A Survey on Security and Privacy Ensuring Techniques for Storage and Computation in Cloud Computing

# Aneesha K Jose $ Roshni Thanka
#P.G Student, Department of Computer Science and Engineering,
$Assistant Professor, Department of Computer Science and Engineering,

Karunya University

Abstract

Cloud computing is defined as a computing model for enabling easy, on-demand access to a collection of configurable computing resources such as server storage, applications and other internet services. They can be easily allocated with minimal management effort or service provider interaction. It is a new style of computing in which the resources are allocated dynamically as a service over internet. Data and computations on the data can be moved to the huge data centers, called cloud. A cloud user has to pay only for the resources he actually uses. In traditional computing users have full control of data storage and computation whereas in cloud computing the managements of data and performance of computations are entrusted to the cloud service providers who perform this with the help of many cloud servers. So the privacy and security of stored data is sometimes violated at the cloud servers. Also the cloud server may not have performed a computation which was declared by it as performed. The cloud server thus may conduct such cheatings for gaining profit. Various measures has been proposed so far to ensure the storage and computation correctness at cloud servers. This survey analyzes the various methodologies used so far to ensure the security and privacy of stored data and computation at cloud servers and to prevent cheating by insincere cloud servers. This survey also reports the merits and limitations of each scheme.

1. Introduction

Cloud computing is a computing paradigm that offers customers a more flexible way to obtain computation and storage resources on demand. Customers can now rent the necessary resources only when they need. Thus the customers can avoid a large initial investment. Security and privacy are the major challenges which inhibit the growth of cloud computing. Cloud computing security can be classified into two major categories such as Cloud Storage Security and Cloud Computation Security. Cloud Storage Security ensures that the outsourced data stored at unreliable cloud servers is not subjected to modification. Cloud Computation Security ensures that the outsourced computation performed by external cloud servers returns correct result. It is also important to maintain privacy of stored data at Cloud Server (CS) controlled by Cloud Service Provider (CSP). By privacy a customer means that the confidential data he stored at cloud server should not be revealed to a third party. So many techniques have been proposed so far to ensure the security and privacy of stored data and computation at the cloud servers. Some of these techniques are studied and analyzed in this survey.

2. Existing Solutions

Safety of the data stored in the cloud servers has been compromised in many cases for monetary profit. Also it may be modified or lost due to some security violation or some human error. Computation error occurs when the cloud servers do not perform the computations which were actually declared by it as performed. This is done in order to save the computation resources and reap profit for undone task. Therefore it is essential to maintain the security and privacy of stored data and computation at cloud servers and therefore various methodologies were introduced.

2.1 Byzantine Fault Tolerant Algorithm (BFT)

BFT algorithm in [14] ensures security of storage and computation at cloud servers. It is used in systems where there is no limitation of time. It can be used to replicate existing data. It is used to develop systems that do not fail under byzantine faults. If a fault is met it redoes computation till the desired output is met. It will recover copies of stored data at regular intervals even if they are not found modified. Thus it ensures storage and computation correctness. Thus BFT satisfies the real world requirements such as web browsing. This algorithm cannot perform well if more than 1/3 of the total replicas become faulty.

2.2 Remote data integrity checking protocol

This protocol in [26] ensures security for stored data at cloud server. The client sends challenge request to the server about the integrity of certain file and server generates responses proving that data in that file is not corrupted. The client does not need to access the complete file to prove integrity. Also client should be able to verify integrity of stored data for unlimited number of times. The remote data integrity checking protocol uses Homomorphic Verifiable Tags (HVT). A HVT of a message m checks whether the stored data is subject to modification or not. HVT of a message m can be denoted by D. By using HVTs, the server can obtain a proof of possession for a set of file blocks that those file blocks were not modified at cloud server. Thus the client needs not have to access the contents of files to ensure storage correctness.

2.3 Cooperative Provable Data Possession (CPDP)

Cooperative Provable Data Possession (CPDP) method is used to ensure storage correctness in hybrid clouds as specified in [25]. It is an enhancement of Provable Data Possession (PDP). It is used when multiple Cloud Service Providers (CSPs) exist to store and manage client's data. It uses homomorphic property. Client sends challenge request to server for proving integrity of stored data. Client will receive response for this challenge request from different CSPs. It can be combined into a single response as a final result of hybrid clouds. A new hash index is created to store and manage resources in hybrid clouds without limitations. Thus clients are ensured that stored file is secure without knowing in which machine or geographical location the file resides.

