International Journal of Engineering & Technology, 7 (3.34) (2018) 606-610

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Key Generation Techniques to Ensure User Data Integrity in Cloud Environments

Yoon-Su Jeong[1], Sang-Ho Lee[2]*

[1]Department of Information and Communication Convergence Engineering, Mokwon University, 88 Doanbuk-ro, Seo-gu, Daejeon, 302-729, Korea

*[*2]Department of Software, Chungbuk National University, Chungdae-ro 1, Seowon-Gu, Cheongju, Chungbuk 28644, Korea,*

**Corresponding author E-mail: shlee@cbnu.ac.kr*

Abstract

Background/Objectives: Cloud services are becoming popular with many users as they provide services based on the Internet. Users who use cloud services can integrate computing resources such as hardware and software, which exist in intangible form, through virtualization technology, and there is a great demand for security technologies related to security problems.

Methods/Statistical analysis: As a result of the evaluation, the proposed method in the security evaluation and the performance evaluation resulted in better data integrity and security than the existing method. In addition, we checked the integrity of different cloud data and obtained the efficiency improved by O (logn) than the existing method.

Findings: In this paper, we propose a robust data integrity protection scheme for various security attacks in the cloud environment. The proposed method effectively guarantees the integrity of the data used by the user through the generation and processing of low-load keys between the TPA, the user and the KGC. To protect the integrity of the data transmitted and received in the cloud environment, the proposed method generates the key through three processes (data generation process, encryption key generation process, and metadata attribute key pair generation process).

Improvements/Applications: The key generated in this process is used by the anonymous key so that sensitive information of the cloud user is not exposed to a third party so that the important information of the user is not remembered. In addition, the proposed scheme keeps synchronization between the TPA and the user at a predetermined time interval so that the important information of the user is not illegally exploited from the third party.

Keywords: *Cloud, Data Integrity, Key generation, Hash, Security*

1. Introduction

Recently, cloud service has attracted much attention from users by providing IT services and computer resources on the Internet [1]. Cloud services have received widespread attention because of the development of computer technology that integrates computing resources, such as hardware and software that exist in intangible forms that exist in different physical locations in tandem with strategic investment strategies of global corporations through virtualization technology. This is because [2].The cloud service is exposed to security threats and new types of security threats existing in the existing system, and when the server is hacked, personal information can be leaked and the data can't be used in case of server failure [3].

Recently, as cloud services have expanded to mobile clouds, users are using various device platforms and operating systems. To date, enterprises have embraced security technologies for cloud security threats, and companies are burdened with protecting the critical data they hold. Patching a huge virtual server is never easy, and hackers can hijack the server, block traffic, and provide room to steal data from vulnerable systems [4].

Most cloud services are limited to specific terminals and cloud services are not supported smoothly. Therefore, it is difficult to develop existing cloud services separately for mobile service. In the future mobile cloud era, the demand for services to be shared by various terminal devices will be increased. For this, an independent service of terminals is required [5, 6].

In this paper, we propose a method to protect user 's data from various security attacks when a user receives a cloud service. The purpose of the proposed technique is to effectively guarantee the integrity of the data used by users through key generation and processing with low load between TPA, user and KGC. The proposed method generates the key through three processes to protect the data transmitted and received in the cloud environment. The first step is to select key data information I(=□^{n}(s)) for generating the key through $_{R}\{0,1\}^{n}$. The second step creates a public / private key that is used to securely encrypt and decrypt the data.Finally, the third step is to generate a key pair suitable for the data attribute to securely process the user's data. The proposed scheme uses anonymous key that can be used without memorizing user 's important information so as not to expose sensitive information of cloud users to a third party. In addition, the proposed scheme keeps the synchronization of the TPA and the user at a predetermined interval so that the important information of the user is not illegally exploited from the third party, thereby preventing the data integrity of the user. The composition of this paper is as follows. In Chapter 2, we discuss the existing research on data integrity in the cloud environment. In Section 3, we propose a data integrity protection scheme suitable for the cloud

environment. In Section 4, we compare and evaluate the proposed scheme and the existing scheme. Finally, we conclude in Section 5.

2. Related Works

Recently, many researchers have been studying the confidentiality and integrity of user data to securely protect users' data in the cloud environment. In particular, while TPA considered designing to improve security and fairness for data auditing, researchers have been working to support additional security features in the cloud.

