Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

Enabling Efficient Cloud Storage and Privacy Preservation with Cloud Auditing

Hariharan R[1], Komarasamy G[2], Daniel Madan Raja S[3]
[1]*Department of Information Technology, Sri Ramakrishna Institute of Technology, Coimbatore*
[2]*Department of Computer Science and Engineering, GITAM School of Technology,*
GITAM Deemed To Be University, Bengaluru, India.
[3]*Department of Information Technology, Bannari Amman Institute of Technology, Sathyamangalam*
hariharanrcse@gmail.com[1], gkomarasamy@gmail.com[2], cross4u@gmail.com[3]

Abstract **-** With the substantial growth in cloud storage, the users for data outsourcing towards cloud servers are drastically increased. Subsequently, increasing data volume over the cloud has led to enormous data duplication. However, the cloud server maintains the unique copy of original data over the storage system; but leads to an immeasurable loss when the data is missing or corrupted. However, duplicate file maintenance and auditing integrity are extremely essential in the industrial and academic field. It helps to maintain the file over the cloud in a secure and protected manner. In this research work, a secure file maintenance scheme over cloud storage with cloud auditing is proposed to maintain the duplication copy of the file by encrypting the files. It helps in the integrity of public auditing of every copy with reduced storage. Specifically, the cloud auditing establishes secure authentication with tag-based duplication. Next, this proposed model uses appropriate encryption techniques to fulfil the confidentiality of data during the auditing integrity and duplication process. This work pretends to support every data owner to maintain data integrity and periodically delegate third party auditing (TPA) independently to deal with the multiple auditing strategy for outsourced data. Thus, the proposed model achieves better security and confirms the performance by performing simulation over MATLAB environment. The experimented results give better trade-off in contrast to prevailing approaches.

Index Terms - Cloud storage, duplication, cloud auditing, data integrity and file encryption

I. INTRODUCTION

Cloud storage is an un-avoidable division in cloud computing that facilitates data owners (DO) to preserve data over the servers and offers low cost, scalability, and resourceful storage [1]. Owing to the storage benefits on management and costs, an enormous number of organizations and individuals maintain various data to cloud service providers for the past few years [2]. Moreover, the promising idea towards data storage faces enormous challenges on efficiency and security [3]. The preliminary aspect is cloud storage efficiency. The cloud data volume is drastically increasing and various users store enormous data in the cloud storage which sometimes causes duplication over the data in cloud [4]. Based on various surveys, it is observed that 75% data is duplicated [5]. Therefore, to maintain the storage and enhance efficiency, storage requires effectual reduction of duplication, i.e. cloud server maintains a copy of duplicate files while the service providers have to offer the link for accessing those unique files for all the data from the data owners who own the file [6]. However, the de-duplication process is divided into two factors: client-side and server-side de-duplication [7].

The latter is defined as the process to validate the duplication over the cloud and carry out the corresponding operation while receiving outsourced data from the clients [8]. This technique preserves the storage space on the server-side. Similarly, the former model specifies that DO interact with every other server to validate data stored in server before providing file to the server [9]. When the outsourced file is considered to be duplicated, client does not want to upload any outsourced file stored in cloud [10]. However, client-side de-duplication maintains storage, network bandwidth, and communication costs which provides advantage to both clients and servers [11].

The successive aspect is security over cloud storage. During the de-duplication process over the cloud storage, the service provider needs to have a unique copy of data owners file. The software failure and storage hardware fails when the unique data copy is being damaged, which causes drastic loss over for service providers and data owners [12]. Henceforth, it is necessary to fulfilling data integrity of data outsourced for de-duplication of cloud storage systems. The outsourced data from the data owners are safely stored and intact in the cloud servers is the preliminary focus of service providers and data owners. Therefore, the de-duplication cloud storage system needs to assist the data integrity.