2.4 Digital signature and RSA algorithm

This method is used to protect the privacy and integrity of stored data in cloud environment according to [9]. Digital signature checks the proof of sender or signer of the file ensuring that the contents are unchanged. This method uses RSA algorithm for encryption and decryption. Digital signature is used for message authentication. It has three participators- Cloud User, Cloud Service Provider and Third Party Auditor. Public keys are shared among Cloud User and Third Party Auditor. First data is signed with user's private key and cipher is again encrypted using public key of Third Party Auditor. This data is now send to cloud and Third Party Auditor. Third Party Auditor now decrypts the encrypted message with his private key and de-signs cipher with user's public key. The same process of decryption is carried out in the cloud by the Third Party Auditor to verify the correctness by comparing the data which he has with the stored one. Third Party Auditor indicates the result to user.

2.5 Distributed storage integrity auditing

This method allows the users to audit the cloud storage with less communication and computation cost as in [5]. It uses homomorphic token and distributed erasure coded data. It ensures storage correctness and fast locating of errors. The faulty or fraud server can be identified easily. Initially the files of users are distributed across different servers. Then tokens are computed that cover a set of blocks and store those tokens at cloud user. When user wants to make sure of storage correctness he gives a set of block indices to cloud server. On receiving the challenge each cloud server computes a short signature over the specified blocks and returns them to user. The value of these signatures should match corresponding tokens previously computed by user. If they do not match it indicates that integrity of data stored at server is disturbed. Once error is detected, user asks server to send these erroneous blocks and corrects them. This creates an extra burden for the cloud user.

2.6. Robust Data Possession (RDP)

This method in [21] integrates Forward Error Correcting codes (FEC) into Provable Data Possession (PDP). A file is first encoded using an

FEC code to form an encoded file. Then PDP is applied on the encoded file instead of original file. Thus it has two benefits. Firstly it prevents modification of a large portion of file i.e. when there is a change in the blocks. This happens when the Cloud Server sells same storage space to multiple clients. Secondly it prevents modification of a small portion of file i.e. changes within the block are also detected since FEC code is used.

2.7 Homomorphic authenticator with random masking

Homomorphic authenticator with random masking in [7] is used to audit stored data by a third party auditor without revealing the file content to them. Homomorphic authenticators are metadata for verification which are generated from individual data blocks. They can be combined to ensure the auditor that a linear combination of data blocks can be verified by verifying only the authenticator. To preserve privacy of stored data linear combination of blocks in the response of server is covered with a random number generated by a Pseudo Random Function (PRF). With random masking, the auditor cannot derive the user's data content and privacy is preserved.

2.8 Provable Data Possession (PDP)

This method in [6] helps a client who has stored data at a deceitful server to verify whether the server has maintained the original data without retrieving it. The server has to generate a proof of possession. For this the client has to maintain some amount of metadata to verify the proof. Client sends challenge request to server for proving storage correctness. Server has to give response in the form of a proof. It uses scheme called homomorphic verifiable tags. Client must have precomputed tags for each block of a file. The file and its tag are stored at cloud server. Client sends challenge request against a random set of blocks. Using blocks and their tags server generates proof. Client is convinced of data possession without actually having to retrieve file blocks.

2.9 Verifiable computation

In Verifiable computation [20] a client is not just depending on a single cloud. A client will perform same computation on two or more different clouds to see if the computation result obtained is correct or not. Initially the client picks N cloud providers. Instead of executing a computation on one CSP, client will perform it in N different CSPs. Client asks each of these CSPs to return the result they obtained. From the results returned client will take the majority of results as his answer.

2.10 Privacy manager

Privacy manager in [22] prevents private data of user getting revealed or misused. Privacy manager has a feature called obfuscation. In this method the user's data is sent to cloud in an obfuscated format by the privacy manager in client and processing is done on encrypted data at cloud server. After processing the result is send to the privacy manager in the client. The privacy manager deobfuscates the result to obtain correct result. This obfuscation is done by using a key shared among the client and the privacy manager. Thus it prevents CSP from revealing user's data to others.

