Light-weight Accountable Privacy Preserving (LAPP) Protocol to Determine Dishonest Role of Third Party Auditor in Cloud Auditing

Mohamed Ben Haj Frej
University of Bridgeport
Computer Science and Engineering
Bridgeport, USA
mbenhaj@bridgeport.edu

Julius Dichter
University of Bridgeport
Computer Science and Engineering
Bridgeport, USA
dichter@bridgeport.edu

Navarun Gupta
University of Bridgeport
Electrical Engineering
Bridgeport, USA
navarung@bridgeport.edu

Abstract*— Cloud computing is surfacing as the next disruptive utility paradigm. It provides massive storage capabilities, the development environment for application developers through virtual machines. It is also the home of software and databases that are accessible, on-demand. As security is the main constraint holding companies to engage into the cloud fully, third-party auditors (TPA) are becoming more and more common in cloud computing implementations. Of course, involving auditors comes with its issues such as trust and processing overhead. To achieve productive auditing, we need to (1) accomplish efficient auditing without requesting the data location, nor introducing processing overhead to the cloud client; (2) avoid involving new security vulnerabilities during the auditing process. In this paper, we propose a novel method allowing to detect a dishonest TPA: the Light-weight Accountable Privacy Preserving (LAPP) Protocol. This protocol determines the malicious behavior of the TPA. To validate the proposed protocol's effectiveness, simulation experiments have been conducted, using the GreenCloud simulator. Based on our simulation results, we confirm that our proposed protocol provides better outcomes as compared to the other known contending protocols concerning reliability, communication cost, and the auditing time per task by a fraction of the invalid responses.

Keywords: public auditing, light-weight privacy preserving (LAPP), third party auditor (TPA), cloud client (CC), cloud service provider (CSP), security and privacy of storage (SPS), panda public auditing (PPA), privacy-preserving public auditing (PPPAS), secure and efficient privacy-preserving public auditing (SEPPPA).

I. INTRODUCTION

With the fast advancement of storage and processing technologies, computing resources have become more inexpensive, dominant, and universally available than ever before. This high-tech tendency has enabled the introduction of a new computing paradigm called cloud computing, in which resources (e.g., storage and Central processing unit) are delivered as additional utilities that can be bought and released by cloud service users in an on-demand manner through internet [1].

Cloud computing allows for the storage of data in either privately owned or third-party data centers. Cloud computing rises to be next IT undertaking engineering era, because of its extensive rundown of phenomenal preferences: on-request self-benefit, pervasive system get to, autonomous area, quick asset versatility, utilization based valuing and transference of hazard [2-3].

Cloud computing utilizes the Internet and central remote servers to store information and applications [4]. There are three main proposed models:

- Software as a Service (SaaS), allowing one to utilize the supplier's software running on a virtual cloud environment.
- Platform as a Service allowing one to install in cloud infrastructure user-generated or purchased software.
- And Infrastructures as a Service (IaaS) offering primary computing infrastructure Service, storage, and networking.

Appealing as it may be, challenges are still at stake in cloud computing because data recovery and coverage against threats need to be always upbeat. Accomplishing the accuracy of the information in a cloud domain, can come with its challenges and increased cost. Hence, for cost-saving and security, enabling public auditing for cloud computing is necessary, the result is the involvement of a Third-Party Auditor (TPA) to increase the end-user's confidence in deploying their IT resources in the cloud.

Moreover, As the rapid development of cloud computing, the foresee future of auditing tasks is increasingly important for various users.

Security worries in cloud computing are separated into seven noteworthy parts: access of favored client, where information found, information isolation, administrative consistency, analytical support, recuperation, and extensive haul suitability. These worries have, in some cases, imperiled the security of information in cloud servers or transmission and because of these dangers; many schemes were proposed to ensure protection in cloud computing [5].

