Middle-East Journal of Scientific Research 24 (Techniques and Algorithms in Emerging Technologies): 484-488, 2016
ISSN 1990-9233;
© IDOSI Publications, 2016
DOI: 10.5829/idosi.mejsr.2016.24.TAET23628

Secure Data in Muliti-Cloud Using Multi Integrity Auditing Protocol in Drops Teqniques

[1]S. Indhumathi and [2]A.S. Radhika D. Renuga Devi

[1]PG Scholar, Department of CSE,
Vivekananda College of Engineering for Women, Tamilnadu, India
[2]Assistant Professor, Department of CSE,
Vivekananda College of Engineering for Women, Tamilnadu, India

Abstract: Cloud computing is provide a dynamically scalable resources provisioned as a service over the webpage. The third-party, on-demand, self-service, pay-per-use and seamlessly scalable computing resources and services are offered by the cloud environment promise to reduce capital as well as operational expenditures for hardware and software. Various distinct architectures are introduced and discussed according to their security and privacy capabilities and prospects. It provides four distinct models in form of abstracted multi-cloud architectures. These developed multi cloud architectures allow to categorize the available schemes and to analyze them according to their security benefits. An assessment of the different methods: Replication of applications, Partition of application System into tiers, Partition of application logic into fragments and Partition of application data into fragments is given in particular. In addition, enabling public audit ability for cloud storage is of critical importance so that users can resort to a third party auditor (TPA) to check the integrity of outsourced data and be worry-free. This paper proposes a secure cloud storage system supporting Isolation-preserving public auditing. It further extends the result to enable the TPA to perform audits for multiple users simultaneously and efficiently.

Key words: Cloud Computing · Multi-cloud · Integrity · Isolation Preserving Auditing · TPA

INTRODUCTION

A Cloud computing creates the security issues and challenges. The required trust in the cloud provider and attacks on cloud interfaces to misusing the cloud services for attacks on other systems.

The main problem that the cloud computing paradigm implicitly contains is that of secure outsourcing of sensitive as well as business-critical data and processes. When considering using a cloud service, the user must be aware of the fact that all data given to the cloud provider leave the own control and protection sphere. Even more, if deploying data-processing applications to the cloud. A cloud provider gains full control on these processes. Hence, a strong authenticate relationship between the cloud provider and the cloud user is considered a general prerequisite in cloud computing. Depending on the political context this trust may touch legal obligations.

An attacker that has access to the cloud storage component can take snapshots or alter data in the storage. An attacker that also has access to the processing logic of the cloud can also modify the functions and their input and output data. A cloud provider to be honest and handling the customer affairs in a respectful and responsible manner, there still remains a risk of malicious employees of the cloud provider, successful attacks and compromisation by third parties.

Replication of applications allows to receive multiple results from one operation performed in distinct clouds and to compare them within the own premise. It Shows the user's to get evidence on the integrity of the result.

Partition of application System into tiers allows separating the logic from the data. It provides additional protection against data leakage due to flaws in the application logic.

Corresponding Author: S. Indhumathi, PG Scholar, Department of CSE, Vivekananda College of Engineering for Women, Tamilnadu, India.

Partition of application logic into fragments allows distributing the application logic to distinct clouds. First, no cloud provider learns the complete application logic. Second, no cloud provider learns the overall calculated result of the application. Also leads to data integrity and confidentiality.

Partition of application data into fragments allows distributing fine-grained data to distinct clouds. None of the involved cloud providers gains access to all the data, which safeguards the data's confidentiality.

Each of the introduced architectural patterns provides individual security merits, which map to different application scenarios and their security needs. Obviously, the patterns can be combined resulting in combined security merits, but also in higher deployment and runtime effort. The following sections present the four patterns in more detail and investigate their merits and flaws with on the stated security requirements under the assumption of one or more compromised cloud systems.

Related Works: In [2] the authors YINQIAN ZHANG and ARI JULES details the construction of an access-driven side-channel attack by which a malicious virtual machine extracts fine-grained information from a victim VM running on the same physical computer. This attack is the first such attack demonstrated on a symmetric multiprocessing system virtualized using a modern VMM.

