Secured Information Sharing in Mobile Cloud Computing using Access Controls

Kaja Masthan, K. Venkatesh Sharma

ABSTRACT: Presently usage of smart mobiles increasing, due to internet availability most of users outsourcing their data to cloud but it is untrusted, so a security mechanism needed in this work proposing Homomorphic cipher text policy-Attribute based encryption (HCP-ABE), it derived from classic ABE. "Mobile cloud computing are combinations of mobile computing and cloud computing", mobile applications are designed and hosted in cloud computing without verifying about mobile environment. *Security models are constructed using Perturbation methods as per literature. These models are not secure compared to cryptographic techniques. When perturbation methods are used, Data reconstruction becomes a significant challenge. Hence computations are complicated to perform. Furthermore, this method suffers a trade-off between accuracy and privacy* ***and most of the research work focused on key management issues and static access policies but due to user dynamic the access control mechanism should design for proactive strategies. To support dynamic access control and operations HCP-ABE scheme intended. In this paper we identify challenges associated with mobile cloud-based security system and possible provide solutions to understand existing research work conducting compressive review on different access control mechanisms.***

Keywords: Cloud computing, Mobile Cloud computing (MCC) Access controls, Homomorphic encryption and ABE.

I. INTRODUCTION

Cloud Computing has become an integral part of our day to day life. We can see the applications of cloud used everywhere either it could be web applications or mobile applications, IOT Applications or Data-based applications, the cloud has become a common term in the IT Industry. Even a layman is also using the cloud with or without the knowledge of cloud. According to a report presented by Statista portal the number of cloud-based consumers has been increased from 2.4 billion in 2013 to 3.6 billion in the year 2018. The world's total population is 7.6 billion people, and if you see the previous statistics from Statista portal, half of the world's population is directly or indirectly accessing/consuming the Cloud Services. When such a vast number of people use the cloud services by storing and accessing data, you can imagine the kind of problems like Storages, Processing Speed, Security, Privacy, etc..,. Somehow, Cloud Service providers have tackled the issues mentioned above, but Security remains the most crucial concern which makes the developers or IT professionals think twice before making use of the cloud services and due to the popularity and availability of cloud computing now many organizations outsource their data to the remote server to prevent economic burden and share globally, cloud service providers are currently unreliable because of the many privacy challenges.

Cloud as the computing or processing of remote resources or services and these services are IaaS (infrastructure as a service), PaaS (platform as a service), SaaS (software as a service) and so on. Cloud can deploy in four ways, such as private, public, hybrid and community cloud.

Every user connects to the Internet and uses the IT infrastructure to meet their daily needs as the demand for the Internet increases, even the service delivered as software, platform, database, storage services, etc. cloud offers "Pay as you go" to the user, maximum benefits can achieve by using these services at a lower cost.

Mobile Cloud Computing [4] is a growing technology and it require is enlarge basically step by step. As of today, a massive measure of population has recognized it due to their mixed personal and business applications and counting. Normally Mobile Cloud storage supplies customers to proactively outsource their data and have the advantage of on-interest superior cloud software without getting local programming and tools apparatus. Despite of the fact that the factor of interest is fair, such a government is likewise taking up customers "physical management " of the outsourced statistics, which inescapably creates new safety risks towards the precision of their information in cloud. To begin working on data access management, initially a careful analysis is critical to find output of cryptographic calculations to ensure information operations on flexible could be fast and consistent. Client flexibility suggests "anytime, anyplace" is moving right to a reality.

II. BACKGROUND WORK

[1] MCC is a promising technology used by scientists and businesses to access shared resources from anywhere on the Internet. Users can also store their sensitive and confidential information on the cloud, which requires a prominent encryption scheme and an accurate access strategy to ensure privacy and security. Symmetric, asymmetric, and disparate attribute-based cryptography has valuable cryptographic schemes for securing sensitive data. The authors analyzed cryptographic patterns of existing data, such as RSA, KP-ABE, CP-ABE, and AES. Comparisons were made between them based on the cost of calculation and the cost of storage. In addition, the authors proposed an improvement scheme to increase the speed of RSA cryptography using the multitasking concept on the latest multicore processors.

