INTERNATIONAL JOURNAL OF SCIENTIFIC & ENGINEERING RESEARCH, VOLUME 4, ISSUE 9, SEPTEMBER-2013
ISSN 2229-5518

Framework of Data Integrity Verification for Multi Clouds Using CPDP Scheme

Krunal N. Chandewar, Shailesh T. Khandare

Abstract— In this paper, we address the construction of an PDP scheme for cloud storage to support the growing amount of work of service and data migration, in which we consider the availability of multiple cloud service providers to cooperatively store and manage the clients' infprmtion.Cloud computing environment building is based on totally open architectures and interfaces. It has the powe to combine multiple clouds whose services are provided by an IT department to those in its own organization and a public or private cloud services that are provided by a third party outside the organization to gether to provide high ability of organizations to work together. So, Security in terms of integrity is most important aspect in cloud computing environment. Our experiments show that our solution introduces lower computation and communication overheads in comparison with non-cooperative approaches. We prove the security of our scheme based on Zero-knowledge.

Index Terms— Cloud computing, Cooperative, Data storage, Integrity verification, Multiple Cloud, Proof of retrievability, Provable Data Possession

1 INTRODUCTION

Cloud computing is nothing but a model for a shared pool of configurable computing resources (e.g. storage, servers, networks) that can be rapidly provisioned and released with minimal management effort or service provider Interaction. In recent years, cloud storage service has become a faster profit growth point by providing a comparably low in price, growing amount of work, and a things like copied to any memory location and executed without Modification platform for clients' data.

Since cloud computing environment is constructed based on open architectures and interfaces, it has the capability to incorporate multiple external and/or internal cloud services together to provide high interoperability. We shall call such a distributed cloud environment as hybrid cloud or multi-Cloud. Often, by using virtual infrastructure management, a multi-cloud allows clients to easily access her/his resources remotely through interfaces such as Web services provided by Amazon EC2. There exist various tools and technologies for multi cloud, such as vSphere, VMware, Platform VM Orchestrator, and Ovirt. These tools help cloud providers construct a distributed cloud storage platform, if this is important platform then it is vulnerable to security attacks, it ariese somelosses to the clients.Therefore, it is indispensable for cloud service providers to provide security techniques for managing their storage services.

The main objective of this paper is to provide security and availability of client's data which stored on cloud. This artical not generae any limitation on to communication and computation and further, performance guarantee shall also be taken care of by allowing TTP to verify the correctness of the cloud data on demand without retrieving a copy of the whole data.

2 MOTIVATION

The cloud computing has been seen as the next generation of enterprise IT infrastructure, applications and users will also concentrated all the information stored in the cloud data center, this new data storage model will bring new challenges. One of the most serious and most attention issues, that is in the cloud, is the data storage with security and integrity verification.

For that we take some example as, storage service providers may easily order their own data to hide an error, storage service providers to save cost and space, remove very less accessed data, Therefore, how to backup data files in the user not the case, but we have to found an efficient and securely ways of good information to perform very periodically verification, by allowing users to know his information file is going to be stored securely on the top of the server, this information storage as data in cloud computing environment is an important security issue.

From the mentioned above, the user information like data files stored on the server in the cloud, in order to know this server actually storing data files, users will be periodically make the server a number of challenges, so that the server that the use of the archives were stored in the cloud does to the user ease. Fig shows the first user data files stored on the serv-

- *Master of Engg. in Computer Science & Engg., BNCOE,Pusad,Sgbau Amravati,India PH-08275299090. E-mail: krual_chandewar@rediffmail.com*
- *Master of Engg. in Computer Science & Engg., BNCOE,Pusad,Sgbau Amravati,India PH-09764996386. E-mail: khandare.shailesh@rediffmail.com*

er, Fig. shows the server certificate data to the user actually stored on the server.

The whole, the user want to his life collection of digital data (such as pictures, video, art, etc.) to a third party to store, share their collections to friends and family with the authorization to use the watch. For owners, these can be quite precious, so the user would like to ensure that his file is actually stored on its servers, and can download at any time. In order to verify data integrity, only users with limited resources, personal devices such as PDA or mobile phone, in this actual case, we have two design requirements are very important – (1) cipher text data outsourcing (2) bandwidth and computational efficiency.

