How to Cite:
Joseph, L. M., & Fredrik, E. J. T. (2022). Ensuring the security for cloud storage data using a novel ADVP protocol by multiple auditing. *International Journal of Health Sciences*, *6*(S2), 9794–9812. https://doi.org/10.53730/ijhs.v6nS2.7561

Ensuring the security for cloud storage data using a novel ADVP protocol by multiple auditing

Libin M Joseph
Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore
Corresponding author email: libinmj@gmail.com

Dr. E. J. Thomson Fredrik
Professor, Department of Computer Applications, Karpagam Academy of Higher Education, Coimbatore
Email: thomson500@gmail.com

***Abstract*---**The increasing growth of storage data in the cloud and virtual reality allows it a significant challenge to maintain the security of data that is outsourced by the data owners. The existing protocols for the auditing of cloud storage normally use post-quantum cryptography to monitor data integrity to solve this issue. Nevertheless, these protocols use strong cryptography to create data tags that reduce their reliability and extensibility. In this research work, we propose a novel protocol named "Advanced Distribution Verification Protocol (ADVP)" to design secure cloud storage to track the quality of cloud-saved data with the help of "Multiple-Third Party Auditors" (mTPAs) and not a "Single-TPA" (sTPA). This protocol requires several SUBTPAs that operate throughout the single TPA, which then needs to be spread equally throughout the SUBTPAs to guarantee that every SUBTPA is checked throughout the entire section. It then has strict safety evidence of malicious cloud resistance and a promise of privacy. It would also broaden the suggested protocol and include additional implementation scenarios to enable data dynamics and batch audit. It sums up a structured process to establish security as a further contribution. The performance review and evaluation are eventually carried out, to ensure that both operating reliability and functional extensibility. The performance metrics for ADVP and "Hybrid-Secure Cloud-Storage (HSCS)" compared, in this ADVP would provide greater security than HSCS, including Auditing-Time, Encryption-Time, Decryption-Time, and Storage-Overhead.

***Keywords*---**cloud, security cryptography, HSCS, ADVP.

Introduction

In the future era the extremely skillful, internet-based distributed computing platforms are provided by "Cloud Computing (CC)", which offers "as a service". The infrastructure model of the cloud-focused on distributed storage, grids, networking, virtualization platforms, and market-oriented computing. Below are the most commonly encountered CC definitions: The "National Institute of Standards and Technology (NIST)" in the United States termed "CC is a model to enable fast, on-demand network access to a popular pool of configurable computing tools (e.g. Networks, servers, storage, software, and services), easily accessible and released with limited maintenance effort or contact between service providers" [1]. A cloud is a sort of distributed processing infrastructure consisting of a series of loosely funded and virtualized machines presented as one or more centralized computational services focused on "Service Level Agreements (SLA)" reached through negotiation with customers.

Security is one of the key challenges to adoption and it addresses this detail in this research work. The "International Data Corporation (IDC)" surveyed CC in September 2019 showed that protection is the key concern, as most CC clients are worried about data from their company and vital cloud-based IT services that are vulnerable [2]. According to an IDC survey conducted by 896 Executives from Information Technology/CEOs and their "Line Of Business (LOB)" employees on the usage of their companies and their views on IT Cloud resources, the efficiency and availability problems remain as three major challenges.

Hence, the CC has no inbuilt solution for these problems, its capabilities, service models, and implementation modeling also increase the need for new protection concerns such as Security for DataStorage, Security in Networks, PrivacyPreservation, Security for Virtualization and so on. Additional monetary capital would be required to enforce these protection implementations [3]. Cloud providers provide fewer robust SLAs than anticipated criteria for an IT cloud provider. There is currently a shortage of accountability and protection practices in this regard [4]. It will require reliable and successful approaches to cope with the following protection problems:

Protection for Infrastructure

In comparison, cloud storage does not explicitly face protection difficulties in its models, including the network level, the host level, and the device level. The problems of Cloud Storage and Infrastructure Protection may be answered by specifically identifying confidence limits and knowing who supplies which aspect of security [5].

Storage and Security of the Data

The security of data is a very complicated and critical activity. Security of data can be seen with considerable care, such as drafting, truncation, obscurement, and others. Not only are these alternate approaches appropriate for this criteria, but there are still no programs to verify the frameworks of something that might

be created. Data authentication may be used with homomorphic encryption, although this method is troublesome [6].

Confidentiality

Confidentiality for cloud storage is an urgent issue, both concerning legal enforcement and the trust of customers, and it should be taken into consideration in all design phases [7]. The biggest obstacle for software developers to develop cloud systems is the privacy concerns that are to be minimized and regulatory enforcement is guaranteed.

For cloud system engineers, architects, creators, and reviewers the following tips are suggested [8].

- In the cloud protect sensitive details submission and preservation.
- In the cloud personal records need to be protected.
- Control of the user should be maximized.
- Choices should be given to the users.
- The usage of data should be limited and specified.
- Feedback should be provided.

Enforcement and Audit

The Cloud Service Provider (CSP) and its customers are helping to plan a programmatic solution to monitoring and enforcement to meet evolving standards and cloud business models [9]. CSPs must incorporate a powerful internal inspection feature paired with a comprehensive external audit mechanism to drive performance, risk control, and enforcement. CSP users have to identify their monitors, grasp their CSP internal monitoring procedures, review the related external audit reports and conduct their roles as CSP users adequately to achieve comfort in their in-cloud operations [10].

This research work problem statement deals with cloud data security whereas the clients retain their cloud data in their data management system and no longer have to process the data in the local environment. The user loses power when data is entered into the cloud. If this data storage is susceptible to attacks or byzantine failures where a competitor may manipulate or erase the data in the storage servers or insert contaminated data or view the data. These attacks or threats will contribute to irrecoverable consumer damages while their data were placed beyond storage companies in an unknown storage pool. The attacks block consumers from fully viewing the original results. Therefore, efficacious audit procedures are mandatory, with minimal overhead measurements, connectivity, and storage to maintain the security, fairness, and usability of customer data over a lifetime.

