Hybrid Algorithm for Symmetric Based Fully Homomorphic Encryption

Kamaldeen Jimoh Muhammed[1][0000-0001-9258-3810], Rafiu Mope Isiaka[2], Ayisat Wuraola Asaju-Gbolagade[3], Kayode Sakariyah Adewole[3][0000-0002-0155-7949] and Kazeem Alagbe Gbolagade[2]

[1] Department of Educational Technology, University of Ilorin, Ilorin, Nigeria
[2] Department of Computer Science, Kwara State University, Malete, Nigeria
[3] Department of Computer Science, University of Ilorin, Ilorin, Nigeria

jimoh.km@unilorin.edu.ng, abdulrafiu.isiaka@kwasu.edu.ng, ayisatwuraola@gmail.com, adewole.ks@unilorin.edu.ng, kazeem.gbolagade@kwasu.edu.ng

Abstract. Fully Homomorphic Encryption (FHE) supports realistic computations on encrypted data and hence it is widely proposed to be used in cloud computing to protect the integrity and privacy of data stored in the cloud. The existing symmetric-based FHE schemes suffer from insecurity against known plaintext/ciphertext attacks and generate a large ciphertext size that required a large bandwidth to transfer the ciphertext over the network. To ameliorate these weaknesses is the aim of this paper. A hybrid algorithm for symmetric-based Fully Homomorphic Encryption that combines N-prime model and Matrix Operation for Randomization and Encryption with Secret Information Moduli Set (MORESIMS) is proposed. The results show that the proposed hybrid symmetric-based FHE framework satisfied homomorphism properties with robust inbuilt security that resists known plaintext, ciphertext and statistical attacks. The ciphertext size produced by the proposed hybrid framework is less than 4 times of its equivalent plaintext size with a considerable encryption execution time and fast decryption time. Thus, guaranteed to provide optimum performance and reliable solutions for securing integrity and privacy of user's data in the cloud.

Keywords: Homomorphic Encryption, Fully Homomorphic Encryption, Residue Number System, Symmetric Encryption, Hybrid Encryption, Cloud Computing.

1 Introduction

Cloud technology allows paradigm of data storage to be shifted from local storage to cloud storage which has brought tremendous improvement over the years. Despite numerous advantages offered by cloud technology in term of flexibility and cost-effectiveness, security and privacy have grown to be of significant concern in cloud technology and is discouraging factor for potential adopters [1]. For instance, unencrypted sensitive data on the cloud are vulnerable to different attacks such as unauthorized access to the server by the adversary, spy on secluded data by legitimate administrator at hosting providers and stealing or destruction of data by attacker with physical access to the server [1].

Traditional encryption schemes such as Data Encryption Standard (DES) and Advanced Encryption Standard (AES) have been used to address the issue of security and privacy facing the data stored in the cloud. However, cloud technology has advanced to the level of cloud computing where the stored data on the cloud are used for computation. In 2008, Cloud computing has evolved as a new revolution in information technology as it provides a variety of services and applications to be run from anywhere in the world to its users through the Internet, traditional encryption schemes do not fit enough to protect the integrity and privacy of data in cloud computing server as it requires the release of secret key prior the computation on the cloud server which threatens the privacy of such computation and the data [2].

Consequently, Homomorphic Encryption (HE) was introduced as an encryption scheme that allows computation to be performed on encrypted data without the use of secret key before the computation. This scheme was proposed by Rivest, et al. [3] aliased as Partial Homomorphic Encryption (PHE) due to its support in either additive or multiplicative homomorphic operation not until Gentry [4] introduced Somewhat Homomorphic Encryption (SHE) scheme that allows both additive and multiplicative operations with limited number of addition and multiplication. He further proposed Fully Homomorphic Encryption (FHE) scheme that allows unlimited number of arbitrary functions that can be evaluated on an encrypted data [5].

FHE schemes were categorized as asymmetric and symmetric cryptosystem as shown in Fig. 1. Asymmetric approach as in [5-15] uses public and private keys for encryption and decryption respectively. This has been proved to be secured [16], however, the scheme is far from being practicable because of its large computational cost and large ciphertexts generation. Since then, though efforts have been made to devise more efficient schemes. Most of the asymmetric based FHE schemes still have large ciphertexts (millions of bits for a single ciphertext). This presents a considerable bottleneck in the practical deployments [17].

