# A SECURE AND VERIFIABLE ACCESS STORAGE FOR BIG DATA IN CLOUDS

Syed Abdur Rahman[1], Syed Niyamathullah[2],
Mr. Pathan Ahmed Khan[3]
B.Tech Students[1,2], Assistant Professor[3]
Department of Computer Science &Engineering
Lords Institute of Engineering and Technology, Himayat Sagar, Telangana, India

**Abstract:**
The complexity and volume, outsourcing cipher texts to a cloud is considered to be one of the most effective approaches for big data storage and access. Nevertheless, verifying the access rules of a user and securely updating a cipher text in the cloud based on a new access policy designated by the data owner are two critical challenges to make cloud-based big data storage practical and effective. Traditional approaches either completely ignore the issue of access policy update or delegate the update to a third party authority; but in practice, access policy update is important for enhancing security and dealing with the dynamism caused by user join and leave activities. In this paper, we propose a secure and verifiable access control scheme based on the NTRU cryptosystem for big data storage in clouds. We first propose a new NTRU decryption algorithm to overcome the decryption failures of the original NTRU, and then detail our scheme and analyze its correctness, security strengths, and computational efficiency. Our scheme allows the cloud server to efficiently update the cipher text when a new access policy is specified by the data owner, who is also able to validate the update to counter against cheating behaviors of the cloud. It also enables (i) the data owner and eligible users to effectively verify the rules of a user for accessing the data, and (ii) a user to validate the information provided by other users for correct plaintext recovery. Thoroughly analysis indicates that our scheme can prevent eligible users from cheating and resist various attacks such as the collusion attack.
**Keywords:** Big Data Storage; Access Control; the NTRU Cryptosystem; Secret Sharing; Access Policy Update; Cloud Computing.

## I.INTRODUCTION

BIG data is a high volume, and/or high velocity, high variety information asset, which requires new forms of processing to enable enhanced decision making, insight discovery, and process optimization [1]. Due to its complexity and large volume, managing big data using on hand database management tools is difficult. An effective solution is to outsource the data to a cloud server that has the capabilities of storing big data and processing users' access requests in an efficient manner. For example in health applications, the genome information should be securely stored in an e-health cloud as a single sequenced human genome is around 140 gigabytes in size [2], [3]. However, when a data owner outsources its data to a cloud, sensitive information may be disclosed because the cloud server is not trusted; therefore typically the cipher text of the data is stored in the cloud.

But how to update the cipher text stored in a

cloud when a new access policy is designated by the data owner and how to verify the legitimacy of a user who intends to access the data are still of great concerns. Most existing approaches for securing the outsourced big data in clouds are based on either attributed-based encryption (ABE) or secret sharing. ABE based approaches [4]–[11] provide the flexibility for a data owner to predefine the set of users who are eligible for accessing the data but they suffer from the high complexity of efficiently updating the access policy and cipher text. Secret sharing [11]–[17] mechanisms allow a secret to be shared and reconstructed by certain number of cooperative users but they typically employ asymmetric public key cryptography such as RSA for users' legitimacy verification, which incur high computational overhead. Moreover, it is also a challenging issue to dynamically and efficiently update the access policies according to the new requirements of the data owners in secret sharing approaches.

As a data owner typically does not backup its data locally after outsourcing the data to a cloud, it cannot easily manage the data stored in the cloud. Besides, as more and more companies and organizations are using clouds to store their data, it becomes more challenging and critical to deal with the issue of access policy update for enhancing security and dealing with the dynamism caused by the users' join and leave activities. To the best of our knowledge, policy up date for outsourced big data storage in clouds has never been considered by the existing research[13],[17]–[20].

Another challenging issue is how to verify the legitimacy of the users accessing the outsourced data in clouds. Existing schemes proposed in [7], [8], [10], [21] do not support user eligibility verification. On the other hand, verifiable secret sharing based schemes rely on RSA [13]–[15] for access legitimacy verification. As multiple users need to mutually verify each other using multiple RSA operations, such a procedure has a high computational overhead. Furthermore, the classic asymmetric crypto solutions such as RSA could be broken by quantum computing in the near future. This is not a science fiction as in 2015 IBM brought quantum computing closer to reality [22], making it urgent to exploit new techniques for quantum computing attack resistance. The NTRU cryptosystem is a type of lattice-based cryptography [23]–[25], and its security is based on the shortest vector problem (SVP) in a lattice [26]. The major advantages of NTRU are quantum computing attack resistance and lighting fast computation capability. However, NTRU suffers from the problem of decryption failures [27]–[30].