II. PROBLEM DEFINITION

In the cloud computing environment, the TPA is considered to be a highly trusted party that can secure the privacy of the cloud users. On the hand, TPA can be a malicious insider to crack the confidential information of the user and sell it to other parties by taking the bribe or other benefits for the exchange of specific information. Thus, we need a protocol to determine the dishonest role of TPA when auditing the records of the cloud users. As, this detection process brings stability in the auditing process, but it can increase the extra communication overhead. Thus, the light-weight

978-1-5386-3025-9/18/$31.00 ©2018 IEEE

accountable privacy-preserving protocol is required not only to detect the malicious role of TPA but having minimum communication overhead.

III. Related Work

A. Security and Privacy for Storage (SPS)

"Secure Public Auditing Cloud Storage Enabling Data Dynamics in the Standard Model" [6], is a proposed protocol to audit and protect the data's integrity using the RSA assumption as a base, extending it to enable a TPA to audit the user's data without being able to learn its contents. It also enables to support the operations of data dynamics, such as "insertion," "modification," and "deletion." The model is composed of three entities: the user or Cloud Client (CC), the CSP, and the TPA. The user has a substantial amount of data to store, the CSP has the means to store all this data at a very economical price, and the TPA is proficient in providing auditing that is unbiased and efficient. It is assumed that the CSP is to be untrusted for the reasons discussed before. To gain trust in the CSP the TPA's services are needed. The TPA is to be trusted but it is also assumed that it could be curious and it could be dangerous if it could learn any of the sensitive information in the data outsourced. The protocol starts with the CC encrypting the data to be outsourced using as a base the strong RSA assumption.

From there the auditing starts, and it is composed of five algorithms:

- "KeyGen": used by the CC to create a public key and a secret one. "Outsource": used by the CC to send the processed data to the CSP. "Audit": used by the TPA to create an audit query to send to the CSP.
- "Prove": used by the CSP after receiving as input the audit query from the TPA and creates a proof by using the data stored.
- "Verify": used by the TPA after receiving the proof from the CSP to check, using the public key, if the proof is correct.

The performance analysis has been divided into two parts, the communication, and the computation cost. The communication cost consists of the communication between the CSP and the TPA. It is based on the proof provided by the CSP to the TPA.
The computation cost for the TPA is determined by the time it takes to audit and verifies the data, which is supposed to be fast. The cost for the CSP is determined by the time it takes to prove to possess the data, which is determined by the block size, how long the audit query is, and how long it takes to create the information to authenticate.

B. PANDA Public Auditing (PPA)

"Public Auditing for Secure Data Storage in Cloud through a Third Party Auditor Using Modern Ciphertext"[7] proposes a scheme of auditing, using cipher cryptography instead of encryption for the communication with the third party auditor. The scheme focuses on the data integrity and its storage.

In this method, the TPA performs audits without the need for copies of the outsourced data. The scheme consists of five algorithms and involves the same terminology as the other papers: CC, CSP, and TPA.

The CC has to use the new ciphertext to encrypt the data to be outsourced. Then the auditing process can begin, using five algorithms somewhat similar to those seen in the previous paper:

- "KeyGen," that is used by the CC and the TPA to generate keys.
- "SigGen," that the TPA runs to create the verification metadata.
- "GenProof," that is used by the CSP to check if the data is correctly stored and create a proof of its state; and "VerifyProof" that the TPA uses to test the evidence given by the CSP and verify its correctness.

During the setup phase, these algorithms are applied as follows, after the CC encrypts the data, it uses "KeyGen" to generate an owner key, and then sends the key and the processed data to the TPA through a private channel. The TPA runs "KeyGen" to create a challenge key and then runs "SigGen" to generate the verification key and then encrypts the processed data to generate crypto-metadata that is then sent to the CSP.

In the auditing phase, the TPA sends a challenge, using the challenge key, to the CSP. Then using "GenProof" generates an audit key that sends to the TPA, which then uses "VerifyProof" to check if the audit key is equal to the verification key allowing to verify the integrity of the stored data [7].

