Secure Data Sharing in The Cloud Through Enhanced RSA

Islam abdalla mohamed[†], Loay F.Hussein[†], Anis Ben Aissa[††], Tarak kallel [†††]

[†]Department of Computer Science, Jouf University, Saudi Arabia
[††]National School of Engineering University of Tunis, Tunisia
[†††]Department of Physics, Jouf University, Saudi Arabia

Summary

Cloud computing today provides huge computational resources, storage capacity, and many kinds of data services. Data sharing in the cloud is the practice of exchanging files between various users via cloud technology. The main difficulty with file sharing in the public cloud is maintaining privacy and integrity through data encryption. To address this issue, this paper proposes an Enhanced RSA encryption schema (ERSA) for data sharing in the public cloud that protects privacy and strengthens data integrity. The data owners store their files in the cloud after encrypting the data using the ERSA which combines the RSA algorithm, XOR operation, and SHA-512. This approach can preserve the confidentiality and integrity of a file in any cloud system while data owners are authorized with their unique identities for data access. Furthermore, analysis and experimental results are presented to verify the efficiency and security of the proposed schema.

Keywords:
Cloud computing; Data sharing; Data confidentiality; Encryption and Authentication.

1. Introduction

People nowadays consume massive amounts of data daily as a result of the rapid growth of wireless technology (5G), cloud services, the Internet of Things, and mobile technologies. At our current rate, 2.5 quintillion bytes of data are created every day [1][2]. Data sharing is one of the most important features of cloud storage services. It enables users to exchange data in the cloud with a group of people. [3]. As a result, a huge amount of significant personal information and critical organization data, such as government documents personal health records [4], and company financial data are distributed across the internet and stored on cloud servers. Although cloud systems could bring enormous benefits to managing massive data, there are still some security and privacy issues that have been raised like Data confidentiality [5]. There is a high chance that the data may be manipulated or deleted if the cloud server gets corrupted, which affects the data's confidentiality and integrity. [6]. Therefore, users should always encrypt the data before outsourcing. Various encryption techniques may be used with the cloud system, each with its pros and downsides which make it difficult to choose one, especially if it's necessary, to secure data confidentiality, integrity, and availability [7][8].

Manuscript received February 5, 2023
Manuscript revised February 20, 2023
https://doi.org/10.22937/IJCSNS.2023.23.2.9

Hypothetical research has developed many algorithms for cloud storage security requirements, such as attribute-based encryption (ABE), distributed systems with information flow control (DIFC), proxy re-encryption (PRE), cipher text search algorithms, and fully homomorphic encryption [9]. However, for providing intricate access control these traditional encryption strategies are not suited to a certain extent [10]. Recently, Identity-based encryption (IBE) technology has had a greater impact on secure data sharing in the public cloud [11]. With IBE, the public identity (e.g., email address) of the intended recipient (also called authorized user) along with data is encrypted by the data owner to provide control over his data. As a result, the authorized user can only access the outsourced data. However, IBE must be used to ensure flexible data exchange amongst users with varying privileges [12]. The RSA is a strong algorithm used in many domains like cloud computing, image encryption, and proxy signature. Its weakness lies in its slowness when using large primary numbers and its vulnerability to factorization attacks and Wiener's attacks if the public key is small. Therefore, this paper presented an Enhanced RSA encryption (ERSA) scheme for providing a secure data-sharing system in the public cloud without losing data confidentiality, integrity, and usability.

2. Related Works

Mohd Saiful et al. [13] presented a study on several RSA schemes. The study concluded that although the RSA algorithm is slow because it depends on a large primary number, it's implemented on applications that are based on internet technology like cloud computing and the internet of things.

Willy Susilo et al. [14] revisit the attack technique on the RSA algorithm by Wiener in 1990 and Boneh and Durfee in 2000. Using the same principle, a new formula of attack is presented. The attack is effective when the public key(e) is substantially less than (N), which is the product of two prime integers multiplied together.

K.Jaspin et al.[15] proposed a Double encryption technique using AES and RAS to encrypt files in the cloud. The time of encryption and decryption is lower compared with other encryption methods like DES, blowfish, and RC5 which make it more efficient. This method also provides resistance against propagation errors.

