Strengthening the Protection of Sensitive Data with Modified Rsa Encryption

Karthikeyan N[1] Hariharan S[2] Amirtharaj S[3] Suneelsree N[4] Abinash C[5]
[1]Assistant Professor, Department of CSE, Government College of Engineering, Sengipatti, Thanjavur, Tamilnadu
[2,3,4,5] Third Year CSE, Department of CSE, Government College of Engineering, Sengipatti, Thanjavur, Tamilnadu

Corresponding Author: nkarthikeyan@gcetj.edu.in

ABSTRACT

In the contemporary digital realm, safeguarding confidential information is vital to ensuring effective communication. Upholding the confidentiality and integrity of exchanged data is imperative. Cryptography emerges as a pivotal asset in this endeavor, with a proposed focus on utilizing a modified RSA algorithm to bolster communication security. The intended model aims to enhance security within communication systems. The modified RSA algorithm introduces improvements to the traditional RSA encryption scheme, aiming to enhance both its security and performance. These enhancements typically involve adjustments to key generation, encryption, or decryption processes. The objective is to bolster resilience against various cryptographic attacks while maintaining compatibility with existing RSA-based systems. By enhancing encryption techniques in digital communication and data security applications, the modified RSA algorithm offers a promising avenue for fortifying data protection measures. This approach not only strengthens confidentiality but also extends the time required for potential cryptanalysis, thereby reinforcing overall security protocols.

Keywords: Communication, Network Security, RSA Algorithm, Modified RSA Algorithm

1. INTRODUCTION

In today's interconnected global landscape, where digital data serves as the cornerstone of economies, societies, and individual lives, the demand for robust security measures has reached unprecedented levels. The rise in digital transactions, communication channels, and data exchanges necessitates a steadfast commitment to ensuring the confidentiality, integrity, and authenticity of information. At the forefront of this critical mission lies cryptography, a field dedicated to developing essential tools and techniques for safeguarding sensitive data from unauthorized access and manipulation. Cryptography, often hailed as the bedrock of modern cybersecurity, operates as the science behind secure communication. By leveraging mathematical algorithms and cryptographic primitives, cryptography empowers the encryption of data, rendering it indecipherable to anyone lacking the proper decryption key. From securing online transactions to protecting governmental communications, cryptography plays a pivotal role across various domains of digital security.

Nevertheless, as technology progresses and cyber threats evolve in sophistication, the reliability of traditional cryptographic methods faces mounting scrutiny. The advent of quantum computing and the relentless march of technological advancements pose new challenges to the resilience of established cryptographic standards. Particularly, the enduring efficacy of the RSA algorithm, a stalwart in the realm of cryptography, is now subject to questioning. The rapid escalation in computational power, coupled with emerging vulnerabilities in cryptographic systems, underscores the imperative for a reassessment of conventional approaches to digital security. In light of these developments, there arises a pressing need for the cybersecurity community to explore avenues for enhancing the RSA algorithm's resistance against emerging threats. By contextualizing the imperative for RSA algorithm modification within the broader landscape of cybersecurity, we aim to chart a path towards a more resilient and secure digital ecosystem. This endeavor underscores the critical importance of adaptive strategies in confronting the evolving challenges of digital security, ensuring the continued protection of sensitive information in an increasingly interconnected world.

2. PROPOSED MODEL

The proposed model improves the security of the confidential messages ensuring the confidentiality, integrity, and authenticity of information.

2.1 Modified RSA Algorithm

Utilizing a variety of cryptographic techniques enhances the security of message transmission by ensuring confidentiality. Asymmetric encryption involves the use of distinct key pairs by the sender and receiver to achieve this confidentiality. The RSA algorithm exemplifies asymmetric encryption, where the ciphertext of a text message is created using the sender's public key and decrypted using their corresponding private key pair. The proposed modification to the RSA algorithm is given below.

2.1.1 Key Generation Phase:

a. Generate any identical random numbers a, b, c, d and e of within the range of 1024 bits.
b. Find n = a *b*c*d*e
c. Find sp = (a-1) * (b-1) * (c-1) * (d-1) * (e-10
d. Choose the random co-prime number (e) for sp where GCD(e, sp) =1 and e>1
e. Find 'd' where (d*e) mod sp = 1 mod sp.