Till now, some prevailing approaches with client-side de-duplication rely on short hash value, static (outsourced file hash value) as the evident for data owners. Moreover, this approach is extremely vulnerable towards attacks: the hash value is extremely attained by the malicious user which proves to be CSP holds the file with shorter hash values [13]. Therefore, it attains the complete file from the CSP. The shortest hash value leakage leads to the complete leakage of the file to some adversaries. Henceforth, cryptographic primitive is termed as 'proof of ownership' is anticipated to resolve certain vulnerability, security which is securely and effectually validated by the

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

data owners with interact file before the service provider creates access towards the link for the users. The cloud data integrity approaches include two different categories: Proof of Retrievability (PoR) and Proof of data possession (PoDP). The data possession facilitates the users to validate the data integrity for the user's file over the cloud environment [14]. When compared to data procession, retrievability offers added benefits to data owners for recovery purpose also. The researches carried out in recent years deals with these two types of schemes in different aspects.

The data integrity based on public audit and file de-duplication have to be considered to enhance the efficiency and security of cloud storage. The most-effectual approach is the integration of both proofs of data possession and proof of retrievability along with proof of work schemes. Moreover, this approaches gives overhead with an order of W^n for all files to CSP, where $'W'$is several owners holding files and $'n'$ is total blocks over the file. Moreover, every data owners generate the authentication tags separately and upload tags to service providers for data auditing [15]. Therefore, service providers have to maintain the data from various owners by tag authentication of those files which is considered to be another form of redundancy and duplication. Hence, it is essential to fulfilling both authentication and de-duplication process to maintain more storage space. It supports both system storage for public cloud auditing and integrity. Additionally, it is motivated by the fact that the data duplication and confidentiality are maintained by the data encryption process which is extremely essential for practical requirement fulfilment.

The work is organized as: section 2 is a detailed explanation for prevailing approaches in terms of data auditing, duplication, and confidentiality. Section 3 discusses the anticipated methodology; section 4 is numerical results and discussions; section 5 is a conclusion with future research directions.

II. RELATED WORKS

Recently, with the fast-growing interest towards the data storage in the cloud, data integrity, data auditing, file de-duplication, and integration of both. This section discusses the existing approaches used for data auditing, integrity, and data de-duplication.

Storer et al. [16], explains about dropbox for de-duplication approach which performs has value computation of file which is considered as the proof of data owners. Regrettably, service providers offer access links of files for malicious users with appropriate hash values which leads to hash value leakage and affects the entire file to the adversaries. Subsequently, these de-duplication approaches have a diverse security vulnerability. Halevi et al. [17], anticipated a protocol termed as 'proof of ownership' where the users can effectually validate service providers which exactly deals with the intact file. Pietro et al. [18] anticipate an enhanced proof of word scheme with constant computational complexity.

However, the above-mentioned proof of word approaches does not determine the file duplication. Next, the data owner needs to initiate encryption of outsourced files to preserve the data confidentiality. Moreover, various encryption algorithms and keys maintain various owners to offer ciphertexts for the user's files which leads the service providers to maintain all the diverse cypher texts. However, de-duplication of encryption is an extremely complex factor. Ng et al., [19] initiate privacy preservation towards the file de-duplication strategy. Douceur et al., [20] initiate cryptographic terms termed as 'encryption convergent' to resolve various issues like data de-duplication, confidentiality. Keelveedhi et al., [21] explored duplication strategy for convergent encryption in various securities with de-duplication.

Wang et al., [23] offered two diverse dynamic PDP strategies using various data structures like Merkle hash tree or skip lists. Then, this process is merged with random masking with various other privacy-preserving policies. It fulfills the third party auditing without any detailed outsourced files. However, these data suffer from enormous computation and communication overheads. Author in [24] modelled significant public auditing strategy for privacy preservation with batch updation. Juels et al., [25] explained proof of retrievability strategy which does not validate the service provisioning with intact file; however, it does not guarantee data retrievability using code. Author in [26] modelled an enhanced retrievability strategy using public cloud service provisioning based on homomorphic authentication for auditing proof and aggregate the constant value. Yuan et al., [27] explained enhanced proof retrievability for complete security proof and attains public auditing with constant complex communication.