The considerations mentioned above motivate us to develop a verifiable access control scheme for securing the big data stored in clouds, tackling the challenges of the following security services:

• Security. The proposed scheme should be able to defend against various attacks such as the collusion attack. Meanwhile, access policy update should not break the security of the data storage, disclose sensitive information about the data owner, and cause any new security problem.

• Verification. When a user needs to decrypt a

stored ciphertext, its access legitimacy should be verified by other participating users and the secret shares obtained from other users must be validated for correct recovery.

• Authorization. To reduce the risk of information leakage, a user should obtain authorization from the data owner for accessing the encrypted data.

## II. PROCEDURE

We consider a cloud storage system that is applicable for both public and private clouds as shown in Fig. 3. It consists of the following three types of entities: cloud server, data owner (owners), and data user (users). Cloud server. A cloud server provides spaces for data owners to store their outsourced ciphertext data that can be retrieved bythe users. It is also responsible for updating the ciphertexts when the data owner changes its access policy.

Owners: A data owner design a test he access policy for its data, encrypts the database do the access policy before outsourcing the data to the cloud server, and requests the cloud server to update the encrypted data when a new access policy is adopted. It can also check whether the ciphertext at the cloud server is correctly updated.

Users. Each user is assigned with a sub-key for an encrypted data the user is eligible to access. Inorder to decrypt the ciphertext, the user's eligibility must be verified by at least t−1 other users that are also eligible to access the data.

The information provided by the t − 1 verifiers must be validated by the user for correct message decryption based on the (t,n)-threshold secret sharing. For a piece of data to be stored in a cloud, the data owner generates a public key and privacy key pair, defines an access policy, and computes a sub-key for each potential user based on the policy.

Then, the data owner produces a message certificate for the data, and stores the encrypted data with the access policy in the cloud. When a user needs to use the data, it solicits help from other users to recover the data.

The cloud server can update the encrypted data with a new policy is designated by the data owner. The access policy is defined by a t − 1 degree polynomial b (x) = b0+Pt−1 j=1 bjxj in this paper. Specifically, the data owner splits the access right of an encrypted plaintext into n pieces for n users, with one piece for each legitimate user of the data.

A user can recover the plaintext data if and only if it obtains the message certificate from the data owner and holds at least t pieces of the access rights (with the help of at least t − 1 other legal users), where t is a threshold designated by the data owner for access control based on (t,n)-threshold secret sharing.

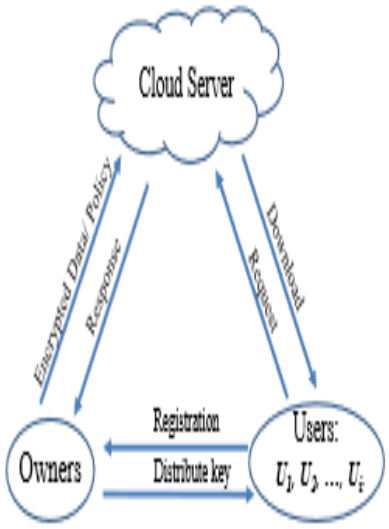


**Figure 1: System Architecture**

## III RELATED WORK

Because big data frequently contains a huge amount of personal identifiable information, how to securely store the data and how to provide access control over the stored data are two biggest challenges[32].Inthissubsection,wemainlysummarizethestateof-the-art of securing the big data stored in clouds. Outsourcing to clouds is one of the most popular approaches to securing the big data storage [33]–[35], in which the data owners encrypt their data based on cryptographic primitives and store the encrypted data to the clouds. In outsourcing, a secure mechanism should be established between a data owner and a cloud.

In order for the cloud to perform operations over the encrypted data, "Fully Homomorphic Encryption" (FHE) [36] was usually adopted, which allows direct addition and multiplication operations over the ciphertexts while preserving decryptability. Homomorphic encryption was also applied to guarantee the security of data storage [37], [38]. Nevertheless, it is an immature cryptosystem, and is extremely inefficient in practice, which renders it hardly applicable in real world applications.

