CEASE: CONFIDENTIALITY AND ACCESS CONTROL FOR SECURING PERSONAL HEALTH RECORDS IN THE CLOUD

KrishnaKeerthi Chennam [1], Lakshmi Mudanna [2]

[1] Gitam School Of Technology, Gitam University, CSE Department
[2] Gitam School Of Technology, Gitam University, IT Department

Corresponding author: KrishnaKeerthi Chennam, krishnakeerthich@gmail.com

ABSTRACT: *Cloud storage is one of the most promising services in cloud computing. It offers elastic scaling and low-cost data storage. However, the security issues in the cloud are the main concern that hinders the popularity and application of cloud services. The most important issues related to data storage in the cloud are data confidentiality, authentication, and regulations on data access. A straightforward solution to protect the data confidentiality is to encrypt the data before outsourcing to the untrusted cloud server. A malicious administrator possibly creates an account as a legitimate user and compromises the security of encrypted database in numerous ways. An access control is essential for categorizing the data based on the sensitivity level of the health records. This work proposes the cryptographically Enforced Access control for Securing Electronic medical records in the cloud (CEASE). The CEASE includes three components to ensure the confidentiality of medical data. Initially, it exploits the trusted proxy server and applies the Advanced Encryption Standard (AES) on the health data before uploading it to the cloud server. Secondly, the proxy server applies access control policy on health data in the cloud using a set of attributes which are offered during user registration. The proxy server involves in processing encrypted queries to read the encrypted data from the cloud and also decrypts the data using the attributes before delivering the data to an end user. Finally, it introduces the partial shuffling within a restricted data block that contains the hot health records and thus, it ensures the data access pattern confidentiality without degrading the querying speed. The performance of CEASE technique is evaluated in the Java platform, and the results show that the CEASE significantly protects the confidentiality of critical data in the cloud platform.*

KEYWORDS: *Cloud Service, Malicious Activities, Encryption, Shuffling, Access Control Policy.*

1. INTRODUCTION

The cloud server offers services for data owners to host their data in the cloud. Due to the untrusted third party, the access control, and confidentiality of critical data becomes challenging issues in a cloud environment [Beh11]. Many potential data owners are reluctant to outsource the confidential data to untrusted cloud storage providers due to the security issues. To provide sufficient protection to the data against the malicious activities, the data owners store their crucial data in an encrypted format. However, the encryption alone does not provide data confidentiality protection in the cloud storage. There are several restrictions and rules need to be followed by the users before accessing an encrypted data from the cloud-based servers. The access control refers to a policy that authenticates a user and permits the authorized user to access the data from the cloud database [LW12]. Even though the data access control policy on the encrypted database avoids illegal access to the database, it does not offer a complete solution to data confidentiality. There is a chance for the malicious administrator to create an account as a legitimate user and compromise the security of cloud server in numerous ways. In such cases, the encrypted database addresses two threats [Rya11, Rya134].

Primarily, an unauthorized user attempts to learn private data of others. For instance a doctor snooping on accessing the patient health records. The data access control policy prevents the malicious user to learn private data. In the case of the malicious database administrator, the encryption is unable to provide any guarantees for users logged into the application during an attack. Moreover, the malicious database server attempts to derive the sensitive data even when the critical data are encrypted. By simply observing the database access patterns, the malicious administrator is likely to delete the frequently accessed data from the cloud database. Therefore, providing the confidentiality for data access patterns is essential. A common solution for achieving the data access pattern confidentiality is shuffling [C+13]. If the database contains few rows, the shuffling time of the database is very fast. However, in the case of extensive data that contain millions of rows, the shuffling process consumes more time. For this reason creating the partial shuffling in the restricted area of the database allows the system to maintain the access pattern confidentiality as well as fast querying.

1.1 CONTRIBUTIONS

This work proposes a Cryptographically Enforced Access control for Securing Electronic health records in the cloud (CEASE) to offer the data confidentiality and access control of outsourced data to the cloud server against security threats.

- The CEASE provides practical and provable confidentiality in the face of curious/malicious cloud administrator by applying the advanced encryption standard for critical data and outsourcing the encrypted data to the cloud server.
- By providing an authority responsible for attribute management and secure key distribution, the attribute authority enables the proxy server to apply the data access control policies on the encrypted database.
- By executing encrypted queries over encrypted data on the cloud server, the CEASE averts the disclosure of the critical data to the malicious user or administrator.
- Introducing the single level data block index and applying partial shuffling on repeatedly accessed data in a restricted area; CEASE attains data access pattern confidentiality without degrading the querying process.

2. RELATED WORKS

The conventional public key encryption based schemes [B+09, CRD11] require encrypting of multiple copies of data using different keys, resulting in high key management overhead. In contrast, the Attribute-Based Encryption (ABE) schemes encrypt the data using a set of attributes. It potentially improves encryption and key management process [G+06, LLR10]. Numerous works which have used ABE on outsourced encrypted data for providing security are [L+10b, CS12, BGK08, I+09, I+10a]. Besides, there has been an increasing interest in applying ABE for providing secure data access of electronic healthcare records. Recently, an attribute-based infrastructure, where the patient data are encrypted using a variant of Cypertext Policy (CP)-ABE [I+10a, NGS10, BAW07] and bilinear pairing [L+15] are being used. There are two types of CP-ABE systems such as single-authority CP-ABE [OSW07, Wat11, G+08, L+10a] and multi-authority CP-ABE. In the single authority CP-ABE, all attributes are managed by a single authority, whereas in multi-authority CP-ABE, all the attributes are from different domains and administered by various authorities.

