IPHV7I10008X
International Journal Of Advanced Research and Innovation -Vol.7, Issue .I
ISSN Online: 2319 – 9253
Print: 2319 – 9245

EVOLUTION OF CLOUD SECURITY IN MULTI CLOUD OUT OF POSSESSION OF SINGLE CLOUD

Vijaya Laxmi Bhonagiri
Associate Professor, Dept of CSE,
Vaageswari College of Enginerring, Karimnagar, A.P

ABSTRACT:

Cloud is a common area for storing services and user data now a days. Security is a considerable issue for this type of data centers. Security consist set of policies, applications and infrastructure. The usage of cloud is increasing drastically due to the reasons of optimal pricing and high accessibility. Cloud stores user's sensitive data in a single cloud. The recent invents in cloud are multi-cloud are also termed as inter cloud. In relation to data intrusion and data integrity, assume we want to distribute the data into three different cloud providers, and we apply the secret sharing algorithm on the stored data in the cloud provider. 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. Hence, we provide a fake object scheme in which an alert through the fake object creation will be given to the admin in order to maintain data integrity. This paper also surveys recent research related to single and multi-cloud security and addresses possible solutions. It is found that the research into the use of multi cloud providers to maintain security has received less attention from the research community than has the use of single clouds. The multi-cloud model will enhance the security. In this article provided the information relates to cloud security with data integrity and data intrusion services by these services security risks are reduced.

Keywords: *Cloud computing, single cloud, multi-clouds, Map Reduce, Fake Object Scheme, Key Management, Secure Storage, cloud storage, data integrity.*

I. INTRODUCTION

The abilities to use multiple Clouds and to migrate, at design or at run-time, applications from one Cloud to another could mitigate the risk of Cloud adoption and would allow building high performance and reliable applications. The business world now demands a mix of many best-of-breed cloud services to form the optimal solution. The answer is proving to be a concept called "multicloud". Its more complex than a hybrid cloud, which is typically a paired private and private cloud. Multi cloud add more clouds to the mix, perhaps two or more public iaas providers, a private paas, on-demand management and security systems form public clouds, private used accounting. Multi clouds requires more thinking around security and governance, given their complexity and distribution, it

may develop resiliency issues, considering the number of moving parts, and these have value only if you select the right providers, whether on-demand or private.

Fake Object Schema:

Cloud computing holds the promise of revolutionizing the manner in which enterprises manage, distribute, and share information. The data owner (client) can out-source almost all its information processing tasks to a "cloud". The cloud can be seen as a collection of servers (we shall sometimes refer to it as the server) which caters the data storage, processing and maintenance needs of the client. Needless to say this new concept of computing has already brought significant savings in terms of costs for the data owner. Among others, an important service provided by a cloud is Database as a Service (DAS). In this service the client delegates the duty of storage and maintenance of his/her data to a third party (an un-trusted server). This model has gained lot of popularity in the recent times. The DAS model allows the client to perform operations like create, modify and retrieve from databases in a remote location [9]. These operations are performed by the server on behalf of the client. However, delegating the duty of storage and maintenance of data to a third party brings in some new security challenges. The two main security goals of cryptography are privacy and authentication. These security issues are relevant to the outsourced data also. The client who keeps the data with an untrusted server has two main concerns. The first one being that the data may be sensitive and the client may not want to reveal the data to the server and the second one is the data whose storage and maintenance has been delegated to the server would be used by the client. The typical usage of the data would be that the client should be able to query the database and the answers to the client's queries would be provided by the server. It is natural for the client to be concerned about a malicious server who does not provide correct answers to the client queries. In this work we are interested in this problem. We aim to devise a scheme in which the client would be able to verify whether the server is responding correctly to its queries.

The problem of interest is of data authentication, and there are well known cryptographic solutions to the basic data authentication problem. In the symmetric key setting this has been addressed by the use of message authentication codes and in the asymmetric key setting signature schemes provide this functionality.

