International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 3 (2016) pp 1894-1899
© Research India Publications. https://dx.doi.org/10.37622/IJAER/11.3.2016.1894-1899

New Public Integrity Auditing Scheme for Cloud Data Storage Using Mac And Symmetric Key Cryptographic Algorithms

Ms. A. Emily Jenifer
PG Scholar, Department of Information Technology,
Periyar Maniammai University, Tanjore District, Tamil Nadu, India.

Ms. S. Karthigaiveni
Assistant Professor, Department of Information Technology,
Periyar Maniammai University, Tanjore District, Tamil Nadu, India.

Abstract

In recent days, the cloud data storage becomes a rising trend that promotes the secure remote data auditing. The existing works consider the problem secure and efficient public integrity auditing for shared dynamic data storage. But, it is not secure against the collusion of cloud data and it revokes the group of users during the user revocation in cloud system. Thus, this paper proposed a new public integrity auditing mechanism with the help of Message Authentication Code (MAC) generation and symmetric cryptographic techniques. The main intention of this paper is to design a secure and an efficient cloud data storage system to reduce the bandwidth and to improve the data integrity. The proposed system supports an effective user revocation and public checking processes. Moreover, it supports some properties such as, confidentiality, countability, efficiency and traceability for secure data storage. The experimental results are analyzed and evaluated in terms of computation time, block size, key size, number of rounds and cycles per block.

Keywords: Cloud Computing, Message Authentication Code (MAC), Message Digest (MD), Symmetric Key Cryptography, Public Integrity Auditing, Security and Data Storage.

Introduction

Cloud computing is a new paradigm that is mainly used to provide an economic resource utilization over the internet. It is a method of computing in which the users can increase or decrease their computing resources. Generally, the cloud contains three models as shown Fig 1. It includes,

- Public
- Private
- Hybrid

Public – This type of cloud is known as an external cloud, where the services are offered by a third party through the internet. The offered services are used for the commercial purposes.
Private – This type of cloud contains a hosting private applications and services, which are mainly used for the private purposes.
Hybrid – This type of cloud is a combination of both private and public clouds. It is more suitable for those who want to invest minimally in infrastructure and data storage with more security. The cloud motivates the organizations and enterprises to outsource their data to a third party service providers. Moreover, the cloud computing delivers the hosted services via the internet based on the user's demand and these services include,

- Software services
- Software applications
- Network resources
- Computing infrastructures
- Platforms
- Virtual servers

The three different services of cloud computing are described as follows:

Software as a Service (SaaS) – In this mode, the software application is hosted as a service and the end users use the application with the help of web browser.
Platform as a Service (PaaS) ¬– In this model, the end user creates, uploads and test the application with the help of library functions and software tools hosted by the service provider.
Infrastructure as a Service (IaaS) – In this model, the hosting of hardware computing services like storage, hard-drive, servers and network components. Here, the service provider is responsible for maintenance and managing all these resources.

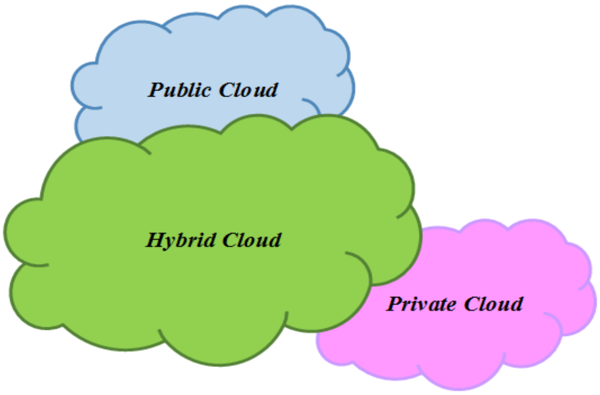

Figure 1: Types of cloud

Security is an important issue in cloud, where the service provider provides full security for both customer and user. The main intention of this paper is to analyze the problem of public integrity auditing for shared dynamic data with group user revocation. The major contributions are as follows:

- The secure and efficient shared data integrate auditing process is explored for ciphertext database with the help of multi-user operation.
- Here, an efficient data auditing scheme is proposed for providing data integrity by incorporating the primitives of victor commitment, asymmetric group key agreement and group signature.

