A Secure and Verifiable Access Control Scheme for Big Data Storage in Clouds

D.Bhavana
Department of Computer Science and Engineering, Narayana Engineering College Gudur
bhavanadevireddy24@gmail.com

Y.Vinay Kumar
Asso.Professor, Department of Computer Science and Engineering, Narayana Engineering College Gudur
vinay@narayanagroup.com

N.Yedukondalu
Asst.Professor, Department of Computer Science and Engineering, Narayana Engineering College
Gudursevenhills.559@gmail.com

D.Rishitha
Department of Computer Science and Engineering, Narayana Engineering College
Gudurrishitadodda20@gmail.com

M.Harika
Department of Computer Science and Engineering, Narayana Engineering College
Gudurharikamanublou98@gmail.com

ABSTRACT

Due to the complexity and volume, extracting cipher documents to the cloud is considered one of the most effective ways to store data and access. However, ensuring the authenticity of accessing and securely updating the ciphertext to the cloud-based on the new data access policy developed by the data owner is two of the key challenges of making cloud-based big data storage efficient and effective. Traditional methods completely ignore the issue of access policy review or pass an update to a third party's authority; but in practice, renewal of the access policy is essential to improve security and address the flexibility caused by a user joining and leaving operations. In this paper, we propose a secure and secure control system based on the NTRU cryptosystem for large-scale data storage in the cloud. We first propose a new NTRU design algorithm to overcome the initial NTRU design failures, and then specify our program and analyze its accuracy, security capabilities, and computational efficiency. Our scheme allows the cloud server to successfully update ciphertext when the new access policy is defined by the data owner, who is able to verify the update to counter the cloud behavior. It also provides (i) the data owner and relevant users to successfully verify the user's access to data, and (ii) the user verifies the information provided by other users for direct recovery.

Keywords: Secret Sharing; Access Policy Update; the NTRU Cryptosystem; Cloud Computing.

1. INTRODUCTION

BIG data is a high volume, and/or high-speed, high-quality data property, which requires new processing techniques to enable improved decision-making, insight detection, and process optimization [1]. Due to its complexity and large volume, large data management using database management tools An effective solution is to extract data from a cloud server that has the ability to store big data and apply user access requests effectively[7].
Many of the possible ways to protect big data used in the cloud are based on database-based encryption (ABE) or private sharing. ABE-based methods provide data owner flexibility to specify a set of users who are eligible for data access but suffer from a high complexity of properly updating access policy and functionality[2],[6]. Secret sharing mechanisms allow a secret to be shared and reconstructed by a certain number of cooperative users but they typically employ asymmetricpublic key cryptography such as RSA for users' legitimacy verification, which incurs 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 back up their data locally after sending data to the cloud, it cannot easily manage data stored in the cloud. Besides, as more companies and organizations use the cloud to store their data, it becomes more difficult and more important to tackle the problem of policy security consolidation and address the huge forces created by users to join and leave jobs. To the best of our knowledge, reviewing the policy on posts big data storage in clouds has never been considered by the existing research [3], [4], [5], [8].

2. LITERATURE SURVEY

The big data frequently contains a huge amount of personally identifiable information, how to securely store the data, and how to provide access control over the stored data are two biggest challenges. In this survey, we mainly summarize the state-of-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, 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) 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 and which renders it hardly applicable in real-world applications.

Secret sharing is another powerful technique to protect big data in cloud storage. The most related work to our proposed scheme, whose verification procedure can resist potential attacks such as collision and cheating. Two schemes were proposed, namely Scheme-I and Scheme-II, based on the homogeneous linear recursion and the RSA cryptosystem, in which the homogeneous linear 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 addition, classic asymmetric crypto solutions will break down in quantum computing; that is, these traditional authentication methods cannot satisfy the requirements for authentication in relation to a quantum computer, made real by IBM in 2015.For this purpose, we use the NTRU cryptosystem to counter the quantum computing attacks in the design of our proposed scheme.

3. METHODOLOGY

In this methodology consists of System Model, Cloud Server, Owners, Users and Improved NTRU Cryptosystem.

A. System Model

We consider a cloud storage system that is applicable for both public and private clouds as shown in Fig. 1. It has the following three types of objects: cloud server, data owner (owners), and data user (users).

