ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

A Procedural Based Encryption Technique for Accessing Data on Cloud

Avinash N[1], Divya C[2]

P.G. Student, Department of Computer Science and Engineering, SVIT, Bangalore, Karnataka, India[1]

Assistant Professor, Department of Computer Science and Engineering, SVIT, Bangalore, Karnataka , India[2]

ABSTRACT: Access controls are security aspects that control how systems and user interact and communicate with other resources and systems. Whenever an access control is being considered on cloud, there exists a mechanism to reduce the common overhead of the internet and to provide a fine-grained access control. An attribute based encryption can be used to address these issues.In a centralized attribute based encryption, there is a single key distribution center that distributes attributes and secret keys to all users. This single key distribution center is a single point of failure and it is also hard to maintain, because of huge number of users who are supported on cloud. Due to the mentioned facts, in this paper a decentralized attribute based encryption mechanism is proposed that would permit any party to act as an attribute authority by generating a user public key and issuing private keys to diverse users which reflect the user attributes without collaboration. The proposed mechanism also supports the features of anonymous authentication. Also, performance can be compared between decentralized attribute based encryption and decentralized cipher policy attribute based encryption.

KEYWORDS: Decentralized attribute based encryption, Decentralized cipher policy attribute based encryption

I. INTRODUCTION

In cloud computing much of the data stored is highly sensitive, thus privacy and security are very important issues in cloud computing. Before initiating any transaction the user should authenticate and must be ensured that outsourced data is not tampered by cloud. Cloud is itself, responsible for the services it gives and cloud also holds the users accountable for the data it outsources; User privacy protection is required so that the cloud or different clients do not know the user identity. Also validity of the users who stores the data to cloud must be verified.

On cloud effective search on encrypted information and data is an imperative. The cloud should not know which query is being executed, however it should be prepared to give back the records that satisfy the user query. This can be achieved by searchable encryption technique proposed by J. Li et. al. [1] and S. Kamara et. al. [2]. The user requested query is sent to the cloud and then cloud gives back the result for the requested query without knowing the actual query of the search. The issue here is that the information and data of records should have queries connected with them to empower the search. The correct records are returned by the cloud only when searched with the precise words.

Access control is gaining attention on cloud, because it is important that access to the valid service should be accessed only by authorized user. A huge amount of data and information is stored on cloud and much of this has more sensitive data information. Access controls are broadly divided into three types: Role Based Access Control (RBAC) [4], User Based Access Control (UBAC), and Attribute Based Access Control (ABAC) [3]. In Role Based Access Control, based on individual roles users are classified. In User Based Access Control, the access control contains the authorized users list to access data. In ABAC, attributes are given to users and access policy is attached to the data, the users who have valid set of attributes and satisfy the access policy can access the data.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

II. RELATED WORK

V. Goyal et. al. [5] presented new cryptosystem for fine grain sharing of encrypted information called Key Policy Attribute Based Encryption (KP-ABE). In this cryptosystem, ciphertexts are marked with set of private keys and attributes are connected with access structures that controls which ciphertext a client can decrypt. They appropriately showed the development to sharing of log data and broadcast encryption. Their development supports assignment of keys that are private which includes Hierarchical Identity Based Encryption (HIBE).Here, a creator whose attributes and keys have been repudiated can't write back stale data. From the attribute authority the beneficiary gets attributes and secret keys and has the capacity to decodedata if it has matching attributes. KP-ABE supports user revocation but this method uses centralized approach to distribute keys to the user.

J. Bethencourt et. al. [6] proposed a framework for acknowledging complex access control on encrypted encoded information called Ciphertext-Policy Attribute-Based Encryption. By utilizing this strategy encrypted information can be kept private even if the server is untrusted; additionally, this system is secure against conspiracy assaults. Here, while Key-Policy ABE and Ciphertext-Policy ABE catch two intriguing and complimentary sorts of system there exist different sorts of systems. The essential test in this line of work is to discover new frameworks with rich types of expression that create more than a self-assertive blend of systems. One constraint of this framework is that they demonstrated secure under the nonspecific gathering heuristic. Their attempt was to prove a framework secure under a more standard and non-intuitive suspicion. This kind of work would be intriguing regardless of the possibility that it brought about a moderate loss of effectiveness from existing system.

The proposed methodology by M. Chase et. al. [7] utilizes a trusted central authority and the utilization of a GID for every user, which implies the privacy that depends discriminatingly on the security of the central authority and the client privacy protection relies upon the legit conduct of the attribute-authorities. They proposed ABE scheme without the trusted authority and an anonymous key issuing convention which meets for both existing and for new schemes.

