ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 7, July 2013

MANAGING ACCOUNTABILITY FOR APPLICATION SHARING IN THE CLOUD USING PUSH-PULL KEY MATCHING ALGORITHM

Satpalsing D. Rajput[1], Ashish T. Bhole[2]

P.G. Student, Department of Computer Engineering, SSBT's COET Bambhori, Jalgaon, Maharashtra, India[1]

Associate Professor, Department of Computer Engineering, SSBT's COET Bambhori, Jalgaon, Maharashtra, India[2]

Abstract: Cloud is large number of computers connected by the communication network. Cloud Computing is the use of Computing resources i.e. hardware and software through the internet. A major feature of the cloud services is that users' application are usually processed remotely in unknown machines which users do not own or operate. While enjoying the convenience brought by this new emerging technology, users' fears of losing control of their own application (particularly, financial and health data) which can become a significant barrier to the wide adoption of cloud services. To address this problem, we propose a Push-pull key matching algorithm to correctly maintain the Log thereby improving the performance of System.

Keywords: Cloud Service Provider, Cloud Subscriber, Audit, Push-Pull, Key Matching.

I. INTRODUCTION

Cloud is use of computer Resources over the internet on needed basis or Cloud computing is service over internet.

A. Cloud Services

There are different cloud services according to the need of user these Cloud services are categories as a Software as a service which is located in upper layer, after that Platform as a service situated on middle layer and finally Infrastructure as a service located on lower layer. According to their application these services are used in our paper we Use the Software as a service. and these Logs are embedded in Saas Service.
Basically in our application we perform the accountability on application So this application we use as a filtering purpose where the Users need are taken in Consideration only that application is accessed by that User.

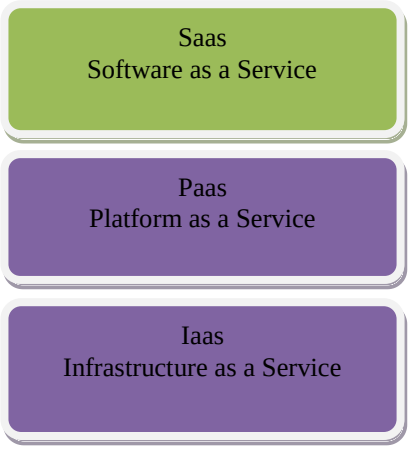

Fig. 1 Different Cloud Services

1. Software as a Service

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 7, July 2013

Software as a service in which software is installed and that will be maintained by the Cloud Service provider .subscriber. Only use that service from client node [7].there are many examples i.e. saleforce.com, Google Apps

2. *Platform as a service*

In platform as a Service the application is deployed on the Cloud so there will be no need to bring the underlying hardware and software this application is built on the Cloud
Service provider cloud subscriber only pay that cost and enjoy the service. For example Planet Lab [7].

3. *Infrastructure as a service*

Infrastructure as a service in which the application requires several storage server, switches, and routers. Now according to the capability of application it will use several resources which are available on Cloud service provider [7].

B. Authorization

Authorization is most useful partin access type mechanism in which there are different access type i.e. Update,View,insert,delete in which it basically depends upon which user is interested to access that data [2] .Basically cloud service provider and Cloud subscriber these two aspects are communicate with each other. Application is used as filtering purpose in which data is located in the database here accountability on the application is most important because when Cloud subscriber request any other data in the Database to Retrieve that bulk data for many subscribers required lot of time and also it reduces the performance of system to take this point in account we analyse the mechanism of accountability on Application Accountability is used to manage the Log because whenever we use the service that require the Access type, User ID, time required and address. Whenever user login at that time the UID .This total time constraints are analyses by timestamp[2] in which the amount of time which is required for executing that service it will be calculated. Here total timing required is calculated by creation timing and Receipt timing. There are different number of features are available in this access information.

C. Flexibility

Same Contents are access by different users that may lie different policies. Easy to operate this system is purely based on log management So it will be easily operated by any other person.

D. Accountability

Accountability is basically used to create log and Accountability is use to avoid Fraudulent access by any other third party. In Cloud accountability whenever several things are discussed with the CSP [9] then that user is authorized user to access that application but many time another access happen so In this method we maintain the Log base on that only authorized user can access application.