3 CONTRIBUTION

Our proposed sheme has two diffrent main points

3.1 Efficiency And Security

The plan decided by the cooperative provable data-possession is very safe to generate public and private key Encryption will be very clear, productive in the use of SecretKeyGen and TagGen algorithms. In this most of the time parameters are always generated and the key exchange takes place so it is more secure than symmetric and asymmetric algorithm. As, our plan is nothing but more efficient than the other techniques which we disscuss before. Because here there is no need of lots of data encryption and no additional posts on the symbol block, the ratio is very goodly secure because we encrypt our data to prevent un-authorized third parties to know about its contents.

3.2 Public Verifiability

We goodly plan a maximum variation of cooperative provable data possession, to provide public validation. which give permission to people other than the owner for information on the server has more efficient because of that it does not need the information for every block encryption. Paper structure Framework for the rmaining paper is as given as follows. In oue pape, we describe the related work. that part purly describes a data integrity for cross cloud environment using cooperative provable data possession scheme to prove that the structure, emphasizing the characteristics of cooperative provable data possession and the related parameters. in that we introduce the cooperative provable data possession can be publicly verifiable information of that to prove a structure

4 RELATED WORKS

To check the availability and also the integrity of outsourced data in our cloud storages, researchers have developed two basic approaches which we called as Provable Data Possession and Proofs of Retrievability. Our one writer first proposed the Provable Data Possession model for telling us the possession of files on that untrusted storages and very efficiently provided an RSA-based scheme for static case that achieves the $O(1)$ communication cost of our proposed system. They further also proposed a very popular publicly verifiable version, which allows anyone who using this system, not just the owner, to challenge the given server for this data possession. These trend very goodly extended application areas of Provable Data Possession protocol due to the separation of these data owners and the given users. However, these schemes are not secured influent of replay attacks in most important dynamic scenarios because of the nature of dependence on the index of blocks. So here they do not fit for multi-cloud storage servers due to the loss of homomorphism property in the verification process.

In order to give support to the dynamic data operations, our scientist named Ateniese developed a dynamic PDP solution which is also called Scalable Provable Data Possession. They again developed a lightweight PDP scheme based on cryptographic function and algorithm like symmetric key encryption, but the given servers can easily deceive the owners by using that previous metadata or a responses due to lack of that randomness in the challenges. The numbers of all updates and those challenges are very limited and totally fixed in advance and users of this system cannot perform the block insertions anywhere in this scheme. Based on this work, the another scientist name Erway introduced the two Dynamic Provable Data Possession schemes with a hash function tree to give realization to $O(\log n)$ communication and computational costs for a n-block file. The first one basic scheme, called DPDP-I, which retains the drawback of our Scalable Provable Data Possession, and in the given 'blockless' scheme, it called DPDPII, the data blocks $\{mij\}j\in[1,t]$ can be easily leaked by the given response of this system challenge, $M = \Sigma t\ j=1\ ajmij$, where aj is nothing but a random challenge value. Where these schemes are not that much effective for a multi-cloud environment because of the verification path of the challenge block cannot stored in a cloud

Juels and Kaliski the name of the scientist presented a POR scheme, which relies largely on the preprocessing steps that this client conducts before sending that file to Cloud Service Provider. But, these operations prevent any efficient method of extension for updating the data. Shacham and Waters the name of the scientist those proposed an improved version of protocol which we called Compact POR, which uses a homomorphic property to very efficiently aggregate proof into $O(1)$ authenticator value and $O(t)$ computation cost for t challenge blocks, but we know that their solution is also static and could not prevent any leakage of our data blocks in the verification process of this system. The scientist named Wang presented a dynamic scheme of system with $O(\log n)$ cost by integrating the Compact POR scheme and Merkle Hash Tree into DPDP. Here, several POR schemes and models have been recently proposed In Bowers introduced a distributed cryptographic of the system which allows a set of known servers to easily solve the problem. This system is mainly dependent on an integrity-protected error correcting code (IP-ECC), which greatly improves the security and efficiency of existing tools, However, a file be transformed into a l distinct segments with the

exactly same length, which are distributed across the *l* servers. Hence, this system is more suitable for the RAID rather than that for a cloud storage.
Our.

part of hand to this paper is, we address the most of the problem of provable data possession in distributed cloud environments from the following aspects as a high security, transparent verification, and a high performance. To achieve these main goals, we first propose a efficient verification framework for the multi-cloud storage along with the two fundamental techniques: hash index hierarchy and homomorphic verifiable response.
We after then demonstrate that possibility of the constructing a cooperative Provable Data Possession (CPDP) scheme without the compromising of data privacy based on that modern cryptographic techniques, such as given interactive proof system (IPS). We then further introduce a effective construction of that CPDP scheme using mentioned structure. Moreover, we also give a security analysis of our this CPDP scheme from the best IPS model. We again prove this construction is nothing but multi-prover zero-knowledge proof system (MP-ZKPS) which has completeness, knowledge soundness, and zero-knowledge properties. These properties ensure that CPDP scheme can be implement the security against data leakage attack and tag forgery attack.

