124	|
21	NAK	47	/	73	I	99	c	125	}
22	SYN	48	0	74	J	100	d	126	~
23	ETB	49	1	75	K	101	e	127	DEL
24	CAN	50	2	76	L	102	f		
25	EM	51	3	77	M	103	g		

Fundamental Concepts of Number Theory

This section presents key number theory concepts utilized in the study.

Greatest Common Divisor (GCD)

If *m* and *n* are integers, a positive integer *d* is the GCD of *m* and *n* if *d* divides both *m* and *n*, and *d* is the greatest among all common divisors of *m* and *n*. For instance, the GCD of 24 and 60 is 12, denoted as (24, 60) = 12.

Relatively Prime Integers

Two or more integers are said to be relatively prime if their GCD is 1, (Hardy & Wright, 2008). For instance, the factors of 8 are {1, 2, 4, 8}, and the factors of 15 are {1, 3, 5, 15}. Thus, GCD (8, 15) = 1, meaning 8 and 15 are relatively prime.

Euler's Totient Function $\varphi(N)$

Euler's Totient Function, denoted by $\varphi(N)$, counts the number of positive integers less than or equal to *N* that are relatively prime to *N*, (Rosen, 2011). Mathematically,

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

$\varphi(N) = k$, such that GCD$(k, N) = 1, k \in \mathbb{Z}, 1 \leq k \leq N$.

For instance: $\varphi(10) = 4$, as numbers less than or equal to 10 are: $1, 2, 3, 4, 5, 6, 7, 8, 9, 10$, for which integers ≤ 10 that are coprime with 10 are: 1, 3, 7, 9.

Similarly, $\varphi(12) = 4$, because the integers ≤ 12 that are coprime with 12 are {1, 5, 7, 11}.

It follows that $\varphi(2) = 1,\ \varphi(3) = 2,\ \varphi(4) = 2,\ \varphi(5) = 4,\ \varphi(6) = 2,$ thus, in general

$\varphi(p) = p - 1$, where p is prime.

Properties of $\varphi(N)$

The Euler Totient function has many useful properties:

(i) φ is multiplicative when GCD$(m,\ n) = 1$, thus, $\varphi(mn) = \varphi(m) \cdot \varphi(n)$

For instance, $\varphi(12) = \varphi(3 \cdot 4)$

$$= \varphi(3) \cdot \varphi(4) \ \Rightarrow\ \varphi(12) = 4$$

(ii) For any prime p, $\varphi(p^{\alpha}) = p^{\alpha}\left(1 - \frac{1}{p}\right) = p^{\alpha-1}(p-1)$, hence,

if $N = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot p_3^{\alpha_3} \cdot \ldots \cdot p_k^{\alpha_k}$ then, $\varphi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right) \ldots \left(1 - \frac{1}{p_k}\right)$, (Rosen, 2011)

Congruence

If a and b are integers and m is a positive integer, then a is said to be congruent to b modulo m if m divides $(a - b)$. This is written as $a \equiv b\ (mod\ m)$, (Rosen, 2011).

For instance, $27 \equiv 13\ (mod\ 14)$, since 14 divides $27 - 13$.

Least Residue

If $m > 0$ and r is the remainder when dividing n by m then r is called the least residue of n modulo m, (Rosen, 2011).

Residue Classes

A residue class modulo m is a set of integers that are congruent to each other modulo m. Each residue class contains exactly one integer from $\{0, 1, 2, \ldots, m-1\}$. For modulo 4, there are four residue classes $[0] = 0 + 4k, [1] = 1 + 4k, [2] = 2 + 4k, [3] = 3 + 4k$, where $k \in \mathbb{Z}$.

Complete Residue System

A complete residue system modulo m is any set of m integers $[c_1, c_2, c_3, \ldots, c_m]$ such that

$c_i \not\equiv c_j (mod\ m)$, (Rosen, 2011).