The system performance analysis is divided into computational, communication, and storage costs. The aim for the computational is to achieve low complexity. Compared with other models with the same scheme it uses the advanced encryption standard on a bilinear map. In the communication cost, the goal is to have the length of the auditing requests shorter than the length of the requests in the other schemes based on bilinear maps. Finally, in the storage cost, it is also compared to schemes based on bilinear maps [7].

C. Privacy-Preserving Public Auditing(PPPAS)

"Privacy-Preserving Public Auditing for Secure Cloud Storage" (PPPAS)" [2], is the oldest one of the batch and is alleging to be one of the pioneers to implement public auditing that preserves the data's privacy in the cloud using the "homomorphic linear authenticator" (HLA) [8]. The HLA is based on keys. This technique allows to audit without having to use a local copy of the data and incorporating it with an arbitrary masking.

This method aims to make the TPA unable to learn the audited data's contents. This scheme uses the same four algorithms the last scheme used as well ("KeyGen," "SigGen," "GenProof," and "VerifyProof"). The performance of their auditing is shown to be on par to the state-of-the-art, with a warranty of privacy-preserving [9].

D. Secure and Efficient Privacy-Preserving Public Auditing (SEPPPA) Protocol

"Secure and efficient privacy-preserving public auditing scheme for cloud storage"(SEPPPA) [10] declares to have an auditing scheme that has the TPA audit without needing the entire data, maintaining its privacy and integrity, as well as being able to audit by batches. It uses a bilinear map to encrypt the data [4]. This scheme uses four algorithms as well: "KeyGen" that the CC uses to create a pair of keys, one public, available to all the auditing participants, but only authorized TPA's can use it to audit, and a private key for the CC. "SigGen" that creates signatures for all the outsourced files. "ProofGen" is run by the CSP when challenged and uses the file to create a proof of its integrity. "VerifyProof" is used to check the integrity of the data using the proof provided by the CSP and the public key, and is run by the TPA [11]. The performance of the scheme was categorized in communication and computation overhead. In the former, it is explained that data outsourcing, challenge-response auditing, and the data retrieval, are the main reasons for complexity in the message exchange [5]. It is considered that the outsourcing and retrieval overhead is inevitable, so they focus on the challenge-response, to which they arrive at the conclusion that the system's complexity is constant [12].

IV. Proposed Solution

Our solution relies on the exchange of keys between the main stakeholders to determine malicious activities; it entails three phases:

1) Key validation process to avoid TPA's malicious role

The goal of this phase is to validate the key presented by the TPA when initiating the audit process. The CC starts the testing process then computes the CSP's response against the group of the secret keys and random numbers; a group of the random number including secret keys must be matched with the list of CC's random numbers. This testing process is encrypted by using secret keys inclusive random numbers. The CC checks if auditing testing process meets the criteria of the random number and the secret keys issued by the CSP are same as the TPA's. If the criteria are not met, then the cloud service user determines the existence of a TPA's malicious activity.

2) Key-Extraction process of the three stakeholders

A trusted party issues a group of secret random shared keys to the three stakeholders. The goal of this phase is to ensure that the three stakeholders (CC, CSP, and TPA) are using the same keys as issued by the trusted party. The CSP issues the blinded key to the TPA. The TPA examines the group of blinded-key if they are the same as the one already available with the CSP. If the group of secret keys including random shared keys is similar with that of blinded-key, then it is proved that both entities (CSP and TPA) are possessing similar keys for authentication. The TPA also checks the blinded-key with CC. Hence, the CC computes the key, if the key of the CC matches the key of the TPA, then it is declared that both parties have the same keys.

3) Detecting the malicious activities of the TPA and the CC

The CSP assigns the encrypted key to the TPA and the CC. As, the TPA uses this key to audit the encrypted contents of the CC, while the CC uses the key to access the cloud and detect the possible malicious activity of the TPA when auditing the data contents. The TPA checks the services provided to the CC from the CSP. The CC presents the encrypted data contents to check and determine the role of the CSP, if the data contents presented by the CC don't match with the provided service from the CSP, then the CSP is considered as malicious (dishonest).