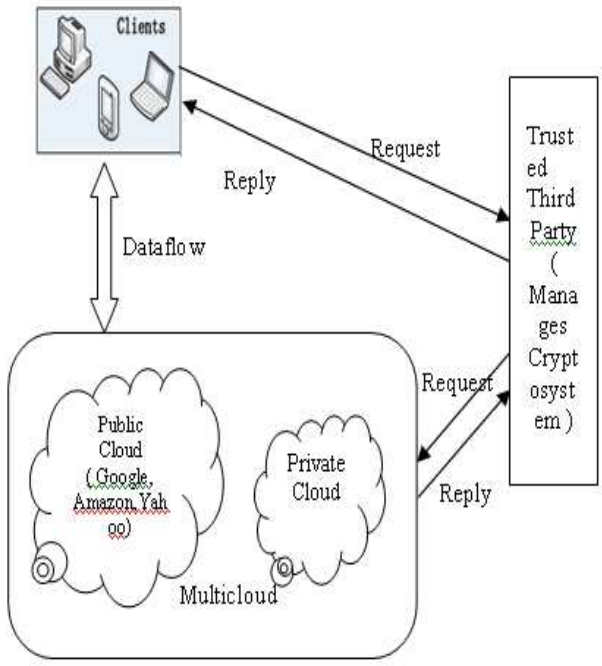

Fig. 1. Verification of integrity in cross cloud environment

5 AN INTEGRITY FOR CROSS CLOUD ENVIRONMENT USING CPDP SCHEME

In this most important Agreement, we know the password system based on Coperative Provable Data Possession, the main idea is outsource the file before the data block encryption, and validation of fixed-size tags, each tag are to be included in the block information. Fig. 1 is a Cross cloud environment in Coperative Provable Data Possession agreement setting the stage diagram: Although existing Coperative Provable Data Possession schemes offer a publicly accessible remote interface for checking and managing the tremendous amount of data, in the majority of that existing Coperative Provable Data Possession schemes are incapable to satisfy the inherent requirements from multiple clouds in terms of communication and computation costs. To address this main problem, we consider a multi-cloud storage service which illustrated in Figure 1.

5.1 System Architecture

In this given architecture, a data storage service involves three different entities which are Clients who have a very large amount of data which is to be stored in multiple clouds and have the permissions to access and manipulate the stored data Cloud Service to the Providers who work together to provide data storage services and have enough storages and the computation resources. and the Trusted Third Party who is trusted to store verification parameters and offer public query services for these parameters In this given architecture, we first consider the existence of multiple Cloud Service Providers to cooperatively store and maintain the given clients' data. The verification procedure is described as follows: Firstly, a client (data owner) uses the secret key to pre-process a file which consists of a aggrigationction of n block, produced a set of public verification information that is stored in TTP, sends the file and some verification tags to CSPs, and may remove its local copy; Then, with the help of a verification protocol, the clients can issue a challenge for one CSP to verify the integrity and availability of outsourced data with respect to public information which is to be stored in TTP.

5.2 Protocol Directions

We never predict that CSP is trust to guarantee the security of that stored data, nor assume that data owner has that ability to collebrate the evidence of the CSP's fault after errors have been occured. To achieve this milstone, a TTP server is build as a core trust base on that cloud for the sake of security. We assume that the TTP is very reliable and independent through the following functions to buildup and 1) SecretKeyGen: Takes a security parameter k as his input, and returns back a secret key Sk or a public-secret keypair (Pk, Sk); 2) VeriTagGen (Sk,F,P): Takes as inputs a secret key Sk, a file