B. Cloud server

A cloud server provides spaces for data owners to store their outsourced ciphertext data that can be retrieved by the users. It is also responsible for updating ciphertexts when the data owner changes the access policy.

C. Owners

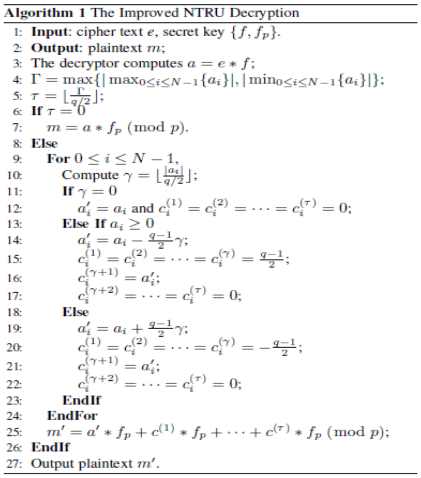
A data owner designates the access policy for its data, encrypts the data based on the access policy before outsourcing the data to the cloud server, and requests the cloud server to update the encrypted data when a new access policy is adopted. It can also check whether the ciphertext at the cloud server is correctly updated.

D. Users

Each user is assigned a sub-key for encrypted data the user is eligible to access. In order to decrypt the ciphertext, the user’s eligibility must be verified by at least t - 1 other user 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.

E. Improved NTRU Cryptosystem

There exist two types of decryption failures in NTRU: the Wrap failure and the Gap failure. In other words, the decryption process of NTRU cannot always output a correct decrypted message. To overcome this problem, we propose an improved NTRU cryptosystem is shown in Fig 1.



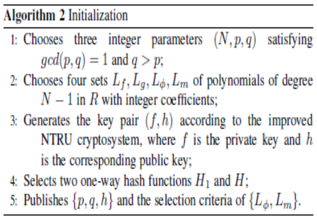
```
Algorithm 1 The Improved NTRU Decryption
Input: cipher text $e$, secret key $\{f, f_p\}$.
Output: plaintext $m$;
The decryptor computes $a = e * f$;
$\Gamma = \max\{|\max_{0\leq i\leq N-1}\{a_i\}|, |\min_{0\leq i\leq N-1}\{a_i\}|\}$;
$\tau = \lfloor \frac{\Gamma}{q/2} \rfloor$;
If $\tau = 0$
    $m = a * f_p \pmod{p}$.
Else
    For $0 \leq i \leq N-1$,
        Compute $\gamma = \lfloor \frac{|a_i|}{q/2} \rfloor$;
        If $\gamma = 0$
            $a_i' = a_i$ and $c_i^{(1)} = c_i^{(2)} = \cdots = c_i^{(\tau)} = 0$;
        Else If $a_i \geq 0$
            $a_i' = a_i - \frac{q-1}{2}\gamma$;
            $c_i^{(1)} = c_i^{(2)} = \cdots = c_i^{(\gamma)} = \frac{q-1}{2}$;
            $c_i^{(\gamma+1)} = a_i'$;
            $c_i^{(\gamma+2)} = \cdots = c_i^{(\tau)} = 0$;
        Else
            $a_i' = a_i + \frac{q-1}{2}\gamma$;
            $c_i^{(1)} = c_i^{(2)} = \cdots = c_i^{(\gamma)} = -\frac{q-1}{2}$;
            $c_i^{(\gamma+1)} = a_i'$;
            $c_i^{(\gamma+2)} = \cdots = c_i^{(\tau)} = 0$;
        EndIf
    EndFor
    $m' = a' * f_p + c^{(1)} * f_p + \cdots + c^{(\tau)} * f_p \pmod{p}$;
EndIf
Output plaintext $m'$.
```

Figure 1. Improved NTRU Decryption

Let B be the set of users that are eligible to access the outsourced sensitive data in the cloud. Assume that Ui 2 B is a user who needs to use the data. According to the access policy, at least t - 1 other user in B should participate in the processes of verifying Ui’s eligibility and obtaining the data for Ui. To achieve this 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.

