ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 4, April 2016

Auditing-As-A-Service for Multiple Cloud

S.J.Vijeyrajkumar[1], C.RajeshBabu[2]

P.G. Student, Department of Computer Engineering, SRM University, Chennai, India[1]

Assistant Professor, Department of Computer Engineering, SRM University, Chennai, India[2]

ABSTRACT: This paper proposes Auditing as a Service for multiple cloud. Our experiments yield interesting and meaningful knowledge that can facilitate the effective utilization of resources in a large scale multi cloud environment. In the cloud environment user can easily modify and share data as a group. To ensure shared data integrity can be verified publicly user in the group need to compute signatures on all the blocks in shared data. Different blocks in shared files are generally signed by different users due to modifications are performed by different users. Authorization/authentication process is missing between the auditor and the cloud service provider anyone can challenge the cloud service provider for a validation of integrity of certain file which potentially puts the quality of the so called 'auditor as a service' at risk. In this paper we provide a formal analysis for possible types of auditing data and propose a scheme that can fully support authorized auditing by Homomorphic and matchmaking algorithm.

KEYWORDS: Cloud Auditor, multi-cloud environment, trust scheme, resource matchmaking

I. INTRODUCTION

The trend toward cloud computing boomed in the late 1980s with the concept of grid computing when, for the first time, a large number of systems were related to a single problem, usually scientific in nature and requiring exceptionally high levels of parallel computing. In Europe , long length optical networks are used to tie multiple universities into a massive computing grid in order that systems could be shared and scaled for large scientific calculations. Grid computing provided a virtual pool of computation sources but it's different than cloud computing. Grid computing specifically refers to power several computers in parallel to solve a particular, individual problem, or to run a specific application. In grid computing, the purpose is on moving a workload to the location of the needed computing systems, which are mostly remote and are readily applicable for use. Usually a grid is a cluster of servers on which a broad task could be divided into minor tasks to run in parallel. From this point of view, a grid could actually be viewed as just one virtual server. Grids also lack applications to conform to the grid software interfaces. In a cloud environment, computing and enlarged IT and business resources, such as servers, storage, network, applications and processes, can be dynamically carved out from the underlying hardware infrastructure and made available to a workload. In addition, while a cloud can plan and support a grid, a cloud can also support non-grid environments, such as a three-tier Web architecture running popular or Web 2.0 applications. In the 1990s, the concept of virtualization was expanded beyond virtual servers to greater levels of abstraction—first the virtual platform, including storage and network resources, and subsequently the virtual operation, which has no specific underlying infrastructure. Utility computing offered bunch as virtual platforms for computing with a evaluated business model. More recently software as a service (SaaS) has raised the level of virtualization to the operation, with a business model of charging not by the resources consumed but by the value of the application to users. The concept of cloud computing has evolved from the concepts of grid, utility and SaaS. It is an developing model through which users can gain access to their applications from anywhere, at any time, through their linked devices. These operations reside in massively scalable data centers where compute resources can be dynamically supplied and shared to achieve significant economies of scale. Companies can choose to contribute these resources using public or private clouds, depending on their specific needs. Public clouds expose services to clients, businesses and users on the Internet. Private clouds are generally restricted to use within a company behind a firewall and have fewer security hazards as a result. The strength of a cloud is its infrastructure management,

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 4, April 2016

implemented by the maturity and progress of virtualization technology to manage and better utilize the underlying resources through automatic supplying, re-imaging, workload rebalancing, monitoring, systematic change request conducting and a changing and automated security and flexibility platform. As more enterprises add cloud computing the level of applications is transfer toward more mission critical and SaaS will become a mainstay of IT strategies. With this computing infrastructure in place some companies are already poised to offer new cloud-based software applications.

