International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue-01
January 2018

Secret Sharing Algorithm Implementation on Single to Multi Cloud

V. Srikanth
MCA,M.Tech(CSE),MBA(HR),PGDBM(HR)

Abstract *Many organizations are gradually shifting towards the use of Cloud computing. Cloud computing is beneficial in terms of low cost and accessibility of data. Ensuring the security of cloud computing is a major factor in the cloud computing environment, as users often store sensitive information with cloud storage providers, but these providers may be untrusted. A set of possible solutions can be seen in the recent research related to single and multi-cloud security. This work aims in promoting the best of these research which when used together can provide a reliable source of dependency on multi clouds. The leverage of security in multi-clouds is furthermore enhanced through application of Secret Sharing Algorithm. A level of customization can be can be achieved by providing and deciding security. A better solution will be gradually shifting from single to multi clouds with better security solutions for the same.*

Index Terms - Cloud computing, DepSky, Secret Sharing algorithm, LaGrange's basis polynomial, multi-clouds.

I. INTRODUCTION

Cloud computing in its most righteous form can be called the next generation of computer technology. Cloud computing offers limitless flexibility, better reliability, enhanced collaboration, portability, unlimited storage but how secure is it after all? If the safety of data cannot be assured when it is stored in our private server how can we be sure of its safety over the cloud? Data stored in the cloud can be compromised or lost. So, we have to come up with a way to secure those files. We can encrypt them before storing them in the cloud, which sorts out the disclosure aspects.

However, what if the data is lost due to some catastrophe befalling the cloud service provider? We could store it on more than one cloud service and encrypt it before we send it off. Each of them will have the same file. A lot of research has been carried out for the same, and to a large extend has helped to strengthen cloud computing security.

II. NEED FOR CLOUD SECURITY

Hassan Takabi et al. [3], in their paper have depicted a complete survey on the issues related to cloud computing security. Cloud computing is both promising and scary. Despite of the attractive economic and technological advantages it has, businesses still think of the potential security threat before entrusting their data. Security is the most crucial aspect of everyday computing; this is very well applicable to cloud computing itself. There are many security concerns in cloud computing security; a few can be listed as follow:

A. Malicious Attacker

Hackers these days can breach the strongest security provisions and hijack confidential data. Malicious attacker can inject viruses or worms into the database, and destroy or corrupt the data that is of value to the company.

B. Service Hijacking

Service hijacking is nothing but gaining unauthorized services. It includes various techniques like fraud, phishing and software exploitation. This is considered to be one of the top most threats.

C. SQL Injection Attack

A SQL code is inserted into the model code. By doing this the invader can gain access to a database and to other unauthorized information. SQL cross scripting is a well-known tool for hackers, wherein on use of special characters the hacker can modify rows and columns.

D. Confidentiality

Confidentiality is preventing the improper disclosure of information. Preserving confidentiality is one of the major issues faced by cloud systems, since the information is stored at a remote location that the Service Provider has full access to. Therefore, there has been some method of preserving the confidentiality of data stored in the cloud. The main method used to preserve data confidentiality is data encryption; however encryption brings about its own issues, some of which are discussed later.

III. THE MULTI-CLOUDS STRATEGY

Multi-cloud strategy is the use of two or more cloud to minimize the risk of service availability failure, Loss and corruption of data, loss of privacy, vender lock-in and the possibility of malicious insiders in the single cloud. The service unavailability can occur due to breakdown of hardware, software or system infrastructure. A multi-cloud strategy can also improve overall enterprise performance by avoiding "vendor lock-in" and using different infrastructures to meet the needs of diverse partners and customers. The cost of using multiple clouds will be higher than that of single clouds. Thus unless and until there is a design which can make use of multi-clouds without increasing cost, the implementation will be highly impractical. A high level design of inter-clouds is presented in [17].

IV. DEPSKY SYSTEM

DepSky is one such architecture design that overcomes all the limitations of multi-clouds by eliminating the requirement of code execution in the servers (i.e., storage clouds). It is still efficient as it requires only two communication round-trips for each operation. Also, it deals with data confidentiality and reduces the amount of data stored in each cloud. It uses an efficient set of Byzantine quorum system protocols, cryptography, secret sharing, erasure codes and the diversity that comes from using several clouds. Several areas of cloud computing that will benefit from DepSky are discussed in [5].

