www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 5 May 2015, Page No. 11882-11888

Re-Encryption with Attribute Key Management for Secure Transactions in Cloud

Yugesh Mothukuru and K. John Singh

School of Information Technology and Engineering
VIT University, Vellore, Tamil Nadu, India
yugesh6@gmail.com

Abstract: ***Transferring of information to the cloud is valuable due to the access and adaptability of critical technical difficulties that are remain. Ensure sensitive information stored in the cloud from being detail and liberated by a cloud supplier that is fair yet curiosity. Novel adjustments to attribute based encryption are permitted to the approved clients to access the cloud information focused on the fulfillment of providing the services such that the higher computational burden from cryptographic operations is allocated to the cloud supplier and the aggregate expenses is brought down for the cloud clients. The cloud provider is to diminish the cost of client repudiation in a portable client environment while saving the security of client information stored in the cloud may alternatively perform information re-encryption. The protocol, which is realize on commercially popular cloud platforms to simulate real- world standards that show the efficiency of the scheme.***

Keywords: Re-encryption, Cloud storage, Cryptography, key management, Energy efficiency**.**

1. Introduction

Cloud computing and Cloud storages are gaining high popularity which offers scalable and reliable security and storage to third parties. Cloud computing uses pay-per-use or on-demand criteria for the economical and corporate data that leads the cost of deploying the resources in-house. Transferring the data to the cloud is appropriate for many applications and available to all the users. Clients have to pay for the cloud provider to data storage and computational to the network communication, which the clients used for further. The cloud provider can enhance the automatic back-ups for the users to make applications safety and high accessibility of the user data. The major concerns had happen from earlier is that users data can be accessed and read by the cloud administrator without the knowledge of the client. The cloud administrator may not be trusted as per the security obligations. If it happen additional risks should done on the sensitive data which carry the persistent risks were intercept by the unauthorized party who despite the firewalls promised by the provider. For such premises, there are different software techniques used such as encryption key management to get the confidentiality of cloud data, which is preserve at all the times. It is very difficult to protect the sensitive user data like e-mails, personal information, financial and medical records.

It must also be identified the recent trends following by the users. The cloud computing applications is for cloud data can be accessed by the resource-constrained mobile devices called as “mobile cloud computing”. Some of the WSN can hold the protocols and make the security with additional features, which decreases the cost for usage, and increase the battery life from drain and the delays of the execution of the task.

The main objectives of the proposed work as detailed below:

1. The data storage to the cloud provider with ensuring a protocol in a secure fashion. The provider can unable to read the stored data without the permissions of the administrator. The protocol can be designed as the efficient resource constrained cloud users for delegating computation to the cloud provider or the trusted third party where the appropriate without compromising the security issues.
2. The traditional attribute encryption is enhanced to key generate between the data owner and the trusted authority.
3. Group key mechanism is provide as an additional security to the data owner .the data owner controls the access the additional secret key with the required attributes. This additional secret key, which analyze optional variant of high sensitive data, which frequently accessed.
4. Re-encryption is a process of transforming the cipher text only permits to the authorized users. Here without the interruption of data owner the administrator can generate the key.

In the section 2,related works is presented ,which is focused on the attribute based policy .in the section 3, model of web based and mobile cloud computing is presented . In the section 3.1 system model and in 3.2 trust model is represent, where the entire system architecture is design. In section 4, the existing system algorithm description and the evaluation process has explained. In section 5, the implementation analysis has presented which shows the security taken by the cloud provider and the administrator among the different users. In section 6, the performance analysis is represent, where the calculations among different workloads, with key and without key usages. In section 7, the Results works can be explains among different clients based on the graphs.

2. Related works

Many solutions may be visualize to exchange the encrypted data with the cloud provider is not entrusted with key material which proven difficult to scale. The drawback of the public key management system such as RSA, which requires the data owner to provide encrypted data for ever user, which may access. If the client data were, encrypt with single key which key must be share with all authorized user. Clients may join

and leave the approved client set frequently, prompting consistent key re-encryption and redistribution through extra communication sessions to handle user revocation. In wireless communication, transmitting is cost oriented and which results quick battery drain. Data should store in the cloud in encrypted format, by that the cloud provider cannot access it. The trouble emerges when the new clients joined into the framework, and existing clients or users may leave, requiring that new keys to be create. The encrypted data should transformed with the unlocked new keys, [3] without an intermediate decryption process that would permit to the cloud provider to access the plaintext. This is termed as "data re-encryption".