Q. Wang et al. [7] proposed an original remote data integrity assurance protocol through a public audit protocol for publicly auditing dynamic data [7]. This technique allows the data to be dynamically processed using the classic Merkle hash tree. In particular, this technique used a bilinear aggregate signature to support multiple audit tasks simultaneously.

Wang et al. technique proposed TPA system for public audit [8]. To guarantee privacy that supports data integrity audit, this technique integrates random masking technology when designing auditability protocol to support batch audit to prevent TPA from obtaining user's data contents during data audit process made.

Li et al. [9] proposed an encryption technique that supports integrity audit and data replication in a cloud environment. This technique supports authorized and fine-grained renewal requests, so that a third party can masquerade as an auditor or prevent an unauthorized attack by malicious auditors.

K. Huang et al. Technique has proposed a feedback-based auditing scheme that assumes that TPA is trustworthy [10]. This technique has the advantage of protecting user's data privacy against malicious TPA. However, this technique uses a blind technique to mask the challenge proof, but there is a problem that the TPA can not verify the validity of the masked proof.

K. Huang et al. technique proposed a feedback-based auditing method that can protect user's data privacy against malicious TPA [10]. However, this technique uses the blind technique to support the proof of the challenge to the mask, but there is a problem that the TPA can not clearly verify the masked proof.

Schwarz et al. [11] proposed an algebraic signature for checking the data ownership of a remote server. This technique uses the algebraic attribute of the hash function to verify the data ownership of the server.

Luo et al. Technique has proposed a technique for checking the possession of cloud data using a trusted third party proxy [12]. Although this technique has the advantage of efficiently checking the data ownership of the remote server, it has a disadvantage that it can not be directly used for cloud computing.

Yang et al. technique proposes an audit protocol in which data of many users using various services in a cloud environment are normally supported by users [13]. This technique uses a cipher method and bilinear pairings to increase the security over the random masking used in [18] to ensure data privacy. In particular, this technique uses a data block index table to respond to batch audit and reply attacks.

Wang et al. technique has proposed a distributed storage integrity a MAA technique [14]. This technique uses a homomorphic token and distributed assurance code data to detect the location of the data error server as well as modify the cloud data. However, this technique assumes that all the workers trust the TBA, but has a low satisfaction in the real world.

Mokadem et al. technique has proposed a technique using an algebraic signature in the data structure to expand the distributed network by checking the files of the nodes located in the distributed network (grid or P2P) [15].

3. Data Integrity Protection Techniques for Cloud Environments

3.1. Overview

The proposed scheme proposes a key generation scheme to protect the data transmitted and received in the cloud environment. The proposed method divides the key for ensuring the integrity of data into three stages. The first step is to select key data information I(=□^{n}(s))for generating the key through $_{R}\{0,1\}^{n}$. The second step creates a public / private key that is used to securely encrypt and decrypt the data. Finally, the third step is to generate a key pair suitable for the data attribute to securely process the user's data. In the proposed method, the entire operation process for data integrity verification is divided into a setup process, an extract process, and a process of generating a metadata attribute key pair.

3.2. System Model

Figure 1 shows the network structure for data storage in the cloud environment that is processed by the proposed method. The components in Figure 1 are largely composed of four users: Cloud Storage Server (CSS), Third Party Auditor, and Key Generate Center.

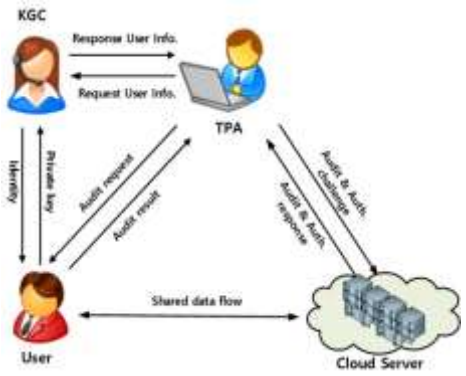

Figure 1: System model operation process

User

The user is responsible for sending personal information or data to the server and uses the application provided in the cloud environment. The user accesses the cloud through a private or wireless network using a personal portable device (mobile device, fixed device, etc.). Users have lower storage and transmission capabilities than servers at different locations.Some users of the cloud service can perform malicious actions.

Cloud Storage Server (CSS)

The cloud storage server has a high storage capacity and computation capability, and can provide users with sufficient resources. A cloud server consists of enough resources and a distributed server, and can provide different applications and data to the user according to the requirements of the server. However, the personal information or sensitive information stored in the cloud server may be exposed to the cloud environment so that a malicious user can acquire or expose the original data or confidential information of the normal user.