Zheng et al., [28] explain a storage proof with de-duplication process to attain superior data auditing and de-duplication concurrently. Similarly, the author explains about the proof retrievability and data possession to attain proof of word. Moreover, the data owners have constrained bandwidth and restriction of computation and cannot attain enormous computational and communication overheads with a linear model using several blocks and segments over each block. Yuan et al., [29] use homomorphic authenticators and polynomial-based authentication

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

tags to model a new public cloud auditing with de-duplication process which diminishes the storage overheads with authentication tags by integrating tag files from various data owners and therefore it attains better authentication tags de-duplication.

Moreover, PCAD strategy cannot assist the de-duplication of encrypted data. Author in [30] offered secure de-duplication and auditing of data strategy which is differentiated by the PCAD strategy as it considered outsourcing tag generation computation to MapCloud auditor and attains auditing and de-duplication. Moreover, it is modelled using a Merkle hash table and therefore it requires heavy computation and communication cost during word proof and integrity verification process. Li et al., [30] explain the privacy preservation of public cloud auditing strategy for assisting data de-duplication using bloom filter. It is utilized to validate the performance when the user exactly maintains the claim of file. This strategy is used to un-cheatable and un-forgeable in de-duplication and public auditing process. It is proven that the service provides have to maintain all the authentication tags of files from various users which is considerably proven to have high storage overheads in CSP.

III. METHODOLOGY

This section discusses the relevant idea towards cloud model and security definition with encrypted data. The proposed framework comprises of various kinds of entities like a third-party auditor, private key generator, cloud service provider, and the client respectively. Fig 1 depicts the system model. It is composed Third Party Auditing, Cloud Service Provider and client respectively. The data transmission and reception are depicted in Fig 1.

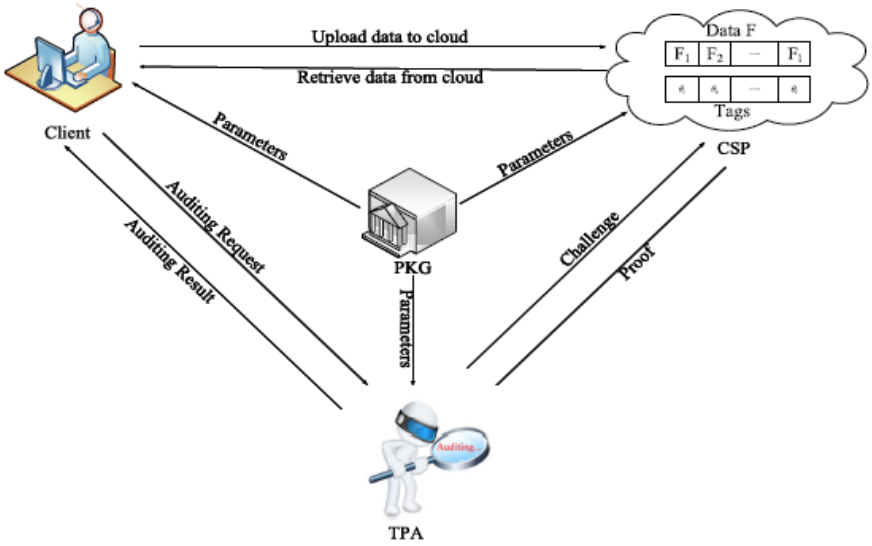

Fig 1: System model

(1) Third-party auditor: It is an independent third party. The users perform data integrity verification based on the request devoid of any data content learning.

(2) Private Key generator: It is completely trusted entity for storage and performs various system parameter generations like private-public key for other entities.

(3) Cloud Service provider (CSP): It is an entity with unconstrained computational competency and storage capability and has the accountable for maintaining and storing the outsourced data of the users. It is a semi-trusted party.
(4) Client: Data owner possesses enormous data needs to be moved for a remote server for sharing, maintenance, and storage. It may be individual organization or users.

The proposed model includes three diverse phases like setup, storage, and auditing phase. The setup phase is used for establishing the system generation and users' key generation. Then, the storage phase includes storage of the de-duplication process where it includes two different cases. 1) The original data users store the file with

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

cipher-text based original file; next DO interacts with service providers for file de-duplication storage. 2) DO interact with service provider for de-duplication of file storage. Finally, the auditing phase includes the auditing integrity over unique file copies to be stored in cloud. It facilitates data owners to audit the data integrity independently with their files.

