International Journal of Research in Engineering, Science and Management
Volume-1, Issue-11, November-2018
www.ijresm.com | ISSN (Online): 2581-5792

Anti-Collusion Access Control Data Sharing Scheme to Dynamic Group in Cloud Environment

Supriya U. Bharajkar[1], B. M. Patil[2]
[1]*Student, Department of CNE, MBES's College of Engineering, Ambajogai, India*
[2]*Professor, Department of CNE, MBES's College of Engineering, Ambajogai, India*

***Abstract*: System proposes a protected information sharing plan for element individuals. Initially, we propose a protected path for key dissemination with no safe correspondence channels, and the clients can safely acquire their private keys from gathering chief. In our proposed system we use three different entities data owner, group manager, cloud server and attacker is untrusted entity. In this module first data owner upload the data file to cloud server using cryptography algorithm once data has store into database, owner gets the notification about file storage successfully. The data owner having a full access of specific data file he can share or access, so data owner can share the any file to any group manager then it will automatically access to all group members. The shared group members can access each file to anytime by cloud server. In first phase if data owner revoke any user from access the file then he can't access such file. If he can try to generate any collusion attack using SQL injection queries, even our system will system will prevent such attacks. Second data owner can share and revoke file to individual user to specific group, and third once any user revoke system will automatically generate proxy key generation that means existing keys will expired. The overall approach improves the system efficiency as well security on drastic level.**

***Keywords*: cloud computing, data sharing, data integrity, authentication, anonymity, forward security.**

1. Introduction

In existing system a user can be a Data Owner and a Data Consumer simultaneously. Authorities are assumed to have powerful computation abilities, and they are supervised by government offices because some attributes partially contain users' personally identifiable information. The whole attribute set is divided into N disjoint sets and controlled by each authority, therefore each authority is aware of only part of attributes. A Data Owner is the entity who wishes to outsource encrypted data file to the Cloud Servers. The Cloud Server, who is assumed to have adequate storage capacity, does nothing but store them. Newly joined Data Consumers request private keys from all of the authorities, and they do not know which attributes are controlled by which authorities. When the Data Consumers request their private keys from the authorities, authorities jointly create corresponding private key and send it to them. All Data Consumers are able to download any of the encrypted data files, but only those whose private keys satisfy the privilege tree Tp can execute the operation associated with privilege p. The server is delegated to execute an operation p if and only if the user's credentials are verified through the privilege tree Tp.

2. Literature survey

Existing examination work is accessible in the parts of uprightness confirmation of outsourced information, data stockpiling security on untrusted remote servers and access control of outsourced data. The word cloud had come to make reference to extensive Asynchronous Transfer Mode frameworks. By 21st century, he term "disseminated registering" had showed up, yet key fixate at this moment was on Software as a Service (SaaS). In 1999, bargains force.com was made by Parker Harris, Marc Benioff. They used various headways of client locales, for example, Yahoo and Google! to association programs. They furthermore gave the thoughts like" On hobby" and" SaaS" with their bona fide association and customers that were compelling. Cloud data stockpiling (Storage as a Service) is an essential organization of dispersed figuring insinuated as Infrastructure as a Service (IaaS). Amazon's Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (S3) are comprehended occurrences of cloud data stockpiling. On the other side close by these preferences' appropriated figuring defies tremendous test i.e. data stockpiling security issue, which is a basic part of Quality of Service (QoS). At the point when customer puts information on the cloud as opposed to locally, he has no impact over it i.e. unapproved customers could change customer's information or ruin it and even cloud server scheme ambushes. Cloud customers are generally struggled with steadiness and the security of the information in the cloud. Amazon's S3 [8] is such a fair case.

Identity based cryptosystem, presented by Shamir [1], disposed of the requirement for confirming legitimacy of open key endorsements, administration of both time and cost expending. In an ID based cryptosystem, general key of each client is effectively process able from string relating to this present client's openly known personality (e.g., a private

location, an email address, and so on.). A private key generator (PKG), figures private keys from its expert mystery for all clients. This property maintains a strategic distance from the need of testaments and partners a verifiable open key (client personality) to every client inside of the framework. With a specific end goal to confirm an ID-based signature, unique in relation to the conventional for signature of open key based, one does not have to check endorsement first. And the end of authentication approval makes an entire confirmation handle more proficient, which will lead to significant recovery in correspondence and calculation at the point when countless are included (say, vitality utilization information partaking in brilliant matrix).