III. PROPOSED SYSTEM

The architecture of proposed system is depicted in Fig 3.1 where there are three users, a creator, a writer and reader. Creator who is assumed to be honest receives a token from the trustee. On presenting creator id like social insurance number/ like health card etc., the trustee generates a token for the creator. There can be multiple KDC's. For instance here there are two KDC's, which can be scattered. These KDC's can be like servers present in different locations of the world. A creator receives keys for encryption/decryption only on presenting the token that has been received from trustee to one or more KDC's.

In the figure as shown below, Secret Keys (SK) are given for decryption; the keys are for signing and under the access policy the message MSG is encrypted. The access policies decide who can access the stored data on the cloud. On a claim policy *Y* the creator decides to prove signature and authenticity of the message. The ciphertext *C* with signature is sent to the cloud. The cloud, by verifying this signature stores the ciphertext (C). If a reader wants to read data the cloud sends ciphertext *C*. The ciphertext *C* can be decrypted and can retrieve back the original message if the user has matching set of attributes with access policy. The verification process is assigned to the cloud and it relieves the individual clients from time consuming verifications. At the point when a reader needs to read some information which is stored in the cloud, it tries to decode and decrypt utilizing the secret keys which is received from KDC. If there are enough matching attributes with the access policy only then it decrypts the data which is stored on cloud.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

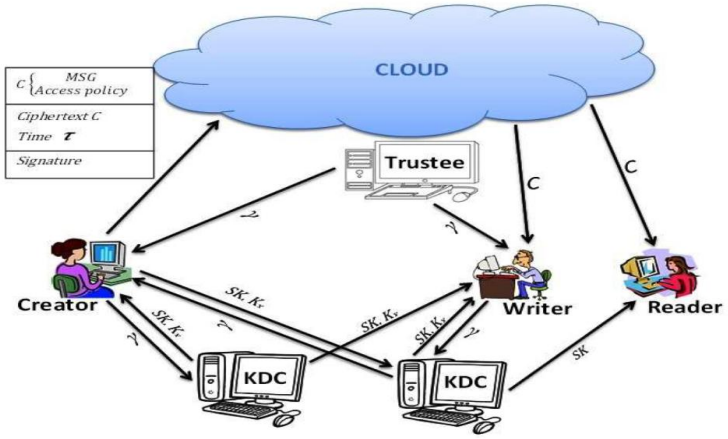

Fig 2.1: Secure cloud storage model

IV. DECENTRALIZED ATTRIBUTE BASED ENCRYPTION

The steps in decentralized attribute based encryption are as follows:

- System Initialization
- User Registration
- KDC setup
- Attribute generation
- Sign
- Verify

System Initialization: Select a prime q and groups G_1 and G_2, which are of order q. We define the mapping $\hat{e}: G_1 \times G_1 \rightarrow G_2$. Let $g1, g2$ be generators of G_1 and h_j be generators of G_2, for $j \in [t_{max}]$, for arbitrary t_{max}. Let H be a hash function. Let $A_0 = h_0^{a_0}$, where $a_0 \in \mathbb{Z}_q^*$ is chosen at randomly. $(TSig, TVer)$ of mean $TSig$ is the private key with which a signed message and for verification $TVer$ is the public key used. The trustee secret key is $TSK = (a_0\, TSig)$ and public key is $TPK = (G_1, G_2, H, g_1, A_0, h_0, h_1, \ldots, h_{t_{max}}, g_2, TVer)$.

User Registration: For a user identity U_u the KDC randomize $K_{base} \in G$. Let $K_0 = K_{base}^{1/a_0}$. The following token γ is output $\gamma = (u, K_{base}, K_0, \rho)$ where ρ is signature on $u||K_{base}$ the signing key $TSign$.

KDC Setup: Choose $a, b \in \mathbb{Z}_q^* q$ randomly and compute: $A_{ij} = h_j^a, B_{ij} = h_j^b$, for $A_i \in \mathbb{A}, j \in [t_{max}]$. The private key of i^{th} KDC is $ASK[i] = (a, b)$ and public key $APK[i] = (A_{ij}, B_{ij} | j \in [t_{max}])$.