The remaining parts of this paper is organized as follows: Section II reviews some of the existing works related to public integrity auditing in cloud. Section III presents the detailed description for the proposed cloud integrity auditing system. Section IV presents the experimental analysis and results of the proposed auditing system. Finally, this paper is concluded and the future work to be carried out is stated in Section V.

Related Work

This section deals with the works related to cloud architecture and the encryption techniques involved in it. Network advancements [1] increased the utilization of computer resources in storage and outsourcing mechanisms. Hence, built of secure cloud storage was the major concern in public cloud infrastructure. *Kamara and Lauter*[2]discussed the secure cloud storage construction. They described the several architectures that contain hybrid non-standard cryptographic primitives to achieve the cloud storage. Besides, they surveyed the benefits of hybrid architecture. Dynamic selection of inputs to the multiple workers in cloud architecture required the function. *Gennaro et al.* [3]introduced and formalized the term called *"verifiable computation"* that enabled the outsourcing under dynamic input selection. The primary constraint for the proof verifications is less computational effort. Research studies focused on provision of data dynamics and public verifiability. *Hao et al.* [4] discussed the public verifiability support with the remote data integrity checking protocol. But, the absence of clear mapping between data and tags results in less support to data dynamics level. The removal of unnecessary copies of repeating data improved the dynamics level termed as client side de-duplication.. *Halevi et al.* [5] identified the attacks that exploited the client side de-duplication with the Proof-of Ownership (PoW) formulization. Merkle tree and the specific encodings were included in PoW and offered the security analysis. The periodical cryptographic checking was required to assure the correctness of outsourced operations over the dynamic sets collection. *Papamanthou et al.* [6]presented the authenticated data structure to support public verification. Bilinear map accumulators and accumulation tree were included in this structure for optimal verification achievement.

The current proof verification methodologies were not sufficient for large and complex data size. The big data challenge is an attractive research area to mine the interesting patterns. *Jiang et al.* [7] overviewed the current status challenges in big data mining and covered the tools required to carry the mining process. The storage of huge size personnel and corporate records have the poor connectivity that leads to theft. *Anderson and Zhang*[8] described the de-duplication mechanism with the common data between the users. The client-end –per user encryption and the unique features were supported by de-duplication algorithm with the maximum speed. *Bellare et al.* [9]extended the existing de-duplication to secure with the new primitive called Message Locked Encryption (MLE). They provided the formal definitions of both privacy and tag consistency. Reliable transmission of data files to the distributed systems was governed by an algorithm called *Information Dispersal Algorithms (IDA).* The two confidentiality levels in IDA are weak and strong. *Li* [10]explored the feasible condition for on adapted code. They investigated the performance of Rabin's IDA and Reed-Solomon code on the computation complexity. The realization of high non-trivial protocols and primitives was required in cryptographic strategies. For that, research studies addressed the important primitive called Vector Commitments (VC). *Catalano and Fiore*[11] discussed the VC for the satisfaction of position bindings with the constraint that the string size independent of vector length. They showed the VC realizations in RSA and Computational Diffie-Hellman to show the usefulness.

The asymptotic complexity in computations were more in key generation methodologies. *Gentry and Halevi*[12] discussed the implementation of Fully Homomorphic Encryption (FHE). The complete working of any encryption methodologies required the functionality called bootstrapping. They provided the optimization in key generation that reduced the complexity with *n*-dimensional lattices. In a data storage outsourcing services, the permission assigned to the data owner to check whether they stored data correctly or not. *Yuan and Yu*[13] proposed the Proof-of-retrievability (POR) with the public verification capability and the constant communication cost. The involvement of constant group members in the exchanging of data between the provider and verifier. The improvement of confidence level during outsourcing operation, client verification of correctness required. *Parno and Gentry*[14]introduced the Pinocchio, an architecture for verification of general computations on cryptographic assumptions with the minimum execution time. The storage of large dataset in an untrusted server leads to security problem. *Benabbas et al.* [15] presented high degree polynomial-based verifiable computation scheme and made the predictions for large datasets. They made the retrieval and update process with the Verifiable Database (VD) that minimized the resource consumption. The time-dependent nature of an input size leads to security problem and more execution time consumption. *Backeset al.* [16] proposed a novel cryptographic techniques solved the quadratic polynomials with the coverage of wide arithmetic computations. The quadratic polynomials utilization reduces the execution time with the improved efficiency.