Third Party Auditor

The TPA is a trusted component for verifying the integrity of the data requested by the cloud user, and verifies the data integrity using the user's dual identifier and key pair.

Key generation center

The key generation center divides the identifier generated by the secret key of the cloud user into arbitrary size and generates an arbitrary dual identifier to be used in the cloud serverand the TPA. Then, a private key corresponding to the public key of the cloud user is generated to generate one key pair.

3.3. Requirement

The proposed method should satisfy the following requirements.

3.3.1. Correctness

Accuracy specifies that the evidence generated by the cloud response algorithm can be accommodated with an overwhelming probability.

3.3.2. Soundness

Soundness requires that it be stored in a strong challenge file so that it can generate proofs that can pass the data integrity verification algorithm.

3.3.3. Collusion Resistance

The authority to audit cloud data among one or more individuals in the cloud is grouped together to minimize collision resistance.In the normal model, the third party may perform an Extract query to query the private key of the selected attribute only if the selected set of challenge attributes is less than d.

3.3.4. Attribute Privacy-Preserving

In the cloud data audit phase, the TPA must not be able to infer the user-selected set of characteristics in order to upload files other than the common characteristics selected on the cloud server.The TPA can only process the user attribute at the intersection with the d attribute if it can guess in the response process.

3.3.5. Public Auditability

To be able to see public auditability in the cloud environment, the TPA should not be able to obtain certain data from the cloud or user. Audit protocol is one of the most important protocols when supporting partial data audit rather than whole data audit.Here, the whole data audit means that all data used in the cloud environment is requested by the user through the TPA, and the partial data audit means requesting a part of data used in the cloud through the TPA.The data audit process must process the original data in advance through the user's required security parameters. If this is not done, additional computation or storage burden will be incurred in the data integrity audit process.

3.3.5. Data Confidentiality

Data confidentiality ensures that data can not be obtained from the TPA or the integrity checking scheme handled by the cloud.

3.3.6. Privacy Preserving and Batch Auditing

Personal information protection should avoid parts that can expose the user's identity information. In particular, audit schemes should protect the user's personal information. Batch audits can be extended to support batch audits because they can not facilitate the dissemination and application of audit plans.

3.4Generate Data Protection Key Used in Integrity Verification Process

In the proposed scheme, the process of verifying the user's data integrity consists of three processes: data generation process, encryption key generation process, and metadata attribute key pair generation process.

3.4.1. Data Generation Process

This process is a step of generating the user data used in the cloud environment and the user's important data information I (=$\square^n$(s)) is expressed by $_R\{0,1\}^n$as equation (1).

$$\square^n(s)=\begin{cases} a_{n(s-1)} + a_{(n-1)(s-1)} & ,ifs, n > 1 \\ 1 & ,otherwise \end{cases} \quad (1)$$

Where, n denotes the number of important information of the user, and s denotes the seed of the important information of the user.

3.4.2. Encryption Key Generation Process

This process verifies that the user's data is provided by the integrity. The user's private key is the number of random numbers x_1^0 and x_1^1 to serve as the private key from $_R\{0,1\}^n$as equation (2).

$$x_i^{\delta[i]}=\begin{cases} x_i^0 & ,if\delta[i]=0 \\ x_i^1 & ,if\delta[i]=1 \end{cases} \quad (2)$$

Where, δ[i] is a binary value for generating the i-th selected private key (x_1^0, x_1^1).
If random numbers x_1^0 and x_1^1 are generated as a private key as in Eq. (2), a public key corresponding to the random numbers x_1^0 and x_1^1 is generated in the same manner as in equation (3).

$$PK_{i+1}^{\delta} = \mathrm{h}(x_{i+1}^{\delta})=\begin{cases} PK_{i+1}^{\delta} = h(x_{i+1}^{\delta}) & ,if\delta=0 \\ PK_{i+1}^{\delta} = h(x_{i+1}^{\delta}) & ,if\delta=1 \end{cases} \quad (3)$$

3.4.3. Metadata Attribute Key Pair Generation Process

Metadata attribute key pair generation process is a process of generating an attribute key pair of data to help data integrity.
In this process, the attribute key for the user's important information is generated. Two hash values are used to verify the data integrity of the user. The first hash value is used for access control of the user and the second hash value is used for the access right to process the encryption / decryption key of the user data. At this time, the generator repeats the process sequentially until the value of n is zero.
In this process, an attribute key for the user's important information is generated. The key pair generation process performs the following two processes to verify the data integrity of the user by using two hash values. In the first case, the generated hash value is used for user access control. In the second case, the hash value is used as the access right for encrypting / decrypting the user data. The generator repeats the process sequentially until the value of n is zero.

