# P2P CLOUD: PEER-TO-PEER BASED SECURITY AWARE CLOUD STORAGE ARCHITECTURE

BY

B. LALITHA * A.V.L.N. SUJITH **

* Assistant professor, Department of Computer Science and Engineering, Jawaharlal Nehru Technological University, Anantapur, India.
** Lecturer, Department of Computer Science and Engineering, Jawaharlal Nehru Technological University, Anantapur, India.

*ABSTRACT*

*Cloud computing is most popular for its ubiquitous behavior, because the users can access the resources irrespective of their location and time. Even though it is cost effective, flexible and efficient platform for delivering services, deployment of cloud computing in real-time business environment brings out substantial security issues in handling essential data. It is very difficult to provide security, maintain privacy and update service availability in cloud, because they outsource their services frequently to a third party. Clouds can be safeguarded by securing virtualized data center resources, preserving data integrity and maintaining user privacy. Thereby, secured and efficient methods are required to address confidentiality and integrity of the outsourced data on untrusted cloud servers. In this paper, the authors propose a Peer-to-Peer based cloud architecture to secure resources in virtualized data centers and implementation of a secured protocol based on cryptographic and random sampling techniques for data integrity in cloud.*

*Keywords: Cloud Security, Peer-to-peer, Cryptography, Data Integrity, Confidentiality*

## INTRODUCTION

The main objective of cloud computing is effective resource sharing, efficient data storage and quick data access. The ability of the cloud to adapt with the fluctuations occurs due to the demand, and its development made it cost effective platform at enterprise level with efficient computing and available resources. Cloud computing is a combination of various computing technologies and concepts like web 2.0, virtualization, service oriented architecture providing various real time applications through browsers to fulfill the needs of the users. Though users are getting benefited adopting cloud, one of the main issue that acts as a barrier for cloud computing is security. As cloud is completely new computing model, researchers are facing a great problem in providing the security at different levels like network [6], host and application. Thus the issue of security in cloud computing has become a great challenge to be focused on. When compared to the legacy trends of computing, cloud computing has massive features in terms of size and resources that are to be totally decentralized, virtualized and maintain heterogeneity in data. Traditional mechanisms of providing security such as authentication, authorization and identity are no longer adequate for cloud in current trend.

Cloud computing offers dynamic large-scale and geographically distributed application services that require independent and highly distributed software entities and security. Perhaps the sober problem of centralized cloud is that, it exhibits single point of failure and surplus network connections also fail to provide security against terrible events such as fires and floods. To address these issues, the cloud services must be decentralized by nature that can be solved using peer to peer overlay networks where, peers identify one another over a decentralized setting. Peer to peer cloud incorporates many of the properties of various peer to peer overlays such as scalability, availability and reliability. The advantage of p2p cloud is that the components are small and consumes less power, the connectivity of p2p cloud is established through various ISPs (Internet Service Provider) reducing hugely the risk of single point of failure at data centers [7]. In [8], a hierarchical reputation system

is proposed to manage trust in cloud computing environment. The security in cloud computing is generally increased due to virtualization. Data isolation and security against the DOS (denial of service attacks) is provided by virtual machines.

The promising way to integrate the distributed entities is based on unstructured peer-to-peer network models. In literature, unstructured peer to peer systems are commonly referred to as efficient and reliable networks [4]. Unstructured peer-to-peer networks such as Gnutella [2], Kazaa [3] and Bit torrent [1] became increasingly popular due to their property of leveraging to build decentralized large scale applications such as distributed data storage cooperative backup, and distributed multicast.

## 1. Peer-to-peer based Cloud Architecture

Peer-to-peer based cloud architecture is designed by interconnecting the core components of the cloud system such as servers, virtual machines and application services through peer-to-peer routing. Resources in the virtualized data centers can be distributed and secured by integrating the components of the cloud system with a peer-to-peer network model as shown in Figure 1. Peer-to-peer cloud provisioning architecture is designed in layered approach that includes physical cloud servers integrated with middleware capabilities to deliver IaaS

