AN EFFICIENT MECHANISM FOR PROVIDING IDENTITY PRIVACY FOR BULK CLOUD INFORMATION SHARING WITH DIFFSERV ARCHITECTURE ON CLOUD INFRASTRUCTURE

Durairaj.M[1] *Chandrasekar.C*[2]

Abstract

Cloud computing enables the convenience system to offer an on-demand network access amongst shared set of cloud pools. The shared cloud on service based applications dynamically adapt to higher QoS requirement system. However the service based application on the cloud infrastructure does not enable richer privacy modelling while sharing bulk software information. The inefficiency on sharing bulk software information leads to the degradation of cloud infrastructure. A small number of privacy preserving data sharing method on cloud infrastructure works to validate network integrity level of the entire cloud software's. However, the software information sharing still needs identity privacy for efficient auditing of bulk information on cloud infrastructure. To achieve an effective sharing of the software information on cloud infrastructure Opportunistic Bulk Software Information Sharing (OBSIS) Mechanism is proposed in this paper. The mechanism OBSIS is integrated with DiffServ Architecture which provides richer privacy modelling while sharing the information on network based cloud infrastructure. To achieve this, the OBSIS mechanism is divided into two phases such as application specific bandwidth aware routing and added resources based bulk data share method. In the first phase of the OBSIS mechanism, software information sharing process is carried out using Association Share Load assessment with DiffServ network Architecture. With the application of Association Share Load assessment, higher bandwidth rate efficiency is attained and low latency software information sharing system with best effort of services. Subsequently, in the second phase of the OBSIS mechanism, added resources based bulk data share method in uses Bergh's demand support matrix method to provide the identity privacy for effective data sharing. Experiment is conducted to work on the factors such as data sharing rate, bandwidth availability and latency time.

Keywords: Bergh's demand support matrix, Cloud Computing, Load assessment, Opportunistic Bulk Software, DiffServ network Architecture, Association Sharing

I. Introduction

Information sharing in cloud has been an active area of research for the past few decades and is seamlessly achieving significant impacts from researchers and

[1]Research Scholar, Department of Computer science Karpagam University, Coimbatore, India.

durairaj.moorthy@gmail.com. Mobile : +91-9941155124

[2]Associate Professor, Department of Computer science Periyar University, Salem-636011, India. ravikasi2001@yahoo.com

Mobile : +91-9994599967

scientists community. The objectives of information sharing between the cloud and the cloud users are to manipulate the information required by the cloud users in a more significant manner and provide sophisticated means that are helpful in not only providing security for the cloud users but also maintaining privacy of the cloud users in cloud infrastructure.

A Decentralized Self-adaptation (DS) mechanism [1] was designed using market-based heuristics that in a way provided with the users to decide which application they require and select accordingly amongst the many on offer. Oruta [2] another method to preserve the privacy of cloud users were designed for sharing data with the cloud using a third party auditor (TPA). The method not only achieved effectiveness in computation cost but was also proved to be efficient while performing batch auditing. However, with shared dynamic groups, the method was proved to be inefficient. Scientific Cloud Computing (SCC) [3] was designed that created the virtual clusters in an automatic manner to perform batch processing in parallel manner improving and simplifying their distribution.

With the increasing use of cloud environment and the continuous paradigm of shift toward software as a service (SaaS) has posed an enhanced demand for assured information sharing (AIS). A cloud based AIS in [4] was designed that included a centralized cloud-based AIS system for the efficient sharing of information that were stored in multiple clouds in a more timely manner. With the more upcoming and updated advancements in technology in mobile device and with the increasing demand for high amount of sophisticated mobile applications, the trend towards computing have changed a lot. But with higher mobility rate, the system did not address the hybrid nature of information sharing. In order to highlight the challenges in mobility [5] addressed the method for mobile devices in cloud. However, the cost related to computation increased with the mobility rate.

Much attention has been received in the recent years with the introduction of cloud computing that has started to own and manage its own servers. In [6], the basic necessities of cloud environment in relation of biomedical field were discussed and were proved to be of significant purpose for the medical practitioners. But issues like scalability with respect to medical data were rarely discussed. The problem related to scalability was addressed in [7] by designing an accountability framework that in a way efficiently kept intact the cloud users' data usage with that of the available data in cloud with the aid of an object-centered approach. Though scalability was addressed, the concept of access control was not addressed. File Assured Deletion (FADE) [8] introduced separate policies for file access and ensured the integrity of users file by maintaining a quorum of key managers.

