View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Sri Shakthi SIET Journals

RESEARCH ARTICLE

International Journal of Computer Communication and Informatics

DOI: 10.34256/ijcci1915

Class Based Multi Stage Encryption for Efficient Data Security in Cloud Environment Using Profile Data

S.K. Mouleeswaran [a, *], J. Kanya Devi [b], Illayaraja [b]

[a] Associate Professor, Department of Computer Science and Engineering, Dayananda Sagar University, Bangalore, India.
[b] Assistant Professor, Department of Computer Science and Engineering, RVSFOE, RVS Technical Campus, Coimbatore, Tamil Nadu, India.

*Corresponding Author
meetmoulee@gmail.com
(S.K. Mouleeswaran)

Received : 03-04-2019
Accepted : 12-05-2019

ABSTRACT: The security issues in the cloud have been well studied. The data security has much importance in point of data owner. There are number of approaches presented earlier towards performance in data security in cloud. To overcome the issues, a class based multi stage encryption algorithm is presented in this paper. The method classifies the data into number of classes and different encryption scheme is used for different classes in different levels. Similarly, the user has been authenticated for their access and they have been classified into different categories. According to the user profile, the method restricts the access of user and based on the same, the method defines security measures. A system defined encryption methodology is used for encrypting the data. Moreover, the user has been returned with other encryption methods which can be decrypted by the user using their own key provided by the system. The proposed algorithm improves the performance of security and improves the data security.

Keywords: Cloud Environment, User Profile, Data Security, Class Based Approach, Multi Stage Encryption.

Introduction

The growing size of data owned by different organizations requires higher storage capabilities. It requires huge size of data base which claims higher cost. The organizations are not able to spend for the huge data base. To overcome the problem of cost, the cloud environment has become. The cloud system enables the organization to store their information in the cloud based on service level agreements. The users of the organizations can access the cloud data using some authentication schemes. Also, not all the data can be accessed by the users of the organization and it is necessary to restrict them using some protocol.

The class based approach is one which restricts the user according to the class and according to the category of the user. The user can access only the data allowed for them and they can access the data belongs to certain cases. How to restrict them according to the class is a highly challenging issue. By categorizing the data into number of class and by providing access profiles based on the classes it can be enforced. The user profile is an ontology which has users and their class and the class of data which can be accessed. The user can access only the data mentioned in their profile and also it has dedicated encryption decryption keys for each class of data allowed.

The data security is the process of restricting the illegal access from malformed user. The data can be encrypted and stored in the cloud. At the time of request, the system would verify the access of user and provide the data to them. The user can obtain the original data by decrypting the data using the decryption key available and provided to them. There are number of encryption schemes available like public/private key encryption. The issue with this approach is missing of key would be used by a malformed user. Similarly, the attribute based encryption has been used in several articles but introduces higher time complexity.

To overcome the problems of other approaches, an class based multi stage encryption algorithm is presented in this paper. The method encrypts the data according to the key presented for the class and the data has been classified into different stages. Each stage of data class has different key for encryption decryption. The encrypted data has been further encrypted using the system key before storing. The detailed approach is presented in the next section.

Related Works:

The data security protocols already defined is explored in this section. Number of methods has been discussed in this section.

In [1], a hierarchical approach with attribute based encryption scheme has been presented. The HASBE algorithm classifies the data into number of hierarchy and for each class of data attribute, different keys has been enforced. The method produces good results in security but produces higher time complexity.

In [2], a novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control for PHRs, we leverage attribute based encryption (ABE) techniques to encrypt each patient's PHR file. Different from previous works in secure data outsourcing, we focus on the multiple data owner scenario, and divide the users in the PHR system intomultiple security domains that greatly reduce the key management complexity for owners and users.

In [3], the author discusses the security issues affecting cloud computing and propose the use of homomorphic encryption as a panacea for dealing with these serious security concerns vis-à-vis the access to cloud data.

In [4], each peers trust is computed based on the evaluation of the peer it receives in providing service to other peers in the past. This reflects the degree of trust that other peers have in the community. For the evaluation of trust they used five different factors. A feedback a peer obtains from other peers, A feedback scope i.e. number of transactions it has with other peers, the credibility factor for the feedback source, the transaction context factor for discriminating mission critical transactions from less or non-critical ones and the community context factor for addressing community related characteristics and vulnerabilities.