The contribution of this research work is to propose a novel protocol named "Advanced Distribution Verification Protocol (ADVP)" to track the quality of cloud-saved data with the help of multiple-TPAs and not a single-TPA. This protocol requires several SUBTPAs that operate throughout the single-TPA, which then

needs to be spread equally throughout the SUBTPAs to guarantee that every SUBTPA is checked throughout the entire section.

The objective of this research work is to ensure the security of the data as follows:

- The proposed Protocol ADVP needs to support public verifiability and enables TPA to validate the cloud storage protection on behalf of customers and also support complex data operations for useful applications such as alteration, insertion, and deletion.
- It needs to give evidence that the security methods perform against attacks internally and externally are safeguarded from data. Only if they provide all data in an incorrupt and satisfied condition then the only cloud will provide services, this will be a good solution to the verifier problems.
- The work will conclude with the data integrity through specific analysis, experimental findings, and similarities by the existing HSCS protocol with the proposed ADVP protocol.

The remainder of this article is categorized as follows: Section 2 covers the recent reviews related to cloud's security, Section 3 briefs about the methodologies with the existing and proposed system, Section 4 discuss the implementation results with the comparison of both existing and proposed system, and Finally, Section 5 concludes this research article with future scope.

Related Works

In [11] the researchers proposed a cloud computing framework with 2 authentication methods. The One Time Password (TOTP) time-based data has been protected by the cloud user accessing data and the third-party data are maintained by the Automatic Blocker Protocol (ABP). This led to the enhancement of data integrity and device consistency. The framework assures the privacy of cloud and TPA data from organizations. The period needed for the key generation did not decrease to the amount required.

In [12] the researchers implemented an authentication protocol focused on smart cards utilizing the chaoticmaps of Chebyshev. Authentication protocol based on intelligent cards supported protection, authorization, and authentication. The two-factor protection solution required two authentication credentials. Increased protection was provided for both traceability and confidentiality. Moreover, the efficiency of protection has deteriorated due to various attacks in cloud storage. Attacks like Multiple replicas were not detected using the authentication protocol dependent on the intelligent card.

In [13] the researchers have established a Shared Storage Mechanism where the data could only be decrypted by an authorized consumer. To prohibit attachments and limit the manipulation of saved files, the access control mechanism has been decentralized. The data are thereby stored in the cloud and avoided wrongdoing. This strengthens the mechanism of secrecy and authentication. By utilizing the decentralized storage method, data integrity was not enhanced.

In [14] the researchers analyzed the device access management scheme. For cloud infrastructure with authorization algorithms, a complex access management scheme has been included. Then, the correct type of access was also incorporated with the regulation model, a model for access control management, and the model of authorization. The contact module for appropriate message transmitting has been equipped with an upgraded protection scheme. But the results of this user access management framework struggled to increase the reliability of task scheduling.

In [15] the researchers have developed a privacy rights index searching algorithm for transmitting the data. Without relying on the trapdoors listed, a conjunctive search query was introduced. Attacks of unauthorized users and untrusted people contributing to a lack of privacy are stopped. The confidentiality-friendly form of query often resolved traditional trapdoor search questions. But in their results, the Privacy-Preserving was not properly enhanced.

Methodologies

Design Methodology

In CC the cloud-storage model comprises 3 modules, namely Clients/Customers, Cloud Service Provider (CSP), and Third Party Auditor (TPA) as shown in Figure 1.

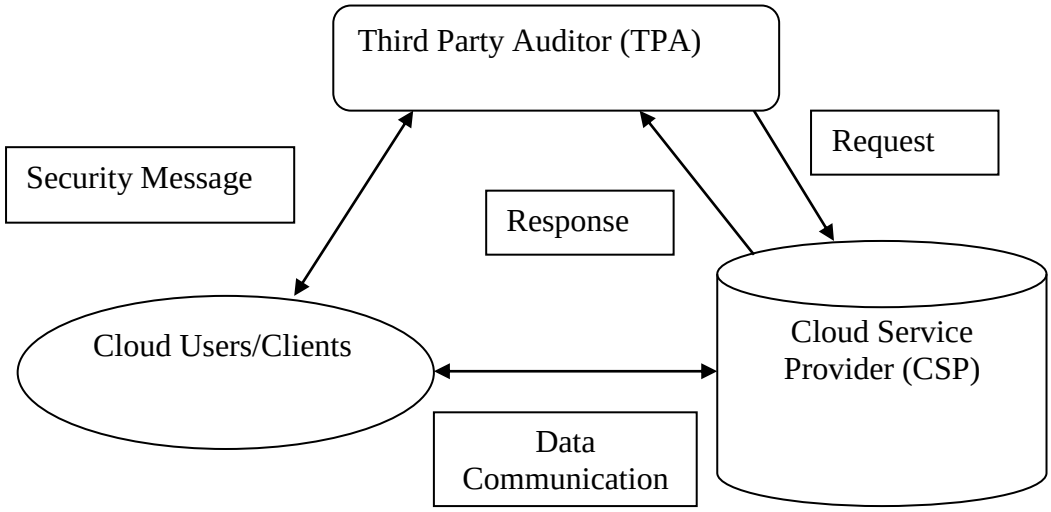

Figure 1: Architecture for Cloud Data Storage

Clients/Customers

The customers must store and access data through CSP. These generally contain computers for desktops, notebooks, smartphones, tablets, etc.

Cloud Service Provider (CSP)

The CSPs involve individuals with substantial construction capital and experience, the operation of remote cloud computing servers, software, technology, and hardware for helping Customers to utilize the "Infrastructure as a Service (IaaS)".