In this work, focus is made on effective information sharing between the cloud users and cloud on cloud infrastructure. The contributions of this work include the following: (i) To provide an effective software information sharing on cloud infrastructure using application specific bandwidth aware routing with DiffServ network architecture, (ii) To provide richer privacy modeling while sharing information on network by integrating OBSIS mechanism with DiffServ on network based cloud infrastructure, (iii) To obtain higher bandwidth rate efficiency and low latency software information sharing system using association share load assessment and (v) To provide identity privacy for effective data sharing using Bergh's demand support matrix method with the help of added resources based bulk data share method.

Based on the aforementioned technique described, we focus on providing an efficient mechanism for providing identity privacy for bulk cloud information sharing with DiffServ architecture on cloud infrastructure. Related work is discussed in Section 2. An elaborate description is provided in the forthcoming sections. The rest of this paper is organized as follows: The structure of our proposed mechanism and their service requirements with a detailed architecture are described in Section 3 that includes an overall architecture and its description. The design of application specific bandwidth aware routing is discussed in Section 3.1. The design of the added resource based bulkdata share method is described in Section 3.2. The design for experiment and results of performance evaluation are described in Section 4 and Section 5 respectively. Finally, Section 6 concludes the paper.

II. Related Works

One of the main problems to be solved in cloud environment is the accountability for information sharing. In [9] the accountability of information sharing was provided using Advanced Encryption Standard (AES) techniques by restricting the unauthorized access. However, inclusion of upper bound was not provided. In [10], an upper bound constraint based approach was designed that not only identified the datasets for preserving the privacy but also the intermediate datasets were also identified for privacy preserving ensuring privacy leakage constraints. However, storage issue was not touched. In [11], a mechanism for dynamic shared data in cloud storage was introduced with the aid of index table management method.

One of the next generation shift in computing technologies is considered to be as the cloud computing. Here both the applications and resources are provided to the users on demand throughput the Internet. In [12] a comparative analysis was made with regard to security for the data provided by the cloud and the user and vice versa and the corresponding privacy protection techniques. In [13], certain security issues related to cloud computing was discussed and a framework for secure based clouds was introduced with the aid of two layers, called as the storage layer and the data layer. However, providing end-to-end security even if certain parts of the cloud fail remained unaddressed.

A new privacy preserving mechanism was introduced in [14] that included three parties that included inner product of two vectors in the cloud setting. The mechanism not only included appropriate analysis but also provided performance analysis with the aid of holomorphic encryption. Followed by this security was said to be guaranteed. However, the malicious nature of the cloud remained unaddressed. Global Authentication Register System (GARS) [15] was designed for providing security in cloud environment to minimize the cloud material outflow risk. The method GARS proved to be secured and perform well at minimum time duration. In [16], trusted computing was introduced to minimize the Phishing attack and identity theft in cloud environment.

III. Integrating Opportunistic Bulk Software Information Sharing Mechanism with Diffserv Architecture

In this section, we design an efficient mechanism for integrating opportunistic bulk software information sharing (OBSIS) mechanism with DiffServ architecture to achieve effective software sharing of information on cloud infrastructure. This section further describe the design of DiffServ architecture on cloud computing with the representation of bandwidth aware routing and finally provided an overall structure of OBSIS mechanism with the help of a diagram.

The proposed OBSIS mechanism on cloud infrastructure with DiffServ architecture provides the solution to share software information with higher privacy level. The proposed Opportunistic Bulk Software Information Sharing mechanism is simple to implement with DiffServ architecture by achieving richer privacy modelling. Differentiated Services (DiffServ) is a network architecture developed on cloud infrastructure to improve the privacy level on sharing the software information to the end users. The enhancement of privacy level in OBSIS mechanism provides richer QoS with best effort on cloud services. The DiffServ architecture on the network based cloud infrastructure is illustrated in Fig1.