In [16] Uma Somani et al. protect the data cloud, by using signature with an RSA method. The paper did not compare the method's efficiency with a different encryption technique.

In [17] Randa Mohamed et al. created a novel hybrid encryption scheme that combines the Blowfish and RSA algorithms. The new method overcomes the disadvantage of the two algorithms and can be used in Cloud and IoT systems.

P. Prem Priya et al. [18] present a new framework based on XOR-RSA, SHA512 for securing the transmission, and Brownian motion based on a squirrel search algorithm for energy efficiency. The proposed framework's security is better than El-Gamal and Diffie-Hellman algorithms.

Hongbo Li al. [19] provided identity-based encryption with equality test supporting flexible authorization (IBEETFA) for safe data sharing in the cloud. This method supports an authorized user to search over cipher texts encrypted with different public keys. The investigational results demonstrated that the IBEETFA scheme is effective and can appease different types of searches over encrypted data. The main disadvantage of this method was that it was slower than public key encryption.

Xu An Wang et al. [20] built safe social cloud data sharing called the identity-based proxy re-encryption plus (IBPRE+) based on the concept of proxy re-encryption plus (PRE+). The data owner encrypted their data contents before sending it to cloud storage servers. By using their secret key, the data could be recovered and decrypted. The experimental result proved that the IBPRE+ scheme provides elastic and secure social cloud data sharing.

M. Kumar et al. [21] presented a method for generating a private key for the user with a single semi-trusted key generation center (KGC). This authenticated the user and generated a partial private key that protected the user's private key with their secret keys. A blind technique based on Elliptic-curve cryptography (ECC) was used to secure the transmission over public channels. The proposed scheme provides protection against indistinguishability chosen ciphertext attack (IND-ID-CCA) on a given ID and has excellent computation performance.

3. RSA Enhancement

The rapid evolution of the internet network lead to what we called today, the Information Age. The exponential growth in Data productivity and sharing increases the need for data protection from manipulation and cyber-attacks through encryption. Encryption is the process of transforming readable information into unreadable data. The essential keys of encryption and decryption are data confidentiality, data integrity, and authentication. In 1977, the RSA public key cryptography algorithm, named after its creators, Ron Rivest, Adi Shamir, and Leonard Adleman, was released. RSA has been widely used throughout the years to secure data and information transmission, in IoT, cloud servers, internet protocols, key exchanges, and any area where secure communication between two parties is necessary. Table 1 provides research on the enhancements of RSA algorithms in cloud computing services between 2018-2022.

Table 1:Enhancement research of RSA from 2018-2022

References	Year	Features	Enhancement
Y. Kiran Kumar et al. [22]	2019	Prime numbers and Multiple public keys	Security
Priyadharshini Kaliyamoorthy et al. [23]	2021	Mathematical logic	Data integrity
Rohini et al. [24]	2018	Hybrid algorithm RSA with HMAC	Security
Anuj Kumar et al. [25]	2020	Hybrid algorithm RSA with DES	Security
Khalid El Makkaoui et al. [26]	2019	Mathematical logic	Decryption speed
I. Arockia Antony Samy et al. [27]	2022	Mathematical logic	Security
Nidhi Kumari et al. [28]	2022	Hybrid algorithm blowfish with RSA	Security

4. Attacks on the RSA Algorithm

Although the RAS algorithm is implanted to secure many systems it has its flows. These flows lead to attacks that can be divided into three categories as follows:

1) Brute Force Attacks

This attack involves trying all possible solutions to get the private and the public key. RSA algorithm relay on primary numbers if these numbers are small attackers can guess them easily [29]. Choosing large public and private keys can prevent this attack but it will consume more CPU time to do the mathematics. This problem can be solved by using artificial intelligence to speed up the process. [30].

2) Factorization Attacks

Any cryptographic algorithm can be divided into a set of mathematical operations. The first mathematical step in RSA is to find two primary numbers q and p and multiply them together to get module N. If the attacker succeeded to calculate q and p using N which is known he can decrypt the cipher with the public key help. Integer Factorization Problem algorithm (IFP) [31] and Fermat's factorization algorithm can be used to conduct this type of attack [32].