E. PUSH –PULL Mechanism

Push Pull [4] mechanism is to performing request from client to the server in this Cloud Subscriber request to the Cloud service provider that is message push to the server similar like that remote procedure Call. When this CSP receives this Request then it trace the information from given request and after that returns to the Cloud Subscriber. And in Pull mechanism the cloud subscriber enjoying the respond whatever the information located by its Server. Server giving the feedback in that instance of Server [5].
There are three different Push pull methods.

1. *Pure-Push method*

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 7, July 2013

In pure Push method whatever the request available in request queue that are not taken in consideration. Simply Server only broadcast data with Push bandwidth=100% at that time Pull bandwidth is=0%.only a front Channel used for broadcasting mechanism there will be no roll of back Channel for Communication.

2. *Pure Pull method*

In Pure Pull method only all bandwidth for responding operation dedicated to pull method. The page which is pull by one Client that can be accessed by front Channel.

3. *Partial Push Pull method*

Here Client will be send the Pull request from back Channel while the respond travel on front channel means at a one-time both this operation perform simultaneously So it also called as partial push pull method.

F. Log Creation

Log manager performs vital role for generating Log in this method Log manager track different four attributes that are nothing but UID, Access Type, Timestamp, and it's Location.

G. UID

Whenever the services require the user then there relevant Access facilities are discussed with the application owner. At that time the access specification will be given by the Cloud service provider that will referred as CSP.

H. Access Type

1. View

After checking the User ID it's relevant activity that whether that user having several access that he can view the application but can't executes his request. In this process of access log user try to print screen that application but it can't possible because it only view the application not save that application.

2. Download

The main thing of Downloading is that link is given to the User if in service level agreement user commit the Download information then only that link is visible to that user if user not commit that download link then he is unable to click that link. The application are taken in account. In this User click the link but here we maintain the download categories if that categories included such a services that user already register to that application owner then that agree to click on download link only that particular user who provided that access.

3. Timestamp

Every application provides a session time according to that service if this access duration will be defined by using [2]. That access time will be grant from the time interval at which the request comes to the Application cloud server. The network time protocol is used to maintain the access log according to time constraint of that application.

4. Location

In this IP and MAC lookup record are registered and according to that the shortest path algorithm followed that which is nearest location to access that record that will be taken in account. This IP and MAC lookup entries are also most important to track exactly from where the legitimate user break the policy of Log manager.

II. PROBLEM STATEMENT

A. More time required for Generating Log

Accountability is generally referred by Log. In existing system there was problem that more time was required when it generates the Log [1]. Generally this Log file created on every data suppose if the Cloud subscriber retrieves any data from Cloud service provider then Log file attached on that data. Generally user only required data there will be no need of Log information. If the size of data is large according to the need of user then abruptly the large log file is also attached so for copying that log file more time is required.

B. More time required for merging the files

For merging the log files if the log file size is small then merging time is small but if the size of log file increases then tremendously it affect on merging time.

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 7, July 2013

C. Performance of System Decreases as data Increases

Generally performance is the ratio of expected output and total load of system. If load or files are large then CSP performance is poor [1].

D. Problem of JAR File Creation

Each time whenever data retrieve from Database the JAR file or Log File attached to that data so it will be more hectic process when log file is large this problem is overcome by our System due to which application is used as filter So there will be no problem because most of the restrictions are provided by application .

III. RELATED WORKS

A. .Average time auditing mechanism

When a new user Subscribes First time at that time the ideal time required to submit the request to the Cloud service provider is noted. This timing information of Subscriber calculate by timestamp protocol i.e.Tsubscriber when it meets to the CSP then the time at which the request visited to the registration manager of CSP is TCSP.Then registration manager calculate timing for i.e T_{CSP}-$T_{Subscriber}$ that ideal time noted to that registration manager at the time of registration. Registration manager sends this information to the log manager. This auditing is important at every time because if any other variation in the timing results in any fraud access or network problem. This is use for diagnosis of security problem. When Subscriber first time meets to the CSP then Tsubscriber sends the Request by putting the timing information that timing information submitted to the Registration manager of CSP.Then it calculate the timing i.e$T_{Subscriber}$ and $T_{CSP.}$after that it calculate average waiting time of CSP i.e$T_{Subscriber}$-T_{CSP}.