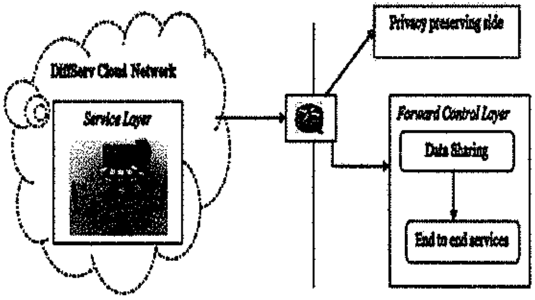

Figure1 : DiffServ Architecture on Cloud Computing

The DiffServ Cloud Architecture includes two types of layers called as the service layer and forward control layer as illustrated in above figure 1. The service layer on the network based cloud computing work to offer services to the end users with high privacy level. The privacy level in turn improves the information sharing properties. The OBSIS mechanism integrated with the DiffServ Architecture operates on software information sharing to different user application based on the requirements of individual users. OBSIS mechanism integrated with the

DiffServ is effectively configured to differentiate class of sharing based on bandwidth rate.

OBSIS mechanism performs the bandwidth aware routing operation using load assessment. Bandwidth Aware Routing developed in proposed work generates the route which is strongly adaptive on sharing the software information on cloud infrastructure. The bandwidth aware routing on cloud structure is illustrated in Fig2.

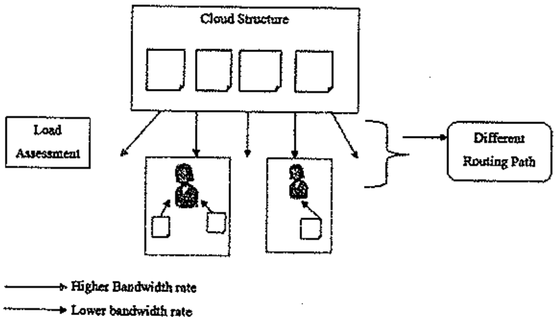

Figure 2 : Representation of Bandwidth Aware Routing on Cloud Structure

The bandwidth aware routing measures the load assessment of each route path to identify the higher privacy route rate. The users obtain bulk software information leading to effective cloud infrastructure. The load factor is measured for every route path of the cloud infrastructure to measure higher privacy range. The overall structural diagram of integrated OBSIS mechanism with DiffServ Architecture is depicted in Fig3.

As illustrated in Figure 3, the proposed OBSIS mechanism is integrated with DiffServ architecture to improve bulk software information sharing to end users. Different user requests through network based DiffServ architecture is obtained to share bulk software information. The user request of software for different types of applications is analyzed and bandwidth aware routing chooses the path to transfer bulk information. With this higher bandwidth rate improves information sharing with lesser latency time during the first phase.

In the second phase of the OBSIS mechanism, Added Resources based Bulk Data Share Method is developed. The resources are added in the cloud infrastructure to improve the software information sharing rate with higher privacy level using the Bergh's demand support matrix in the second phase. Bergh's demand support matrix achieves the result according to the requested users with higher demand of software information by adding the additional resources.

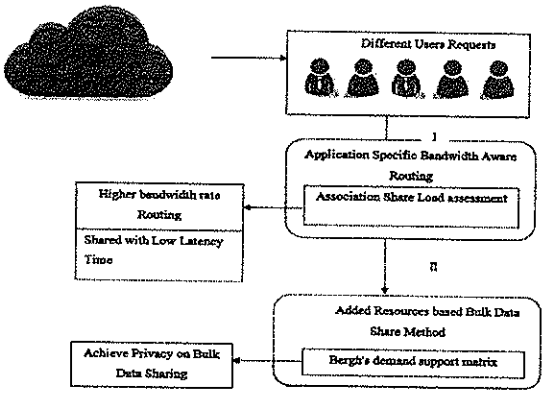
Different Users Requests
I
Application Specific Bandwidth Aware Routing
Association Share Load assessment
Higher bandwidth rate Routing
Shared with Low Latency Time
II
Added Resources based Bulk Data Share Method
Bergh's demand support matrix
Achieve Privacy on Bulk Data Sharing

Figure3 : Overall structural diagram of Integrated OBSIS mechanism with DiffServ Architecture

In the second phase of the OBSIS mechanism, Added Resources based Bulk Data Share Method is developed. The resources are added in the cloud infrastructure to improve the software information sharing rate with higher privacy level using the Bergh's demand support matrix in the second phase. Bergh's demand support matrix achieves the result according to the requested users with higher demand of software information by adding the additional resources.