Reduced Residue System (RRS)

A Reduced Residue System (RRS) modulo m is a set of integers that represent all possible residue classes modulo m that are relatively prime to m (Rosen, 2011).

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

A set S is an RRS modulo m if:

i) $n(S) = \varphi(m)$ (ii) $a_i \not\equiv a_j (mod\ m)$ (iii) GCD $(a_i, m) = 1$

The set $S = \{-1, 7, 13, 11, 14, 19\}$ forms an RRS modulo 9; since

(i) $\varphi(9) = 6$ and $n(S) = 6$ (ii) $\{-1, 7, 13, 11, 14, 19\} \equiv \{8, 7, 4, 2, 5, 1\} \Rightarrow a_i \not\equiv a_j (mod\ m)$

(iii) $(a_i, 9) = 1$, $a_i \in S$

3.3 Composition of Four Ciphers

The four ciphers: Transposition, Caesar, Hill, and RSA were modelled as mathematical functions and composed sequentially using the concept of function composition.

Encryption Process

Step I: Transposition Cipher (T) - Rearrange the plaintext according to a specific key to obtain ciphertexts X_t as:

$T: T(p) = X_t$ that is $T: p \rightarrow X_t$

Step II: Caesar Cipher (C) - Shift each letter in X_t by a fixed number (K_c) of positions down the ASCII table to obtain $X_{tc} = X_t + K_c$ as:

$C: C(T(p))$ that is $C: X_t \rightarrow X_{tc}$.

Step III: Convert X_{tc} into its ASCII numerical equivalents, denoted X'_{tc}.

Step IV: Hill Cipher (H): Encrypt X'_{tc} using an invertible key matrix K_h to obtain X_{tch} as:

$H: H(C(T(p))) = C(T(p)) * K_h$ that is $H: X'_{tc} \rightarrow X_{tch}$,

Step V: RSA Encryption (R) - Encrypt X_{tch} using modular exponentiation with public key e to obtain X_{tchr} as:

$R: R(H(C(T(p)))) = (H(C(T(p)))^e)\ (mod\ N)$ that is $R: X_{tch} \rightarrow X_{tchr}$

Thus, the composed cipher is: $(R \circ H \circ C \circ T)p = R\big(H(C(T(p)))\big)$

Where: X_{tc} = ciphertexts from Caesar encryption X'_{tc} = ASCII numerical equivalent of X_{tc}

X_{tch} = ciphertexts from Hill encryption X_{tchr} = ciphertexts from RSA encryption

N = the RSA modulus

Decryption Process

Decryption reverses the encryption steps as follows:

Step I: RSA Decryption - decrypt X_{tchr} using private key d to obtain X_{tch} as:

$$R^{-1}(X_{tchr}) = X_{tchr}^d\ (mod\ N)$$

$$= X_{tch}$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

Step II: Hill Decryption - Multiply X_{tch} inverse key matrix K_h^{-1} to recover X'_{tc} as:

$$H^{-1}(X_{tch}) = X_{tch} \times K_h^{-1}$$

$$= X'_{tc}$$

Step III: Convert X'_{tc} to ASCII characters to obtain X_{tc}

Step IV: Caesar Decryption - shift each character of X_{tc} upward by K_c positions:

$$C^{-1}(X_{tc}) = (X_{tc} - K_c)(mod\ n)$$

$$= X_t$$

Step V: Transposition Decryption - rearrange X_t back to its original order using the transposition key K_t: $T^{-1}(X_t) = p$

Hence, the complete decryption process is represented as: $T^{-1}\left(C^{-1}\left(H^{-1}\left(R^{-1}(X_{tchr})\right)\right)\right) = p$

Where: R^{-1} = RSA decryption function H^{-1} = Hill decryption function

C^{-1} = Caesar decryption function T^{-1} = Transposition decryption function

K_h^{-1} = inverse of Hill key matrix X_{tchr} = ciphertexts of the composed system

RESULT AND ANALYSIS

This section presents a comprehensive analysis of the findings, highlighting the performance and security enhancements achieved through the proposed approach.