2.11 Proofs Of Retrievability (POR)

POR in [1] helps a client to receive a proof from the server that its data is not deleted or modified at the server. POR encrypts file and randomly attaches a set of values called sentinels. Sentinels is different from the original file block. To verify correctness of stored data user or verifier gives positions of sentinels to cloud servers' and asks to return the sentinel values. If the cloud server has modified or deleted a large portion of file, it cannot return the correct sentinel value.

2.12 Incentivized computation

This method in [8] prevents cheating of computation at cloud servers. This method assigns an extra reward for the servers who complete the computations sincerely. There is a central authority called boss who assigns computational tasks to servers called contractors. The boss will reward a contractor for correctly completing a job. If contractor returned incorrect result boss will fine him. The fine is deducted from contractor's account. The boss will assign job to a contractor only if he has the minimum balance to pay the fine. To check whether the result returned is correct or not boss uses two methods. Either the boss can double check every result or appoint multiple contractors to do same job.

2.13 LP computators

This method mentioned in [3] focuses on securely outsourcing Linear Programming (LP) computations. LP computations are distributed to public LP solvers running in cloud. In order to do an LP computation problem at cloud, first the problem is encrypted using a secret key. Then the encrypted problem is given to cloud server. Cloud Service Provider gives result for encrypted problem back to cloud user. Cloud User verifies the result of encrypted problem. If it is correct, Cloud User uses secret key to map result to desired answer of the original problem.

Table 1: Comparison of methodologies

Methods	Merits	Demerits
Byzantine fault tolerant algorithm (BFT)	(1)System will be able to survive byzantine faults (2)Ensures storage and computation correctness at cloud server	(1)The algorithm will work only if maximum number of faulty processors is (n-1)/3 where n is the total number of servers
Remote data integrity checking protocol	(1)Supports public verifiability (2)Allows update on stored data	(1)Only storage correctness is ensured
Cooperative Provable Data Possession (CPDP)	(1)Ensure storage correctness in hybrid clouds	(1)Does not ensure computation correctness (2)Managing of hash index hierarchy is difficult
Digital signature method and RSA algorithm	(1)Ensures authentication and security of stored data (2)Simple to implement	(1)Public keys has to be shared without fail (2) TPA can know content of original file
Distributed storage integrity auditing mechanism	(1)Checks storage correctness (2)Locates and corrects error	(1)If no. of misbehaving servers is too much more redundancy and time needed to recover (2)Does not check computation correctness
Robust Data Possession (RDP)	(1) Corruption of large portions as well as error within the block also detected	(1)More time needed to check (2)More redundancy (3)Only storage correctness is verified
Public And Constant Cost Storage Integrity Auditing Scheme With Secure Deduplication (PCAD)	(1)Securely deduplicates the authentication tags (2)Saves time in uploading already existing file (3) storage overhead independent to the number of owners of the file (4)performs public auditing	(1)Does not handle computation security
Commitment based sampling scheme	(1)Ensures computation security	(1)Communication cost increases as users has to send commitment for every computation
Data anonymization	(1)Simple to implement (2)Anonymous data can be stored and processed without concern that others may capture the data	(1)Checks only storage correctness (2)Translation table must be kept secure
Homomorphic authenticator with random masking	(1)Ensures storage security	(1)Computational overhead since authenticators has to be again masked with a random number
Provable Data	(1)Storage correctness can be verified	(1)Both files and tags has to stored at

Possession (PDP)	(2)Verification done without actually accessing the whole file. Thus it reduces communication overhead.	server. It increases storage overhead at server
Verifiable computation	(1)Easy method to verify the computations	(1)More cost is incurred (2)This method fails if majority of CSPs are dishonest
Privacy manager	(1)Ensures privacy of stored data	(1)All cloud applications cannot work on obfuscated data (2)In order to keep privacy manager at client side , user must have enough resources for obfuscation and de-obfuscation
Proofs Of Retrievability (POR)	(1)Ensures storage correctness	(1)More storage overhead at server side since both data and sentinels has to be stored (2)Due to limited number of sentinels, verification can take place only limited number of times (3)Applied only to static data
Incentivized computation	(1)Ensures computation security	1)This method fails if all servers are dishonest
Designated verifier signature (DVS)	(1)Ensures privacy of stored data	(1) Suffers from delegatability attack where the signer was able to delegate his signing ability, to a third party
Lp computators	(1)Ensures computation correctness	(1) Communication overhead since every input has to go from CU. (2)Does not ensure storage correctness (3) User has to take up the extra burden of checking correctness of results.