Cloud Security with Virtualized Defense and Reputation-Based Trust Management [5], present a new approach to integrating virtual clusters, security-reinforced data centers, and trusted data accesses guided by reputation systems. A hierarchy of P2P reputation systems is suggested to protect clouds and data centers at the site level and to safeguard the data objects at the file-access level. Different security countermeasures are suggested to protect cloud service models: IaaS, PaaS, and SaaS.

Trusted P2P Transactions with Fuzzy Reputation Aggregation [6], analyzed the eBay auction-based transaction trace data to sort out client behavioral characteristics. Fuzzy Trust, a prototype P2P reputation system is built that helps establish mutual trust among strangers in P2P transaction applications. It uses fuzzy logic inference rules to calculate local trust scores and to aggregate global reputation. This system benefits from the distinct advantages of fuzzy inferences, which can handle imprecise linguistic terms effectively.

In [7], A proactive content poisoning scheme is discussed to stop colluders and pirates from alleged copyright infringements in P2P file sharing. They developed a new peer authorization protocol (PAP) to distinguish pirates from legitimate clients. Detected pirates will receive poisoned chunks in their repeated attempts. Pirates are thus severely penalized with no chance to download successfully in tolerable time. Based on simulation results, we find 99.9 percent prevention rate in Gnutella, KaZaA, and Freenet.

In [8],they proposed a solution for the issue of reconfigurable service modeling and efficient service composition decision making. They introduce a novel compositional decision making process, CDP, which explores optimal solutions of individual component services and uses the knowledge to derive optimal QoS-driven composition solutions. They have developed a case study system to validate the proposed approach to measure the effectiveness of the services.

In [9], the author deal with the issues of reconfigurable service modeling and efficient service composition decision making. We introduce a novel compositional decision making process, CDP, which explores optimal solutions of individual component services and uses the knowledge to derive optimal QoS-driven composition solutions.

In [10], the author proposes a peer-to-peer-based decentralized service discovery approach named Chord4S. Chord4S utilizes the data distribution and lookup capabilities of the popular Chord to distribute and discover services in a decentralized manner. Data availability is further improved by distributing published descriptions of functionally equivalent services to different successor nodes that are organized into virtual segments in the Chord4S circle. Based on the service publication approach, Chord4S supports QoS-aware service discovery. Chord4S also supports service discovery with wildcard(s).

Attribute-based encryption schemes with constant-size cipher texts [11], proposes the first attribute-based encryption (ABE) schemes allowing for truly expressive access structures and with constant cipher text size. Our first result is a cipher text-policy attribute-based encryption (CP-ABE) scheme with O(1)-size cipher texts for threshold access policies and where private keys remain as short as in previous systems.

Attribute-based encryption with fast decryption [12], present the first key-policy ABE system where cipher texts can be decrypted with a constant number of pairings. We show that GPSW cipher texts can be decrypted with only 2 pairings by increasing the private key size by a factor of $|\Gamma|$, where Γ is the set of distinct attributes that appear in the private key. We then present a generalized construction that allows each system user to independently tune various efficiency tradeoffs to their liking on a spectrum where the extremes are GPSW on one end and our very fast scheme on the other. This tuning requires no changes to the public parameters or the encryption algorithm.

In [13], the author presents the first key-policy ABE system where ciphertexts can be decrypted with a constant number of pairings. We show that GPSW ciphertexts can be decrypted with only 2 pairings by increasing the private key size by a factor of |Γ|, where Γ is the set of distinct attributes that appear in the private key. We then present a generalized construction that allows each system user to independently tune various efficiency tradeoffs to their liking on a spectrum where the extremes are GPSW on one end and our very fast scheme on the other.

Fully secure unbounded inner-product and attribute-based encryption [14], present the first inner-product encryption (IPE) schemes that are unbounded in the sense that the public parameters do not impose additional limitations on the predicates and attributes used for encryption and decryption keys. All previous IPE schemes were bounded, or have a bound on the size of predicates and attributes given public parameters fixed at setup. The proposed unbounded IPE schemes are fully (adaptively) secure and fully attribute-hiding in the standard model under a standard assumption, the decisional linear (DLIN) assumption. In our unbounded IPE schemes, the inner-product relation is generalized, where the two vectors of inner-product can be different sizes and it provides a great improvement of efficiency in many applications. We also present the first fully secure unbounded attribute-based encryption (ABE) schemes, and the security is proven under the DLIN assumption in the standard model.