3.1. Application Specific Bandwidth Aware Routing

The first phase involved in the OBSIS mechanism is the design of application specific bandwidth aware routing. The OBSIS mechanism develops application specific bandwidth aware routing for different users. The application specific bulk information about the software's is fetched through the network based cloud infrastructure. Bandwidth aware routing in OBSIS mechanism is carried out using the Direct Load Assessment Dependency 'DLAD' rate. The lesser dependency rate route path is removed from DiffServ network structure to reduce the latency time.

The bulk information sharing in cloud infrastructure using OBSIS mechanism entirely depends on the direct load assessment dependency rate. Let us assume that 'p1' and 'p2' are the paths to share bulk software information with dependence set. Then, the dependence set from 'd1','d2'....'dn', are formularized as,

$$DLAD\,(d1, d2, d3 \dots dn) = DLAD\,(p1) \cap DLAD\,(p2)) \qquad (1)$$

The dependency set'd' is used for higher bandwidth path chosen from the set of routing paths. The intersection operation is performed on (1) to identify the level of same bandwidth rate between path '1' and '2'. If the bandwidth of path '1' is higher than the bandwidth path '2' during the intersecting operation, then path 1 is chosen for bulk transfer of software information from the cloud structure to the end users. The algorithmic step involved in the design of bandwidth aware routing is described as,

```
// Bandwidth Aware Routing algorithm
Input: Applications 'A1, A2...An', User requests 'U1, U2...Un', Route path 'p1,p2....pn'
Output: Effective route path 'p' with higher bandwidth range is chosen
Step 1: While Cloud Server accepts the user requests
Step 2: Chooses the route path of higher load balancing rate
Step 3: Dependence set are computed directly using 'DLAD' measure
Step 3.1: DLAD measure follows the intersection operation
Step 4: If(DLAD ($p1$ )DLAD ($p2$ ))
Step 4.1: Transfer bulk software information through path '1'
Step 4.2: Else
Step 4.3: Transfer bulk software information through path '2'
Step 5: End If
Step 6: Load Assessment link the network based cloud structure to transfer the bulk information
```

The above algorithmic step briefs about the bandwidth aware routing. The bandwidth aware routing in OBSIS mechanism developed with the best effort of services on cloud infrastructure. The cloud server chooses the higher bandwidth rate path and transfers the bulk software information to the cloud client systems.

3.1.1 Association Share Load Assessment Procedure

The association in application specific bandwidth aware routing denotes the link path between the cloud server (i.e.,) cloud structure to the clients (i.e.,) end users. With this initially the load is assessed and then the association work is carried out to share bulk software information. As a result, the assessment procedure helps to reduce the latency time by removing the lesser load balancing route path.

$$\mathit{Load\ Assessment\ on}\,(BAR) = \sum_{d_i \in N} \frac{DLAD^2(d1,d2,d3...dn)}{DLAD\,(p1)*DLAD\,(p2)-DLAD^2(d1,d2,d3...dn)} \qquad (2)$$

The load assessment on direct dependency set is removed in such a way that minimal load balancing route paths are removed from the set of route path using DiffServ architecture. The higher load balance path produces higher bandwidth availability rate on application specific Bandwidth Aware Routing (BAR).

3.2. Added Resources based Bulk Data Share Method

The second phase involved in the OBSIS mechanism is the design of added resources based bulk data share method to not only improve the data sharing between the cloud and users but also to improve the privacy level. The chosen bandwidth aware routing for bulk information sharing is modified (i.e.,) added to the resources to improve the data share rate. The resources are added on the chosen path using the Bergh's Demand Support Matrix. The resources are added to the selected maximal bandwidth route path and computed as,

$$\begin{aligned}&\mathit{Identity\ Privacy\ Maintenance}\,(IPM)\\&\qquad = \mathit{Selected\ Path}\,(P) + \mathit{Added\ Resources}(R1, R2\ldots Rn)\end{aligned} \qquad (3)$$

The privacy maintenance is clearly explained using (3) by adding resources in the specific path. The specific path privacy on the network based cloud computing is defined as the Identity Privacy Maintenance (IPM).