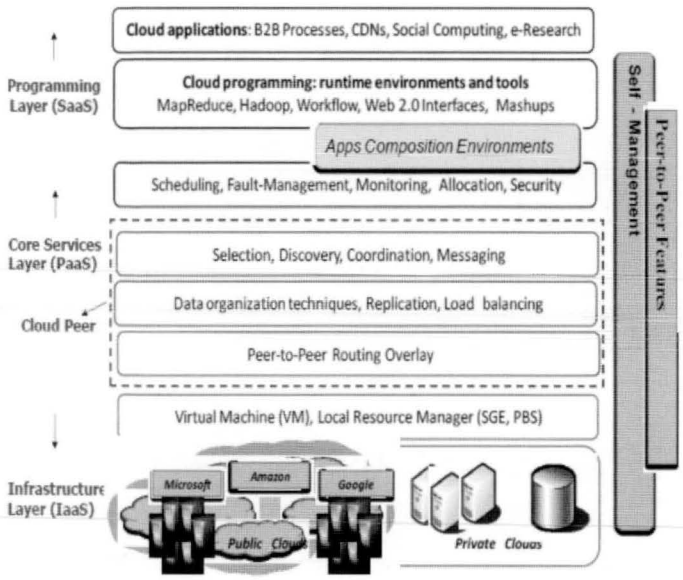


Figure 1. P2P based Cloud Architecture

(Infrastructure as a Service). The middleware at the user-level aims to provide PaaS (Platform as a Service) capability. The top most layer in the architecture utilizes the services of both underlying layers focusing on the application level services such as SaaS (Software as a Service). The Services provided by SaaS and PaaS are developed and offered by the third party service providers that are different from IaaS.

### *1.1 Application Services in Cloud*

Business level applications in cloud includes Enterprise applications such as Sales force CRM (Customer Relationship Management) and ERP's (Enterprise Resource Planning), E-commerce applications, Social networking applications and CBS (Content Based Searching) applications. Depending on the usage, these applications vary radically with their work load profiles and characteristics. QoS (Quality of Service) plays a vital role in business applications in terms of optimal resource usage, dynamic scaling of the deployed services and achieving noteworthy performance based on the service request by the end users.

### *1.2 Framework Layer*

Framework layer includes diverse software frameworks similar to Web interfaces for instance, Visual Studio.net, IBM work place and Ajax which facilitate users to create rich and cost effective user interfaces for browser oriented applications. This layer facilitates creation, deployment and execution of the cloud application in various programming environments such as Hadoop, MapReduce and Dryad with specific composition tools.

### *1.3 Services Layer*

Using user-level middleware, PaaS facilitates runtime environment by enabling cloud computing capabilities to implement application services. The services offered by this layer are monitoring, Accounting, Billing and Pricing, Dynamic SLA Management, Scheduling and Fault-Management. Further, this layer is capable of providing distributed coordination, message passing between cloud components. Amazon EC2's Cloudwatch and Load-balancer service, Google App Engine, MS Azure's fabric controller and Aneka [5] are few core

services that are being operated currently at this layer.

*1.4 Cloud infra-layer (IaaS)*

A group of data centers are installed along with numerous servers to provide computational power in cloud environments. This layer consists of immense physical servers such as application servers and storage servers that influence data centers. These physical servers are apparently managed with the help of virtualized services to share load and capacity among virtual servers. To achieve fault tolerant behavior, the virtual machines in cloud are isolated from each other. As the resources in the data centers can be utilized either through public clouds or private clouds, there is a chance of forming hybrid clouds where the requested resources are given lease form private cloud to public to cope up with demand.

## 2. Peer-to-Peer Cloud Storage

Novel cloud storage architecture is designed based on P2P computing principles which consist of a solitary database and various chunk servers, connected with multiple clients. P2P strategy of "allocating huge amount of files to unreliable peers" is incorporated in this architecture such that the distribution capacity of these peers is restricted. Three modules are mainly focused in this architecture namely Client, Gateway and Chunk Server as depicted in Figure 2.

Initially, the clients have to send their authentication details to get connected with server. Whenever a client requests a data on a chunk server, the database directs it to a particular location of the replicas through gateway. Each chunk server is referred to as P2P node, which includes Index module as a part of global resource index allocated with DHT (Discrete Hartley Transform) arithmetic such as pastry and chord. Routing module routes the request to the next hop in the routing table and the data module locates the data resource in chunk servers.

The client application is initially designed to acquire data from the platform. Initially, the client specifies his user name and password to authenticate and send request through the gateway and obtain response after processing the request thereby providing the user profile privacy. The gateway acts as an entity that acts as a

