SECURE DATA TRANSMISSION IN CLOUD STROAGE SYSTEM USING NTRU ALGORITHM

M.Rajesh

Professor,Department of Computer Science and Engineering,Sanjivani College of Engineering,Kopragon,India
E-mail:rajesmano@gmail.com

Abstract – Cloud storage system had played a significant role by enabling services such as data storage and management. Currently it is at the peak , because of the active migration of massive amount of data to the cloud by enterprises, individual users and organizations. A vast volume of data yields enormous value. However it faces susceptible security challenges while sharing the data over the network. The data has to be encrypted by the user before uploading for maintaining its secrecy.The introduced approach enables sending vital information in a secret way. In this proposed approach, before uploading the data, masking of data is performed by embedding an original message into another text. The position of each character of masked data is represented in an index table. The index table known as RIF is ciphered using NTRU and then encrypted index table along with masked data in which real data gets embedded is send to the receiver side. At the receiver side, decryption is performed to extract the embedded real message from masked data. The proposed NTRU achieves faster encryption and decryption compared to powerful RSA technique. The faster achievement is due to the simple multiplication of polynomial.
Keywords: Number Theory Research Unit (NTRU), Real message index file (RIF),Decryption, Cloud,Security.

1 INTRODUCTION

Due to the fast expansion of data, it is becoming increasingly difficult for consumers to preserve huge amounts of data domestically. As a result, many businesses prefer cloud storage.So to have a protection of data against malicious attack, high level security is needed [1-3]. Recently, data sharing in cloud becomes one of the most common feature, which makes it possible for numerous people to share data with one another.These exchanged information contains some sensitive informations. So to ensure security to these data, some cryptographic techniques were used. However, encryption of the whole shared file to mask critical information is impossible. [4-6]. A mass of research work was carried out to provide protection. One of them was virtualization approach, but it fails to monitor the attacks which thwart IDS at a tenant virtual machine [7] and an another approach involved in cloud security is, in order to preserve the encoded domain's spatial correlation, homomorphic cryptosystem is used [8-13]. To this crypto system, user possesses encryption key and data embedding key. It takes more time to perform encryption and decryption to retrieve the original data but by the proposed approach faster encryption and decryption is achieved. Access ontrol has been proposed as a viable approach for ensuring anonymity in cloud computing. [16, 17]. But, it cannot guarantee the trustworthiness of unknown people. The majority of conventional systems rely on attribute-based techniques to maintain identity secrecy. However, correlation analysis of user characteristic information poses a danger of identity exposure and it fails to protect user attribute privacy [14, 15]. In addition to these approaches, existing searchable encryption techniques were used [18-23]. It provides inadequacy on the required functionality and confidentiality viewpoints, and is vulnerable to an inside keyword guessing attack.Compared to existing searchable encryption scheme, Encryption using two servers and public keys, alongside keyword search framework [24] achieves strong against inside keyword guessing attack. But it uses homomorphic approach to perform security against this attack. Analogizing with NTRU, RSA is also a powerful existing 1024 bit encryption key and provides good level of security and an adequate level of resistance to assaults [25-29]. However, NTRU has great advantage in generating short keys and it has a capacity to operate even under low memory condition. To overcome from these issues, this proposed approach focuses in generating high level secured data by performing masking of data along with the efficient encryption based on NTRU. Prior to storing the data on the cloud, the suggested work prepare the masked data by embedding the original data into another text and the position of each character is placed in RIF file which is encrypted using NTRU. After performing encryption, it is send to the receiver with masked data along with encrypted RIF file and the hash value of original data. After retrieving the original data by performing decryption. To achieve authentication, Hashing is done in the recovered data and it compared with the received hash value. This proposed work achieves faster performance, works even under low memory condition and tracking of data becomes difficult while sharing data over the network.

The following is the structure of this paper: a short preface about existing approaches and its impacts and the proposed approach is explained in section 1,

Elaborateexplanation of Proposed technique is done in section 2, results and discussion is discussed in section 3 and finally, concluded in section 4.