A. DepSky Architecture

The DepSky system model contains three parts: readers, writers, and four cloud storage providers, where readers and writers are the client‘s tasks. A. Bessani et al. in [5], explain the difference between readers and writers for cloud storage. Readers can fail arbitrarily (for example, they can fail by crashing, they can fail from time to time and then display any behavior) whereas, writers only fail by crashing. The DepSky Architecture is as presented in fig. 1.

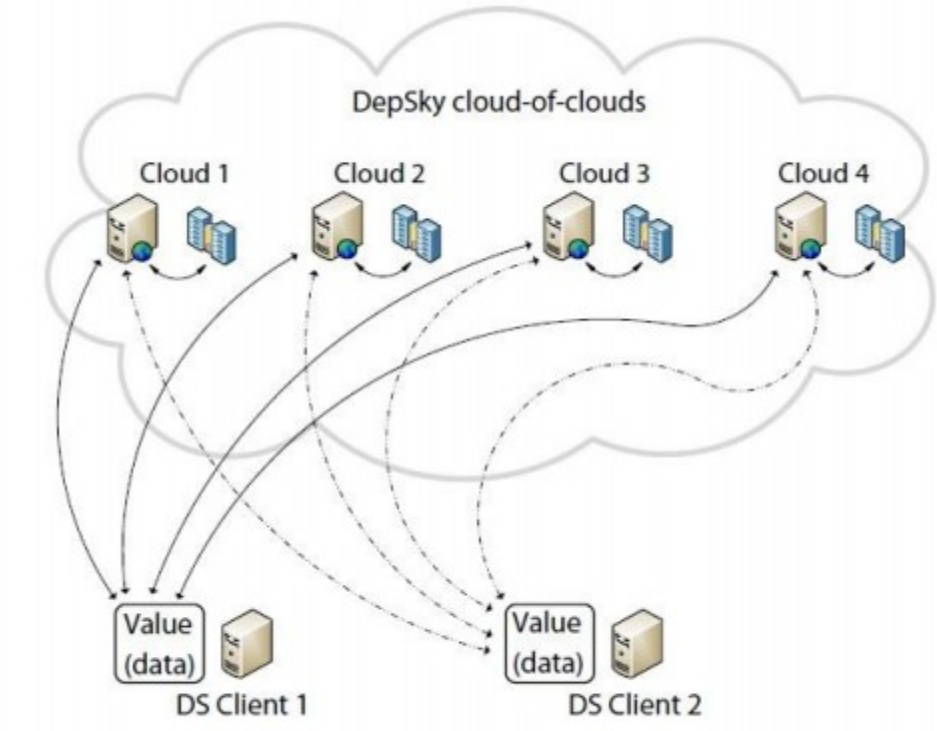

Figure 1. DepSky Architecture

B. Protocol Design Rationale There are many protocols for implementing Byzantine fault-tolerant (BFT), but most of them require that the servers execute some code. This functionality is not available on storage clouds. A key difference between the DepSky protocols and these classical BFT protocols is that the metadata and data are written in separate quorum accesses. The DepSky protocols provide consistency proportional semantics, i.e., the semantics of a data unit is as strong as the underling clouds allow, from eventual to regular consistency semantics. To ensure confidentiality of stored data on the clouds without requiring a key distribution service, we employ a secret sharing scheme.

V. SECRET SHARING STRATEGY

Simply storing the data on multiple clouds solves the problem of data availability, but what about security? If there are multiple copies of the data, it will just open more doors for the intruder to hack in. Thus there needs to be a way in which we can make sure that the data over multiple clouds is safe, or safer than it was in a single cloud. This is when we can apply the Secret sharing algorithm presented by, Adi Shamir elaborated in [4].

Invented in 1979, the algorithm has occupied a huge place in the area of cryptography. A somewhat similar algorithm was discovered by George Blakley, its use is depicted in [19]. Its performance is more or less the same, but the mathematical evolution is more complicated. That‘s where the beauty of Shamir‘s secret sharing algorithm lies – in its simplicity of implementation.

