SRBAC : Secure Role Base Acess Control in Dynamic Cloud

Miss. Shubhada S. Paithankar[1], Prof. Pratap M. Mohite[2], Prof. Sathya Praveen D.[3]
[1]P.G. Student, [2]Assistant Professor, [3]HOD:
[123]Department of Computer Engineering,
[123]Shreeyash College of Engineering & Technology, Aurangabad, Maharashtra, India

Abstract- The Attribute Base Encryption (ABE)provides the secure data storage and data sharing to data owners in cloud environments without any third party dependency. The former ABE schemes contain justone authority to used for end user verification, which isalways generate a single-point bottleneck on both security as well as performance. Therefore, in existing too many multi-authority schemes are proposed, in which multiple authorities can maintain the privacy and verification of multiple attribute subsets. Even though, in many system single-point bottleneck issues does not solved. In this proposed research work, from system focus, tomulti authority verification scheme called Robust Access Control (RAAC) which is the part of Role Base Access Control (RAC), the system proposed a secure data transmission in cloud environments. In RAAC, taking benefit of (t out of n) threshold secret sharing, the master key and private keys can be shared amongst multiple authorities as well as middleware trusted party, and a authenticate user can generate his/her secret key by cooperating with any available authorities during the transaction. The elgamal encryption provides the highest security from data with two type cipher text policy and MK and PK with middle ware authorities. Furthermore, by capably uniting the conventional multi-authority scheme with RAAC, we design an approach that can verify multiple user parallel and provide consistent data to end users. The experimental analysis also shows the proposed schemes how better thatexisting approaches.

Keywords- RBAC, Elgamal encryption scheme; secure user Revocation; Proxy Key Generation, Role Base Access Control (RBAC)

I. INTRODUCTION

Forward data security during the data data transmission as well as storage in cloud environemnt its todays need for all applications. It moreover give of the authenticity and namelessness of the end user. Ring topology is the promising prospect to build an unknown and credible information sharing framework. It authorizations an information owner to thein mystery validate his information which can be put into the cloud for capability or examinationexplanation. The proposed framework work in all cloud environment with multi user verification appraoch. The Trusted Third Party (TTP) and multiple Attribute Authorities (AA) provide the user verification that can remove the end user certificate verification process that can reduce the internal overhead of system.

There ate too many schemes has developed by the existing authors for cloud data security. Some Identity base, Key Base, CipherText base etc. each system having some drawbacks like collusion issues [3], bottleneack issues, SQL injection issues etc. Every scheme find the new security appraoch for eliminate such drawbacks with new techniques. In this work, system propose a RAAC with CP-ABE access control plan, which eliminates the single-point queue on both security and execution. In this plan, numerous powers mutually deal with the entire property set however nobody has full control of a particular characteristic.

II. LITERATURE SURVEY

According to KaipingXue [1] propose a novel heterogeneous framework to remove the problem of single-point performance bottleneck and provide a more efficient access control scheme with an auditing mechanism. Our framework employs multiple attribute authorities to share the load of user legitimacy verification. Meanwhile, in our scheme, a *CA* (Central Authority) is introduced to generate secret keys for legitimacy verified users. Unlike other multiauthority access control schemes, each of the authorities in our scheme manages the whole attribute set individually. To enhance security, we also propose an auditing mechanism to detect which *AA* (Attribute Authority) has incorrectly or maliciously performed the legitimacy verification procedure.

Kan Yang and et. Al.[2], proposed a revocable multi-authority CP-ABE scheme, and apply it as the underlying techniques to design the data access control scheme. Our attribute revocation method can efficiently achieve both forward security and backward security. System also design an expressive, efficient and revocable data access control scheme for multi-authority cloud storage systems, where there are multiple authorities co-exist and each authority is able to issue attributes independently.

The system [3] proposed a secure way for anti collusionkey distribution without any secure third party channels, and the users can securely get their private keys from group owner. second, this method can propsed fine grained access control, any user in the group can use the source in the cloud and revoked users cannot access the cloud all over again after they are revoked. Thirdly, system can shield the scheme from collusion attack, that means that revoked users cannot get the actual data file even if they combine with the untrusted cloud. In this approach, by exploit polynomial capability, framework can complete a safe client negation conspire, Finally, this plan can accomplish fine efficiency, which implies past clients necessitate not to refresh their revoked from the group.