[2] Accepting privacy and security conditions for Mobile and also Cloud systems made in isolation.

Revised Manuscript Received on October 30, 2019.
***** Correspondence Author

Kaja Masthan*, Reacher Scholar, Shri Jagdish Prasad Jhabarmal Tibrewala University, Rajasthan, India.

Dr. K. Venkatesh Sharma, Professor, Dept of CSE, C.V.R. College of Engineering, Hyderabad, Telangana, India.

This lack of coherence has effects if data will be traveling between such programs. Within this paper, we propose a Flexible structure that takes into consideration the security and privacy requirements of both systems also will be offering an easy instantiation blueprint for understanding an end-to-end solitude solution where data remains safely traded and absorbed Mobile or Cloud platforms.

[3] Taking advantage of this accessibility to centers, MCC growing number popular being part of technology. This availability enables mobile users to use the cloud-computing infrastructure to successfully overcome the limitations of mobile technologies, namely restricted data storage, computing capacity, and battery life lifetime.

III. MOTIVATION FOR RESEARCH

In traditional cloud model the data owner upload data to cloud but due to security issues, before uploading data must encrypt and after it can upload to cloud storage. Data user download data from cloud but due to encryption cannot open the data, to open the data user get secret key from corresponding data owner, for sending secret key to data users data owner must be in online. However, it is not possible for data owner to stay always in online, so the solution is a central authority. Even central authority is not fully trusted, and key management is the big challenge in this model.

Data access control is an efficient way to ensure the confidentiality and privileges in the cloud environment, Access control is defined as a policy or procedure and set of limitations that allows owner denies or restrict access over cloud data in order to access data to achieve this various techniques have been proposed but all techniques have different limitations. Mobile cloud computing (MCC) has many security models in first the security done with either symmetric encryption or asymmetric encryption, in first scenario the owner encrypt data with secrete key after transfer secrete key to appropriate user, where as in second scenario for encryption using public key for decryption secrete key, in some scenario central authority manages secrete key whenever user requesting then authority transfers secrete key, finally all these three scenarios secrete key should be shared by either owner or authority and it is not possible to owner to share key to user, whereas authority is not trusted. Access controls (AC) are suitable to overcome above three scenarios, but in this owner before uploading data to cloud storage can decides set of users but due access control mechanism limitations whenever dynamic policies modified then it becomes tedious process. Implementing AC in mobile cloud is a challenging task.

IV. ALGORITHMS OF ACCESS CONTROL

Identity-based encryption (IBE)

Here the data owner can specify access control over cloud data and the data owner while encrypting data he can specify identity of user’s identities are like email id, phone number or pan card no, but using IBE data owner can’t revoke the user, and for key generating data owner need an authority.

Fuzzy Identity-Based Encryption (FIBE)

Was proposed, uniqueness a collection of detailed characteristics. Here data owner (dw) can define an assortment of qualities instead of single identity like IBE scheme, here also dw can stipulate data access control but he can’t revoke the particular user.

Attribute Based Encryption (ABE)

Back in ABE Accessibility Controller technique, an encryptor having a list of descriptive features tags just about every ciphertext. Each personal secret is related to an entry arrangement which defines which kind of ciphertext that the key may synthesize. It merely allows a authority to issue keys which state brink access coverages, by that a definite amount of stated features have to become contained from the ciphertext, to allow an individual to sew.

Access shrub arrangement is utilized to make the secret. From shrub structure, every non-leaf node signifies a brink and Foliage node of this shrub is also clarified by means of feature and threshold worth. Even the ciphertexts are given a group of attributes that are descriptive. Personal keys are identified with way of a tree-access arrangement by which every single inner node of this Shrub is really a brink club and also leaves are all correlated together with features. An individual Is likely to have ability to decrypt a ciphertext having confirmed secret if and as long as there is certainly A mission of features by your ciphertext into elements of the tree for example the shrub is more fulfilled.