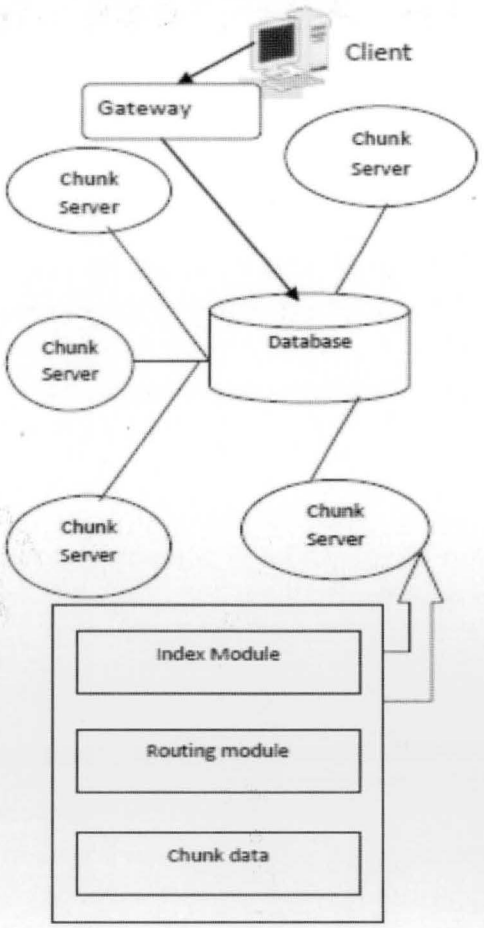


Figure 2. Cloud Storage

media to transfer the request and response between the client app and the network. It is considered as an important interface between the client and the chunk server as the request is initially transferred to the nearest chunk server. The gateway initially constructs the P2P request package that includes the chunk server and the request. The functional module in the chunk server facilitates the data integrity and the efficient processing of the user request. Selection of the chunk servers for the purpose of the storage is done based on the traditional techniques in cloud computing platform.

## 3. Secured cloud storage protocol

To make sure of data integrity and confidentiality in cloud, a secured and efficient protocol is designed by using cryptographic and random sampling techniques. This protocol mainly consists of three phases that include Key generation phase, Encryption phase and M-data

generation phase.

### 3.1 Key-Gen Phase

Key-Gen Algorithm

1. *Input : $K_G(M) \leftarrow \{P_K, PR_K\}$*
2. *Consider Security constraint (M >512)*
3. *Select two random primes A and B with Size 'M' such that A ≡B ≡2(mod 3).*
4. *Compute s = AB.*
5. *Compute $S_s = LCM(A+1, B+1)$*
6. *Produce Random integer q<w, Gcd(q,w)=1*
7. *Compute p*
8. *Public Key '$P_K$' = {$S_s$}*
9. *Private Key '$PR_K$' = {q,w,e}*
10. *End.*

In this phase, user takes 'M' as input and generates public key and private key pair using the above algorithm. Here, from the security constraint (M >512), the user selects a dual set of large primes 'A' and 'B' with size 'M' such that A ≡ B ≡ 2(mod 3). Then s = AB and $S_s$ = LCM (A+1, B+1) where $S_s$ is the order of elliptic curves and finally the output is public key '$P_K$' = {$S_s$} and private key '$PR_K$' = {q,w,e}.

### 3.2 Encryption

In this phase, the user encrypts every block $D_i$ in file 'f' using following algorithm. Here it considers a secret random the limit '$S_i$' as input and '$D_i$' as output.

Encrypt-Algorithm

1. *Input : Encrypt($D_i$, S)*
2. *For 1 to n*
3. *Compute $D_i' = D_i + f_k(S_i)$*
4. *End for*
5. *End.*

### 3.3 M-Data Gen

Subsequent to data encryption, the user computes the metadata to validate data integrity using the following algorithm that takes $D_i'$, private key and public key to generate metadata $V_i$ as output $V_i \leftarrow D_i' P(\text{mod } S_s)$

M-data Gen Algorithm

1. *Input : M-data Gen ($D_i'$, )←$V_i$*
2. *For 1 to n.*
3. *Calculate $V_i \leftarrow D_i' P(\text{mod } S_s)$*
4. *End for*
5. *End.*

### 4. Integrity analysis and Performance Results

This section presents a formal integrity analysis of the data in the proposed cloud architecture. All the experiments are conducted using java and a system with dual core processor and windows 2007 operating system. The analysis is done based on three attributes including soundness, completeness and probability detection. The performance evaluation of the cloud architecture is done in terms of coordination delay, response time and message count with respect to the problem complexity and is depicted as follows.

Figure 3 shows the evaluation of the response time for different cloud service providers with respect to problem complexity. Figure 4 depicts the evaluation results of the message count for the proposed scheme.