2.1.2 Encryption Phase

Confidential messages undergo encryption through the expression $C = M^e \bmod n$, where M is the confidential message, C represents the cipher text and (e,n) represent the public key pair.

2.1.3 Decryption Phase

At the receiver's end, the decryption phase is executed. Here, the receiver retrieves the original confidential messages from the received ciphertext utilizing the expression $M = C^d \bmod n$, where C denotes the ciphertext, M represents the original message, and (d,n) signifies the private key pair used for decryption.

3. RESULTS AND DISCUSSION

The proposed model has been implemented using Python 3.10 on a Windows 10 operating system. It has undergone rigorous testing across various input sizes to evaluate its performance. The results of these tests are presented in Table 1, Table 2, and Table 3. Table 1 illustrates the encryption and decryption times required for different input sizes, utilizing fixed sets of five random integers paired with distinct public and private key combinations. Analysis of Table 1 reveals that the robustness of the proposed model is contingent upon the selection of the public key, particularly in scenarios where an intruder attempts to predict the random integers. In Table 2, encryption and decryption times are depicted for an input size of 5.09KB, varying the random integers and corresponding public-private key pairs. The findings presented in Table 2 underscore the critical role played by the selection of both public and private key pairs in determining the strength of the cryptographic system. Moreover, it highlights that the resilience of the algorithm remains intact even in situations where an intruder may successfully predict the random integers, reaffirming the significance of judiciously chosen key pairs.

Table 1. Execution time of Analysis of proposed model for same random numbers of different private key and public key pairs with different key size

Five random prime numbers [19,29,31,11,3]					
File Size (in KB)	**Length of the File**	**Public Key**	**Private Key**	**Encryption Time (in ms)**	**Decryption Time (in ms)**
0.68	695	150197	16733	13.8785	1.4793
1.05	1081	166609	104689	14.9487	17.2592
1.52	1559	251939	90059	59.8423	17.5088
3.04	3118	171029	92669	44.4002	36.1248
3.53	3620	65951	137951	20.6419	62.4231
4.04	4146	280717	78853	100.9902	38.8229
4.74	4862	206651	123251	79.745	71.1291
5.09	5221	120569	283529	53.092	185.5333
5.56	5699	199673	52937	99.0738	38.0045
16.7	17098	219761	57041	558.4925	192.4033

Table 2. Execution time of Analysis of proposed model for different random numbers with fixed input size

File size = 5.09 KB, length of the file = 5996 characters								
Random Prime Numbers within range 2 to 24					**Public Key**	**Private Key**	**Encryption Time (in ms)**	**Decryption Time (in ms)**
Number 1	**Number 2**	**Number 3**	**Number 4**	**Number 5**				
19	3	5	11	7	8537	5033	2.0356	2.2216
19	5	7	13	11	22091	48611	1.6877	28.7539
13	3	7	17	5	2593	6625	20.9669	3.5351
19	5	23	7	3	7481	16457	38.8581	8.5641
13	5	11	17	7	24443	16307	5.1215	9.7501
11	13	17	5	3	2609	7889	12.1538	3.6636
3	17	11	7	19	1399	21319	3.2089	13.2574
3	19	11	13	17	2087	48023	3.9475	32.5005
13	11	23	7	19	166541	223301	226.9505	156.6151
7	19	5	3	11	4357	3853	8.6934	1.6117