Data and service management were not trustworthy for the large dataset in cloud architecture. The unique attribute increased the security challenges. *Wang et al.* [17] focused on cloud data storage security. They proposed flexible distributed scheme to assure the correctness of user's data in the cloud model. They integrated the correctness with the error localization with the homomorphic token. The arrival of new vulnerabilities and additional burden affects the secure storage. *Shrinivas*[18]proposed a secure cloud storage system with the privacy preserving public auditing. They utilized the homomorphic authenticator and random masking guarantee the Third Party Auditor (TPA) to unaware of data content in storage during an auditing process. But, the preservation of identity policy was the main challenge in public auditing mechanism. Wang et al. exploited the ring signatures computation to keep the identity of the block as a private one. The public verification of data integrity was achieved without retrieving the entire file.

Bhaskar and Umadevi[19] extended the privacy preserving public audit scheme to large number of users involved. They utilized the group signatures for the construction of homomorphic authenticators. The information amount and the computational time were not affected by the users. The computational cost and the collusion of misbehaving and revoked servers were high. *Yuan and Yu*[20] proposed the novel data integrity checking with the multi-user modification, collusion resistance and the constant computational cost. From the study it is observed that the encryption and decryption stage played a major role in proof and integrity verification. But, the computational rounds increased the time consumption. Hence an optimized reduction method proposed with the cipher-block symmetry to reduce the computational time and offer the secure cloud data storage.

Proposed System

This section presents the detailed description of the proposed cloud storage model. In this paper, the problem of designing public integrity auditing based shared dynamic data with group user revocation is analyzed. The main contributions of this paper are listed as follows:

- An efficient and secure public integrity auditing scheme is proposed for cipher-text base with the help of multi-user operation.
- The primitives of victor commitment, group signature and asymmetric group key agreement are integrated.
- The new features like, traceability and countability are provided at the time of auditing.

Fig 2 shows the overall flow of the proposed public integrity auditing scheme. At first, the hash key is generated for the user input and the block of the data is encrypted. Consequently, the hash key is also encrypted and the file is uploaded in the cloud. If the data is already exists, it is identified that the upload file is duplicated. Otherwise, it will divided into various blocks, after that the availability of the blocks is also verified. If the block is not exists, it will be given to the Cloud Service Provider (CSP). Hence, the file is downloaded from the server and to access that file, the hash key and data blocks are need to be decrypted. In this work, the two different algorithms such as, Message Digest Code i. e. Message Authentication Code (MAC) algorithm is used for code generation. Then, the symmetric key cryptography algorithm is used for encryption and decryption processes.

Algorithm I – MAC Algorithm

```
Input: Block of file B = (B1, B2, B3 … BN}
Step 1: Generate the hash key;
Step 2: Encrypt the data block with the help of hash key;
Step 3: Generate the tag and hash key from the content of the file;
Output: Obtain the decrypted file with the help of hash key;
    1. H(M) = K (Key Generation)
    Where,
        M= Input file
        H(M)=Hash value of file
        K= Hash Key
    2. E(K, M)=C (Encryption)
    Where.
        M-File,
        K-Hash Key,
        E-Encryption
        C-Cipher Text
    3. Tag(M)=T (Tag Generation)
    Where,
        M-File
        T-Tag from File M
```