In [3], the authors JURAJ SOMOROVSKY, MARIO HEIDERICH, NILS GRUSCHKA and LUIGI LO IACONO stated that Cloud Computing resources are handled through control interfaces. It is through these interfaces that the new machine images can be added, existing ones can be modified and instances can be started or ceased. Effectively, a successful attack on a Cloud control interface grants the attacker a complete power over the victim's account, with all the stored data included. The authors provided a security analysis about the control interfaces of a large Public Cloud (Amazon) and widely used Private Cloud software (Eucalyptus). Their research results are alarming: in regards to the Amazon EC2 and S3 services, the control interfaces could be compromised via the novel signature wrapping and advanced XSS techniques. Similarly, the Eucalyptus control interfaces were vulnerable to classical signature wrapping attacks and had nearly no protection against XSS. As a follow up to those discoveries, they additionally describe the countermeasures against these attacks, as well as introduce a novel "black box" analysis methodology for public Cloud interfaces. It considered security and privacy aspects of real-life cloud deployments, independently from malicious cloud providers or customers. They focused on the popular Amazon Elastic Compute Cloud (EC2) and gave a detailed and systematic analysis of various crucial vulnerabilities in publicly available and widely used Amazon Machine Images (AMIs) and show how to eliminate them.

In [5], the authors GEORGE DANEZIS and BENJAMIN LIVSHITS stated that privacy is considered one of the key challenges when moving services to the Cloud. A Solution like access control is brittle while fully homomorphic encryption has been hailed as the silver bullet for this problem is far from practical. And can we already deploy architectures with similar security properties? They proposed one such architecture that provides privacy, integrity and leverages the Cloud for availability while only using cryptographic building blocks available today.

In [6], the authors STEPHAN GROG and ALEXANDER SCHILL stated that cloud computing, i.e. providing on-demand access to virtualized computing resources over the Internet, is one of the current trends in IT. Today, there are already several providers offering cloud computing infrastructure, platform and software services. Although the cloud computing paradigm promises both economical as well as technological advantages, many potential users still have reservations about using cloud services as this would mean to trust a cloud provider to correctly handle their data according to previously negotiated rules.

Existing System Methodology: The existing system, an assessment of the different methods: Replication of applications, Partition of application System into tiers, Partition of application logic into fragments and Partition of application data into fragments are carried out.

The existing system utilizes the technique of public key based homomorphic linear authenticator (or HLA for short), which enables Third Party Auditor to perform the auditing without demanding the local copy of data and thus drastically reduces the communication and computation overhead as compared to the straight forward data auditing approaches.

By integrating the HLA with random masking, the protocol guarantees that the TPA could not learn any knowledge about the data content stored in the cloud server during the efficient auditing process. The aggregation and algebraic properties of the authenticator further benefit our design for the batch auditing.

Various prime numbers are assigned to tags for each segment of the file which is stored in the server. Each segment is having two prime numbers each of which belongs to a different prime order. The third party auditor knows the prime numbers in a random manner.

Unauthorized data leakage remains a problem due to the potential exposure of decryption keys. Only single cloud provider environment is considered.

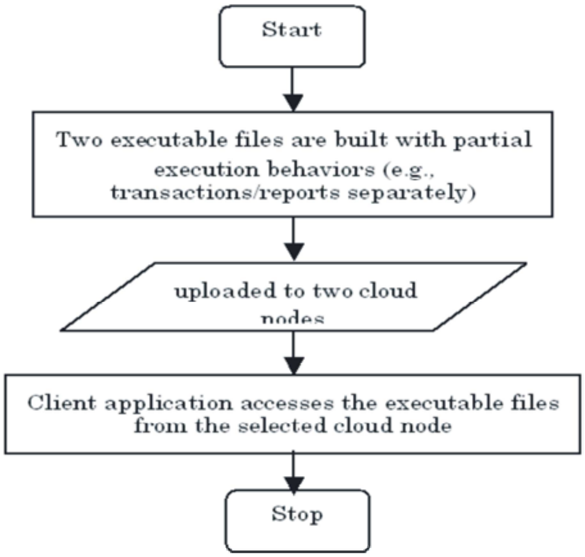
Start
Two executable files are built with partial execution behaviors (e.g., transactions/reports separately)
uploaded to two cloud nodes
Client application accesses the executable files from the selected cloud node
Stop

Fig. 1.1: Multi-Cloud Methodology

During verification, the third party auditor sends the numbers as a random challenge and if the numbers to be matched with tags, then the file integrity is said to be verified.

All the nodes are treated equally and weak capable nodes also require huge computations. All the mirror nodes store the file with same encryption mechanism. Unauthorized data leakage still remains a problem due to the potential exposure of decryption keys. Only single cloud provider environment is considered.