3) Timing Attack

The idea behind this attack is to calculate the running time needed to get a secret parameter from different input values [33]. When using the RAS algorithm, the public and the private keys must be big numbers to make them unbreakable. The downside of this is the time needed to encryption and decryption a message. To overcome this issue, a mathematical method such as the Chinese Remainder Theorem (CRT) is applied to decryption. CRT-RSA, on the other hand, is four times faster than the basic RSA algorithm and is used to launch timed attacks against RSA [34]. Also, the Bellcore attack [35] used the power of the (CRT) algorithm to reveal the secret modulus N by introducing a single fault result in a signature [36].

5. Proposed Methodology

Cloud computing offers a flexible and appropriate way for data sharing, which brings a variety of benefits to the community and individuals. The main crisis for data sharing in the public cloud is privacy and encryption algorithm performance. Because data often contains valuable information, there is a natural resistance to outsourcing user-shared data directly to the cloud server. Therefore, it is necessary to place advanced cryptography system access control with shared data. Enhanced Rivest Shamir Adleman (ERSA) scheme provides a secure data-sharing system in the public cloud without losing data confidentiality and usability. The proposed framework of the ERSA encryption model on the public cloud model is shown in Fig.1.

The proposed framework consists of four phases, Register, Key Generation and encryption, Authentication, and Decryption.

1) Register Phase

In this phase, users registered their information to the cloud to share data or use the system. For that purpose, users choose a unique identity i.e., username (u_{name}) and password (u_p), and then send a registration request to the cloud server along with $\{u_{name}, u_p\}$ as in (1).

$$\text{New user} \xrightarrow{\text{RREQ}\{u_{name},u_p\}} \text{Cloud server} \quad (1)$$

Where $\text{RREQ}\{u_{name},u_p\}$ represents the registration request. After receiving the registration request from the user, the cloud server starts key generation.

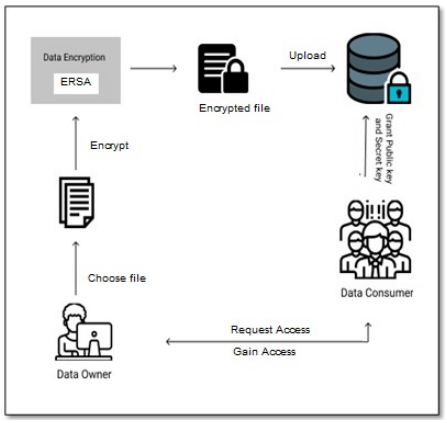


Fig.1 A framework of Data Sharing Model on Cloud

2) Key Generation and encryption Phase

The RSA algorithm is an asymmetric cryptography algorithm that uses a public key and a private key. The public key is shared openly, while the private key is secret and cannot be shared with others. The RSA algorithm is a type of asymmetric encryption that uses both a public and a private key. The drawback of the RSA is its slowness in key generation, and the level of security will be weak if the primary keys are small numbers, thus another key (secrecy key) is generated to improve the security level. The secret key is generated by applying XOR between the public key and the private key then combining N with the result and finally adding the result with the SHA-512 signature of the uploaded file using XOR. If the complexity of encryption is elevated in addition to decryption, it is difficult to find the original data. To improve the speed of the RAS algorithms NF method [37] is used to generate the primary number. The following steps involved using ERSA in file encryption and decryption.

A) Baseline ERSA

- The initial process starts with choosing two large prime numbers, p and q. To calculate N as in (2), these numbers should be similar in magnitude but differ in length ($p \neq q$).

$$N=p\times q \quad (2)$$

- Calculate ϕ as a derived number, which should be greater than 1 and less than (u-1) and (v-1) as in (3).

$$\phi =(p-1)(q-1) \quad (3)$$

- Then choose (e) which represents the encryption key bigger than 1 and less than ϕ, also e must not be a factor of ϕ.
- Calculate (d) which represents the decryption key as in (4).

$$D=e^{-1} \ (\text{mod } \phi) \quad (4)$$

- Use XOR with the public key and private key, to generate T as in (5), so if e=29 and d=41 T will be 52.