Cryptanalysis of the Combined Four Ciphers

If an attacker knows the cipher order (Transposition → Caesar → Hill → RSA), the hybrid can possibly be attacked by peeling the layers in reverse: first break RSA, then the Hill cipher, then Caesar, and finally the transposition, as illustrated below:

Step I: Apply RSA cryptanalytic methods to recover the decrypted block stream

Step II: Use Hill-specific attacks (known-plaintext, linear algebraic key recovery, or brute force) on the result to recover the message before Hill.

Step III: Break the Caesar shift via frequency analysis, known plaintext, or simple brute force to get the pre-Caesar text.

Step IV: Determine and reverse the transposition pattern to reconstruct the original plaintext.

Mathematically, the process can be illustrated as follows:

Step I: RSA Attack: $R^* = R^{-1}(X_{tchr})$

$$= X_{tch} \quad (1)$$

Step II: Hill Attack: $H^* = H^{-1}(X_{tch})$ (From equation 1)

$$= X_{tc} \quad (2)$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

Step III: Caesar Attack: $C^* = C^{-1}(X_{tc})$ (From equation 2)

$$= X_t \quad (3)$$

Step IV: Transposition Attack: $T^* = T^{-1}(X_t)$ (From equation 3)

$$= P$$

Where: $X_t =$ Ciphertext obtained using the Transposition method

$X_{tc} =$ Ciphertext obtained after applying the Caesar cipher

$X_{tch} =$ Ciphertexts obtained after applying the Hill cipher

$X_{tchr} =$ Ciphertexts obtained after applying the RSA algorithm

$R^*, H^*, C^*, T^* =$ Attacks on RSA, Hill, Caesar, and Transposition ciphers respectively

Generating RRS Sequence

To generate a Sequence of Reduce Residue System (S_R) a modulus M is selected such that the number of elements in the set corresponds to $\varphi(M) \geq 128$, where 128 represents the total number of ASCII characters, and $\varphi(M)$ denotes Euler's Totient Function.

Let $M = 255$

$$= 3 \cdot 5 \cdot 17$$

Then: $\varphi(255) = \varphi(3 \cdot 5 \cdot 17)$

$$= \varphi(3)\varphi(5)\varphi(17)$$

$$= 2 \cdot 4 \cdot 16$$

$$= 128$$

Thus, the RRS sequence modulo 255 can be represented as:

$$S_R = \left\{ \begin{array}{c} 1, 2, 4, 7, 8, 11, 13, 14, 16, 19, 22, 23, 26, 28, 29, 31, 32, 37, 38, 41, 43, 44, \\ 46, 47, 49, 52, 53, 56, 58, 59, 61, 62, 64, 57, 71, 73, 74, 76, 77, 79, 82, 83, \\ 86, 88, 89, 91, 92, 94, 97, 98, 101, 103, 104, 106, 107, 109, 112, 113, 116, \\ 118, 121, 122, 124, 127, 128, 131, 133, 134, 137, 139, 142, 143, 146, \\ 148, 149, 151, 152, 154, 157, 158, 161, 163, 164, 166, 167, 169, 172, 173, \\ 176, 178, 179, 181, 182, 184, 188, 191, 193, 194, 196, 197, 199, 202, 203, \\ 206, 208, 209, 211, 212, 214, 217, 218, 223, 224, 226, 227, 229, 232, \\ 233, 236, 239, 241, 242, 244, 247, 248, 251, 253, 254 \end{array} \right\}$$

Assigning this generated RRS sequence S_R to the corresponding ASCII character values allows the replacement of standard ASCII encoding with residue-based encoding. This substitution increases encryption complexity, as each ASCII value is represented by its equivalent in the residue sequence, thus enhancing security before the final RSA encryption phase.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