Third Party Auditor (TPA)

The TPA with skills and resources in which it possesses and verifies the integrity of cloud-stored data on behalf of the customers. TPA will publish an audit report to the client based on the audit findings. Clients maintain their data files in the cloud in a cloud-based paradigm and access the data in the cloud through assistance from CSPs. The cloud comprises a series of cloud servers that run simultaneously, collaboratively, and distributively. Replication of the data should be used to better withstand failures or server crashes using erasure-correcting programming methods since user data and data crypt can avoid data leakage. Moreover, the user will also check data integrity without a local backup of the data file. If the customers are not willing to track their details in time, workability, or money, they assign it to TPA. The TPA will test on behalf of customers the integrity of the results. Often, for useful purposes, clients can need to conduct block-level operations on their results. Block changes, deletes, inserts, and updations are the most popular types of this process.

Design goals

The design mechanism to be developed by which the data from a storage server may be safely processed (potentially very large) without being retrieved often, easily, and securely checked. In the proposed design, first, before outsourcing to the remote server, the client calculates the metadata for the file. Then the server may be confronted with Challenge-Response (CR) protocol for file integrity. Then, only the metadata is retained by the Client or sent to the own agent (TPA), then the file is forwarded to the remote server, and the copy in the local location is erased. Later, any time the verification (the initial user or the trustworthy TPA) needs to remotely validate the storage data in the server is called upon to prove the security of the storage data. When the verifier has a challenge, the server can measure the reply and give it back to the verifier as evidence of the data storage. The verifier then checks the consistency of the data in comparison with metadata previously measured. If real, the verifier is confident that the data on the server is secure.

Hybrid-Secure Cloud-Storage (HSCS): (Existing Model)

The HSCS is a hybrid concept that combines the "Schnorr Elliptical Curve (SEC)" framework with the "Bloom-Filter (BF)". It contains several modules to provide the trade-off between the high security and less execution time as follows: Entity Module, Security Module, Audit Module, Generation and Validation of Signatures Modules, and Deduplication Module. The module for entity initialization is the preliminary step in the HSCS algorithm. Hence, the modules of an entity and the security are the major process to improve security. The users/data owners

present in the system can store a large amount of data. The assumptions for entity modules are listed as computation of data owner data's metadata neither concerning the security-keys, reliable data storage in the cloud and the interaction between the old and updated data is dynamic. The TPA is the main concern of the framework model, with no details being leaked although during the audit process period. However, the emergence of threats leads to data theft. The audit phase is then carried out using the modules developed to resolve the different protection problems and threats. SEC-based cryptography authentication is then done on a cloud protection model. Eventually, using the improved SEC-BF method, the overall protection efficiency is improved with far reduced processing time and overhead [16].

Advanced Distribution Verification Protocol (ADVP): (Proposed Model)

This section entails the proposed Advanced Distribution Verification Protocol (ADVP) to verify the integrity of the cloud data, backed by Multiple Verifiers (i.e mTPAs) rather than Single Verifiers (i.e sTPA). This protocol is transmitted to the cloud.

For the assurance of the credibility of the data, it was dependent on Challenge-Response Protocol. Here it uses one TPA in one auditor verification protocol. The TPA holds the metadata referring to the file blocks in this authentication phase and provides a task for the CSP. The CSP produces the honesty evaluation for the related challenge and response to the TPA. Following that, TPA tests the response for the metadata originally present and provides the Client with the final audit result. In this single auditing method, though, the whole phase of inspection will be aborted when TPA crashes because of a heavy workload. Also, the network activity near the TPA organization and network congestion may be very large throughout the verification phase. In single audit authentication systems, the efficiency would then decline.

Thus, it requires a mandating verification procedure, under which a multiple of mSUBTPAs operates simultaneously with the sTPA and workload must therefore be equally distributed around the whole portion of the SUBTPAs such that each SUBTPA verifies. In contrast with single verifier programs, the distribution authentication protocol can easily detect data manipulation in the cloud.

ADVP Architecture

The Protection measures are distributed via this protocol. Here the verifier 'n' universally challenges the servers 'n' and if the server 'm' response is right, it may assume that data integrity is assured. Figure 2 shows the architecture of the proposed model.

