International Conference on Recent Trends in Engineering & Science
Shree Ramchandra College of Engineering, Pune (India) ICRTES - 16
29th-30th September 2016, www.conferenceworld.in ISBN : 978-93-86171-06-1

REVIEW ON DATA POSSESSION AND SECURITY IN COLLUDING SERVERS

Amudha T[1], Sumithra Devi K A[2], K Saravanan[3]

[1]Research Scholar, [3]Dean, Department of CS, *Prist University, Thanjavur*

[2]Principal, *GSSSITW-Mysore*

ABSTRACT

Imperceptibly more and more organizations are opting for outsourcing data to remote cloud service providers (CSPs). Clients can rent the CSPs storage infrastructure to store and get back almost infinite amount of data by paying amount per month. On behalf of an improved level of scalability, availability, and durability, some clients may want their data to be virtual on multiple servers across multiple data centers. The additional metaphors the CSPs is stated to store, the more amounts the clients are charged. As a result, clients need to have a strong assurance that the CSP is storing all data images that are decided upon in the service contract, and all these counterparts are reliable with the most recent modifications issued by the clients. Map-based provable multicopy. 1) it affords an proof to the clients that the CSP is not corrupt by storing less images; 2) it supports outsourcing of dynamic data, i.e., it supports block-level functions, such as block alteration, addition, deletion, and append; and 3) it permits official users to effortlessly access the file copies stored by the CSP. In addition, we discuss the security against colluding servers, and discuss how to recognize corrupted copies by a little revising the projected scheme.

Keywords: Cloud computing, dynamic environment, data duplication, outsourcing data storage.

I. INTRODUCTION

Outsourcing data to a remote cloud service provider (CSP) permits society to store additional data on the CSP than on private computer systems. Such outsourcing of data storage allows society to focus on improvement and relieves the load of constant server updates and other computing matter. On one occasion the data has been outsourced to a remote CSP which may not be dependable, the data owners drop the direct control over their confidential data. This need of control raises new difficult and demanding tasks connected to data confidentiality and integrity protection in cloud computing.

The confidentiality issue can be feeling by encrypting confidential data before outsourcing to remote servers. As such, it is a vital demand of customers to have strong proofs that the cloud servers still have their data and it is not being corrupt with or partially deleted over time. As a result, many researchers have played attention on the Problem of provable data possession (PDP) and proposed different systems to review the data stored on remote

servers. PDP is a method for authenticating data integrity over remote servers. In a typical PDP model, the data owners produce some metadata for a data file to be used later for verification purposes through a challenge-response protocol with the remote/cloud server.

The owner sends the file to be stored on a remote server which may be untrusted, and erases the local copy of the file. One of the core design ethics of outsourcing data is to provide dynamic behavior of data for a variety of applications. This means that the slightly stored data can be not only accessed by the authorized users, but also efficient and scaled Examples of PDP constructions that deal with dynamic data [1], [6][7][8]. The final are how-ever for a single copy of the data file. PDP method has been obtainable for multiple copies of static data [3]-[5]. PDP system directly deals with multiple copies of dynamic data. When proving multiple data copies, generally system integrity check fails if there is one or more corrupted copies were present. To deal with this issue and recognize which copies have been corrupted, a slight modification has been applied to the proposed scheme.

II. SYSTEM ARCHITECTURE

The cloud computing storage model measured in this work includes three main components as illustrated in Fig. 1: (i) A data owner that can be an organization initially possessing confidential data to be stored in the cloud. (ii) A CSP who handles cloud servers (CSs) and offers paid storage space on its infrastructure to store the owner's files. (iii) Authorized users — a set of owner's clients who have the right to access the remote data. The storage model used in this work can be assumed by much practical requests. For example, e-Health applications can be predicted by this model where the patients' database that includes large and confidential information can be stored on the cloud servers. In these types of applications, the e-Health organization can be measured as the data owner, and the physicians as the approved users who have the right to access the patients' medical history. Many other practical applications like financial, scientific, and educational applications can be observed in similar settings.