V. ALGORITHMS

Our proposed solution is based on three main algorithms that will be detailed below:

- The Key Validation process to avoid the malicious role of TPA.
- The key-extraction process of three stakeholders.
- Detecting the malicious activity of the third-party auditor and the cloud service provider.

***Algorithm 1*:** Key Validation process to avoid the malicious role of TPA

1. Initialization: (g^k: Guarantee of the key; s_k: Secret key; A_{test}: Auditor testing process; A_{res}: Auditor response; SP_{res}: Service provider response; CC: Cloud Client; p: prime number, R_n: Random number, and V: validation) Input: D_c
2. Input: (g^k; s_k; A_{test}; R_n; p)
3. Output: (A_{res} ; SP_{res}; V)
4. Pick R_n such that $1 < R_n < p$
5. Compute g^{Rn}// Group of random numbers
6. CC initiates A_{test}
7. Set $A_{test} = g^{Rn}$
8. CC computes $SP_{res} = g^{Rn.K}$
9. $SP_{res} \rightarrow$ CC: $g^{Rn.K}$
10. CC computes $A_{test} = (g^k)^{Rn}$
11. CC checks if $A_{test} = g^{Rn.K}$ then
12. CC confirms successful V
13. else if CC determines the malicious activity of TPA
14. End if
15. End else if

The goal of this algorithm is to validate the key presented by the TPA when initiating the audit process:

- From lines 1-3, initialization, input and output processes are explained respectively.
- In line-4, a random number is picked that should be greater than 1 and less than a prime number.
- In line-5, the group of random numbers is computed.
- From lines 6-7, Cloud client starts testing process for auditor and group of random numbers are set for the testing process.
- In line-8, the cloud client computes service provider's response against a group of the secret keys and random numbers; a group of a random number including secret keys must be matched with the list of cloud client's random numbers.
- In line-9, cloud service provider sends its response to the cloud client to confirm the group of secret keys including random numbers.
- In line 10, cloud client computes testing process that should be encrypted by using secret keys inclusive random numbers.
- In lines 11-13, the cloud client checks if auditing testing process meets the criteria of a random number and secret keys issued by the cloud service provider are same as third party auditor (TPA) presents for auditing then, cloud client confirms the successful key validation process. If the criteria are not met, then the cloud client determines the existence of a TPA's malicious activity.

Algorithm 2: Key-Extraction process of three stakeholders

1. Initialization: (g^{Rs} : Group of Random shared key; CSP: Cloud service provider; S_k: Secret key; B_k: Blinded key; CC: Cloud client; p: prime number; R_s: Random shared key; TPA: Third Party Auditor; and T: trusted party)
2. Input: (g^{Rs}; CSP; S_k; T)
3. Output: (T_{pa}; B_k)
4. CC &CSP use R_s
5. Set $1<R_s<p$ && Tpa knows g^{Rs} && $g^{Rs} \epsilon$ T
6. CSP →TPA : $B_k = S_k + R_s \bmod p$
7. TPA examines g^{Bk} if $g^{Sk}\, g^{Rs} = g^{Sk+Rs} \bmod p$ then
8. Set $g^{Bk} = g^{Sk+Rs}$
9. Elseif $g^{Sk}\, g^{Rs} \neq g^{Sk+Rs}$ then
10. Set CSP ∄ g^{Bk}
11. Endif
12. End else-if
13. TPA → CS ϵ B_k
14. CC computes $B_k - R_s = S_k \bmod p$ if CC = B_k then
15. Set TPA = CC
16. endif

The goal of this algorithm is to ensure that the three stakeholders (Cloud Client, Cloud Service Provider, and Third-Party Auditor) are using the same keys as issued by trusted party:

- From lines 1-3, used parameter-initialization, input, and output processes are explained respectively.
- In line-4, cloud service provider and cloud client use the secret random shared key.
- In the line-5, the value of the secret random shared key is set more than 1 and less than a prime number. Furthermore, trusted party is responsible for issuing the group of random shared keys for the three stakeholders.
- In line-6, cloud service provider issues the blinded key to the third-party auditor.
- In lines-7-10, third-party auditor examines the group of blinded-key if they are same that are already available with cloud service provider. If the group of secret keys including random shared keys is similar with that of blinded-key, then it is proved that both entities (cloud service provider and third-party auditor) are possessing similar keys for authentication. If a group of secret keys including random shared keys is not similar with that of blinded-key, then it is proved that cloud service provider does not have valid keys that should match with a group of blinded keys.
- In lines-11-13, Third party auditor also checks the blinded-key with cloud client. Hence, the cloud client computes the key, if the key of cloud client matches the key of the third-party auditor, then it is declared that both parties have same keys.

Algorithm 3: Detecting the malicious activity of third-party auditor and Cloud service provider

1. Initialization: (***f(h)***: Hash function; CSP: Cloud service provider; K_{en}: Encrypted key; Dc: Data contents; CC: Cloud client; TPA: Third party auditor; $O(D_s)$: Original data service; Me: Malicious)
2. Input: (D_c)
3. Output: (Me)
4. CSP assigns K_{en} to TPA & CC
5. TPA audits {***f(h)*** K_{en} (D_c)} ϵ CC
6. If {***f(h)*** K_{en} (D_c)} ϵ CC = $O(D_s)$ then
7. CSP ϵ h otherwise Me
8. Endif
9. CC → TPA: {***f(h)*** K_{en} (D_c)}
10. If{***f(h)*** K_{en} (D_c)} = $O(D_s)$ then
11. TPA ϵ h otherwise Me
12. Endif

This algorithm aims to detect the malicious activity of third-party auditor when auditing the data contents of cloud client:

- From lines 1-3, parameter-initialization, input and output processes are described respectively.
- In line-4, the cloud service provider assigns the encrypted key to the third-party auditor and cloud client. As, the third-party auditor uses this key to audit the encrypted contents of cloud data user, while the cloud client uses the key to access the cloud and detect the malicious activity of third-party auditor when auditing the data contents.
- In lines-5-7, the third-party auditor checks the services provided to cloud client from the cloud service provider. The cloud client presents the encrypted data contents to check and determine the role of the cloud service provider, if
- The data contents presented by the cloud client matches with provided service from the cloud service provider, then the cloud service provider is considered as honest otherwise the cloud service provider is considered as malicious (dishonest).
- From lines 9-11, the role of the third party auditor has checked whether it is honest or doing malicious activities with the data contents.

VI. SIMULATION

To confirm the performance of the proposed Light-weight Accountable Privacy Preserving (LAPP) protocol, we have developed the LAPP using C++ programming language and integrated into the GreenCloud simulator. The GreenCloud is run on the IBM z13 to obtain quick and realistic outcomes. We generated several scenarios that were almost identical to the real environment. The used parameters are described in Table 1 and are for testing purposes.

TABLE 1: Simulation Parameters

Parameters	Details
Number of chassis switches at L4	1920
Line cards at L4	1630
Ports at L4	72
Number of racks at L4	16
Number of chassis switches at L3	432
Line cards at L3	164
Ports at L3	48
Number of racks at L3	128
Used virtual machines	1800
Number of Servers	64
Maximum number of Cloud Service Users	18000
Hosts in each rack	132
Each Host supports	16 processors
Memory with each processor	256 GB
Storage Memory	512 GB
Virtual Disk Memory	430 GB
Bandwidth for L4	256 GB/Sec
Bandwidth for L3	128 GB/Sec
Bandwidth for L2	64 GB/Sec
Bandwidth for L1	16 GB/Sec
Queue delay	0.005 Seconds
Burst time	0.0056 Seconds
Idle time	0.0032 Seconds
Packet Size	1260 KB

A. Simulation Results

All simulations have been done with malicious attempts rates at 0%, 1%, 2% and 5% malicious TPA activities. All simulation results have shown superiority in performance for our proposed method (LAPP). We only include the graphs of malicious attempts at 2% to abide by the conference's paper length requirements.