Multi-cloud environments breaks infrastructure-as-a-service (IaaS) clouds to integrate resources from multiple cloud infrastructures. These classifications allow users to take advantage of differences in various clouds, including price, achievement, capability, and opportunities differences. As an example, a user may leverage multiple clouds, including community clouds, such as Future Grid, and for-pay public clouds, such as Amazon EC2. A community cloud, for example, often is the user's liked cloud because it is served at a reduced monetary cost, or possibly even free. However, it may offer somewhat fewer assets than larger for-pay public cloud providers. Users are then faced with a dilemma of choosing where to deploy their occurrences. In such a scenario, a user may choose to delay deployment, and thus delay processing his workload, until his elected cloud is available. However, another option is to define an explicit preference for the clouds (e.g., specifying that the less costly clouds should be used whenever available) but immediately deploy occurrences wherever possible and then rebalance the implementation at a later time as needed. For example, because community clouds with limited assets are not always available, especially if demand is high, the environment can launch occurrences on lower-selected clouds, such as public cloud providers. As clouds with a higher-preference become available, the environment should rebalance, naturally downscaling, that is, terminating occurrences, in lower-preferred clouds, and up scaling, launching occurrences in higher-selected clouds. Up scaling is typically automated by auto-scaling services, such as Nimbus Phantom or Amazon's Auto Scaling Service, which allow users to define the number of occurrences that should be implemented in each cloud. Auto-scaling services then launch the occurrences and monitor them, replacing them if they crash or are dismissed too soon. However, downscaling is not typically automated and presents a number of unique challenges that must be forwarded. For instance, users must be able to clearly define their preferences for different clouds and policies must be generated. These policies should identify which clouds to conclude occurrences in, which occurrences to conclude, and how the occurrences should be concluded (e.g., wait until occurrences are idle or immediately conclude occurrences with running jobs, thus causing jobs to be restarted). Rebalancing policies should balance user requirements as well as workload and environment aspects to avoid introducing excessive workload overhead. To accomplish this, rebalancing performances must integrate with current workload resource managers and provide the functionality required by the policies, such as identifying the state of occurrences (e.g., busy or idle) or marking workers as "offline".

II. LITERATURE SURVEY

[1] Kai hwang et al proposed the cloud platforms for trust and security. Providers secures from the virtualised data center,user privacy and data integrity from protecting cloud environment. By suggesting a trust overlay network of the multiple data centres by reputation system between services providers and data owners for getting the data colouring and software watermarking techniques. By these techniques to safeguard multi way authentications, sign on in the cloud and access control for protective data from public and private clouds.

[2] Sheikh mahbub habib et al proposed the cost efficient for enterprises by using a variety of scable,dynamic and shared services in cloud computing techniques. By offering with the cloud Providers ,Service Level Agreement(SLAs) is not consistent among the customers. To support the customers, author proposed the Multi-Facted Trust Management (TM)system to identify trustworthy cloud providers in different set of attributes.

[3] Xing Pu et al proposed the performance in CPU and network workloads in the Xen virtual Machine Monitors(VMMs).Cloud environment conclude with four keys that are more effective for both the cloud Service Providers and cloud Consumers. First key running on a shared hardware can lead to the high overheads due to extensive switches and driver domain and VMM.Second key co-locating CPU on a hardware platform can lead to high CPU contention due to the demand for fast memory pages exchanges in I/O channel.Third key delivering the higher

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 4, April 2016

aggregate in running part of CPU intensive workloads and network intensive workloads. The fourth key identifying the factors on total demand of exchanged memory pages in I/O channel in VMM and driver domain.

[4] Nicola Dragoni et al proposed the service oriented computing(SOC) by using Web Services, such as evolution in the internet age. It moves from the collection of pages to collection of services. we review the field of trust based web services selection to provide a classification of current approaches and highlight in each field.Finally,a soft notion of trust lies in weakness part to get the new approach based on notion of trust in stronger part.

[5] M.Anan et al proposed under future internet with the new requirements to provide, monitor and enhance the multimedia streaming services .this environment was built in the global environment for Network innovations(GENI).This can be function under unstable and changing network conditions, recognise potential service problems without the need of human intercession.

III.PROPOSED SYSTEM

In Proposed System, Service operator-aware trust scheme is used for evaluating the trust. If users request the file with services to Auditor, Using Matching Algorithm Auditor matches the services and policy with the Provider services. Services with high trust scheme file only, Auditor will send to user. In Future, For more security Purpose, we used Homomorphic Encryption Algorithm, Homomorphic authenticators (also denoted as Homomorphic verifiable tags) are basic tools to construct data auditing mechanisms Besides unforgetability (only a TPA can generate valid signatures to user), the encrypted file is uploaded in cloud system with the secret key, if the secret key matches with the cloud services then only the user will going to access the particular file in the cloud systems and download the original file.

Adaptive Trust Evaluation: This module is the core of the trust-aware cloud computing system, and is the major focus of this paper. Using this module, the Auditor can dynamically sort high-performance resources by analyzing the historic resource observation in terms of providing highly trusted resources.