Setup phase: It is comprised of two steps.

(1) Initialization: It gives parameter like hashing, encryption, decryption, extractors. These are made public with a larger prime number to be determined for security parameters. This process carries out a secure encryption process with a pair of symmetric encryption and decryption like AES, extractors for key extraction that is determined using file content.

(2) Key generation: Here, random elements are generated which is provided publicly. Then, the seed values are randomly picked and secretly transmit files to the data owners. The user selects random value that generates a set of public and secret keys for signatures and data owners make secret and public keys.

Storage phase: To acquire both secure integrity auditing and client-side de-duplication and data integrity and auditing. This storage phase is different from conventional auditing schemes. More specifically, it has two diverse file uploading process for unique owners and other owners. The encryption scheme is depicted with four diverse algorithms:

KeyGen→ it considers security parameter and file as inputs and outputs of a given file.

Encryption →It considers file as input and outputs and convergent key. It encrypts the ciphertext of given files.

Decryption →The decryption process considers convergent key and cipher text as input and outputs of file over plaintext.

TagGen → It is generated based on the algorithm that considers a file as input/outputs as tags of a given file.

Ownership proof

It is an interactive protocol that functions among the user (data owner) and service provider (verification) where the user convinces the service provider to claim for the file. It is stored in the cloud storage space. It is essential for client de-duplication. The proof of work formulation is explained below:

$\boldsymbol{Sum}\left(\boldsymbol{F}, \boldsymbol{1^{\lambda}}\right) \rightarrow \boldsymbol{sum_F}$. It performs random summation function and security parameters as input and output for summary with a bit-length of sum_F where the file size is independent $|F|$.

$\boldsymbol{ChalGen}\ (\boldsymbol{n}, \boldsymbol{d}) \rightarrow \boldsymbol{chal}$. The challenge process considers the block of file $'F'$ and input sample size is $'d'$ and the output sample is given as $chal \subset [1, n]$ with $|chal| = d$.

$\boldsymbol{ProofGen}\ (\boldsymbol{F}, \boldsymbol{chal}) \rightarrow \boldsymbol{P}$. The proof generation process is run to produce ownership PoF. It considers the file blocks and challenge set as inputs /outputs as proof of ownership.

$\boldsymbol{ProofCheck}\ (\boldsymbol{P}, \boldsymbol{sum_F}, \boldsymbol{chal}) \rightarrow \frac{\boldsymbol{Accept}}{\boldsymbol{reject}}$. The proof check algorithm is executed to verify and validate whether the provider holds the File F. It considers the challenge set $chal$, proof $'P'$, and summary value Sum_F as inputs/outputs which is either accepted or rejected.

a. Data auditing

The de-duplication process over the cloud storage system stores the unique copy of files. The software failures and storage hardware lead the file copy to be failed [31]-[32]. Hence, the service provisioning is delegated periodically to batch audit the data integrity of a given source file. Consider, an assumption of third party auditing with audit files $\{F_1, F_2, \dots, F_k\}$ and file F_i has data owners.

(1) Challenge Generation: For the given file $F_i (i = 1,2, \dots, k)$, TPA arbitrarily picks subset element from the set $[1, n]$ and random number set. TPA selects data owner for all every file and evaluates random masking of

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

ownership public key as R_i where $r \in Z_p^*$ is random number. TPA transmits challenge message to the cloud service provider.

(2) Proof Generation: Based on the challenges attained from TPA, the service provider needs to generate tag proof for all the files. It is expressed as in given Equation. (1):

$$\Theta_i = \theta_{i,j}^{w_{i,j}}, C_i = \sum_{j \in I_i} w_{i,j} C_{i,j} \qquad (1)$$

(3) Proof check: The third-party auditor evaluates the aggregated public keys of the owners with the corresponding files. It is expressed as in Equation. (2):

$$Y_i = \prod_{j \in [o,w_i]/\{i_t\}} y_j; \qquad 1,2,3,\dots,k \qquad (2)$$

Thus, the TPA carry-outs batch integrity verification.

