SECURE CLOUD APPROACH WITH AUTO-UPDATE MECHANISM

Shrutika Khobragade, Prof. Rohini Bhosale
Pillai HOC College of Engineering and Technology
Rasayani, India, University of Mumbai
shrutikak@mes.ac.in, bhosalerohini123@gmail.com

Abstract
Cloud Computing has the gigantic storage for data which we need to save and retrieve. We can store any file on the cloud and retrieve as and when required. As the data is handled by the third party organization, security is the biggest concern to matter. A high security measure is needed to keep the data up-to date and to protect data within the cloud. This mechanism is used to increase the security and performance issues. As the file is stored in a particular node which might get affected due to attack and will create a problem. So, in this work instead of saving a whole file in a particular node, we can divide the file into various fragments and store at various nodes at a particular distance by the centrality measure. This mechanism will not reveal all the information regarding that file even after successful attack. Instead of downloading the whole file we can update the prescribed fragment only.
Keywords: Cloud Security, file fragmentation, file replication, auto-update mechanism of file.

1.INTRODUCTION

Cloud computing encompasses various activities such as the more use of networking sites and other forms of interpersonal computing. However huge amount of resources on cloud storage, data or software applications has been accessed online which play an important role for the privacy and security of the data. As cloud computing is a flexible, cost-effective, and authenticated delivery platform for providing consumer IT services and business on the internet. However, cloud Computing presents an added level of hazard as essential services are often outsourced to a third party, which makes it difficult to maintain data security and privacy, demonstrate consent and also support data and service availability. The cloud computing paradigm has reformed the control and management of the information technology infrastructure[7] Cloud computing is characterized by on-demand self-services, resource pooling, ,elasticity, ubiquitous network accesses and measured assurance of the services.[22,8] The aforesaid characteristics of cloud computing make it a remarkable candidate for businesses, organizations, and individual users to endorse in this technique[25]. However, the benefits of low-cost, imperceptible management (from users perspective),and greater resilience come with increased security concerns[7] .

The data which needs to be stored on the cloud's virtualized and common environment leads to high risk in security. Flexibility and pooling of a cloud is used to share the physical resources among many users [22]. As the data on the cloud can be a shared resource assigned to the users which may lead to data compromise. Virtual machine also stores some data for the sharing purpose whereas in multi-tenant virtual system there is a problem for recovery of the data. Likewise cross-tenant virtualization may also distract data integrity and security. The virtual data as it moves between VMs or in the cloud is easily lost or exposed [5].

The data stored on the cloud must be secured. Other unauthorized users should not access the user's data. Any weak entity can lead to whole cloud at risk. For such scenario, a high security

measure should be maintained by increasing an attacker's effort so that feasible amount of data can be retrieved even after successful intrusion in the cloud. Apparent loss of the data (cause due to data leakage) has to be minimized.

A large amount of data is stored on cloud which should ensure reliability, throughput and security [15].Unless the security performance also play an important role in cloud computing. The performance of the system has to be maintained. The data or any file is uploaded and downloaded on the cloud has to be secured. However to maintain this and to increase a performance, actual action needs to be taken or proper data has to be retrieved by the user after uploading the data .But such problems should be dealt with data replication strategies [3].However to store this replicas data over a number of node increases the attack on that peculiar data. Instead of storing one replica on a node, we can store m number of replicas in different nodes. As replica of particular file is stored on a different nodes the probability of attack on nodes increases from 1/n to m/n where n is the total number of nodes.

The major contribution in this paper are :

- To maintain the security and performance both, we will develop a scheme for data which is present in cloud storage .
- This scheme will fragments the file which needs to upload on cloud and replicates that file on different nodes.
- This proposed system ensures that even if that nodes get attacked whole file will not get revealed as its stores the fragmented data.
- As we do not need to rely on traditional cryptographic techniques for data security. The proposed scheme depicts non cryptographic techniques which makes it faster to perform the required operations (placement and retrieval) on the data file which is stored.

We need to ensure that a controlled replication of the file fragments is maintained, for the purpose of improved security where each of the fragments is replicated only once .

2.LITERATURE SURVEY

Cloud computing is a service model that offers users (tenants) on-demand network access to a huge shared pool of computing resources ("the cloud").From the tenant's perspective, the ability to utilize and pay for data as well as resources on demand and the rapid adaptability of the cloud are strong enticement of the data for migration to the cloud.