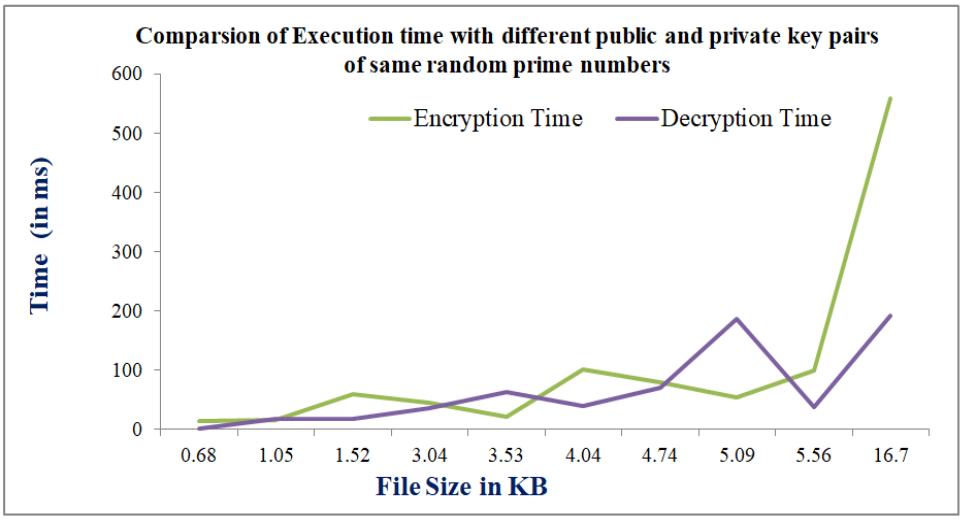
Table 3. Execution time of Analysis of proposed model with different random numbers of different private and public key pairs with different input size

File Size (in KB)	Length of the File	Random Prime Numbers range(2,24)					Public Key	Private Key	Encryption Time (in ms)	Decryption Time (in ms)
		Number 1	Number 2	Number 3	Number 4	Number 5				
0.68	695	7	23	3	13	19	30089	21497	1.7459	2.0356
1.05	1081	17	13	5	3	11	1031	13751	0.1741	1.6877
1.52	1559	19	17	23	3	7	50873	50825	12.8321	20.9669
3.04	3118	23	17	3	19	13	23857	47569	13.3347	38.8581
3.53	3620	7	19	13	17	11	31973	5357	21.4223	5.1215
4.04	4146	17	11	13	7	5	3391	12991	2.6452	12.1538
4.74	4862	11	3	17	19	13	7309	2629	6.4172	3.2089
5.09	5221	5	7	3	13	19	4567	4327	4.3041	3.9475
5.56	5699	13	19	17	23	3	151141	151405	156.2111	226.9505
16.7	17098	19	5	7	3	11	7793	2897	23.9836	8.6934

Table 4. Comparison of Execution Time Analysis of Modified RSA algorithm with standard RSA Algorithm

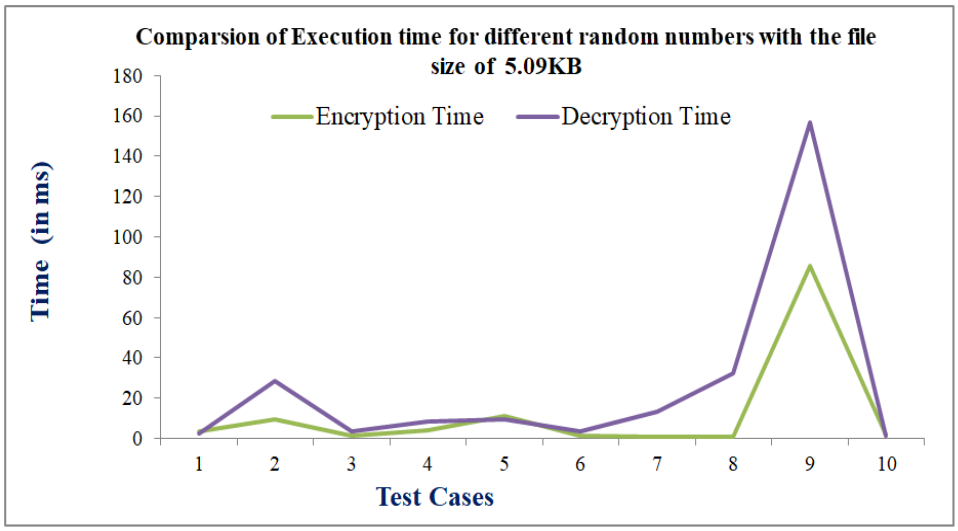
File Size (in KB)	Length of the File	Modifed RSA Algorithm				Standard RSA Algorithm			
		Public Key	Private Key	Encryption Time (in ms)	Decryption Time (in ms)	Public Key	Private Key	Encryption Time (in ms)	Decryption Time (in ms)
0.68	695	30089	21497	1.7459	2.0356	8009	10889	0.3443	0.4536
1.05	1081	1031	13751	0.1741	1.6877	151	31	0.0035	0.1328
1.52	1559	50873	50825	12.8321	20.9669	22817	14153	2.1416	1.3442
3.04	3118	23857	47569	13.3347	38.8581	331	151	0.062	0.031
3.53	3620	31973	5357	21.4223	5.1215	42967	15247	9.5488	3.7346
4.04	4146	3391	12991	2.6452	12.1538	2141	2285	0.5313	0.5627
4.74	4862	7309	2629	6.4172	3.2089	18341	26909	5.5326	8.5826
5.09	5221	4567	4327	4.3041	3.9475	12821	8381	4.3757	2.735
5.56	5699	151141	151405	156.2111	226.9505	4271	1163	1.5632	0.4069
16.7	17098	7793	2897	23.9836	8.6934	12809	41369	14.146	51.1391