However, the length of the ciphertext is linear when increasing the number of users. The delegation of access rights is proposed in [B+11]. In [A+11], the generated self-protecting health records are stored on cloud servers and allowing the users to access the data in a secure manner using ABE even when the health provider is offline. The access policies are derived according to the attributes which are provided during user registration [L+13]. In which, the patient can selectively share their details among a set of users by encrypting the data under a set of attributes. However, based on the number of attributes, the complexity of encryption, key generation, and decryption is fixed. To integrate ABE into a large-scale system, the key management scalability and efficient access control policy are non-trivial to solve in a cloud server. The key policy ABE distributes the keys to users with the aim of securing the outsourced data in the cloud [Y+10b]. As it aggregates the key update operations over time, it achieves less overhead.

An encryption scheme in [Y+11] integrates an access control mechanism to ensure the confidentiality of data stored in cloud databases. In [FCM13], the patient is allowed to encrypt all stored and transmitted data and define the standard database access control policies. It extracts the data by executing the query operations on encrypted data stored in a cloud provider. Another important issue in cloud service is data access pattern confidentiality. To enforce the data confidentiality, the encrypted B-tree is proposed in [C+11]. Shuffling the data after each database query [C+14, Y+11] avoids inference attacks, but increases the computational cost. However, there is no specific approach proposed to provide the ability to insert, update or delete data using shuffled B-trees. To prevent this kind of security issues, a database system needs to be effectively encrypted to achieve the confidentiality.

2.1 Problem Statement

Outsourcing data into the cloud has become the standard for storing, as it is cost-effective and being large-scale data storage. Despite the numerous benefits, the third party cloud servers are likely to be untrusted and raise security and privacy obstacles to the cloud adoption. There remain two problems in providing data confidentiality, such as curiosity or malicious behavior of administrators. Due to the curiosity of administrators, they view the stored data of legitimate users resulting in data confidentiality loss. Data confidentiality is possibly protected by applying an encryption layer wrapping data before outsourcing them to external cloud providers. Having an encrypted database ensures confidentiality in its storage. In contrast, malicious data administrators can alter the queries of deleting, inserting, copying or swapping data in the database and decrease the query integrity. Also, as the compromised administrator is capable of creating an account as a legitimate user,

this compromises the security of encrypted DB in numerous ways.

3. OVERVIEW OF THE PROPOSED METHODOLOGY

Storing and processing sensitive data on cloud service increases the risk of unauthorized access to individual/hot health data when the third party is curious/malicious. This work proposes the CEASE to provide the confidentiality of personal/hot medical records. The proposed work includes three components to ensure the cloud data security: Applying Encryption on Sensitive Patient Health Records, Secured Data Retrieval via Data Access Control scheme and query encryption, and Confidentiality Protection of Hot Health Data via Optimized Shuffling.

Firstly, the data owner enables the trusted proxy server to apply the Advanced Encryption Standard (AES) on the health data before uploading it to the cloud server. Secondly, the proxy server which is responsible for attribute management identifies the user using a set of attributes and applies access control policy on health data in the cloud. Processing encrypted queries enables to read the encrypted data from the cloud and to decrypt the data using attributes in the proxy server before delivering the data to an end user. Even though, querying encrypted data in a controlled manner is an efficient way of ensuring the confidentiality of individual patient record in the cloud, inferring hot health records are still possible by observing the encrypted query executions. Thirdly, the CEASE technique introduces the partial shuffling within a restricted data block that contains the hot health records and ensures the hot data confidentiality as well as fast querying. Thus, the proposed CEASE algorithm ensures that the curious/malicious administrator on the cloud cannot get or modify any (hot) data either from the stored sensitive health records or encrypted query executions and also ensure faster querying process.

3.1 Security Model

Consider a Medical Organization (M) contains a set of users U with different classes such as Patient (U_P), Doctor (U_D), and Researcher (U_R), where U = {U_P, U_D, and U_R}. Fig. 1. show how the proposed CEASE system generates User id (U-ID) and identity token for each user based on their class. It applies data access control policy according to a class of U in M, wherein U_P can access their data, and others can access the data of their patients in a restricted manner. The design of the CEASE system depends on the six types of entities such as a Certificate Authority (CA), Attribute Authorities (AAs), Data Owner (owner), Proxy Server (PS), the Cloud Server (server) and Data Consumers (users). These entities contact with one another, either in direct or indirect ways to perform different tasks in the cloud-based CEASE system. The owner who holds the PEHRs categorizes the user and defines access policies over Attributes (Att). During the user registration process, trusted CA generates Att ∀ data.