Table 2: Integrated ASCII Table with RRS

ASCII Value	Chr	RRS Value		ASCII Value	Chr	RRS Value		ASCII Value	Chr	RRS Value
0	**NUL**	1		43	+	88		86	V	172
1	**SOH**	2		44	,	89		87	W	173
2	**STX**	4		45	-	91		88	X	176
3	**ETX**	7		46	.	92		89	Y	178
4	**EOT**	8		47	/	94		90	Z	179
5	**ENQ**	11		48	0	97		91	[	181
6	**ACK**	13		49	1	98		92	\	182
7	**BEL**	14		50	2	101		93	]	184
8	**BS**	16		51	3	103		94	^	188
9	**HT**	19		52	4	104		95	_	191
10	**LF**	22		53	5	106		96	`	193
11	**VT**	23		54	6	107		97	a	194
12	**FF**	26		55	7	109		98	b	196
13	**CR**	28		56	8	112		99	c	197
14	**SO**	29		57	9	113		100	d	199
15	**SI**	31		58	:	116		101	e	202
16	**DLE**	32		59	;	118		102	f	203
17	**DC1**	37		60	<	121		103	g	206
18	**DC2**	38		61	=	122		104	h	208
19	**DC3**	41		62	>	124		105	i	209
20	**DC4**	43		63	?	127		106	j	211
21	**NAK**	44		64	@	128		107	k	212
22	**SYN**	46		65	A	131		108	l	214
23	**ETB**	47		66	B	133		109	m	217
24	**CAN**	49		67	C	134		110	n	218
25	**EM**	52		68	D	137		111	o	223
26	**SUB**	53		69	E	139		112	p	224
27	**ESC**	56		70	F	142		113	q	226
28	**FS**	58		71	G	143		114	r	227
29	**GS**	59		72	H	146		115	s	229
30	**RS**	61		73	I	148		116	t	232
31	**US**	62		74	J	149		117	u	233
32		64		75	K	151		118	v	236
33	!	67		76	L	152		119	w	239
34	"	71		77	M	154		120	x	241
35	#	73		78	N	157		121	y	242
36	$	74		79	O	158		122	z	244
37	%	76		80	P	161		123	{	247
38	&	77		81	Q	163		124	|	248
39	'	79		82	R	164		125	}	251
40	(	82		83	S	166		126	~	253
41	)	83		84	T	167		127	**DEL**	254
42	*	86		85	U	169				

Improved Composed Hybrid

The improved composed hybrid encryption method involves a two-stage processes. Firstly, the generated sequence of Reduced Residue System (RRS) will be applied to replace the characters of the ciphertexts

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

obtained after third encryption (Hill cipher encryption, C_H^*) with their corresponding assigned numerical values as in Table 2. In the second stage, the resulting numerical string is encrypted using RSA encryption techniques.

Procedure for Encryption /Decryption Process of the Modified Hybrid

Encryption Process:

1. Select a transposition key k_T

2. Rearrange the plaintext characters base on the key k_T to have ciphertexts $C_T = p_i \circ k_T$ as:

$T: P \rightarrow C_T \Rightarrow \mathrm{T}: \mathrm{T}(\mathrm{P}) = C_T$ that is

$$C_T = (t_1 t_2 \dots t_n) \quad (4)$$

Where P is the plaintext, $p_i \in P$, $i = 1, 2, \dots, n$ and n is the number of characters in the plaintext.

3. Select Caesar key k_C

4. Compute$C_C = t_i + k_C$, to obtain a new ciphertexts as:

$C: C_T \rightarrow C_C \Rightarrow \mathrm{C}: \mathrm{C}(C_T) = C_C$ that is

$$C_C = (c_1 c_2 \dots c_n) \quad (5)$$

where $t_i \in C_T,\ i = 1, 2, \dots,\ n.$

5. Split equation (4.5) as column vectors: $C_C = \binom{c_1}{c_2}, \binom{c_3}{c_4}, \dots, \binom{c_{n-1}}{c_n}$ (6)

6. Replace the column vectors of (4.6) with their corresponding ASCII values to have:

$$C_C^* = \binom{\alpha_1}{\alpha_2}, \binom{\alpha_3}{\alpha_4}, \dots, \binom{\alpha_{n-1}}{\alpha_n} \quad (7)$$