Securely outsourcing big data computations to the clouds was also extensively studied[39]– [41] but this topic is out of the scope of the paper. Adequate access control is key to protect the stored data. Access control has traditionally been provided by operating systems or applications restricting access to the information, which typically exposes all the information if the system or application is hacked [42]. A better approach is to protect the information using encryption that only allows decryption by authorized entities. Attribute-Based Encryption(ABE)[4],[7],[8],[10],[11]is one of the most powerful techniques for access control in cloud storage systems. In the past years, quite a few attribute-based access control schemes [19], [20], [43], [44] were proposed, in which the data owners define the access policies based on the attributes required by the data and encrypt the data based on the access policies. By this way the data owners are able to ensure that only the users meeting the access policies can decrypt the ciphertexts. However, it is difficult to update the policies when these ABE based schemes are applied because the data owners do not store the data in their local systems once they outsource the data into the cloud servers. It is also difficult to verify the legitimacy of the downloaded data as the clouds housing the data are not trustworthy. Besides, the operations of encryption and decryption in ABE have a high computational overhead and incur a large

energy consumption. Secret sharing[45]is another powerful technique to protect the big data in cloud storage. The most related work to our proposed scheme are [15] and [14], whose verification procedure can resist potential attacks such as collusion and cheating. In [15], two schemes were proposed, namely Scheme-I and Scheme-II, based on the homogeneous linear recursion and the RSA cryptosystem, in which the homogeneous line a recursion is used to construct the secret share and reconstruct the secret, and RSA is used to verify the users' access legitimacy. The difference between these two schemes lies in that the users in Scheme-I mutually verify each other's legitimacy without seeking help from public values while in Scheme-II the users need the help of public values. In [14], the authors presented a verifiable multi-secret sharing scheme based on cellular automata, which is used to construct the secret share and reconstruct the secret with a linear computational complexity, and the RSA cyptosystem, which is used for verification. In these schemes, as multiple users mutually verify each other using multiple RSA operations, a very high computational overhead occurs. In addition, the classic asymmetric crypto solutions would be broken by quantum computing; that is, these traditional verification methods cannot satisfy the verification requirements with respect to quantum computing, which is made closer to reality by IBM in 2015 [22]. Thus we need to seek new verification methods to meet the future requirements. For this purpose, we utilize the NTRU cryptosystem to counter the quantum computing attacks in the design of our proposed scheme.

## IV PROPOSED SYSTEM

Let B be the set of users that are eligible to access the outsourced sensitive data in the cloud. Assume that Ui∈ B is a user who needs to use the data. According to the access policy, at least t − 1 other users in B should participate in the processes of verifying Ui's eligibility and obtaining the data for Ui. To achievethis objective, we propose a secure and verifiable access control scheme for the big data storage in a cloud server in this section. Our scheme consists of the following four stages:

i)Setup:the data owner initializes the system to generate the public and private keys via the improved NTRU cryptosystem;
ii) Construction: the data owner generates a sub-key for each user in B, produces a message certificate for each message, and stores the data securely in the cloud server;
iii) Reconstruction: the user Ui and t−1 other users in B mutually verify each other and work together to help Ui reconstruct the data;
iv) Policy update: the cloud server instead of the data owner updates the encrypted data with a new policy if needed.

5.1 Setup A data owner has a set of messages S = {S1,S2,...,SM} for cloud storage, with M being the total number of messages in S. At the setup stage, the data owner generates its public key h and private key f according to the improved NTRU cryptosystem and then initializes the system according to the process presented in algorithm.

5.2 Construction The data owner constructs a sub-key for each legitimate user in B, generates a certificate for each message in S, and stores the encrypted data into the cloud server.

5.2.1 Sub-Key Construction The data owner randomly generates t different integers b0,b1,⋯ ,bt−1, where bj∈ Z[X]/(XN − 1) for j = 0,1,⋯ ,t − 1, and uses them as coefficients to construct the following t−1 degree polynomial b(x):

b(x) = b0 +

t−1 X j=1

bjxj. (10)

Each user Ui chooses a random integer ri, encrypts ri using the data owner's public key h to get the ciphertext vi = pφ∗ h + ri (mod q), and then sends{idi,H(ri),vi}to the data owner. When the data owner receives the ciphertextvi, it decrypts it using its private key f to get the plaintext r0 i. Then the data owner checks the following two conditions: i) H(r0 i) = H(ri) for the user Ui; and ii) ri 6= rσ for the user Ui and any user Uσ who has received a sub-key from the data owner. If condition i) cannot hold, the message is falsified during transmission and it could be sent again; if condition ii) cannot hold, the data owner requests the user Ui to choose a different secret number and repeat the procedure to compute a sub-key for Ui.