Decentralizing attribute-based encryption [15], propose a Multi-Authority Attribute-Based Encryption (ABE) system. In our system, any party can become an authority and there is no requirement for any global coordination other than the creation of an initial set of common reference parameters.

A party can simply act as an ABE authority by creating a public key and issuing private keys to different users that reflect their attributes.

A user can encrypt data in terms of any Boolean formula over attributes issued from any chosen set of authorities. Finally, our system does not require any central authority. In [16], a decentralized approach of attribute based encryption is presented. Similarly in [17], how the encryption standards has been used for real world applications in detail. In [18], SSDO based cipher text generation has been presented to provide network security.

In [19], the author presents a cost-driven decision maker which considers the cloud client's cost preferences and uses the genetic algorithm to configure a rule-based system to minimize the total auto-scaling cost. The proposed cost-driven decision maker together with a prediction suite makes a predictive auto-scaling system which is up to 25% more accurate than the Amazon auto-scaling system. The proposed auto-scaling system is scoped to the business tier of the cloud services.

In [20], the author presents a new negotiation strategy basket with the aim of approaching the strategies of the real-world negotiation markets. Then modeled the critical condition of the cloud trading market, and adopted a new fuzzy decision system (i.e., Fuzzy Negotiation Strategy Selection System (FNSSDS)) for conducting negotiator agents of type resource providers and resource customers in how to select a suitable negotiation strategy from negotiation_strategy_basket according to the critical condition of the cloud trading market.

Class Based Multi Stage Encryption Using User Profile:

The proposed class based algorithm receives the user request and provides set of keys at the time of registration.

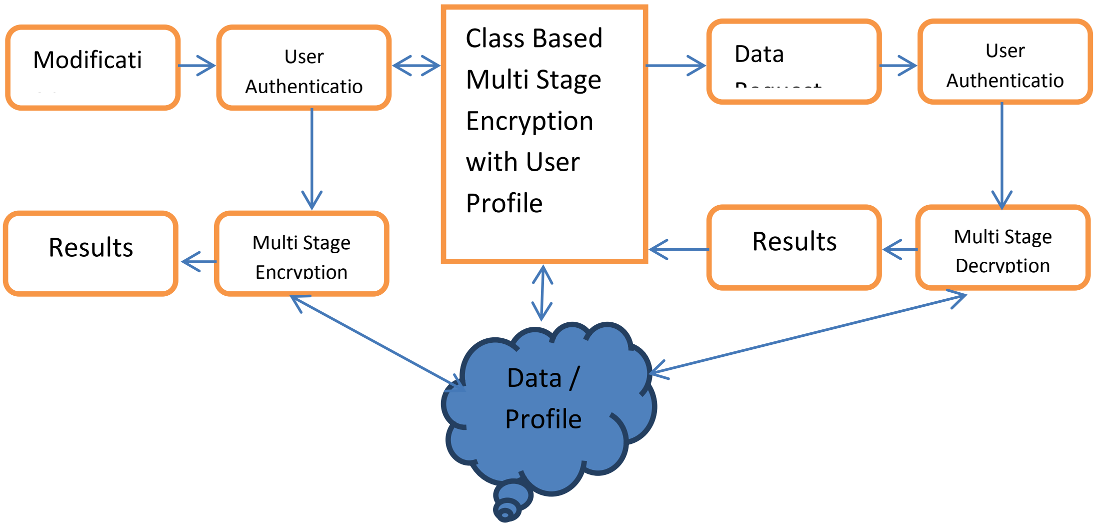

Figure1. Architecture of Proposed Class Based Multi Stage Encryption

The Figure 1 shows the functional architecture of proposed multi stage encryption algorithm and shows various functional components.

Request Handler:

The request generated by the user has been handled by the request handler. The method receives the request and identifies the type of request. If it is a data access request, then the method verifies the authentication key provided with the user profile being maintained. If the user has clears the trust verification, then the data requested has been identified. Second, the data classes are identified and the user has been verified for their possession of access based on profile data. If both gets cleared, the method decrypts the data using the system key and give to the user.