7. Choose an invertible key matrix $k_H = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}$

8. Compute $C_H = k_H\, C_C^* \,(mod\ 128)$, to have: $H: C_C^* \rightarrow C_H \Rightarrow \mathrm{H}: \mathrm{H}(C_C^*) = C_H$ that is

$$C_H = (h_1 h_2 \dots h_n) \quad (8)$$

9. Convert the values of equation (4.8) to their corresponding characters in ASCII to obtain

$$C_H^* = (\beta_1, \beta_2, \dots, \beta_n) \quad (9)$$

10. Select $M \in Z^+$ such that $\varphi(M) \geq 128$, that is value greater than or equals to total number of characters in ASCII.

11. Generate sequence of Reduce Residue System then, assign each character of the ASCII (A) with a sequential value of RRS, denoted as:

$A: A \rightarrow S_R \Rightarrow A: A(a_i) = r_i$ that is

$$S_R = \{r_1, r_2, \dots, r_{\varphi(M)}\} \quad (10)$$

12. Replace each letter of equation (4.9) with the corresponding value of S_R, to have:

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

$A: C_H^* \rightarrow C_H^{**} \Rightarrow \mathrm{A}: \mathrm{A}(C_H^*) = C_H^{**}$ that is

$$C_H^{**} = (\gamma_1 \gamma_2 \dots \gamma_n) \qquad (11)$$

13. Select RSA public key (e, N) such that GCD $\big(e, \varphi(N)\big) = 1$, where e is the encryption key, N is the RSA modulus and $\varphi(N)$is the Euler number.

14. Encrypt equation (4.11) using RSA method, with a public key (e, N), to obtain a final ciphertexts C_R^* as:
$$C_R^* = (\gamma_1)^e (mod\ N) \qquad (12)$$

Thus, the improved composed cipher becomes: $(R \circ C_H^{**} \circ C \circ T)p = R\big(C_H^{**}(C(T(p)))\big)$

Decryption Process

The decryption process involves reversing the above encryption process:

1. Compute RSA private key d using RSA key equation $ed - k\varphi(N) = 1$.

2. Decrypt the ciphertexts in equation (4.12) to recover $C_H^{**} = (\gamma_1 \gamma_2 \dots \gamma_n)$ as in equation (11):

$$R^{-1}\ (C_R^*)\ =\ (C_R^*)^d\ mod\ N\ = C_H^{**}$$

3. Map each value of C_H^{**} with the corresponding value in RRS(S_R)then, replace it with corresponding values in ASCII to recover C_H^* as: $C_H^* = (\beta_1, \beta_2, \dots, \beta_n)$ (as in equation (9))

where $\beta_i \in A$ and A =ASCII characters.

4. Convert the characters in C_H^* to numerals corresponding to ASCII characters to recover C_H which is the Hill ciphertexts, as:$C_H = (h_1 h_2 \dots h_n)$ (as in (8))

5. Decrypt the resulting ciphertexts $C_H = (h_1 h_2 \dots h_n)$ using inverse key matrix to recover C_C^*:

$H^{-1}(C_H) =\ C_H \times k_H^{-1} (mod\ 128)$ to have:

$$C_C^* = \binom{\alpha_1}{\alpha_2}, \binom{\alpha_3}{\alpha_4}, \dots, \binom{\alpha_{n-1}}{\alpha_n}$$ (as in equation (7))

6. Convert the values ofC_C^* to corresponding characters in ASCII to recover C_C:

$$C_C = (c_1 c_2 \dots c_n)$$ (as in equation (6))