3.2. Bergh's Demand Support Matrix

Bulk Data Share Method with high privacy maintenance works with Bergh's Demand Support Matrix range to satisfy the user demand software information request through supportive route path selection. The Bergh's matrix improves the privacy range of the system by providing the supportive resources factors R1,R2…Rn to the selected route path. In this way, the OBSIS mechanism achieves the goal of user demand and privacy factor on the network monitoring cloud infrastructure. DiffServ architecture integrated with the OBSIS mechanism achieves higher privacy on the bulk data sharing on the cloud infrastructure. Bergh's Demand Support Matrix satisfies the user demand with the minimal latency time.

IV. Experimental Evaluation

Opportunistic Bulk Software Information Sharing (OBSIS) Mechanism is integrated with DiffServ Architecture to monitor the network based cloud infrastructure while sharing bulk software information. The OBSIS Mechanism is experimented on JAVA platform using CloudSim simulator. Experimental machine is simulated with cloud data center comprising of 8 GB RAM for data sharing between the client machines. The goal of CloudSim is to offer a comprehensive and extensible simulation framework on cloud data center with the server and client systems.

For testing purposes on the existing and proposed system, CloudSim simulator is used for the network monitoring on the cloud infrastructure. Opportunistic Bulk Software Information Sharing (OBSIS) mechanism is compared against the existing Decentralized Self-Adaptation (DS) [1] mechanism and Privacy-Preserving Public Auditing for Shared Data (Oruta) [2] method. The experiment is conducted on factors such as data sharing rate, bandwidth availability, privacy preserved data sharing ratio, and latency time.

The data sharing rate DSR of OBSIS mechanism obtained using Bergh's Demand Support Matrix is the difference between the data requested by the user $Requested\ data_{user}$ and the data provided by the cloud $Provided\ data_{cloud}$. It is measured in terms of percentage (%)

$$DSR = (\frac{Requested\ data_{user} - Provided\ data_{cloud}}{Requested_{user}} * 100 \qquad (4)$$

$$BA = \frac{AU_{BIS}}{Total\ Bandwidth_{avail}} \qquad (5)$$

The bandwidth availability BA is the ratio of average utilization for bulk information sharing AU_{BIS} to the total bandwidth available $Total\ Bandwidth_{avail}$ in cloud infrastructure. It is measured in terms of bits per second.

Privacy preserved data sharing ratio is the ratio of data being shared between the cloud infrastructure and cloud users using the Bergh's matrix. This is measured in terms of percentage (%). Latency time LT using OBSIS mechanism refers to the difference between the time interval it takes to place the request of bulk information from the cloud by the cloud user $Bulk\ information_{req}$ and the time when the bulk information is obtained from the cloud $Bulk\ information_{res}$.

$$LT = (Bulk\ information_{req} - Bulk\ information_{res}) \qquad (6)$$

V. Results Analysis of Obsis Mechanism

The OBSIS mechanism is analyzed against existing Decentralized Self-Adaptation (DS) [1] mechanism and Privacy-Preserving Public Auditing for Shared Data (Oruta) [2] method. Each method has its own respective data sharing rate with respect to the requested data by the cloud owners in cloud infrastructure. The existing and proposed result is analyzed with the help of table values and graph points. Table 1 tabulates the data sharing rate with respect to the requested data by the cloud users in cloud infrastructure. We make a comparison of our model OBSIS against the DS and Oruta method.

Table1 : Tabulation for Data sharing rate

Requested data by cloud users (MB)	Data sharing rate (MB)		
	OBSIS mechanism	DS mechanism	Oruta method
500	485	445	425
1000	920	900	800
1500	1450	1300	1150
2000	1900	1750	1550
2500	2450	2250	2000
3000	2690	2850	2550
3500	3400	3300	3100

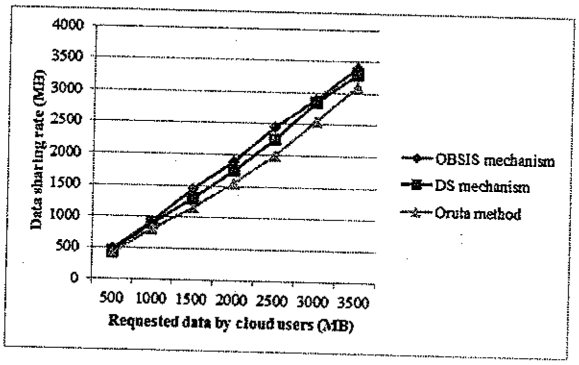

Figure 4 : Measure of data sharing rate with respect to requested data by cloud users