3. Conclusion

Around twenty papers were surveyed for finding out the existing solutions for maintaining security and privacy for stored data and computations performed at cloud server. Every technique found solution for a particular issue. Every technique contain its own merits and demerits over the other analyzed methods. Researches are going on to find out a single solution that could overcome all the issues related to security and privacy of stored data and computation in cloud.

4. References

[1] A. Juels, B. Kaliski , "PORs: proofs of retrievability for large files" (2007)

[2] Boyd, Kang, E. Dawson, "A novel identity-based strong designated verifier signature scheme" (2009)

[3] Cong Wang, Kui Ren, J. Wang, "Secure and practical outsourcing of linear programming in cloud computing" (2011)

[4] Cong Wang, Kui Ren, Wenjing Lou, Jin Li, "Toward Publicly Auditable Secure Cloud Data Storage Services" (2010)

[5] Cong Wang, Qian Wang, Kui Ren, Ning Cao, Wenjing Lou, "Towards Secure and Dependable Storage Services in Cloud Computing" (2009)

[6] Mancini, Ateniese, Di Pietro, G. Tsudik, "Scalable and efficient provable data possession" (2008)

[7] Jachak, Korde, Ghorpade P.P, Gagare G.J "Homomorphic authentication with random masking technique ensuring"(2012)

[8] Mira Belenkiy, Melissa Chase,John Jannotti, Alptekin Kupcu ,Anna Lysyanskaya
"Incentivizing Outsourced Computation" (2008)

[9] K.Govinda, V.Gurunathaprasad , H.Sathishkumar, "third party auditing for secure data storage in cloud through digital signature using rsa" (2012)

[10] K.RaviTeja, Srinivasa Narasanna Pilli, B.Sreenivasa Rao, M.JangaReddy, "Secure Storage in Cloud Computing & Emergence of Intruder Detection" (2012)

[11] L. Wei, H. Zhu, Z. Cao, W. Jia, A. Vasilakos, "Seccloud: bridging secure storage and computation in cloud" (2010)

[12] A. Fox, Armbrust, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, “A view of cloud computing” (2010)
[13] M. Belenkiy, M. Chase, C. Erway, J. Jannotti, A. Küpçü, A. Lysyanskaya, “Incentivizing outsourced computation” (2008)
[14] Liskov, Castro, “Practical byzantine fault tolerance and proactive recovery” (2002)
[15] Sako, Jakobsson, Impagliazzo, “Designated verifier proofs and their applications” (1996)
[16] Ning Cao, Shucheng Yu, Zhenyu Yang, Wenjing Lou, Y. Thomas Hou “LT Codes-based Secure and Reliable Cloud Storage Service” (2012)
[17] P. Golle, I. Mironov, “Uncheatable distributed computations” (2001)
[18] Q. Wang, C. Wang, J. Li, K. Ren, W. Lou, “Enabling public verifiability and data dynamics for storage security in cloud computing” (2009)
[19] Qin Liua, Guojun Wanga, Jie Wub, “Secure and privacy preserving keyword searching for cloud storage services” (2011)
[20] R. Canetti, B. Riva, G. Rothblum, “Verifiable computation with two or more clouds” (2011)
[21] Osama Khan, Reza Curtmola, Randal Burns, “Robust Remote Data Checking” (2008)
[22] Y. Shen, S. Pearson, M. Mowbray, “A privacy manager for cloud computing” (2009)
[23] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou, “Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing” (2010)
[24] Du, Jia, M. Mangal, M. Murugesan, “Uncheatable grid computing” (2004)
[25] Huaixi Wang, Yan Zhu, Zexing Hu, Gail-Joon Ahn, Hongxin Hu, Stephen S. Yau, “Efficient Provable Data Possession for Hybrid Clouds” (2010)
[26], Sheng Zhong, Zhuo Hao , Nenghai Yu, “ Privacy-Preserving Remote Data Integrity Checking Protocol with Data Dynamics and Public Verifiability” (2011)