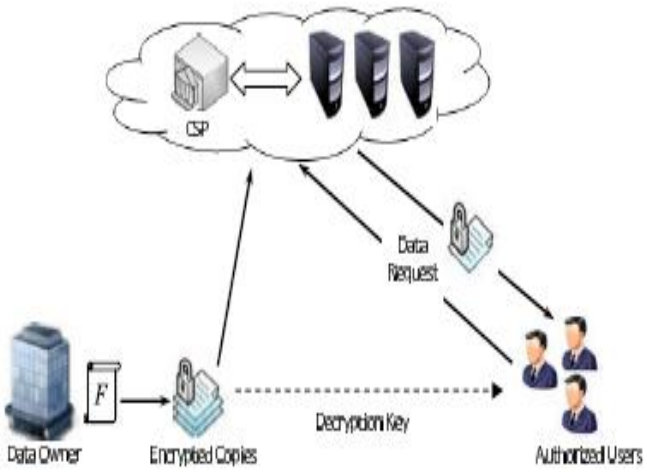

Fig-1: Architecture

III. RELATED WORKS

3.1 Dynamic Provable Data Possession [1]

As storage-outsourcing services and resource-sharing networks have become popular, the problem of efficiently proving the integrity of data stored at untrusted servers has received increased attention. In the provable data possession (PDP) model, the client pre-processes the data and then sends it to an untrusted server for storage, while keeping a small amount of meta-data. The client later asks the server to prove that the stored data has not been tampered with or deleted (without downloading the actual data). However, the original PDP scheme applies only to static (or append-only) files.

3.2 Provable Data Possession at Un-trusted Stores [2]

We introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems. We present two provably-secure PDP schemes that are more efficient than previous solutions, even when compared with schemes that achieve weaker guarantees. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data. Experiments using our implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation.

3.3 MR-PDP: Multiple-Replica Provable Data Possession [3]

Many storage systems rely on replication to increase the availability and durability of data on untrusted storage systems. At present, such storage systems provide no strong evidence that multiple copies of the data are actually stored. Storage servers can collude to make it look like they are storing many copies of the data, whereas in reality they only store a single copy. We address this short- coming through multiple-replica provable data possession (MR-PDP): A provably-secure scheme that allows a client that stores t replicas of a file in a storage system to verify through a challenge-response protocol that (1) each unique replica can be produced at the time of the challenge and that (2) the storage system uses t times the storage required to store a single replica. MR-PDP extends previous work on data possession proofs for a single copy of a file in a client/server storage system [4]. Using MR-PDP to store t replicas is computationally much more efficient than using a single-replica PDP scheme to store t separate, unrelated files (e.g., by encrypting each file separately prior to storing it). Another advantage of MR-PDP is that it can generate further replicas on demand, at little expense, when some of the existing replicas fail.

3.4 Provable Possession and Replication of Data over Cloud Servers [4]

Cloud Computing (CC) is an emerging computing paradigm that can potentially offer a number of important advantages. One of the fundamental advantages of CC is pay-as-you-go pricing model, where customers pay only according to their usage of the services. Currently, data generation is outpacing users' storage availability, thus there is an increasing need to outsource such huge amount of data. Outsourcing data to a remote Cloud Service Provider (CSP) is a growing trend for numerous customers and organizations alleviating the burden of local data storage and maintenance. Moreover, customers rely on the data replication provided by the CSP to guarantee the availability and durability of their data. Therefore, Cloud Service Providers (CSPs) provide storage infrastructure and web services interface that can be used to store and retrieve an unlimited amount of data with fees metered in GB/month. The mechanisms used for data replication vary according to the nature of the data; more copies are needed for critical data that cannot easily be re- produced. This critical data should be replicated on multiple servers across multiple data centers. On the other hand, non-critical, reproducible data are stored at reduced levels of redundancy. The pricing model is related to the replication strategy. Therefore, it is of crucial importance to customers to have a strong evidence that they actually get the service they pay for. Moreover, they need to verify that all their data copies are not being tampered with or partially deleted over time. Consequently, the problem of Provable Data Possession (PDP) has been considered in many research papers. Unfortunately, previous PDP schemes focus on a single copy of the data and provide no guarantee that the CSP stores multiple copies of customers' data. In this paper we address this challenging issue and propose efficient Multi-Copy Provable Data Possession (EMC-PDP) protocols. We prove the security of our protocols against colluding servers. Through extensive performance analysis and experimental results, we demonstrate the efficiency of our protocols.

3.5 Multi-Copy Dynamic Data Possession MN-Pmddp

A map-based provable multi-copy dynamic data possession (MB-PMDDP) has been proposed in this method. This method provides a sufficient guarantee that the CSP stores all copies that are agreed upon in the service contract. Additionally, the method supports outsourcing of dynamic data, i.e., it supports block- level functions such as block alteration, insertion, removal, and append. The certified users, who have the right to access the owner's file, can effortlessly access the copies received from the CSP. ii)A thorough comparison of MB-PMDDP with a reference scheme, which one can obtain by expanding existing PDP models for dynamic single-copy data has been carried.

.