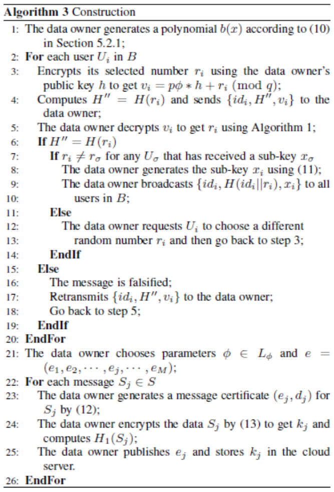
A data owner has a set of messages S = {S_1, S_2,…S_M} for cloud storage, with M being the total number of messages in S. At the setup stage, the data owner generates its public key h and private key f according to the improved NTRU cryptosystem and then initializes the system according to the process presented in Fig2.



```
Algorithm 2 Initialization
Chooses three integer parameters $(N, p, q)$ satisfying $gcd(p, q) = 1$ and $q > p$;
Chooses four sets $L_f, L_g, L_\phi, L_m$ of polynomials of degree $N - 1$ in $R$ with integer coefficients;
Generates the key pair $(f, h)$ according to the improved NTRU cryptosystem, where $f$ is the private key and $h$ is the corresponding public key;
Selects two one-way hash functions $H_1$ and $H$;
Publishes $\{p, q, h\}$ and the selection criteria of $\{L_\phi, L_m\}$.
```

Figure 2. Initializes the system

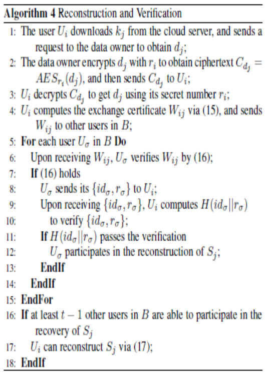
The data owner constructs a sub-key for each legitimate user in B, generates a certificate for each message in S, and stores the encrypted data into the cloud server is shown in Fig 3.



```
Algorithm 3 Construction
1: The data owner generates a polynomial $b(x)$ according to (10) in Section 5.2.1;
2: For each user $U_i$ in $B$
3:     Encrypts its selected number $r_i$ using the data owner's public key $h$ to get $v_i = p\phi * h + r_i \pmod q$;
4:     Computes $H'' = H(r_i)$ and sends $\{id_i, H'', v_i\}$ to the data owner;
5:     The data owner decrypts $v_i$ to get $r_i$ using Algorithm 1;
6:     If $H'' = H(r_i)$
7:         If $r_i \neq r_\sigma$ for any $U_\sigma$ that has received a sub-key $x_\sigma$
8:             The data owner generates the sub-key $x_i$ using (11);
9:             The data owner broadcasts $\{id_i, H(id_i||r_i), x_i\}$ to all
10:                users in $B$;
11:        Else
12:            The data owner requests $U_i$ to choose a different
13:                random number $r_i$ and then go back to step 3;
14:        EndIf
15:    Else
16:        The message is falsified;
17:        Retransmits $\{id_i, H'', v_i\}$ to the data owner;
18:        Go back to step 5;
19:    EndIf
20: EndFor
21: The data owner chooses parameters $\phi \in L_\phi$ and $e = (e_1, e_2, \cdots, e_j, \cdots, e_M)$;
22: For each message $S_j \in S$
23:     The data owner generates a message certificate $(e_j, d_j)$ for $S_j$ by (12);
24:     The data owner encrypts the data $S_j$ by (13) to get $k_j$ and computes $H_1(S_j)$;
25:     The data owner publishes $e_j$ and stores $k_j$ in the cloud server.
26: EndFor
```

Figure 3. Construction stage

The processes of reconstruction and verification are summarized in Fig 4.



```
Algorithm 4 Reconstruction and Verification
The user $U_i$ downloads $k_j$ from the cloud server, and sends a
    request to the data owner to obtain $d_j$;
The data owner encrypts $d_j$ with $r_i$ to obtain ciphertext $C_{d_j} =$
    $AES_{r_i}(d_j)$, and then sends $C_{d_j}$ to $U_i$;
$U_i$ decrypts $C_{d_j}$ to get $d_j$ using its secret number $r_i$;
$U_i$ computes the exchange certificate $W_{ij}$ via (15), and sends
    $W_{ij}$ to other users in $B$;
For each user $U_\sigma$ in $B$ Do
  Upon receiving $W_{ij}$, $U_\sigma$ verifies $W_{ij}$ by (16);
  If (16) holds
    $U_\sigma$ sends its $\{id_\sigma, r_\sigma\}$ to $U_i$;
    Upon receiving $\{id_\sigma, r_\sigma\}$, $U_i$ computes $H(id_\sigma||r_\sigma)$
      to verify $\{id_\sigma, r_\sigma\}$;
    If $H(id_\sigma||r_\sigma)$ passes the verification
      $U_\sigma$ participates in the reconstruction of $S_j$;
    EndIf
  EndIf
EndFor
If at least $t-1$ other users in $B$ are able to participate in the
    recovery of $S_j$
  $U_i$ can reconstruct $S_j$ via (17);
EndIf
```