Benefits of cloud computing:

The benefits of cloud computing are Reduced Data Leakage, Decrease evidence acquisition time, they eliminate or reduce service downtime, they Forensic readiness,they decrease evidence transfer time.

Drawbacks of cloud computing:

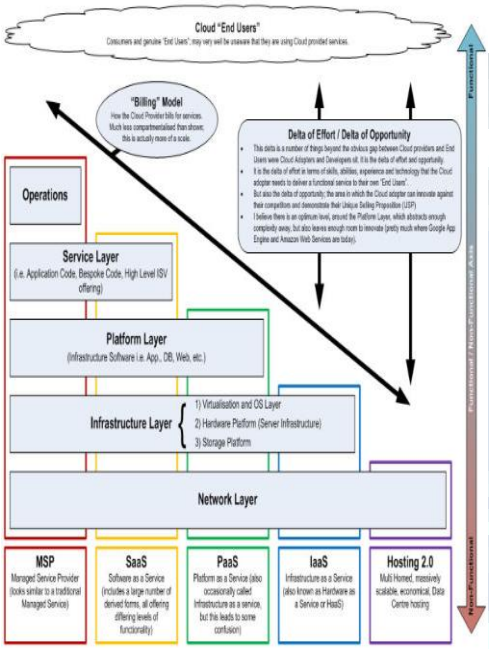
Few of the disadvantages associated with cloud computing are:

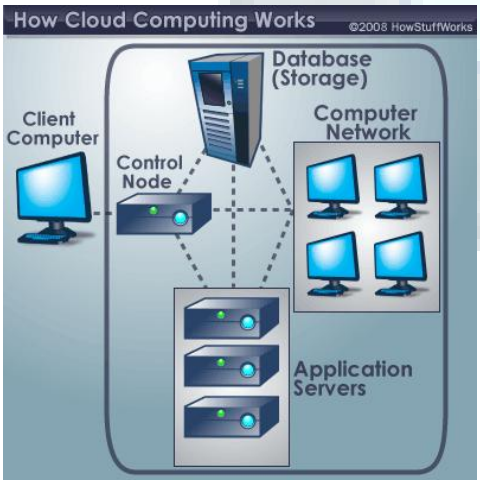
- High Speed Internet Required
- Constant Internet Connection
- Limited Features
- Data Stored is not secure.

II.RELATED WORK

Cloud architecture, the systems architecture of the software systems

involved in the delivery of cloud computing, comprises hardware and software designed by a cloud architect who typically works for a cloud integrator. It typically involves multiple cloud other over application programming interfaces, usually web services.





Cloud architecture extends to the client, where web browsers and/or software applications access cloud applications.

Cloud storage architecture is loosely coupled, where metadata operations are centralized enabling the data nodes to scale into the hundreds, each independently delivering data to applications or user.

A typical cloud computing system:

Soon, there may be an alternative for executives like you. Instead of installing a suite of software for each computer, you'd only have to load one application. That application would allow workers to log into a Web-based service which hosts all the programs the user would need for his or her job. Remote machines owned by another company would run everything from e-mail to word processing to complex data analysis programs. It's called cloud computing, and it could change the entire computer industry. In a cloud computing system, there's a significant workload shift. Local computers no longer have to do all the heavy lifting when it comes to running applications. The network of computers that make up the cloud handles them instead. Hardware and software demands on the user's side decrease. The only thing the user's computer needs to be able to run is the cloud computing system's interface software, which can be as simple as a Web browser, and the cloud's network takes care of the rest.

There's a good chance you've already used some form of cloud computing. If we have an e-mail account with a Web-based e-mail service like Hotmail, Yahoo! Mail or Gmail, then we've had some experience with cloud computing. Instead of running an e-mail program on our computer, we log in to a

Web e-mail account remotely. The software and storage for our account doesn't exist on our computer – it's on the service's computer cloud.

WHAT IS DRIVING CLOUD COMPUTING

The CLOUD COMPUTING is driving in two types of categories.