A. IRIS FILE SYSTEM

Major hurdle to adoption are the security and operational risks to which existing cloud infrastructures are prone to malware, software bugs, power outages, including hardware failures, server misconfiguration and insider threats among others. Such failures and attack vectors aren't new, but their risk is amplified by the large scale of the cloud. Their impact can be disastrous, and can include data loss and corruption, data breach and data confidentiality, and malicious tampering with data. Therefore, strong stability of data beyond mere encryption are a necessity for data outsourced to the cloud. For this a technique to ensure the integrity, freshness, and availability of data in a cloud. The data migration to the cloud is performed by the Iris file system. A gateway application is designed and employed in the organization that ensures the integrity and freshness of the data using a Merkle tree. The file blocks, MAC codes, and version numbers are stored at various levels of the tree. This technique heavily depends on the user employed scheme for data confidentiality. Moreover, the probable amount of loss in case of data tampering as a result of intrusion or access by other VMs cannot be decreased.

B. DIKE AUTHORIZATION SYTEM

As previous work says that it is block based access but remote storage access through a file-based interface often improves the performance of virtual machines in comparison to block-based access. The block-based interface detach a virtual disk into a protection domain under full control by the guest owner. On the other hand, a file-based interface provides the "killer" advantage of configurable guest isolation and sharing support at fine granularity with the file system. Due to this improper sanitization of file is not handled and may lead to leakage of data. This disadvantage will get overcome in the proposed method of File system.

A data on the file has to be secure and optimal, the placement of data objects in a distributed file system is presented in [21]. An encryption key is divided into number of shares(n) and it is distributed on different sites within the network. The n shares has to divide a key through the (k, n) threshold secret sharing scheme. As we know the network is divided into clusters. The number of replicas of a particular file and their placement is determined by examining of the data. A primary site is selected in each of the clusters that allocates the replicas within the cluster. The scheme is used to combine the replication problem with high security and assured access time improvement with private data or file.

III PROPOSED SYSTEM MODEL

This proposed system model defines the automatic update of file which is uploaded on the cloud. A large amount of files are stored on the cloud. In this system a file need to be uploaded and then it gets saved in a cloud as a third party organization. An attacker may attack a particular system and so the users data may get crashed or loss due to any attacks as the data is to be secured. So this proposed systems says that if any file is uploaded on cloud, that particular will divide into fragments and store each fragment on different nodes at certain distance. Using this system even if the attacker attacks on that particular file, the attacker will not get the complete information about the file which was stored on the cloud.

The file after fragmentation and replication, it is secured but the fragment of a file needs to be updated. The main advantage is that the file will get stored in a secured format. Other work we will do here is when the particular fragment is stored on the cloud which needs to update the data will not be sufficient. So instead of downloading the whole file to update we will download the part of fragment which the user need to update and the upload again. This auto update of file will be secured and well as of immense importance.

In a cloud environment, a complete file is stored at a node leads to a single point of failure in the cloud. The data stored on the cloud must be secured as well with privacy enable system. A successful attack on a node in the cloud will might put the data integrity or confidentiality, or both at risk. The preceding scenario can occur both in the case of intrusion or accidental errors while uploading and downloading the data.. In such systems, performance in terms of retrieval time can be heighten by employing replication strategies. However, in the replication strategy the number of copies are replicated and stored at different nodes. Thus increases the probability of the node where that file will be a victim of an attack. Security and replication are essential for a large-scale system, such as cloud, as both exploit services to the end user. Security and replication of a file or data on cloud must be balanced such that one service must not lower the service level of the other.

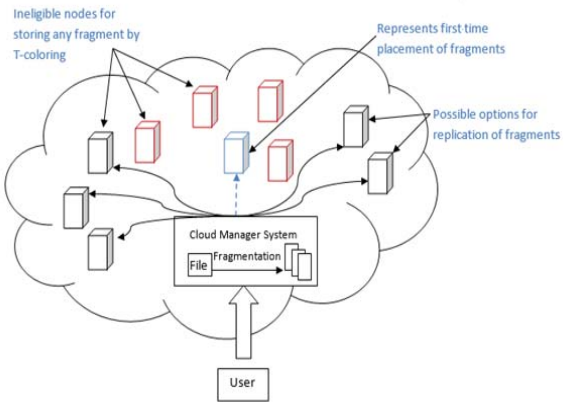

Fig.1 : Proposed Architecture

IV PROPOSED SYSTEM TECHNIQUE

A. Data Fragmentation

The security of a large-scale system, such as cloud depends on the security of individual nodes and the security of the system as a whole both. A successful intrusion into a single node may have strict emanation, not only for applications and data on the victim node, but also for the other nodes. Due to whole file, the data on the victim node may be revealed. A successful intrusion may be a result of some software or administrative vulnerability [17].In case of homogenous systems, the same defect can be utilized to target other nodes within the system. The success of an attack on the subsequent nodes will require less effort as compared to the effort on the first node. Comparatively, more effort is required for heterogeneous systems. However, compromising a single file will have to penetrate only a single node. The amount of compromised data stored on cloud storage can be reduced by making fragments of a data file and storing them on independent or isolated nodes [17, 21]. A successful intrusion on a single or few nodes will only provide access to a portion of data that

