Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

Attribute Based Privacy Preserving Access Control with Authentication for Secure Data in Cloud

[1]Prof. Mahip Bartere, [2]Miss Bharati Dalvi

[1]Assistant Professor, [2]student M.E., Dept. of Computer Sci. & Engg., G. H.Raisoni College of Engg. &Management, Amravati, India

Abstract: **In this paper we propose a new decentralized access control scheme for secure data storage in clouds which supports anonymous authentication. The cloud verifies the authenticity of the series without significant knowledge in the user's identity before storing information. This scheme also has the added feature of access control. In access control scheme only valid users are able to decrypt the stored data/information. This scheme prevents replay attacks also supports creation, modification, and reading information stored in the cloud. These schemes also address user revocation. Moreover, the authentication and access control scheme is decentralized and robust in nature unlike other access control schemes designed for clouds which are centralized. The computation, communication, and storage overheads are comparable to centralized approaches.**

Keywords: **Access control, anonymous authentication, decryption/ encryption, data storage, cloud.**

I. INTRODUCTION

In cloud computing, users can catch out their computation and storage to servers (also called clouds) using Internet. This frees users form the stability of maintaining resources on-site. Several types of services like applications (e.g., Google Apps, Microsoft online), infrastructures (e.g., Amazon's EC2, Eucalyptus, Nimbus), and platforms (e.g., Amazon's S3, Windows Azure) can be provided by cloud to help developers to maintain secure connections. Information which is stored in clouds is very sensitive. For example, medical records and social network, we can consider these as a sensitive data.

In cloud computing, security and privacy are considered as a big issue. At first step the user should authenticate itself before initiating any transaction, and on the second step, it must be ensured that the cloud does not alter with the data that is outsourced.

User privacy is also required in cloud. By using privacy the cloud or other users do not know the identity of the other user. The cloud can hold the user accounts for the data in cloud, and likewise, to provide services the cloud itself is accountable. The validity of the user who stores the data is also verified. There is also a need for law enforcement apart from the technical solutions to ensure security and privacy.

The cloud is also prone to data modification and server colluding attacks. The adversary can compromise storage servers in server colluding attack, so that server can modify data files even though the servers are internally consistent. The data needs to be encrypted to provide secure data storage. However, the data is often modified and this dynamic property needs to be taken into account while designing efficient secure storage techniques [1].

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

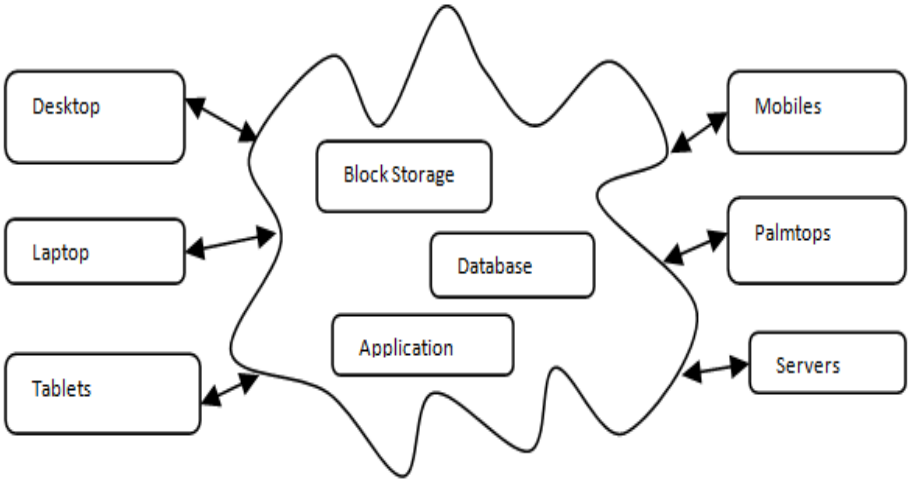

Fig1: Example diagram for data sharing with cloud storage.

Efficient search on encrypted data is also an important fear in clouds. The clouds should not know the query but it can be able to return the records that satisfy the query. Searchable encryption used to achieve this scheme.Users authentication scheme using public key cryptographic technique is used in cloud computing. Many homomorphism encryption techniques have been optional to ensure that the cloud is not able to read the data while performing computations on the data. By using this encryption scheme, the cloud receives cipher text of the data and performs computations on the cipher text and returns the encoded value of the result to user then the user is able to decode the result, even though the cloud does not know what data it has operated on. In such circumstances, it must be probable for the user to verify that the cloud returns correct results. Neither the clouds nor users should deny any operations performed or requested. It is important to have log of the transactions performed.