Customer perspective:

- In one word: economics
- Faster, simpler, cheaper to use cloud computation No upfront capital required for servers and storage.
- No ongoing for operational expenses for running datacenter.
- Application can be run from anywhere.

Vendor perspective:

- Easier for application vendors to reach new customers.
- Lowest cost way of delivering and supporting applications.
- Ability to use commodity server and storage hardware.
- Ability to drive down data center operational cots.
- Computer hardware (Dell, HP, IBM, Sun Microsystems)
- Storage (Sun Microsystems, EMC, IBM)
- Infrastructure (Cisco Systems)
- Computer software (3tera, Hadoop, IBM, RightScale)
- Operating systems (Solaris, AIX, Linux including Red Hat)
- Platform virtualization (Citrix, Microsoft, VMware, Sun xVM, IBM)

SECRET SHARING ALGORITHMS:

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. What if we use an insecure, easily guessable password to protect the 2012 45th Hawaii International Conference on System Sciences file, or the same one to protect all files? I have often thought that secret sharing algorithms could be employed to good effect in these circumstances instead.

EXISTING SYSTEM:

Cloud providers should address privacy and security issues as a matter of high and urgent priority. Dealing with "single cloud" providers is becoming less popular with customers due to potential problems such as service availability failure and the possibility that there are malicious insiders in the single cloud. In recent years, there has been a move towards "multi clouds", "inter cloud" or "cloud-of-clouds".

DISADVANTAGES:

1. Cloud providers should address privacy and security issues as a matter of high and urgent priority.

2. Dealing with "single cloud" providers is becoming less popular with customers due to potential problems such as service

availability failure and the possibility that there are malicious insiders in the single cloud.

PROPOSED SYSTEM:

This paper focuses on the issues related to the data security aspect of cloud computing. As data and information will be shared with a third party, cloud computing users want to avoid an untrusted cloud provider. Protecting private and important information, such as credit card details or a patient's medical records from attackers or malicious insiders is of critical importance. In addition, the potential for migration from a single cloud to a multi-cloud environment is examined and research related to security issues in single and multi-clouds in cloud computing are surveyed.

ADVANTAGES:

1. Data Integrity
2. Service Availability.
3. The user runs custom applications using the service provider's resources
4. Cloud service providers should ensure the security of their customers' data and should be responsible if any security risk affects their customers' service infrastructure.

III. MAP REDUCE

MapReduce is a programming model for processing large data sets with a parallel, distributed algorithm on a cluster. A Map Reduce program is composed of a Map() procedure that performs filtering and sorting (such as sorting students by first name into queues, one queue for each name) and a Reduce()procedure that performs a summary operation (such as counting the number of students in each queue, yielding name frequencies). The "Map Reduce System"(also called "infrastructure" or "framework")orchestrates by marshalling the distributed servers, running the various tasks in parallel, managing all communications and data transfers between the various parts of the system, and providing for redundancy and fault tolerance.

The model is inspired by the map and reduce functions commonly used in functional programming, although their purpose in the Map Reduce framework is not the same as in their original forms. Furthermore, the key contributions of the MapReduce framework are not the actual map and reduce functions, but the scalability and fault-tolerance achieved for a variety of applications by optimizing the execution engine once. Map Reduce libraries have been written in many programming languages, with different levels of optimization. A popular open-source

implementation is cloud computing. The name MapReduce originally referred to the proprietary Google technology but has since been genericized.

'MapReduce' is a framework for processing parallelizable problems across huge datasets using a large number of computers (nodes), collectively referred to as a cluster (if all nodes are on the same local network and use similar hardware) or a grid (if the nodes are shared across

geographically and administratively distributed systems, and use more heterogenous hardware). Computational processing can occur on data stored either in a filesystem (unstructured) or in a database (structured). MapReduce can take advantage of locality of data, processing data on or near the storage assets to decrease transmission of data. "Map" step: The master node takes the input, divides it into smaller sub-problems, and distributes them to worker nodes. A worker node may do this again in turn, leading to a multi-level tree structure. The worker node processes the smaller problem, and passes the answer back to its master node.