2 PROPOSED WORK

This proposed system focuses to perform security in cloud against malicious users. To share data over the network in a secure way. An efficient method of encryption based on NTRU is implemented. Before uploading the data, the encryption process is to be done. Initially, the original data is embedded into another text known as masking. In the index table known as Real Message Index file (RIF), the position of each character is placed. It is encrypted with the use of NTRU. NTRU is quite different from other cryptosystem because security is based on integer factorization problem. This significant feature in this algorithm assist in obtaining a rapid performance for encrypting or decrypting a message. To perform this operation, it requires $O(N^2)$ time, whereas other crypto technique such as RSA needs $O(N^3)$ time and also it creates a minimized key size of O(N). The performance of this approach is evaluated based upon the time required for performing delay time, encryption/decryption, security and complexity. Hence delay time, encryption and decryption are changed based upon the count of bits per second. In this proposed work, there is no packet loss while sharing the data over the network. The receiver receives the masked data along with encrypted RIF and the hash value of original message. To obtain the real data, decryption is performed. After retrieving the original data, once again hashing is performed and then it is analogized with the received hash value. If they found to be similar, the original data is authenticated.

Parameters involved in NTRU

To perform this operation, NTRU takes parameters N,P as well as q. N denotes the polynomial degree at major N-1, p and q are utilized for decreasing the polynomial coefficients, p should be less when compared to q with no divisor in common.

Key Generation

To secure a message while transmitting between two nodes, it needs the creation of private as well as public key. f and g which is a key pair of two polynomials are created with coefficients lesser when compared to q with degree at major N-1 and with coefficients in {-1,0,1} are needed.

The polynomial f which was taken has to fulfill the need of the existence of inverse modulo p and modulo q.

2.1 Encryption

Encryption takes place in the quotient ring of polynomials. The operation of NTRU is performed over a ring of truncated polynomials.

$$P = Z_q[x]/(X^N - 1) \quad (1)$$

From polynomial P, the polynomial F is computed by using the equation,

$$f = (f_0, f_1, f_2, f_{N-1}) \quad (2)$$

$$f = f_0 + f_{1x} + f_{N-1}x^{N-1} \quad (3)$$

Here adding of polynomials is represented as pairwise addition related to coefficients of similar degree, the multiplication involved here is its convolution. The polynomial norm f is defined in the normal way, since the square root of the sum of the square of its coefficients and width of the polynomial F is obtained by taking the differences between its larger and smaller coefficient. The rudiment parameter N is denoted as the ring dimension and it is choosen to be prime to hamper the attack from malicious users and the other parameter p and q is also taken as prime and the parameter m in the encryption denotes the binary polynomial of degree N. To perform encryption, the message to be send is in polynomial m form with coefficients between -1/2 p and ½ p. Receiver chooses randomly another value which is known as binding value r to obscure the message. The binding value r is selected as,

r1*r2+r3.

Here r_1, r_2 *and* r_3 are generated by setting them to 0. After adding the binding value, the encrypted message gets forwarded to the receiver side. The working of encryption is done as represented in fig 1.

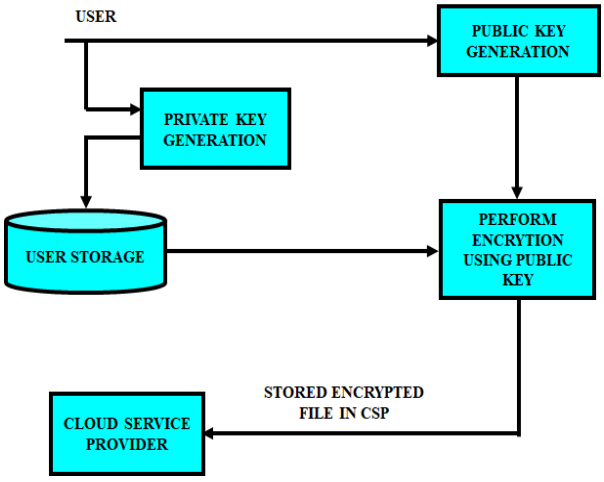

Figure 1 Operation performed in Encryption

2.2 Decryption

Receiver performs decryption on the cipher text to extract the real message by performing computation. Receiver receives the cipher text C and the secret key S_k for decrypting the message as well as the polynomial in decryption 'm' is evaluated by using,

$$m = F * c(mod\ q) \quad (4)$$

Where $S_k = f$, S_k denotes the secret key
C= cipher text.