might not be of any significance. Moreover, if an attacker is uncertain about the locations of the fragments, the probability of finding fragments on all of the nodes is very low.

Let us consider a cloud with M nodes and a file with y number of fragments. Let s be the number of successful intrusions on distinct nodes, such that s>y. As more number of nodes, the probability of that state reduces. So we can say that more the value of M nodes more will the attack on that particular file.

B. Centrality

As the nodes are stored in a network at various places in the form of graph. Here we can find the distance between the nodes among the network. A file is divided into fragments and each fragment is stored at different node so the distance between the node can be calculated using the centrality measure by closeness and betweeness of the different nodes in the system.

C. T-coloring

Consider we have graph G which has G(V,E) and a set which has non negative integers and even 0 is added. So to place a particular nodes or to decide which node will be allowed to fragment, we can set a color to that particular node as OPEN or CLOSE. OPEN denotes that node is ready to fragment and CLOSE indicates it cannot be fragmented further.

When we upload a small data (.txt) it is converted into fragments and encrypted and then stored in various nodes (fragments). When we download the data using authorised key it is first decrypted and then the fragments are combined from various nodes used and the file is presented in its original form. While working on future work, we will tamper the downloaded file and then upload the data. When we upload the data, the application will check each fragments separately to see whether tampering is performed in the fragments and the fragments which are tempered will be saved in the temporary database (and will not be replaced by the original file in the cloud). The admin will be given the details regarding the changes and admin will forward it to the owner of the file to check whether the changes that are performed in the original file are acceptable to the owner or not. If the owner replaces the original file with the tampered file only the tampered fragments will be replaced and not all the fragments. The file will be deleted from temporary database and will be replaced in the cloud with the original one.

These were the procedure which will be implemented to upload the file then division and replication is done with auto-update mechanism.

V CONCLUSION

The proposed system which is a cloud storage security scheme which altogether deals with the security and performance in terms of data retrieval time. The auto update of a particular fragment of a file will become easiest way to save time as well as the performance also increases. This system will develop an automatic update mechanism that will identify and update the prescribed fragments only. The work will save the time and resources utilized in downloading, updating, and uploading the file again.

References

[1] 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. Computer and Comm.Security (CCS '07), Pp. 598-610, 2007

[2] C. Wang, Q. Wang, K. Ren, and W. Lou, Ensuring Data Storage Security in Cloud Computing", Proc. 17th Int'l Workshop Quality of Service (IWQoS'09), Pp. 1-9, 2009.

[3] C. Wang, Q. Wang, K. Ren, and W. Lou, Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing", Proc. IEEE INFOCOM, Pp. 525-533, 2010.

[4] C. Wang, S.S. Chow, Q. Wang, K. Ren, and W. Lou, \Privacy-Preserving Public Auditing for Secure Cloud Storage", IEEE Trans.Computers, Vol. 62, no. 2,Ppp. 362-375, Feb. 2013.

[5] A. Juels and B.S. Kaliski, \PORs: Proofs of Retrievability for Large Files", Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07), Pp. 584-597, 2007.

[6] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, Scalable and Eficient Provable Data Possession", Proc. Fourth Int'l Conf. Security and Privacy in Comm. Networks (SecureComm'08), 2008

[7] B. Wang, B. Li, and H. Li, Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud", Proc. IEEE Fifth Int'l Conf. Cloud Computing, Pp. 295-302, 2012.

[8] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, Dynamic provable data possession", Cryptology ePrint Archive, Report 2008/432, 2008, http://eprint.iacr.org/.

[9] H. Shacham and B. Waters, Compact proofs of retrievability", in ASIACRYPT '08: Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security. Berlin, Heidelberg: Springer-Verlag, 2008, Pp. 90-107.

[10] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, Enabling public verifiability and data dynamics for storage security in cloud computing", in ESORICS, 2009, Pp. 355-370.