II. RELATED WORK &LITERATURE SURVEY

In 2006, A. Sahai and B. Waters worked on "Fuzzy Identity-Based Encryption" In Identity Based Encryption scheme, user has a set of attributes in addition to its unique ID. A Fuzzy IBE scheme can be applied to enable encryption .In Fuzzy scheme biometric input used as identity. This proposed method was error-tolerant and also secure the data against collusion attacks [2].

In 2006, V. Goyal, O. Pandey, A. Sahai, and B. Waters, proposed another technique for encryption of data i.e.; "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data". In this scheme, the sender has an authorization to encrypt information. A revoked attributes and keys of users cannot write again to steal information. The attribute authority receives attributes and secret keys from the receiver and the user will be able to decrypt information if he/she has matching attributes [3].

In 2007, J. Bethencourt, A. Sahai, and B. Waters worked on "Cipher text-Policy Attribute-Based Encryption". By using this approach the receiver has the access policy in the form of a tree. The tree contain attributes as leaves and monotonic access structure with AND, OR and other threshold gates The advantage of using this technique is that the encrypted information can be kept confidential even if the storage server is not trusted; and it is also secured against collusion attacks [4].

In 2007, M. Chase, worked on "Multi- Authority Attribute Based Encryption". This scheme describes several Key Distribution Authorities (coordinated by a trusted authority) which distribute attributes and secret keys to users. Multi-authority Attribute Based Encryption protocol does not require a trusted authority, which means every user must have attributes from at all the KDCs. The benefit of using this technique is that it allows more number of attributes [5]. In 2008, H.K. Maji, M. Prabhakaran, and M. Rosulek worked on "Attribute-Based Signatures: Achieving Attribute-Privacy and Collusion-Resistance". In this paper, ABCs were introduced to ensure anonymous user authentication. This was also a centralized approach. The basic advantage of using this technique is that the user significantly saves decryption time, without raising the number of transmissions [6].

In 2010, M. Li, S. Yu, K. Ren, and W. Lou worked on "Securing health records in cloud computing: Patient-centric and fine-grained data access control in multi owner settings". A colossal measure of data is constantly archived in the cloud, and much of this is sensitive data. Utilizing Attribute Based Encryption (ABE), the records are encrypted under a few access strategy furthermore saved in the cloud. Clients are given sets of traits and corresponding keys. Just when the clients have matching set of attributes, would they be able to decrypt the data saved in the cloud [7].

In 2011, A.B. Lewko and B. Waters worked on "Decentralizing Attribute-Based Encryption," In this proposed work users could have zero or more number of attributes from each authority and did not require a trusted server. This proposed technique is collision resistant [8].

In 2011, M. Green, S. Hohenberger and B. Waters proposed another work based on "Outsourcing the Decryption of ABE Ciphertexts," .This proposed scheme subcontract the decryption task to a proxy Server, so that the user made computation on minimum resources like hand held devices. The advantage of using this is that the user significantly saves bandwidth, without raising the number of transmission [9].

In 2011, S. Jahid, P. Mittal, and N. Borisov proposed EASIER ie; "EASiER: Encryption-based access control in social networks with efficient revocation". Access control is likewise gaining imperativeness in online social networking where users store their personal data, pictures, films and shares them with selected group of users they belong. Access control in online social networking has been studied in [10].

In 2011, the work done by F. Zhao, T. Nishide, and K. Sakurai in ""Realizing fine-grained and flexible access control to outsourced data with attribute-based cryptosystems," gives privacy preserving authenticated access control in cloud. Nonetheless, the researchers take a centralized methodology where a single key distribution center (KDC) disperses secret keys and attributes to all clients. Unfortunately, a single KDC is not just a single point of failure however troublesome to uphold due to the vast number of clients that are upheld in a nature's domain. This scheme] uses a symmetric key approach and does not support authentication [11].

In 2011, S. Ruj, A. Nayak, and I. Stojmenovic proposed "DACC: Distributed access control module in clouds". On the other hand, the approach did not provide client verification. The other weakness was that a client can make and store a record and different clients can just read the record. write access was not allowed to clients other than the originator [12].

In 2012, Kan Yang, Xiaohua Jia and Kui Ren proposed a decentralized approach in "DAC-MACS: Effective Data Access Control for Multi- Authority Cloud Storage Systems", their strategy does not confirm clients, who need to remain anonymous while accessing the cloud [13].

