An Approach for Security and Privacy Enhancing by Making Use of Distinct Clouds

Sasikumar Gurumurthy, T. Niranjan Babu, G. Siva Shankar

Abstract— Security challenges are the major concern when we considering the acceptence of cloud service. A lot of research activities regarding to cloud security resulting in an amount of application and targeting the cloud security threats. The cloud concept comes with a new set of unique features, techniques and architectures. This paper is related to security and privacy enhancing by making use of multiple distinct clouds. Based on the different cloud architecture, the security and privacy capabilities can be approximated. Cloud computing refers to applications and services that run on a distributed network using virtualized resources and accessed by common internet protocols. In this paper, we are introduced different clouds for encryption, decryption and storage process.

Keywords---Cloud, Security, Privacy, Multicloud, data partitioning.

I. INTRODUCTION

Security challenges are still amongst the biggest obstacles when considering the adoption of cloud services. This triggered a lot of research activities, resulting in a quantity of proposals targeting the various cloud security threats. Alongside with these security issues the cloud paradigm comes with a new set of unique features which open the path towards novel security approaches, techniques and architectures. This paper provides a survey on the achievable security merits by making use of multiple distinct clouds simultaneously. Various distinct architectures are introduced and discussed according to their security and privacy capabilities and prospects. Cloud computing offers dynamically scalable resources provisioned as a service over the Internet. The third party, on-demand, self-service, pay-per-use and seamlessly scalable computing resources and services offered by the cloud paradigm promise to reduce capital as well as operational expenditures for hardware and software. Clouds can be categorized taking the physical location from the viewpoint of the user into account. A Public Cloud is offered by third-party service providers and involves resources outside the user’s premises. In case the cloud system is installed on the user’s premise usually in the own data center this setup is called Private Cloud. A hybrid approach is denoted as Hybrid Cloud. This paper will concentrate on public clouds, since these services demand for the highest security requirements but also as this paper will start arguing includes high potential for security prospects.

Manuscript Received on April, 2014.

Prof. Sasikumar Gurumurthy, School of Computing Science and Engineering, VIT University, Vellore, Tamil Nadu, India.

T. Niranjan Babu, School of Computing Science and Engineering, VIT University, Vellore, Tamil Nadu, India.

G.Siva Shankar, School of Computing Science and Engineering, VIT University, Vellore, Tamil Nadu, India.

II. MOTIVATION

We have developed this project by using multi cloud, those cloud details maintained by admin. In this system we are using different cloud for encryption, decryption and storage process. Once you register yourself in the cloud, you can able to upload and download the files from anywhere, where there is an Internet connection. More security because encrypted file will split into two files store in the different clouds. You can easily access your files anywhere as long as you have an internet connection and Unlimited Storage.

III. PROPOSED WORK

In proposed system, using the n clouds approach (and its integrity guarantees) in combination with sound data encryption (and its confidentiality guarantees) may result in approaches that suffice for both technical and regulatory requirements. We identified the fields of homomorphic encryption and secure multiparty computation protocols to be highly promising in terms of both technical security and regulatory compliance. As of now, the limitations of these approaches only stem from their narrow applicability and high complexity in use. However, given their excellent properties in terms of security and compliance in multi-cloud architectures, we envision these fields to become the major building blocks for future generations of the multi-cloud computing paradigm.Homomorphic encryption and secure multi-party computation both use cryptographic means to secure the data while it is processed. In homomorphic encryption, the user encrypts the data with his public key and uploads the cipher texts to the Cloud. The cloud can independently compute on the encrypted data to obtain an encrypted result which only the user can decrypt. Therefore, in our scenario, homomorphism encryption uses an asymmetric fragmentation, where the user (or a small trusted private cloud) manages the keys and performs the encryption and decryption operations, while the massive computation on encrypted data in done by an untrusted public cloud. By this approach, application parts are distributed to different clouds in such a way, that every single cloud has only a partial view on the application and gains only limited knowledge. Therefore, this method can also hide parts of the application logic from the clouds. For application splitting, a first approach is using the existing sequential or parallel logic separation. Thus, depending on the application, every cloud provider just performs Sub tasks on a subset of data.