According to Rivest [2], the prime appearance of ring mark in 1994 and the official presentation in 2001.Ring mark is utilized (Actual personality of endorser is covered up). Ring mark is a gathering focused mark with security assurance on mark maker. He formalize the thought of a ring mark, which makes it conceivable to determine an arrangement of conceivable underwriters without uncovering which part really created the mark. Dissimilar to gathering marks, ring marks have no gathering chiefs, no repudiation techniques, no setup systems, and no synchronization- any client can pick any arrangement of conceivable underwriters that incorporates himself, and sign any of the messages by using his mystery key and others open keys, without taking their help.

Bresson, Stern and Szydlo [3], first upgraded their indicating in order to ring mark perspective that it holds under a totally weaker assumption, specifically the unpredictable prophet show rather than the ideal figure. By then utilize extensions to make ring marks suitable in practical circumstances, for instance, edge arranges or uncommonly delegated social occasions.

Liu and Wong [4], showed another however culminate model whose security level can be thought to be lying amidst for the most part used models. Fine-grained refinements on the security models is basic to make following a couple arrangements may be secure in a portion of the models however not in the others.

In Approach [5], Identity based ring mark was secure in self-assertive prophet model.

According to [6], first Identity based ring mark expressed to be secure in the standard model is a result of this paper under the trusted setup supposition. However their affirmation isn't correct and is pointed out by [9]

In [7], formalize the definition and security considerations for a forward-secure character based imprint arrangement, and after that add to a capable arrangement. All parameters that are accessible in our arrangement have, at most, log-squared multifaceted nature to the extent the total number of time periods. Without subjective prophets the arrangement is provably secure.

3. Proposed system model

In this system, there are exist 6 entities:

- Single i.e. global Certificate Authority(CA)
- Multiple Attribute Authority(AA's)
- Data Owner
- User
- Cloud server Trusted
- Third Party (TTP)

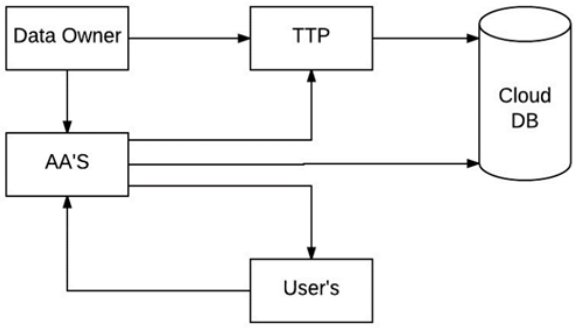

Fig. 1. Proposed system block diagram

The system will execute using below procedure:

1. AA registers to CA to get (aid,aid.cert)
2. User register to CA to get (uid,uid.cert)
3. User gets his/her SK from any t out of n Aas as well as from TTP.
4. Owners get PK from CA
5. Owners upload (CT) to the cloud server.
6. Users download (CT) from the cloud server.

- The system can perform Attribute revocation method can efficiently achieve both forward security and backward security. An attribute revocation method is efficient in the sense that it incurs less communication cost and computation,
- Cost, secure in the sense that it can achieve both backward security and forward security.

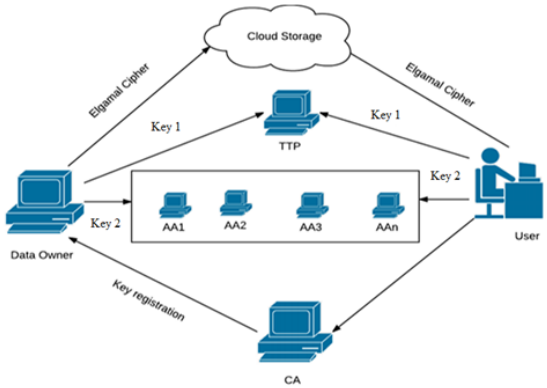

Fig. 2. Proposed system architecture

There are five types of entities in the system as in Fig 1: a certificate authority (CA), characteristic authorities (AAs), data owner (owners), the cloud server (server) and data consumers (users). The CA is a global trusted certificate