Proposed System Methodology: The proposed system includes all the existing system approach which covers multiple cloud service provider environments. Also the size blocks of data are being processed with varying size nature in different cloud locations having the same copy of data. The data blocks are stored and retrieved in the different cloud locations based on the storage and computational capability. The system explores such issue to provide the support of variable-length block verification the privacy level for all cloud providers is analyzed by trusted authority and security degree and performance is quantified for encryption algorithms. The proposed system has following advantages. Partial data of files are taken from multiple mirror locations and send to selected client. Suitable for very large size files. Irrelevant size blocks of data are handled by among the multiple cloud service providers based on their computational capabilities.

The Different trust level is set to different cloud providers and encryption/decryption is varied based on the clouds computational capability.

ADD Cloud Node Data: The cloud node id and the cloud provider name is added. There are more cloud nodes for the single cloud provider. From the trusted authority, the cloud node receives secret tags for file blocks so that the blocks can be processed/ verified by the cloud nodes.

Multi-Cloud Security: The files are added to cloud nodes and executed based on a) Replication of applications from the random cloud node, b) Partition of application System into tiers such that even the web server does not know the location of record in database server, c) Partition of application logic into fragments such that half of the application login in one file stored in one cloud node and another half of the application logic in other file stored in another cloud node and d) Partition of application data into fragments such that partial records in one cloud database and remaining records in other cloud database.

Privacy Preserving Auditing Protocol: The file name is selected, the file content is split into various segments and each segment is given two prime numbers each of which belongs to two prime order. One is given to the user, other is given to third party auditor. The combination of the two is kept in server. During auditing, third party auditor randomly picks the segment ids and send corresponding prime number vector to cloud server. If the credentials match, then the file integrity is said to be verified.

Batch Auditing Protocol: During auditing, two processes of same third party auditor randomly pick the two set of segment ids and send corresponding prime number vectors to cloud server. If the credentials match, then the file integrity is said to be verified.

Storage and Computational Capability Based File Storage File Selection: The file content is selected from client files. The file data is saved in the cache.

Encryption: DES (Data Encryption Standard), or AES (Advanced Encryption Standard) encryption work is carried out and the selected file is encrypted.

Speed: The requirement of this level presents that no sensitive information in the data. Cloud location with low computational capability uses weak encryption composition (DES) and high computational capability uses more encryption (AES) to obtain more performance for using cloud services.

Decryption: In this module, decryption work (DES and AES) is carried out.

RESULTS AND DISCUSSION

Table 1.1 Shows the Replication Requirement in Cloud Nodes and Computational Overhead in Client/cloud Node/cloud Database and Third Party System.

Table 7.1: Replication Requirement and Computation Overhead

Method	Replication	Computation Overhead
Resource Replication	Required	In client only
PIR based segmentation	Not required	Low in client tier/ More in database tier (stored procedure) and negligible in web tier
Segmentation of application	Not required	In client only
Third party auditing	Not required	High In client, Low in third party system and negligible in cloud node

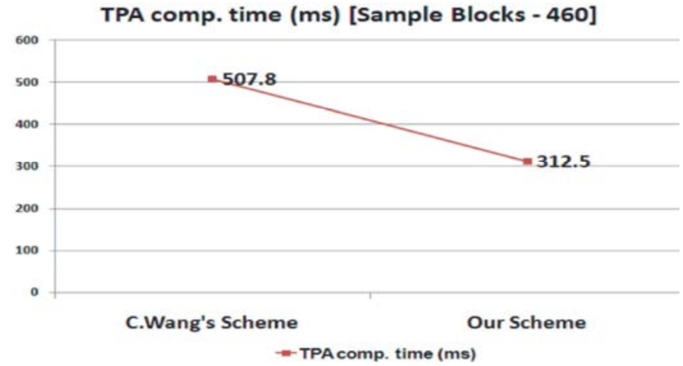

Fig. 1.2: TPA Computation Time Chart Comparison (C. WANG, S.S. CHEME VS Our Scheme) (Chart Type – Line Chart).