IV. CONCLUSIONS

Outsourcing data to remote servers has turn into a growing trend for many organizations to ease the burden of local data storage and protection. In this work a review is carried out by considering the difficulty of creating

multiple copies of dynamic data file and confirm those copies stored on untrusted cloud servers. The communication between the authorized users and the CSP is measured in our system, where the authorized users can effortlessly access a data copy received from the CSP using a single secret key shared with the data owner. Furthermore, the proposed scheme supports public verifiability, allows arbitrary number of auditing, and allows possession-free verification where the verifier has the capability to verify the data integrity even though they neither possesses nor retrieves the file blocks from the server.

REFERENCES

[1]. C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proc. 16th ACM Conf. Comput. Commun. Secur. (CCS), New York, NY, USA, 2009, pp. 213–222.

[2]. G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), New York, NY, USA, 2007, pp. 598–609.

[3]. R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: Multiple-replica provable data possession," in Proc. 28th IEEE ICDCS, Jun. 2008, 411–420.

A. F. Barsoum and M. A. Hasan. (2010). "Provable possession and replication of data over cloud servers," Centre Appl. Cryptograph. Res., Univ. Waterloo, Waterloo, ON, USA, Tech. Rep. 2010/32.

[4]. Z. Hao and N. Yu, "A multiple-replica remote data possession checking protocol with public verifiability," in Proc. 2nd Int. Symp. Data, Privacy, E- Commerce, Sep. 2010, pp. 84–89.

[5]. G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proc. 4th Int. Conf. Secur. Privacy Commun. Netw. (SecureComm), New York, NY, USA, 2008, Art. ID 9.

[6]. Andhe Dharani, Manjuprasad B., Shantharam Nayak, Vijayalakshmi M. N., Sensor Networks - An Insight on Market Perspective and Real Time Border Monitoring System, International Journal of Sensors and Sensor Networks. Vol. 3, No. 3, 2015, pp. 18-23. doi: 10.11648/j.ijssn.20150303.11

[7]. C. Wang, Q. Wang, K. Ren, and W. Lou. (2009). "Ensuring data storage security in cloud computing," IACR Cryptology ePrint Archive, Tech. Rep. 2009/081. [Online]. Available: http://eprint.iacr.org/

[8]. Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proc. 14th Eur. Symp. Res. Comput. Secur. (ESORICS), Berlin, Germany, 2009, pp. 355–370.

[9]. Z. Hao, S. Zhong, and N. Yu, "A privacy- preserving remote data integrity checking protocol with data dynamics and public. verifiability," IEEE Trans. Know lEng., vol. 23, no. 9, pp. 1432–1437, Sep. 2011.

[10]. Z. Hao and N. Yu, "A multiple-replica remote data possession checking protocol with public verifiability," in Proc. 2nd Int. Symp. Data, Privacy, E- Commerce, Sep. 2010, pp. 84–89.

[11]. F. Dabek, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica. Wide-area cooperative storage with CFS. In Proc. of SOSP '01, 2001.

[12]. F. Dabek, J. Li, E. Sit, J. Robertson, M. F. Kaashoek, and R. Morris. Designing a DHT for low latency and high throughput. In Proc. of NSDI ’04, 2004.

[13]. Y. Deswarte, J.-J. Quisquater, and A. Saidane. Remote integrity checking. In Proc. of Conference on Integrity and Internal Control in Information Systems (IICIS’03), 2003.

[14]. D. L. G. Filho and P. S. L. M. Baretto. Demonstrating data possession and uncheatable data transfer. Cryptology ePrint Archive, 2006. Report 2006/150.

[15]. P. Golle, S. Jarecki, and I. Mironov. Cryptographic primitives enforcing communication and storage complexity. In Proc. of Financial Cryptography, 2002.

A. Haeberlen, A. Mislove, and P. Druschel. Glacier: Highly durable, decentralized storage despite massive correlated failures. In Proc. of NSDI ’05, 2005.

[16]. Juels and B. S. Kaliski. Proofs of retrievability for large files. In Proc. of ACM CCS, October 2007

[17]. Maniatis, M. Roussopoulos, T. Giuli, D. Rosenthal, M. Baker, and Y. Muliadi. The LOCKSS peer-to-peer digital preservation system. ACM Transactions on Computer Systems, 23(1):2–50, 2005.

[18]. S. Reed and G. Solomon. Polynomial codes over certain finite fields. Journal of the Society for Industrial and Applied Mathematics, 8(2):300–304, 1960.

[19]. P. Rogaway. Bucket hashing and its application to fast message authentication. In Proc. of CRYPTO ’95, 1995.

[20]. B. Schroeder, S. Damouras, and P. Gill. Understanding latent sector errors and how to pro

[21]. tect against them. In Proc. of FAST’10,
2010.