KP-ABE

Inside KP-ABE strategy Ciphertexts related to collections of descriptive features, also people' means are all directly correlated together with coverages, that clarifies consumer's individuality, ciphertext can be decrypted by an individual if and only in case features inside their personal secret is fulfilled from the ciphertext. The security plan is clarified or so the encrypter doesn't need total control within the encryption plan. He's got to expect keys are issued by generators together with structures that are correct to users. Moreover, after re-encryption happens, users in same arrangement all necessity need their private keys re-issued so as to access their documents, also this procedure causes issues in execution.

CP-ABE

Back in CP-ABE [9] every single and every consumer is correlated having a pair of features. User key is related to an entry arrangement. Statistics are encoded within a couple of features. The decryption of information necessitates the info features to meet consumer accessibility arrangement. Their identities need to be comprised of the private key of feature checklist to find a person. That was not any consumer identity data in this ciphertext. This strategy uses Bilinear Map premises where feature is directly connected with users' key. It will take benefit of the operator. It's immune to collusion assault.

CPASBE

CP-ASBE can be just really actually a kind of CP-ABE which utilizes recursive established based arrangement on features related to keys. Makes it possible for user features to become coordinated into a group of coverages and places which may prohibit people permit them to unite traits from sets or to make work with of features.

Comparison-Based Encryption for “Fine-grained Access Control” in Clouds

The attributes like level, time, position and location are also considered under ordering relation which can be mapped to integers. The values of attributes are made countable and range constraint is imposed.

Temporal based access control Encryption (TACE)

The model is constructed by integrating temporal access control solution [6] along with a proxy-based re-encryption mechanism. It uses forward and backward derivative function and uses integer comparison rather than bitwise comparison. This model uses current time while checking for the attribute of each user before providing access to the data. It uses one-way property to represent the total ordering relation in integer.

V. DATA SECURITY & CRYPTOGRAPHY IN MOBILE CLOUD

Data security is matter of utmost concern these days and due to heavy usage of online applications, users tend to outsource their data on remote servers. Use of proper cryptographic technique may save data from various hazards. In this chapter we provide comprehensive details about data security using various cryptographic techniques, attacks on those techniques and comparisons between popular cryptography algorithms to assess the best suited cipher technique for mobile users.

Upon so, occasionally of transfer prices is more problematic than initially glance. Most SAs require a Few user Quotes to deliver improved Code profiling works nicely on CPU intensive jobs however neglects Sometimes It's even debilitating for consumers to Offer Run Time Assessing non-preemptive functions after they transferred. The difficulty which don't require them whatsoever. These calculations take under account merely resource load and proceed tasks just once their abilities eventually become erratic. This approach is useful, and evaluations have demonstrated that programming heuristics like round robin provide results like some other classic heuristics-based on run time quotes.

VI. CATEGORIZATION OF HOMOMORPHIC-ENCRYPTION

A cryptosystem may hold homomorphic-encryption attributes beneath several scientific processes. On this basis, we can describe a homomorphic-encryption into two classes:

- The “fully homomorphic-encryption”
- The “partially homomorphic-encryption”

The fully homomorphic cryptography assist adding and multiplication process on encrypted data, while partial homomorphic cryptography helps merely one process on encrypted text. The RSA coded system, “Paillier, Goldwasser Micali,Elgamal” and NaccaheStern has an unjust homomorphic property, while Craig Gentry's cryptographic system in 2009 has a completely homomorphic characteristic. Later, this cryptographic system was revised by Van Dijk, Craig Gentry, Halevi and Vaikuntanathan. Table 1 offers particulars on homomorphic operations supported by different cryptosystems with public keys.