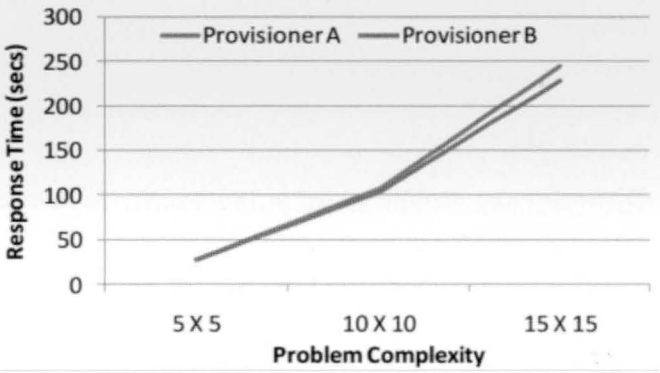


Figure 3. Response Time

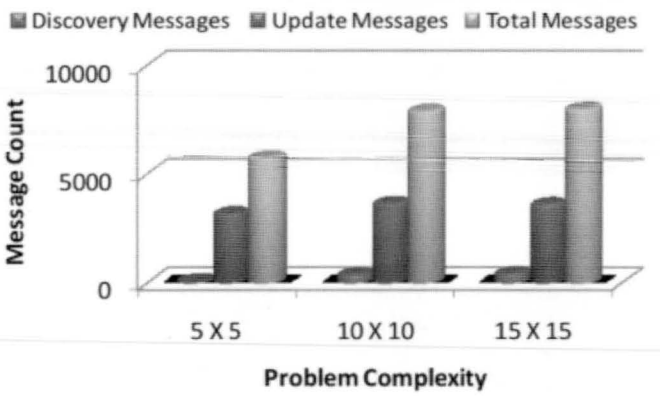


Figure 4. Message count

## Conclusion

This paper presents a P2P based distributed cloud storage architecture without any central server for monitoring the process of secured data storage and retrieval. The main advantage of this schema is that it overcomes the bottleneck problem that most commonly arises in client server based architecture. As the storage process is based on peer-to-peer policies, monitoring is done to identify the best chunk servers to perform efficient utilization of resources and load balancing between the servers. As a further direction, the proposed scheme is modified by reducing the number of servers and evaluates this architecture in terms of scalability, fault-tolerance and manageability.

## References

[1]. **Pouwelse, J. A, Garbacki, P, Epema, D. H. J, and Sips, H. J, (2005),** "The BitTorrent P2P file-sharing system: Measurements and analysis", *4th International Workshop on Peer-to-Peer Systems*, pp. 205-216.

[2]. **Portmann, M.; Sookavatana, P. ; Ardon, S.; Seneviratne, A. (2001),**" The cost of peer discovery and searching in the Gnutella peer-to-peer file sharing protocol", *9th IEEE International Conference on Networks*. pp.263-268.

[3]. **Bin Tang ; Zongheng Zhou ; Kashyap, A. ; Tzi-cker Chiueh , (2005),** " An integrated approach for P2P file sharing on multi-hop wireless networks", *IEEE International Conference on Wireless And Mobile Computing, Networking And Communications*. Vol.3, pp. 268-274.

[4]. **Demchenko, Y., D.R. Lopez, J.A. Garcia Espin, C. de Laat, (2010).** "Security Services Lifecycle Management in On-Demand Infrastructure Services Provisioning",*2nd IEEE International Conference on Cloud Computing Technology and Science* (CloudCom2010).

[5]. **Zhao, G., C. Rong, J. Li, F. Zhang, Y. Tang, (2010).** "Trusted Data Sharing over Untrusted Cloud Storage Providers", *IEEE International Conference on Cloud Computing Technology and Science*. pp. 97-103.

[6]. **William Acosta and Surendar Chandra, (2005),**"Unstructured PeertoPeer Networks Next Generation ofPerformance and Reliability", *IEEE INFOCOM*.

[7]. **Ke Xu, Meina Song, Xiaoqi Zhang, Junde Song, (2009),** " A Cloud Computing Platform Based on P2P ", *IEEE International Symposium on IT in Medicine & Education*,Vol.1,pp. 427-432.

[8]. **Ruixuan Li, Li Nie, Xiaopu Ma, Meng Dong, Wei Wang,(2011),** "SMEF: An Entropy Based Security Framework for Cloud-Oriented Service Mashup ", *Int.Conf on Trust, Security and Privacy in Computing and Communications*. pp.304-311.

## ABOUT THE AUTHORS

*Lalitha is working as Assistant Professor in CSE in JNTUA college of Engineering Anantapur. She received her B.Tech Degree from SITAMS, Chitoor in the year 2003 and M.Tech degree from JNTU Anantapur in the year 2005. She has 8 years of teaching experience. Her research interests are Distributed Computing, Peer to peer networks.*

Sujith is working as *Lecturer in the Department of Computer Science and Engineering, Jawaharlal Nehru Technological University, Anantapur, India.*