Attribute Generation: In token verification algorithm, using the signature verification key $TVer$ in TPK verifies the signature contained in γ. This algorithm extracts K_{base} from γ using (a, b) from $ASK[i]$ and computes $K_x = K_{base}^{1/(a+bx)}, x \in J[i,u]$. The key K_x can be checked for consistency using algorithm $ABS.KeyCheck(TPK, APK[i], \gamma, K_x)$ which checks $\hat{e}(K_x A_{ij} B_{ij}^x) = \hat{e}(K_{base}, h_j)$ for all $x \in j[i,u]$ and $j \in [t_{max}]$.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Sign: The algorithm $ABS.Sign(TPK, \{APK[i]: i \in AT[u]\}, \gamma\{K_x: x \in J_u\}, MSG, y)$ has the secret key of the signer, the public key of the trustee, the policy claim y and the message to be signed. The claim policy is first converted into the span program $M \in \mathbb{Z}_q^{lxt}$, with rows labeled with attributes. Row x of M is denoted by M_x. Let π' denote the rows to the attributes mapping. So, $\pi'(x)$ is the mapping from M_x to attribute x. A vector v is computed that satisfies the assignment $\{x: x \in j[i,u]\}$. Compute $\mu = \mathcal{H}(MSG \| y)$. Choose $r_0 \in \mathbb{Z}_q^*$ and $r_i \in \mathbb{Z}_q, i \in J_u$ and compute $Y = K_{base}^{r_0}, S_i = \left(K_i^{v_i}\right)^{r_0} \cdot \left(g2g_1^{\mu}\right)^{r_i} (\forall i \in |J_u|)$, $W = K_0^{r_0}, P_j = \pi_{i \in AT[u]} \left(A_{ij} B_{ij}^{\pi'(i)}\right)^{M_{ij} r_i} (\forall j \in [t])$. The signature is calculated as $\sigma = (Y, W, S_1, S_2, \dots, S_t, P_1, P_2, \dots, P_t)$.

Verify: Algorithm $ABS.Verify(TPK, \sigma = (Y, W, S_1, S_2, \dots, S_t, P_1, P_2, \dots, P_t), MSG, y)$, converts y to the corresponding monotone program $M \in \mathbb{Z}_q^{lxt}$, with rows marked with attributes. Compute $\mu = \mathcal{H}(MSG \| y)$. $Y = 1, ABS.Verify = 0$ meaning false. Otherwise, the accompanying requirements are checked $\hat{e}(W, A_0) = \hat{e}(Y, h_0), \prod_{i \in I} \hat{e}\left(S_i, A_{i'j} B_{i'j}^{\pi'(i)}\right)^{M_{ij}} = \begin{cases} \hat{e}(Y, h_1)\hat{e}\left(g_2 g_1^{\mu}, P_j\right), j = 1 \\ \hat{e}\left(g_2 g_1^{\mu}, P_j\right), j > 1 \end{cases}$ where $i' = AT[i]$.

V. DECENTRALIZED CP-ABE

- System Initialization
- Authority Setup
- Authority KeyGen
- Encrypt
- Decrypt

System Initialization (κ). At the initial system setup phase, a trusted initializer chooses global public parameters GP according to the security parameter κ. Any authority or any user in the system can make use of these parameters GP in order to perform their executions.

Authority Setup (GP, Uj). This algorithm is run by every authority Aj ∈ A once during initialization. It accepts as input the global public parameters GP and a set of attributes Uj for the authority Aj and outputs public key PubAj and master secret key MkAj of the authority Aj.

Authority KeyGen (GP, ID, a, MkAj). Every authority executes this algorithm upon receiving a secret attribute key request from the user. It will take as input global public parameters GP, a global identity ID of a user, an attribute a hold by some authority and the master secret key of the corresponding authority. It returns a secret attribute key SKa, ID for the identity ID.

Encrypt (GP,M,A, {PubAj}). This algorithm is run by an encryptor and it takes as input the global public parameters GP, a message M to be encrypted, an access structure A, and public keys of relevant authorities corresponding to all attributes appeared in A. It then encrypts M under A and returns the ciphertext CT, where A is embedded into CT.

Decrypt (GP, CT, {SKa, ID|a ∈LID}). On receiving a ciphertext CT, a decryptor with identity ID runs this algorithm with the input the global public parameters GP, a ciphertext CT which is an encryption of M under A, and {SKa,ID|a ∈LID} is a set of secret attribute keys obtained for the same identity ID. Then it outputs the message M if the user attribute set LID satisfies the access structure A; otherwise, decryption fails.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

VI. EXPERIMENTAL RESULTS

Once decentralized attribute based encryption and decentralized cipher policy attribute based encryption mechanisms are implemented, performance between them can be compared by considering metrics like CPU and memory utilization for both of these. In order to obtain the CPU and memory utilization, C code is being written and with the help of system task manager, values are obtained

Fig 6.1 below shows the graph of the memory usage of the Decentralized ABE and Decentralized CP-ABE. where X-axis is the time in seconds and Y-axis memory size in megabytes.

Fig 6.2 below shows the graph of the CPU usage of the Decentralized ABE and Decentralized CP-ABE. where X-axis is the time in seconds and Y-axis CPU usage percentage.

Decntralized ABE	Decentralized CPABE
24.2	26.7
16.3	22.8
22.5	22.9
22.5	22.4
18.4	20.5
16.4	18.5
24.4	27.3
28.8	30.8
30.4	30.9
26.8	28.8
28.4	29.6
26.4	30.3

Table 6.1: Values of memory usage

Based on the values obtained as shown in the table above, memory usage graph is generated.