B.Service Level Agreement

Quality of Service is important aspect in Cloud. This QoS maintained by Service level agreement. This agreement is Nothing but the negotiation process. Whatever things which are discussed before that it is not possible to fulfill all Expectation of cloud subscriber for that purpose a formal negotiation process conducted in between Cloud service provider and cloud subscriber and at that time there will be commitment between Cloud service provider and cloud subscriber is known as service level agreement. Generally SLA includes average response time of system and also throuput these points are taken in consideration.
These Service Level agreement consist following parameters in auditing mechanism

1. Accomplishment

If the Service Level agreement is violated, AUDIT will report this eventually, and it will produce evidence of the violation.

2. Correctness

If the Service level Agreement is not disturbed means there will be no other access of fraudulent user and the auditing mechanism successfully conducted, AUDIT will not report a violation.

3. Inspection

Any evidence of an alleged violation can be checked independently by a third party, even If the third party trusts neither the customer nor the provider.

IV. PROPOSE METHODOLOGY

ISSN: 2319-8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 2, Issue 7, July 2013

A. Architecture of Accountability for Application Sharing in Cloud

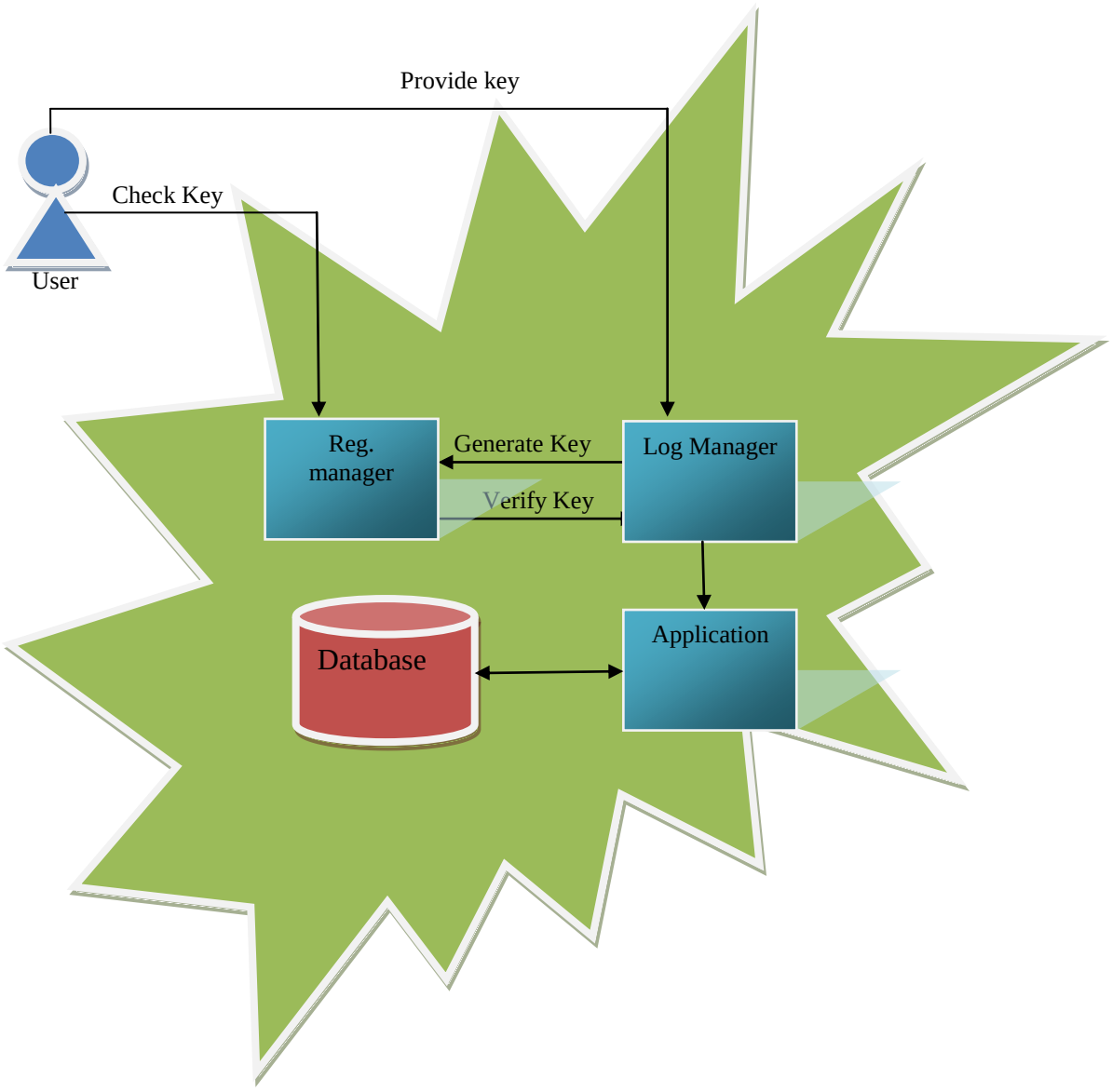

Fig. 2 Accountability for Application Sharing in Cloud

In above fig 2. When a new user meets to the Application server of CSP then it first registered to the Registration manager. When new user comes to the registration manager it first check that it will be already registered or not if already registered then request will be handled by log manager instead of registration manager. If the user is new then according to that user id the access will be specified to that user. Then UID,Accesstype, Time, and address information send to that log manager according to these information log manager generates token and that will be submitted to the registration manager for Auditing purpose and another to that application in which according to that accountability the access in that application are specified.This method is most important instead of creating the log on data we perform the log on application So this application is used as filter for retrieving the data means whatever things which are Committed in agreement only that part is accessed.

