ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)
Vol. 2, Issue 2, Ver. 3 (April - June 2014)
International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014)

Using Secret Sharing Algorithm for Improving Security in Cloud Computing

[I]Swapnila S Mirajkar, [II]Santoshkumar Biradar
[I]Dept. of Computer Engineering, Dr D Y Patil College of Engineering, Ambi, Pune, India
[II]Associate Professor, Dept. Computer Engineering, Dr D Y Patil College of Engg., Ambi, Pune, India

Abstract
Cloud Computing is an emerging technology which has considerable potential as an alternative process for traditional silo computing. One can deploy applications more speedily across shared server storage resource pools than is possible with conventional enterprise solutions .Deploying modern web applications across a cloud framework enables a new level of agility that is very difficult to accomplish with traditional silo computing mode. Beside all the benefits cloud computing has big issue to be concern which is its security, reason is involvement of third party. Now days enterprises preferring "multi-clouds", or in other words, "interclouds" or "cloud-of-clouds" rather than single cloud provider. This paper focuses on multicloud security by using secret sharing algorithm.

Keywords
Computing, DepSky Architecture, Multi-clouds, Secrete Sharing, Security

I. Introduction

Adopting cloud computing can help organizations to their conduct core business activities more effectively since the managing and monitoring task for data centres is reduced. Again businesses can also save on power costs as the resources required are reduced. Then one may think, if cloud computing is such a great thing then why most businesses are not going for it, and as per the research the reason is poor security. The third party is involved called CSP (cloud Service Provider) to whom businesses have to provide their data including sensitive data. This paper surveys recent research related to security of single and multi-cloud and comes up with possible solutions for preservation of security. Though multicloud computing is relatively new concept, biggest security factors in cloud computing, such as data intrusion, data integrity, and service availability are handled in better way in multi-cloud than that of single cloud computing [4]. This project work promotes the use of multi-cloud architecture than that of single cloud architecture

II. Cloud Computing Background

NIST defined cloud computing as "a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction".[1]

A. Cloud Computing Components

The cloud computing environment comprises of five characteristics, three delivery models and four deployment models (see fig. 1). The five important characteristics of cloud computing are comprising first stratum are: location-independent resource pooling that is provider resources pooled to server multiple clients, on-demand self-service, rapid elasticity which is ability to quickly scale in/out service, broad network access, and measured service that is renting the services use per pay basis.

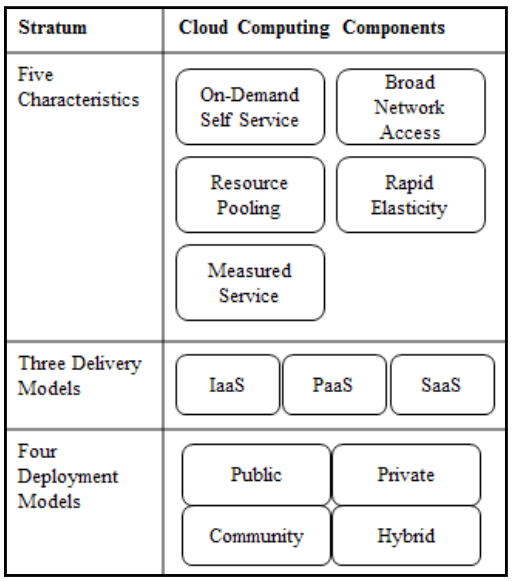

Fig. 1: Cloud Computing Environment

Three Cloud Delivery models are Iaas, PaaS and SaaS, comprises middle stratum of cloud computing environment.
In Software as a Service (SaaS), applications are there that are enabled for the cloud. It supports an architecture that can run multiple instances of itself which are location independent. This is nothing but a monthly subscription based pricing model and it is stateless. Examples of SaaS are MobileMe, Google docs, Zoho.
Platform as a Service (PaaS) includes platform on which developers can write their applications to be run on cloud environment. This platform normally has multiple application services available for quick deployment. Examples of PaaS are Google App Engine, Microsoft AZURE, Force.com.
Infrastructure as a Service (IaaS) used by consumer by providing storage, processing, networking, and other fundamental computing resources where the consumer is able to deploy and run software, which can include operating systems and applications. It is highly scaled redundant and shared computing Infrastructure approachable using internet technologies. Examples of this type of delivery model include Amazon EC2, Sun's cloud services, Terremark cloud offering etc.
Third stratum in the cloud computing environment consists of cloud deployment models which include public, private, community,