The technique ciphertext-policy attribute-based encryption (CP-ABE) offers various benefits that permits a client to acquire read to encrypted information in the cloud in view of the ownership of specific attributes that fulfill an entrance structure characterized in the cloud, instead of the ownership of a key that must be share to all invested individuals in advance. Normally, a scheme based on CP-ABE depends upon [4] the data owner giving access authorization through an access tree, which requires his or her consistent availability. Further key material is distributed among multiple parties for information, a data owner and a trusted authorize may work in show to give access authorization to different clients.

Revocation of an authorized user is especially difficult to fulfill effectively in CP-ABE and is addresses by expanding attributes or keys with expiration dates. A tree of revocable attributes may need to be keep and a trusted party assign to validate revocation status. A mechanism using linear secret sharing and binary tree techniques is one example, but users [7] have to incur the communication cost of continually requesting new keys. In addition, the data owner is normally also a mobile user or web user, and thus cannot manage access control on demand for all due to its transient network.

An alternate related methodology combines Hierarchical Identity Based Encryption (HIBE) [12] and CP-ABE utilizing various leveled domain experts to appropriate client keys. The expense of expanded storage requirements for key material held by clients and a high amount measure of handling when producing ciphertext. A strategy for trusted information offering has recommended that use a dynamic elliptic curve encryption scheme. Then again, it depends upon a writer transferring encrypted data to the cloud, then distributing certifications to the cloud to perform re-encryption, furthermore to the reader on every data access attempt; this is clearly impractical when connected to resource-constrained devices and networks. To handle repudiation in a high adaptable system. Different other proxy re-encryption schemes have applied to secure distributed storage. One method is to re-encrypt the stored content during recovery. Such a technique has applied to an encrypted file storage system where a content owner encrypts blocks of content with unique, symmetric content keys, and these keys are further encode to structure a lockbox; users communicate with an access control server to decrypt them. The issue is that the content owner manages access control for all other users, which is a great burden and [9] requires dynamic re-encryption of the same data whenever multiple users access it. In the model herein, one-time re-encryption only occurs whenever membership enrollment changes, probably a less continuous event than that of information access. Other approaches require a trusted proxy for each decryption, which increases the communication cost.

Proxy re-encryption has also combined with CP-ABE such that the cloud provider based on a secret that is per-shared between the data owner and the provider, as well as the provider's internal clock figures re-encryption keys. The re-encryption keys must figure for all attributes in the access structure, which could be extremely various. An alternate thought is too safely and install the data key inside the header of the record stored in the cloud. An authorized manager group is responsible for [10] generation of re- encryption keys, but it must also distribute the secret header key to the recipient to complete the process. A procedure that combines CP-ABE with proxy re-encryption does not appear to be highly efficient for users, For instance, the decryption process requires processing two access subtrees instead of one.

An alternate related work proposes the converging of ABE with proxy re-encryption, permitting fine-grained access control of resources which offloading re-encryption action to the cloud provider. The data owner is included in creating a key for every new client that joins or leaves the system, instead of offloading this assignment. It is not just a restrictive expense for a web user, additionally unrealistic because of the client's mobility. An alternate distinction is that a secret key must be [13] regenerated and redistributed for every client. Besides, the re-encryption happens because of attribute redefinition, the scheme is on key-policy attribute-based encryption (KP-ABE), and not CP-ABE, ciphertext is connect with a policy.

A multi-authority framework has that uses attribute based access control to break off the single point of failure of a single key authority, and depends upon the communication of data owner with attribute authorities to create huge encryption keys for facilitated content. The cost of communicating with numerous authorities and storing various keys could get to be restrictive for clients. In a related work, every client was submits different secret keys release by authorities to a server to produce a decrypting token for every ciphertext, that is utilize the client's global secret key to perform a decryption.