authority in the scheme. It sets up the system and accepts the registration of all the users and AAs in the system. For each legal user in the system, the CA assign a global unique user identity to it and also generates a global public key for this user. However, the CA is not involved in any attribute organization and the formation of secret keys that are connected with attribute [6] [8]. For example, the CA can be the Social Security Administration, an independent agency of the United States government. Each user will be issued a Social Security Number (SSN) as its global identity. Every AA is an independent attribute influence that is responsible for entitling and revoking user's attributes according to their role or identity in its domain. In our scheme, every attribute is associated with a single AA, but each AA can manage an arbitrary number of attributes. Every AA has full control over the structure and semantics of its attributes. Each AA is responsible for generating a public attribute key for each attribute it manages and a secret key. For each user reflecting his/her attributes. We propose a protected information sharing plan for element individuals. Initially, we propose a protected path for key dissemination with no safe correspondence channels, and the clients can safely acquire their private keys from gathering chief. In our proposed system we use three different entities data owner, group manager, cloud server and attacker is untrusted entity. In this module first data owner upload the data file to cloud server using cryptography algorithm once data has store into database, owner gets the notification about file storage successfully. The data owner having a full access of specific data file he can share or access, so data owner can share the any file to any group manager then it will automatically access to all group members. The shared group members can access each file to anytime by cloud server. In first phase if data owner revoke any user from access the file then he can't access such file. If he can try to generate any collusion attack using SQL injection queries, even our system will system will prevent such attacks. Second data owner can share and revoke file to individual user to specific group, and third once any user revoke system will automatically generate proxy key generation that means existing keys will expired. The overall approach improves the system efficiency as well security on drastic level.

A. Mathematical module

Let's,

D is denoted by dataset which includes the n number of paragraphs in file

D = {Ci1,Ci2,Ci3…..Cin}

Here, C is the intermediate module which holds the data processing for security as well as data privacy.

C = {C1,C2,C3…Cn}

C1= key generation

C2= encryption of data

C3= Authentication and Authorities verification phase

C4 = decryption of data

C5=Revocation phase

C6=Resign key generation

Here R is web base approach which handles the parallel searching, the result of query classified into n number of result pages. All R instances might be different authorities which will holds the data and when intermediate module generate the requires it will execute parallel.

R = {R1, R2, R3….Rn}

4. Algorithms

A. Elgamal Encryption scheme

Key Generation phase

Input: Plain text as text data d.

Output: a, b, p, g all which cointain public key, master keyand private key

Step 1: Initialize the random message from user as d. (it should be any kind of text data).

Step 2: Initialize a, b, p ,g for private key purpose.

Step 3: generate P as randomly base on bit length of d.
so Ans [] =GetRandomP (d.getbyte).bitlength base on probable prime no.

Step 4: p=Ans [0]
g=Ans [1]

Step 5: Generate a using P
a=Random A(p)
its calculate like p.bitLength ()-1,Random.

Step 6: Calculate b= calculateb(g, a, p);
So, b= g.modPow (a, p);

Step 7: Key generation done

Encryption

Input: Text data d, p, b, g

Output: cipher as C1, and C2.

Initialize Big Integer [] rtn = {null, null};

Message=d.getbytes ();

[] result= ElGamal.encrypt (message, p, b, g);

[] rtn = {null, null};

k = ElGamal.getRandomk (p);

C1 = g.modPow (k, p);

C2 = m.multiply (b.modPow (k, p)).mod (p);

Decryption

Input: input c1 and c2 as cipher a and p as private keys

Output: Plain text d.

Step 1: m = C2.multiply (C1.modPow (a.negate (), p)).mod (p);

Step 2: return m.

B. SQL Injection and prevention algorithm for Database Security

1: Procedure SPMA (Query, SPL[ ])
INPUT: Query=UserGenerated Query SPL[ ] =Static Pattern List with m AnomalyPattern

2: For j = 0 to m do

3: If (AC (Query, String. Length (Query), SPL[j][0]) = =0 then

4: Calc anomaly score

5: If ( ) Score Value Anomaly = Threshold

6: then
7: RetrunAlarm. Administrator
8: Else
9: Return Query. Accepted
10: End If
11: Else
12: Return Query .. Rejected
13: End If
14: End For
End Procedure

5. Implementation details

A. System requirements

- System interfaces: Windows Operating System
- User interfaces: User interface using Jsp and Servlet
- Hardware interfaces