3.5Attorney-based Signature Process

In the proposed scheme, anonymous ID and random number are applied to a hash function between a user and a server in a cloud environment and a power of attorney is used.

3.5.1. Initialization Process

• Step 1: The user encrypts the user ID_i of the user using the attribute information $p'(=p_1, p_2,\cdots, p_n)$ of the data processed in the cloud environment as equation (4) and transmits it to the KGC.

$$\mathrm{Transfer}E_K \quad (\quad ID_i \quad , \quad p' \quad) \quad (4)$$

Where K is the shared key shared between the user and the KGC.
• Step 2 : KGC decrypts the received information and generates two random keys (r_1, r_2) as shown in equation (5).

$$\mathrm{Generate}\ (r_1, r_2) \in Z_q^* \quad (5)$$

3.5.2. Delegation Process

This process is a process in which the KGC transmits a power of attorney to the TPA that verifies the user's data, and consists of three stages of delegating the signature authority of the user between the server and the user.

. Step 1: The user generates a power of attorney m_i (i $\in Z^*$) containing information related to the proxy signature, such as the authority to sign or the expiration date, as shown in Equation (6) and sends it to the KGC.

$$\text{Generate} m_i \quad (\quad i \quad \in \quad Z^* \quad) \tag{6}$$

. Step 2 : The user generates a signature Sig to be used for the mandate m_i as Equation (7).

$$\text{Sig} = (-1)^{x_i^0} \cdot e^{x_i^1} \cdot \text{H}(m_i, \text{T}) \bmod \tag{7}$$

Where, T is selected through a random number r$\in Z_N$ selected by the KGC and then computed by the same calculation as T=r^2 mod N.

. Step 3 : The user encrypts the mandatory m_i and signature Sig as shown in Equation (8) to the KGC, and the TPA verifies the user's authority and rating. In this case, Sig, T, x_i^0, x_i^1, etc., other than the mandate m_i are encrypted with the public key PK_{i+1}^{δ} as shown in Equation (8).

$$\text{Transfer} E_{PK_{i+1}^{\delta}} (m_i, \text{Sig}, \text{T}, x_i^0, x_i^1) \tag{8}$$

3.5.3.Signature Process

The TPA sends the proxy signature information to the TPA and then uses the signature to verify the user's data integrity.

. Step 1: The TPA checks the user's important data information I(=□^{n}(s)) received from the user and computes a random number R (=r^2 modN) using a random integer r$\in Z_N$.

. Step 2 : The TPA generates random keys as Eqs. (9) and (10) to perform the proxy signature of the user on the data integrity of the user.

$$r_1 = \text{Sig} \cdot \text{R} \bmod \text{N} \tag{14}$$

$$r_2 = (-1)^{x_i^0} \cdot e^{x_i^1} \cdot \text{H(m, T)} \bmod \text{N} \tag{15}$$

. Step 3 : The TPA generates the surrogate signature ρ using the r_1, r_2, x_i^0 and x_i^1 as in Eq. (11).

$$\rho = (\text{m}, \text{T}, r_1, r_2, x_i^0, x_i^1, \text{N}) \tag{11}$$

4. Evaluation

The evaluation of the proposed method is divided into security evaluation and performance evaluation.

4.1. Security Evaluation

Table 1 shows the security evaluation of the proposed scheme and the existing scheme [11, 12, 13]. Because the proposed method supports all the items not provided by the existing method, it can efficiently process the muscle control on the aspect side. Data confidentiality can be expressed as (3) and (3) when random numbers x_1^0 and x_1^1 are generated from private keys generated from $_R\{0,1\}^n$ In the same way, it provides data confidentiality because it generates the public key PK_{i+1}^{δ} = h(x_{i+1}^{δ}) corresponding to the random numbers x_1^0 and x_1^1.

Data integrity uses the attribute key pair of the data to aid in the integrity of the data in the process of generating the metadata attribute key pair. The data attribute key pair used for data integrity provides data integrity because it is used for access control purposes that can handle user access control and data encryption / decryption keys.