3. Model of web based and mobile cloud computing

3. 1 System model

A cloud service provider (CSP), simply referred to as a "cloud," provides permanent data storage in a centralized data center or a small host of geographically dispersed but interconnected centers. Users may directly access the user data stored in the cloud over the public Internet infrastructure by referencing a particular data partition.

Appeals are made over public Internet, which is considered as a regularly reliable however unreliable medium. A network access model shown in Fig. 1. Internet traffic is directed through a topology of network switches, which culminate in individual Internet service provider (ISP) network switches joined by high-capacity optical links like OC-3,This parcel data network may be crossed over to a remote 3G or 4G base through an entryway hub, permitting cell phones to associate remotely to 3G or 4G towers and switches. Furthermore, cell phones may convey among themselves utilizing short-run nearby connections, for example, Bluetooth. An alternate section point into the Internet is by means of a switch and Wi-Fi access point, empowering journals and remote sensors to join through some Wi-Fi.

The framework model of the proposed work is indicate in Fig. 1. Inside the public cloud, a controller administrates access through outside client interfaces. Requests, including data uploads and downloads are made over the reliable but insecure medium of the Internet.

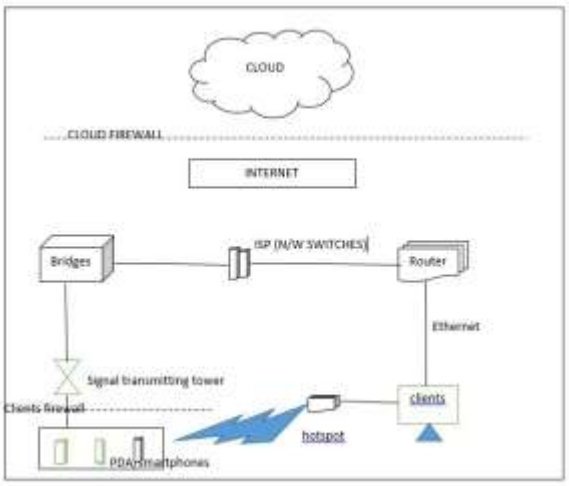

Fig 1: Cloud network model

A wireless packet data infrastructure serves to extension mobile clients. The supervisor is a trusted in supporting network component that may be situated behind an organization firewall Constant intervention by a single data owner during all transactions is unfeasible, as a web client is liable to constrained battery lifetime and transient network. The network some piece of a private cloud having a place with the client. It is consider fast and scalable, but less capable than the public cloud. It may maintain the database of private key data regarding to a set of authorized mobile users. The controller maintains a complementary public key information database, and stores and reads user data on behalf of clients to and from the permanent and replicated data store.

A client may go about as a data manager and choose what access benefits are proper for the information that it uploads to the cloud and retains control over a particular subset. The Fig 2 which refers the client population may be recognized as having sufficient authorization based on unique identities, or clients may be assigned out different attributes that characteristically allow consistent paying little respect to the particular identity that expect them.

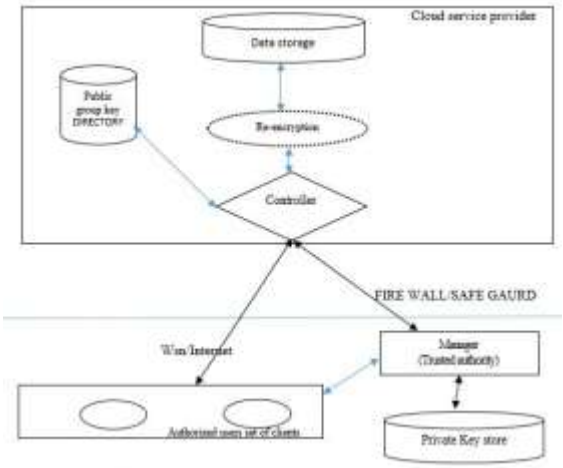

Fig 2: Cloud computing system model.