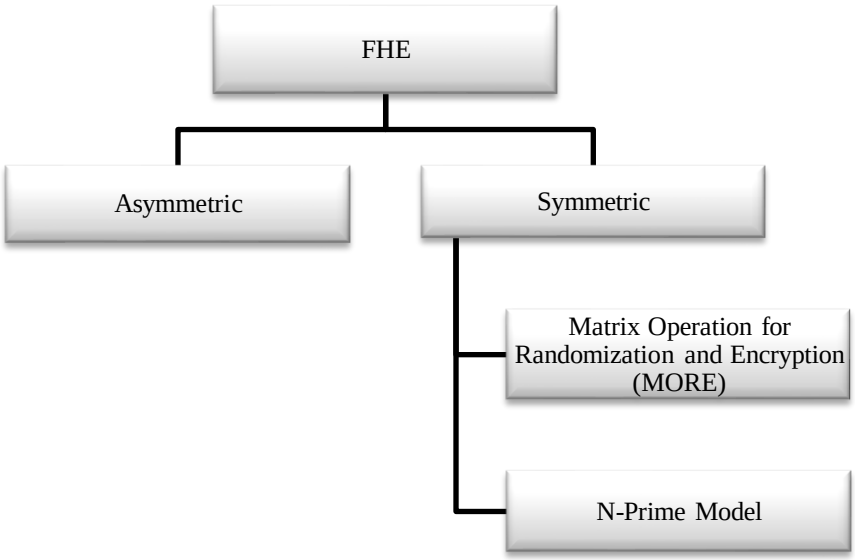


Fig. 1. Taxonomy of FHE

Symmetric approach on the other hand, as in [18-26] uses a single private or secret key for both encryption and decryption. It has its strength in low computational overhead and practical consideration for real-world deployments. However, some of these symmetric based FHE schemes such as [24, 26-29] still suffer from either large ciphertext expansion or low immunity against security attack as argued in [25, 30].

To ameliorate the drawbacks, such as low immunity against security attack, ciphertext expansion, encryption and decryption time; hybrid symmetric-based FHE is proposed

which comprises of two layers. The top layer enhances N-prime model proposed in Mohammed and Abed [24] with Residue Number System (RNS) to address ciphertext expansion problem. The bottom layer is based on Enhanced Matrix Operation for Randomization and Encryption (EMORE) as presented in Muhammed and Gbolagade [31] to further strengthen the security of the proposed hybrid framework.

The rest of this paper is organized as follows. Section 2 Discusses N-prime Model and EMORE Approaches. Section 3 Describes the Research Method in line with Modified N-prime Model hybridize with EMORE. Results Findings and Discussion are given in Section 4. Conclusions to the study are drawn in Section 5.

2 N-prime Model and EMORE approach

The two major approaches involve in this study are discussed as follow:

2.1 N-prime Model

In Mohammed and Abed [24], a new mathematical model was developed and designed to achieve symmetric-based FHE scheme that supports additive and multiplicative homomorphism called N-prime model. The security of this model depends on the problem of factorization of integers to their primary numbers. It operates in a way that, instead of dealing with two prime numbers, it is expanded to deal with n prime numbers, this forms the bases for the security of the presumptive algorithm to be more efficient in the face of the security challenges facing cloud computing. The N-prime model operates in three stages which are Key Generation, Encryption and Decryption.

The proof for both additive and multiplicative homomorphism properties for this model was extensively discussed in [24]. However, the encryption stage of this model generates excessively big ciphertext which serves as the main drawback of this model due to the multiplication of big prime number *Sk*, *n* (product of numerous prime numbers) and random number *r* that also serve as noise to the ciphertext. However, the drawback of N-prime model has been addressed in [32].

2.2 EMORE Approach

In Muhammed and Gbolagade [31], Matrix Operation for Randomization and Encryption(MORE) was enhanced using Secret Information Moduli Set (SIMS) to preserve the privacy of a given data in addition to the secret key and the scheme was called MORESIMS. This scheme converts the plaintext in a ring Z_n to residues ($x_1, x_2, \ldots\ldots x_k$) using SIMS $\{m_i\}_{i=1,k}$, the principal diagonal of the matrix was formulated based on residues and send it to encryption block where MORE-like encryption ($K^{-1} * M * K$) was carried out in conjunction with secret invertible matric K to generate the ciphertext.

MORESIMS scheme was proofed to be symmetric-based FHE and satisfy additive and multiplicative homomorphism properties. The algorithms for encryption, decryption and other different stages involved in MORESIMS scheme are extensively discussed in [31].