F, and set of that cloud storage providers P = {Pk}, and returns the triples (St,Vp,At), where St is the secret in that tags, Vp = (u,H) is nothing but a set of verification parameters u and an index hierarchy H for F, At = { At (k) bn} Pk belongs to P) which denotes a set of all tags, At maintain the Coopeative provable data posession cryptosystem; to generate and store data owner's public key; and to store the public parameters which used to execute the verification protocol in the Coopeative provable data posession the fraction F (k) of F in Pk.(k) is the verification tag of that scheme. Note that the TTP is not directly takes participate in the Coopeative provable data posession scheme in order to minimize the complexity of cryptosystem that (Cooperative-PDP). A Cooperative provable data possession scheme S' is a colleboration of two algorithms And an interactive proof system, S' = (K, T, P):

The verification phenomenon is described given as follows: Firstly the client (data owner) has uses the secret key which is being generated by proposed algorithm SecretKeyGen to pre-process a file which composd of a collection of n blocks, which produces a set of public verification information generated by that VeriTagGen algorithm that is stored in TTP, which transmits the given file and some verification tags to CSPs, and then it may delete its local copy. Then, by taking use of this verification protocol,what happen the clients can issue a challenge for one CSP to verify the integrity and availability of that outsourced data with respect to public information stored in that TTP.

In proposed system a CPDP in cross cloud S= (SecretKeyGen, VeriTagGen, proof) is a collebration of that two algorithms (SecretKeyGen, VeriTagGen) and the interactive proof system proof, as given as follows: Proof**(**P,V**):** Is nothing but a protocol of proof of that data possession between CSPs (P = {Pk) and the verifier (V), that is, $< \Sigma \text{Pk E P P(k)} , \text{F (k)} , \text{Vp (k) (Pk, Vp,)}$ where Pk takes input file F(k) and a set of tags u(k), and a public key pk and a set of public parameters Vp is the common input between the P and V. At the end of the protocol run, V returns a bit {O / 1} denoting false and true

The given Proposed work neither assumes that CSP is trust to guarantee the security of that stored data, nor assume's data owner has the power to combine the evidence of that CSP's fault after errors have been occured. To reach this goal, a TTP server is developeded as a core trust base on the cloud for the sake of security. The given work assumes the TTP is reliable and independent with help of the following functions: to setup and maintain the CPDP cryptosystem; to produce and store data owner's public key; and to store the public parameters which is used to execute the verification protocol in the CPDP scheme. But that Trusted Third Party is not directly included in the CPDP scheme in order to minimize the complexity of that cryptosystem. This is proposed cross cloud scheme for key generation, tag generation and to check that protocol.

5.3 Flow Chart of Proposed System

This proposed system on that client side will work for two conditions for storing the data request and for Accessing the data request. If client want to store his data, with the help of that Trusted Third Party Secret key is generated, by using that secret key data is to be stored. For accessing tht data, Firstly Trusted Third Party check for trust between that clouds and then check for trusted key between the client and Trusted Third Party,And user will get the data. As shown in fig 2, the flow chart of the proposed system.

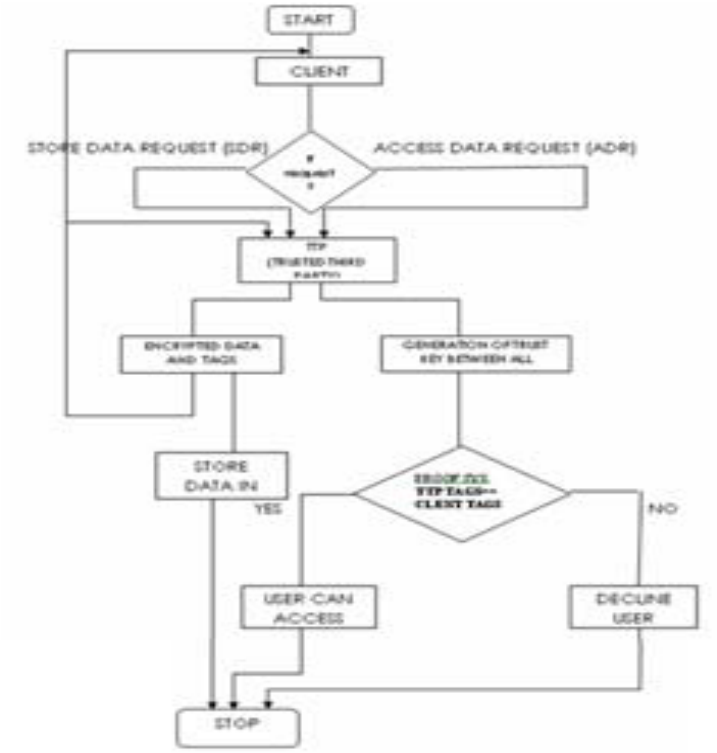

Fig. 2. Flow chart for proposed system

Conclusion

As we have to disscuss about Conclusion We focused the core issues, that untrusted server is able to store customer information. We can use cooperative provable data possession scheme, in this system which reduce our data block access, and amount of computation on the server and client. Also it reduces server traffic. Our design and development on the cooperative provable data possession program is mainly based on the usage of Public and Private keyencryption system. It goes again one step more what we did in the past, the improvement has brought to the bandwidth, computation and storage system. And then it applied the public TTP verification.\ finally, we also expect our program, it supports dynamic outsourcing of the given information make it a more realistic application of that cloud computing environment.