Trusted Resource Matchmaking And Distributing: In general, each cloud manager logs its service resources through the cloud Auditor. The service user negotiates with the service Auditor on the service-level agreement (SLA) details, they eventually prepare an SLA contract. According to this contract, the Auditor selects, and then presents highly trusted resources to users from the trusted resource pool.

Resource Register: It manages and indexes all the resources available from multiple cloud providers, and obtains observations from each particular cloud resource, acting as pricing interface for users, and updating the database when new data is available.

User Request And Policies: User request for the service to the admin with the policies so that client needs the specific services for production usability, reliability and security to the consumers. Then, Auditor will find the suitable resources given by the Cloud Manager and Produce to the Corresponding User using Matching algorithm. For more security, in future Enhancement we used Encryption algorithm for encrypting our file before uploading to cloud with a suitable secret key using Key Generation algorithm. So that malicious user cannot misuse our file without a proper key.

Auditing Cloud Resources: In this module, we use resources provided by the cloud provider after that, we can upload our file in cloud. Admin allocate Cloud with enough space to us, so that we can upload more files to cloud. If size is not enough to upload our file, then we can request to admin, to allocate extra space in cloud. Then, after allocated we can upload our file in cloud. Uploaded file will be in encrypted format for more security. So that other user cannot access our file. Using a secret key, we can download our file.

Allocate Size In Cloud: For every cloud user, Enough space will be allocated (E.g. : box , it's a free cloud , we can upload our file up to 10GB with more secure). Likewise, whatever the file we uploading in cloud, if space is not enough then, we can give request to admin, Extra space will be allocated to our cloud.

Homomorphic Encryption Algorithm: After auditing the message, Cloud Manager will encrypt the resources which are requested by the user, using Homomorphic Encryption Algorithm. The Encrypted resources or file sends to a user through Cloud Broker (TTP). Alternatively a key is generated along with the encrypted file content or resources. A secret key is send to the user mail.

Key Generation: Generating key for an Encrypted File using Key Generation Algorithm. After receiving the key from corresponding Cloud Manager, User can access the file or download the file using that secret key.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 4, April 2016