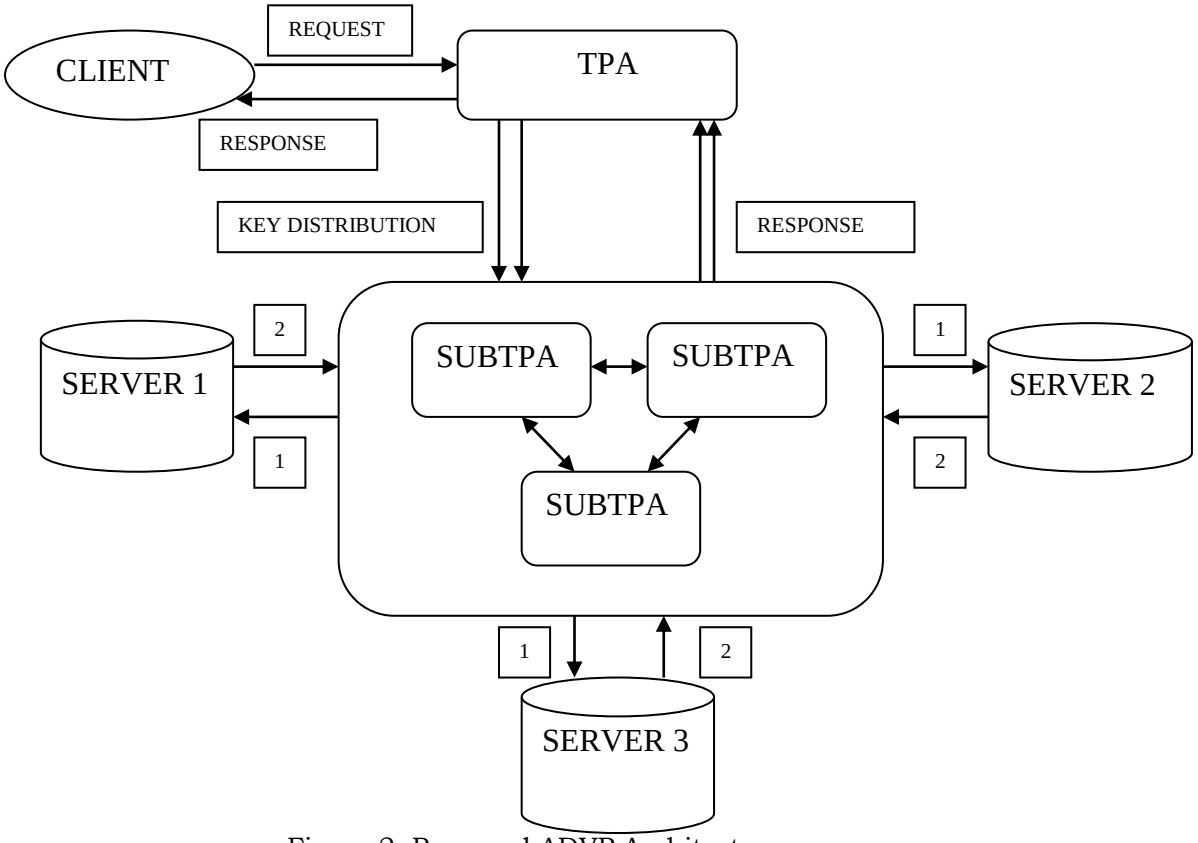

Figure 2: Proposed ADVP Architecture

Here "1" represents as the verifier generates the challenge and sends it to the server and "2" represents as the server computes the response and sends it back to the verifier. The verifier compares the response with the precomputed metadata.

The verification method uses many TPAs to check the integrity of the results. Of the various TPAs, one TPA will function as the key TPA, and SUBTPA will stay. If the main TPAs fail, one of SUBTPA functions as the main TPA, and both SUBTPAs are used to effectively identify data manipulation. SUBTPAs don't communicate with each other rather want to confirm the integrity of the data held in the cloud and the accuracy of the responses given by the provider. The framework proposed ensures all of TPA's nuclear operations. (i.e) TPA that monitors every process of the SUBTPA's is available which means that its operations are equivalent atomically, as well as the operations whose results it sees.

Centrally managed and distributed data paradigm that manages all SUBTPAs through the TPA and communication from the SUBTPA to every Cloud's Data Storage Server to be checked as given in Figure 2. A synchronous framework with many TPAs and servers is considered. Each SUBTPA is linked to the Server through a secure synchronous channel that provides the server with challenges. The Server and SUBTPA are usually termed as 'P' parties.

A policy points out both parties' behaviors. Here 'P' is a sequence of alternate states and state changes defined as incidents happening according to the machine component specification. They do not fail. All SUBTPA's adapt to the protocol. Each SUBTPA has a tiny, trustworthy local memory to store distribution and authentication keys. It is also considered to be a defective or malicious server that arbitrarily deviates from the protocol. The 'P' Parties are right and don't struggle to work.

ADVP Working Principle

There are three stages in this ADVP: 1) Configuration of the System 2) Phase for Verification 3) Operation of Data Dynamically. The first two phases are common like other protocols.

This research mainly concentrates primarily on the phase for verification. There a 3 processes in this phase. They are (i) KeyDistribution (ii) VerificationProcessing (iii) ValidatingIntegrity.

Key Distribution

The TPA produces the random key in this phase and distributes the following to its SUBTPAs:

The SOBOL Random function first produces the Random key as given in the below as:

$K = f_{k1}(i)$ **Eq->1**

Here $1 \le i \le n$ and the key is indexed on secret-key: f:{0, 1}* x key→Z_p

The SecretSharingScheme (m, n) is used and the 'K' RandomKey is partitioned to 'n' bits. The Client selects a polynomial an (x), with degree m-1, to separate 'k' into 'n' sections and calculates the 'n' parts as:

$K_j = K + a_1 i + a_2 i^2 + .. + a_k i^{m-1}$ **Eq->2**

$K_j = K + \sum_{j=1}^{n} a_j \, i^j$ **Eq->3**

Then the TPA picks and distributes 'n' parts to all chosen nSUBTPAs. Algorithm 1 defines the method of KeyDistribution.

Key Generation and Distribution: ADVP

1. Generates a random key, K using Sobol Sequence.
   $K = f_{k_1}(i)$
2. Then, the TPA partition the K into n pieces using (m, n) secret sharing scheme
3. TPA select the Number of SUBTPAs, n, and threshold value m;
4. **for** i ←1 to n **do**
5. TPA send k_1 to the all $SUBTPA_i$ s
6. end **for**
7. end

Verification Process

In this phase, while inspecting all SUBTPAs validates the data integrity and transfer results to the TPA. The TPA assumes that data integrity is correct when the 'm' SUBTPA answer reaches this threshold value. The procedure functions as follows at a higher level: Each SUBTPA activity is given a local timestamp by a TPA. Each SUBTPA then preserves in its trustworthy memory a timestamp vector 'T'. In any SUBTPAi, the T[j] entry is equivalent to the timestamp of SUBTPAj's most recent service.

Each SUBTPA provides a challenge to check data integrity and sends it to the CSP by the following: Initially, it produces a Randomindices 'c' a collection of [1, n] using a RandomKeyfrom SobolRandomPermutation (SRP).

$$j = \pi_{Kj}(c) \qquad \textbf{Eq->4}$$

Here 1≤c≤1 and $\pi_{Key}(.)$ is a SRP, which is indexed under key: π:$\{0, 1\}^{\log2(1)}$ x key-$\{0, 1\}^{\log2(1)}$. Then every SUBTPA can also pick a new RandomKey 'r_j', where

$$r_j = f_{k2}(l) \qquad \textbf{Eq->5}$$

- Instead, the chal=$\{j, r_j\}$ as a challenge was a pair of RandomIndices. Every SUBTPA will wait for responses by placing the request for the CSP.
- Then CSP determines the solution to the challenges of the SUBTPA and replies to SUBTPAs.
- The time stamp is checked for initially by the SUBTPA, ensuring that V ≥T (by VectorComparison) and V [i] = T[i].
- The Client may discontinue the process and quit since it implies the server is in breach of service continuity when the condition is If NOT.
- ELSE uses the SUBTPACOMMIT to confirm whether or not the metadata and responses stored (IntegrityProof) are correct?