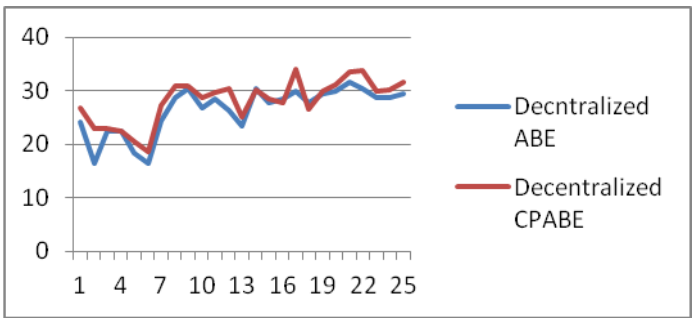

Fig 6.1: Graph of memory usage

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

Decntralized ABE	Decentralized CPABE
1	3
2	3
3	7
3	7
6	8
4	5
3	7
6	6
6	8
4	7
2	3

Table 6.2: Values of CPU usage

Based on the values obtained as shown in the table above, CPU usage graph is generated.

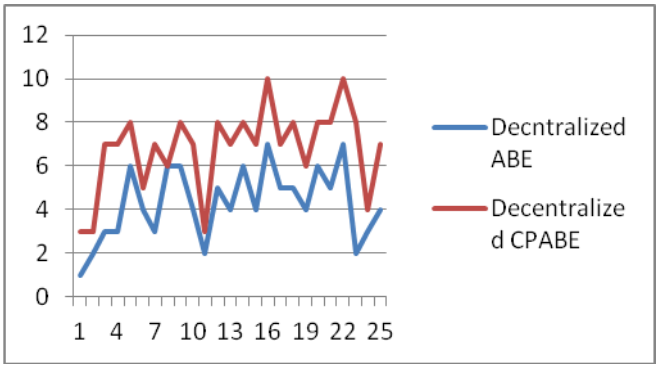

Fig 6.2: Graph of CPU usage

VII. CONCLUSION

In this paper Attribute Based Encryption (ABE) system is proposed which is decentralized in nature. Key distribution is done in a decentralized manner which distributes the access policy and attributes of a user. The proposed scheme prevents from replay attacks and it also supports user revocation. ABS technique is used to accomplish privacy and authenticity. In CP-ABE, the data is encrypted as ciphertext on cloud and user should use only public key to decrypt the data. It also does not support user revocation. In this paper, both the decentralized ABE and decentralized CP-ABE are implemented. Finally the performance between decentralized ABE and decentralized CPABE is compared based on the resources being consumed. As future work, cryptographic metrics like brute force attack, factoring or differential cryptanalysis may be considered.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

REFERENCES

[1] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren and W. Lou, "Fuzzy Keyword Search Over Encrypted Data in Cloud Computing," Proc. IEEE INFOCOM, pp. 441-445, 2010.

[2] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc. 14th International Conference Financial Cryptography and Data Security, pp. 136-149, 2010.

[3] S. Ruj, M. Stojmenovic and A. Nayak, "Privacy Preserving Access Control with Authentication for Securing Data in Clouds," Proc. IEEE/ACM International Symp. Cluster, Cloud and Grid Computing, pp. 556-563, 2012.

[4] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward Secure and Dependable Storage Services in Cloud Computing," IEEE Transaction Services Computing, vol. 5, no. 2, pp. 220-232, April- June 2012.

[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conference Computer and Communication Security, pp. 89-98, 2006.

[6] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption," Proc. IEEE Symp. Security and Privacy, pp. 321-334, 2007.

[7] M. Chase and S.S.M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," Proc. ACM Conference Computer and Communication Security, pp. 121-130, 2009.

[8] H. Li, Y. Dai, L. Tian, and H. Yang, "Identity-Based Authentication for Cloud Computing," Proc. First International Conference Cloud Computing (CloudCom), pp. 157-166, 2009.

[9] C. Gentry, "A Fully Homomorphic Encryption Scheme," PhD dissertation, Stanford University, http://www.crypto.stanford.edu/craig, 2009.

[10] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing," Proc. Fifth ACM Symp. Information, Computer and Communication Security (ASIACCS), pp. 282-292, 2010.

[11] D.F. Ferraiolo and D.R. Kuhn, "Role-Based Access Controls," Proc. 15th National Computer Security Conference, 1992.

[12] D.R. Kuhn, E.J. Coyne, and T.R. Weil, "Adding Attributes to Role-Based Access Control," IEEE Computer, vol. 43, no. 6, pp. 79-81,June 2010.