3.2 Trust model

The cloud provider is assume to completely honest in practice. A cloud administrator may read the data of user's information stored in the cloud for malicious purposes or essentially just curiosity. A party with administrator benefits may even duplicate or alter data without knowing the clients. Therefore, data stored in the cloud remain encrypted at all times, and any required changes or transformations is should not be reveal the plaintext simultaneously. The communications channel between a user and the cloud is open and subject to eavesdropping or data leakage. Thus, sensitive user data may not be exchange in the clear.

The administrator is a trusted power inside the framework and is control under the space of the clients being refer too; it is very autonomous of the CSP. It is sufficiently trusted to approve access to the cloud and to contain key material as basics.

Notwithstanding, to minimize the danger of it being traded off, a client will just impart as quite a bit of its own key material with the administrator as is vital in the security plan used. Moreover, the supervisor will not be as practical an open cloud supplier because of its more restricted computational assets.

Table 1: Notations and its description.

S.no	Symbol	Description
1	P_K	Public Parameters
2	M_K	Master Key
3	C_T	Cipher Text
4	S_K	Secret Key
5	T	Access Structure
6	S_K	Attribute Set
7	M	Merge
8	S_T	Nodes in access tree
9	S	Set of Attributes
10	PPK	Partition public key
11	OSK	Owner secret key
12	GSK	Group secret key

4. Algorithm analysis

The existing algorithm for the key generation, the distribution, and the usage of the clients. Algorithm, which consists of key management techniques, which ensure the data outsourcing [3] to the cloud in a high adaptable manner for the mobile and web clients. The above table, which summarizes the notations of the algorithm and description done in creation of the key. The following improvements done in the CP-ABE algorithm. Such that key components have reassigned to the various entities in the system model to achieve scalable key management while reducing the mobile user and web users computational and communication workload:

- A single authority does not produce all key data to the data owner and cloud entity coordinate to together process keys. The cloud provider has deficient data to interpret the client data that permanently stores; yet, it supports in the circulation of a share of the entire key material to all authorized clients to minimize the communication cost for the data owner.
- Proxy-based re-encryption has been coordinated with CP-ABE, the cloud provider may perform automated data re-encryption, this is an optional which permits further control over revocation than is managed by an attribute-based scheme, and it advantages the cloud provider computational versatility. This double encryption plan is, hence, a hybrid methodology, joining cryptographic methods, that offers greater adaptability in access control.

A CP-ABE scheme has the following four algorithms:

Setup:

This randomized algorithm takes a security parameter as inputs, and yields public parameters (PK) and a master key (MK). PK utilized for encryption and MK is to create user secret keys and known just to the central authority.

Encryption:

The randomized algorithm that takes as input parameter a message M and an access structure T, and the public parameters PK. It yields the output the ciphertext CT.

KenGen:

The randomized algorithm that takes as input parameters as the set of a user (say X)'s attributes SX and the master key MK and yields outputs a secret key SK which identifies with SX.

Decryption:

This algorithm takes as input parameters as the ciphertext CT, and a secret key SK for an attribute set SX. If SX satisfies the access structure integrated in CT, it will generates the original message M.

The security authority holds the master key MK and distributes public parameters PK. Utilizing PK; a client can encode a cipher message, M utilizing any arbitrary access structure T in view of an arrangement of properties, S. It is significant that there is no specific public key for a client in this framework as opposed to normal public-private key system, for example, RSA. Additionally, the size of the ciphertext is of O (ST), if there are ST, nodes in the access tree, T. A client's secret key is dictate by the subset of qualities will claims. Subsequently, her private key size is O (τ) if it will possesses τ attributes.

Let G0 be a bilinear gathering of prime order p, and let g be a generator of G0.in addition to it, let e: G0 ×G0 → G1 signify the bilinear mapping. We additionally utilize a function H: Z∗ p →G0 to guide any ascribe to a component in G0, where H(i) = gi. Our crypto framework comprises of the accompanying algorithms.