Table 3 provides a comprehensive analysis of execution times for various combinations of random pairs and key value pairs across different input sizes. Additionally, Table 4 presents the execution time analysis of the proposed model in comparison to a standard algorithm. The findings from Table 4 indicate that the proposed model requires more time for both encryption and decryption processes. However, it is important to note that the strength of the proposed model lies in the size of the key value pairs. Unlike conventional RSA algorithms, where key value pairs are predetermined, in the proposed method, they are determined by the selection of random numbers.



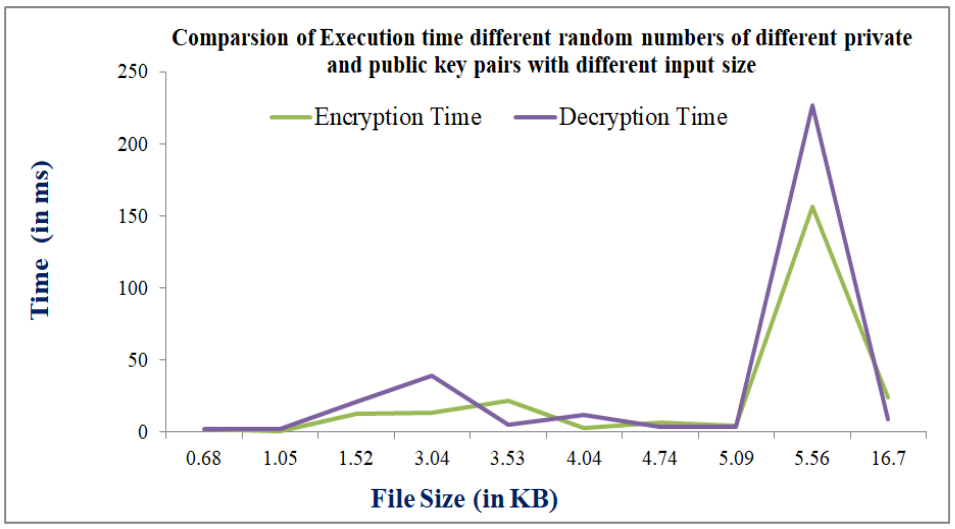
File Size	Public Key	Private Key
0.68	150197	16733
1.05	166609	104689
1.52	251939	90059
3.04	171029	92669
3.53	65951	137951
4.04	280717	78853
4.74	206651	123251
5.09	120569	283529
5.56	199673	52937
16.7	219761	57041

Figure 1.1 (a) Comparison of Execution time analysis of the proposed model for same random numbers of different key pairs with different input size



Case	Public Key	Private Key
1	8537	5033
2	22091	48611
3	2593	6625
4	7481	16457
5	24443	16307
6	2609	7889
7	1399	21319
8	2087	48023
9	166541	223301
10	4357	3853