5.3 Message Reconstruction Assume that a user Ui∈ B needs to get the message/data Sj. It downloads kj from the cloud server and then seeks help from other users in B to decrypt kj. This procedure consists of the following three stages:

i) Exchange certificate computation: Ui gets Sj's message certificate from the data owner and computes anexchangecertificate.ii)Certificateverification: otherusersinB and Ui mutually validate each other.

iii) Message reconstruction: Ui reconstructs the message Sj.

5.3.1 Exchange Certificate Computation Ui sends a request to the data owner to obtain Sj's message certificate dj.Up on receiving this request, the data owner encrypts dj using Ui's secret number ri based on AES as shown in (14).

Cdj = AESri(dj). (14)

Upon receiving the ciphertext Cdj from the data owner, Ui first decrypts Cdj to obtain dj. Then, it uses its sub-key xi to compute the exchange certificate Wij via (15) and sends Wij to other users in B. Wij = xi ∗dj.

## V MODULES

The considerations mentioned above motivate us to develop a verifiable access control scheme for securing the big data stored in clouds, tackling the challenges of the following security services:

Security: The proposed scheme should be able to defend against various attacks such as the collusion attack. Meanwhile, access policy update should not break the security of the data storage, disclose sensitive information about the data owner, and cause any new security problem.

Verification: When a user needs to decrypt a stored ciphertext, its access legitimacy should be verified by other participating users and the secret shares obtained from other users must be validated for correct recovery.

Authorization: To reduce the risk of information leakage, a user should obtain authorization from the data owner for accessing the encrypted data.

## VI CONCLUSIONS

In this paper, we first propose an improved

NTRU cryptosystem to overcome the decryption failures of the original NTRU and then present a secure and verifiable access control scheme based on the improved NTRU to protect the outsourced big data stored in a cloud. Our scheme allows the data owner to dynamically update the data access policy and the cloud server to successfully update the corresponding outsourced cipher text to enable efficient access control over the big data in the cloud. It also provides a verification process for a user to validate its legitimacy of accessing the data to both the data owner and t□1 other legitimate users and the correctness of the information provided by the t□1 other users for plaintext recovery. The security of our proposed scheme is guaranteed by those of the NTRU cryptosystem and the threshold secret sharing. We have rigorously analyzed the correctness, security strength, and computational complexity of our proposed scheme. Designing a secure, privacy preserving, and practical scheme for big data storage in a cloud is an extremely challenging problem. In our future research, we will further improve our scheme by combining threshold secret sharing with attribute based access control, which involves an access structure that can place various requirements for a user to decrypt an outsourced cipher text data in the cloud. Meanwhile, we will investigate the security problems when a data owner outsources its data to multi cloud servers and consider an attribute-based access structure that can be dynamically updated, which is more applicable for practical scenarios in big data storage.

## VII REFERENCES

[[1] M. A. Beyer and D. Laney, "The importance of big data: a definition," Stamford, CT: Gartner, 2012.

[2] V. Marx, "Biology: The big challenges of big data," Nature, vol. 498, no. 7453, pp. 255–260, 2013.

[3] G. P. Consortium et al., "A map of human genome variation from population scalesequencing,"Nature,vol.467,no.7319,pp.1061–1073, 2010.

[4] A. Sahai and B. Waters, "Fuzzy identity-based encryption," Advances in Cryptology–EUROCRYPT 2005, pp. 457–473, 2005.

[5] C. Hu, F. Zhang, X. Cheng, X. Liao, and D. Chen, "Securing communications between external users and wireless body area networks," in Proceedingsofthe2ndACMworkshoponHottopicsonwirelessnetwork security and privacy. ACM, 2013, pp. 31–36.

[6] C. Hu, H. Li, Y. Huo, T. Xiang, and X. Liao, "Secure and efficient data communication protocol for wireless body area networks," IEEE Transactions on Multi-Scale Computing Systems, vol. 2, no. 2, pp. 94– 107, 2016.

[7] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM conference on Computer and communications security. ACM, 2006, pp. 89–98.

[8] B. Waters, "Ciphertext-policy attribute-

based encryption: An expressive, efficient, and provably secure realization," Public Key Cryptography– PKC 2011, pp. 53–70, 2011.

[9] C. Hu, N. Zhang, H. Li, X. Cheng, and X. Liao, "Body area network security: a fuzzy attribute-based signcryption scheme," IEEE journal on selected areas in communications, vol. 31, no. 9, pp. 37–46, 2013.

[10] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," Advances in Cryptology–EUROCRYPT 2011, pp. 568–588, 2011.