According to [4] proposes the major of key-approach feature which is based on KP-ABE with reflection of non-monotonic access structures and with regular ciphertext size. System also proposes the first Key-Policy Attribute-based Encryption (KPABE) method allowing for non-granted access structures (i.e., that may contain negated attributes) and with constant ciphertext size. Towards achieving this goal, system first show that a certain class of identity based broadcast encryption schemes generically yields monotonic KPABE systems in the selective set model. System then describes a new efficient identity-based revocation mechanism that, when combined with a particular instantiation of our general monotonic construction, gives rise to the first truly expressive KP-ABE realization with constant-size ciphertext.

According to F. Zhang and K. Kim [5] proposed an ID-based ring signature approach, both approaches has defined base on bilinear pairings as well as Java pairing library. Also system analyzes their security and efficiency with different existing strategies. The Java Pairing library (JPBC) has used for data encryption and decryption purpose. Some user access control policies has design for end users that also enhance the privacy and anonymity of data ownr.

In appraoch [6], propose the first Identity-based threshold ring signature approach that does not support to java pairings. It propose the first Identity -based threshold verifiable ring signature strategy. System also analyze that the secrecy of the actual signers is maintained even against the PK generator (PKG) of the Identity -based system. Finallysystem shows how to add identity collusion and other existing base different schemes. Due to the dissimilar levels of signer inscrutability they support, the system proposed in this paper actually form a suite of Identity -based threshold ring signature method which is related to many real-world systems with varied anonymity needs.

In [7], system first validates the security requirements of whole architecture, and after that adds to in the security architecture. System proposed AES 128 16 bit encryption approach for end to end user verification and data encryption/ decryption purpose.

According to Kan Yan [10], System proposed Ciphertext-Policy Attribute-based Encryption (CP-ABE) is a promising technique for access control of encrypted data. It requires a trusted authority manages all the attributes and distributes keys in the system. In cloud storage systems, there are multiple authorities co-exist and each authority is able to issue attributes independently. However, existing CP-ABE schemes cannot be directly applied to data access control for multi-authority cloud storage systems, due to the inefficiency of decryption and revocation. In this paper, system propose DAC-MACS (Data Access Control for Multi-Authority Cloud Storage), an effective and secure data access control scheme with efficient decryption and revocation. Specifically, system construct a new multi-authority CP-ABE scheme with efficient decryption and also design an efficient attribute revocation method that can achieve both forward security and backward security

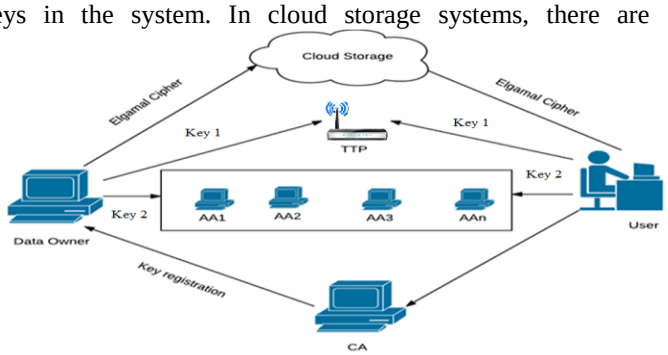

Fig.1: proposed system architecture

III. PROPOSED SYSTEM MODEL

The proposed system having 6 different objects, which work independently in different phases. Below we have shown all the entities

1) **Certificate Authority called as CA**
2) **Attribute Authorities called as AA's**
3) **Data Owner who upload the data**
4) **User who download the data**
5) **Trusted Middleware**
6) **Cloud Data Storage server**

A) Implementation Details and Procedure:

1. All Attribute Authorities has register with CA and get the unique Authority ID like {A1….An}
2. All accessible users has also registered with CA
3. All users when access the data from cloud servers it will communicate with multiple AA's and Trusted Middleware and gets the private keys for data decryption.
4. All data owner collect the private keys from CA when upload any plane text, for decryption purpose.
5. After that data owner upload the encrypted text on to cloud server and same time distribute both keys to AA's and trusted middleware.
6. With the help of both keys user can download cipher data from cloud server.