Figure 4. Reconstruction and verification Stage

The data owner can dynamically control the access of its data by updating the access policy defined for the data. Intuitively, as the ciphertext of the data is stored in the cloud server, the data owner needs to download the ciphertext, decrypt it to get the plaintext, and then re-encrypt the plaintext according to the new access policy. This is the approach adopted by all existing research. In this subsection, we consider updating the encrypted data at the cloud server based on the new access policy, which differs significantly from the traditional approaches.

4. RESULTS AND DISCUSSION

Home page of A Secure and verifiable access control scheme for big data storage in clouds is shown in Fig 5.

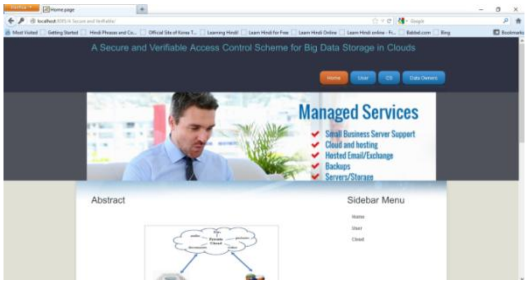


Figure 5. Home page of Secure and Verifiable control scheme

User login page is shown in Fig 6; here the user needs to give correct user id and password to login.

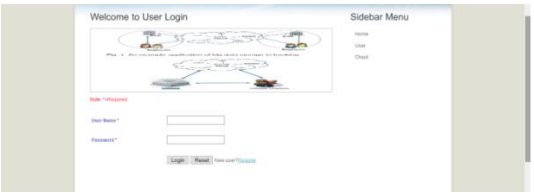
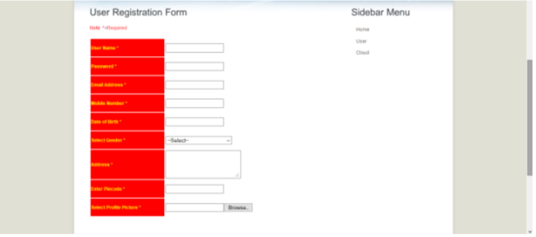


Figure 6. Login Form of user

End user registration page is shown in Fig 7; here the user needs to give necessary information to get registered.



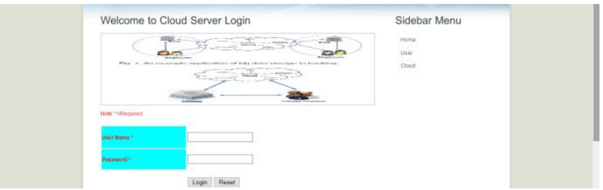
User Registration Form

Note: *=Required

User Name *	
Password *	
Email Address *	
Mobile Number *	
Date of Birth *	
Select Gender *	--Select--
Address *	
Enter Pincode *	
Select Profile Picture *	Browse..

Figure 7. Registration Form of user

Cloud Server login page is shown in Fig 8; here the cloud service owner needs to give correct user id and password to login.



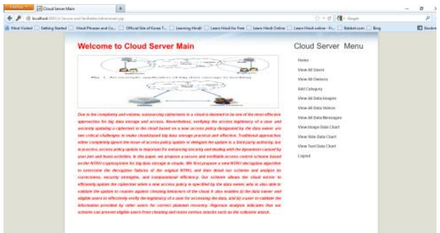
Welcome to Cloud Server Login

Note: *=Required

User Name *	
Password *	

Figure 8. Login Form of Cloud Server

Home page of Cloud Server and the categories of cloud server menu are shown in Fig 9.