Setup: This algorithm will identifies three arbitrary types a, b, γ ∈ Zp. In the fundamental development, a parameter d specifies the number of attributes each private key has. One polynomial v(x) of degree d is picked at arbitrary subject to the constraint that v(0) = b. The public parameters are as per the follows.

$$PK= [G_0, g, h=g^b, h^\gamma, f = g^{\frac{1}{\gamma}}, e(h,g)^a, \{g^{\theta(0)}, g^{\theta(1)} \dots \dots . g^{\theta(d)}\}]$$

Encryption (PK, M, T): The encryption algorithm encodes a message M, M∈G1 under the tree access structure T. The algorithm first picks a polynomial qx for every node x (combing with the leaves) in the tree T. These polynomials are gather in the forwarded path in a top-down way, beginning from the root node R. For every node x in the tree, set the degree dx of the polynomial qx to be one not exactly the threshold kx of that node, that is, dx = kx −1. Beginning with the root node R the algorithm picks an irregular s∈Zp and sets qR(0) = s. At that point, it picks dR different points of the polynomial qR arbitrarily to define it totally. For whatever other hub x, it sets qx(0) = qparent(x)(index(x)) and picks dx different directs haphazardly toward totally define qx. Let, Y1 and Y2 be the arrangement of the leaf hubs in T with positive characteristics and negative properties, individually. For every hub y ∈ Y2, we arbitrarily pick uy from Zp. We define a capacity V(i) : Zp → G as V(i)=gv(i).

Note that however v(i) is not known to the encryptor, the process V(i) by introduction utilizing the set {gv(0),gv(1),...,gv(d)} accessible as a part of people in general key, PK. We additionally review that capacity F maps every credit to a component in Z∗ p. The ciphertext CT is develop by giving the tree access structure T and figuring alternate components as follows.

$$CT= [T, C1=Me(h,g)^{as}, C2 = (h^r)^s,$$
$$\forall nodes\ y \in Y1 :\ C1_y = g^{q_y(0)}, C2_y = H(i)^{q_y(0)}$$
$$where\ \ i = F(att(y)), and$$
$$\forall nodes\ y \in Y2 :\ C3_y = h^{q_y(0)+u_y},$$
$$C4_y = (V(i))^{u_y}, C5_y = g^{u_y}$$
$$where\ i = F(att(y))]$$

KenGen (MK, S): The key generation algorithm will takes as inputs a set of attributes S ⊂Z∗ p and yield a private key that identifies with S. The algorithm first choose an irregular r∈Zp, and after that irregular r∈Zp for every trait j ∈S. At that point, it computes the key as follows

$$SK= [D=g^{(a+r)/\gamma}, D1 = g^r, \forall j \in S :$$
$$D1_j = h^r.H(j)^{r_j}, D2_j = g^{r_j},$$
$$D3_j = (V(j))^r]$$

We expect that size of set S is fixed, which is the greatest number of properties connected with a user. Practically, we can get around this confinement as takes after. If a client with Id X is connected with d0 attributes where d0 < d, then we can simply include (d−d0) filler attributes in X's set of properties, for example, "X: Filler1", "X: Filler2", and others. We expect that d is a little steady.

Delegate (SK, S): The delegation algorithm chose the parameters as a secret key SK, which is of set S of attributes, and another set that is $\bar{S} \subset S$. The algorithm chooses random $\bar{r}$ and $\bar{r}k\ \forall k \in \bar{S}$. Then it yields a new secret key as

$$SK = [\bar{D} = D.f^{\bar{r}}, \bar{D}1 = g^r.g^{\bar{r}}, \forall k \in \bar{S} : \bar{D}1_k =$$
$$D1_K.h^{\bar{r}}.H(k)^{\bar{r}k},$$
$$\bar{D}2_k = D2_K.g^{\bar{r}k},$$
$$\bar{D}3_k = D3_K.(V(k))^{\bar{r}}]$$

Decrypt (CT, SK): We determine our decryption method as a recursive algorithm. For simplicity of explanation, we exhibit the least complex manifestation of the decryption algorithm. We may adjust the advancements reported in to outline a more efficient decryption algorithm. We first define a recursive calculation Decrypt Node (CT, SK, x) that takes as include a ciphertext CT, a private key SK, which is connected with a situated S of attributes, and a node x from T.