- The system can achievesecure data sharing approach and access control. The RBAC is also proposed for data access and revocation has used for prevention for unauthenticated usage,
- Cost, secure in the sense that it can accomplish both backward security and forward security.

The above figure 1 shows the overall system architecture with all different entities. The master entity CA executes all necessary requirements of system. CA gives all four keys for data privacy, when data owner generates any requirements. With the help of those keys data owners generate encryption and distributed the master key and private key to middle ware entities. The system also focuses on data sharing approach like Role Base Access Control (RBAC). Data owner can share the owner data to different users and those users can access the same data from middle ware parties and multiple authorities. The verification has done from any available AA. The system also enhance the secure user revocation approach for data owner and proxy key generation against old keys. The overall scenario can eliminate the certificate verification process and bottleneck issues.

B) Mathematical Module:

The proposed system having six different module set
S={ Dow, CA, TPA, CloudDB, AA's, User}

Data owner can upload the different text documents with respective key set
Dow ={Dm1<k1….kn>, Dm2…….Dmn<k1….kn>}

Generate a cipher data when upload
CData={Dm[i]<k1….kn>}

TPA holds the all master keys which distributed by data owner
TPA= {F1<mk>, F2<mk>……. Fn<mk>}

AA's also holds all private keys of each file
AA's= {F1<pk>, F2<pk>……. Fn<pk>}

Data owner can share the specific file to particular user
ShareGroup ={Uid<F,PK,MK>[1],…… Uid<F,PK,MK>[n]}

When user request for any file for cloud server, cloud server

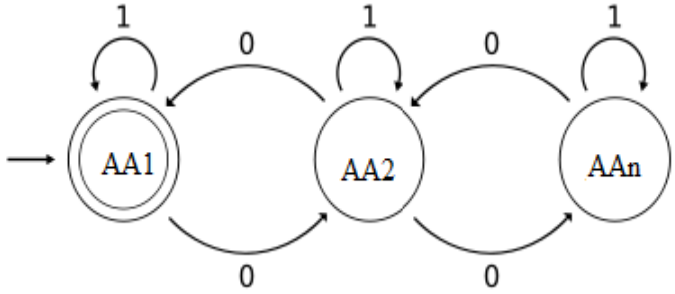

Fig.2 : State for each authority verification

Any t(n) return 1 then it will provide the private key otherwise state has change to another authority.

$M = (Q, \Sigma, \delta, q_0, F)$ where

$Q = \{S_1, S_2\}$,

$\Sigma = \{0, 1\}$,

$q_0 = S_1$,

$F = \{S_1\}$, and

The proposed state diagram shows the how users request processed by middle ware authority. When request has generated from end user it first receives by AA. If the AA1 is already acquired by some other process or busy then it will forward to another AA. This process works like recursively or base on round robin approach, when any t authority gets free it will return the keys to authenticated user.

4.3 Algorithms:

1. Elgamal Encryption scheme

Key Generation phase

Input : Plain text as text data d.

Output: a,b,p,g all which cointain public key, master key and 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=RandomA(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 BigInteger [] 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.

2. SQL Injection and prevention algorithm for Database Security

1: Procedure QueryVerify(PlainQuery, PatternList[ ])
INPUT: PlainQuery=User input Query, Threshold T
PatternList[]=user define Pattern List with m AnomalyPattern
2: For j = 0 to m do
3: If (Function (PlainQuery, String.Length(PlainQuery), PatternList[j][0]) = =0)then

4: Calcsimilarity score of input query
5: If (Score Value Anomaly = T)
6 : Block query or stop query execution
7 : else execute on database

IV. SOFTWARE REQUIRMENTS

1. System interfaces: Ubuntu Operating System
2. User interfaces: User interface using Jsp and Servlet
3. Hardware interfaces
Processor :- Intel R-Core i3 2.7 or above
Memory :- 4GB or above
Hard Disk :- 500 GB
4. Software interfaces:
Front End: Jdk 1.7.0, Eclipse
IE 7.0/above
Back-End: Mysql 5.1.
5. Communications interfaces
System will use HTTP as well as SMTP and SOAP protocol for establishing connection and transmitting data over the network.
6. Services: Amazon EC2 as Public cloud Environemnt

V. EXPERIMENTAL RESULT

For the system performance evaluation, calculate the matrices for accuracy. The system is executed 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 reasonable level. The below table 1 shows the proposed Elagamal algorithm performance for user plain text conversion as well encryption decryption.