Table 1: Comparisons of Security Evaluation

Property	[11]	[12]	[13]	Proposed Scheme
Confidentiality	×	×	Δ	√
Dynamic Operations	×	√	√	√
Integrity	×	×	×	√
Batch Operation	×	×	√	√
Free Riding Resistance	√	×	√	√
Privacy preservation	×	×	×	√

√ : provide the corresponding property
Δ : partially provide the corresponding property
× : not process the corresponding property

Privacy preserving is one of the security items provided by default in the cloud environment. The proposed scheme provides the user's ID information through the TPA and the signature process of the sender and the receiver. Free Riding Resistance is guaranteed to be safe for various threats using P2P system in cloud environment. Technique of [23] is not as secure as Free Riding Attack because users do not share files stored on the server with each other. The proposed method provides dynamic operations and batch operations because it uses encryption and decryption for outsourced data to avoid data exposure in the cloud environment and a challenge and response mechanism is used to reduce data collision.

4.2. Integrity-Related Performance Comparison Evaluation

Table 2 shows the results of evaluating the integrity of the data in the different cloud environments with the proposed method and the existing method [11,12,13].Table 2shows the results of the comparative evaluation of six items such as complexity and verifier storage complexity. As shown in Table 2, the proposed scheme supports all of the items not provided by the existing scheme, thus showing high performance efficiency. Especially, in the proposed method, since the user's important data information I(=□^{n}(s)) is selected through $_R\{0,1\}^n$. If you do not know the information in □^{n}(s), it is difficult for the third party to gain access to the cloud through the shared key PK_{i+1}^{δ}.

Table 2: Performance Comparison of Proposed Scheme vs. Previous Scheme

Scheme	Data dynamics	Public auditability	Complexity			
			Server	Verifier	Encryption	Storage
[11]	No	Yes	O(1)	O(1)	O(1)	O(1)
[12]						
[13]	Yes	No				
Our Scheme		Yes	O(logn)	O(logn)	O(logn)	O(logn)

Since the proposed method generates the attribute key using the random key (t_1, t_2) when the server processes the important information of the user as O(logn)in the cloud environment.

When a user's important information is layered in a cloud environment, the proposed method generates a random key for accessing the important information of the user, which transmits

the user data property set information, and transmits the random key to the cloud user together with the previously shared key K_i do. Also, the attribute key for important information of the cloud users at each layer is used for merging processing of important information, so that the verifier storage complexity such as O(logn) appears.

4.3. Overhead

Table 3 shows the results of comparing the proposed method and the existing method with the overhead in terms of User side, Servier side, and TPA. As shown in Table 3, the overhead of the proposed method of handling the user's privacy using TPA and KGC is lower than that of the existing method. This result is the result of user's additional information processing as well as important information analysis. Especially, in the proposed method, the TPA does not separately check and analyze the user's important information, but handles it by linking with KGC and transferring it to TPA.

Table 3: Overhead comparisons of proposed scheme vs. previous schemes

Scheme	User Side	Server Side	TPA
[11]	(2n+2)E.+3C.+n(H.+Mul.)	2P.+(2+c)(E.+Add.+(2c+1)Mul)	(c+1)Add. + 4Mul. + 3P.+(c+1)E.
[12]	(2n+2)E.+5C.+(n+1)H.+nMul.	C(E.+Add.)+c2Mul.	(c+1)Mul.+4P.+(c+2)(H.+E.)
[13]	(2n+1)E.+(n+1)C.+n(H.+Mul.)	1P.+(c+1)(E.+Add.)+(2c+1)Mul.+1H.	(c+1)H.+2Mul.+2P.+(c+3)E.
Our Scheme	(2n+1)E+C+nMul.	(c+1)E+2cH	cn+cE

Add. : Addition C : ConcatenationE : Exponentiation H : Hash Function
Mul. : Multiplication n : The number of OperationsP. : Bilinear Pairing
c : The Number of Corresponding Operations in an Algorithm