- It Stores TRUE into its table and sends this signal to TPA while the condition is correct, else it saves as FALSE and transmits the message to the TPA to erase corrupted blocks of data.

The VerificationProcess is detailed in Algorithm 2:

Verification Process: ADVP

```
1.  Procedure: Verification Process
2.  Timestamp T
3.  Each SUBTPA_i computes
4.          Compute  j = π_{k_SRP}(c)
5.  the Generate the sobol random key r_j
6.  Send (Chal=(j, r_j)  as a challenge to the CSP;
7.  the server computes the Proof  PR_i send back to the SUBTPAs;
8.  PR_i ←Receive(V);
9.       If(V≥TΛ V [i] = T[i])
10.        return COMMIT then
11.              if PR_i equals to Stored Metadata then
12.                 return TRUE ;
13.                 Send Signal, (Packet_j, TRUE_i) to the TPA
14.               else
15.               return FALSE;
16.               Send Signal, (Packet_i,  FALSE_i) to the TPA;
17.         end if
18.    else
19.        ABORT and halt the process
20. end if
21. end
```

Validating Integrity

The TPA receives a report from every subset 'm' out of 'n' SUBTPAs to verify the accuracy of the data and validate its integrity. The TPA determines that the data has little impact otherwise it declares that the data has been changed, while the 'm' SUBTPAs provide the COMMIT signal to TPA. The TPA would transfer for the customer with an audit result in the final phase. The integrity validation processes are given in Algorithm 3, in which the distribution of integrity can be applied in any verification protocol. It may therefore use cloud distribution authentication.

Validating Integrity: ADVP

```
1.  Procedure: validation(i)
2.  TPA receives the response from the m SUBTPAs
3.   for i←1 to m do
4.     If(response==TRUE)
5.        Integrity of data is valid
6.   else if(response==FALSE)
7.        Integrity is not valid
8.     end if
9.  end for
10. end
```

Advantages of Proposed System

- Verification operations will of course raise the overheads of the infrastructure for connectivity and computing. Here the SecretSharing is used to share the 'k' key to increase the output, which offers minimal overhead during communication and reduces the compute difficulty. It thus eliminates the overhead coordination between TPA and SUBTPAs.
- The TPA can adjust the 'k' key to any SUBTPA, and can only submit the multiset portion to the SUBTPA for further verification. The SobolSequences based probabilistic testing schemes were used to ensure uniformity not just for whole sequences but also for each subsequency, such that each SUBTPA verifies the file blocks separately. So the chance of identifying the failure site is high.
- A SobolSequence also offers clear justification for the credibility of data processed remotely. In ADVP, it creates a file block number using Sobol's RandomSequence producer, since the sequence is spread equally over [0,1] and occupies the entire area. To construct integer numbers, the sequences created will increase the constant power of two.

Results and Discussions

These protocols have been deployed using the Open-Stack cloud environment and it has measured the time required to encrypt, decrypt and audit the user authentication data access service, storing data service for secure object storage and data sharing service to ensure the versatile data sharing with the users that are managed by data proprietors. The system was implemented through the eclipse android simulator which is operating on the Intel-core I5 processor in Windows 7 on desktop at 260GHz and has a RAM of 8GB. The data owner and users utilize android-based data communication and distribution applications during this research to access cloud data resources. Under Java Environment, algorithms are written.

Audit-Time

The audit-time denotes the time required to audit the file for verifying the data integrity. The auditing time is defined as the ratio of total auditing time to the number of auditing tasks. A lower value of the auditing time indicates the

efficiency of the proposed algorithm and high data security. Thus, the delay for computation concerning the number of auditing requests is lesser for the proposed system compared to the existing system.

Table 1: Audit-Time Performance

AUDITING REQUEST	HSCS	ADVP
10	450	300
30	500	350
50	550	400
70	650	500
90	750	600

Table 1 shows the auditing time analysis concerning the number of auditing requests. From the table, it is observed that there is a decrease in the computational delay concerning the increase in the number of auditing requests. The proposed method achieves a lower computational delay of about 35% than the existing system.