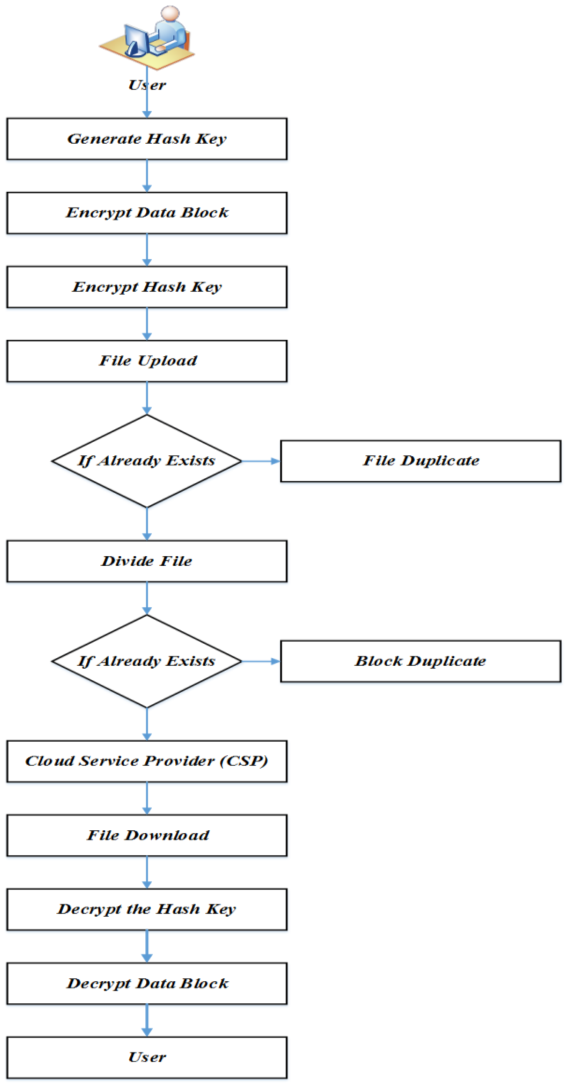

Figure 2: Overall flow of the proposed system

Algorithm II – Symmetric Block Cipher
Input: *Block of File B={B1, B2, ...BN}p 1 :*
Step 1: *Encrypt the block of file based on the secret key;*
Step 2: *Convert the plain text into the cipher text that is in the encrypted form;*
Output: *Identical data copies of different users will lead to produce same cipher text.*
C=SBC. Encrypt (File, χ)
Where,
C=Cipher text,
SBC=Encryption
File=message (data)
Steps:
1. Get a 64-bit key from the user.
2. The keys are actually stored as being 64 bits long.
3. Encrypt each 64-bit block of data.

Cloud Storage Model

Fig 3 shows the cloud storage model, where the three different entities are used such as, cloud storage server, group user and a Third Party Auditor (TPA). The group users have a data owner and a number of authorized users, who have an access to change the data. The cloud storage server is semi-trusted that provides the data storage services to group users. The TPA can be any entity in the cloud that conducts the data integrity in the cloud server. In the proposed system, the data owner can encrypt and upload its data in the remote cloud storage server. Moreover, the cloud service provider manages an enterprise class infrastructure to deliver a secure, reliable and scalable environment to user. Here, the TPA can efficiently verify the data integrity in the storage server. When a group of user is found as malicious or expired, the data owner can securely revoke a group of user. Hence, it is difference from other group of users.

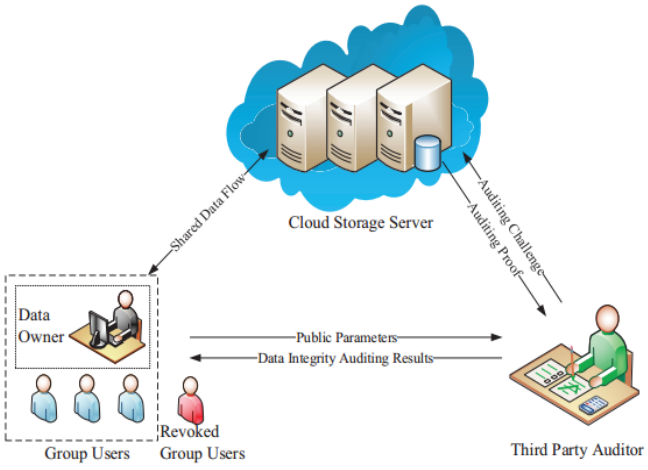

Figure 3: Cloud data storage model

Security Goals

The security goals in cloud are as follows:

Security – If the scheme is secure, the adversary cannot convince a verifier to accept an invalid output.

Correctness – If an honest cloud storage server has the cipher text value for any database and any updated data, scheme is correct.

Efficiency – If the computation and a storage overhead are invested by any client user for any data, a scheme is efficient. Hence, the client must be independent of the shared data.