The coefficient of m is to be selected between –q/2 and q/2 to avoid decryption problem. It should be normally lie in an interval of length q.

The cipher text retrieved to receiver in terms of $C_1 and\ C_2$. $C_1\ and\ C_2$ along with S_1, S_2, e_1, e_2 as noise terms with h. It utilizes secret key f and joint secret key f^2 to perform decryption and extract the real message from it. If the addition of two cipher text $C_1\ and\ C_2$ is fed to perform decryption, it returns the addition of two plaintext $m_1\ and\ m_2$ respectively by using the private key f.

If the multiplication of two cipher text $C_1\ and\ C_2$ is given, it returns the multiplication of two plaintext $m_1 and\ m_2$ by using the joint secret key f^2. The working of decryption is performed as represented in the figure 2.

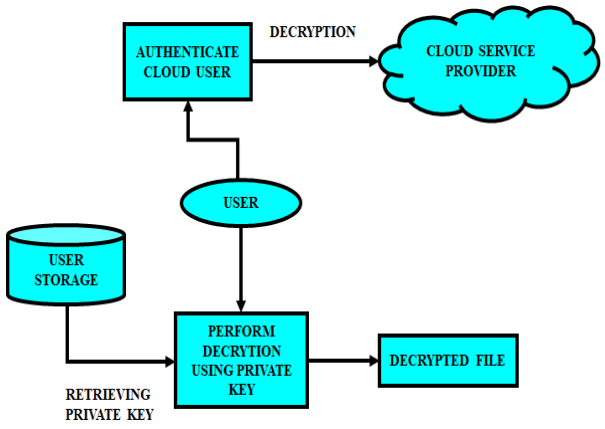

Figure 2 Operation performed in decryption

If the user is found to be authenticated , the private key is shared, by using the secret key, the masked data in which the real data gets embedded is retrived through the process of performing decryption.

This algorithm varies from other existing approaches because of its following features,

- Its security gets reduced to shortest vector problem
- Faster performance of encryption and decryption
- High security and
- Resistance to quantum computing attacks.

Figure 3 shows Proposed work. The mathematical computation done to perform encryption and decryption by using NTRU as follows below.

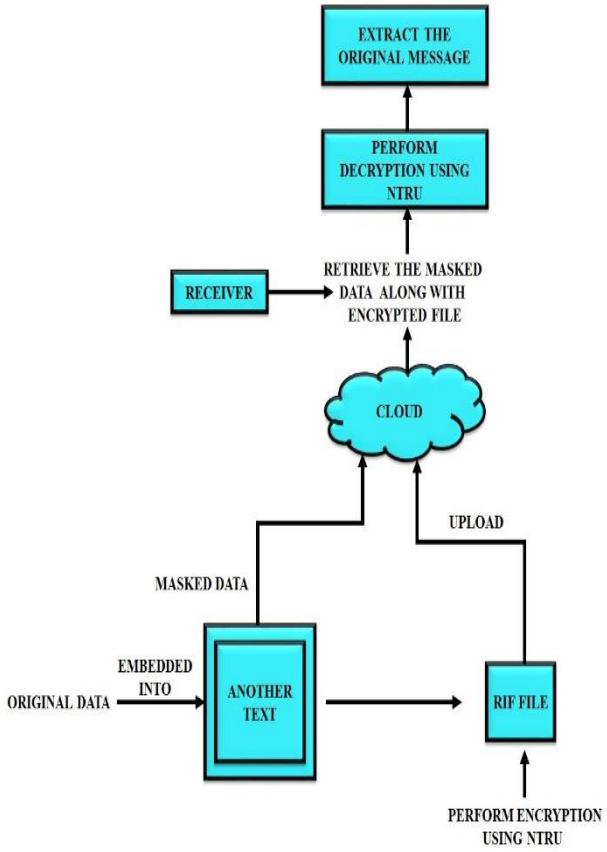

Figure 3 Proposed work

Algorithm of proposed approach is as follows

Step 1: Obtain the input value as plaintext.
Step 2: Perform masking
Step 3: Verify the masked data whether the plaintext was existed in the masked data or not. If existed go to step 4 otherwise go to step2.
Step 4: Generate RIF from masked data.
Step 5: Perform encryption using NTRU
Step 6:Deliver the encoded data to the intended recipient. On the receiving end,
Step 7: retrieve the encrypted data
Step 8: Perform decryption using NTRU
Step 9: Obtain the real message from the masked data.