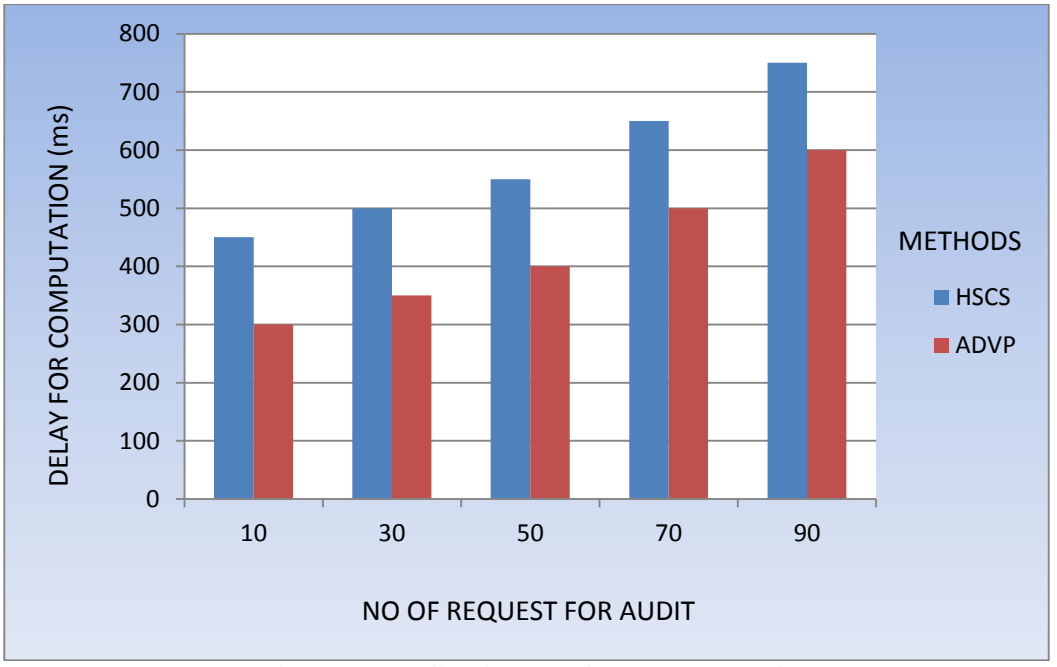

Figure 3: Audit-Time Performance Graph

Figure 3 shows the time to audit concerning a total request for audit. The X-axis of the graph represents the total request for audit and Y-axis represents the computational delay. The computational delay is indicated in milliseconds (ms). From the graph, it is observed that the proposed system achieves a lower computational delay than the existing system.

Encryption-Time

The time for encryption is termed as the time taken for the method to encrypt the file utilizing a secret-key. The Input/Output (IO) file time is not included.

Table 2: Encryption-Time Performance

NO OF ATTRIBUTES	HSCS	ADVP
10	0.7	0.3
30	1.1	0.6
50	1.6	0.9
70	2.1	1.3
90	2.7	1.9

Table 2 presents the comparative analysis of the encryption time on data owners for the proposed scheme and existing scheme. The variation in the encryption time is analyzed concerning total attributes. The attributes total varies from 10-90. Initially, the encryption time of the proposed scheme is 0.3 seconds and the existing scheme is 0.7 seconds. For 90 attributes, the encryption time of the proposed scheme is 1.9 seconds and the existing scheme is 2.7 seconds.

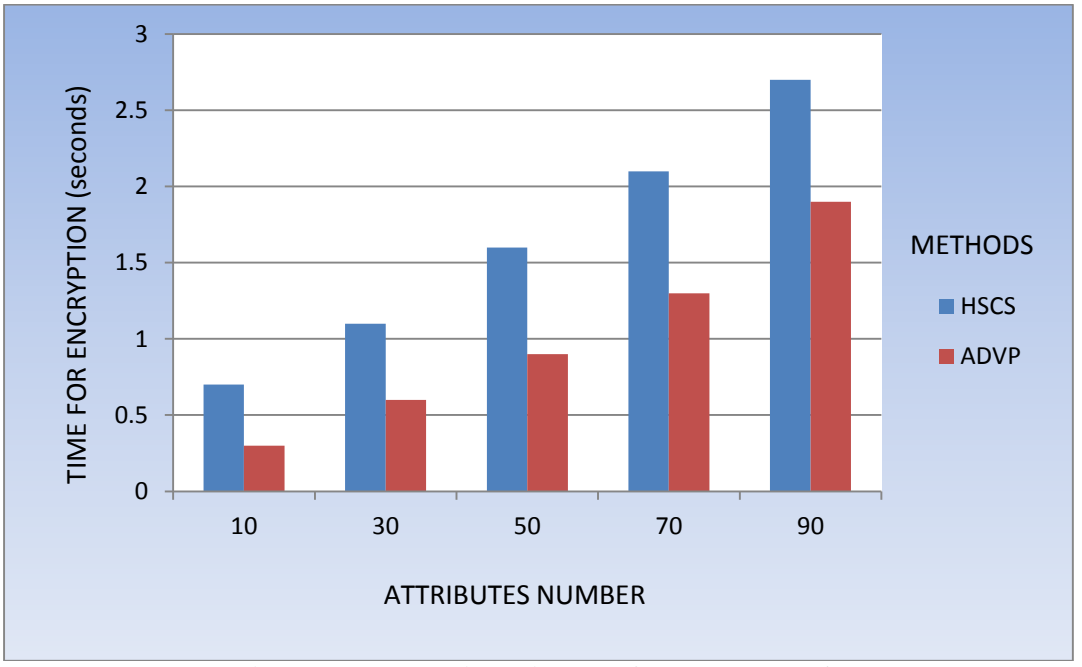

Figure 4: Encryption-Time Performance Graph

Figure 4 illustrates the encryption time on the data owner for the proposed ADVP and existing HSCS. In the graph, the X-axis denotes the number of attributes and Y-axis denotes the encryption time. The encryption time is measured in seconds. The encryption time increases with the increase in the number of attributes. The encryption time of the proposed scheme is lower than the existing scheme.

Decryption Time

The time for decryption is termed as the time took for the method to decrypt the file utilizing a secret-key. There is a linear increase in the encryption and decryption time with the increase in the length of the primes. Lower encryption and decryption time shows the efficiency of the proposed security schemes.

Table 3: Decryption-Time Performance

NO OF ATTRIBUTES	HSCS	ADVP
10	0.5	0.2
30	0.7	0.4
50	0.9	0.5
70	1.3	0.7
90	1.7	0.9

Table 3 presents the comparative analysis of the decryption time on data owners for the proposed scheme and existing scheme. The variation in the decryption time is analyzed concerning attributes total. Attributes total varies from 10-90. Initially, the decryption time of the proposed scheme is 0.2 seconds and the existing scheme is 0.5 seconds. For 90 attributes, the decryption time of the proposed scheme is 0.9 seconds and the existing scheme is 1.7 seconds.

Figure 5 shows the decryption time on the user for the proposed ADVP scheme and the existing HSCS scheme. The X-axis shows the number of attributes and the Y-axis represents the decryption time in seconds. The decryption time of the proposed ADVP scheme is lower than the existing HSCS scheme.