3 Research Method

The proposed hybrid scheme is built on the modified N-prime model as discussed in [32] and MORESIMS approach in [31] without altering their homomorphic properties. The steps are illustrated in Fig. 2 and explained as follow:

3.1 Encryption Process

Using SIMS approach in Stage 2 of Fig. 2, parameters such as plaintext (x), random number (r), big integer (l) and N-prime (np) at Stage 1 of Fig. 2 will be converted to residues using Residue Number System (RNS) forward conversion module which addresses ciphertext expansion problem in N-prime model and performs the first encryption based on RNS arithmetic operation with respect to SIMS as shown in Stage 1 of Fig. 2. Thus, the result of stage 1 produces ciphertext (ct).

$$Ciphertext\ ct = \{x_1, x_2, x_3, \dots x_k\}_{mi} + \{r_1, r_2, r_3, \dots r_k\}_{mi} * \{l_1, l_2, l_3, \dots l_k\}_{mi} * \{np_1, np_2, np_3, \dots np_k\}_{mi} \quad (1)$$

$$ct = \{ct_1, ct_2, ct_3, \dots ct_k\}_{mi} \quad (2)$$

where $ct1, ct2, ct3, \dots ctk$ are ciphertext residues based on modified N-Prime model and $mi\ \forall\ i = 1, k$ is the moduli set.

Stage 2 of the framework in Fig. 2 receives the output of Stage 1 which is equation (2) as an input to form the principal diagonal of matrix M $(M = diag(ct))$ as shown in equation (3) and pass it to MORE-like encryption block.

$$M = \begin{bmatrix} m & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & r \end{bmatrix} = \text{diag(ct)} = \begin{bmatrix} ct_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & ct_k \end{bmatrix} \quad (3)$$

The second encryption at this stage requires the generation of Secret Invertible Matrix (K) and find K^{-1} to perform encryption operation as shown in equation (4). Thus, $E(m, k)$ as an output with ciphertext C, will be sent to cloud server for storage and to perform homomorphic computation on the ciphertext as presented in Stage 3 of Fig. 2.

$$E(m, k) = K^{-1} \begin{bmatrix} ct_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & ct_k \end{bmatrix} K = C \quad (4)$$