IV. SECURITY PROSPECTS

In OSN multiple users require multiple authorization facilities for a single source. The three important scenarios are profile sharing, content sharing and relationship sharing. often lack of collaborative multiple access control has the greatest impact on these scenarios.

A. Partition of Application Data into Fragments

It allows distributing fine-grained fragments of the data to distinct clouds. None of the involved cloud providers gains access to all the data, which safeguards the data's confidentiality. The most common forms of data storage are files and databases. Files typically contain unstructured data (e.g., pictures, text documents) and do not allow for easily splitting or exchanging parts of the data. This kind of data can only be partitioned using cryptographic methods. Databases contain data in structured form organized in columns and rows. Here, data partitioning can be performed by distributing different parts of the database (tables, rows, columns) to different cloud providers. Finally, files can also contain structured data (e.g., XML data). Here, the data can be splitted using similar approaches like for databases. XML data, for example, can be partitioned on XML element level. However, such operations are very costly. Thus, this data are commonly rather treated using cryptographic data splitting.

B. Partition of Application logic into Fragments

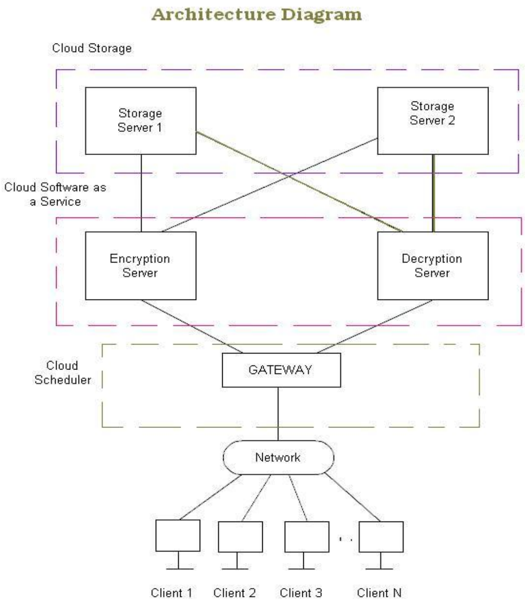
It allows distributing the application logic to distinct clouds. This has two benefits. First, no cloud provider learns the complete application logic. Second, no cloud provider learns the overall calculated result of the application. Thus, this leads to data and application confidentiality. This architecture variant targets the confidentiality of data and processing logic. It gives an answer to the following question: How can a cloud user avoid fully revealing the data or processing logic to the cloud provider? The data should not only be protected while in the persistent storage, but in particular when it is processed. The idea of this architecture is that the application logic needs to be partitioned into fine-grained parts and these parts are distributed to distinct clouds. This approach can be instantiated in different ways depending on how the partitioning is performed. The clouds participating in the fragmented applications can be symmetric or asymmetric in terms of computing power and trust. Two concepts are common. The first involves a trusted private cloud that takes a small critical share of the computation, and a untrusted public cloud that takes most of the computational load. The second distributes the computation among several untrusted public clouds, with the assumption that these clouds will not collude to break the security.

V. HOMOMORPHIC ENCRYPTION

Homomorphic encryption and secure multiparty computation both use cryptographic means to secure the data while it is processed. In homomorphic encryption, the user encrypts the data with his public key and uploads the ciphertexts to the Cloud. The cloud can independently compute on the encrypted data to obtain an encrypted result, which only the user can decrypt. Therefore, in our scenario, homomorphic encryption uses an asymmetric fragmentation, where the user (or a small trusted private cloud) manages the keys and performs the encryption and decryption operations, while the massive computation on encrypted data is done by an untrusted public cloud. In the case of homomorphic encryption, the cloud has the main share of work, as it operates on the encrypted inputs to compute the encrypted output. However, the algorithms are far from being practical, so the vision of clouds based on homomorphic encryption seems unreal for the foreseeable future. In addition, the applicability is limited, as for services that go beyond the outsourcing of computation, intermediate or final results need to be decrypted. This requires either interaction with the entity that holds the key (e.g., a private cloud) or the key is shared among several clouds who then assist in decrypting values that are needed in clear with a threshold encryption scheme. Fully homomorphic encryption has numerous applications. For example, it enables private queries to a search engine-the user submits an encrypted query and the search engine computes a succinct encrypted answer without ever looking at the query in the clear. It also enables searching on encrypted data- a user stores encrypted files on a remote file server and can later have the server retrieve only files that (when decrypted) satisfy some boolean constraint, even though the server cannot decrypt the files on its own. More broadly, fully homomorphic encryption improves the efficiency of secure multiparty computation.