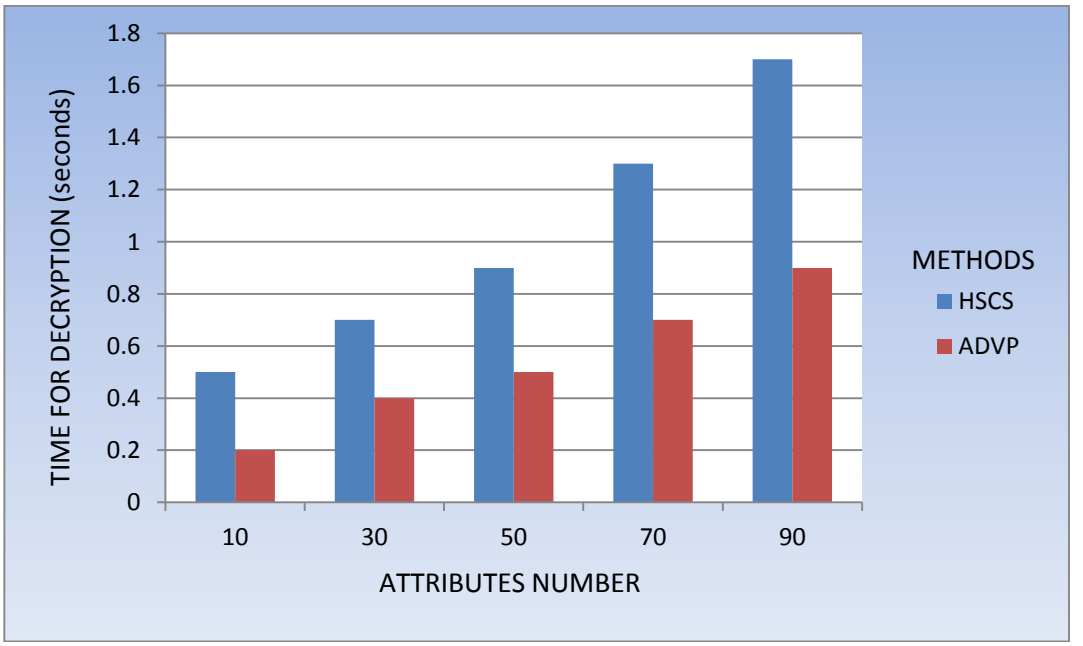

Figure 5: Encryption-Time Performance Graph

Storage-Overhead

The storage-overhead is the additional storage space required for storing the necessary information other than the file. The storage-overhead denotes the amount of data stored at the CSP side. Reduction in the storage-overhead at the CSP side is a key feature to reduce the service charge paid by the customer. The TPA allows users to check data integrity. The TPA helps to check the freshness of data but at the same time increases storage complexity. The computation delay overhead is more compared to the data auditing with TPA. Hence, the storage overhead with TPA is generally higher but it improves security. There is an increase in the storage overhead corresponding to the increase in the number of servers. Here, the proposed system is highly efficient than the existing techniques with less storage-overhead.

Table 4: Storage-Overhead Performance

SERVER DENSITY	HSCS	ADVP
10	1600	1100
15	1700	1200
20	1800	1250
25	2100	1400
30	2400	1550

Table 4 illustrates the storage overhead analysis concerning server total. Server total ranges from 10-30. For 10 servers, the storage overhead for the proposed system is 1100 KB/s and the existing system is 1600 KB/s. For 40 servers, the storage overhead for the proposed system is 1550 KB/s and the existing system is 2400 KB/s. The proposed system yields a lower storage overhead of about 40% than the existing system.

Figure 6 shows the overhead of storage overhead concerning server total. The X-axis represents the number of servers and Y-axis represents the storage overhead. The storage overhead is denoted in Kilobyte per second (KB/s). The graph shows that the storage overhead of the proposed system ADVP is lower than the existing techniques HSCS.