7. Shift each character of C_C back to a fixed number of positions upward the ASCII based on the Caesar key k_C to recover C_T as: $C^{-1}\ (C_C)\ =\ (C_C\ -\ k_C)\ (mod\ n)$.

$$= C_T$$ (as in equation (4))

8. Rearrange the letters of C_T back to the original order using the transposition key k_T to recover the plaintexts $P = t_i \circ k_T$as: $T^{-1}: C_T \rightarrow P\ \Rightarrow\ \mathrm{T}^{-1}: \mathrm{T}^{-1}(C_T) =\ \mathrm{P}$ that is:

$$T^{-1}(C_T) = (p_1 p_2 \dots p_n)$$

Thus, the combined decryption process becomes:

$$(\mathrm{T}^{-1} \circ \mathrm{C}^{-1} \circ \mathrm{H}^{-1} \circ \mathrm{R}^{-1}) C_R^* = T^{-1}(C^{-1}\ (H^{-1}(R^{-1}(C_R^*))))$$

Where: R^{-1} = RSA decryption function H^{-1} = Hill decryption function

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

C^{-1} = Caesar decryption function T^{-1} = Transposition decryption function

k_H^{-1} = Inverse of Hill cipher key matrix C_H^* = ASCII values of the Hill ciphertexts

C_H^{**} = RRS value of the ciphertexts C_H^* C_R^* is the ciphertexts of the modified hybrid.

d = RSA secret key N = RSA modulus k_C = Caesar key

Application of the Improved Hybrid Cryptosystem

Suppose, to encrypt the message "The message is confidential. Please, keep it secret."

Firstly, using Transposition to encrypt the statement with the transposition key $k_T = 6$, we have:

Table 3: Encryption Table for Transposition Cipher

T	h	e		m	e
s	s	a	g	e	
i	s		c	o	n
f	i	d	e	n	t
i	a	l	.		P
l	e	a	s	e	,
	k	e	e	p	
i	t		s	e	c
r	e	t	.		

The ciphertexts consists of the characters read from the top left box going down the column to have ciphertexts C_T as: Tsifil irhssiaekteea dlae t gce.ses.meon epee ntP, c

Secondly, encrypt the ciphertexts C_T using Caesar method with key, $k_C = 3$, to obtain C_C as: $C_C = (t_i + 3)\ mod\ 128$, illustrated below:

$$T + 3 = 84 + 3 = 87 = W \text{ (From ASCII Table)}$$

$$\vdots$$

$$c + 3 = 99 + 3 = 102 = f$$

To have a ciphertexts C_C *as:* Wvlilo#lukvvldhnwhhd#godh#w#jfh1vhv1phrq#hshh#qwS/#f

Thirdly, encrypt ciphertexts C_C using Hill cipher technique with encryption matrix key

$k_H = \begin{pmatrix} 5 & 2 \\ 7 & 3 \end{pmatrix}$ as:

$$\begin{pmatrix} 5 & 2 \\ 7 & 3 \end{pmatrix}\begin{pmatrix} 087 \\ 118 \end{pmatrix}(mod\ 128) = \begin{pmatrix} 435 + 236 \\ 609 + 354 \end{pmatrix}(mod\ 128) = \begin{pmatrix} 671 \\ 963 \end{pmatrix}(mod\ 128) = \begin{pmatrix} 031 \\ 067 \end{pmatrix} = \begin{pmatrix} \boldsymbol{US} \\ C \end{pmatrix}$$

$$\vdots$$

$$\begin{pmatrix} 5 & 2 \\ 7 & 3 \end{pmatrix}\begin{pmatrix} 035 \\ 102 \end{pmatrix}(mod\ 128) = \begin{pmatrix} 175 + 204 \\ 245 + 306 \end{pmatrix}(mod\ 128) = \begin{pmatrix} 379 \\ 551 \end{pmatrix}(mod\ 128) = \begin{pmatrix} 123 \\ 039 \end{pmatrix} = \begin{pmatrix} \{ \\ ' \end{pmatrix}$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