b. Secure replication

This process is executed more securely. More specifically, SP's runs de-duplication while receiving files tags from cloud. DO upload file where SP adds a tag to the provided file tags. Else, the service provider holds the duplicate copies and the user utilizes PoW protocol along with the SP. When the file is given to the SP, the user authorizes the stored file to save communication cost and storage costs.

c. Authentic tag replication

The conventional way for data integrity realization is validated by the SP to store all authorized tags for the generation of data owners for all block files, even in case of file duplication. The time complexity is expressed as $O(W_n)$ on the service provider. The proposed model migrates the authentication tag computation to the service provider and then it aggregates the tags of similar blocks. However, CSP has to maintain a set for tag authentication $\{\theta_1, \theta_2, \dots, \theta_n\}$ when there are $'n'$ elements. However, the tag authentication is expressed as in Equation. (3):

$$\theta_i = \theta_i^0 \theta_i^1 \dots . \theta_i^W \qquad (3)$$

Hence, the proposed model can diminish the complete SO's encountered by the service providers during tag authentication from $O(W_n)$ to $O(n)$.

d. Confidentiality establishment

Here, the encryption process is considered for realizing the deterministic encryption and content identified to ensure data confidentiality and also to realize de-duplication for those encrypted data. Additionally, the service provider helps to avoid privacy leakage against TPA during data generation. The sample blocks with TPA masking and public key generation. It is infeasible for TPA to acquire user's private content during hardness assumption.

e. Sampling

The service provider and third party auditing choose data blocks randomly for a challenge from the entire block file during integrity auditing. It is not feasible to challenge complete blocks for validating integrity verification and file correctness as there are huge data outsourced in cloud. However, random sampling is extremely affordable for predicting the service provider misbehaviour. Consider, ρ as the corrupted block proportion, P is probability detection for CSP misbehaviour and $'C'$ is number of challenge blocks. It is expressed as in Equation. (4):

$$P = 1 - (1 - \rho)^c \qquad (4)$$

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

Hence, the number of challenge block is based on corrupted blocks proportion and detection probability. There is 1% lesser than error where the service provider requires to challenge blocks to predict corrupted blocks with 99% probability respectively.

Algorithm 1:
//SignatureGen

1. For every data owner divide file into blocks
2. Evaluate the signature of the given blocks;
3. IP sends data to the service provider with a set of signature
4. end process

//Replication

5. CSP validates data integrity and revokes blocks
6. CSP verifies the file for unique copy or de-duplication copy
7. If the auditing results in 0, then service provider outputs 1.
8. Else the de-duplication process is performed to send the data to CSP for block revocation.
9. CSP de-duplicates the revoked block
10. Service provider checks to de-duplication to avoid data leakage.

//Third Party auditing – Chal

11. TPA generates verification message for all auditing requests
12. It selects the elements from the subset
13. TPA delivers a challenge to CSP

//ProofGen

14. a security challenge for all files
15. CSP response to TPA

//Verify

16. TPA accepts storage proof from the available service provider
17. Response by verification analysis
18. When the output is 1, then TPA considers total blocks to share information more appropriately else TPA is 0.

IV. NUMERICAL RESULTS

The numerical results and discussions of the proposed model is discussed for maintaining the original copy of the system file to avoid data leakage. The simulation is performed in MATLAB where the public and private keys are generated within configured PC. The evaluated is done with Identity-Based Key Encryption (IBKE) and Location-Based Privacy Proof of File security (LB-PPFS). Here, Time (Sec), number of challenge data blocks, computation cost based on storage phase is compared effectively. Table 1 depicts the number of data blocks. The data block ranges from 0-200 where the time is measured in seconds for the de-duplication process. The time needed for encrypting the proposed model is 0.1, 0.35, 0.47, 0.56, 0.86, 0.92, 1.11, 1.26, and 1.38 respectively. Fig 2 shows the pictorial representation of the number of data blocks Vs Time (s).