Processor: - Intel Pentium 4 or above
Memory: - 512 MB or above
Other peripheral: - Printer
Hard Disk: - 40GB

- Software interfaces: Front End: Jdk 1.7.0,Netbeans 6.9.1 or Eclipse Juno,IE 6.0/above
- Back-End: MySQL 5.1
- Services: Amazon EC2 for cloud

6. Experimental result

For the system performance evaluation, calculate the matrices for accuracy. The system is implemented on java 3-tier architecture framework with INTEL 2.8 GHz i3 processor and 4 GB RAM with public cloud Amazon EC2 consol. For the system evaluation we create 2 machines on physical environment with Wi-Fi and 10 VM with Amazon EC2 as public cloud environment. After implementing some part of system we got system performance on satisfactory level. The below table shows the first algorithm performance for user plain data conversion as well encryption decryption.

A. System performance

Table 1
System performance (Estimated)

Data Size in MB	Encryption time (Milliseconds)		Decryption time (Milliseconds)	
	Existing	Proposed	Existing	Proposed
5	595	515	724	612
10	1120	1026	1132	1033
15	1680	1547	1687	1556
20	2260	2064	2231	2033

In second experiment Fig. 3, shows data encryption performance which works to show that the data it will encrypt in how much time in seconds. Suppose there is a 100kb data is encrypted in 150 sec so the result will display automatically in that time of encryption data from the users.

From Fig. 5 to Fig. 8 shows the how time will change when number of authorities has change for user process and verification.