Encryption and decryption approach using NTRU in this proposed approach is elaborately explained as follows,

2.3 Number Theory Research Units (NTRU) To Perform Encryption and Decryption

NTRU is faster when analogized with existing powerful 1024 bit RSA key. An NTRU is a 263 bit encryption key and the significant feature of this kind of an approach is that the finally retrieved ciphered content is robust and secure when compared with existing encryption techniques. It has great advantage in creating short keys and it has a capability to operate even under low memory condition. The following operation of NTRU is as follows.

2.3.1 Setting Parameters

To perform encryption, the parameters n, p,q,$\propto$, σ as follows. Message space in plaintext

$p = R/pR$ (5)

the parameter p $\in R_q^*$ defines the above equation (1). Polynomial p has a modest coefficient when compared to q.

To encode massive level bits at instant mode, we require

$N(p) = |P| = 2^{\Omega}(n)$ (6)

The element p is expressed below by obtaining the smallest residue of modulus p.

$\sum_{0<i<n} \varepsilon_i \in (-\frac{1}{2}, \frac{1}{2})$ (7)

Since,

$R = Z[x]/(x^n + 1)$ (8)

An element of R consists of any element of P having infinity norm ≤ (deg (p)+1). ||P||

α which is utilised to generate the keys, is the standard deviation related to discrete Gaussian distribution.

2.3.2 NTRU Key Generation

Choosing parameters n, q ∈ Z.

$P = 2 \in R_q^x, \sigma \epsilon R$ (9)

We can sample f' from D_{Z_n}, σ

Let, $f = 2f' + 1$, if f(mod q)≠ R_q^x (10)

Furthermore, we may choose a sample of g from D_{Z^n}, σ

$If\ g(mod\ q) \neq R_q^x$ (11)

The following keys can be returned eventually, the secret key

$S_k = f\ \in R_q^x\ with$ f=1(mod2) (12)

And Public key $P_k = h = 2g/f \in R_q^x$,

$(S_k, P_k) \in R_x R_q^x$, is the key pair.

2.3.3 NTRU Encryption

m∈ P, s,e← r_α are the plaintext and sample provided

Where r_α is the distribution over polynomials and generates cipher text.

It is computed as follows,

$C = hs + 2e + m\ \in R_q\ with\ Pk = h$ (13)

2.3.4 NTRU Decryption

C(cipher text) and $s_k = f_1$are provided.

Calculate $C^1 = f.c \in R_q\ and$ using

$m = c^1(mod\ 2) = f.c(mod\ 2)$ (14)

$= (2\ (gs\ and\ ef) + fm)(mod\ 2) \in p$ (15)

And finally, the value of m is returned.

NTRU add (P_k, c_1, c_2) and NTRU mult(P_k, c_1, c_2)

The cipher texts c_1 and c_2 ,

$C_1 = hs_1 + 2e_1 + m_1\ \in R_q$ (16)

$C_2 = hs_2 + 2e_2 + m_2\ \in R_q$ (17)

with $S_1, s_2, e_1\ and\ e_2$ as noise terms with h, the public key, encryption is performed in the plaintext $m_1 and\ m_2$ respectively.