III. PROPOSED SYSTEME

The main contributions of this paper are the following:

1. Distributed access control of data stored in cloud so that only authorized users with valid attributes can access them.
2. The identity of the user is protected from the cloud during authentication.
3. The architecture is decentralized, meaning that there can be several KDCs for key management.
4. The access control and authentication are both collusion resistant, meaning that no two users can collude and access data or authenticate themselves, if they are individually not authorized.
5. Revoked users cannot access data after they have been revoked.
6. The proposed scheme is resilient to replay attacks. A writer whose attributes and keys have been revoked cannot write back stale information.
7. The protocol supports multiple read and writes on the data stored in the cloud.
8. The costs are comparable to the existing centralized approaches, and the expensive operations are mostly done by the cloud.
9. The identity of the user is protected from the cloud during authentication.

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

10. Authentication of users who store and modify their data on the cloud.

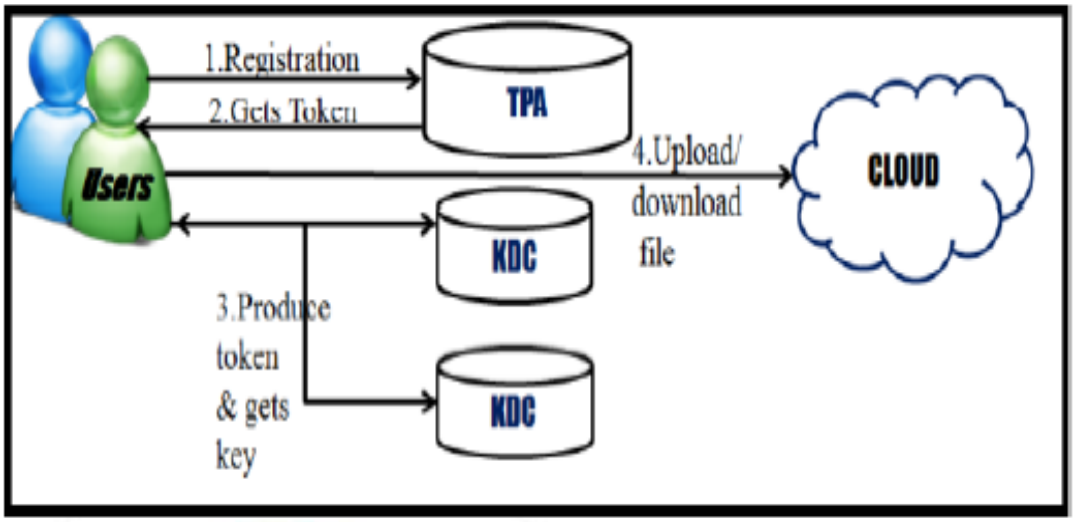

Fig.2 Decentralized Architecture

This work proposes a decentralized approach, their technique does not authenticate users, who want to remain anonymous while accessing the cloud. In an earlier work, Ruj *et al.* proposed a distributed access control mechanism in clouds. However, the scheme did not provide user authentication. The other drawback was that a user can create and store a file and other users can only read the file. Write access was not permitted to users other than the creator. In the preliminary version of this paper, we extend our previous work with added features which enables to authenticate the validity of the message without revealing the identity of the user who has stored information in the cloud. In this version we also address user revocation. We use attribute based signature scheme to achieve authenticity and privacy.