In our simulation, we have proceeded with the following measurements:

- Reliable Auditing Detection (Number of Cloud Auditing Users)
- Communication Cost (Block Size)
- Auditing Time per Task (Fraction of Invalid Responses)

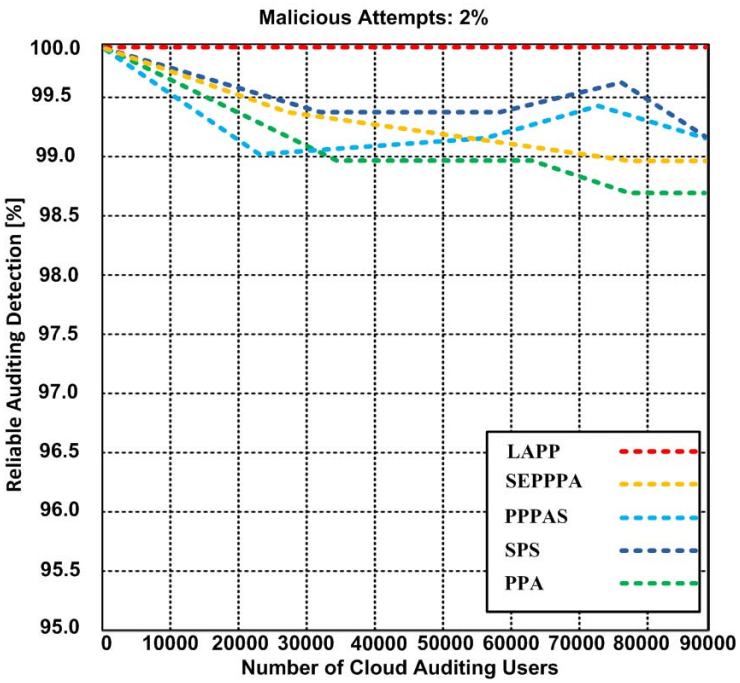

Figure 1: Reliable Auditing Detection (Number of Cloud Auditing Users)

We measure the percentage of reliable auditing detection with some cloud auditing users. The measurements have been taken by increasing the number of cloud auditing users by 10,000 each time. At a malicious rate of 2%. LAPP is showing around 100% reliability measurement while showing a fluctuation of the other methods we have compared our results to, between 98.6% and 99.6%.

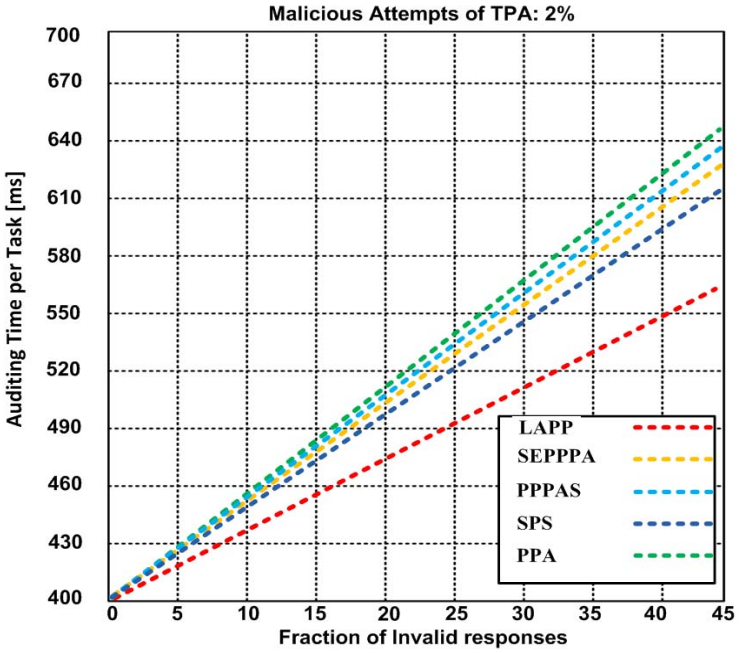

Figure 2: Communication Cost (Block Size)

In this simulation, we measure the auditing time per task in milliseconds in comparison with the fraction of invalid responses; increasing by five on each occurrence. At the 45th fraction of invalid responses, the simulation showed the result of 560 ms for LAPP, while all other methods show a fluctuation between 590 ms and 610 ms.