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

Table 1: No. of data blocks Vs Time (s)

No. of data blocks	IBKE	LB-PPFS	Proposed model
0	0.2	0.3	0.1
25	0.4	0.54	0.35
50	0.7	0.89	0.47
75	0.9	0.95	0.56
100	1	1.24	0.86
125	1.25	1.35	0.92
150	1.38	1.46	1.11
175	1.56	1.54	1.26
200	1.64	1.86	1.38

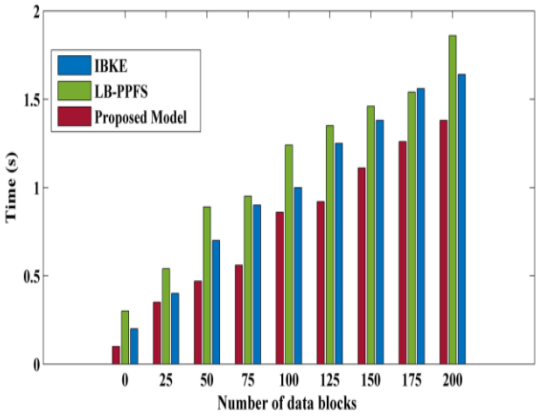

Fig 2: Graphical representation of the number of blocks Vs Time (s)

Table 2: No. of challenge data blocks Vs Time (s)

Number of data blocks	IBKE	LB-PPFS	Proposed model
0	0.01	0.25	0.35
20	0.15	0.32	0.52
40	0.20	0.46	0.63
60	0.25	0.52	0.78
80	0.30	0.63	0.86
100	0.35	0.72	0.95

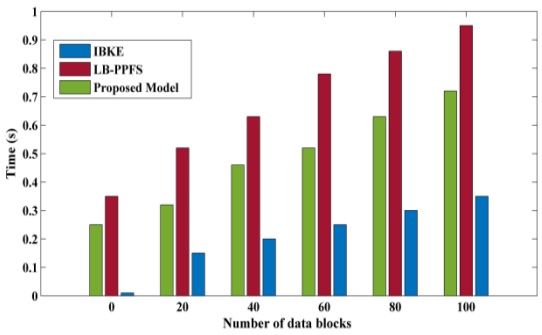

Fig 3: Graphical representation of the no. of data blocks Vs Time (s)

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

Table 3: No. of challenge blocks Vs Time (s)

No. of data blocks	IBKE	LB-PPFS	Proposed model
0	0.01	0.20	0.01
20	0.015	0.35	0.010
40	0.020	0.48	0.018
60	0.025	0.56	0.024
80	0.035	0.69	0.030
100	0.048	0.82	0.52

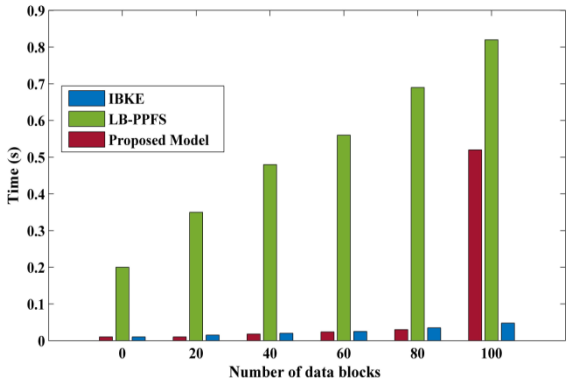

Fig 4: Graphical representation of the number of blocks Vs Time (s) in successive iteration

Table 2 depicts the comparison of the number of challenge data blocks Vs Time (s) where the block size ranges from 0 to 100 respectively. The anticipated model gives an outcome of 0.35, 0.52, 0.63, 0.78, 0.86, and 0.95 respectively. Fig 3 shows the pictorial representation of several challenge data blocks versus time in seconds. Table 3 shows the successive iteration of the number of challenge data blocks concerning time in seconds. The outcomes are given as 0.01, 0.010, 0.018, 0.024, 0.030 and 0.52 respectively.