1. Overall system:

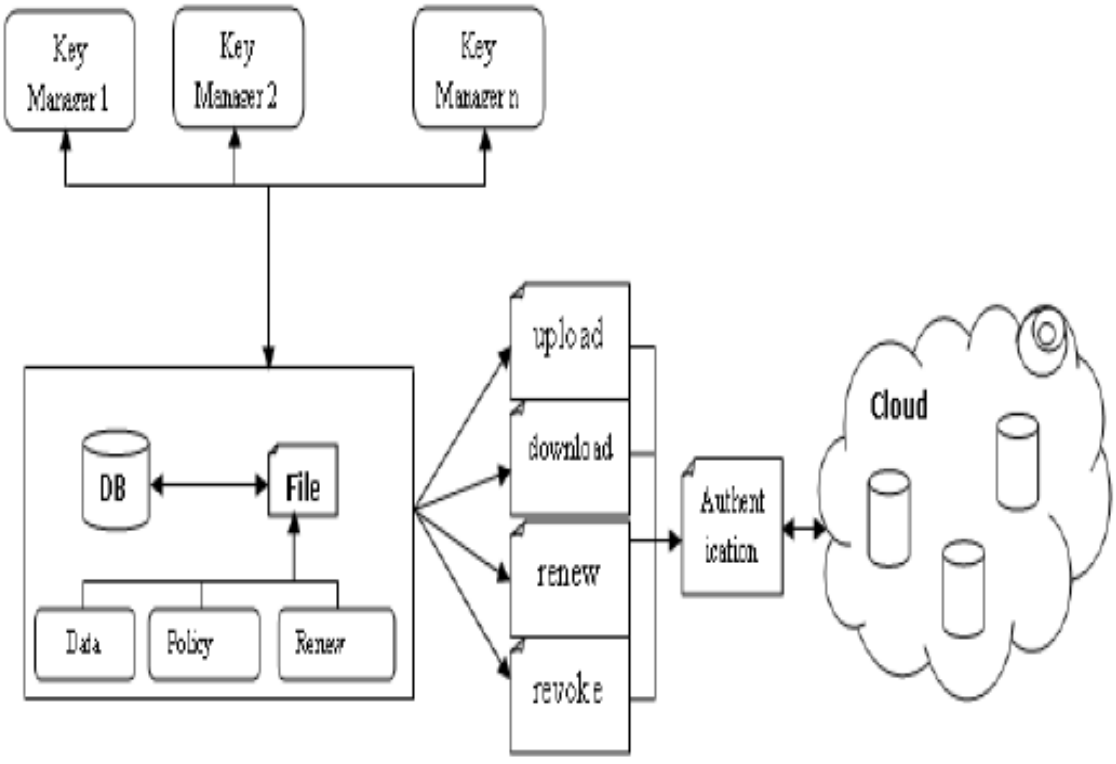

Fig.3 Overall system Diagram

We propose policy based file access and policy based file assured deletion for better access to the files and delete the files which are decided no more. We propose effective renewal policy for making better approach to renew the policy without downloading the data key and control keys, which is available now a day.

2. Flow of System:

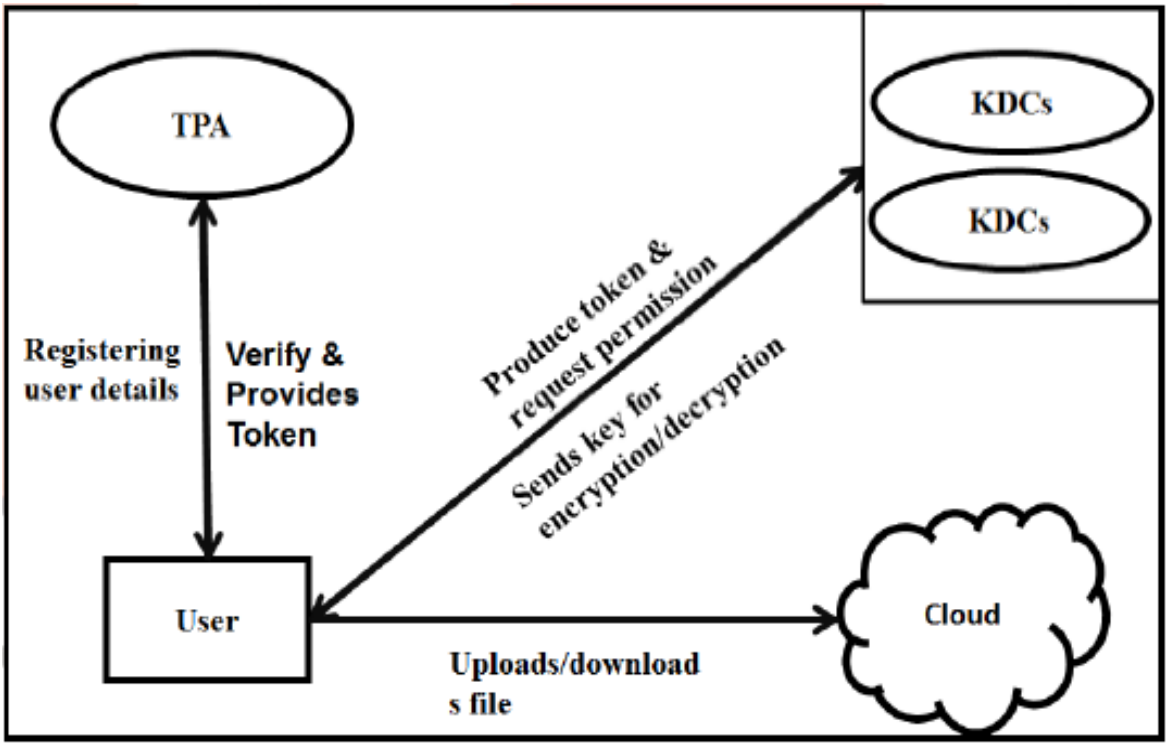

Fig 4. Flow of System

First the client was authenticated with the username and password, which is provided by the user. Then the user was asked to answer two security levels with his/her choice. Each security levels consist of 5 user selectable questions. The user may choose any one question from two security levels. The private key for encrypt the file was generated with the combination of username, password and the answers for the security level questions. After generating the private key the client will request to the key manager for the public key. The key manager will verify the policy

Associated with the file. If the policy matches with the file name then same public key will be generated. Otherwise new public key will be generated. With the public key and private key the file will be encrypted and uploaded into the cloud. If a user wants to download the file he/she would be authenticated. If the authentication succeeded, the file will be downloaded to the user. Still the user cant able to read the file contents. He / she should request the public key to the key manager. According to the authentication, the key manager will produce the public key to the user. Then the user may decrypt the file using the login credentials given by the user and the public key provided by the key manager.