The algebraic multiplication shows that to retrieve the ciphertext, encryption is done in addition and multiplication of plaintext $m_1\ and\ m_2$. It is performed by

$C_{add} = C_1 + C_2$ (18)

$C_{mult} = C_1.C_2$ (19)

To perform decryption in $C_1 + C_2\ and\ C\_1.C_2$ with secret key f and joint secret key f^2

It is computed as follows,

$f(c_1 + c_2) = 2\big(f(e_1 + e_2) + g(S_1 + s_2)\big) +$

$f(m_1 + m_2) \triangleq 2E_{Add} + f(m_1 + m_2) mod\ 2$ (20)

Where, by utilizing the private key f following the decoding process, the addition of 2 cipher text $C_1$and C_2 is similar to the addition of plaintext $m_1\ and\ m_2$. Hence we obtain,

$f^2(c_1.c_2) = 2(2g^2 s_1 s_2 + g s_1 f(2e_2 + m_2) +$

$g s_2 f(2e_1 + m_1) + f^2(e_1 m_1 + e_2 m_1 + 2e_1 e_2) +$

$f^2(m_1 m_2) \triangleq 2E_{mult} + f^2(m_1 m_2) mod 2$ (21)

It represents that multiplication of 2 cipher texts $C_1 and\ c_2$ is equal to multiplication of plain text $m_1\ and\ m_2$ using the joint secret key f^2 after decryption.

3 RESULTS AND DISCUSSION

To evaluate the proposed system, both the algorithms were provided with same set of data and the execution time is noted. The time taken for encrypting and decrypting the data are measured and taken as evaluation metrics. The evaluation is performed with the following metrics.

3.1 Time for Encryption

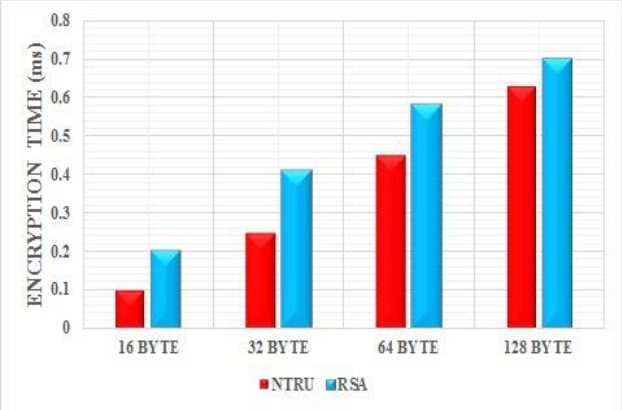

Figure 4 Time for encryption

To perform encryption and decryption and creation of key with NTRU is fast and facile when compared with RSA. For the encoding or decoding the N-length message block, NTRU requires O (N_2) operation. It performs faster because RSA requires O (N_3) operation . In this graph, it clearly shows that, RSA takes more time than NTRU. It is represented in fig 4.

3.2 Average Energy Consumption

Average energy consumption is represented in fig 5. It clearly indicates that, for every byte of information, the energy consumed using NTRU was low compared to the consumption of energy using RSA.

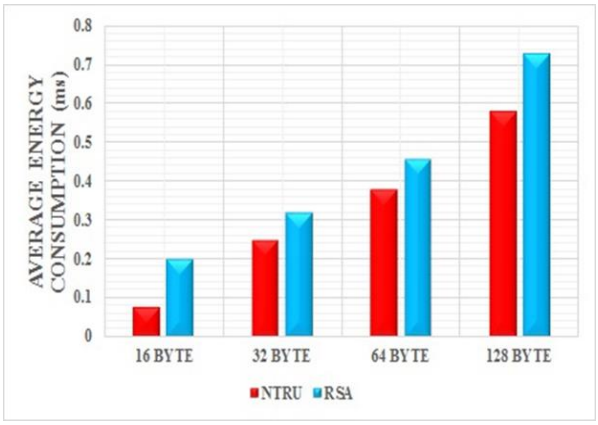

Figure 5 Average energy consumption

3.3 Time Taken for Decryption

The time required to decrypt the data using both the algorithm was analogized to assess the proposed system functionality. Faster performance is the result of simplicity in the implementation of NTRU algorithm. The faster performance in NTRU is due to the simple multiplication of polynomial. In the below mentioned fig.6, it is clearly represented.

