ISSN:0975-9646

Towards Secure the Multi-Cloud using Homomorphic Encryption Scheme

Suresh Babu Bodduluri[1], P.Savaridassan[2]

1 Research Scholar, M.Tech, Database Systems,
Dept of Information Technology, SRM University, Chennai, India,

2 Assistant Professor, Dept of Information Technology,
SRM University, Chennai, India

Abstract-**Cloud computing is a model to acces shared pool of configurable computing resources which include servers, storage, applications, and also services to network effort or service interaction provider . This cloud model promotes availability and is composed of five essential fetures, in that service models are three, and four are deployment models. However, as the shape of the cloud computing is emerging and developing rapidly both conceptually and legal in reality economic, quality service, interoperability to exchange, security and privacy issues still pose significant challenges. we describe various service and deployment models of cloud computing and identify grater important challenges. In specific, we examine three crucial challenges: regulatory, security and privacy issues in Multi cloud computing. We use AES scheme for encryption of the data. As with any MAC (Medium Access Control), it is used to simultaneously verify both the data integrity and the authentication of a message. Erasure coding is used for rebuilding lost encoded fragments from existing encoded fragments.**
Objective-**The scope of our project is it's 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**

Keywords- **Cloud, Security, Privacy, Multiple clouds, Application Partitioning, Tier Partitioning, Data Partitioning, Multi-party Computation,encapsulation, Polymorphism.**

1. INTRODUCTION

A *Public Cloud* is offered by third-party service providers and involves resources outside the user's premises. Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The agencies may employ one model or a combination of different models for efficient and optimized delivery of applications and business services. These four delivery models are: (i) Private cloud in which cloud services are provided solely for an organization and are managed by the organization or a third party.

The main threat on data privacy roots in the cloud itself. When users outsource their private data onto the cloud, the cloud service providers are able to control and monitor the data and the communication [8] between users and the cloud at will, lawfully or unlawfully,. Instances such as the secret NSA program, working with AT&T and Verizon, which recorded over 10 million phone calls between American citizens, cause uncertainty among privacy advocates [2], and the greater powers it gives to telecommunication companies to monitor user activity. To ensure privacy, users usually encrypt the data before outsourcing it onto cloud, which brings great challenges to effective data utilization. However, even if the encrypted [4] data utilization is possible, users still need to communicate with the cloud and allow the cloud operates on the encrypted data, which potentially causes leakage of sensitive information [7]. Furthermore, in cloud computing, data owners may share their outsourced data with a number of users, who might want to only retrieve the data files they are interested in. One of the most popular ways to do so is through keyword-based retrieval. Keyword-based retrieval is a typical data service and widely applied in plaintext scenarios, in which users retrieve relevant files in a file set based on keywords. However, it turns out to be a difficult task in cipher text scenario due to limited operations on encrypted data. Besides, in order to improve feasibility and save on the expense in the cloud paradigm, it is preferred to get the retrieval result with the most relevant files that match users' interest instead of all the files, which indicates that the files should be ranked in the order of relevance by users' interest and only the files with the highest relevance's are sent back to users.

System Architect

Cloud computing consist of two components the front end and the back end. The front end of the cloud computing system comprises the client's device and some applications that are needed for accessing the cloud computing system. Back end refers to the cloud itself which encompass various computers, data storage systems and servers. The whole system is administrated via a central server that is also used for monitoring client's demand and traffic ensuring smooth function of the system [9]. Cloud computing systems also will have a copy of all its client's data to restore the service which may arise due to device breakdown.

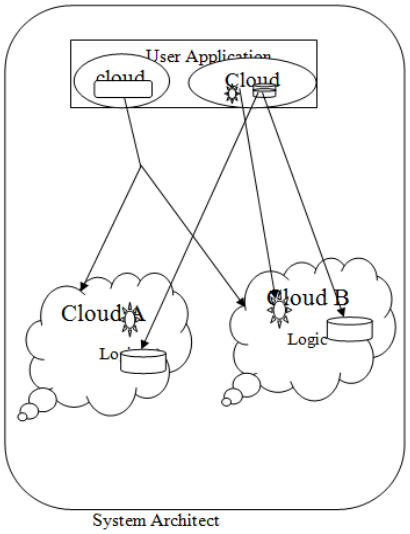
Multi-Cloud based applications:

Multi-cloud strategy is the concomitant use of two or more cloud services to minimize the risk of widespread data loss or downtime due to a localized component failure in a cloud computing environment. Such a failure can occur in hardware, software, or 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.