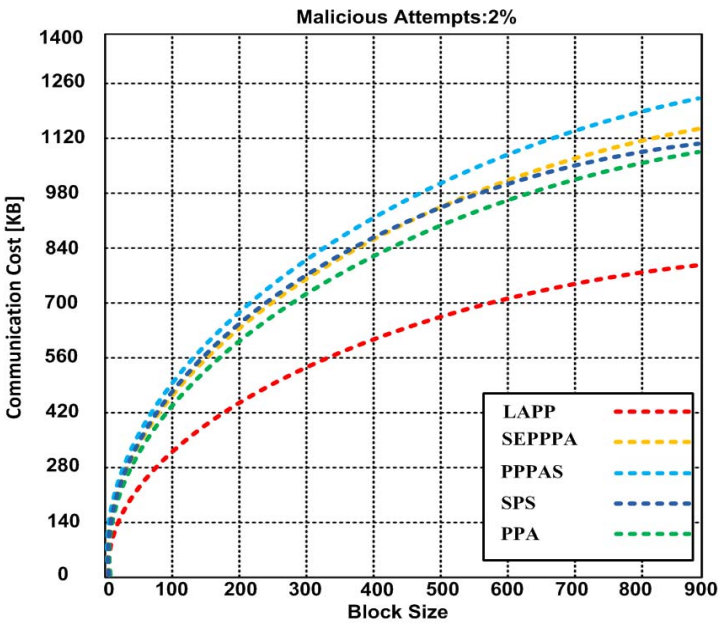

Figure 3: Auditing Time per Task (Fraction of Invalid Responses)

In this simulation, we measure the communication cost in KB while increasing the block size from 0 to 900. At the block size of 900, the simulation result shows 830 KB while other methods show a fluctuation between 1100 KB and 1250 KB.

VII. DISCUSSION

Our novel model consists on a secure validation between the tree stakeholders (CSP, CC, and TPA) allowing to build a lightweight privacy-preserving protocol allowing on the spot detection of a dishonest role of the TPA while auditing the data.

In our first simulation, we have measured the percentage of the reliable auditing detection for up to 90000 cloud auditing users, incrementing the number of users by 10000 for each measurement. The simulation results showed a neat reliability close to 100% for our proposed method (LAPP).

In our second simulation, we have measured the auditing time per task compared to the fraction of invalid responses, and the results showed a shorter average of 50 ms to 80 ms depending on the methods we have compared our results to.

In the last simulation, we have measured the communication cost depending on the block size up to 900 KB, increasing block size by a 100 each time. Our results showed a noticeable lower computation cost compared to the other methods we have compared our results to.

These results show the efficiency of our proposed method (LAPP) as well as the optimization in reducing the processing and communication overheads.

VIII. CONCLUSION

In this paper, Light-weight Accountable Privacy Preserving protocol is introduced for the sake of securing cloud computing. As third-party auditing is a common practice in information technology for large companies; nowadays, many cloud computing solutions encompass a TPA in their solution to engage skeptical clients to engage with their appealing technology fully.

Nonetheless, a TPA implementation comes with its disadvantages, mainly overhead and trust concerns. This paper aims to assure a smooth "auditing of the auditor" process. To accomplish this goal, we have introduced a novel method: Light-weight

Accountable Privacy Preserving (LAPP) Protocol relying on the following phases:

- The Key Validation process to avoid the malicious role of TPA.
- The key-extraction process of three stakeholders.
- Detecting the malicious activity of third-party auditor and cloud service provider.