5. Conclusion

In recent years, cloud technology has mainly focused on subscribers using computers, but it is expected that services using various types of IoT devices such as wearables and sensors will appear in addition to mobile phones. In this paper, we propose a robust data integrity protection scheme for various security attacks in the cloud environment. The proposed method is to protect the integrity of data transmitted and received in the cloud environment to effectively guarantee the integrity of the data used by users through key generation and processing with low load between TPA, user, and KGC. A key generation process, and a metadata key pair generation process. The keys generated in this process are used to prevent important information of cloud users from being exposed to third parties and anonymous keys are used to prevent third parties from storing important information of users. In addition, the proposed scheme keeps synchronization between the TPA and the user at a predetermined interval so that the important information of the user is not illegally exploited from the third party. In the performance evaluation, it can efficiently process the muscle control on the aspect sidebecause the proposed method supports all the items not provided by the existing method. The proposed scheme supports all of the items not provided by the existing scheme, thus showing high performance efficiency. Especially, in the proposed method, since the user's important data information I(=□^{n}(s)) is selected through $_R\{0,1\}^n$. The overhead of the proposed method of handling the user's privacy using TPA and KGC is lower than that of the existing method.In future research, we plan to apply the proposed method to the actual cloud environment based on the results of this research to evaluate the performance of the proposed method.

References

[1] Kim, D. W., Han, J. W., &Chung, K. I. (2009). Trend of Home Device Authentication/Authorization Technology.*Weekly IT BRIEF*, 1329, pp. 1-11.

[2] Lee,S.Y., Yim, K.B., Bae,K.J., Jeong, T. Y., &Han, J. W. (2009). Counterplan of Ubiquitous Home Network Privacy based on Device Authentication and Authorization.*Korea Institute of Information Security & Cryptology, Review of KIISC*, 18(5), pp.125-131.

[3] Halperin, D., Heydt-Benjamin, T. S., Ransford, B., Clark, S. S., Defend, B., Morgen, W., Fu, K., Kohno, T., & Maisel, W. H. (2009). Pacemakers and implantable cardiac defibrillators: software radio attacks and zero-power defenses.*2008 IEEE Symposium on Security and Privacy (sp 2008)*, pp. 129-142.doi : 10.1109/SP.2008.31.

[4] Liu, C., Chen, J., Yang,L. T., Zhang,X., Yang,C., Ranjan,R., & Kotagiri, R. (2014). Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates.*IEEE Transactions on Parallel & Distributed Systems*, 25(9), 2234–2244.

[5] Mishra,P., Varadharajan, V., Pilli,E., &Tupakula,U.(2018). VMGuard: A VMI-based Security Architecture for Intrusion Detection in Cloud Environment.*IEEE Transactions on Cloud Computing*, pp. 1-1.

[6] Meng, Y., Qin,T., Liu,Y., &He,C. (2018). An Effective High Threating Alarm Mining Method for Cloud SecurityManagement.*IEEE Access*, pp. 1-1, 2018.

[7] Wang,Q., Wang,C., Ren,K., Lou,W., & Li,J. (2011). Enabling public auditability and data dynamics for storage security in cloud computing.*IEEE Transactions on Parallel & Distributed Systems*, 22(5), 847–859.

[8] Wang,C., Chow,S. S. M., Wang, Q., Ren,K., & Lou,W. (2013). Privacy-preserving public auditing for secure cloud storage.*IEEE Transactions on Computers*, 62(2), 362–375.

[9] Li,J., Li,J., Xie,D., & Cai,Z. (2016). Secure auditing and deduplicating data in cloud.*IEEE Transactions on Computers*, 65(8), 2386–2396.

[10] Huang, K., Xian, M., Fu,S., & Liu,J.(2014). Securing the cloud storage audit service: defending against frame and collude attacks of third party auditor.*IET Communications*, 8(12), 2106–2113.

[11] Schwarz, T. S. J., & Miller,E. L.(2006). Store, forget, and check: Using algebraic signatures to check remotely administered storage. in *Proceedings of the International Conference on Distributed Computing Systems*, 12–21.

[12] Luo, Y., Fu,S., Xu,M., & Wang,D. (2014). Enable data dynamics for algebraic signatures based remote data possession checking in the cloud storage.*China Communications*, 11(11), 114–124.

[13] Yang, K., & Jia,X. (2013). An efficient and secure dynamic auditing protocol for data storage in cloud computing.*IEEE Transactions on Parallel & Distributed Systems*, vol. 24, no. 9, pp. 1717–1726, 2013.

[14] Wang,C., Wang,Q., Ren,K., Cao,N., & Lou,W. (2012). Toward secure and dependable storage services in cloud computing.*IEEE Transactions on Services Computing*, 5(2), 220–232.

[15] [15]Mokadem, R., & Litwin,W.(2006). String-matching and update through algebraic signatures in scalable distributed data structures.*Proceedings of the International Workshop on Database and Expert Systems Applications*, 708–711.