Fig4 illustrates the data sharing rate measured using OBSIS mechanism. Comparisons of data sharing rate are made with two other methods, Decentralized Self-Adaptation (DS) [1] mechanism and Privacy-Preserving Public Auditing for Shared Data (Oruta) [2] method. From the figure it is illustrative that the data accuracy rate is higher using OBSIS mechanism. This is because of the application of Bergh's Demand Support Matrix, with the help of added resources improves the data sharing rate between the cloud and the users by increasing the data accuracy rate. Also with the introduction of Bergh's Demand Support Matrix, the user demand software information request through supportive route path selection increases the data sharing rate by 1 – 10 % and 8 – 20 % comparing DS and Oruta respectively.

The bandwidth availability of our OBSIS mechanism is presented in table 2. It is significant to identify that the bandwidth availability is improved using OBSIS mechanism than compared to the state-of-art works.

Table 2 : Tabulation for bandwidth availability

Users (U)	Bandwidth availability (bps)		
	OBSIS mechanism	DS mechanism	Oruta method
20	185	165	161
40	170	155	148
60	164	160	151
80	182	172	165
100	168	158	152
120	165	145	138
140	162	142	132

The figure 5 describes the bandwidth availability based on the different number of users. As the number of users is increased, the bandwidth availability is also reduced. But comparatively, the bandwidth availability of data sharing is increased using the OBSIS mechanism compared to the state-of-the-art works which shows to be improved gradually. This is because with the application of Direct Load Assessment Dependency 'DLAD' rate, the bulk information sharing in cloud infrastructure is carried out using the intersection operation for different number of users. As a result, the bandwidth availability is increased by 12 – 12 % compared to DS [1] mechanism and 7 – 18 % compared to Oruta [2] method respectively.

Table 3 describes the privacy preserved data sharing ratio efficiency on different users where different users get entered into the cloud infrastructure. As the number of users increases, the privacy preserved data sharing ratio is also improved.

Figure 6 describes the privacy preserved data sharing ratio with respect to the different number of users images using OBSIS mechanism and comparison is made with the existing DS [1] mechanism and Oruta [2] method respectively. From the figure it is illustrative that the privacy preserved data sharing ratio is improved using the proposed OBSIS mechanism when compared to two other existing methods.

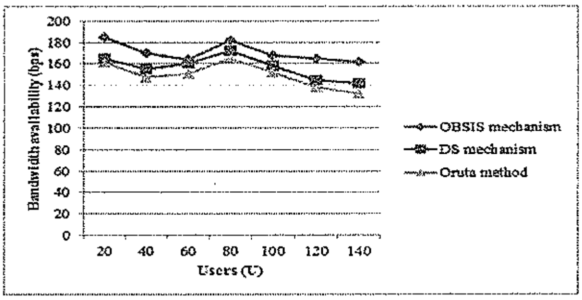

Figure 5 : Measure of bandwidth availability with respect to different users

Table 3 : Tabulation for privacy preserved data sharing ratio

Users (U)	Privacy preserved data sharing ratio (%)		
	OBSIS mechanism	DS mechanism	Oruta method
20	55.25	50.23	47.20
40	62.45	57.33	53.30
60	68.35	63.23	60.20
80	60.45	55.32	52.29
100	72.35	65.22	62.19
120	74.55	69.41	66.38
140	71.25	65.11	62.08