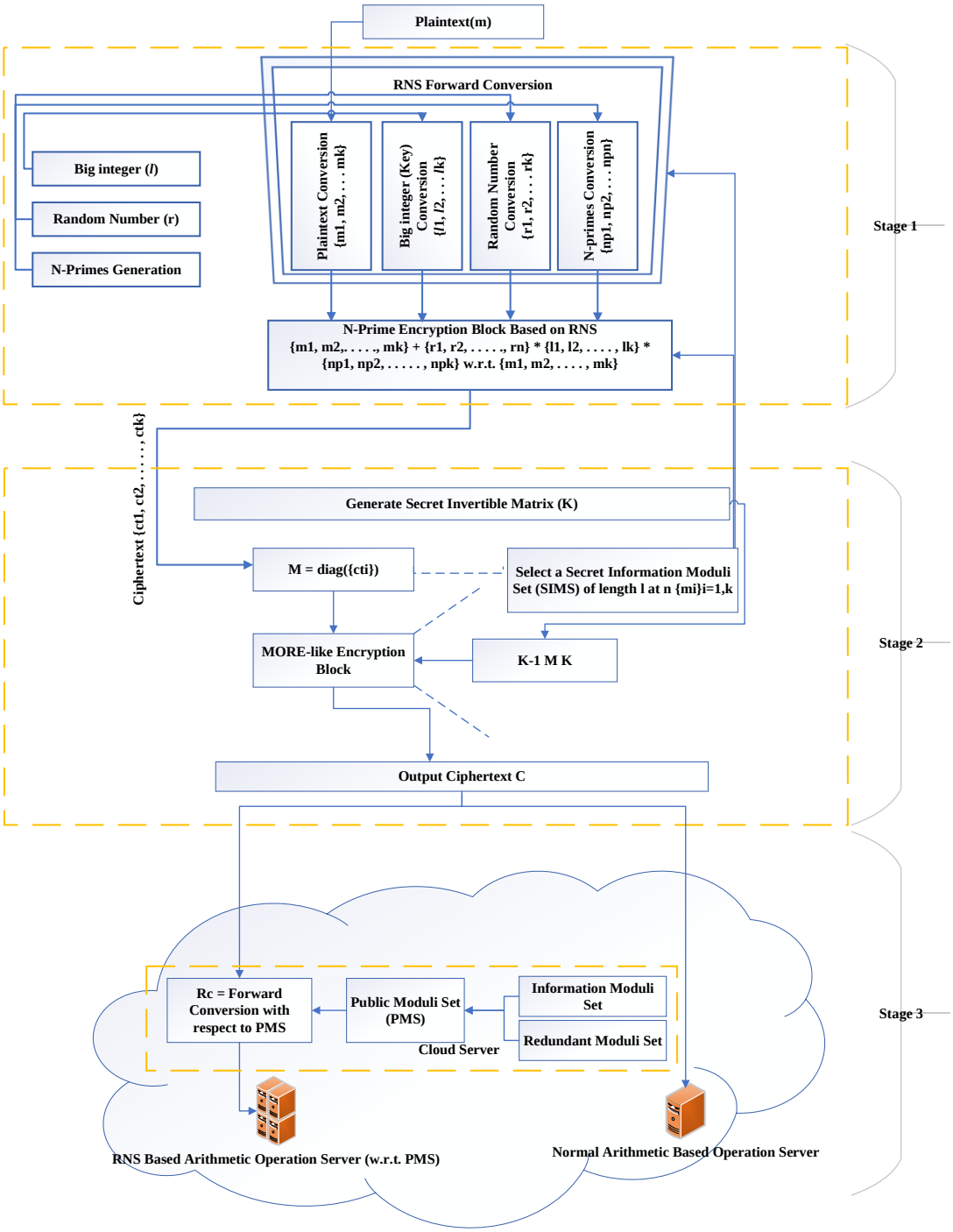


Fig. 2. Hybridized Symmetric Based FHE Framework

The operations in Stage 3 of Fig. 2, shows that homomorphic computation can be done in two ways in the cloud. The first one is through Normal Arithmetic Based Operation Cloud Server (NABOCS) shown at the right-hand side where the cloud server receives the output of stage 2 together with evaluation function $(f(x, C))$ provided by the user and perform homomorphic operation (addition and multiplication) based on the normal arithmetic operation. The second one is through RNS Based Arithmetic Operation Cloud Server (RNS-BAOCS) at the left-hand side of Stage 3 where further optimization is provided for homomorphic bit operation. This optimization stage requires the use of Public Moduli Set (PMS) that comprises of Information Moduli Set (IMS) and Redundant Moduli Set (RMS) to convert the ciphertext C received from stage 2 to residues using RNS Forward Conversion Module with respect to PMS and perform homomorphic operation (evaluation function $(f(x, C))$) on the residues with respect to PMS in parallel with no carry propagations using RNS based arithmetic operations.

3.2 Decryption Process

The inverse of encryption process is simply the decryption process. User with the following secret parameters can decipher the result of homomorphic operation as shown in Fig. 3:

i. Invertible Matrix (k)
ii. Secret Key (l)
iii. SIMS and
iv. Value of n

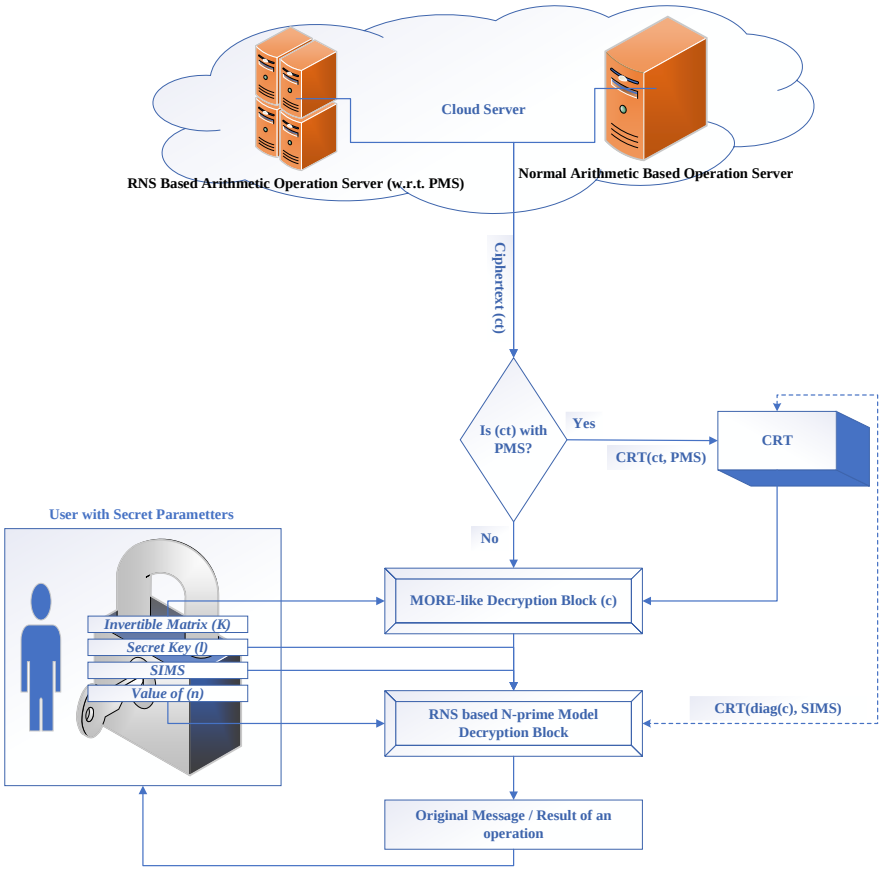


Fig. 3. Decryption Model

Depending on which side of the server (either NABOCS or RNS-BAOCS) is ciphertext coming from, the decryption model directs the ciphertext (ct) into CRT module if it contains PMS before sending it to MORE-like decryption block otherwise it goes straight down to MORE-like decryption block where it requires secret invertible matrix (k) from the user to decrypt $c \leftarrow ct$ and pass c together with a secret key (l), $SIMS$ and n from the user's end to RNS based N-prime decryption block where $diag(c)$ form the residues which need to be converted using CRT with respect to $SIMS$ and n to get its equivalent weighted number (wn) and finally compute $|wn|_{Secret\ Key\ (l)}$ as the original message or result of an operation as summarized in ***Table*** 1.