$$T=e \oplus d \quad (5)$$

- Determine C, by concatenating T, e, and N in order. for example, T=52, e=29, and N=133, C=5229133. Note that T will be always on the left side, e in the middle between T, and N on the right side.
- Finally, C and the SHA-512 signature of the encrypted file will be added using XOR as in (6) to calculate S which represents the secret key. To encrypt the file (e, N) is used as a public key, and to decrypt the file (e, S) will send by Email or secured network channel. Note that the private key will never be shared which will add sold secure layer to the RSA algorithm. Whenever a user registers their details to the cloud and uploads a file to encrypt, the SHA-512 of the original file will be stored in the cloud and then the will KGC generates the needed keys. Also, the secret key will be generated before file encryption to protect the integrity of the file.

$$S= C \oplus SHA_{file} \quad (6)$$

B) NF-ERSA

Based on Pollard rho Factorization algorithm which was published by J.M.Pollard in 1975 the NF method can calculate the N module [38]. Since the NF method is very fast in calculating N, q, and p it will be used with the ERSA as follows:

- Let any positive integer is N.
- Apply the ceiling function to the result of the square root of N as in (7). The outcome will be a positive integer number.

$$K = \lceil \sqrt[2]{N} \rceil \quad (7)$$

- Decrement by one the square of K as in (8).

$$K=K^2-1 \quad (8)$$

- Calculate the greatest common divisor as in (9), if the result is greater than one then it's the factor of N, then calculate q by (10), Otherwise, increment the value of K by one and recalculate the greatest common divisor.

$$p =gcd (K,N) > 1 \quad (9)$$

$$q =N / p \quad (10)$$

- Use (3) to get ϕ and then choose e, $1< e < \phi$.
- Determine d using (4), T using (5), and S and (6).
- Where M represents the plain text by using (11) the text will be encrypted.

$$E=(M \bmod N * T) \oplus S \quad (11)$$

3) Login and Authentication Phase

Authentication is the security policy that allows access to the information after verifying the identity of the user. Data on the cloud environment are stored based on the identities of the owners: username and password. By using these details, the cloud verifies the data owners. If data consumer wants to access a file in the cloud, they use their u_{name} and u_p to login into the cloud system and then send a request to access the file to the data owner using Email. if the owner wishes to share the file, he shares his public key and secrecy key with those data consumers and then the data consumers will allow access to the file as in (12).

$$DC \xrightarrow{LREQ\{e,S\}} \text{Cloud server} \quad (12)$$

Where LREQ{e, S} represents the request using the public key and the secret key and DC denotes the data consumer. The system verifies whether the user is an authorized user or not. if the DC sends the correct keys, the server confirms that the user is the authorized DC and allows him to decrypt and download the cloud data, otherwise it declines the user request. Fig. 2 shows the system architecture for file download.

4) Decryption Phase

In this phase, the data consumer decrypts the data by using their encrypted private key and secret key by following the ERSA decryption process.

Fig. 2 File Download Process

When a consumer wants to download the file, he will send a request to the owner to get the public key and the secret key. The cloud system first will get the stored SHA-512 of the requested file and then use XOR as in equation (13) to get C. The value of C is the concatenation of T, e, and N in order, so by searching for e in C the value of T will be on the left side of e and N will be on the right side. Then the system will calculate the private key by using XOR with the public key and T as in (14).

$$C = S \oplus SHA_{file} \quad (13)$$

$$d = e \oplus T \quad (14)$$

By using (15), the original data can be recovered by using the private key, secret key, T, and N. Therefore, the encrypted file can only be downloaded by the owner or the authorized users, thus the confidentiality of the data is improved significantly.

$$M = (E \oplus S) \bmod N/T \quad (15)$$

6. Result and Discussion

In this section, the proposed technique's outcome is discussed with an experimental evaluation which includes performance metrics and comparative analysis with graphical plots. The performance of the ERSA algorithm is compared to the existing encryption algorithms Blowfish, RSA, and Elliptic Curve Cryptography (ECC) based on encryption time, decryption time, and security analysis.