"Reduce" step: The master node then collects the answers to all the sub-problems and combines them in some way to form the output – the answer to the problem it was originally trying to solve. MapReduce allows for distributed processing of the map and reduction operations. Provided that each mapping operation is independent of the others, all maps can be performed in parallel – though in practice this is limited by the number of independent data sources and/or the number of CPUs near each source. Similarly, a set of 'reducers' can perform the reduction phase, provided that all outputs of the map operation that share the same key are presented to the same reducer at the same time, or that the reduction function is associative. While this process can often appear inefficient compared to algorithms that are more sequential, MapReduce can be applied to significantly larger datasets than "commodity" servers can handle – a large server farm can use MapReduce to sort a petabyte of data in only a few hours. The parallelism also offers some possibility of recovering from partial failure of servers or storage during the operation: if one mapper or reducer fails, the work can be rescheduled – assuming the input data is still available.

Another way to look at MapReduce is as a 5-step parallel and distributed computation:

1. Prepare the Map() input – the "MapReduce system" designates Map processors, assigns the K1 input key value each processor would work on, and provides that processor with all the input data associated with that key value.
2. Run the user-provided Map() code – Map() is run exactly once for each K1 key value, generating output organized by key values K2.
3. "Shuffle" the Map output to the Reduce processors – the MapReduce system designates Reduce processors, assigns the K2 key value each processor would work on, and provides that processor with all the Map-generated data associated with that key value.
4. Run the user-provided Reduce() code – Reduce() is run exactly once for each K2 key value produced by the Map step.
5. Produce the final output – the MapReduce system collects all the Reduce output, and sorts it by K2 to produce the final outcome.

Logically these 5 steps can be thought of as running in sequence – each step starts only after the previous step is completed – though in practice, of course, they can be intertwined, as long as the final result is not affected. In many situations the input data might already be distributed ("sharded") among many different servers, in which case step 1 could sometimes be greatly simplified by assigning Map servers that would process the locally present input data. Similarly, step 3 could sometimes be sped up by assigning Reduce processors that are as much as possible local to the Map-generated data they need to process.

IV.IMPLEMENTATION

In this concept we implement the four types of implementation methods, they are
1. Data Integrity
2. Map Reduce
3. Service Availability
4. DepSKy System Model

DATA INTEGRITY:

One of the most important issues related to cloud security risks is data integrity. The data stored in the cloud may suffer from damage during transition operations from or to the cloud storage provider. Cachinet al. gives examples of the risk of attacks from both inside and outside the cloud provider, such as the recently attacked Red Hat Linux's distribution servers.

One of the solutions that they propose is to use a Byzantine fault-tolerant replication protocol within the cloud. Hendricks et al. State that this solution can avoid data corruption caused by some components in the cloud. However, Cachinet al.Claim that using the Byzantine fault tolerant replication protocol within the cloud is unsuitable due to the fact that the servers belonging to cloud providers use the same system installations and are physically located in the same place.

MAP REDUCES:

The *Map* and *Reduce* functions of *MapReduce* are both defined with respect to data structured in (key, value) pairs. *Map* takes one pair of data with a type in one data domain, and returns a list of pairs in a different domain:

Map(k1,v1) → list(k2,v2)

The *Map* function is applied in parallel to every pair in the input dataset. This produces a list of pairs for each call. After that, the MapReduce framework collects all pairs with the same key from all lists and groups them together, creating one group for each key. The *Reduce* function is then applied in parallel to each group, which in turn produces a collection of values in the same domain:

Reduce(k2, list (v2)) → list(v3)

Each *Reduce* call typically produces either one value v3 or an empty return, though one call is allowed to return more than one value. The returns of all calls are collected as the desired result list. Thus the MapReduce framework transforms a list of (key, value) pairs into a list of values. This behavior is different from the typical

functional programming map and reduce combination, which accepts a list of arbitrary values and returns one single value that combines *all* the values returned by map. It is necessary but not sufficient to have implementations of the map and reduce abstractions in order to implement MapReduce. Distributed implementations of MapReduce require a means of connecting the processes performing the Map and Reduce phases. This may be a distributed file system. Other options are possible, such as direct streaming from mappers to reducers, or for the mapping processors to serve up their results to reducers that query them.