Countability – If the TPA provides a proof of misbehavior for any data, a scheme is countable.
Traceability – When the data is generated by the generation algorithm and the generated signature, the data owner is required to trace the last user, who update the data.

Performance Analysis

This section presents the performance analysis of proposed method and compared with the existing Advanced Encryption Standard (AES) scheme on the parameters of computational time, block size, key size, number of cycles/block, rounds. The optimization in the key generation provided by symmetric block cipher reduces parameters effectively.

i. Computational time

The time required for the data to be encrypted and decrypted termed as computational time. An algorithm effectiveness decided by the low computational time. Fig 4 shows the variation of computational time for both the proposed and existing AES method. The computation time for AES is 30 ms and for proposed symmetric cipher block is 7. 5 ms. The optimal steps reduction in proposed work offers 75 % reduction in computational time.

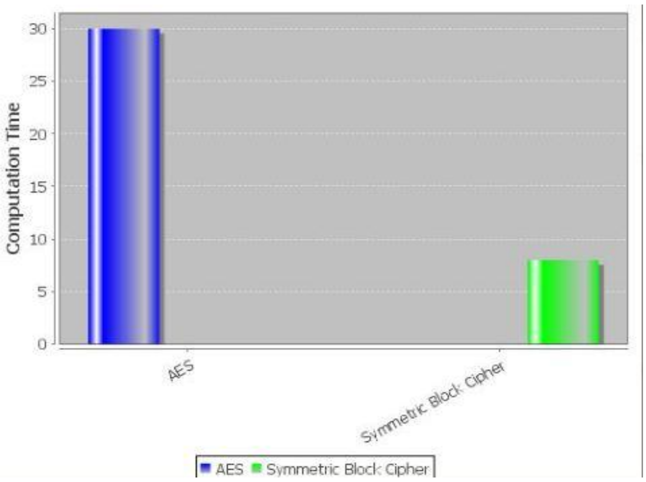

Figure 4: Computational time

ii. Block size

The data and associated key primitives consume the space in cloud architecture refers block size. An algorithm effectiveness decided by the low space consumption. Fig 5 shows the variation of block size for both the proposed and existing AES method. The block size for AES is 128 bits and for proposed symmetric cipher block is 65 bits. The optimal steps reduction in proposed work offers 49. 21 % reduction in block size.

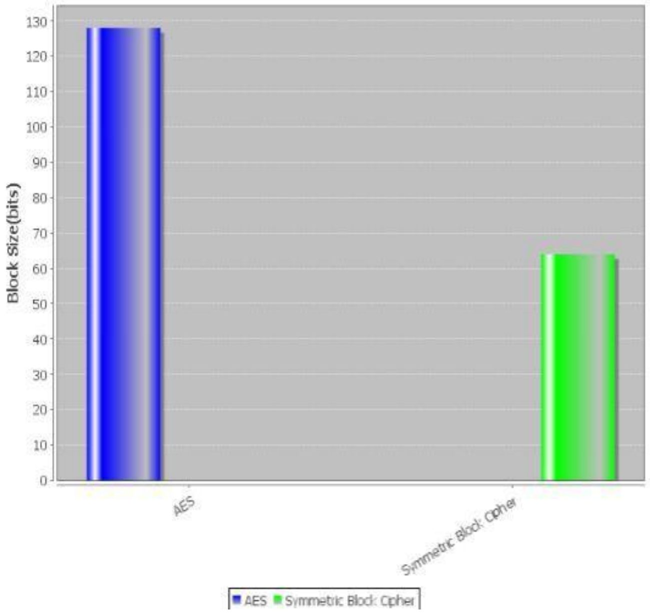

Figure 5: Block size analysis

iii. Key size

The number of bits consume the space in key generation refers key size. An algorithm effectiveness decided by the low key size. Fig 6 shows the variation of key size for both the proposed and existing AES method. The key size for AES is 128 bits and for proposed symmetric cipher block is 58 bits. The optimal steps reduction in proposed work offers 54. 69 % reduction in key size.