To obtain a new ciphertexts C_H as:

031067110047122065007057031116058028005027100034035121080004125042115053078065025042094024106107030114048077000072028113127045015093078065035124125082123039

C_H^* = ***US***Cn/zA***BEL***9***US***t:***FSENQESC***d"#yP***EOT***}*s5NA***EM****^***CAN***j***RS***r0M***NUL***H***FS***q**DEL**-***SI***]NA#|}R{'

Fourthly, apply Sequence of Reduce Residue System (RRS) with $M = 255$, to obtain C_H^{**} *as:*

$M = 255$, to obtain C_H^{**} *as:*

062134218094244131014113062232116058011056199071073242161008251086229106157131052086094211212061227043197001146058226254091031184157131073248251218247079

Next, generate RSA Public key $(e,\ N)$ and Private key $(d,\ \varphi(N))$ as:

Let $p = 23$ and $q = 17 \Rightarrow N = pq = 391$

$\varphi(N) = (p-1)(q-1) = 22 \times 16 = 352$

with $e = 7,$ then compute the decryption exponent d, such that $ed - k\varphi(N) = 1$, with $e = 7$, $\varphi(N) = 352$ to have $d = 151$.

Thus, Public key $(e,\ N)$ is $(7,\ 391)$, Private key (d, $\varphi(N)$) is $(151,\ 352)$

Table 4: Converted ciphertexts C_H^* to RRS

ASCII	031	067	110	047	122	065	007	057	031	116	058	028	005
Chr	***US***	C	n	/	z	A	**BEL**	9	***US***	t	:	***FS***	***ENQ***
RRS	062	134	218	094	244	131	014	113	062	232	116	058	011
ASCII Value	027	100	034	035	121	080	004	125	042	115	053	078	065
Chr	***ESC***	d	"	#	y	P	**EOT**	}	*	s	5	N	A
RRS	056	199	071	073	242	161	008	251	086	229	106	157	131
ASCII Value	025	042	094	024	102	101	030	114	020	099	000	072	028
Chr	**EM**	*	^	**CAN**	j	K	**RS**	r	0	M	**NUL**	H	**FS**
RRS Value	052	086	188	049	211	212	061	227	043	197	001	146	058
ASCII Value	113	127	045	015	093	078	065	035	124	101	110	123	039
Chr	q	**DEL**	−	**SI**	]	N	A	#	|	}	R	{	'
RRS Value	226	254	091	031	184	157	131	073	248	251	218	247	079

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XII Issue X October 2025

From Table 4, the new ciphertexts becomes:

$C_H^{**} =$
062134218094244131014113062232116058011056199071073242161008251086229106157131052086188049211212061227043197001146058226254091031184157131073248251218247079

Finally, encrypt ciphertexts C_H^{**} using the public key to obtain final ciphertexts C_R^* as: $C_R^* = (C_H^{**})^7 (mod\ 391)$ that is

$$062^7 \ \ mod\ 391 = 156$$

$$134^7 \ \ mod\ 391 = 314$$

$$\vdots$$

$$079^7 \ \ mod\ 391 = 037$$

To have final ciphertexts:

$C_R^* =$156314329036065041295309156105346131122129126266146200253219378273321387268041256273307025284270379320274124001311131061254252380023268041146328378329342037

Decryption process

Reverse the process of the above encryption process to recover the original message.

SUMMARY AND CONCLUSION

This study introduces an improved hybrid cryptosystem that inserts a Reduced Residue System (RRS) substitution between Hill and RSA. The implemented pipeline is Transposition → Caesar → Hill → RRS → RSA (baseline omits RRS). Using modulus M = 255, Hill outputs (mapped to ASCII indices) are replaced by residues from a secret RRS mapping, severing direct numeric correlation with ASCII before RSA encryption. Even if RSA is compromised, an adversary recovers only RRS-coded residues, forcing a combinatorial mapping reconstruction rather than direct plaintext recovery. The scheme was tested by encrypting and decrypting a short message; empirical results show the RRS layer increases attacker workload and enhances key-generation/management, security, performance, and overall efficiency—producing a more robust hybrid encryption solution.