$$\text{DescryptNode (CT, SK, x)} = \frac{e(D1_i, C1_x)}{e(D2_i, C2_x)}$$
$$= \frac{e(h^r\ ,H(i)^{r_1}, g^{q_x(0)})}{e(g^{r_1}\ ,H(i)^{q_x(0)}}$$
$$= \frac{e(h^r, g^{q_x(0)}) X\ e(H(i)^{r_1}, g^{q_x(0)})}{e(g^{r_1}\ ,H(i)^{q_x(0)}}$$
$$= e(h^r, g^{q_x(0)}) = e(h,g)^{rq_x(0)}$$

$$\text{DescryptNode (CT, SK, x)}$$
$$= \frac{e(D1_i, C3_x)}{e\left(C5_x, \pi_{j\in s}(D3_j)^{\sigma_j}\right).e(D1, C4)^{\sigma_i}}$$
$$= \frac{e(h,g)^{rq_x(0)}.e(g,g)^{rbu_i}}{e(g,g)^{u_i r \sum_{j\in s'} \sigma_j \theta(j)}}$$
$$= e(h,g)^{rq_x(0)}$$

For decrypting the cipher, text by computing in the following algorithm will used:

$$C1 \Big/ \left(\frac{e(C2, D)}{A}\right)$$
$$= C1/(e(h^{rs}, g^{\frac{a+r}{\gamma}})/e(h,g)^{rs})$$
$$= \frac{M.e(h,g)^{as}}{e(h,g)^{as}} = M$$

5. Implementation analysis:

The protocol was actualize and profiled to its execution and performance. It acknowledged on well-known existing business platforms, including the Google Android mobile and the Google App Engine (GAE) cloud stages. A simulation adjusted to the performance benchmarks was then process to inspect the versatility of the existing algorithm.

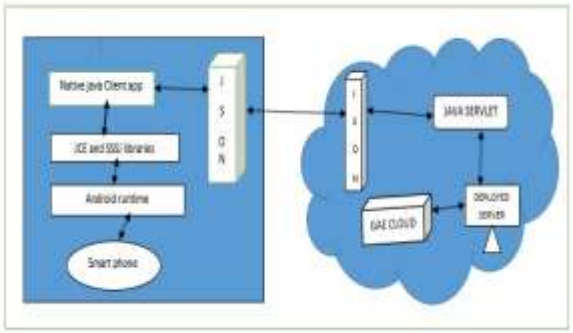

Fig 3: Model of Implementation

Here from the fig 3: the implementation has described, as that JSON which acts a middle ware between the clients. The java servlet, which transfers the data to the cloud server, and GAE, which will accessed by the authorized users who have the keys. The transformations have been occurred in the client side is to transfer the data by using the JCE and SSSI. By using the compiler, android runtime the data can be access by the user.

6. Performance Analysis

Performance implications are modest for web based and mobile clients. The data owner should just perform operations amid its key generation stages, while the administrator [6] performs the more valuable pairing operation in processing partition key generation in the SETUP. Additionally, with the help of the manager, the client performs just a single bonding or pairing operation on decryption.

The group key may transferred among a group or simply controlled by a single user; the tradeoff made in its utilization is the obliged circulation of the gathering key and the additional blending operation presupposed amid the encryption stage, yet the cloud provider favorably processes it. The manager can also aid with distribution. Besides, it might be favorable for the administrator to process new key forms and re-encryption keys, and deal with their capacity and appropriation. A suitable key version component is recommend.

Table 2: performance measurements based on group key and without group key.

By using the attributes of the users the performances between the group key with algorithm and the without group key. In the performance calculations has revised below; where the following symbols are used: μ is the sample mean, Σ is the sample standard deviation, and LB and UB are the lower and upper bounds, respectively.