The client can revoke the policy and renew the policy due to the necessity.

In this paper, following are the cryptographic keys to protect data files stored on the cloud

Public Key: The Public key is a random generated binary key, generated and maintained by the Key manager itself.

Particularly used for encryption/ decryption.

Private Key: It is the combination of the username, password and two security question of user's choice. The Private Key is maintained by client itself. Used for encrypt / Decrypt the file.

Access key: It is associated with a policy. Private access key is maintained by the client. The access key is built on Attribute based encryption. File access is of read or write.

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

IV. PROPOSED WORK

A. USER REGISTRATION:

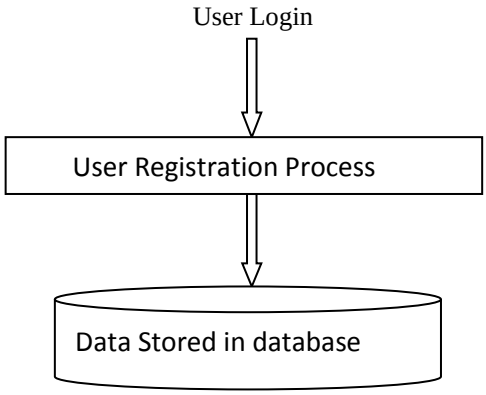

Fig.5 User Login Process

Users have an initial level Registration Process at the web end. The users provide their own personal information for this process. The server in turn stores the information in its database.

B. TRUSTEE AND KEY DISTRIBUTION CENTRE:

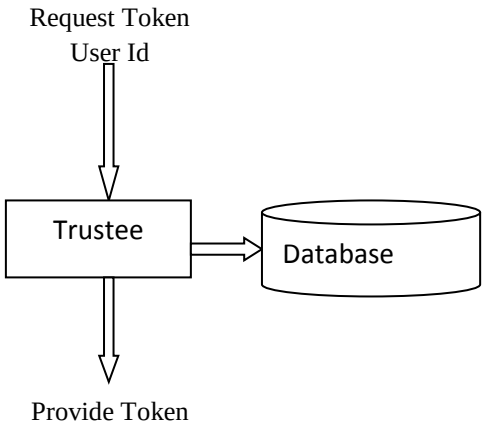

Fig. 6 User Accessibility

Users receive a token from the trustee, who is assumed to be honest. A trustee can be someone like the federal government who manages social insurance numbers etc. On presenting her id (like health/social insurance number), the trustee gives her a token. There are multiple KDCs (here 1), which can be scattered. Users on presenting the token to KDC receive keys for encryption/decryption and signing. SK are secret keys given for decryption, Kx are keys for signing

C. ENCRYPTION/DECRYPTION:

We used AES algorithm for encryption/Decryption. This

Algorithm is the proven mechanism for secure transaction.

Algorithm:

a) Key Expansion: Using the key schedule of Rijndael, round keys are derived from the cipher key

b) Initial Round - AddRoundKey: Then using bitwise X OR each byte of the state is combined with the round key.

c) Rounds

i) SubBytes*:* This is a non-linear substitution step where each byte is swapped with another according to a lookup table.

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

ii) ShiftRows*:* In this transposition step each row of the state is shifted cyclically a certain number of steps.

iii) MixColumns*:* A mixing operation which operates on the columns of the state, combining the four bytes in each column.

iv) AddRoundKey

d) Final Round (no Mix Columns)