REFERENCES

1. Ariffin, S., Wijonarko, D., Suwarno, & Kristianto, E. S. (2024). Application of unimodular Hill cipher and RSA methods to text encryption algorithms using Python. Journal of Computer Science, 20(5), 548–563. https://doi.org/10.3844/jcssp.2024.548.563
2. ASCII Table. (2024). Google Search. https://www.google.com/search?q=ascii+table&client=ms-opera-mini-android&channel=new&tpsf=opiminienpf
3. Hardy, G. H., & Wright, E. M. (2008). An introduction to the theory of numbers (6th ed.). Oxford University Press.
4. Hassan, A., Garko, A., Sani, S., Abdullahi, U., & Sahalu, S. (2023). Combined techniques of Hill cipher and transposition cipher. Journal of Mathematics Letters, 1(1), 57–64. https://www.scipublications.com/journal/index.php/jml/article/view/822
5. Ibrahim, A. A., Muhammad, A. H., Shehu, S., Abubakar, T. U., Zaid, I., & Bello, U. (2021). Cryptanalysis on RSA using decryption exponent. IOSR Journal of Mathematics (IOSR-JM), 17(5), 1–8.
6. Khan, H. K., Pradhan, R., & Chandavarkar, B. R. (2021). Hybrid cryptography for cloud computing. In 2nd International Conference for Emerging Technology (INCET) (pp. 1–5).

https://doi.org/10.1109/INCET51464.2021.9456210

7. Manna, S., Prajapati, M., Sett, A., Banerjee, J., and S. Dutta, (2017). Design and implementation of a two-layered hybrid cryptosystem. Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), 327–331, DOI: 10.1109/ICRCICN.2017.8234529
8. Prakash, K., Saeed, Q. Y. A., Rajan, J., Mohammad, H., & Ahmed, A. S. M. (2023). A hybrid encryption system for communication and financial transactions using RSA and a novel symmetric key algorithm. Bulletin of Electrical Engineering and Informatics, 12(2), 1148–1158. https://doi.org/10.11591/eei.v12i2.4967
9. Rosen, K. H. (2011). Elementary number theory and its applications (6th ed.). Addison-Wesley Publishing Company, Pearson.
10. Rufa'i, A., Balarabe, A. T., Muazu, I., & Sirajo, M. (2020). Formulation of an improved hybrid cipher system. International Journal of Innovative Science and Research Technology, 5(12).
11. Saja, M. S., Zaynab, A. A., & Jaafer, A. H. (2023). Proposed hybrid cryptosystems based on modifications of Playfair cipher and RSA cryptosystem. Baghdad Science Journal. https://doi.org/10.21123/bsj.2023.8361
12. Shehu, S., Abdullahi, H., Ibrahim, A. A., & Ahmad, R. (2023). Breaking modulus of the form $N = p^r q^s$ with improved polynomial attacks. Journal of Advances in Mathematics and Computer Science, 38(8), 33–46. https://doi.org/10.9734/JAMCS/2023/v38i81788
13. Suhasini, C. A., & Bushra, S. N. (2021). Securing of cloud data with duplex data encryption algorithm. In 5th International Conference on Computing Methodologies and Communication 252–256. https://doi.org/10.1109/ICCMC51019.2021.9418247
14. Susmitha, C., Kumar, S. K., & Bulla, S. (2023). Hybrid cryptography for secure file storage. In 7th International Conference on Computing Methodologies and Communication (ICCMC). https://doi.org/10.1109/ICCMC56507.2023.10084073.
15. Yao, F. and Su, J., (2021). Hybrid Encryption Scheme for Hospital Financial Data Based on Noekeon Algorithm. Security and Communication Network. DOI: 10.1155/2021/7578752.