ACKNOWLEDGMENT

The work of krunal N chandewar & shailesh T khandare was supported by the National Natural Science Foundation of India (Project No.212014026). This work was partially supported by the grants from US National Science Foundation (NSFIIS-0900970 and NSF-CNS-0831360)

REFERENCES

[1] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. T. Foster, "Virtual infrastructure management in private and hybrid clouds," IEEE Internet Computing, vol. 13, no. 5, pp. 14–22, 2009.

[2] Protection of Identity Information in Cloud Computing without Trusted Third Party by Rohit Ranchal, Bharat Bhargava, Lotfi Ben Othmane, Leszek Lilien, Anya Kim, Myong Kang, Mark Linderman 2010

[3] A. Juels and B. S. K. Jr., "Pors: proofs of retrievability for large files," in ACMConference on Computer and Communications Security, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds. ACM, 2007, pp. 584–597.

[4] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proceedings of the 4th international conference on Security and privacy in communication netowrks, SecureComm, 2008, pp. 1–10.

[5] An Open Source Solution for Virtual Infrastructure Management in Private and Hybrid Clouds by Borja Sotomayor, Rub´en S. Montero, Ignacio M. Llorente, and Ian Foster 2009

[6] H. Shacham and B. Waters, "Compact proofs of retrievability," in ASIACRYPT, ser. Lecture Notes in Computer Science, J. Pieprzyk, Ed., vol. 5350. Springer, 2008, pp. 90–107.

[7] Q. Wang, C.Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in ESORICS, ser. Lecture Notes in Computer Science, M. Backes and P. Ning, Eds., vol. 5789. Springer, 2009, pp. 355–370.

[8] Scalable and Efficient Provable Data Possession by Giuseppe Ateniese,Roberto Di Pietro,Luigi V. Mancin 2008

[9] K. D. Bowers, A. Juels, and A. Oprea, "Hail: a high-availability and integrity layer for cloud storage," in ACM Conference on Computer and Communications Security, E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds. ACM, 2009, pp. 187–198.

[10] Y. Dodis, S. P. Vadhan, and D. Wichs, "Proofs of retrievability via hardness amplification," in TCC, ser. Lecture Notes in Computer Science, O. Reingold, Ed., vol. 5444. Springer, 2009, pp. 109–127.

[11] L. Fortnow, J. Rompel, and M. Sipser, "On the power of multi-prover interactive protocols," in Theoretical Computer Science, 1988, pp. 156–161.

[12] Provable Data Possession at Untrusted Stores by Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner 2007

[13] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: A berkeley view of cloud computing," EECS Department, University of California, Berkeley, Tech. Rep., Feb 2009.

[14] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in Advances in Cryptology (CRYPTO'2001), vol. 2139 of LNCS, 2001, pp. 213–229.

[15] O. Goldreich, Foundations of Cryptography: Basic Tools. Cambridge University Press, 2001.

[16] A Fresh Look at the Reliability of Long-term Digital Storage by Mary Baker,Mehul Shah,David S. H. Rosenthal 2006

[17] J.-L. Beuchat, N. Brisebarre, J. Detrey, and E. Okamoto, "Arithmetic operators for pairing-based cryptography," in CHES, ser. Lecture Notes in Computer Science, P. Paillier and I. Verbauwhede, Eds., vol. 4727. Springer, 2007, pp. 239–255.

[18] Space-Ef?cient Block Storage Integrity by Alina Oprea, Michael Reiter, and Ke Yang 2005

[19] A. Bialecki, M. Cafarella, D. Cutting, and O. O'Malley, "Hadoop: A framework for running applications on large clusters built of commodity hardware," Tech. Rep., 2005. [Online]. Available: http://lucene.apache.org/hadoop/

[20] E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds., Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, USA, November 9-13, 2009. ACM, 2009.