A. Basic Principle

The basic idea behind secret sharing algorithm is, when we want to secure a certain data _D‘, we divide it into n parts say D1, D2… Dn in such a way that:

- The Knowledge of any k or few Di pieces makes D easily computable.
- The Knowledge of any k-1 or fewer Di pieces leaves D completely undetermined (in the sense that all its possible values are equally likely).

This scheme is called (k,n) threshold scheme. The value of factor K can be decided depending on the level of security we desire. For example, if the data is of top most priority such as bank account password or transaction ids we can keep k=n. In such a case all participants will be required to reconstruct the secret original data.

B. Mathematical Implementation

The mathematical implementation of Secret Sharing algorithm can be understood with the help of a simple example as given by Md Kausar et al. in [6]. The generalized idea is as follow:

- We choose at random (k-1) coefficients i.e. a1…ak-1
- We divide our secret data ‘S‘ by picking a random degree polynomial.

$$q(x) = a_0+a_1x+a_2x +\ldots+a_{k-1}x^{k-1}$$

Where a_0=_S‘ (i.e the data).

Now if we wish to divide the data into n parts, we will substitute ‘n‘ different values of x in the polynomial q(x) and obtain ‘n‘ such sets of (x, y), here y is nothing but our polynomial q(x).

The essential idea of Adi Shamir's threshold scheme is that 2 points are sufficient to define a line, 3 points are sufficient to define a parabola, 4 points to define a cubic curve and so forth. That is, it takes “k”‖ points to define a polynomial of degree “k-1”.

Select ‘k‘ such sets, any k combination of the available n parts will generate the same result. The value in these sets are meaningless alone, it is only when ‘k‘ sets are brought in together and further worked upon that we get our secret back. These ‘k’‖ instances of original polynomial are processed using Lagrange polynomials.

The Lagrange basis is:

$$l_0 = \frac{x - x_1}{x_0 - x_1} \cdot \frac{x - x_2}{x_0 - x_2}$$

$$l_1 = \frac{x - x_0}{x_1 - x_0} \cdot \frac{x - x_2}{x_1 - x_2}$$

$$l_2 = \frac{x - x_0}{x_2 - x_0} \cdot \frac{x - x_1}{x_2 - x_1}$$

Substitute the values of x from the selected ‘k‘ sets into the Lagrange basis and we obtain ‘k‘ fractional equations for the same. Finally on taking summation of the equations obtained from Lagrange basis and y form the selected ‘k‘ sets, we get back our original polynomial. The summation can be represented mathematically as:

$$f(x)=\sum_{j=0}^{2} y_j.l_{j(x)}$$

The above explanation helps in understanding the working of the secret sharing algorithm. When done manually the entire calculation can be done in minutes, while on implementation, as the microprocessor technology has elevated its level to a new high, thousands of such calculations can be done in seconds.

C. Related Work

Shamir's Secret sharing algorithm is a powerful base for securing data and is often seen as a replacement to encryption. Mohammed A. et al. in [10] have given specific elaboration on how the algorithm is well suited for database using the classic example of an ERP system. Taking it a step further he also proposes it as a future work in [1]. The scope of its application is also vast and the methods to do so have been exhibited by Dnyaneshwar Supe et al. in [21].

D. Properties

The secret sharing algorithm possesses some dynamic properties that make it further more powerful, these properties, as descried by Adi shamir in [4] are as follows:

- The size of each piece does not exceed the size of the original data.
- When k is kept fixed, D pieces can be dynamically added or deleted without affecting the other Di pieces.
- It is easy to change the Di pieces without changing the original data D - all we need is a new polynomial q(x) with the same free term. This can enhance security.
- By using tuples of polynomial values as Di pieces, we can get a hierarchical scheme in which the number of pieces needed to determine D depends on their importance.

VI. FUTURE WORK

The combination of multi-clouds and secret sharing algorithm is promising, but as of yet it deals with many uncertainties. The current work ensures implementation of only text and relational database. Inclusion of images and audio handling capability might increase the size and complexity of system. The number of data instances depends on user's affiliation with cloud service. The maximum size of data is again one of the factors that we have to deal with in practical implementation. An earnest attempt has been made by Alfonso Cevallos Manzano in [20].