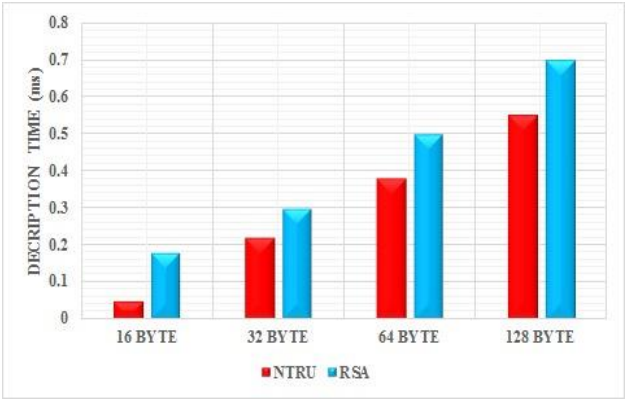

Figure 6 Time taken for decryption

3.4 Average Network Usage

Network usage between NTRU and RSA is shown in Fig 7. NTRU shows lower network usage compared to RSA. For 128 byte of information, NTRU uses 0.6 Kb but the usage in RSA is 0.74 kb. By this, less network usage in NTRU is shown.

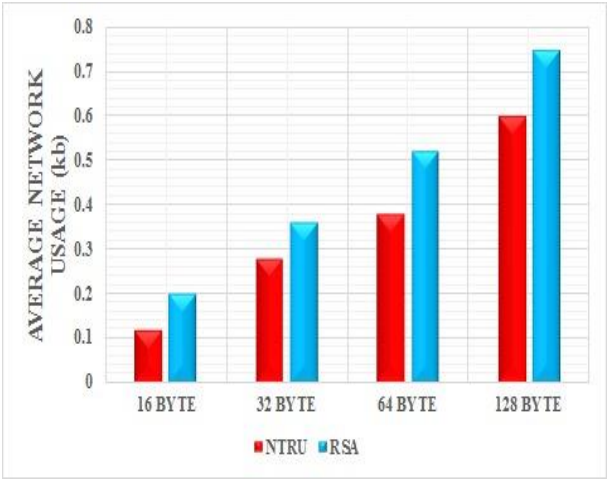

Figure 7 Average network usage

3.5 Average Network Delay

The average network delay involves the processing delay, transmission delay and propagation delay. By considering all this, average network delay is computed. It is represented in figure 8. It also shows good performance when compared to the existing

system. NTRU is found to be quantum safe because still there is no algorithm which breaks it in a quantum time.

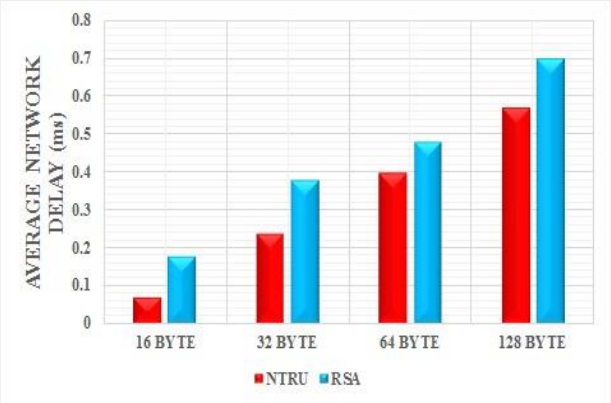

Figure 8 Average network delay

4 CONCLUSION

With the rapid improvement in cloud computing data sharing techniques, several security problems have arisen. So to enhance the security and data sharing in cloud, this proposed work focuses on masking of data and the efficient encryption based on NTRU. By this approach, the data's are protected from the attacks of unauthorized users. In future, a huge variety of public key cryptographies has to be employed owing to the rising need for privacy and security. The suggested embedding and retrieval method is frequently utilised in the process of encryption. The encryption technique in this approach provides robust and secure when comparing with prevailing approaches. It is resistant to quantum computing threats allowing accelerated encoding and decoding.

References

[1] Enrico Bacis, Sabrina De Capitani di Vimercati, Sara Foresti, Stefano Paraboschi, Marco Rosa, PierangelaSamarati, "Securing Resources in Decentralized Cloud Storage", IEEE Transactions on Information Forensics and Security, Vol. 15, No.5, pp. 286-298, 2020.

[2] Pan Yang, NaixueXiong, Jingli Ren, "Data Security and Privacy Protection for Cloud Storage: A Survey",IEEE Access, Vol.8, No. 7, pp.131724-131740, 2020.