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

Table 1: System performance (Estimated)

In second experimentation system show the user verification time with different approaches. In current system we consider as four different authorities for runtime verification. The below Fig. 3 shows the performance measures using different parameters with some existing approaches.

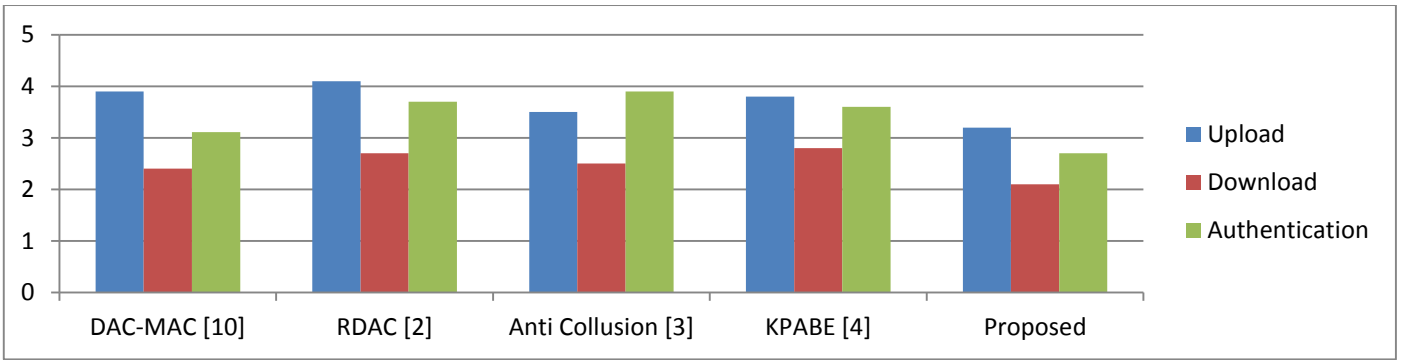

Fig.3: System Performance Measures proposed vs Existing approaches

VI. CONCLUSION

In this work system propose a secure Role Base Access Control (RBAC) data sharing scheme for untrusted environment in the cloud. In our scheme, the users can securely get their master and private keys from middleware authorities, CA provide and secure communication between multi parties. Also, our scheme is able to to provide the secure revocation for untrusted user. The proxy key generation has also proposed in this work. When data owner revokes any specific end user system automatically expired the existing keys and generates new keys for all shared users. The system can achieve highest level security as well as privacy through such approaches.

VII. FUTURE WORK

The current architecture is very efficient for security purpose, but sometime its utilized multiple resources. When the 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.

VIII. REFERENCES

[1]. Xue K, Xue Y, Hong J, Li W, Yue H, Wei DS, Hong P. RAAC: Robust and auditable access control with multiple attribute authorities for public cloud storage. IEEE Transactions on Information Forensics and Security. 2017 Apr;12(4):953-67.
[2]. Kan Yang and XiaohuaJia, "Expressive, Efficient, and Revocable Data Access Control for Multi-Authority CloudStorage", IEEE Transactions on parallel and distributed systems, VOL. 25, NO. 07, July 2014.
[3]. Zhongma Zhu and Rui Jiang proposed A Secure Anti-Collusion Data Sharing Scheme for Dynamic Groups in the Cloud in IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 27, NO. 1, JANUARY 2016.

[4]. N. Attarpadung, B. Libert, and E. Panafieu, "Expressive keypolicy attribute-based encryption with constant-size ciphertexts," in 2011.

[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. [Online]. Available: 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.

[10]. Yang K, Jia X. DAC-MACS: Effective data access control for multi-authority cloud storage systems. InSecurity for Cloud Storage Systems 2014 (pp. 59-83). Springer, New York, NY..