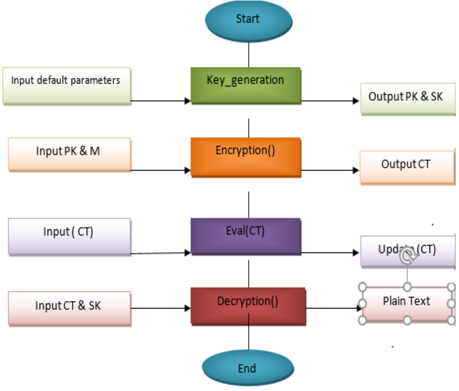
Table 1 Comparative analysis of Homomorphic-operation

Cryptosystem	**Homomorphic operation**
RSA-Asymmetric	X OR ||
Homomorphic-Paillier-	X, +
The Homomorphic-Elgamal	X, + & -
Boneh-Goh	X, +, e^1
Benaloh	+, -
Naccahe	X^1, -, +
Micaliac	XOR
YoungYung	X

The homomorphic cryptography scheme consists of four algorithms:

{ KeyGen, encrypt, evaluate, decipher}

- **Key generation**: In this step, enter as default parameters and issue as public key (PK) and secret key (SK), public key internally contain set of attributes.
- **Encryption (PK, M)**: input as public key and plain text (M) and produce output as cipher text (CT).
- **Eval(f(CT))**: input as cipher text and f indicates function which will perform on cipher text and output CT^*
- **Decryption (SK, CT)**: input as SK and CT, if SK satisfies set of policies in CT then CT will decrypt M.



The above flow chart describes HE different steps and each step consists input, process and output. First step key_generation(), it takes input as default parameters and produce output as public & secret key. In encryption () step input is PK and plain text M after encryption process it produce Cipher text CT. in Eval(CT) step perform computation on CT and produce updated CT. In Decryption () step input updated CT and SK and if secret key satisfies then it produce plain text.

First operation of homomorphic_encryption:

Enc $(\pi_{i=1}^{n} Z_i) = \pi_{i=1}^{n} \text{Enc}(Z_i)$

Now Enc () is cryptographic operation it signify me the normal text.

Second operation of homomorphic_encryption:

Enc $(\sum_{i=1}^{n} Z_i) = \pi_{i=1}^{n} E(Z_i)$

UpdEnc $(\int_1^n \quad CT1, CT2 \ldots \ldots . CTn.$

Algorithm for RSA Encryption:

```
Algorithm RSA_Encryption ( )
{
CT=PT^1 mod L(“CT= cipher text & PT= plaintext)
```

$$\mathrm{Enc1}(PT1 + PT2)^n = \sum_{J=0}^{j+1} \begin{pmatrix} \mathrm{g} \\ \mathrm{h} \\ \mathrm{k} \end{pmatrix} enc^k g^{m-n}$$

}

3.7.3. RSA_Dec():

Algorithm RSA_Decryption ( )

{

$$\mathrm{Dec1}(CT1 + CT2)^n = \sum_{J=0}^{j+1} \begin{pmatrix} \mathrm{PT1} \\ \mathrm{PT2} \\ \mathrm{PT3} \end{pmatrix} dec^{\mathrm{W}} \mathrm{h}^{g-r}$$

}

Homomorphic_cryptosystem RSA():

RSA_Homocrypto operations ability to perform Homomorphic_Enc() & Homomorphic_Dec() operations be multiplication operation. This involve to facilitate user desires to multiply original texts PT1 and PT2, then converted CT1 and CT2 accumulate at server end will be multiplied and accumulated secondary converted message is equivalent to the encrypted variant of original message PT1.