Algorithm:

```
Input: Profile Data Pd, Meta Data Md, Request R
Output: Null.
    Read Pd, Md, R.
    If R.Type == Read then
If$\int_{i-1}^{Pd} Pd(i).User == R.User \&\& Pd(i).Key == R.Key$
then
        Find list of datas requested.
Drl = $\sum_{i=1}^{size(R)} R(Data) \in Md$
        For each data Di
        If $\int_{i=1}^{size(Drl)} Drl(i) \in Md \&\& Drl(i) \in Pd$ then
        Read requested data D.
Perform system decryption Sd (D, systemkey)
        Else
            Return false
 End
        End
 Else
            Return false.
 End
 Else
        If $\int_{i-1}^{Pd} Pd(i).User == R.User \&\& Pd(i).Key ==$
$R.Key$ then
Find list of data given
Drl = $\sum_{i=1}^{size(R)} R(Data) \in Md$
For each data Di
Find the class of data Di
Perform multi stage encryption.
 End
      Encrypt entire data with system key.
Store in cloud.
   End
```

The above discussed algorithm receives the user request and verifies the user for access of modification of any data. If the user clears the trust, then appropriate action has been taken.

Multi Stage Encryption:

The multi stage encryption is performed on the data given. The data given would fall to different category. As the method classifies the data into number of classes and for each class, a separate scheme and key has been used. Using such method and key, the method perform encryption of the data. The encrypted data has been further encrypted with the system key defined. The encrypted information has been indexed to the cloud.

```
Algorithm:
Input: Data D, Meta Data Md
Output: Null
Read D, Md.
Find all the data instances given as Dl = $\sum Instances\ of\ Data\ \in D$
For each instance of data Di
Class c = $\int_{i=1}^{size(Md)} Md(i).type == Di.Type$
Key k = C.EKey
   Di = Encryption (Di,K)
End
Dl = Perform system encryption with system key.
Add to the cloud.
```

The above discussed algorithm presents how the original data has been encrypted in multiple stages to produce higher security.

Multi Stage Decryption:

The cipher data or encrypted data has been given. The encrypted data has been identified for their class. The method identifies the key for each stage of class from the meta data. Then each class data has been decrypted according to the specific key belongs to the class. The decrypted data has been given to the user.

```
Algorithm:
Input: Data D, Meta Data Md
Output: Null
```

```
    Read D, Md.
    Perform decryption with system key.
    Find all the data instances given as Dl = $\sum Instances\ of\ Data\ \in D$
    For each instance of data Di
        Class c = $\int_{i=1}^{size(Md)} Md(i).type == Di.Type$
        Key k = C.EKey
        Di = Decryption (Di,K)
    End
    Dl = Perform system encryption with system key.
    Give to the user.
```

The above discussed algorithm performs decryption of data in multiple stages to produce the original information.

Results and Discussion:

The class based multi stage encryption algorithm for improved data security has been implemented and evaluated for its efficiency. The method has produced efficient results and their performance has been measured in various parameters. The result produced by the method has been presented below.

Table1. Simulation details

Parameter	Value
Tool Used	Advanced Java, Azure
Number of users	500
Number of classes	10
Number of stages	10

The details of simulation being used for the evaluation of proposed algorithm have been presented in Table 1. The method has been validated for their performance in various parameters and the results has been presented below.

The performance on security achieved by the methods has been measured and analyzed. The comparative result has been presented in Figure 2. The result notices that the proposed CBMSE algorithm has produced higher security performance than other methods.

The performance on encryption and decryption has been measured for different methods. The proposed CBMSE algorithm has produced higher performance in both process higher than other methods.