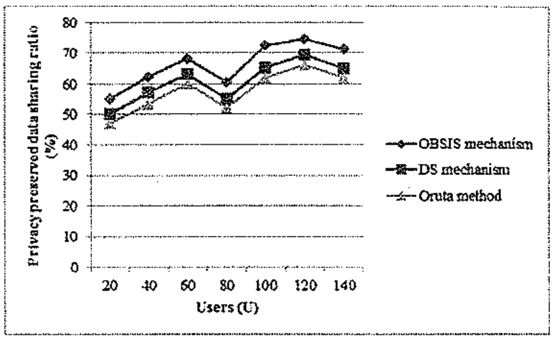

Figure 6:Measure of privacy preserved data sharing ratio

In OBSIS mechanism, the support resource factors are used for the selected route path using Bergh's matrix. Furthermore, the selected bandwidth aware routing for bulk information sharing is modified with the resources using DiffServ architecture to improve the privacy preserved data sharing ratio. Also the resources are added on the chosen path using the Bergh's Demand Support Matrix. With this the identity privacy maintenance is evaluated to improve the privacy preserved data sharing ratio by 7 – 9 % and 10 -14 % compared to DS and Oruta respectively.

Table 4 : Tabulation for latency time

Requested data by cloud users (MB)	Latency time (ms)		
	OBSIS mechanism	DS mechanism	Oruta method
500	45.52	40.47	36.45
1000	51.35	46.30	40.28
1500	58.45	53.40	47.38
2000	65.44	60.39	54.37
2500	62.35	57.29	51.27
3000	68.45	63.39	59.37
3500	74.55	69.42	60.40

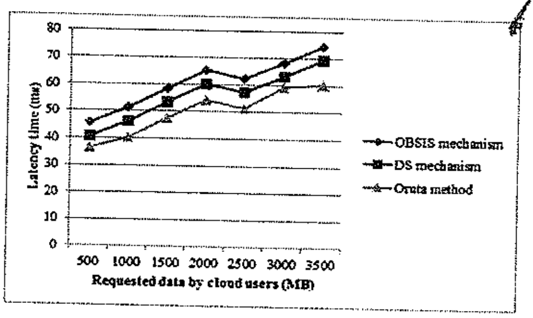

Figure7 : Measure of latency time

Table 4and Figure 7 illustrates the latency time based on the requested data made by the cloud users in the range of500 to 3500 MB. The OBSIS mechanism with DiffServ architecture in cloud infrastructure performs the association between the cloud server and the cloud clients using application specific bandwidth aware routing. As a result, the latency time using the OBSIS mechanism is reduced by 6 – 11 % compared to DS [1]. The load assessment on direct dependency set using OBSIS mechanism is removed in such a way that minimal load balancing route paths are removed using DiffServ architecture followed by which the association is carried out to share bulk software information that finally reduce the latency time by 13 – 21 % compared to Oruta method.

VI . Conclusion

We have presented an efficient mechanism for providing identity privacy for bulk cloud information sharing with DiffServ architecture on cloud infrastructure. First, we proposed an application specific bandwidth aware routing method for performing software information sharing based upon the different dependence sets from differing set of routing paths using the Bandwidth Aware Routing algorithm. With this, higher bandwidth rate efficiency was attained and low latency software informationsharing system with best effort of service was achieved. We also proposed added resources based bulk data share method, to provide the identity privacy for effective data sharing using Bergh's demand support matrix method which basically consists of identity privacy maintenance and supportive resource factors. Then user demand software information request is performed using the supportive route path selection for differing number of users. Moreover, to provide identity privacy for effective data sharing using Bergh's demand support matrix method the resources added to significantly increase the privacy modeling at a relatively lower latency time. Experimental results demonstrate that the proposed OBSIS mechanism only leads to noticeable improvement over the parameters data sharing rate and privacy preserved data sharing ratio, but also outperforms latency time required to fetch the user requests from the cloud infrastructure in a relatively lesser amount of time compared over other methods, namely, DS and Oruta.

References

[1] Vivek Nallur, Rami Bahsoon, *"A Decentralized Self-Adaptation Mechanism For Service-Based Applications in The Cloud,"* IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, 2012

[2] Boyang Wang, Baochun Li, and Hui Li, *"Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud,"* IEEE Transactions on Cloud Computing, (Volume: 2, Issue: 1), 2014

[3] K. Jorissen, F.D. Vila, J.J. Rehr, *"A high performance scientific cloud computing environment for materials Simulations"*, Computer Physics Communication, Elsevier, Apr 2012

[4] Bhavani Thuraisingham, Vaibhav Khadilkar, Jyothsna Rachapalli, Tyrone Cadenhead, Murat Kantarcioglu, Kevin Hamlen, Latifur Khan, and Farhan Husain, *"Cloud-Centric Assured Information Sharing"*, © Springer-Verlag Berlin Heidelberg 2012