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

and hybrid clouds. A cloud architecture which can be accessed by multi-tenants and is available to the public is called a public cloud. Cloud which is available for a particular group is private cloud, while a community cloud is modified for a specific group of consumers. Hybrid cloud infrastructure is a combination of two or more clouds[4].

III. Literature Survey

Research illustrates that in 2009, 67% of the research on security in cloud computing covered the issue of a single cloud, whereas 33% of the research in the same year covered the issue of multi-clouds. In 2010, 80% of research focused on single clouds while only 20% or research was directed in the area of multi-clouds[8].

HAIL (High Availability and Integrity Layer) which is combination of Proofs and cryptography, presented in the year 2009 used to control multiple clouds. It ensures data integrity and service availability. But the limitation of HAIL is that it needs code execution in their servers and it does not deal with multiple versions of data[5].

RACS (Redundant Array of Cloud Storage) is a protocol for intercloud storage in the year of 2010.This Technique is similar to RAID and normally used by disks and file systems and replication offers better fault tolerance. But the problem is unable to cooperate with vendor lock-in and economic failure. Cachin [11] presented a design for intercloud storage named ICStore in 2010. ICstore is client centric distributed protocol which can handle data integrity issue but has poor performance in case of data intrusion and service availability. Same thing happened with encrypted cloud VPN[4].

Moving from single clouds multi-clouds is sensible and significant for many reasons. According to Cachin et al [5], "Services of single clouds are still subject to outage". Vukolic [15] accepts that the primary purpose of moving to interclouds is to amend what was offered in single clouds.

DepSky presented by Bessani [9] in 2011 is virtual storage cloud system comprising of a combination of different clouds to build a cloud-of-clouds. None of above problems are found in DepSky as it combines Byzantine fault tolerance protocol, secrete sharing and cryptography[16].

IV. Implementation Details

Primary objective of our work is to make the assurance that data is in secure and stable form. We are using DepSky system in our work which contains four commercial storage clouds (Amazon S3, Windows Azure, Nirvanix and Rackspace). It increases the system availability as data is not relayed on a single cloud, also avoids vendor lock-in issue since lack of dominant cloud. The DepSky system also reduces cost of than using single cloud, which is a significant advantage. DepSky uses a set of Byzantine quorum system protocols in order to implement the read and write operations in the system, so it needs only two communication round trips for each operation to deal with several clouds[4]. To make a shift towards more secure cloud computing, we are using multi-cloud computing than that of single cloud computing.

A. DepSky architecture

Bessani et al. [9] present a virtual storage cloud system called DepSky on which prototype of our system is based. As fig. 2 shows it is a multi-cloud architecture which consists of a combination of different storage clouds. There are no codes to be executed as clouds are used for data storage and maintenance. The DepSky system accosts the confidentiality and the availability of data in their storage system.