VI. SECURE MULTIPARTY COMPUTATION

The idea of secure multiparty computation was first presented and as a solution to the millionaires problem: Two millionaires want to find out who is richer without disclosing any further information about their wealth. Two main variants of secure multiparty computation are known: Based on linear secret sharing or garbled circuits. The clouds will jointly compute the function of interest on these shares, communicating with each other when necessary. In the end, the clouds hold shares of the result which is sent back to the user who can reconstruct the result. At least three clouds are necessary for this scheme and no two of them should collude. The approach of garbled circuits works as follows: One cloud generates a circuit that is able to compute the desired function and encrypts this circuit producing a garbled circuit, which is however still executable. Then, this cloud assists the users in encrypting their inputs accordingly. Another cloud needs now to be present to evaluate the circuit with the user's inputs. Thus, this scheme requires in general only two clouds. Although the ideas of multiparty computation are old, it is ongoing research to reduce the overhead by multiparty computation.



VII. SYSTEM ARCHITECTURE

Above Figure shows the system consists of two storage servers namely storage server1 and storage server2 respectively. For encrypption and decryption the system maintains two individual servers. The two storage servers were controlled by cloud storage. N number of clients may participate in this network and these are seperated with a gateway. The gateway can be controlled by cloud schedular.

VIII. PRIVACY ISSUES IN THE CLOUD

Cloud computing consists of three main delivery models where systems provide (1) *Software as a Service* (SaaS), (2) *Platform as a Service* (PaaS), and (3) *Infrastructure as a Service* (IaaS). In SaaS, the service provider supplies the hardware infrastructure, the software product and interacts with the user through a front-end portal. In PaaS, the service provider provides a set of software and product development tools for the host to develop the application. In IaaS, the service provider provides virtual servers with unique IP addresses and blocks of storage on demand. Systems communicating over the cloud need to consider privacy and integrity issues where the information must be private before (i.e., *creation*), during (i.e., *transport*), and after (i.e., *storage*) transmission over the cloud. A user needs to trust remote systems in handling his/her sensitive information. There is no guarantee that a cloud provider will keep the information private and preserve its integrity unless explicitly provided.

IX. ALGORITHM

In this project we have used RNS (Residue number system) Algorithm. This algorithm having the following

Step 1: First we have to select two random numbers.

Step 2: Generate the key by using two random numbers.

$M = P1 * P2 = 143$

$A1 = M / P1 = 143 / 11 = 13$

$A2 = M / P2 = 143 / 13 = 11$

T Value is, it can be anything

$T1 = ((A1 * T) \bmod P1) == 1$

$T1 = 6$

$T2 = ((A2 * T) \bmod P2) == 1$

$T2 = 6$

Step3: Encrypt the file with help of key.

$R1 = N \% P1 = 80 \% 11 = 3$

$R2 = N \% P2 = 80 \% 13 = 2$

Step4: Then Decrypt the file

$E = [(A1 * T1 * R1) + (A2 * T2 * R2)] \bmod M$

$E = [(13 * 6 * 3) + (11 * 6 * 2)] \bmod 143$

$E = [234 + 132] \bmod 143$

$E = [366] \bmod 143$

$E = 80$

The above algorithm is known as RNS namely Residue Number System algorithm. By using this algorithm the encryption and decryption processes happened on the given cloud data storage. The cloud storage consists of these encrypted and decrypted data as well as the downloaded data at last the cloud schedular handles this process and cloud data may download from cloud storage to whereever we want to store.

These encrypted data will split into two different clouds and as well as the decrypted data also splits into two different clouds. Given the vast amount of specific approaches for realizing each of the presented multicloud architectures, it is not feasible to perform a general assessment adequately covering all of them. Furthermore, many approaches are only suitable in very special circumstances, rendering each comparison to other approaches of the same domain inadequate.