A) Encryption Time:

It indicates the time needed by the encryption algorithm to create an encrypted text from plain text. The encryption time equals the difference between the encryption's ending and start times as in (16), where P_e represents the encryption ending time and S_b is the start time.

$$E_{eb} = P_e - S_b \quad (16)$$

B) Decryption Time:

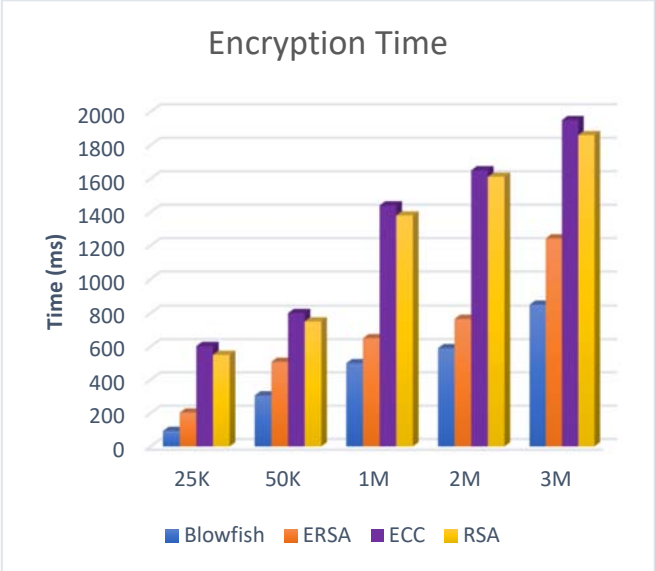
It indicates the time needed by the encryption algorithm to get the original file from the encrypted text. It is the time difference between decryption ending and starting times as in (17), where C_e represents the decryption ending time and R_b is the decryption start time.

$$D_{eb} = C_e - R_b \quad (17)$$

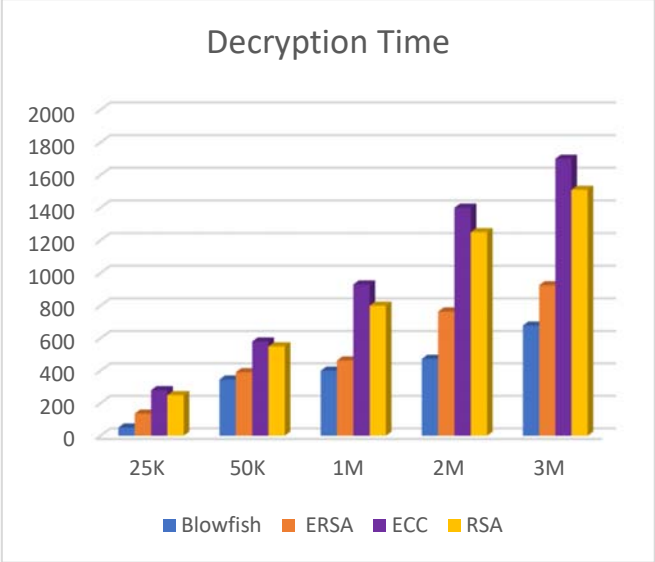
C) Security Analysis:

Cloud storage security is very important. It is calculated by dividing the hacked text by the entire amount of original text as in (18), where H_d denotes hacked data and O_t the original text.

$$S_{dt} = H_d / O_t \quad (18)$$



(a)



(b)

Fig. 3 Performance analysis of ERSA and existing algorithms based on Encryption Time (a) and Decryption Time (b).

Fig. 3 represents the comparative analysis of the proposed and existing technique's performance in terms of (a) encryption time and (b) decryption time. The performance is assessed on the data size ranges from 25 Kilobytes to 3 Megabytes. For encrypting 25K data, the proposed ERSA takes 203ms. The existing RSA, Blowfish, and ECC consume 550ms, 50ms, and 603ms. Similarly, for the remaining data size, the ERSA takes less time compared to the original RSA, and ECC. For decrypting 25 Kilobyte to 3 Megabyte data, the ERSA takes 136ms, 391ms, 463ms, 764ms, and 925ms respectively, which is lesser than existing

RAS and ECC. This comparison proved that the proposed algorithm works faster than these cryptography algorithms (RAS and ECC). The Blowfish algorithm scores less time in encryption and decryption than ERAS, but it's a symmetric-key block cipher while ERSA is an asymmetric cipher. Fig. 4 evaluates the performances of ERSA, Blow-fish, ECC, and RSA in terms of security level.