7. Results

A custom simulation was produce that allows an appraisal of the adaptability capability of the existing scheme. The process was execute on a desktop PC yet it was align with the capacity timing results from the benchmarks acquired as portrayed in the past segment. That is, the timings served as the premise for computing the collected transforming workload of the different systems:

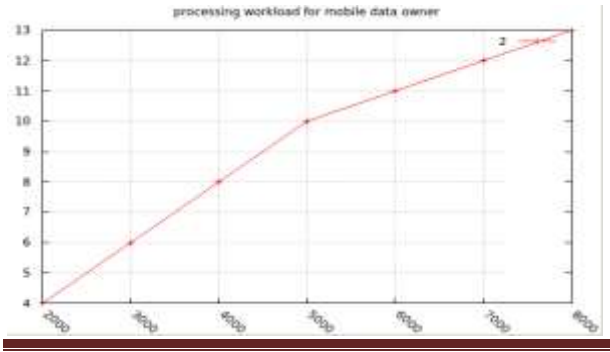

Fig 5: processing workload for mobile data owner.

Fig 6: processing workload for manager.

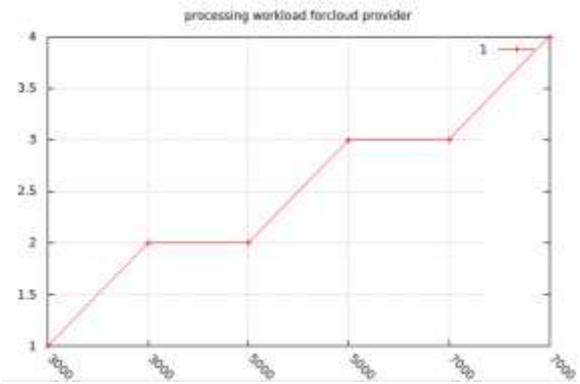

Fig 7: processing workload for cloud provider.

S.no	Algorithm	Task	Timings(in msec)			
			μ	Σ	LB	UB
1	Proposed algorithm (without group key)	Owner setup	30	44.5	0	119.1
		Manager setup	20	3.1	14.9	24.3
		Keygen	73.2	4.9	64.0	84.1
		Owner encryption	64	4.8	52.9	75.9
		Decryption	23.9	3.1	18.9	30.2
2	Proposed algorithm (With group key)	Owner setup	38.1	25.7	0	89.0
		Manager setup	17.9	1.1	15.9	22.1
		Keygen	71.2	2.9	62.9	75.9
		Owner encryption	61.0	2.1	55.4	69.6
		Cloud encryption	19.8	1.2	17.0	22.5
		Decryption	41.9	2.3	39.9	43.9

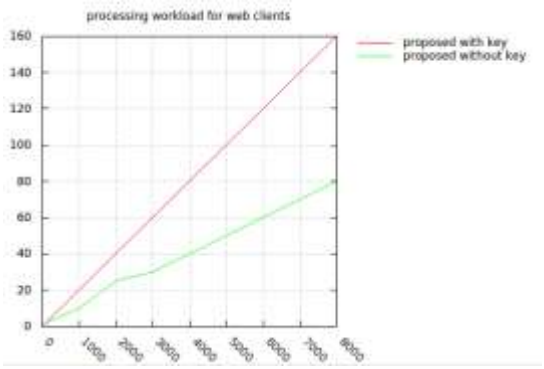

Fig 8: processing workload for web clients.

A beginning unapproved client population is display, and in each round of the simulation, clients will join or leave a client set that is approved to get to a specific information record. A single data owner in charge of information encryption is display. Every client randomly makes an action for each round, with some predefined calculations; activities incorporate getting to the encrypted data and performing a decryption, joining or leaving the authorized client set. The encrypted information record stored in the cloud might likewise be replace in a round by the data owner, once it has outlasted its convenience with later information, which launches another key setup stage.

8. Conclusion

In this paper revived all the key management system has been used for secure the data such as outsourcing applications, whereby attribute-based encryption significantly permits authorized clients to access the secure content in the cloud based on the satisfaction of an attribute based encryption. This is of significance to a population of versatile clients that must monitor their utilization of battery and use of remote communication. Therefore, suitable technique can used to avoid or to prevent from losing the data based on throughput, delay and traffic and battery drainage.