v) SubBytes

vi) ShiftRows

vii) AddRoundKey

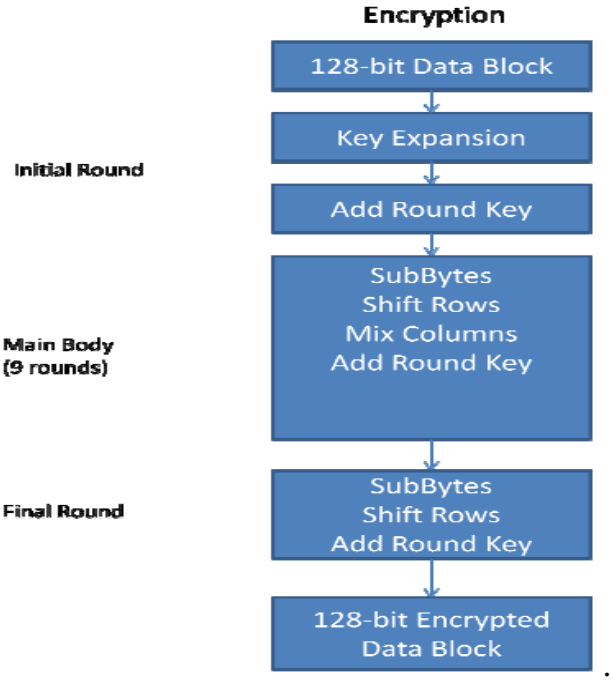

Fig.7 Process of AES

D. FILE UPLOAD / DOWNLOAD:

1. File Upload:

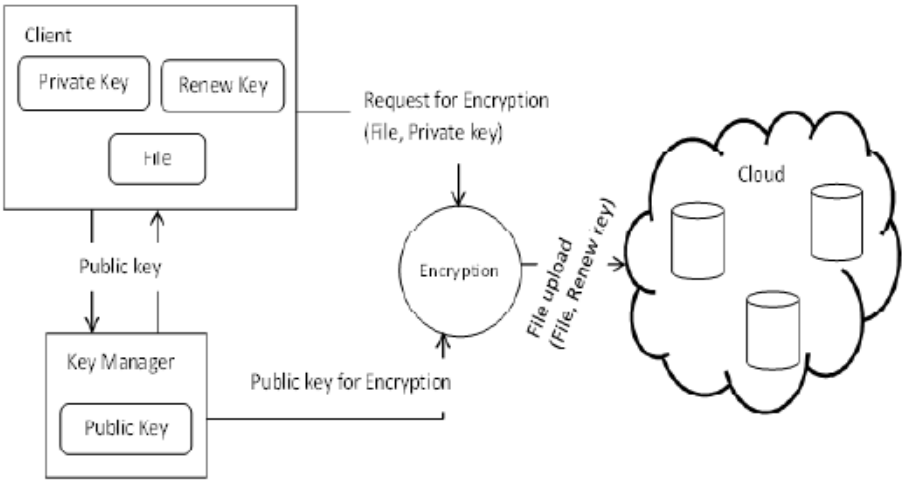

Fig.8 File uploading process.

The client made request to the key manager for the public key, which will be generated according to the policy associated with the file. Different policies for files, public key also differs. But for same public key for same policy will be generated. Then the client generates a private key by combining the username, password and security credentials. Then the file is encrypted with the public key and private key and forwarded to the cloud.

2. File Download:

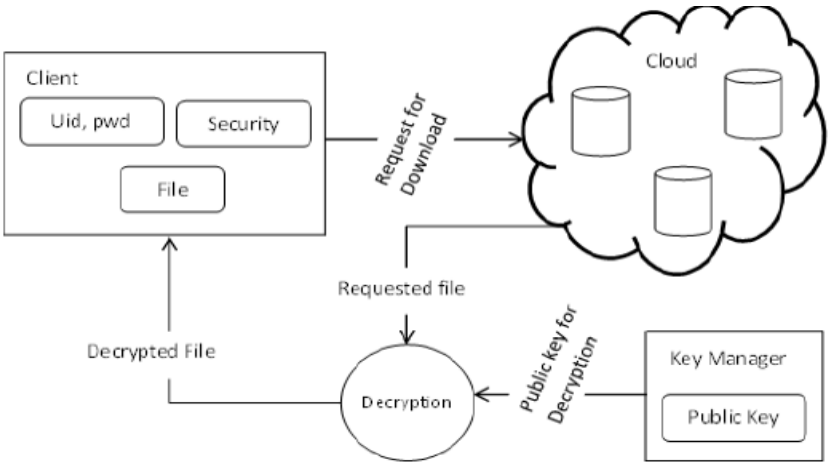

Fig.9 File downloading process

The client can download the file after completion of the authentication process. As the public key maintained by the key manager, the client request the key manager for public key. The authenticated client can get the public key. Then the client can decrypt the file with the public key and the private key. The users credentials were stored in the client itself. During download the file the cloud will authenticate the user whether the user is valid to download the file. But the cloud doesn't have any attributes or the details of the user.

E. POLICY REVOCATION FOR FILE ASSURED DELETION:

The policy of a file may be revoked under the request by the client, when expiring the time period of the contract or completely move the files from one cloud to another cloud environment. When any of the above criteria exists the policy will be revoked and the key manager will completely removes the public key of the associated file. So no one recover the control key of a revoked file in future. For this reason we can say the file is assuredly deleted.

Automatic file revocation scheme is also introduced to revoke the file from the cloud when the file reaches the expiry and the client didn't renew the files duration.