SERVICE AVAILABILITY:

Another major concern in cloud services is service availability. Amazon mentions in its licensing agreement that it is possible that the service might be unavailable from time to time. The user's web service may terminate for any reason at any time if any user's files break the cloud storage policy. In addition, if any damage occurs to any Amazon web service and the service fails, in this case there will be no charge to the Amazon Company for this failure. Companies seeking to protect services from such failure need measures such as backups or use of multiple providers.

DEPSKY SYSTEM MODEL:

The DepSky system model contains three parts: readers, writers, and four cloud storage providers, where readers and writers are the client's tasks. Bessani et al. 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.

V.CONCLUSION

Cloud computing is a powerful new abstraction for large scale data processing systems which is scalable, reliable and available. Cloud computing security is still considered the major issue in the cloud computing environment. Customers do not want to lose their private information as a result of malicious insiders in the cloud. In addition, the loss of service availability has caused many problems for a large number of customers recently. Furthermore, data intrusion leads to many problems for the users of cloud computing.

In this paper we have analyzed the challenges and viability of deploying a computing cluster on top of a multi-cloud infrastructure. We also proposed a different data fragmentation schemes for multi cloud storage in cloud computing, which seeks to provide each customer with reliability, availability and better cloud data storage decisions. we proposed a secured cost-effective multicloud storage (SCMCS) in cloud computing, which seeks to provide each customer with a better cloud data storage decision, taking into consideration the user budget as well as providing him with the best quality of service (Security and availability of data) offered by available cloud service providers.

The purpose of this work is to survey the recent research on single clouds and multi-clouds to address the security risks and

solutions. We have found that much research has been done to ensure the security of the single cloud and cloud storage whereas multiclouds have received less attention in the area of security. We support the migration to multi-clouds due to its ability to decrease security risks that affect the cloud computing user.

VI. REFERENCES

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

[2] Multi-Cloud Deployment of Computing Clusters for Loosely-Coupled MTC Applications Rafael Moreno Vozmediano, Ruben S. Montero, Ignacio M. Llorente (special issue on many task computing) july 2010.

[3] B.AmarNadh Reddy, P.Raja Sekhar Reddy / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 5, September- October 2012, pp.1130-1134

[4] 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.

[5] H. Abu-Libdeh, L. Princehouse and H.Weatherspoon, "RACS: a case for cloud storagediversity", SoCC'10:Proc. 1st ACM symposium onCloud computing, 2010, pp. 229-240.

[6] D. Agrawal, A. El Abbadi, F. Emekci and A.Metwally, "Database Management as a Service: Challenges and Opportunities", ICDE'09:Proc.25thIntl. Conf. on Data Engineering, 2009, pp. 1709-1716.

[7] 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.

[8] Amazon, Amazon Web Services. Web services licensing agreement, October3,2006.

[9] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson and D. Song, "Provable data possession at untrusted stores", Proc. 14th ACM Conf. on Computer and communications security,2007, pp. 598-609.

[10] 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.

[11] K. Birman, G. Chockler and R. van Renesse,"Toward a cloud computing researchagenda", SIGACT News, 40, 2009, pp. 68-80.

[12] 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.

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

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

AUTHOR PROFILE:

Vijaya Laxmi Bhonagiri working as Assoc. Professor in CSE dept. at Vaageswari college of engineering. I have completed my Master of Technology from TRR College of engineering in Computer Science Engineering. My research area including Cloud Computing with Data Mining.