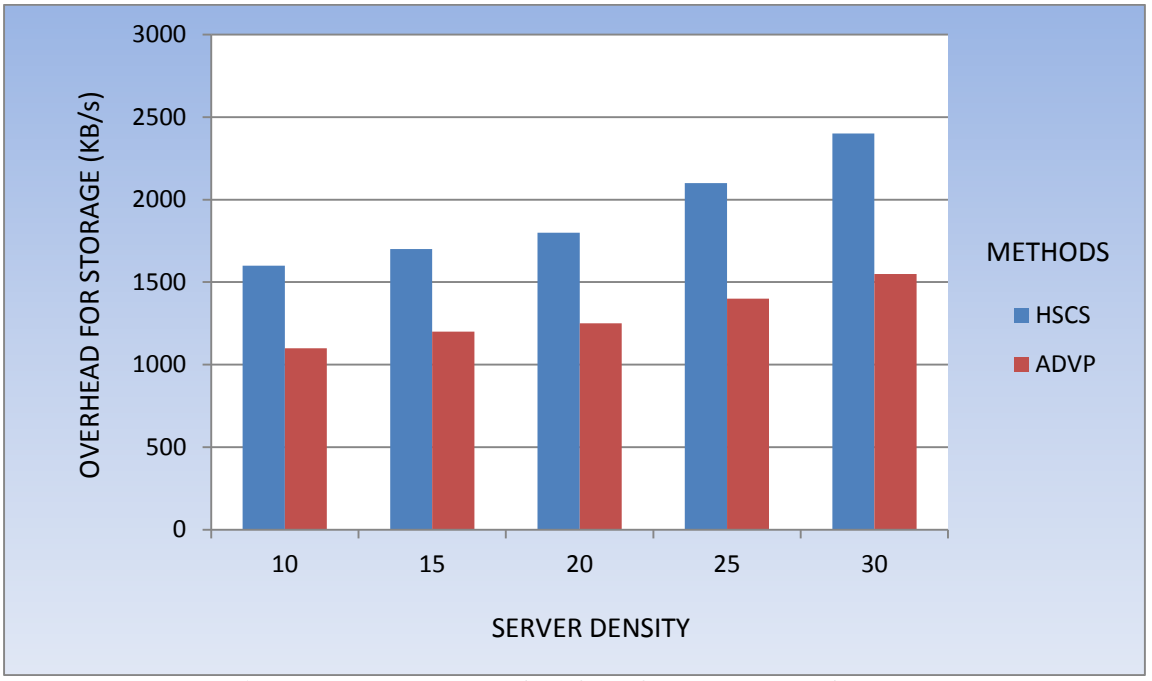

Figure 6: Storage-Overhead Performance Graph

Conclusion

More and more citizens and companies outsource their data to central repositories, reducing the responsibility of local data collection and management. The confusion about the protection of external computer environments poses many security problems concerning availability, authorization, authentication, legitimacy, integrity, and data credibility. To accelerate cloud storage, too many enhancements are required in existing systems and more developed and emerging protection frameworks. In this research work, we proposed a novel ADVP protocol that was modern, effective, and stable. This process ensures data access by erasure codes and data security through data encryption techniques. The core component of this system is spot inspection assistance that assures that the proposals stay lightweight and metadata that enables data accuracy to be checked without access to the original data file and awareness of full data. In this suggested system, the probabilistic evidence of credibility was shown by the usage of SobolSequence from the CSP, which greatly decreased the overhead calculation and correspondence. Moreover, the proposed schemes would require a TPA, on behalf of Clients, to verify the integrity of the Cloud with the Public-Key of the Client, to verify that the verifier has verification of data integrity, even if it does not own or access the CSP file blocks. This public verification would allow the verifier to check data integrity in full. This ADVP used multiple-TPAs to improve the utility of this protocol. This protocol often supports the externalization of complex data through which the Client is not only capable of archiving and accessing the CSP saved data copies, but also to refresh and size these copies on remote servers. It has been shown by comprehensive performance and experimental findings that the proposed ADVP transcends the existing HSCS, which can be interpreted as an advanced extension centered on a PseudoRandom

series. In the future, this work can be extended with advanced swarm optimization methods.

References

1. Herman, M., Iorga, M., Salim, A. M., Jackson, R. H., Hurst, M. R., Leo, R., Sardinas, R. (2020). NIST Cloud Computing Forensic Science Challenges. doi:10.6028/nist.ir.8006
2. Bhardwaj, A., & Goundar, S. (2020). Cloud Computing Security Services to Mitigate DDoS Attacks. Cloud Computing Security [Working Title]. doi:10.5772/intechopen.92683
3. Gupta, B. B., & Badve, O. P. (2016). Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a Cloud computing environment. Neural Computing and Applications, 28(12), 3655-3682. doi:10.1007/s00521-016-2317-5
4. Yousif, M. (2017). The State of the Cloud. IEEE Cloud Computing, 4(1), 4-5. doi:10.1109/mcc.2017.4
5. Liu, T., & Wu, G. (2018). Universal SaaS platform of internet of things for real-time monitoring. doi:10.1063/1.5033740
6. Stevenson, D. M., & Pasek, J. (2015). Privacy Concern, Trust, and Desire for Content Personalization. SSRN Electronic Journal. doi:10.2139/ssrn.2587541
7. Shynu, P. G., & Singh, K. J. (2016). A Comprehensive Survey and Analysis on Access Control Schemes in Cloud Environment. Cybernetics and Information Technologies, 16(1), 19-38. doi:10.1515/cait-2016-0002
8. Zhang, R., Ma, H., & Lu, Y. (2017). Fine-grained access control system based on fully outsourced attribute-based encryption. Journal of Systems and Software, 125, 344-353. doi:10.1016/j.jss.2016.12.018
9. Zhang, J., Chen, B., Zhao, Y., Cheng, X., & Hu, F. (2018). Data Security and Privacy-Preserving in Edge Computing Paradigm: Survey and Open Issues. IEEE Access, 6, 18209-18237. doi:10.1109/access.2018.2820162
10. Salman, T., Zolanvari, M., Erbad, A., Jain, R., & Samaka, M. (2019). Security Services Using Blockchains: A State of the Art Survey. IEEE Communications Surveys & Tutorials, 21(1), 858-880. doi:10.1109/comst.2018.2863956
11. Sheren A. El-Booz, Gamal Attiya & Nawal El-Fishawy 2016, 'A secure cloud storage system combining time-based one-time password and automatic blocker protocol', EURASIP Journal on Information Security, vol. 13, pp. 1-13.
12. Suye Namasudra & Pinki Roy 2016, 'A new secure authentication scheme for cloud computing environment', Willey, Concurrency And Computation: Practice And Experience, pp. 1-20.
13. Shraddha Mokle & Nuzhat F Shaikh 2016, 'Anonymous Authentication for Secure Data Stored on Cloud with Decentralized Access Control', IEEE WiSPNET, pp. 216-220.
14. Mansura Habiba, Md. Rafiqul Islam, Shawkat Ali, ABM & Md. Zahidul Islam 2016, 'A New Approach to Access Control in Cloud', Arabian Journal for Science and Engineering, Springer, vol. 41, no.3, pp. 1015-1030.
15. Zeeshan Pervez, Mahmood Ahmad, Asad Masood Khattak, Sungyoung Lee &Tae Choong Chung 2016, 'Privacy-Aware Relevant Data Access with

Semantically Enriched Search Queries for Untrusted Cloud Storage Services', PLOS ONE, vol. 11, no.8, pp. 1-20.

16. Joseph, L.M., Thomson Fredrik, E.J. (2022). A Novel Hybrid Approach Based on Filters to Ensure Cloud Storage Data Security. In: Hu, YC., Tiwari, S., Trivedi, M.C., Mishra, K.K. (eds) Ambient Communications and Computer Systems. Lecture Notes in Networks and Systems, vol 356. Springer, Singapore.