Fig 4 shows the graphical representation of successive iterations with the data blocks. Table 4 shows the comparison of the number of challenge blocks with computational cost in the storage phase in seconds. The data blocks over the server storage range from 20000, 40000, 60000, 80000, and 100000 respectively. Fig 5 depicts the graphical representation of computational cost over storage phase in seconds. Table 5 explains the comparison of cloud functionalities with parameters like storage, block-less verification, probability sampling, privacy preservation, and security. The proposed model performs effectually in all the parameters while the existing approaches lack in forwarding security. Table 6 depicts the performance of the anticipated model with existing approaches. The proposed model gives better trade-off in contrast to prevailing approaches.

Table 4: Number of challenge data blocks Vs computation cost in storage phase (s)

Number of data blocks	IBKE	LB-PPFS	Proposed model
0	500	80	20
20000	800	90	35
40000	1100	110	52
60000	1350	200	63
80000	1450	350	85
100000	1800	480	110

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

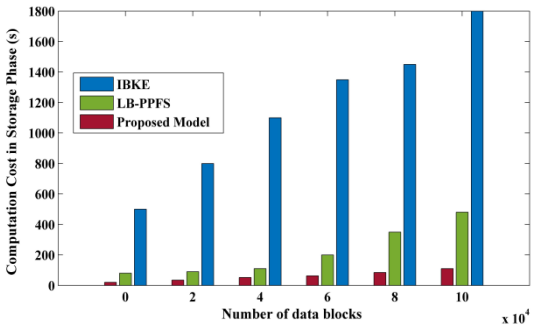

Fig 5: Graphical representation of the number of blocks Vs computational cost

Table 5: Comparison of cloud functionalities

Schemes	Storage correctness	Block less verification	Sampling	Public auditing	Privacy preservation	Security
IBKE	Yes	Yes	Yes	Yes	Yes	No
LB-PPFS	Yes	Yes	Yes	Yes	Yes	No
Proposed model	Yes	Yes	Yes	Yes	Yes	Yes

Table 6: Comparison of performance measures

Schemes	De-duplication	Privacy preservation	Efficiency
IBKE	Performed in the cloud storage system	---	Needs various authentication tags for auditing and duplication
LB-PPFS	---	File uploaded to TPA	Loss in bandwidth
Proposed model	---	----	Perform computation with a single tag and gives better authentication. Also ensures bandwidth benefits over client-side

V. CONCLUSION

This section discusses the data storage over a remote cloud where the users' needs to guarantee the outsourced data to be preserved more precisely in remote storage. Thus, there should not be any corruption over the cloud data. Additionally, the server needs to make use of storage more appropriately. To fulfil these requirements, the anticipated model needs to attain integrity over data auditing or data de-duplication in a cloud environment. The anticipated model adopts data de-duplication to avoid information leakage by preserving the unique data. Similarly, encrypted data should be maintained over the cloud server. The authentication tag is attached with the block files for proof of word to maintain auditing integrity. The anticipated model needs to fulfil security preservation. It offers better efficiency than the prevailing approaches like IBKE, and LB-PPFS respectively. The computational overhead needs to be examined in user's side. Finally, the anticipated model gives better performance and higher security. In future, this work is extended with the simulation of medical applications to evaluate storage capacity over the cloud environment. Similarly, optimization approaches can be adopted to measure the optimal solution attained to achieve efficient storage.

REFERENCES

[1] G. Ateniese, R. Di Pietro, “Scalable and efficient provable data possession,” Int. conf on Security and privacy in communication networks, pp. 1–10, 2008.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

[2] Y. Dodis, S. Vadhan, "Proofs of retrievability via hardness amplification," Int. conf on Theory of Cryptography Conference on Theory of Cryptography, pp. 109–127, 2009.

[3] C. Erway, "Dynamic provable data possession," ACM conf. on Comp and comm. security, pp. 213–222, 2009.

[4] S. Halevi, D. Harnik, "Proofs of ownership in remote storage systems," ACM conf. on Computer and communications security, pp. 491–500, 2011.

[5] S. Keelveedhi, "DupLESS: serveraided encryption for deduplicated storage," USENIX Security Symposium, pp. 179–194, 2013.

[6] X. Liu, Li, "Publicly verifiable inner product evaluation over outsourced data streams under multiple keys," IEEE Trans. on Services Computing, vol. 10 (5), pp. 826-838, 2017.