B. Push Pull Key Matching Algorithm

Size**:** Registration Server give the Log File according to UID of the given by User [1].
Time=maximum time required for travelling from Client to server.

Address=IP address and MAC address
Pull=Acknowledgement Received from Application Server
EC=Error Correcting bit

1. Start
2. KEY =(UID, Access Type, Time, Address)
3. if((TServer-TClient)<time) && (pull==0)) then
4. Key=ENCRYPT(KEY)
5. If PING User-Registration Manager then
6. PUSH EC_{client}(Key)//Registration Manager Broadcast Same encrypted key to the Log Manager and application User.
7. Else
8. Exit(1)
9. End if
10. If PING Registration Manager-Log Manager then
11. PUSH $EC_{Log\ manager}$(Key)
12. Else
13. Exit(1)
14. End if
15. PING USER- LOG MANAGER//When User Ping to the Log Manager
16. PUSH ECclient(Key)
17. If(($EC_{Log\ Manager}$(Key)==EC_{Client} Key(Key)) then
18. If PING Log manager-Registration manager then
19. Key=Decrypt(Key)
20. PUSH(Key) //Push key from Registration manager to application manager
21. Else
22. Exit(1)
23. End if
24. Else
25. Exit(1)
26. End if
27. End if
28. if((TServer-TClient)>time) &&(pull!=0) then
29. EC(log):=NULL
30. Tprev=TS(NTP)
31. Else
32. Exit(1)
33. End if
34. End if
35. Stop

In Push Pull key matching algorithm the key is generated by UID,Access type, Time and AddressKey is generated by UID, Access type, Time and address if Tserver-Tclient is minimum than expected time the no network issues So the key is encrypted after that Registration Manager Broadcast Same encrypted key to the Log Manager and application User. If key of Log manager and user key matched then decrypt the key. Another case is that if Tserver-Tclient is larger than expected time then permission for that log is denied.

C. Algorithm for managing Load on Server

Generally in Distributed Cloud more number of users are performing the operation with the CSP, Problem is occur when more number of user request same content from the server [4].