Table 1: Decryption Algorithm

Stages in Decryption Process
1: $Input \ \rightarrow Dec_{ct}(ct, k, l, sims, n, pms = "")$
2: $Check\ if\ (pms\ != "")\ then$
$Store\ the\ output\ of\ CRT(ct, pms) \rightarrow ct$
3: $Compute\ MORE - Like\ Decryption\ (c \leftarrow k \times ct \times k^{-1})$
4: $Compute\ RNS - Based\ N - Prime\ model\ Decryption$
$(wn = CRT(diag(c), sims_n)$
5: $Compute\ \lvert wn\rvert_l\ and\ return\ the\ result\ as$
$Original\ Message\ or\ the\ Result\ of\ the\ Operation$

4 Results and Discussion

The cryptographic strengths of the proposed hybrid symmetric-based FHE framework were investigated. The framework is subjected to a series of tests to affirm its efficiency and security standard in term of uniformity, randomness, and key sensitivity. In addition, the encryption and decryption execution time of the proposed hybrid framework is measured and compared to MORE algorithm and N-prime model. The proposed hybrid framework has been verified on different file sizes and it indicates that the proposed hybrid framework has better performance for encrypting and decrypting different file sizes as shown in Table 2, which illustrates the encryption time measured in millisecond together with items count in each file. More so, Table 2 reveals that decryption process of the proposed hybrid framework is faster than its encryption process. This was simulated using Python programming language on Ubuntu 18.04 64-bit operating system, Intel Core i7 processor and 8GB RAM.

Table 2: Test of Hybrid Framework on different File Sizes

Plaintext File Size (KB)	**Character Count**	**Encryption Time (Ms)**	**Decryption Time (Ms)**
1	458	38.12	23.81
2	916	78.71	47.98
4	1832	148.19	97.34
8	3664	305.59	195.52
16	7328	583.47	419.79
32	14656	1183.75	784.53
64	29312	2333.04	1554.43
128	58624	4630.17	3104.95
256	117248	9273.98	6212.47
512	234496	18666.11	12755.58

Furthermore, Table 3 shows ciphertext expansion ratio for each plaintext files that was investigated and findings show that averagely for any given plaintext size, proposed

hybrid framework generates ciphertext size that is less than 4 times of its original plaintext size. In comparison with existing studies, Table 4 shows that the proposed hybrid symmetric based FHE framework outperforms the existing state-of-art schemes in term of reduction in ciphertext size.

Table 3: Size Ratio (Ciphertext Size / Plaintext Size)

Plaintext File Size (KB)	**Ciphertext Size (KB)**	**Size Ratio (Plaintext : Cipher-text)**
1	3.68	1 : 3.68
2	7.8	1 : 3.90
4	16.18	1 : 4.05
8	29.42	1 : 3.68
16	60.0	1 : 3.75
32	122.0	1 : 3.81
64	247.7	1 : 3.87
128	502.52	1 : 3.93
256	1019.11	1 : 3.98
512	1836.73	1 : 3.59
Average Size Ratio		**1 : 3.82**

Table 4: Comparing the proposed Hybrid Framework with related studies.

Authors (Year)	**Size Ratio (Plaintext : Ciphertext)**
Chillotti, et al. [10]	1: 400,000
Mohammed and Abed [24]	1: 10.21
Proposed Hybrid framework	1: 3.82

In addition, the proposed hybrid framework was compared with MORESIMS model and RNS based N-Prime model. The results show that approximately, there is no significant difference in encryption execution time between the proposed hybrid framework and MORESIMS model as shown in Fig. 4. However, Fig. 5 shows that there is a slightly difference in encryption execution time between the proposed hybrid framework and RNS based N-prime model while Fig. 6 reveals that RNS based N-prime model took the least encryption execution time in compared with proposed hybrid framework and MORESIMS model because it involved single stage of encryption and RNS arithmetic operations were used in the encryption process of RNS based N-prime model with carry free propagation and parallelism.