F. FILE ACCESS CONTROL:

Ability to limit and control the access to host systems and applications via communication links. To achieve, access must be identified or authenticated. After achieved the authentication process the users must associate with correct policies with the files.

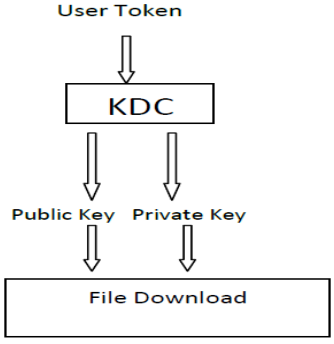

Fig. 10 File Accessing

Novelty Journals

ISSN 2394-7314

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

To recover the file, the client must request the key manager to generate the public key. For that the client must be authenticated. The attribute based encryption standard is used for file access which is authenticated via an attribute associated with the file. With file access control the file downloaded from the cloud will be in the format of read only or write supported. Each user has associated with policies for each file. So the right user will access the right file. For making file access the attribute based encryption scheme is utilized.

G. KEY GENERATION:

While working on MySQL databases and PHP for web applications we came across MD5.The MD5 message-digest algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA.

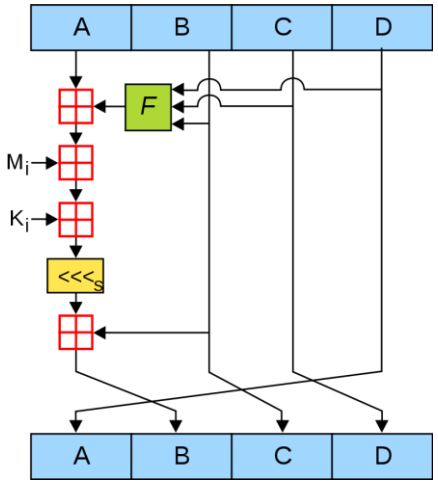

Fig.11 Function of MD5

V. COMPUTATION AND COMPLEXITY

We have done experimental analysis on 4 GB RAM and we have used AES encryption system to encrypt the data before uploading the cloud. As we have used anonymous authentication system it will take time from normal single step authentication system but definitely will achieve greater security. The base paper specifies the time of 2.9 ms to encrypt the data of 512 bits and it goes exponentially so we can consider approx 8 ms for approx 1536 bits file. Our AES system shows the time of 2.6 ms for 512 bits of data and 6.1 ms for 1536 bits of data. It can be visible from graph

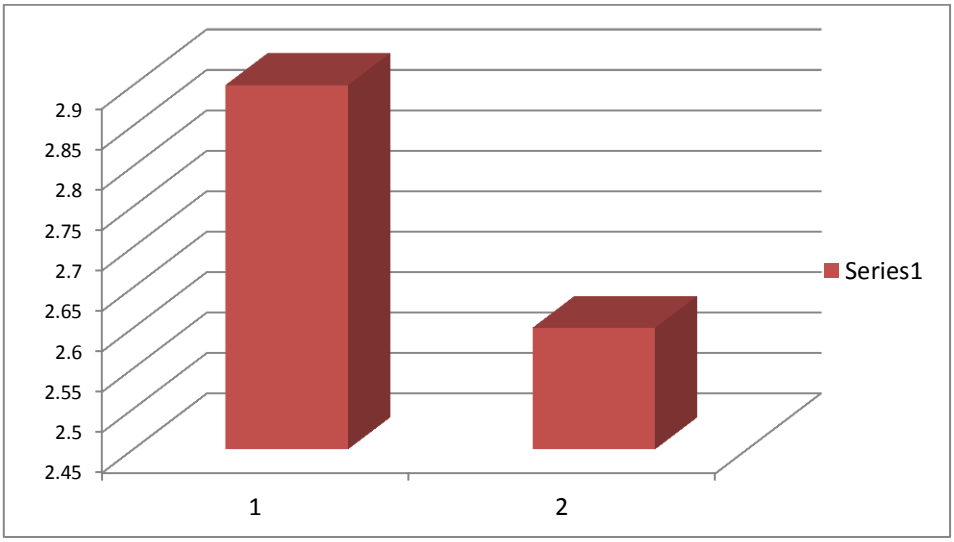

Fig.12 Graph 1

International Journal of Novel Research in Computer Science and Software Engineering
Vol. 3, Issue 1, pp: (260-270), Month: January-April 2016, Available at: www.noveltyjournals.com