1. Inheritance : It is the process of creating the new classes and using the behavior of the existing classes by extending them just to reuse the existing code and adding addition a features as needed.
2. Encapsulation: It is the mechanism of combining the information and providing the abstraction. 3. Polymorphism: As the name suggest one name multiple form, Polymorphism is the way of providing the different functionality by the functions having the same name based on the signatures of the methods. 4. Dynamic binding: Sometimes we don't have the knowledge of objects about their specific types while writing our code. It is the way of providing the maximum functionality to a program about the specific type at runtime.

SYSTEM ARCHITECT



System Architect

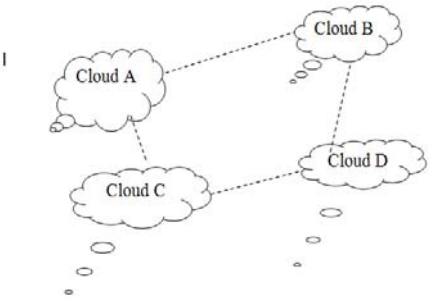
2.MODULES

2.1 Implementing Multi-Cloud.
2.2 Partition of Application System into Tiers
2.3 Partition of Application Logic into Fragments
2.4 Partition of Application Data into Fragments
2.5 Replication of Applications

2.1 Implementing Multi-Cloud.

To implements multi clouds it is a more advanced, but also more complex approach comes from the distributed algorithms discipline. Assume the existence of *n* cloud providers, of which *f* collaborate maliciously against the cloud user, with $n > 3f$. In that case, each of the *n* clouds performs the computational task given by the cloud user.

• Implementing Multi-Cloud.

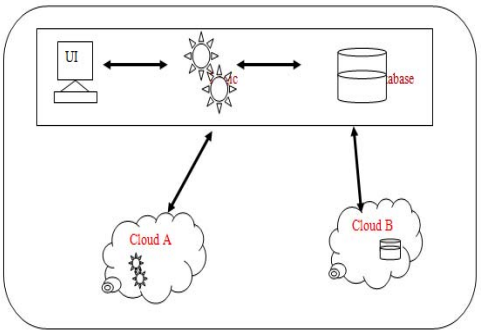


Then, all cloud providers collaboratively run a distributed algorithm that solves the General Byzantine Agreement problem [6] (e.g. the TurpinCoan or Exponential Information Gathering algorithms). After that it is guaranteed that all non-malicious cloud providers know the correct result of the computation. Hence, in the final step, the result is communicated back to the cloud user via a Secure Broadcast algorithm (e.g. plain flooding, with the cloud user taking the majority as the result). Hence, the cloud user can determine the correct result even in presence of *f* malicious cloud.

2.2 Partition of Application System into Tie

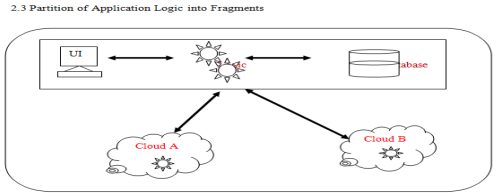
In this Partition of Application System into Tiers allows separating the logic from the data. This gives additional protection against data leakage due to flaws in the application logic . In this section targets the risk of undesired data leakage [7]. It answers the question on how a cloud user can be sure that the data access is implemented and enforced effectively and that errors in the application logic do not affect the user's data. In that that the security services provided by this architecture can only be fully exploited if the execution of the application logic on the data is performed on the cloud user's system. Only in this case, the application provider does not learn anything on the users' data.

2.2 Partition of Application System into Tiers



2.3 Partition of Application Logic into Fragments

This module allows distributing the application logic to distinct clouds. It has 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 [6].

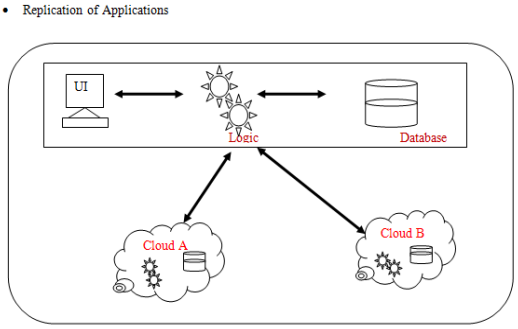


2.4 Partition of Application Data into Fragments

This 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. This multi-cloud architecture specifies that the application data is partitioned and distributed to distinct clouds. The most common forms of data storage are files and databases [4]. Files typically contain unstructured data (e.g. pictures, text documents) and do not allow for easily splitting or exchanging parts of the data.