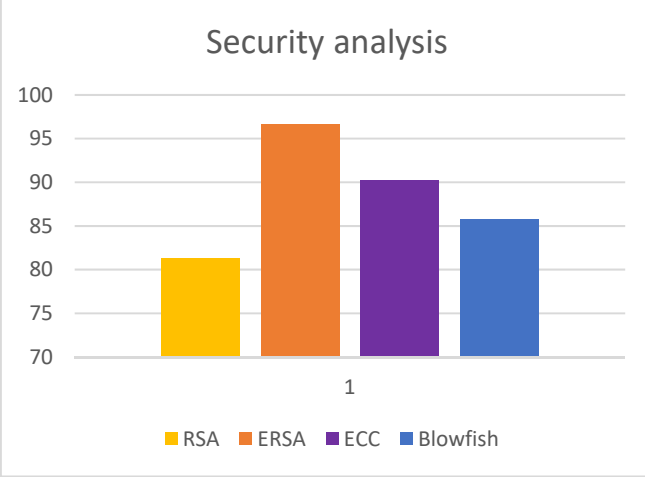


Fig. 4 Security Analysis

The proposed algorithm ERSA provides a security level of 96.6%, but the Blowfish, ECC, and RSA scored 85.7%, 90.23%, and 81.3% respectively, which are lower than ERSA.

Blowfish is the fastest in encryption and decryption, but it's only 64-bit block size and therefore it is vulnerable to birthday attacks and should not be used to encrypt files larger than 4Gb in size.

The ECC algorithm is suffering from physical attacks like Side Channel Analysis (SCA) and Fault Analysis (FA). So, it is clear that the proposed algorithm is more secure than the others algorithms. It protects the data from unauthorized users efficiently by combining the SHA-512 of the file with RSA and the secret key. Also, it solves the vulnerability that lies with choosing a small primary number by using the secret key.

7. Conclusion

The security issue in the cloud environment is one of the major barriers to public cloud implementation. To overcome this issue, this paper proposes an ERSA algorithm, which protects data owners from unauthorized adversaries. To strengthen data confidentiality, the owner encrypts his data using the ERSA method and then uploads it to the cloud, and only based on the secret key, file SHA-512, and the private key data consumer authorized to access the data. To verify the effectiveness of the system, the proposed cryptography algorithm is compared with three cryptography algorithms based on encryption time, decryption time, and security. The experimental results showed that the proposed technique has the highest score in security. Also, the ERSA has a great defense against attacks that rely on a small primary key, and by using the NF algorithm the IRSA algorithm became faster and simpler in generating the public and the private key than the original RSA.

References

[1] Y. Zhang, J. Yu, R. Hao, C. Wang, and K. Ren, "Enabling efficient user revocation in identity-based cloud storage auditing for shared big data," IEEE Transactions on Dependable and Secure Computing, vol. 17, no. 3, pp. 608 - 619, 2018.

[2] Islam Abdalla Mohamed, Anis Ben Aissa and Loay F Hussein, "Classification For Iot Threats Based On The Analytic Hierarchy Process," International Journal of Scientific & Technology Reaearch vol. 9, 2020.

[3] O. A. Khashan, "Secure outsourcing and sharing of cloud data using a user-side encrypted file system," IEEE Access, vol. 8, pp. 210855–210867, 2020.

[4] X. Zhang, Y. Tang, S. Cao, C. Huang, and S. Zheng, "Enabling identity-based authorized encrypted diagnostic data sharing for cloud-assisted E-health information systems," Journal of Information Security and Applications, vol. 54, pp. 102568, 2020.

[5] X. Zhang, Y. Tang, H. Wang, C. Xu, Y. Miao, and H. Cheng, "Lattice-based proxy-oriented identity-based encryption with keyword search for cloud storage," Information Sciences, vol. 494, pp. 193–207, 2019.

[6] H. Wang, Y. Zhang, K. Chen, G. Sui, Y. Zhao et al., "Functional broadcast encryption with applications to data sharing for cloud storage," Information Sciences, vol. 502, pp. 109–124, 2019.

[7] D. K. Shukla, V. K. R. Dwivedi, and M. C. Trivedi, "Encryption algorithm in cloud computing," Materials Today: Proceedings, vol. 37, pp. 1869–1875, 2020.