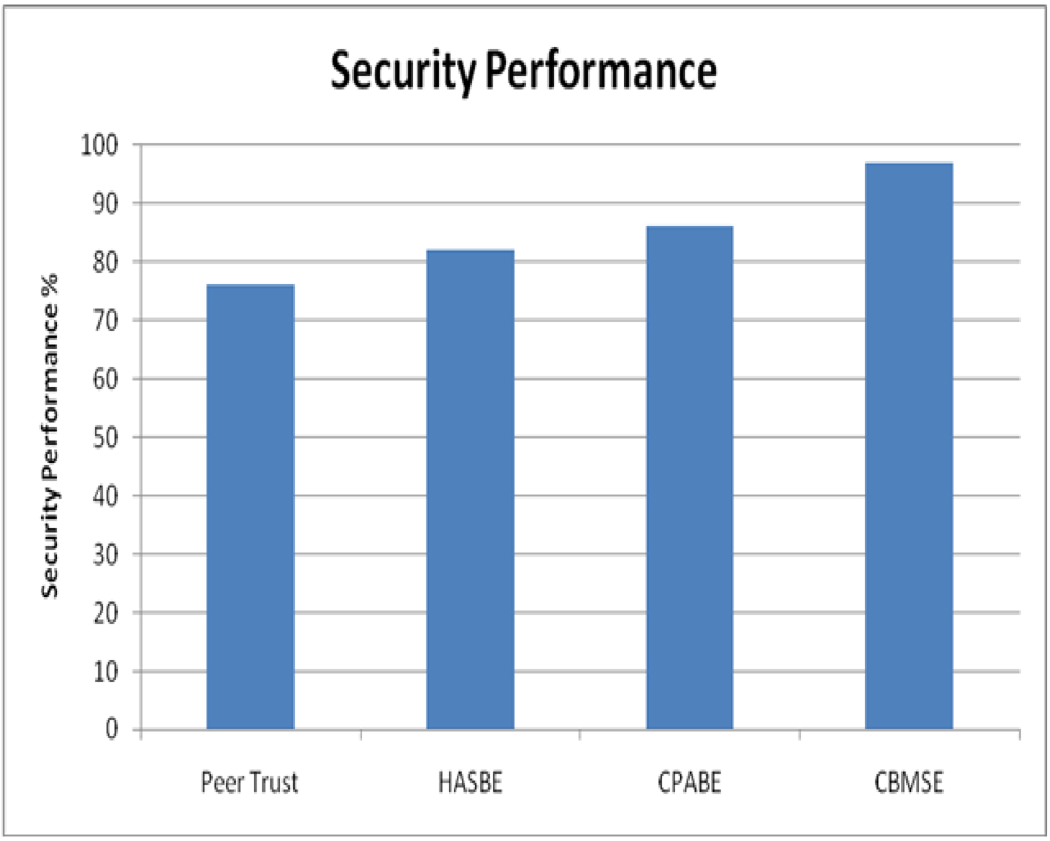

Figure2. Performance analysis on security

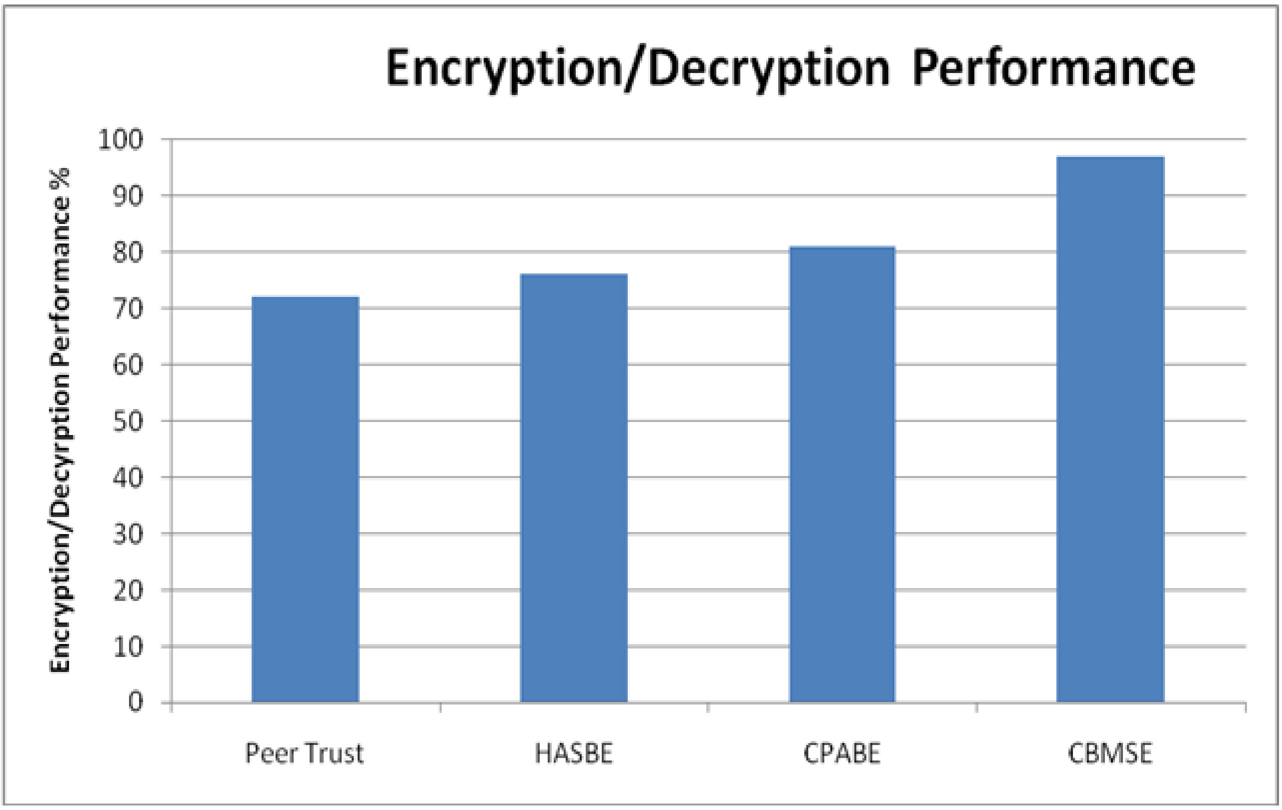

Figure3. Performance analysis on encryption/decryption performance

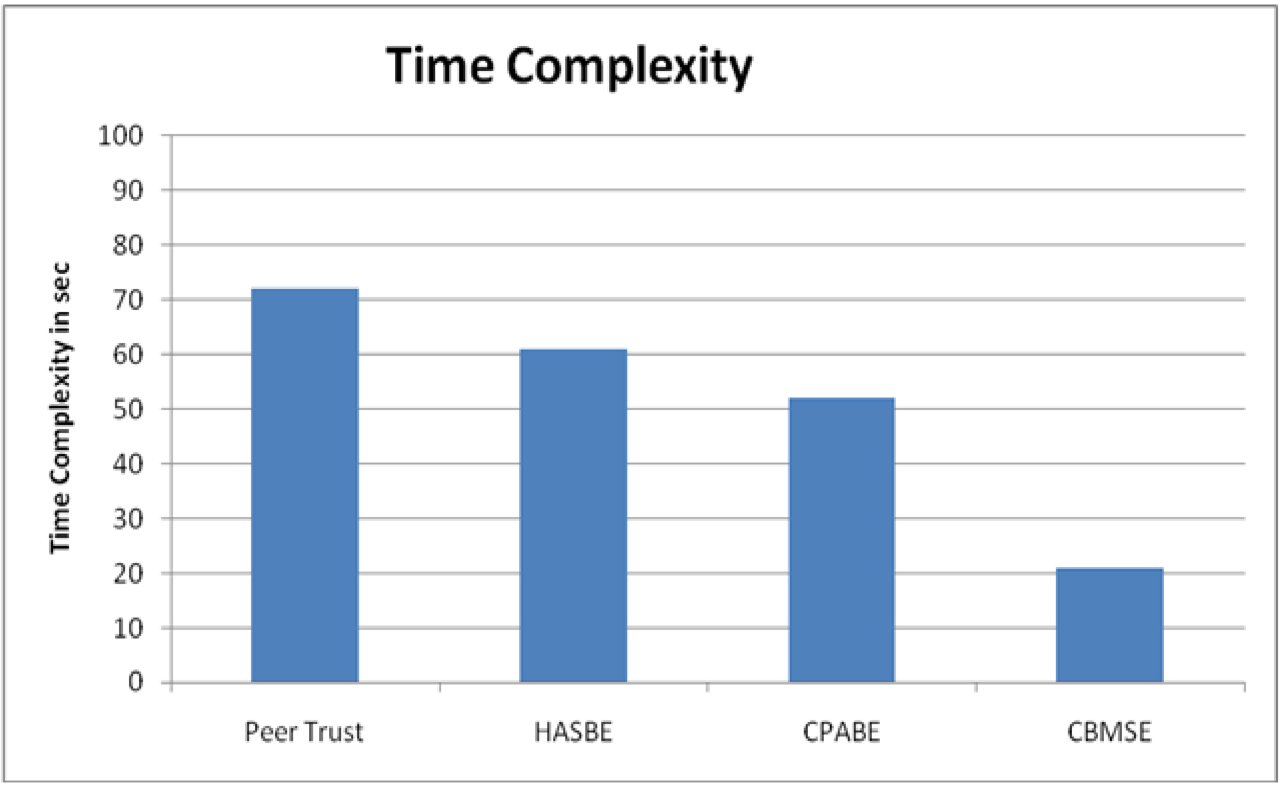

Figure4. Comparison on time complexity