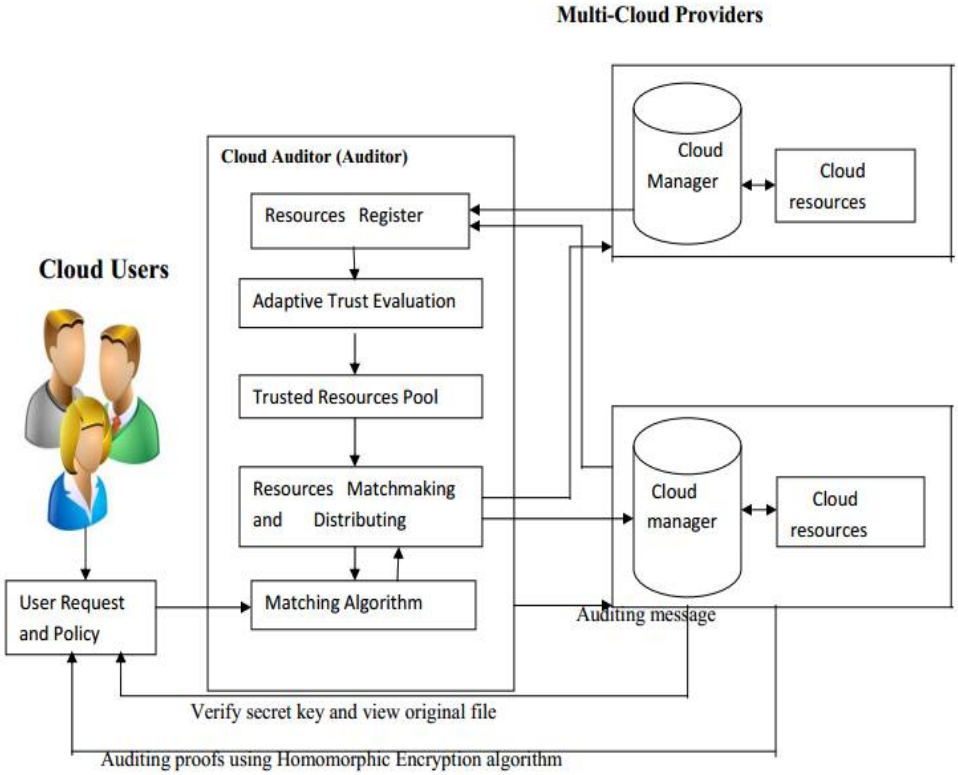

Figure 1. Architecture Diagram

IV. CONCLUSION

In this work, we propose SOTS for trustworthy resource matchmaking across multiple clouds. We have shown that SOTS yields very good results in many typical cases. However, there are still some open issues we can apply to the current scheme. First, we are interested in combining our trust scheme with reputation management to address concerns in users' feedback. A universal measurement and quantitative method to assess the security levels of a resource is another interesting direction. Evaluation of the proposed scheme in a larger-scale multiple cloud environment is also an important task to be addressed in future research.

REFERENCES

[1] H. Kim, H. Lee, W. Kim, and Y. Kim, "A trust evaluation model for QoS guarantee in cloud systems," Int. J. Grid Distribute. Computing vol. 3, no. 1, pp. 1–10, 2010.
[2] K. Hwang and D. Li, "Trusted cloud computing with secure resources and data coloring," IEEE Internet Computing, vol. 14, no. 5, pp. 14–22, Sep./Oct. 2010.
[3] S. M. Habib, S. Ries, and M. Muhlhauser, "Towards a trust management system for cloud computing," in Proc. IEEE 10th Int. Conf. Trust, Security Privacy Computing Communication, pp. 933–939, 2011.
[4] N. Dragoni, "A survey on trust-based web service provision approaches," in Proc. 3rd Int. Conf. Dependability, pp. 83–99, 2010.
[5] S. Clayman, A. Galis, C. Chapman, G. Toffetti, L. Rodero-Merino, L. M. Vaquero, K. Nagin and B. Rochwerger, "Monitoring service clouds in the future internet," in Future Internet Assembly. Amsterdam, the Netherlands: IOS Press, pp. 115–126, 2010.
[6] K. M. Khan and Q. Malluhi, "Establishing trust in cloud computing," IT Prof., vol. 12, no. 5, pp. 20–27, Sep./Oct. 2010.
[7] P. D. Manuel, S. Thamarai Selvi, and M. I. A. E. Barr, "Trust management system for grid and cloud resources," in Proc. 1st Int. Conf. Adv. Computing, pp.176–181, Dec. 2009.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 4, April 2016

[8] Z. Liang and W. Shi, "A reputation-driven scheduler for autonomic and sustainable resource sharing in grid computing," J. Parallel Distributed Computing, vol. 70, no. 2, pp. 111–125, 2010.
[9] S. A. de Chaves, R. B. Uriarte, C. B. Westphall, "Toward an architecture for monitoring private clouds, " IEEE Communication Mag., vol.49, no. 2, pp. 130–137, 2011.
[10] Xiaoyong Li, Huadong Ma, Member, IEEE, Feng Zhou, and Xiaolin Gui, "Service Operator-Aware Trust Scheme For Resource Matchmaking Across Multiple Clouds" IEEE transactions on parallel and distributed systems, vol. 26, no. 5, pg. no. 1419-1429, may 2015.
[11] Terry Bearly And Vijay Kumar, "Expanding Trust Beyond Reputation In Peer-To-Peer Systems" 15th International Workshop On Database And Expert Systems Applications (DEXA'04).
[12] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk Graziano Obertelli, Sunil Soman, Lamia Youseff, Dmitrii Zagorodnov, "The Eucalyptus Open-Source Cloud-Computing System" 9th IEEE/ACM international symposium on cluster computing and the grid pg. no. 125-131.
[13] Yiduo Mei, Ling Liu, Senior Member, IEEE, Xing Pu, Sankaran Sivathanu, and Xiaoshe Dong, "Performance Analysis of Network I/O Workloads in Virtualized Data Centers", IEEE transactions on services computing, vol. 6, no. 1, pg. no. 48-63, january-march 2013.
[14] Xing Pu, Ling Liu, Yiduo Mei, Sankaran Sivathanu, Younggyun Koh, Calton Pu, "Understanding Performance Interference of I/O Workload in Virtualized Cloud Environments", IEEE 3rd International Conference on Cloud Computing, 2010.
[15] Right Scale. [Online]. Available: http://www.rightscale.com/, 2013.