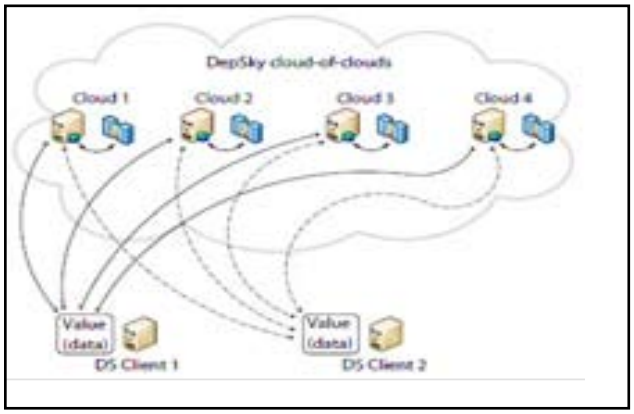

Fig. 2: DepSky Architecture

(i) System model of Depsky

It has readers, writers and cloud storage providers. Readers and writers are nothing but the client. As shown in fig 2, clouds 1-4 are cloud storage providers. A cloud storage provider does the tasks defined by readers and writers. Readers can fail irregularly, can crash and can present any behavior. But we cannot consider that writers can fail arbitrarily because of replicas. But replicas may be inconsistent, faulty writers may be able to write wrong values of data. To deal with this public key cryptography is used. Readers have access to public keys while common private key is shared by all writers of data unit. The DEPSKY algorithms are implemented as a software library in the clients

(ii) Data model of Depsky

DepSky library deals with different cloud interface providers as it is multi-cloud architecture. The data format DepSky should be acceptable by each cloud .Data model comprises of three abstraction levels: the conceptual data unit, a generic data unit, and the data unit implementation. The conceptual data unit contains a version number (to support updates on the object), verification data (usually a cryptographic hash of the data) and the data stored on the data unit object. Second level is generic data unit which has container for data, metadata and data object. Third abstraction level is data unit implementation in which container interpreted into the specific constructions supported by each cloud provider (Bucket, Folder, etc.). Four cloud providers are their which are Amazon S3, Windows Azure, Nirvanix and Rackspace.

B. Security using secret sharing Algorithm

In our system we aim to provide a framework to supply a secure cloud database that will assure to prevent security risks that the cloud computing community is facing. This framework will go for multi-clouds architecture and the Shamir's secret sharing algorithm to reduce the risk of data intrusion and the loss of service availability in the cloud and ensure data integrity.

The scope of this project is to upload and download a file from multi-cloud. If one cloud is failed, we can download the same file from other cloud as the data is replicated among multiple clouds. Files should be uploaded using Byzantine fault tolerance (BFT) algorithm. The Byzantine protocols involve a set of storage clouds (n) where $n = 3f + 1$, and f is maximum number of clouds which could be faulty. In addition, any subset of $(n - f)$ storage cloud creates byzantine quorum protocols [2, 9].

Each file is encrypted and secrete generated. Next step in

implementation is using Shamir's secrete sharing algorithm. In the Shamir's Secrete sharing scheme invented by Adi Shamir, secrete is divided into parts and then all parts are stored at different places (clouds in our case). So to reconstruct original secrete, one has to acquire all or some parts of the secrete from those different places[15]. Along with Shamir's secrete sharing scheme we are using Byzantine Fault Tolerance Protocol for deciding minimum number of parts of secrete require to generate original file. Message Digest concept MD5 is used for ensuring integrity of data at the time of upload phase as shown in figure 3. And at the time of download phase, reconstruction algorithm is applied to get original file and then verified with its message digest, if match found then file is considered to be integral.