9. References

[1] *R.L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," proceedings of. ACM, vol. 26, pp. 96-99. 1983.*

[2] *N.Balasubramanian, A.Balasubramanian, andA.Venkataramani, "Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications," Proceedings. Of Ninth ACMSIGCOMM Conference Internet Measurement Conference, pp. 280-293, 2009.*

[3] *J. Bettencourt, A. Sahai, and B. Waters, "Cipher text-Policy Attribute-Based Encryption," Proceedings of IEEE Symposium on Security and Privacy, pp. 321-334, 2007.*

[4] *G. Zhao, C. Rong, J. Li, F. Zhang, and Y. Tang, "Trusted Data Sharing over Untrusted Cloud Storage Providers," Proceedings of IEEE Second international Conference Cloud Computing Technology and Science , pp. 97-103, 2010.*

[5] *S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute Based Data Sharing with Attribute Revocation," Fifth ACM Symposium on Information, Computer and Communication Security, pp. 261-270, 2010.*

[6] *Shaikh, F.B, Haider, S.; "Security threats in cloud computing" Internet Technology and Secured Transaction (ICITST), International Conference for Dec. 2011, pp: 214-219, 2011.*

[7] *Seungwon Shin, Guofei Gu; " CloudWatcher: Network security monitoring using OpenFlow in dynamic cloud networks (or: How to provide security monitoring as a service in clouds?)" Network Protocols (ICNP), 2012 20th IEEE International Conference Pp: 1 – 6,2012.*

[8] *Yegui Cai, Yu. F.R, Shengrong Bu; "Cloud radio access networks (C-RAN) in mobile cloud computing systems" Proceedings of Computer Communications Workshops (INFOCOM WKSHPS), Pp: 369 – 374,2014*

[9] *Caifeng Zou , Huifang Deng , Qunye Qiu; "Design and Implementation of Hybrid Cloud Computing Architecture Based on Cloud Bus" Proceedings of Mobile Ad-hoc and Sensor Networks (MSN), Pp: 289 – 293,2013.*

[10] *Li Ling, Ma Xiao Zhen, Huang Yulan; "CDN cloud: A novel scheme for combining CDN and cloud computing" Proceedings of Measurement, Information and Control (ICMIC), 2013 International Conference, Pp: 687 – 690, 2013.*

[11] *J.Hur and D.K. Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems," IEEE Transaction Parallel and Distributed Systems, vol. 22, Pp: 2317-2322, 2011.*

[12] *G. Wang, Q. Liu, and J. Wu, "Hierarchical Attribute-Based Encryption for Fine-Grained Access Control in Cloud Storage Services," Proceedings of 17th ACM Conference Computer and Communication Security (CCS '10), Pp: 925-930, 2010.*

[13] *Y. Ming, L. Fan, H. Jing-Li, and W. Zhao-Li, "An Efficient Attribute Based Encryption Scheme with Revocation for Out- sourced Data Sharing Control," Proceedings of First International of Conference Instrumentation, Measurement, Computer, Communication. and Control,Pp:423-429,2011.*

[14] *S. Jahid, P. Mittal, and N. Borisov, "EASiER: Encryption-Based Access Control in Social Networks with Efficient Revocation," Proceedings of Sixth ACM Symposium on Information on Computer and Communication Security (ASIACCS '11), Pp: 1095-2003, 2011.*

[15] *Encryption Scheme with Revocation for Out- sourced Data Sharing Control," Proceedings on First International of Conference Instrumentation, Measurement, Computer, Communication and Control, Pp: 423-429, 2011.*

Mothukuru Yugesh received B.Tech. Degree in information Technology from JNTU University-Ananthapur in 2013 and pursuing M.Tech from VIT University, Vellore, India. His research interests include Network Security and Internet Programming.

Dr. K. John Singh received Ph.D. degree in the Faculty of Information and Communication Engineering from Anna University, Chennai, India in 2013. He received M.S degree in Information Technology from Manonmaniam Sundaranar University, Tirunelveli, India in 2002 and M.Tech degree in Computer and Information Technology from Center for Information Technology and Engineering of Manonmaniam Sundaranar University, Tirunelveli, India in 2004. Currently, he is working as Associate Professor in the School of Information Technology and Engineering, VIT University, Vellore, India. His research interests include Network Security and Cloud.