Table 2: Homomorphic Property of RSA cryptosystem

Homomorphic Property	Example
$E(\pi_{i=1}^{n} Q_i)$ =[$\pi_{i=1}^{n}$ Enc(Q_i)]mod T P_1P_2= D(C_1C_2) Or E(P_1P_2) = C_1C_2	P_1=6, C_1=41 P_2=5, C_2=31 P_3=P_1P_2=30 C_3 = C_1C_2 = 1271 Enc(44*51) = [41*31] mod 119

ADVANCED ENCRYPTION STANDARD (AES):

AES was presented by "Joan Daemen and Vincent Rijmen". At the point when deliberated organized Rijndael's mix of security, execution, productivity, actualize capacity, and adaptability made it a suitable selection for the AES. By design AES is quicker in programming and works effectively in equipment. It works quickly even on small devices, advanced cells; keen cards etc.AES gives more security because of bigger bit size and more keys.AES uses 128 piece altered square size and works with 128, 192 and 256 piece keys. Ragsdale calculation is sufficiently adaptable to work with key and block size of any several of 32 bit with least of 128 bits and maximum of 256 bits.AES is replacement for 3DES as indicated by NIST both cipher will exist together until the year 2030 taking into consideration slow move to AES.

DES Algorithm

The DES algorithm stands for Data Encryption Standard Algorithm. The DES algorithm was established in 1970 when it was necessary for a government standard to encrypt susceptible information. The DES was developed by IBM, and was approved by NSA.

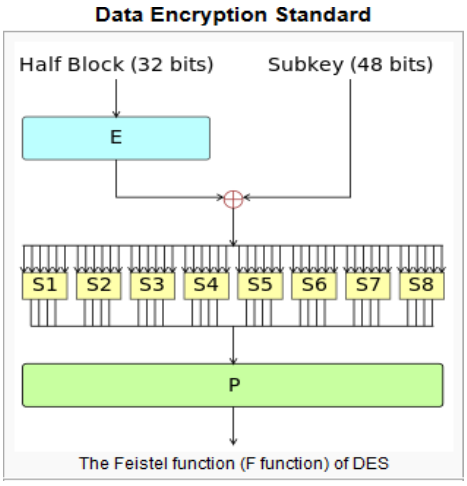


Figure 1: Data Encryption Standard

DES is a symmetric lump of mystery code actualized by IBM. DES uses a 56- bit key to encode/interpret a 64-bit piece of message. The key at all the times kept as a 64-bit lump, each eighth piece of that is disregarded. Yet, it is regular to put each eighth piece so that each arrangement of 8 bits has an odd no. of bits spots to 1.

This cryptographic standard is principal to actualize in h/w, maybe to it can be executed in applications also, yet contrast with equipment, operations sets to the side additional time in programming. However, modern registering gadgets are speedy to the point that we get satisfying results.

The Data Encryption Standard Algorithm makes use of the symmetric key concept, such that the same key that was used for the encryption process can also be used for the same key used for the decryption process. According to the Data Encryption Standard Algorithm, a 64-bit number key is being used. Among those 64- bits, 56 bits are generated randomly and are directly used for encrypting the information. Then the left over 8 bits are useful in error discovery process [11].

Initially there was a suggestion about the DES algorithm, that when this algorithm was developed, the 56 bits are very small for offering best security. Still the Data Encryption Standard Algorithm was believed to be secure, and thus it came into practice during 1977. During 1993, an update on the existing government standard called Triple DES was developed. This Triple DES proved to be better secure method of DES. During 1997, the original Data Encryption Standard Algorithm crashed because the AES algorithm known as the Advanced Encryption Standard algorithm replaces the DES algorithm.

The AES is more secure than DES and 3DES - since the calculation is straight forward and uses more key lengths. It enables quicker encryption than DES and 3DES, which makes it ideal for programming programs, firmware and hardware which need either low-dormancy or higher throughput, by way of instance, firewalls and switches. It's used as part of many conventions, by way of instance, SSL/TLS and could be located in many cutting-edge devices and applications that require encryption usefulness. Advance Encryption Standard (AES) and Triple DES (TDES or 3DES) are usually used block cipher.

Whether you may select AES or 3DES, it depends upon your own requirements. Within this section their disparities concerning safety and implementation is emphasized.