[5] Abdullah Gani, GolamMokatderNayeem, MuhammadShiraz, MehdiSookhak, Md Whaiduzzaman, SulemanKhan, *"A review on interworking and mobility techniques for seamless connectivity in mobile cloud computing"*, Journal of Network and Computer Applications

[6] Arnon Rosenthal, Peter Mork, Maya Hao Li, Jean Stanford, David Koester, Patti Reynolds, *"Cloud computing: A new business paradigm for biomedical information sharing"*, Journal of Biomedical Informatics, Aug 2009

[7] Smitha Sundareswaran, Anna C. Squicciarini, and Dan Lin, *"Ensuring Distributed Accountability for Data Sharing in the Cloud"*, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 9, NO. 4, JULY/AUGUST 2012

[8] Yang Tang, Patrick P. C. Lee, John C. S. Lui, Radia Perlman, *"Secure Overlay Cloud Storage with Access Control and Assured Deletion"*, IEEE TRANSACTIONS ON DEPEDABLE AND SECURE COMPUTING, Feb 2012

[9] K.Rajendra Naidu1, N.Naveen Kumar, *"Ensuring Accountability for Data Sharing in Cloud"*, International Journal of Computer Trends and Technology (IJCTT) – volume 9 number 4– Mar 2014

[10] Xuyun Zhang, Chang Liu, Surya Nepal, Suraj Pandey, Jinjun Chen, *"A Privacy Leakage Upper-bound Constraint based Approach for Cost-effective Privacy Preserving of Intermediate Datasets in Cloud"*, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEM VOL:24 NO:6 YEAR 2013

[11] Ohmin Kwon, Dongyoung Koo, Yongjoo Shin, and Hyunsoo Yoon, *"A Secure and Efficient Audit Mechanism for Dynamic Shared Data in Cloud Storage"*, Hindawi Publishing Corporation, The Scientific World Journal Volume 2014,

[12] Yunchuan Sun, Junsheng Zhang, Yongping Xiong, and Guangyu Zhu, *"Data Security and Privacy in Cloud Computing"*, Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2014

[13] Kevin Hamlen, Murat Kantarcioglu, Latifur Khan, Bhavani Thuraisingham, *"Security Issues for Cloud Computing"*, International Journal of Information Security and Privacy, 4(2), 39-51, April-June 2010

[14] Gang Sheng, TaoWen, Quan Guo, and Ying Yin, *"Privacy Preserving Inner Product of Vectors in Cloud Computing"*, Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2014

[15] Chih-Yung Chen and Jih-Fu Tu, *"A Novel Cloud Computing Algorithm of Security and Privacy"*, Hindawi Publishing Corporation Mathematical Problems in Engineering Volume 2013

[16] Eghbal Ghazizadeh, Mazdak Zamani, Jamalul-lail Ab Manan, andMojtaba Alizadeh, *"Trusted Computing Strengthens Cloud Authentication"*, Hindawi Publishing Corporation, The Scientific World Journal Volume 2014

Author's Biography

Durairaj M, Tamilnadu, India, 05July 1982. I received Bachelor of Science (B.Sc) in Computer Technology (Regular) from KONGU Engineering College, Perundurai, Tamilnadu, India in 2002. I completed Master of Computer Applications (M.C.A) in Computer Application (Regular) from Kongu Engineering College, Perundurai, Tamilnadu, India in 2005. I have received Best Student award in B.Sc and MCA courses.

I am having 9 years' experience in Software Development of Network Management Product. Currently I am working in Anuta Networks Pvt Ltd, Bangalore as a Senior Software Development Engineer. Previously I have worked in HCL Cisco System as Senior Technical Lead.

Dr.C.Chandrasekar, Tamilnadu, India, 16 May 1972. He received Master of Computer Applications (MCA) from Madurai Kamarajar University, Madurai, Tamilnadu, India and Doctor of Philosophy (PhD) from Periyar University, Salem, Tamilnadu, India.

He has 17 years of experience in both Teaching and research. Currently he is working as Associate Professor in Department of Computer Science from Periyar University, Tamilnadu, India. Previously he worked as Assistant Professor in K.S.Rangasamy College of Technology.