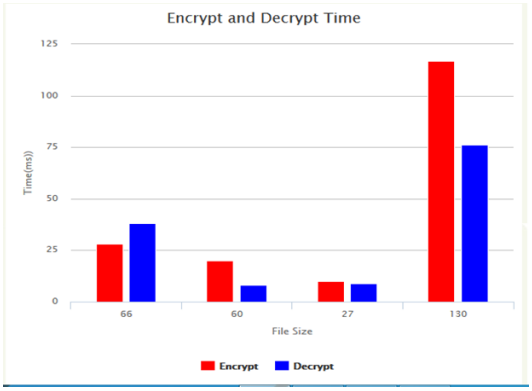

Fig. 3. File size in kb with required time in ms

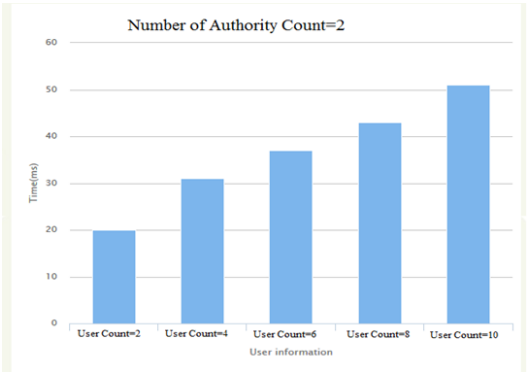

Fig. 4. Time required when AA=2

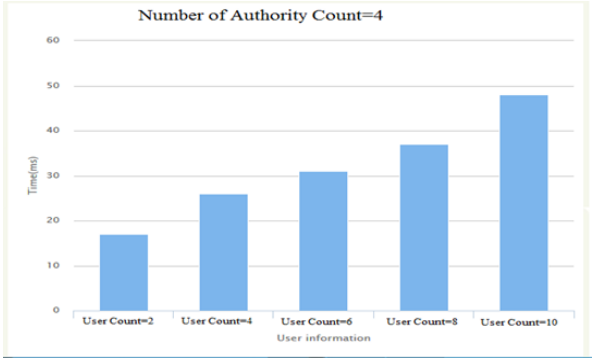

Fig. 5. Time required when AA=4

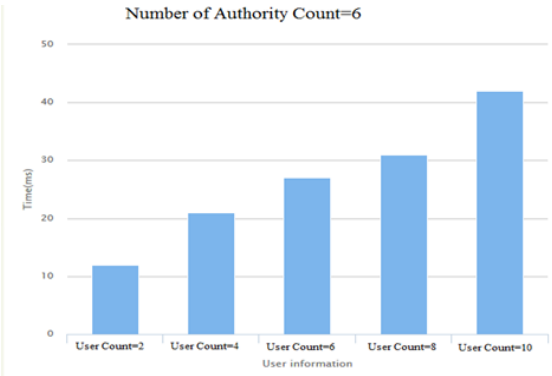

Fig. 6. Time required when AA=6

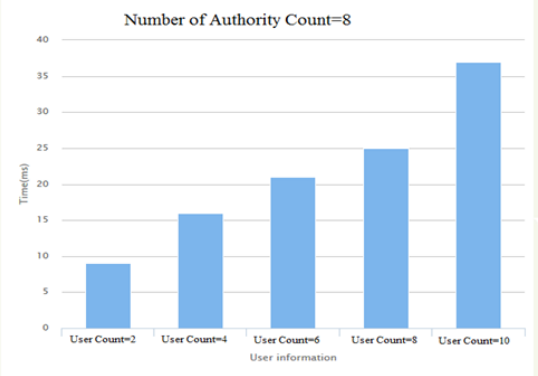

Fig. 7. Time required when AA=8

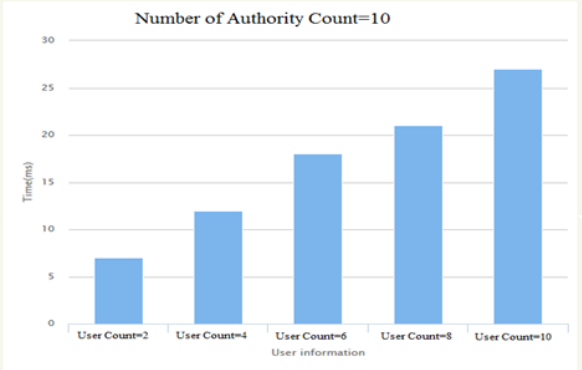

Fig. 8. Time required when AA=10

7. Conclusion

In this work system design a secure anti-collusion data sharing scheme for dynamic groups in the cloud. In our scheme, the users can securely obtain their private keys from data owner, Certificate Authorities and secure communication channels. Also, our scheme is able to support dynamic groups efficiently, when a new user joins in the group or a user is revoked from the group, the private keys of the other users do not need to be recomputed and updated. Moreover, our scheme can achieve secure user revocation, the revoked users can not be able to get the original data files once they are revoked even if they conspire with the untrusted cloud.

8. Future work

The current architecture is very efficient for security purpose, but sometime it's utilized multiple resources. When such system allocate multiple resources it will generate a lot of dependencies. For the next updation we can focus on minimum resource utilization with system flexibility like power, VM's, network, memory etc.

References

[1] A. Shamir. Identity-Based Cryptosystems and Signature Schemes. In CRYPTO 1984, volume 196 of Lecture Notes in Computer Science, pages 47–53. Springer, 1984.
[2] R. L. Rivest, A. Shamir, and Y. Tauman. How to Leak Secret. In ASIACRYPT 2001, volume 2248 of Lecture Notes in Computer Science, pages 552–565. Springer, 2001.
[3] E. Bresson, J. Stern, and M. Szydlo. Threshold ring signatures and applications to ad-hoc groups. In M. Yung editor, CRYPTO 2002, volume 2442 of Lecture Notes in Computer Science, pages 465–480. Springer, 2002.
[4] J. K. Liu and D. S. Wong. On the Security Models of (Threshold) Ring Signature Schemes. In ICISC 2004, Lecture Notes in Computer Science. Springer, 2004.
[5] F. Zhang and K. Kim. ID-Based Blind Signature and Ring Signature from Pairings. In ASIACRYPT 2002, volume 2501 of Lecture Notes in Computer Science, pages 533-547. Springer, 2002.
[6] J. Han, Q. Xu, and G. Chen. Efficient id-based threshold ring signature scheme. In EUC (2), pages 437–442. IEEE Computer Society, 2008.
[7] J. Yu, R. Hao, F. Kong, X. Cheng, J. Fan, and Y. Chen. Forward secure identity-based signature: Security notions and construction. Inf. Sci., 181(3):648–660, 2011
[8] Amazon.com, Amazon Web Services (AWS), 2008. http://aws.amazon.com.
[9] P. P. Tsang, M. H. Au, J. K. Liu, W. Susilo, and D. S.Wong. A suite of non-pairing id-based threshold ring signature schemes with different levels of anonymity (extended abstract). In ProvSec, volume 6402 of Lecture Notes in Computer Science, pages 166–183. Springer, 2010.