[3] Mohsen KarimzadehKiskani, Hamid R. Sadjadpour, "Secure and Private Information Retrieval (SAPIR) in Cloud Storage Systems", IEEE Transactions on Vehicular Technology, Vol.67, No. 12, pp. 12302-12312, 2018.

[4] Wenting Shen, Jing Qin, Jia Yu, RongHao, Jiankun Hu, "Enabling Identity-Based Integrity Auditing and Data Sharing With Sensitive Information Hiding for Secure Cloud Storage", IEEE Transactions on Information Forensics and Security, Vol. 14, No. 2, pp. 331-346, 2019.

[5] Leyou Zhang, Yilei Cui, Yi Mu, "Improving Security and Privacy Attribute Based Data Sharing in Cloud Computing", IEEE Systems Journal, Vol. 14, No. 1, pp. 387-397, 2020.

[6] Kennedy A. Torkura, Muhammad I. H. Sukmana, Feng Cheng, ChristophMeinel, "CloudStrike: Chaos Engineering for Security and Resiliency in Cloud Infrastructure", IEEE Access, Vol. 8, No. 7, pp. 123044-123060, 2020

[7] Preeti Mishra, Vijay Varadharajan, Emmanuel S. Pilli, UdayTupakula, "VMGuard: A VMI-Based Security Architecture for Intrusion Detection in Cloud Environment", IEEE Transactions on Cloud Computing, Vol. 8, No.3, pp. 957-971, 2020.

[8] Ming Li, Lanlan Wang, Jingjing Fan, Yushu Zhang, Kanglei Zhou, Haiju Fan, "Fidelity Preserved Data Hiding in Encrypted Highly Autocorrelated Data Based on Homomorphism and Compressive Sensing", IEEE Access, Vol. 7, No. 5, pp. 69808-69825, 2019.

[9] LizhiXiong, Danping Dong, Zhihua Xia, Xianyi Chen, "High-Capacity Reversible Data Hiding for Encrypted Multimedia Data With Somewhat Homomorphic Encryption", IEEE Access, Vol. 6, No. 10, pp. 60635-60644, 2018.

[10] Pauline Puteaux, ManonVialle, William Puech, "Homomorphic Encryption-Based LSB Substitution for High Capacity Data Hiding in the Encrypted Domain", IEEE Access, Vol. 8, No.6, pp. 108655-108663, 2020.

[11] Feng Hu, Bing Chen, "Channel Coding Scheme for Relay Edge Computing Wireless Networks via Homomorphic Encryption and NOMA", IEEE Transactions on Cognitive Communications and Networking, Vol. 6, No.4, pp. 1180-1192, 2020.

[12] GuoweiQiu, XiaolinGui, Yingliang Zhao 2020, "Privacy-Preserving Linear Regression on Distributed Data by Homomorphic Encryption and Data Masking", IEEE Access, Vol. 8, No. 11, pp. 210855-210867, 2020.

[13] AbdulatifAlabdulatif, Ibrahim Khalil, Albert Y. Zomaya, ZahirTari, Xun Yi , "Fully Homomorphic based Privacy-Preserving Distributed Expectation Maximization on Cloud", IEEE Transactions on Parallel and Distributed Systems, Vol. 31, No.11, pp. 2668-2681, 2020.