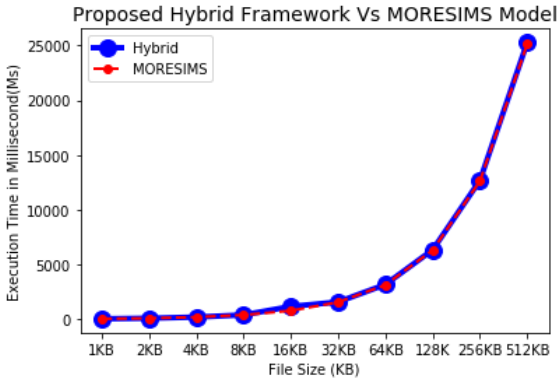


Fig. 4. Encryption execution time between proposed Hybrid Framework and MORESIM Model

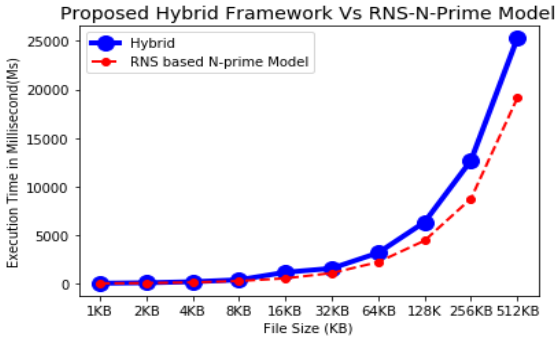


Fig. 5. Encryption execution time between proposed Hybrid Framework and RNS based N-Prime Model

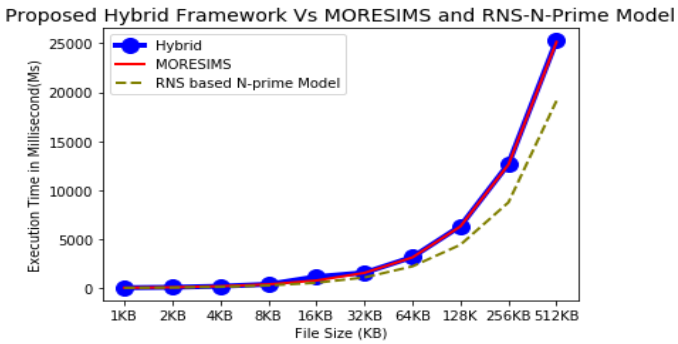


Fig. 6. Comparing the proposed Hybrid Framework with MORESIMS Model and RNS based N-Prime Model

For the security of the proposed hybrid framework, correlation coefficient, uniform distribution and key sensitivity was used to measure its security strength against attacks.

Correlation Coefficient: for any strong encryption scheme to be achieved, among the important requirement is that the ciphertext should be greatly different from its plaintext form and the correlation coefficient between the plaintext and ciphertext should be kept so low. Thus, the correlation coefficient between the plaintext and ciphertext is computed as follows:

$$\rho_{x,y} = \frac{cov(x,y)}{\sqrt{D(x)\times D(y)}} \tag{5}$$

$where\ \ cov(x,y) = E[\{x - E(x)\}\{y - E(y)\}],$
$E(x) = \frac{1}{n} \times \sum_{k=1}^{n} x_i,\ \ E(y) = \frac{1}{n} \times \sum_{k=1}^{n} y_i,\ \ D(x) = \frac{1}{n} \times \sum_{k=1}^{n} \{x_i - E[x]\}$ and $D(y) = \frac{1}{n} \times \sum_{k=1}^{n} \{y_i - E[y]\}$

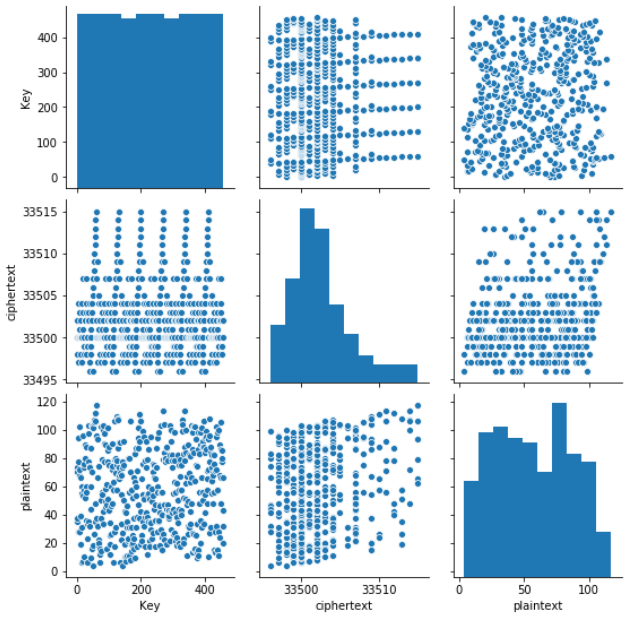


Fig. 7. Correlation Matrix between Plaintext, Ciphertext and Key

Fig. 7 shows the correlation coefficient between the plaintext and ciphertext with different keys. This result shows that there is a very weak/low correlation that is nearly undetectable between the plaintext and its corresponding ciphertext.