VII. RESULTS AND ANALYSIS

Primarily in this work performs proportional analysis of prevailing public crypto-system based on parameters, i.e. coding time, decoding time and encrypted file size. Their consequences go behind: Considerations of AES, DES and 3DES calculation for versatile information security was taken after a brief exploration on their configuration and security highlights. Experimental Results obtained through this method are provided in results section which demonstrates the efficiency of AES over other algorithms on mobile environment. As AES is proven to be more efficient for mobile devices so AES is considered as a base encryption technique in rest phases of work.

After comparing trios' cryptographic systems on several parameters In fig 2,3 & 4, it becomes discovered that RSA is suitable amidst the three public cryptographic methods discussed on Parameters, such as cryptographic time, decryption time and encryption size, the application of is the version encrypted difference. So, as regards the patch of, Size le, same association in terms of encrypted. So RSA will override the other two public cryptographic systems regarding the size of patch.

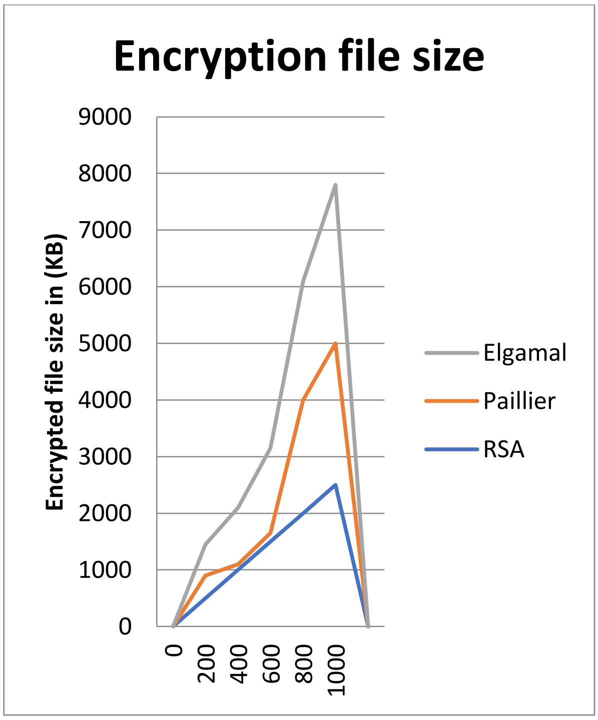


Fig: 2 Comparison based on encrypted file size

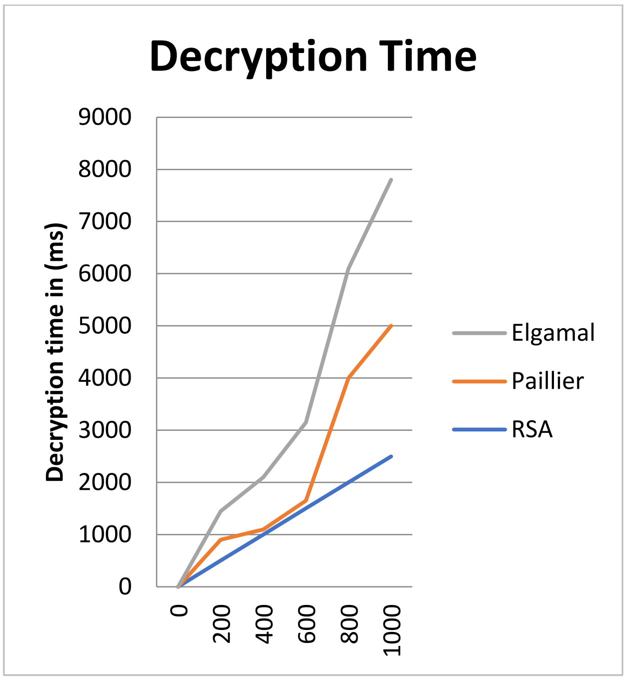


Fig 3. Comparison of different crypto algorithm for Decryption time

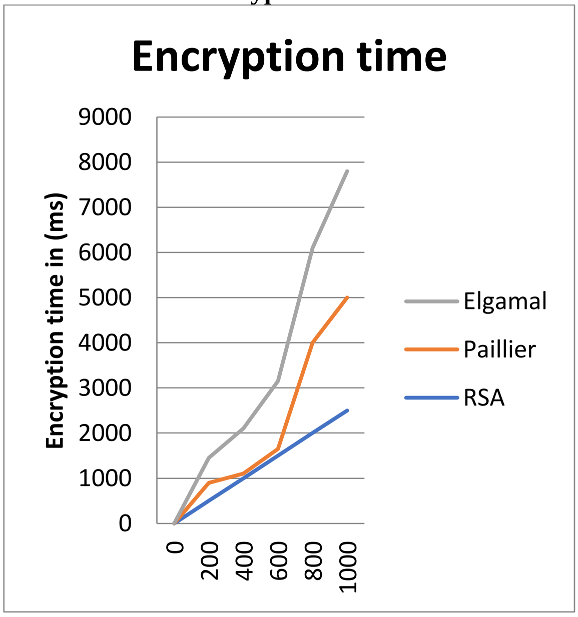


Fig 4. Comparison of different crypto algorithm for Encryption time

VIII. CONCLUSION

In this work, addressed the problem of dynamic policies and operations on Mobile cloud, generally in Mobile cloud storage in order to control outscored data, DW enforces access control mechanism by using this he can grant the privileges to set of desired users or revoke the privileges of particular users so to achieve this many access control mechanisms are implemented, like Identity based encryption (IBE), ABE and CP-ABE etc.