[14] Hu Xiong, Hao Zhang, Jianfei Sun, "Attribute-Based Privacy-Preserving Data Sharing for Dynamic Groups in Cloud Computing", IEEE Systems Journal, Vol. 13, No.3, pp. 2739-2750, 2019.
[15] Suqing Lin, Rui Zhang, Hui Ma, Mingsheng Wang, "Revisiting Attribute-Based Encryption With Verifiable Outsourced Decryption", IEEE Transactions on Information Forensics and Security, Vol. 10, No.10, pp. 2119-2130, 2015.
[16] Yichen Zhang, Jiguo Li, Hao Yan, "Constant Size Ciphertext Distributed CP-ABE Scheme with Privacy Protection and Fully Hiding Access Structure" IEEE Access, Vol. 7, No. 4, pp. 47982-47990, 2019.
[17] Kai He, Yijun Mao, JiantingNing, Kaitai Liang, Xinyi Huang, EmmanouilPanaousis, George Loukas, "A New Encrypted Data Switching Protocol: Bridging IBE and ABE Without Loss of Data Confidentiality",IEEE Access, Vol. 7, No.4, pp. 50658-50668, 2019.
[18] Li Lin, Ting-Ting Liu, Shuang Li, Chathura M. SarathchandraMagurawalage, Shan-Shan Tu, "PriGuarder: A Privacy-Aware Access Control Approach Based on Attribute Fuzzy Grouping in Cloud Environments", IEEE Access, Vol. 6, No. 11, pp. 1882-1893, 2018.
[19] Xueqiao Liu, Guomin Yang, Willy Susilo, Joseph Tonien, Ximeng Liu, Jian Shen, "Privacy-Preserving Multi-Keyword Searchable Encryption for Distributed Systems", IEEE Transactions on Parallel and Distributed Systems, Vol. 32, No.3, pp. 561-574, 2021, 2021.
[20] Biwen Chen, Libing Wu, Li Li, Kim-Kwang Raymond Choo, Debiao He, "A Parallel and Forward Private Searchable Public-Key Encryption for Cloud-Based Data Sharing", IEEE Access, Vol.8, No. 2, pp. 28009-28020, 2020.
[21] Chi Chen, Xiaojie Zhu, Peisong Shen, Jiankun Hu, Song Guo, ZahirTari, Albert Y. Zomaya, "An Efficient Privacy-Preserving Ranked Keyword Search Method", IEEE Transactions on Parallel and Distributed Systems, Vol. 7, No.4, pp. 951-963, 2016.
[22] Cheng Guo, Ruhan Zhuang, Chin-Chen Chang, Qiongqiong Yuan, "Dynamic Multi-Keyword Ranked Search Based on Bloom Filter Over Encrypted Cloud Data", IEEE Access, vol. 7, no.3, pp. 35826-35837, 2019.
[23] Hua Dai, Yan Ji, Geng Yang, Haiping Huang, Xun Yi, "A Privacy-Preserving Multi-Keyword Ranked Search Over Encrypted Data in Hybrid Clouds", IEEE Access, Vol. 8, No.12, pp.4895-4907, 2020.
[24] Rongmao Chen, Yi Mu;Guomin Yang, FuchunGuo, Xiaofen Wang 2016, "Dual-Server Public-Key Encryption With Keyword Search for Secure Cloud Storage", IEEE Transactions on Information Forensics and Security, Vol.11, No. 4, pp. 789-798, 2016.
[25] Xinming Huang, Wei Wang, "A Novel and Efficient Design for an RSA Cryptosystem with a Very Large Key Size", IEEE Transactions on Circuits and Systems II: Express Briefs, Vol. 62, No. 10, pp. 972-976, 2015.
[26] Iqra Mustafa, Imran Ullah Khan, SherazAslam, AhthashamSajid;Syed Muhammad Mohsin, Muhammad Awais,Muhammad Bilal Qureshi, "A Lightweight Post-Quantum Lattice-Based RSA for Secure Communications", IEEE Access, Vol. 8, 2020.
[27] Osama Fouad Abdel Wahab, Ashraf A. M. Khalaf, Aziza I. Hussein, Hesham F. A. Hamed, "Hiding Data Using Efficient Combination of RSA Cryptography, and Compression Steganography Techniques", IEEE Access, Vol. 9, No. 2, pp. 3105-31815, 2021.
[28] Hanlin Zhang, Jia Yu, ChengliangTian, Le Tong, Jie Lin, LinqiangGe, Huaqun Wang, "Efficient and Secure Outsourcing Scheme for RSA Decryption in Internet of Things", IEEE Internet of Things Journal, Vol. 7, No. 8, pp. 6868-6881, 2020.
[29] Damien Vergnaud, "Comment on "Efficient and Secure Outsourcing Scheme for RSA Decryption in Internet of Things", IEEE Internet of Things Journal, Vol. 7, No. 11, pp. 11327-11329, 2020.
[30] Victor Chang, Muthu Ramachandran, "Towards Achieving Data Security with the Cloud Computing Adoption Framework", IEEE Transactions on Services Computing, Vol. 9, No.1, pp. 138-151, 2016.