Uniform Distribution: Looking at the distribution of plaintext and its corresponding ciphertext in Fig. 7, the result clearly shows that the contents of the ciphertexts have a good level of mixing and it is uniformly distributed. If the frequency counts of the encrypted generation are close to a uniform distribution, then it is possible to categorize that the concerned cipher under test has a good level of mixing.

Key sensitivity: The sensitivity of random keys K and plaintext in bit-level are computed and analyzed using hamming distance with the following equations.

$$K = \frac{\sum_{k=1}^{T} E_k(P) \oplus E_{k'}(P)}{T} \times 100\% \quad (6)$$

Where T is the length of the plaintext and ciphertext message (in bits), All the elements of k are equal to those of k' and P is the plaintext or ciphertext.

The correlation sensitivity of the plaintext and ciphertext with respect to random keys are shown in Fig. 7. This shows the vast change in the ciphertext with respect to a small change in the secret keys or the plaintext.

The proposed hybrid framework was subjected to decryption attack algorithm on symmetric FHE proposed by Vizár and Vaudenay [25] and compared with the existing state-of-art schemes. The results show that the decryption attack algorithm failed to recover the original plaintext message after several tests using different series of ciphertexts, Table 5 shows the status of the decryption attack algorithm on the proposed hybrid framework as compared with other symmetric FHE algorithms.

Table 5: Results of Decryption Attack Algorithm

Authors	Type of Model/Algorithms	Status of Decryption Attack
Kipnis and Hibshoosh [29]	MORE Algorithm	Succeed
Hariss, et al. [26]	Enhance MORE	Succeed
Vizár and Vaudenay [25]	Simplified EMORE	Succeed
Proposed Hybrid Symmetric based FHE Framework	Hybrid of RNS based N-prime model and MORESIMS	Failed

5 Conclusion

In this paper, hybrid symmetric-based FHE framework based on MORESIMS and modified N-prime model was proposed as an improvement over existing symmetric FHE schemes for real-world applications. The proposed hybrid framework takes in integer-based plaintext together with generated secret keys plus secret information moduli set (SIMS), passes it to RNS forward conversion module and RNS arithmetic operation to produce the first level of ciphertext ct_i as residues. MORESIMS channel received ct_i

and secret invertible matric k to produce MORRE-like encryption as the second level of encryption. The ciphertexts produced at the end of the proposed hybrid framework were tested and analyzed, the results show that the ciphertext size is less than 4 times of its equivalent plaintext size, this is a promising result as compared with the existing state-of-art schemes that produce exceptionally large ciphertext size of like 10 times of its plaintext size.

Despite double encryption layers involved in proposed hybrid symmetric based FHE framework to resist plaintext/ciphertext attacks or any other forms of statistical attacks, we found out that the framework still has a very considerable encryption execution time as compared with existing state-of-art schemes. Further research can be investigated to determine the effect of Public Moduli Set (PMS) on the proposed hybrid symmetric based FHE framework on the RNS-BAOCS. Also, optimization of RNS based matrices arithmetic operations with PMS can as well be investigated on the cloud server.

References

1. X. Song and Y. Wang, "Homomorphic cloud computing scheme based on hybrid homomorphic encryption," in 2017 3rd IEEE International Conference on Computer and Communications (ICCC), 2017: IEEE, pp. 2450-2453.
2. B. Seth, S. Dalal, and R. Kumar, "Hybrid Homomorphic Encryption Scheme for Secure Cloud Data Storage," in Recent Advances in Computational Intelligence: Springer, 2019, pp. 71-92.
3. R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," Foundations of secure computation, vol. 4, no. 11, pp. 169-180, 1978.
4. C. Gentry, A fully homomorphic encryption scheme (no. 09). Stanford University Stanford, 2009.
5. C. Gentry, "Fully homomorphic encryption using ideal lattices," in Stoc, 2009, vol. 9, no. 2009, pp. 169-178.
6. C. Gentry and S. Halevi, "Fully homomorphic encryption without squashing using depth-3 arithmetic circuits," in 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science, 2011: IEEE, pp. 107-109.
7. Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-LWE and security for key dependent messages," in Annual cryptology conference, 2011: Springer, pp. 505-524.
8. Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," SIAM Journal on Computing, vol. 43, no. 2, pp. 831-871, 2014.
9. Y. Chen, B. Gu, C. Zhang, and H. Shu, "Reliable Fully Homomorphic Disguising Matrix Computation Outsourcing Scheme," in 2018 14th

International Wireless Communications & Mobile Computing Conference (IWCMC), 2018: IEEE, pp. 482-487.