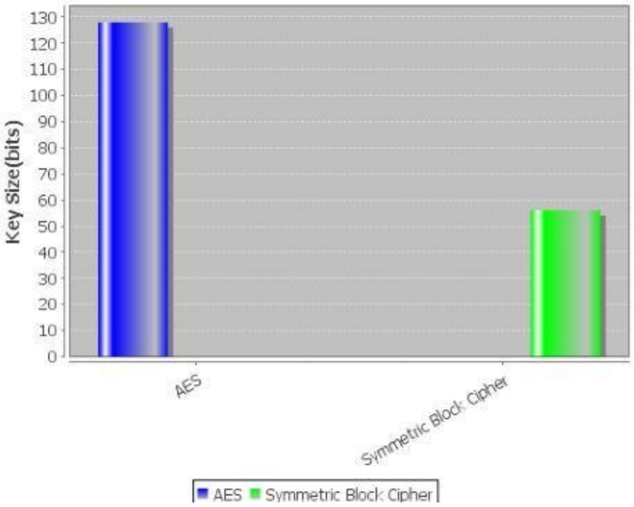

Figure 6: Key size analysis

iv. *Number of rounds*

The number of bits consume the space in key generation refers key size. An algorithm effectiveness decided by the low key size. Fig 7 shows the variation of rounds for both the proposed and existing AES method. The number of rounds for AES is 14 and for proposed symmetric cipher block is 12. The optimal steps reduction in proposed work offers 14. 56 % reduction in rounds.

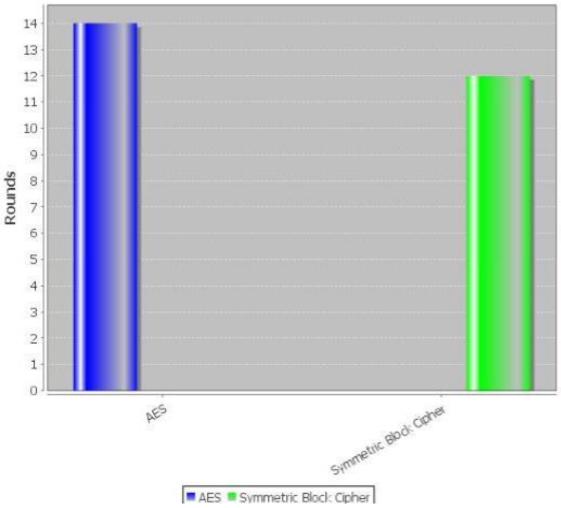

Figure 7: Rounds analysis

v. Cycles/Block

The number of cycles per block in key generation should be minimum for less computational time. An algorithm effectiveness decided by the low cycles. Fig 8 shows the variation of cycles for both the proposed and existing AES method. The cycles for AES is 128 ms and for proposed symmetric cipher block is 55 ms. The optimal steps reduction in proposed work offers 52. 69 % reduction in cycles.

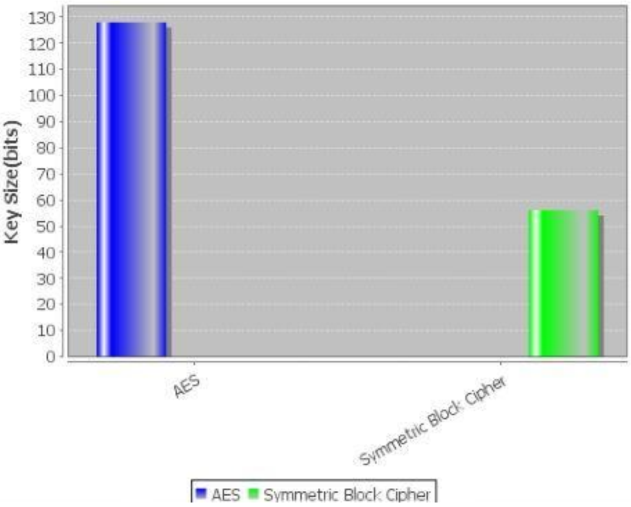

Figure 8: Cycles/block analysis.