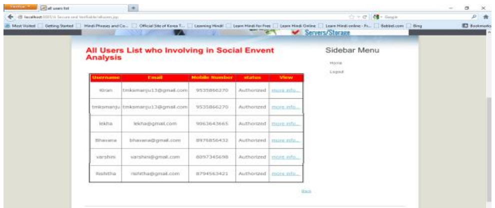


Welcome to Cloud Server Main

Due to the complexity and volume, outsourcing ciphertexts to a cloud is deemed to be one of the most effective approaches for big data storage and access. Nevertheless, verifying the access legitimacy of a user and securely updating a ciphertext 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 ciphertext 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 legitimacy of a user for accessing the data, and (ii) a user to validate the information provided by other users for correct plaintext recovery. Rigorous analysis indicates that our scheme can prevent eligible users from cheating and resist various attacks such as the collusion attack.

Figure 9. Home page of Cloud Server Main

All end users are shown in Fig 10 ; in this the cloud service owner can view all the end users, and their details.

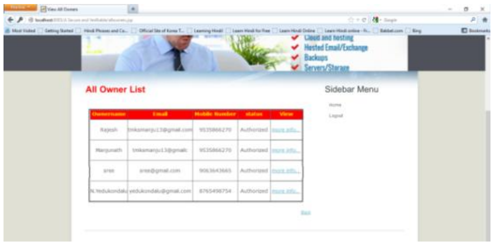


All Users List who Involving in Social Envent Analysis

Username	Email	Mobile Number	status	View
Kiran	tmksmanju13@gmail.com	9535866270	Authorized	more info...
tmksmanju	tmksmanju13@gmail.com	9535866270	Authorized	more info...
lekha	lekha@gmail.com	9963643665	Authorized	more info...
Bhavana	bhavana@gmail.com	8976856432	Authorized	more info...
varshini	varshini@gmail.com	8897345698	Authorized	more info...
Rishitha	rishitha@gmail.com	8794563421	Authorized	more info...

Figure 10. View of all end users in the Cloud Server

All Data Owners are shown in Fig 11 in this the cloud service owner can view all the data owners, and their details.



All Owner List

Ownername	Email	Mobile Number	status	View
Rajesh	tmksmanju13@gmail.com	9535866270	Authorized	more info...
Manjunath	tmksmanju13@gmail.	9535866270	Authorized	more info...
sree	sree@gmail.com	9063643665	Authorized	more info...
N.Yedukondalu	yedukondalu@gmail.com	8765498754	Authorized	more info...

Figure 11. View of all data owners in the Cloud Server

5. CONCLUSION

In this paper, we first propose to optimize the NTRU cryptosystem to overcome the inherent NTRU design failures and present a secure and validated control system based on the advanced NTRU to protect large data sent to the cloud. Our scheme allows the data owner to dynamically update the data access policy with the cloud server to successfully update the corresponding compatible cipher text to enable effective access to large data management in the cloud. It also provides a user authentication process to verify its data integrity for both the data owner and the correct t-1 users and for the accuracy of information provided by other t-1 users for recovery.

REFERENCES

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

[2] V. Goyal, O. Pandey, A. Sahai, and W. Waters, "They provide privacy in the proper management of the contents of written information," in Proceedings of the 13th ACM conference on computer security and communications. ACM, 2006, p. 89-98.

[3] C. Hu, X. Liao, and X.Cheng, "Sharing Private Data Based on LFSR Sequences," Theoretical Computer Science, vol. 445, 2012.

[4] C. Hu, X. Liao, and D. Xiao, "Sharing of private photos based on a map of chaos and fossils in China," International Journal of Wavelets, Multiresolution and Information Processing, vol. 10, no. 03, 2012.

[5] V. Marx, "Biology: The Big Challenges of Big Data," Nature, vol. 498, no. 7453, p. 255-2260, 2013.

[6] A. Sahai and B.Wers, "Encryption Based on Good Identity," Advances in Cryptology- EUROCRYPT 2005, p. 457- 473, 2005.

[7]M.Geeta Bhargava, P.Vidyullath,, P.Venkateswara Rao, V.Sucharita "A study on potential of big visual data analytics in construction arena", International Journal of Engineering & Technology, 7 (2.7) (2018) 652-656

[8]K. Yang, X. Jia, K. Ren, B. Zhang, and R. Xie, "Dac-macs: Active data access control for multiple cloud storage systems," Forensics and Security Information, IEEE Transaction on, vol. 8, no. 11,p. 1790-1801, 2013.