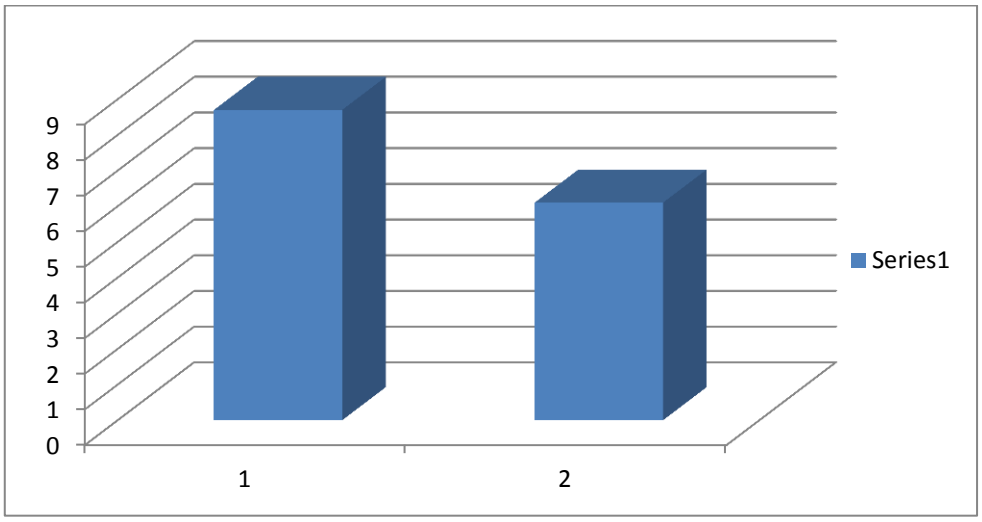

Fig.13 Graph 2

File Size in Bits: 512 & 1536

Execution Time: 2.9 & 8.7

AES Execution Time: 2.6 & 6.1

VI. CONCLUSION

This review paper presented a decentralized access control technique with anonymous authentication. This decentralized scheme provides user revocation and prevents replay attacks. Even though the cloud does not know the identity of the user who stores information, but it verifies the user's credentials. This paper made key distribution is done in a decentralized way.

REFERENCES

[1] R.Ranjith, D.Kayathri Devi, "Secure Cloud Storage using Decentralized Access Control with Anonymous Authentication", International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 11, November 2013.

[2] A. Sahai and B. Waters, "Fuzzy Identity- Based Encryption," Proc. Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 457-473, 2005.

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

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

[5] M. Chase, "Multi-Authority Attribute Based Encryption," Proc. Fourth Conf. Theory of Cryptography (TCC), pp. 515- 534, 2007.

[6] H.K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-Based Signatures: Achieving Attribute-Privacy and Collusion- Resistance," IACR Cryptology ePrint Archive, 2008.

[7] M. Li, S. Yu, K. Ren, and W. Lou, "Securing health records in cloud computing: Patient-centric and fine-grained data access control in multi owner settings," in SecureComm, pp. 89–106, 2010.

[8] A.B. Lewko and B. Waters, "Decentralizing Attribute-Based Encryption," Proc. Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT), pp. 568-588, 2011.

[9] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the Decryption of ABECiphertexts," Proc. USENIX Security Symp., 2011.

[10] S. Jahid, P. Mittal, and N. Borisov, "EASiER: Encryption-based access control in social networks with efficient revocation," in ACM ASIACCS, 2011.

[11] F. Zhao, T. Nishide, and K. Sakurai, "Realizing fine-grained and flexible access control to outsourced data with attribute-based cryptosystems," in ISPEC, ser. Lecture Notes in Computer Science, vol. 6672. Springer, pp. 83–97, 2011.

[12] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Distributed access control in clouds," in IEEE TrustCom, 2011.

[13] Kan Yang, Xiaohua Jia and Kui Ren, " DAC-MACS: Effective Data Access Control for Multi- Authority Cloud Storage Systems", IACR Cryptology ePrint Archive, 419, 2012.

[14] Sushmita Ruj, Member, IEEE, Milos Stojmenovic, Member, IEEE, and Amiya Nayak, Senior Member, IEEE "Decentralized Access Control with Anonymous Authentication of Data Stored in Clouds" IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 25, NO. 2, FEBRUARY 2014

[15] Geetanjali. M1 , Saravanan. N2 International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) "Attribute Based Encryption with Privacy Protection in Clouds " Vol.2, Special Issue 1, March 2014

[16] B. Sri Varsha1, P.S. Suryateja2," Using Attribute-Based Encryption with Advanced Encryption Standard for Secure and Scalable Sharing of Personal Health Records in Cloud" B. Sri Varsha et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (5) , 2014

[17] Piyush Gupta, Sandeep Kumar "A Comparative Analysis of SHA and MD5 Algorithm Piyush Gupta, Sandeep Kumar" Piyush Gupta et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014