The performance on time complexity produced by different methods has been measured and presented in figure 4. The proposed CBMSE algorithm has produced less time complexity than other methods.

The false classification on authentication produced by different methods has been measured and compared. The comparative result has been presented in Figure 5, which notifies that the proposed CBMSE algorithm has produced less false ratio than other methods.

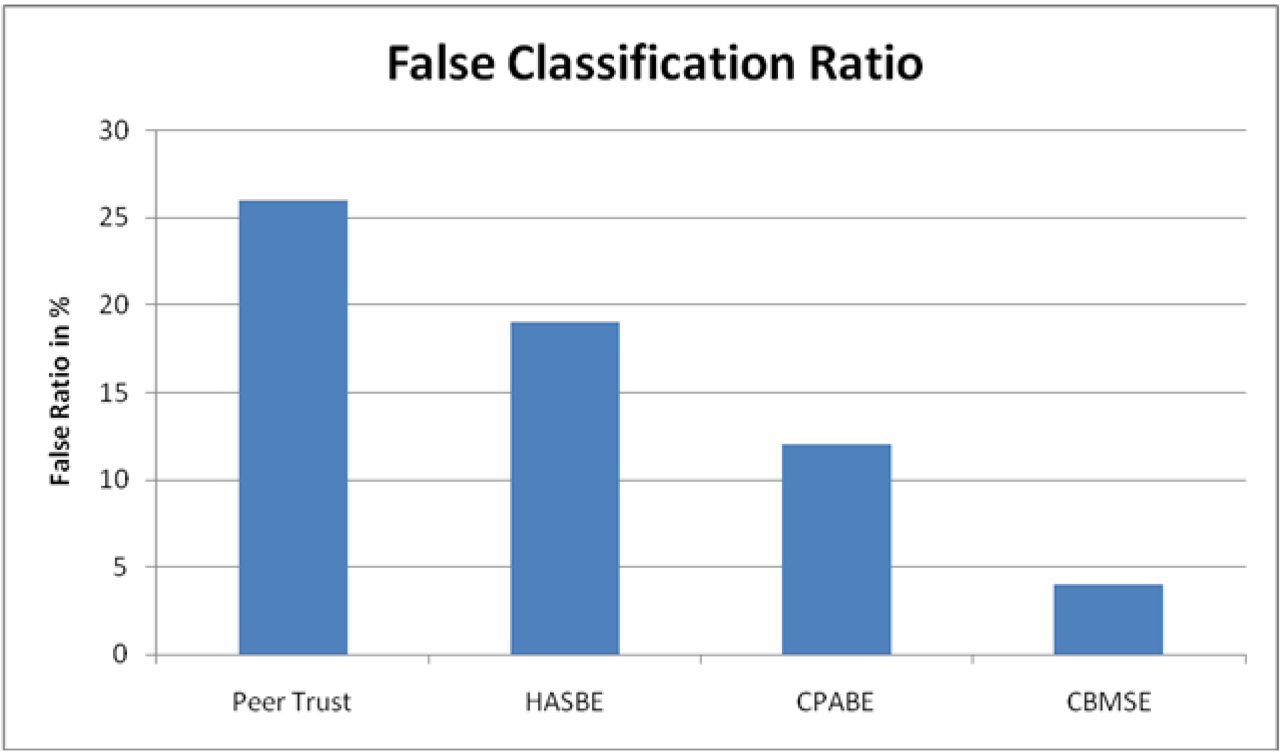

Figure 5.performance analysis in false ratio

Conclusion:

In this paper, an efficient class based multi stage encryption algorithm has been presented. The method classifies the data into number of classes and stages. For each class and for each stage of data, the method maintains different keys for encryption and decryption. The user has been authenticated with the secure key for him and his request has been verified for the possession of access in each stage of the data being requested. Only if the user has access for all the stage of data access he will be given result. Further, the data stored has been carried with multi stage encryption where the data has been initially encrypted based on the class and stage properties. Finally, the entire data has been encrypted with the system key. In the similar fashion, the retrieval or data access request has been performed. On the request phase, the user has been validated for his access in data and their stages. The entire data requested has been first decrypted with the system key and given to the user. The user can decrypt the next level data based on the class or data key he has with the user profile. The proposed algorithm improves the performance of security in cloud and improves overall performance.

References

[1] Zhiguo Wan, HASBE: A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing, IEEE Trans. Infor. Foren. Sec. 2(2012).

[2] Ming Li, Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption, IEEE Trans. Par. Dist. Sys., (2012).

[3] A.Aderemi, Security Issues in Cloud Computing: The Potentials of Homomorphic Encryption, J. Emer. Trends Comp. Infor. Sci. 2 (2011) 10.

[4] L. Xiong and L. Liu, Peer Trust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic communities ,IEEE Trans. Knowledge and Data Eng., (2004) 843-857.

[5] K. Hang, S. Kulkarni, and Y. Hu, Cloud Security with Virtualized Defense and Reputation-Based Trust Management, IEEE Int'l Conf. Depen. Auto. Sec. Com. IEEECS Press, (2009).

[6] S. Song, K. Hwang,R. Zhou, Y.-K. Kwok,Trusted P2P Transactions with Fuzzy Reputation Aggregation, IEEE Inter. Comp., 9 (2005) 24-34.

[7] X.LouandK.Hwang, Collusive Piracy Prevention in P2P Content Delivery Networks, IEEE Trans. Computers, (2009) 970–983.

[8] Alhamad M, SLA based trust model for cloud computing, (2010) 321-324.

[9] Hui Ma, QoS-Driven Service Composition with Reconfigurable Services, IEEE Ser. Comp., 6 (2013) 20-34.

[10] Qiang He, Jun Yan, A Decentralized Service Discovery Approach on Peer-to-Peer Networks, Services Computing, IEEE Transactions, 6(2013) 64-75.

[11] A. Kannagi, K-Partitioned smallest distance Mining Tree for path Optimization in Wireless Sensor Network, J. Wir. Per. Comm. (2018).

[12] A. Kannagi, Smart curiosity sink node prediction mining algorithm for path optimization in wireless sensor network, Inter. J. Eng. Tech., 7 (2018) 180-184.

[13] Y. Jarma, Dynamic Service Contract Enforcement in Service-Oriented Networks, IEEE Trans. Ser. Comp. 6 (2013) 130-142.

[14] N. Attrapadung, J. Herranz, F. Laguillaumie, B. Libert, E. de Panafieu, and C. Ràfols, Attribute-based encryption schemes with constant-size ciphertexts, Theor. Comput. Sci., 422(2012)15–38.

[15] S. Hohenberger and B. Waters, Attribute-based encryption with fast decryption, in Proc. Public Key Cryptography, (2013) 162–179.

[16] S. Chatterjee and A. Menezes, On cryptographic protocols employing asymmetric pairings-The role of revisited, Discrete Appl .Math., 159 (2011) 1311–1322.

[17] T. Okamoto and K. Takashima, Fully secure unbounded inner-product and attribute-based encryption, In Proc. ASIACRYPT, (2012) 349–366.

[18] A. B. Lewko and B. Waters, Decentralizing attribute-based encryption, In Proc. EUROCRYPT, (2011) 568–588.

[19] S. Goldwasser, Y. T. Kalai, R. A. Popa, V. Vaikuntanathan, and N. Zeldovich, Succinct functional encryption and applications: Reusable garbled circuits and beyond, IACR Cryptology ePrint Archive, (2012) 733.

[20] A. Kannagi, M. Muthuraja, Data security Description of enhanced data mining analysis using Symmetric Inference Model, Inter. J. Adv. Infor. Comm. Tech. 1 (2014) 461-465.

About The License

© 2019 The Authors. This work is licensed under a Creative Commons Attribution 4.0 International License which permits unrestricted use, provided the original author and source are credited.