Conclusion

This paper proposed a new public integrity auditing scheme for cloud data storage. For this purpose, two different algorithms such as, Message Authentication Code (MAC) generation and symmetric key cryptographic techniques are used in this work. The main objective of this paper is to reduce the bandwidth and to improve the content integrity in cloud data storage. Here, the third party auditor verifies the data availability for reducing the bandwidth. Before upload the data in cloud, the Message Digest (MD) value stored in the database is verified. Then, the MAC is generated and the auditor compares the MD value of the uploaded file with the database. If the values are match, it is identified that the data is duplicate and it will not be processed. Otherwise, the data will be uploaded in the cloud by performing the encryption process. Hence, the uploaded data will be retrieved by decrypting both the hash key and data block. The performance of the proposed public integrity auditing scheme is evaluated in terms of computation time, block size, key size, number of rounds and cycles per block. From this analysis, it is observed that the proposed system provides the better results

References

[1] A. E. C. Cloud, "Simple Storage Service, " ed.

[2] S. Kamara and K. Lauter, "Cryptographic cloud storage, " in *Financial Cryptography and Data Security*, ed: Springer, 2010, pp. 136-149.

[3] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers, " in *Advances in Cryptology–CRYPTO 2010*, ed: Springer, 2010, pp. 465-482.

[4] Z. Hao, S. Zhong, and N. Yu, "A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability, " *IEEE transactions on Knowledge and Data Engineering*, vol. 23, pp. 1432-1437, 2011.

[5] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems, " in *Proceedings of the 18th ACM conference on Computer and communications security*, 2011, pp. 491-500.

[6] C. Papamanthou, R. Tamassia, and N. Triandopoulos, "Optimal verification of operations on dynamic sets, " in *Advances in Cryptology–CRYPTO 2011*, ed: Springer, 2011, pp. 91-110.

[7] T. Jiang, X. Chen, and J. Ma, "Public Integrity Auditing for Shared Dynamic Cloud Data with Group User Revocation. "

[8] P. Anderson and L. Zhang, "Fast and Secure Laptop Backups with Encrypted De-duplication, " in *LISA*, 2010.

[9] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure deduplication, " in *Advances in Cryptology–EUROCRYPT 2013*, ed: Springer, 2013, pp. 296-312.

[10] M. Li, "On the confidentiality of information dispersal algorithms and their erasure codes, " *arXiv preprint arXiv:1206. 4123*, 2012.

[11] D. Catalano and D. Fiore, "Vector commitments and their applications, " in *Public-Key Cryptography–PKC 2013*, ed: Springer, 2013, pp. 55-72.

[12] C. Gentry and S. Halevi, "Implementing Gentry’s fully-homomorphic encryption scheme, " in *Advances in Cryptology–EUROCRYPT 2011*, ed: Springer, 2011, pp. 129-148.

[13] J. Yuan and S. Yu, "Proofs of retrievability with public verifiability and constant communication cost in cloud, " in *Proceedings of the 2013 international workshop on Security in cloud computing*, 2013, pp. 19-26.

[14] B. Parno, J. Howell, C. Gentry, and M. Raykova, "Pinocchio: Nearly practical verifiable computation, " in *EEE Symposium on Security and Privacy (SP), 2013 I*, 2013, pp. 238-252.

[15] S. Benabbas, R. Gennaro, and Y. Vahlis, "Verifiable delegation of computation over large datasets, " in *Advances in Cryptology–CRYPTO 2011*, ed: Springer, 2011, pp. 111-131.

[16] M. Backes, D. Fiore, and R. M. Reischuk, "Verifiable delegation of computation on outsourced data, " in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013, pp. 863-874.

[17] B. Wang, B. Li, and H. Li, "Oruta: Privacy-preserving public auditing for shared data in the cloud, " in *IEEE 5th International Conference on Cloud Computing (CLOUD), 2012* 2012, pp. 295-302.

[18] D. Shrinivas, "Privacy-preserving public auditing in cloud storage security, " *International Journal of computer science nad Information Technologies*, vol. 2, pp. 2691-2693, 2011.

[19] M. Bhaskar and G. Umadevi, "Public Auditing For Shared Data With Efficient User Revocation In The Cloud. "

[20] J. Yuan and S. Yu, "Efficient public integrity checking for cloud data sharing with multi-user modification, " in *INFOCOM, 2014 Proceedings IEEE*, 2014, pp. 2121-2129.