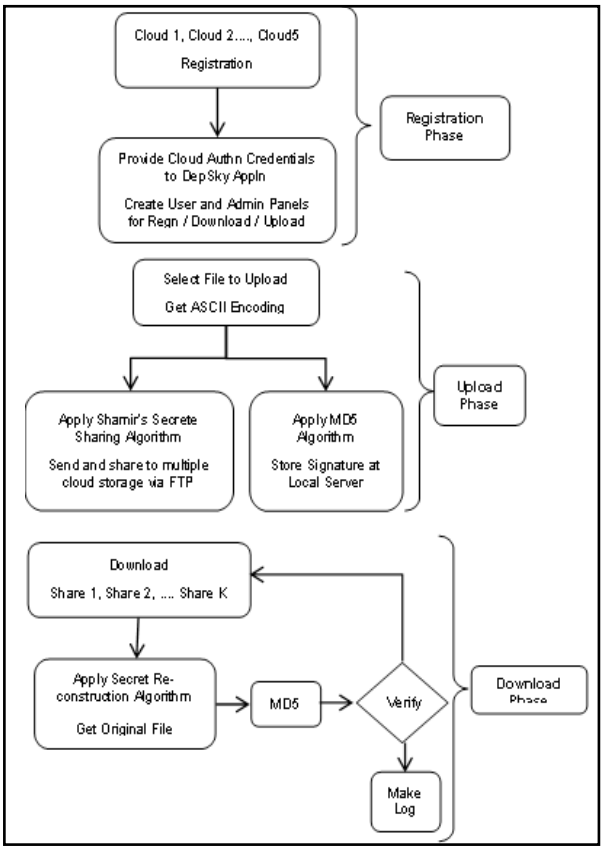

Fig. 3: Block Diagram of Proposed System

Shamir's secret sharing scheme is a threshold scheme based on polynomial function technique. It allows a Server S to distribute a secret value s to n clouds, such that some of parts required to reconstruct the secret. The protocol information theoretically secure, i.e., any fewer than t(threshold) clouds cannot gain any information about the secret by themselves.

V. Results

Data set for this system is trusted users file to upload or download to/from the cloud. The result set for this system is retrieval of same file in secure manner, preserving integrity of the file and reducing the risk of data intrusion.

System addresses major security issues that particularly affect single clouds, namely data integrity, data intrusion, and service availability. Multi-cloud architecture is used by which data availability is increased than that of single cloud since if that cloud crashes then that will affect to all stored data on that cloud but in case of multi-cloud architecture replication of data is done. The data intrusion issue is addressed by using Shamir's secret sharing algorithm. An intruder needs to retrieve at least three values to be able to find out the real value that we want to hide from the intruder. This depends on Shamir's secret sharing algorithm with a polynomial function technique which claims that even with full knowledge of (k – 1) clouds. Replicating data into multi-clouds by using a secrete sharing encryption reduces the risk of data intrusion and increase data integrity. Again data integrity is preserved by using MD5 algorithm. Comparison of MD5 form database and MD5 file from local system is done at each download. If both MD5 are equal integrity status is true then file is not corrupted or else failed then file is corrupted. In this way security is improved.

VI. Conclusion

Use of cloud computing has rapidly growing but with major issues to be taken care is cloud security[11]. Security is the reason due to which most of the businesses hesitating for moving their workload to cloud computing. Cloud clients fear to lose their private information if malicious insiders in the cloud. Also service availability is area to be concerned in single cloud, if that cloud fails[17]. In multi-cloud data is replicated so available even if on cloud fails. Integrity of data is also maintained in our proposed work by making use of MD5.We are making use of strongest cryptographic algorithm named Shamir's secrete sharing algorithm, which has number of advantages including security, client-side aggregation. It claims that security is maintained even when k or more servers collude. We have found that much research has been done to ensure the security of the single cloud and cloud storage whereas multi-clouds have received less attention in the area of security. We affirm the moving to multi-clouds due to its ability to decrease security risks that affect the cloud computing user. The key conclusion is that proposed work provides confidentiality, data integrity, improved availability and capacity to handle multiple requests at a time.

VII. Future Scope

In future scope, we aim to implement privacy-preserving public auditing system for data storage security instead of verifying file at each upload and download phase by using its message digest (MD5). Storage auditing will be performed by TPA without demanding the local copy of data. Homo-morphic authenticator and random mask technique is used. This process eliminates the burden of cloud user from the tedious and possibly expensive auditing task. Proposed schemes will shed light on economies of scale for Cloud Computing [7].

VIII. AcknowledgEment

I take this opportunity to thank all those, who have contributed to the completion of this work and helped me with valuable suggestions for improvement.