Figure 1.1 (b) Comparison of Execution time analysis of the proposed model for same random numbers of different key pairs with fixed input size



Case	Public Key	Private Key
1	30089	21497
2	1031	13751
3	50873	50825
4	23857	47569
5	31973	5357
6	3391	12991
7	7309	2629
8	4567	4327
9	151141	151405
10	7793	2897

Figure 1.1 (c) Execution time of Analysis of proposed model with different random numbers of different private and public key pairs with different input size

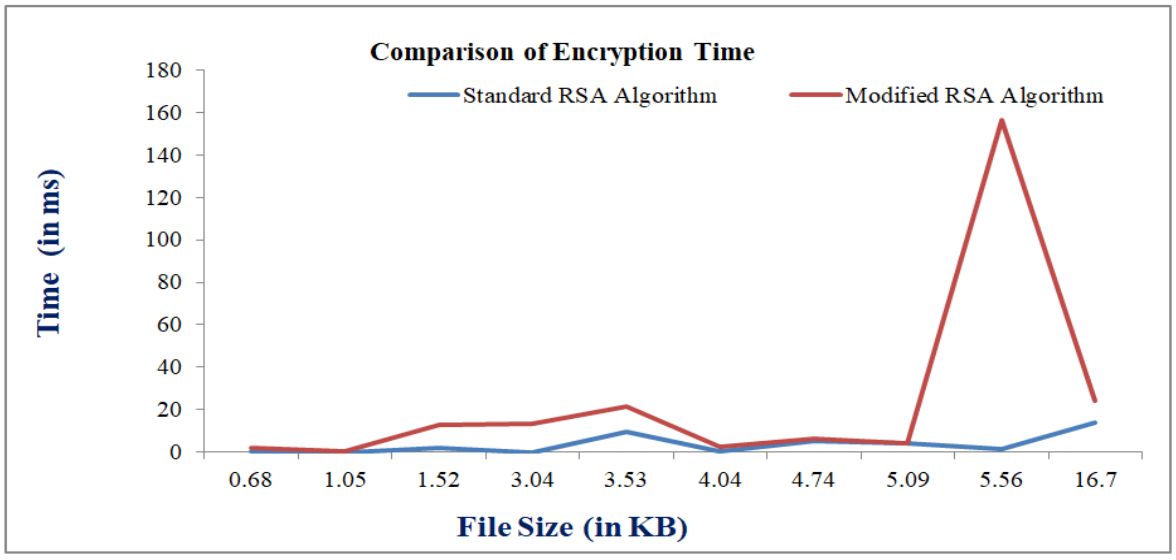


Figure 1.1 (d) Comparison of Encryption time Analysis of proposed Model with standard RSA Algorithm