1. Start
2. When request meet to the CSP and that will be already registerd user then
3. Cloud Subscriber :=Pull
4. Else
5. If CSP load< Moderated load
6. Client:=Push
7. Else
8. If CSP Load>moderated Load then
9. Repeat

10. Reduced Server load()
11. Until CSP load, timed out
12. Else divert some Push Subscriber to pull.
13. End if
14. End if
15. End if
16. stop

In this algorithm when any other request meet to the cloud server and that will be already register by the log manager then Cloud Subscriber only performing pull operation performed. If CSP load is minimum than that of moderated load then server is light weighted so whatever the request available in request queue that will be executed. If in Case load on the server increases i.e CSP load is more than that of expected load then reduced server load if due to higher load the CSP is timed out then Switch Push mode to pull mode.

D. Temporal Identity Based Encryption

This Policy generates temporary policy based on the Identity of user. This require the following parameter such as UserId, Access type as a Timing information about both the User and Registration Server and Location determine by the IP and MAC address [8].
Step 1: Start
Step3: GenKey (MSK,uk,A,T,L):Registration server Takes the user's ID, Access Type A, Time information T and Location of User U.
Step 4: Encrypt (EK): Registration Manager Delegate Key to the user and also to the Log manager.
Step 5: Decrypt (LSK,EK): Log Manager Issues the Decryption key from Registration manager LSK over Encrypted key EK and both the keys in Log manager and User are decrypted.
Step 6: Stop

V. EXPECTED OUTCOME

i)Generally in Existing System there will be overhead of creating Log data Instead of Creating Log data we only making the Log of application by a simple thing so it reduced the log Creation time.
ii) It also made the JAR of each data for that purpose it require more time for creation of this JAR. This Problem overcomes by maintaining Log in application.
iii) Due to managing Load on Server by comparing with moderated load Push or Pull operation will be performed

VI. CONCLUSION

Push pull key matching algorithm is useful to manage the Log. Registration manager, Log manager that are help to filtering the fraud access of third party. Managing load on Server is helpful to improve the performance of Server. As in this method Logs are generated by application so instead of using JAR file Log we use Log on the application.

REFERENCES

[1] Sundareswaran, Smitha, Anna C. Squicciarini, Member, IEEE, and Dan Lin, "Ensuring Distributed Accountability for Data Sharing in the Cloud" IEEE Transactions on Dependable and Secure computing, Vol 9, No. 4, pp. 556 – 568, July/August 2012.
[2] P. Ammann and S. Jajodia, "Distributed Timestamp Generation in Planar Lattice Networks," ACM Trans. Computer Systems, vol. 11, pp. 205-225, August 1993.
[3] Punyada M. Deshmukh, Achyut S. Gughane, Priyanka L. Hasija, and Supriya P. Katpale, "Maintaining File Storage Security in Cloud Computing" International Journal of Emerging Technology and Advanced Engineering, vol 2,October 2012.
[4] Engin Bozdag, and Arie van Deursen, "An Adaptive Push/Pull Algorithm for AJAX Application" Delft University of Technology Software Engineering Research Group Technical Report Series.
[5] Qihua Wang and Hongxia Jin," Data Leakage Mitigation for Discretionary Access Control in Collaboration Clouds" June 2011.
[6] Madhan Kumar Srinivasan, K Sarukesi , Paul Rodrigues, Sai Manoj M, and Revathy P, "State of the art Cloud Computing Security Taxonomies a classification of Security Challenges in the present Cloud Computing Environment", International Conference on Advances in Computing, Communications and Informatics.
[7] Pradeep Kumar Tiwari, and Dr. Bharat Mishra.,*Cloud* "Computing Security issues, Challenges and Solution", International Journal of Emerging Technology and Advanced Engineering vol.2, pp. 306-310, Aug.2012.
[8] Yan Zhu,Hongxin Hu,Gail-Joon Ahn,Xiaorui Gong, and Shimin Chen, "POSTER: Temporal Attribute-Based Encryption in Clouds ",ACM 978-1-4503-0948-6/11/10,pp.881-310,Oct-2011
[9] Shucheng Yu, Cong Wang,Kui ren, and Wenjing Lou "Achieving Secure,Scalable.and Fine-grained Data Access Control in Cloud Computing", IEEE INFOCOM 2010