I express my deep gratitude to my guide, HOD and Principal for their valuable support, help & guidance during the project work & providing me with best facilities and atmosphere for the creative work, guidance and encouragement.

References

[1] (NIST), http://www.nist.gov/itl/cloud/

[2] C. Cachin, R. Haas and M. Vukolic, "Dependable storage in the Intercloud", Research Report RZ, 3783, 2010.

[3] H. Abu-Libdeh, L. Princehouse and H. Weatherspoon, "RACS: a case for cloud storage diversity", SoCC'10:Proc. 1st ACM symposium on Cloud computing, 2010, pp. 229-

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

240.

[4] Mohammed A. AlZain, Eric Pardede, Ben Soh, James A. Thom, "Cloud Computing Security: From Single to Multi-clouds," hicss, pp.5490-5499, 2012 45th Hawaii International Conference on System Sciences, 2012.

[5] K.D. Bowers, A. Juels and A. Oprea, "HAIL: A high-availability and integrity layer for cloud storage", CCS'09: Proc. 16th ACM Conf. on Computer and communications security, 2009, pp. 187-198.

[6] C. Cachin, I. Keidar and A. Shraer, "Trusting the cloud", ACM SIGACT News, 40, 2009, pp. 81-86.

[7] Cong Wang1, Qian Wang1, Kui Ren1, and Wenjing Lou2," Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing",IEEE INFOCOM 2010, San Diego, CA, March 2010

[8] K. Birman, G. Chockler & R.van Renesse, "Toward a cloud computing research agenda", SIGACT News, 40, 2009, pp. 68-80.

[9] A. Bessani, M. Correia, B. Quaresma, F. André and P. Sousa, "DepSky: dependable and secure storage in a cloud-of-clouds", EuroSys'11:Proc. 6thConf. on Computer systems, 2011, pp. 31-46.

[10] Shamir, A.: How to share a secret. Communications of the ACM, 612–613 (1979).

[11] Clavister, "Security in the cloud", Clavister White Paper, 2008.

[12] I. Abraham, G. Chockler, I. Keidar and D. Malkhi, "Byzantine disk paxos: optimal resilience with Byzantine shared memory", Distributed Computing, 18(5), 2006, pp. 387-408.

[13] S. Kamara and K. Lauter, "Cryptographic cloud storage", FC'10: Proc. 14thIntl.Conf. on Financial cryptograpy and data security,2010, pp. 136-149.

[14] G.R. Goodson, J.J. Wylie, G.R. Ganger and M.K. Reiter, "Efficient Byzantine-tolerant erasure-coded storage",DSN'04: Proc. Intl. Conf. on Dependable Systems and Networks,2004, pp.1-22.

[15] E. Grosse, J. Howie, J. Ransome, J. Reavis and S. Schmidt, "Cloud computing roundtable", IEEE Security & Privacy, 8(6), 2010, pp. 17-23.

[16] J. Hendricks, G.R. Ganger and M.K. Reiter, "Low-overhead byzantine fault-tolerant storage", SOSP'07: Proc. 21st ACM SIGOPS symposium on Operating systems principles, 2007, pp. 73-86.

[17] Midong Yhou, Zygmunt J. Hass, Securing Ad-Hoc Networks, IEEE Networks Special Issue on Network Security, November/December 1999.

[18] Data Integrity, from Wikipedia, the free encyclopedia, [Online] http://en.wikipedia.org/wiki/Data_integrity.

[19] Z. Hu, C. Peter, S. Johnson, Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks, in Proceedings of ACM MOBICOM'02, 1986.

[20] R. Sonzgiri, M. Dehill, V. P. Levine, C. Shields, E. M. Belding-Royer," A Secure Routing Protocol for Ad-Hoc Networks, in Proceedings of ICNP'02, 1978.

[21] Y. Hu, A. Perrig, D. Johnson,"Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad-Hoc Networks", in Proceedings of IEEE INFOCOM'03, 1965.