[8] J. Shen, X. Deng, and Z. Xu, "Multi-security-level cloud storage system based on improved proxy re-encryption," EURASIP Journal on Wireless Communications and Networking, vol. 2019, pp. 277, 2019.

[9] J. Raja and M. Ramakrishnan, "Confidentiality-preserving based on attribute encryption using auditable access during encrypted records in. cloud location," The Journal of Supercomputing, vol. 76, no. 8, pp. 6026–6039, 2020.

[10] H. Deng, Z. Qin, L. Sha, and H. Yin, "A Flexible privacy-preserving data sharing scheme in cloud-assisted IoT," IEEE Internet Things Journal, vol 7, pp. 11601–11611, 2020.

[11] S. Belguith, N. Kaaniche, M. Laurent, A. Jemai, and R. Attia, "Accountable privacy preserving attribute based framework for authenticated encrypted access in clouds," Journal of Parallel and Distributed Computing, vol. 135, pp. 1–20, 2020.

[12] Q. Wei, F. Qi, and Z. Tang, "Remove key escrow from the BF and Gentry identity-based encryption with non-interactive key generation," Telecommunication Systems: Modelling, Analysis, Design and Management, vol. 69, no. 2, pp. 253–262, 2018.

[13] Mohd Saiful Adli Mohamad, Roshidi Din, and Jasmin Ilyani Ahmad, "Research trends review on RSA scheme of asymmetric cryptography techniques," Bulletin of Electrical Engineering and Informatics, Vol. 10, No. 1, pp. 487~492, 2021.

[14] Willy Susilo, Joseph Tonien and Guomin Yang, " Divide and capture: An improved cryptanalysis of the encryption standard algorithm RSA," Computer Standards & Interfaces, Vol 74, 2021.

[15] K.Jaspin, Shirley Selvan, Sahana.S and Thanmai.G, "Efficient and Secure File Transfer in Cloud Through Double Encryption Using AES and RSA Algorithm," 2021 International Conference on Emerging Smart Computing and Informatics,2021.

[16] Uma Somani, Kanika Lakhani and Manish Mundra, "Implementing digital signature with RSA encryption algorithm to enhance the Data

Security of cloud in Cloud Computing," First International Conference On Parallel, Distributed and Grid Computing, 2010.

[17] Randa Mohamed Abdel Haleem and Eltyeb Elsamani Abd Elgabar, "Enhancing the Integrity of Cloud Computing by Comparison between Blowfish and RSA Cryptography Algorithms," International Journal of Engineering Research & Technology,2022.

[18] P. Prem Priya and Jeevaa Katiravan, "Privacy-Preserving and Energy-Centered QoS for IoT Using XOR-RSA and BM-SSA," Wireless Personal Communications,2021.

[19] Hongbo Li, Qiong Huang, Sha Ma, Jian Shen, and Willy Susilo, "Authorized equality test on identity-based ciphertexts for secret data sharing via cloud storage," IEEE Access, vol. 7, pp. 25409–25421, 2019.

[20] Xu An Wang, Fatos Xhafa, Jianfeng Ma, Zhiheng Zheng, "Controlled secure social cloud data sharing based on a novel identity based proxy re-encryption plus scheme," Journal of Parallel and Distributed Computing, vol. 130, pp. 153–165, 2019.

[21] M. Kumar and S. Chand, "ESKI-IBE: Efficient and secure key issuing identity-based encryption with cloud privacy centers," Multimedia Tools and Applications, vol. 78, no. 14, pp. 19753–19786, 2019.

[22] Y. Kiran Kumar and R. Mahammad Shafi, "An efficient and secure data storage in cloud computing using modified RSA public key cryptosystem," International Journal of Electrical and Computer Engineering, 2019.

[23] Priyadharshini Kaliyamoorthy and Aroul Canessane Ramalingam, "QMLFD Based RSA Cryptosystem for Enhancing Data Security in Public Cloud Storage System," Wireless Personal Communications, 2021.

[24] Rohini and Er Tejinder Sharma, "Proposed hybrid RSA algorithm for cloud computing," Proceedings of the Second International Conference on Inventive Systems and Control, 2018.

[25] Anuj Kumar, Vinod Jain and Anupam Yadav, "A New Approach for Security in Cloud Data Storage for IOT Applications Using Hybrid Cryptography Technique," International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control, 2020.