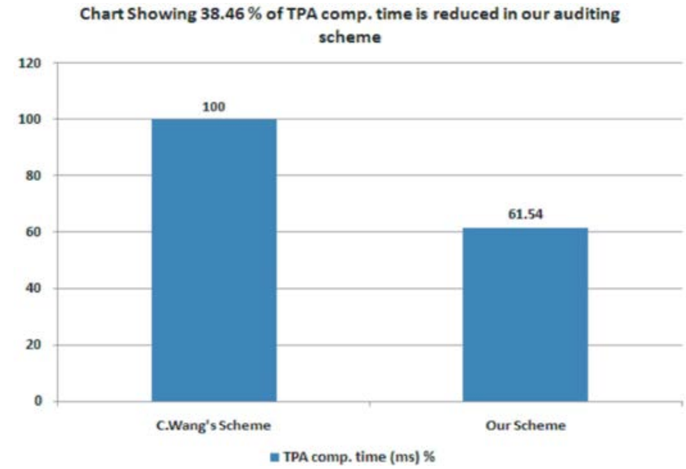

Fig. 1.3: Chart Comparison for TPA Computation Time in %.

CONCLUSION AND FUTURE ENHANCEMENT

The problem of secure communication is eliminated. Also the application required less working experience in systems to run the software. The application is tested well so that the end users use this software for their whole operations.It is believed that almost all the system objectives that plans have the commencement of the software development have been net with and the implementation process of the project is completed. A trial run of the system has been made and is giving good results the procedures for processing is simple and regular order. The process of preparing plans have been missed out which might be considered for further modification of the application. Also effectively stores and retrieves the records from the cloud space database server. The records are encrypted and decrypted whenever necessary so that they are secure. The following enhancements are should be in future. The data integrity in the cloud environment is not considered. The error situation is to be recovered if there is any mismatch. The web site and a database has been hosted in real cloud place during the implementation.

REFERENCES

1. Ristenpart, T., E. Tromer, H. Shacham and S. Savage, 2009. "Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), pp: 199-212.
2. Zhang, Y., A. Juels, M.K.M. Reiter and T. Ristenpart, 2012. "Cross-VM Side Channels and Their Use to Extract Private Keys," Proc. ACM Conf. Computer and Comm. Security (CCS '12), pp: 305-316.
3. Somorovsky, J., M. Heiderich, M. Jensen, J. Schwenk, N. Gruschka and L. Lo Iacono, 2011. "All Your Clouds Are Belong to Us: Security Analysis of Cloud Management Interfaces," Proc. Third ACM Workshop Cloud Computing Security Workshop (CCSW '11), pp: 3-14.
4. Bugiel, S., S. Nu ̈ rnberger, T. Po ̈ppelmann, A.R. Sadeghi and T. Schneider, 2011. "AmazonIA: When Elasticity Snaps Back," Proc. 18th ACM Conf. Computer and Comm. Security (CCS '11), pp: 389-400.
5. Danezis, G. and B. Livshits, 2011. "Towards Ensuring Client-Side Computational Integrity (Position Paper)," Proc. ACM Cloud Computing Security Workshop (CCSW '11), pp: 125-130.
6. Groß, S. and A. Schill, 2011. "Towards User Centric Data Governance and Control in the Cloud," Proc. IFIP WG 11.4 Int'l Conf. Open Problems in Network Security (iNetSeC), pp: 132-144.
7. Burkhart, M., M. Strasser, D. Many and X. Dimitropoulos, 2010. "SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics," Proc. USENIX Security Symp., pp: 223-240.
8. Hubbard, D. and M. Sutton, 2010. "Top Threats to Cloud Computing V1.0," Cloud Security Alliance, http://www.cloudsecurityalliance.org/topthreats, 2010.
9. Armbrust, M., A. Fox, R. Griffth, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica and M. Zaharia, 2010. A view of cloud computing. Commun. ACM, 53(4): 50-58.
10. Meushaw, R. and D. Simard, 2000. A network on a desktop. NSA Tech Trend Notes, 9(4), 2000. http://www.vmware.com/pdf/TechTrendNotes.pdf.
11. England, P. and J. Manferdelli, 2006. Virtual machines for enterprise desktop security. Information Security Technical Report, 11(4): 193-202.
12. Garfinkel, T., B. Pfaff, J. Chow, M. Rosenblum and D. Boneh, 2003. Terra: a virtual machine-based platform for trusted computing. In ACM Symposium on Operating Systems Principles, pp: 193-206. ACM, 2003.
13. Acii¸cmez., O., 2007. Yet another microarchitectural attack: Exploiting I-cache. In ACM Workshop on Computer Security Architecture, pp: 11-18, October 2007.
14. Ristenpart, T., E. Tromer, H. Shacham and S. Savage, 2009. Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds. In 16th ACM Conference on Computer and Communications Security, pp: 199-212.