2.5 Replication of Applications

In this Replication of Applications allows to receive multiple results from one operation performed in distinct clouds and to compare them within the own premise. This enables the user to get evidence on the integrity of the result. In this multiple distinct clouds executing multiple copies of the same application can be deployed. Instead of executing a particular application on one specific cloud, the same operation is executed by distinct clouds. By comparing the obtained results, the cloud user gets evidence on the integrity of the result [6]. In such a setting, the required trust towards the cloud service provider can be lowered dramatically. Instead of trusting one cloud service provider totally, the cloud user only needs to rely on the assumption, that the cloud providers do not collaborate maliciously against herself.



3. GENERAL

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product .It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

Types of TESTINGs

3. 1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration.

Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

3.2 Performance Test

The Performance test ensures that the output be produced within the time limits,and the time taken by the system for compiling, giving response to the users and request being send to the system for to retrieve the results.

3.3 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Acceptance Testing

User Acceptance Testing is a critical phase of any work and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Acceptance testing for Data Synchronization:

1) The Acknowledgements will be received by the Sender Node after the Packets are received by the Destination Node
2) The Route add operation is done only when there is a Route request in need
3) The Status of Nodes information is done automatically in the Cache Updation process

4. ADVANTAGE:

- One idea on reducing the risk for data and applications in a public cloud is the simultaneous usage of multiple clouds.
- The use of multiple cloud providers leads to a perceived advantage in terms of security, based on the perception of shared and thus mitigated risks.
- The advantage of cryptDB lies in the fact that the database part is a standard MySQL database, and in that its efficiency is only decreased marginally, as compared to unencrypted data storage.

5. CONCLUSION

The problem of data security in cloud data storage, which is essentially The use of multiple cloud providers for gaining security and privacy benefits is non-trivial. As the approaches investigated in this clearly show, there is no single optimal approach to foster both security and legal compliance in an Omni-applicable manner. The approaches that are favorable from a technical perspective appear less appealing from a regulatory point of view, and vice versa. As can be seen from the discussions of the four major multi-cloud approaches, each of them has its pitfalls and weak spots, either in terms of security guarantees, in terms of compliance to legal obligations, or in terms of feasibility. Given that every type of multi-cloud approach falls into one of these four categories, this implies a state of the art that is somewhat dissatisfying.

REFERENCES

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Computing,Version 15," National Institute of Standards and Technology,Information Technology Laboratory,2010.[Online].Available:http://csrc.nist.gov/groups/SNS/cloud-computing

[2] F. Gens, "IT Cloud Services User Survey, pt.2: Top Benefits & Challenges," Blog post on IDC Survey, 2008. [Online]. Available: http://blogs.idc.com/ie/?p=210

[3] P. Malinverno, "Cloud computing in europe," *Gartner Application Architecture, Development & Integration Summit*, June 2012.[Online]. Available: http://www.gartner.com/it/page.jsp?id=2032215

[4] J.-M. Bohli, M. Jensen, N. Gruschka, J. Schwenk, and L. L. Iacono, "Security prospects through cloud computing by adopting multiple clouds," in *4th IEEE International Conference on Cloud Computing (CLOUD)*. IEEE, 2011.

[5] D. Hubbard and M. Sutton, "Top Threats to Cloud Computing V1.0," Cloud Security Alliance, 2010. [Online]. Available: http://www.cloudsecurityalliance.org/topthreats

[6] M. Jensen, J. Schwenk, N. Gruschka, and L. Lo Iacono, "On technical security issues in cloud computing," in *Proceedings of the IEEE International Conference on 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 thirdparty compute clouds," in *CCS '09: Proceedings of the 16th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2009, pp. 199–212.

[8] Y. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Cross-vm side channels and their use to extract private keys," in *Proceedings of the 2012 ACM conference on Computer and communications security*, ser. CCS '12. New York, NY, USA: ACM, 2012, pp. 305–316.[Online].Available: http://doi.acm.org/10.1145/2382196.2382230

[9] N. Gruschka and L. Lo Iacono, "Vulnerable Cloud: SOAP Message Security Validation Revisited," in *ICWS '09: Proceedings of the IEEE International Conference on Web Services*. Los Angeles, USA: IEEE, 2009.

[10] M. McIntosh and P. Austel, "XML signature element wrapping attacks and countermeasures," in *SWS '05: Proceedings of the 2005 workshop on Secure web services*. New York, NY, USA: ACM Press, 2005, pp. 20–27.