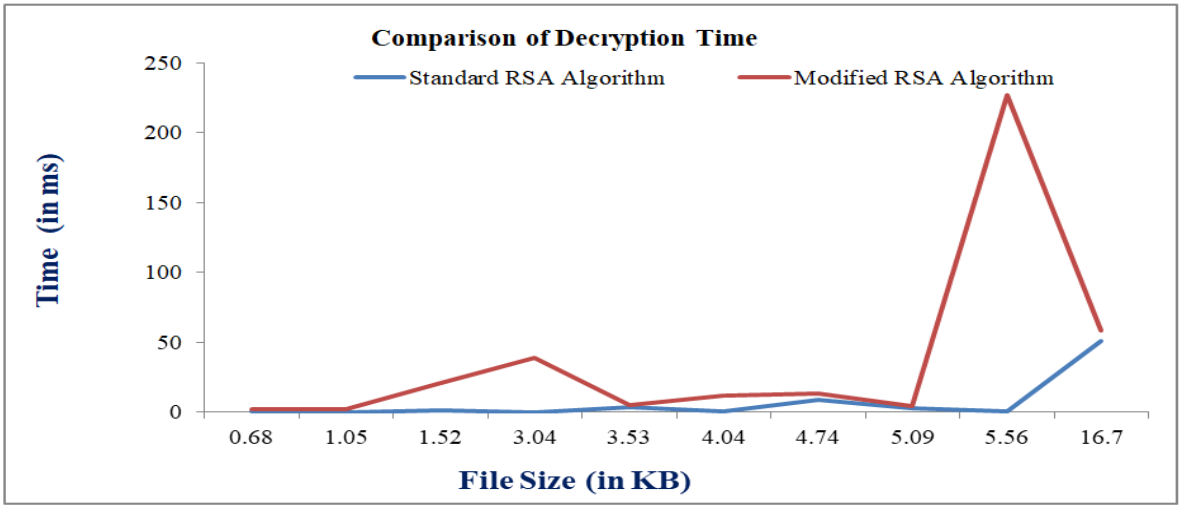


Figure 1.1 (e) Comparison of Decryption time Analysis of proposed Model with standard RSA Algorithm

The probability of selecting different combinations of five random numbers in the proposed method is significantly lower compared to conventional RSA algorithms. This unique feature enhances the security of the proposed model by introducing additional complexity and reducing the likelihood of successful attacks based on predicting key values. These results emphasize the robustness of the proposed model and underscore the importance of meticulously selecting key pairs to mitigate the risks associated with potential intruder attacks. The performance analysis of the proposed model from various perspectives is visually depicted in Figure 1, where subfigures (a) to (e) offer insights into different aspects of the model's efficiency and effectiveness

CONCLUSION

The utilization of five random prime numbers in the proposed model significantly lowers the probability of selecting suitable key pairs, thereby increasing the complexity of cryptanalysis. The outcomes presented underscore the robustness of the proposed approach and highlight the critical role of carefully chosen key pairs in minimizing the vulnerabilities to potential intruder attacks. Looking ahead, integrating this model with additional techniques holds promise for bolstering security across various real-time applications.

REFERENCES

[1] Nasution, Nadia, Efendi, Syahril and Nasution, Sawaluddin, “Analysis of RSA Variants in Securing Message.” IOP Conference Series: Materials Science and Engineering, vol. 725, no. 1, p. 012131, Jan. 2020,doi:10.1088/1757-899x/725/1/012131.

[2] Amalarethinam, I. George, and H. M. Leena, “Enhanced RSA Algorithm With Varying Key Sizes for Data Security in Cloud.” 2017 World Congress on Computing and Communication Technologies (WCCCT), Feb. 2017, doi:10.1109/wccct.2016.50.

[3] Majumder, Sudipto, and Md Mahfuzur Rahman. "Implementation of Security Enhanced Image Steganography with the Incorporation of Modified RSA Algorithm." 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE). IEEE, 2019.

[4] Solak, Serdar, and U. M. U. T. Altınışık. "LSB Substitution and PVD performance analysis for image steganography." International Journal of Computer Sciences and Engineering 6.10 (2018): 1-4.

[5] Jabbar A, Lilhore PU, “Design and Implementation of Hybrid EC-RSA Security Algorithm Based on TPA for Cloud Storage”, International Journal Online Of Science, (2017), 3(11) pp. 6–26.

[6] MohapatraA K, Gupta N, Prakash N (2016) Step-Wise Calculation of Performance and Complexity Analysis of Safer with RSA Algorithm. University School of Information Technology.

[7] Adamu, Ismail Abdulkarim, Boukari Souley (2018) Performance Analysis of Text and Image Steganography with RSA Algorithm in Cloud Computing. International Journal of Software Engineering & Applications 9:65-76.

[8] Al-Kaabi, Shaheen Saad, Belhaouari, Samir Brahim (2019) Methods toward Enhancingr RSA Algorithm: A Survey. International Journal of Network Security & Its Applications. 11:3. http://dx.doi.org/10.2139/ssrn.3412776

[9] Khuma ZN (2019) Secure Data Transfer using RSA and Steganography. International Journal of Science and Engineering Applications 8:08:312-316.

[10]. Al-Kaabi, Shaheen Saad, Belhaouari, Samir Brahim (2019) Methods toward Enhancing RSA Algorithm: A Survey. International Journal of Network Security & Its Applications (IJNSA) 11:3. http://dx.doi.org/10.2139/ssrn.3412776