However, all these existing access control mechanisms are static means if once policy is defined on encrypted text if any user policies are changes why because users are dynamic, not static once polices updated privileges of user also get update or if any user removed or owner want to want to take back the privileges which are assigned previously for this these circumstances existing access control mechanisms not suitable for MCC. Sometimes once inserting data toward cloud, owner could want to update data or remove data so policies should be updated, so the objective of this work is if any user policies are changes or if any dynamic operations are done on a cloud server without changing any user policies over encrypted data the cloud data should be updated. To achieve this new access control mechanism should require, in this work proposing new access control scheme called Homomorphic cipher text policy Attribute-based encryption (HCP-ABE) which is designed based on Cipher text policy Attribute-based encryption (CP-ABE).

REFERENCES

1. Afnan Babrahem & Muhammad Monowar, 2017, “Maintaining security and privacy of the Patient's EHR using cryptographic organization based access control h cloud environment”, PP: 182-188.
2. Amin Fallahi; et.al, 2017, “Towards Secure Public Directory for Privacy-Preserving Data Sharing”, ISSN: 1063-6927, PP: 2577-2578.
3. Assad Abbas; et.al, 2015, “A Cloud Based Framework for Identification of Influential Health Experts from Twitter”, PP: 831-838.
4. Prasadu Peddi, 2018, Data sharing Privacy in Mobile cloud using AES, ISSN 2319-1953, volume 7, issue 4.
5. Avuya Mxoli; et.al, 2014, “Information security risk measures for Cloud-based personal health records”, PP: 187-193.
6. Iniya Shree et al. 2016, “An multi-authority attribute based encryption for personal health record in cloud computing”, PP: 1-5.
7. Lucas et al. 2014, “A new architecture for secure storage and sharing of health records in the cloud using federated identity attributes”, PP: 194-199.
8. Matteo Maffei; et.al, 2015, “Privacy and Access Control for Outsourced Personal Records”, ISSN: 1081-6011, PP: 341-358.
9. Richard Ssembatya 2015, “Secure and Efficient Mobile Personal Health Data Sharing in Resource Constrained Environments”, PP: 411-416.
10. Xueping Liang et al. 2017, “Integrating blockchain for data sharing and collaboration in mobile healthcare applications”, ISSN: 2166-9589, PP: 1-5.
11. Yang et al. 2017, “Lightweight Sharable and Traceable Secure Mobile Health System”, ISSN: 1545-5971, Volume: PP, Issue: 99, PP: 1-1.