So for future work we will try an overcome all these limitations or find an alternative for the same.

VII. CONCLUSIONS

It is fairly lucid that storing the data over multi clouds is efficient, and when this data is encrypted using Shamir's secret sharing algorithm it is guaranteed to be more secure and harder to compromise. The worst case failing probability of the system is low and the time complexity of the system is reduced. The purpose of this work is to survey the possibilities of a system that would take the best of both multi-clouds and secret sharing algorithm, to address the present security issues and present a possible solution.

REFERENCES

[1] Mohammed A. AlZain, Eric Pardede, Ben Soh, James A. Thom, ―Cloud Computing Security: From Single to Multi-Clouds‖, 45th Hawaii International Conference on System Sciences, 2012.

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

[3] Hassan Takabi, James B.D., Joshi, Gail-Joon, Ahn, ―Security and Privacy Challenges in Cloud Computing Environments‖, University of Pittsberg, October 2010.

[4] A. Shamir, "How to share a secret", Communications of the ACM, 22(11), 1979, pp. 612-613.

[5] 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.

[6] Md Kausar Alam, Sharmila Banu K, ―An Approach Secret Sharing Algorithm in Cloud Computing Security over Single to Multi Clouds,‖ International Journal of Scientific and Research Publications, vol. 3, issue 4, April 2013

[7] A. Juels and B.S. Kaliski Jr, "PORs: Proofs of retrievability for large files", CCS '07: Proc. 14th ACM Conf. on Computer and communications security, 2007, pp. 584-597.

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

[9] 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. 229-240.

[10] M.A. AlZain and E. Pardede, "Using Multi Shares for Ensuring Privacy in Database-as-a-Service", 44th Hawaii Intl. Conf. on System Sciences (HICSS), 2011, pp. 1-9.

[11] R.C. Merkle, "Protocols for public key cryptosystems", IEEE Symposium on Security and Privacy, 1980, pp. 122-134.

[12] F. Rocha and M. Correia, "Lucy in the Sky without Diamonds: Stealing Confidential Data in the Cloud", Proc. 1stIntl. Workshop of Dependability of Clouds, Data Centers and Virtual Computing Environments, 2011, pp. 1-6.

[13] M. Van Dijk and A. Juels, "On the impossibility of cryptography alone for privacy-preserving cloud computing", HotSec'10: Proc. 5th USENIX Conf. on Hot topics in security, 2010, pp.1-8.

[14] Dawson, E.; Donovan, D. (1994), "The breadth of Shamir's secret-sharing scheme", Computers & Security 13: 69–78.

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

[16] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing", Journal of Network and Computer Applications, 34(1), 2011, pp 1-11.

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

[18] R.C. Merkle, "Protocols for public key cryptosystems", IEEE Symposium on Security and Privacy, 1980, pp. 122-134.

[19] Ilker Nadi Bozkurt, Kamer Kaya, Ali Aydın Selcuk, Threshold Cryptography Based on Blakley Secret Sharing.

[20] Alfonso Cevallos Manzano, Reducing the Share Size in Robust Secret Sharing, Master's Thesis, defended on October 18, 2011.

[21] Dnyaneshwar Supe, Amit Srivastav, Dr. Rajesh S. Prasad, ―Review of methods for secret sharing in cloud computing,‖ International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 2, Issue 1, January 2013.

[22] S. Chen, B. Mulgrew, and P. M. Grant, ―A clustering technique for digital communications channel equalization using radial basis function networks,‖ IEEE Trans. on Neural Networks, vol. 4, pp. 570-578, July 1993.

[23] J. U. Duncombe, ―Infrared navigation—Part I: An assessment of feasibility,‖ IEEE Trans. Electron Devices, vol. ED-11, pp. 34-39, Jan. 1959.

[24] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, and M. Miller, ―Rotation, scale, and translation resilient public watermarking for images,‖ IEEE Trans. Image Process., vol. 10, no. 5, pp. 767-782, May 2001.