10. I. Chillotti, N. Gama, M. Georgieva, and M. Izabachene, "Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds," in International Conference on the Theory and Application of Cryptology and Information Security, 2016: Springer, pp. 3-33.
11. L. Ducas and D. Micciancio, "FHEW: bootstrapping homomorphic encryption in less than a second," in Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2015: Springer, pp. 617-640.
12. C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in Annual Cryptology Conference, 2013: Springer, pp. 75-92.
13. Y. Hu, "Improving the efficiency of homomorphic encryption schemes," 2013.
14. M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2010: Springer, pp. 24-43.
15. Z. Brakerski, "Fundamentals of fully homomorphic encryption," in Providing Sound Foundations for Cryptography, 2019: ACM, pp. 543-563.
16. B. Alaya, L. Laouamer, and N. Msilini, "Homomorphic encryption systems statement: Trends and challenges," Computer Science Review, vol. 36, p. 100235, 2020.
17. J. H. Cheon and J. Kim, "A hybrid scheme of public-key encryption and somewhat homomorphic encryption," IEEE transactions on information forensics and security, vol. 10, no. 5, pp. 1052-1063, 2015.
18. P. Burtyka and O. Makarevich, "Symmetric fully homomorphic encryption using decidable matrix equations," in Proceedings of the 7th International Conference on Security of Information and Networks, 2014: ACM, p. 186.
19. C. Gupta and I. Sharma, "A fully homomorphic encryption scheme with symmetric keys with application to private data processing in clouds," in 2013 Fourth International Conference on the Network of the Future (NoF), 2013: IEEE, pp. 1-4.
20. J. Li and L. Wang, "Noise-free Symmetric Fully Homomorphic Encryption based on noncommutative rings," IACR Cryptology ePrint Archive, vol. 2015, p. 641, 2015.
21. M. Liang, "Symmetric quantum fully homomorphic encryption with perfect security," Quantum information processing, vol. 12, no. 12, pp. 3675-3687, 2013.

22. C. Umadevi and N. Gopalan, "Outsourcing Private Cloud Using Symmetric Fully Homomorphic Encryption Using Q^{n}p Matrices with Enhanced Access Control," in 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), 2018: IEEE, pp. 328-332.
23. Q. Qu, B. Wang, Y. Ping, and Z. Zhang, "Improved Cryptanalysis of a Fully Homomorphic Symmetric Encryption Scheme," Security and Communication Networks, vol. 2019, 2019.
24. M. A. Mohammed and F. S. Abed, "An improved Fully Homomorphic Encryption model based on N-Primes," Kurdistan Journal of Applied Research, vol. 4, no. 2, pp. 40-49, 2019.
25. D. Vizár and S. Vaudenay, "Cryptanalysis of Enhanced MORE," 2018.
26. K. Hariss, H. Noura, and A. E. Samhat, "Fully Enhanced Homomorphic Encryption algorithm of MORE approach for real world applications," Journal of Information Security and Applications, vol. 34, pp. 233-242, 2017.
27. K. Hariss, H. Noura, A. E. Samhat, and M. Chamoun, "An Efficient Solution Towards Secure Homomorphic Symmetric Encryption Algorithms," in ITM Web of Conferences, 2019, vol. 27: EDP Sciences, p. 05002.
28. K. Hariss, H. Noura, A. E. Samhat, and M. Chamoun, "Design and Realization of a Fully Homomorphic Encryption Algorithm for Cloud Applications," in International Conference on Risks and Security of Internet and Systems, 2018: Springer, pp. 127-139.
29. A. Kipnis and E. Hibshoosh, "Efficient Methods for Practical Fully Homomorphic Symmetric-key Encrypton, Randomization and Verification," IACR Cryptology ePrint Archive, vol. 2012, p. 637, 2012.
30. D. Vizár and S. Vaudenay, "Cryptanalysis of chosen symmetric homomorphic schemes," Studia Scientiarum Mathematicarum Hungarica, vol. 52, no. 2, pp. 288-306, 2015.
31. K. J. Muhammed and K. Gbolagade, "Enhanced MORE Algorithm for Fully Homomorphic Encryption Based on Secret Information Moduli Set," in 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2019: IEEE, pp. 469-473.
32. Muhammed K.J., Isiaka R.M., Asaju-Gbolagade A.W., Adewole K.S. and Gbolagade K.A. (2021) “Improved Cloud-Based N-Primes Model for Symmetric-Based Fully Homomorphic Encryption Using Residue Number System“. In: Chiroma H., Abdulhamid S.M., Fournier-Viger P., Garcia N.M. (eds) Machine Learning and Data Mining for Emerging Trend in Cyber Dynamics. Springer, Cham. https://doi.org/10.1007/978-3-030-66288-2_8