Therein, the security considerations indicate an approach’s general improvements and aggravations in terms of integrity, confidentiality, and availability of application logic or data, respectively. For instance, the n clouds approach is highly beneficial in terms of integrity (every deviation in execution that occurs at a single cloud provider only can immediately be detected and corrected), but quite disadvantageous in terms of confidentiality (because every cloud provider learns everything about the application logic and data). The feasibility aspect covers issues of applicability, business readyness, and ease of use.

Herein, applicability means the degree of flexibility of using one approach to solve different types of problems. Business-readyness evaluates how far the research on a multicloud approach has progressed and if it is ready for real-world applications.

X. CONCLUSION

The use of multiple cloud providers for gaining security and privacy benefits is non-trivial. As the approaches investigated in this paper clearly show, there is no single optimal approach to foster both security and legal compliance in an Omni-applicable manner. Moreover, the approaches that are favorable from a technical perspective appear less appealing from a regulatory point of view, and vice versa. A few approaches that score sufficiently in both these dimensions lack versatility and ease of use, hence can be used in very rare circumstances only. However, two major indications for improvement can be taken from the examinations performed in this paper. First of all, given that for each type of security problem there exists at least one technical solution approach, a highly interesting field for future research lies in combining the approaches presented here.

For instance, using the n clouds approach (and its integrity guarantees) in combination with sound data encryption (and its confidentiality guarantees) may result in approaches that suffice for both technical and regulatory requirements.

REFERENCES

1. P. Mell and T. Grance, “The NIST Definition of Cloud Computing, Version 15,” Nat’l Inst. of Standards and Technology, Information Technology Laboratory, vol. 53, p. 50, http://csrc.nist.gov/groups/ SNS/cloud-computing/, 2010.
2. F. Gens, “IT Cloud Services User Survey, pt.2: Top Benefits & Challenges,” blog, http://blogs.idc.com/ie/?p=210, 2008.
3. Gartner, “Gartner Says Cloud Adoption in Europe Will Trail U.S. by at Least Two Years,” http://www.gartner.com/it/page. jsp?id=2032215, May 2012.
4. J.-M. Bohli, M. Jensen, N. Gruschka, J. Schwenk, and L.L.L. Iacono, “Security Prospects through Cloud Computing by Adopting Multiple Clouds,” Proc. IEEE Fourth Int’l Conf. Cloud Computing (CLOUD), 2011.
5. D. Hubbard and M. Sutton, “Top Threats to Cloud Computing V1.0,” Cloud Security Alliance, http://www. cloudsecurityalliance.org/topthreats, 2010.
6. M. Jensen, J. Schwenk, N. Gruschka, and L. Lo Iacono, “On Technical Security Issues in Cloud Computing,” Proc. IEEE Int’l Conf. Cloud Computing (CLOUD-II), 2009.
7. T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, “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, 2009.
8. N. Gruschka and L. Lo Iacono, “Vulnerable Cloud: SOAP Message Security Validation Revisited,” Proc. IEEE Int’l Conf. Web Services (ICWS ’09), 2009.
9. M. McIntosh and P. Austel, “XML Signature Element Wrapping Attacks and Countermeasures,” Proc. Workshop Secure Web Services, pp. 20-27, 2005.
10. J. Kincaid, “Google Privacy Blunder Shares Your Docs without Permission,” TechCrunch, http://techcrunch.com/2009/03/07/ huge-google-privacy-blunder-shares-your-docs-withoutpermission/, 2009.

AUTHORS PROFILE

Sasikumar Gurumurthy is an assistant professor (Sr.) in SCSE, VIT University, Vellore, Tamil Nadu, India. He received B.E Degree in Computer Science and Engineering from Kamaraj University, Madurai in 2003 and M.E Degree in Computer Science & Engineering from Anna University, Chennai in 2005. He has published more than 70 technical papers in international journals proceedings of international conferences. He is having more than 8 years of teaching Experience. He is a member of international professional associations like CSI, IAENG, AIRCC, MHRO and is a reviewer of around 2 international journals. He is currently doing his Phd in VIT University. His current fields of research interest include image processing, signal processing and bio-medical engineering.

T. Niranjan Babu is a M.Tech, student in School of Computing Science and Engineering, VIT University, Vellore, Tamil Nadu.

G. Siva Shankar is a M.Tech, student in School of Computing Science and Engineering, VIT University, Vellore, Tamil Nadu.