[7] J. Li, Cai, "Secure auditing and deduplicating data in cloud," IEEE Trans. on Computers, vol. 65 (8), 2386–96, 2017.

[8] T. Y. Youn, Shin, "Public Audit and Secure Deduplication in Cloud Storage using BLS signature," Research Briefs on Information & Communication Technology Evolution (ReBICTE), vol. 3 (14), 1-10, 2017.

[9] J. Yuan, Yu, "Secure and constant cost public cloud storage auditing with deduplication," IEEE Conf. on Comm. and Net. Security, 145- 153, 2013.

[10] J. Yuan, "Proofs of retrievability with public verifiability and constant communication cost in cloud," Int. workshop on Security in cloud computing, 19–26, 2013.

[11] HuiTian, "Enabling public auditability for operation behaviors in cloud storage", Soft Computing, 21 (8), 2175–87, 2017.

[12] Smitha Sundareswaran, "Ensuring distributed accountability for data sharing in the cloud", IEEE Trans. on Dependable and Secure Computing, Vol.9 (4), 556–68, 2012.

[13] Zhen Yang, "Ensuring reliable logging for data accountability in untrusted cloud storage", IEEE Int. Conf. on Communications, 1966–71, 2017.

[14] HuiTian, "Dynamichash- table based public auditing for secure cloud storage", IEEE Trans. on Services Computing, Vol.10 (5), pp.701–714, 2015.

[15] Ryan K.L. "Trustcloud: A framework for accountability and trust in cloud computing", IEEE World Congress on Services, pp.584–88, 2011.

[16] M. W. Storer, ``Secure data deduplication," in Proc. storage, Alexandria, pp. 1-10, 2008.

[17] S. Halevi, ``Proofs of ownership in remote storage systems," Proc. CCS, pp. 491-500, 2011.

[18] R. Di Pietro ``Boosting ef_ciency and security in proof of ownership for deduplication," ASIACCS, 2012.

[19] W. K. Ng, ``Private data deduplication protocols in cloud storage," in Proc. SAC, 2012

[20] Douceur, ``Reclaiming space from duplicate _les in a serverless distributed _le system," ICDCS, 2002

[21] M. Bellare, S. Keelveedhi, ``DupLESS: Serveraided encryption for deduplicated storage," in Proc. USENIX Secur., pp. 179-94, 2013

[22] Q.Wang, C.Wang, "Enabling public auditability and data dynamics for storage security in cloud computing," IEEE Trans. Parallel Distrib. Syst., 22 (5), pp. 847_859, 2011.

[23] Tian, Y. Chen, ``Dynamic-hash-table based public auditing for secure cloud storage," IEEE Trans. Services Comput., vol. 10 (5), pp. 701_714, 2017.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 4, 2021, Pages. 5730 - 5740
Received 05 March 2021; Accepted 01 April 2021.

[24] Juels, ``PORs: Proofs of retrievability for large _les," in Proc. CCS, 2007

[25] Shacham ``Compact proofs of retrievability," J. Cryptol., vol. 26 (3), pp. 442_483, 2013.

[26] Yuan, S. Yu, ``Proofs of retrievability with public veri_ability and constant communication cost in cloud," Int. Workshop Secure. Cloud Comput., 2013

[27] Zheng, ``Secure and ef_cient proof of storage with deduplication," Proc of codaspy, 2012

[28] Yuan, ``Secure and constant cost public cloud storage auditing with deduplication," IEEE cns 2013

[29] C. Li, ``A secure privacy-preserving cloud auditing scheme with data deduplication," Int. J. Netw. Secure., vol. 21 (2), 2019

[30] Li, Cai, ``Secure auditing and deduplicating data in cloud," IEEE Trans. Comput., vol. 65 (8), pp. 2386_2396, 2016.

[31] Feng Wang, "Lightweight Certificate-Based Public/Private Auditing Scheme Based on Bilinear Pairing for Cloud Storage ", IEEE Access, pp. 2258-2271, 2020.

[32] Xiong Li, "Comments on "A Public Auditing Protocol With Novel Dynamic Structure for Cloud Data ", IEEE Trans. on Information Forensics and Security, 2881-2883, 2020