[26] Khalid El Makkaoui, Abderrahim Beni-Hssane and Abdellah Ezzati, " Speedy Cloud-RSA homomorphic scheme for preserving data confidentiality in cloud computing," Journal of Ambient Intelligence and Humanized Computing, 2019.

[27] I. Arockia Antony Samy and M. Safish Mary, "Secure Data Transmission in Cloud Computing Using Std-rsa With Eslurnn Data Classification and Blockchain Based User Authentication System," Research Square, 2022.

[28] Nidhi Kumari and Vimmi Malhotra, "Secure Cloud Data Storage Using Hybrid Cryptography," International Journal for Research in Applied Science & Engineering Technology, 2022.

[29] S. Selvi and M. Vimala Devi, "A Novel Hybrid Chaotic Map–Based Proactive RSA Cryptosystem in Blockchain," Recent Trends in Blockchain for Information Systems Security and Privacy, 2021.

[30] Chu Jiann Mok, Chai Wen Chuah, "An Intelligence Brute Force Attack on RSA Cryptosystem," Communications in Computational and Applied Mathematics, 2019.

[31] Majid Mumtaz and Luo Ping, "Forty years of attacks on the RSA cryptosystem: A brief survey," Journal of Discrete Mathematical Sciences and Cryptography,2019.

[32] Kritsanapong Somsuk, "The new integer factorization algorithm based on fermat's factorization algorithm and euler's theorem," International Journal of Electrical and Computer Engineering, 2020.

[33] Manish Kant Dubey, Ram Ratan, Neelam Verma and Pramod Kumar Saxena, "Cryptanalytic Attacks and Countermeasures on RSA," Proceedings of the Third International Conference on Soft Computing for Problem Solving, 2014.

[34] Werner Schindler, "A Timing Attack against RSA with the Chinese Remainder Theorem," Cryptographic Hardware and Embedded Systems, 2002.

[35] Andrey Sidorenko, Joachim van den Berg, Remko Foekema, Michiel Grashuisa, and Jaap de Vos, "Bellcore attack in practice," Cryptology ePrint Archive, 2012.

[36] Johannes Blömer, Martin Otto and Jean-Pierre Seifert, "A New CRT-RSA Algorithm Secure Against Bellcore Attacks", Proceedings of the 10th ACM conference on Computer and communications security, 2003.

[37] B. R. Ambedkar and S S Bedi, "A New Factorization Method to Factorize RSA Public Key Encryption, " IJCSI International Journal of Computer Science Issues, Vol. 8, November 2011.

[38] J. Pollard, "Monte Carlo methods for index computation (mod p)," Math. Comp., Vol. 32, pp.918-924, 1978.

Islam Abdalla Mohamed received his B.Sc. and M.Sc. degrees from Al Neelain University, Sudan in 2007 and 2010. He received his Ph.D. degree in Information security from Al Neelain University, Sudan in 2015. Currently, working as an assistant professor at Jouf University, Saudi Arabia. His research interest in information security and cyberattacks.

E-mail: iaabass@ju.edu.sa

Loay F. Hussein received his B.Sc. degree in Electrical and Electronics Engineering (Computer Engineering) from the university
of science & technology, Sudan, 2005. In addition, he received MSc and Ph.D. degrees in Computer and Information Engineering from International Islamic University Malaysia in 2011 and 2016, respectively. Meanwhile, he is working as an assistant professor at Jouf University (KSA) from 2017 to till date. His research interest in the areas of Quality of Service in IP networks, IoT, network security and mobile communications. E-mail: lfahmed@ju.edu.sa

Anis Ben Aissa received his B.S, M.S, and Ph.D. in Computer Science from Tunis-El Manar University in 2005, 2007, and 2013, respectively. Currently, he is an Assistant Professor at Computer Science Department at National School of Engineering University of Tunis, E-mail:anis.benaissa@enit.utm.tn.

T. KALLEL received his Bachelor's, Master's, and Ph.D. in physics from Sfax University in 2005, 2011, and 2014, respectively. Currently, he is an Assistant Professor in the Physics Department at Jouf University, KSA. Email: tkallel@ju.edu.sa