The simulation result with the introduction of 0%, 1%, 2% and 5% malicious TPA activities for the following measurements:

- Reliable Auditing Detection (Number of Cloud Auditing Users)
- Communication Cost (Block Size)
- Auditing Time per Task (Fraction of Invalid Responses)

All simulation results have shown noticeable superiority of our introduced method (LAPP) compared to the other methods we have compared our results to. Allowing us to conclude that our method allows determining, in an optimal fashion, any dishonest role of the TPA.

In our future work, we are planning to measure the computation time on auditing, the average auditing time for each user, the accuracy, and the time complexity.

ACKNOWLEDGMENTS

We are so grateful to Dr. Abdul Razaque, from New York Institute of Technology - Nanjing Campus - China, for reviewing the paper and providing valuable recommendations. Special thanks to Vignesh Mandalapa Bhoopathy and Hai Van Nguyen, graduate students at the Electrical Engineering Department at the University of Bridgeport, for their help in compiling the information together.

IX. REFERENCES

1. Razaque, Abdul, and Syed S. Rizvi. "Privacy-preserving model: a new scheme for auditing cloud stakeholders, " Journal of Cloud Computing 6, no. 1 (2017): 7.
2. Wang, Cong, Sherman SM Chow, Qian Wang, Kui Ren, and Wenjing Lou. "Privacy-preserving public auditing for secure cloud storage." *IEEE transactions on computers* 62, no. 2 (2013): 362-375.
3. Razaque, Abdul, and Syed S. Rizvi. "Triangular data privacy-preserving model for authenticating all key stakeholders in a cloud environment." *Computers & Security* 62 (2016): 328-347.
4. Rizvi, Syed, Abdul Razaque, and Katie Cover. "Third-Party Auditor (TPA): A Potential Solution for Securing a Cloud Environment." In *Cyber Security and Cloud Computing (CSCloud), 2015 IEEE 2nd International Conference on*, pp. 31-36. IEEE, 2015.
5. Moghaddam, Faraz Fatemi, Omidreza Karimi, and Maen T. Alrashdan. "A comparative study of applying real-time encryption in cloud computing environments." In *Cloud Networking (CloudNet), 2013 IEEE 2nd International Conference on*, pp. 185-189. IEEE, 2013.
6. Wei, Lifei, Haojin Zhu, Zhenfu Cao, Xiaolei Dong, Weiwei Jia, Yunlu Chen, and Athanasios V. Vasilakos. "Security and privacy for storage and computation in cloud computing." *Information Sciences* 258 (2014): 371-386.
7. Hussien, Z.A., et al. *Public auditing for secure data storage in cloud through a third party auditor using modern ciphertext*. in *Information Assurance and Security (IAS), 2015 11th International Conference on*. 2015. IEEE.
8. Wang, Boyang, Baochun Li, and Hui Li. "Panda: Public auditing for shared data with efficient user revocation in the cloud." *IEEE Transactions on services computing* 8, no. 1 (2015): 92-106.
9. Rizvi, Syed, Abdul Razaque, and Katie Cover. "Third-Party Auditor (TPA): A Potential Solution for Securing a Cloud Environment." In *Cyber Security and Cloud Computing (CSCloud), 2015 IEEE 2nd International Conference on*, pp. 31-36. IEEE, 2015.
10. Worku, Solomon Guadie, Chunxiang Xu, Jining Zhao, and Xiaohu He. "Secure and efficient privacy-preserving public auditing scheme for cloud storage." *Computers & Electrical Engineering* 40, no. 5 (2014): 1703-1713.
11. Yang, Kan, and Xiaohua Jia. "An efficient and secure dynamic auditing protocol for data storage in cloud computing." *IEEE transactions on parallel and distributed systems* 24, no. 9 (2013): 1717-1726.
12. Lee, W-B., and C-C. Chang. "Efficient group signature scheme based on the discrete logarithm." *IEE Proceedings-Computers and Digital Techniques* 145, no. 1 (1998): 15-18.