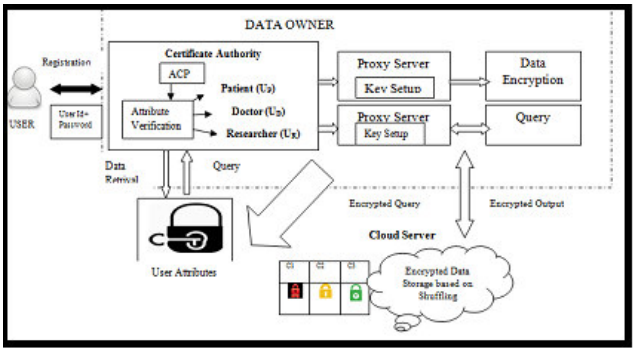

Fig. 1. Block Diagram of the Proposed Methodology

The AA is responsible for entitling and revoking Att, and PS is in charge of allowing the user to access the data according to their class. A trusted PS encrypts the data content before storing it to the cloud server and encrypts the user query using Att to avoid the data leakage due to curious as well as a malicious cloud server. The data ($d^{Att}_{U\text{-}ID}$) is stored in the database H in which all the attribute values of each user are alphabetically sorted based on the unique attribute (search key), for instance, encrypted U-ID is a search key. An index appears on every search key value of any frequently accessed d^{Att} of a user, and it is shuffled within its corresponding block. It ensures the data confidentiality as well as fast querying.

3.2 Sensitive Patient Health Records on Cloud

The Patient Electronic Health Records (PEHRs) are the personal data and ensuring its security is inevitable. Moving PEHRs to a cloud server and their management via the cloud ease the way of health and medical data access for everyone and everywhere in the world. In Cloud computing, there are many security threats and behaviors such as illegal data access and internal staff malicious behavior. The ownership controlled encryption mechanism is the best way to attain Secure PEHR storage in cloud service. The key idea of CEASE system is twofold. Firstly, in user registration, the users are classified according to their attributes, such as a patient, doctor, and researcher. Secondly, owners encrypt the PEHRs of the user under a selected set of attributes, and it allows the users who have a proper set of attributes to read the encrypted data. Thus, the proposed CEASE technique enables the encrypted data storage and attempts to provide secure access of PEHRs in the

cloud against unauthorized access or change in healthcare critical data content and user agreements.

3.2.1 User Registration and Encrypted Data Storage on Cloud

The PEHR user creates the PEHRs data. When a user accesses the system, the CA connects the user to the data owner and generates an account in terms of username, identity token, and Certificate (U-ID) which is used in the future to identify the user. These credential information authenticate the user to the server and allow them to access the data from the cloud. Firstly, the CA generates User ID (U-ID), which is used to identify the user and secure communication with the cloud in the system. The patient can access their data, and others (doctor and researcher) can access the data of patients according to the access control policy.

Consider a system contains the K number of users and N number of attributes. The attribute authority derives a set of attribute ids and keys (equal to the number of Columns {C1, C2, ... Cn} containing the basic information of patient). Then the attribute authority uploads the generated attribute ids {AA_{id1}, AA_{id2}, ... AA_{idn}} and secret key {K_{id1},K_{id2}, ... K_{idn}} to the proxy server via SSL channel. During patient registration, the patient offers some basic information such as User-ID, Disease, Treatment, Age, and Contact Details to upload to a proxy server via data owner. The proxy server encrypts the health records of the patient using a symmetric encryption method, in which both the encryption of plaintext and decryption of ciphertext exploits same secret key (K_{Att} = n1,..., n–k). Finally, it sends the encrypted information and encrypted index to the cloud server (via SSL). The cloud server stores each encrypted information. Fig. 2. shows the process of the CEASE system during patient registration in medical organizations.

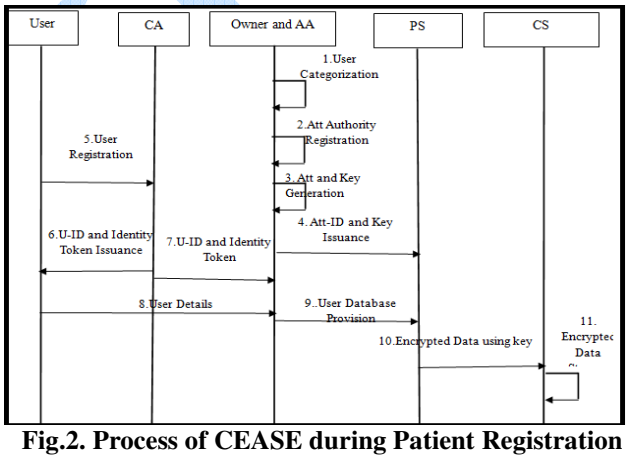

Fig.2. Process of CEASE during Patient Registration

The efficient way to solve the issues of attack on sensitive data is to execute an effective encryption technique on health records before storing the data in a cloud server. Thus, the CEASE technique applies the ownership controlled encryption on PEHRs, before uploading it to cloud service. The best symmetric encryption technique used for securing the sensitive data while outsourcing to the cloud is the AES since it is iterative rather than data encryption standard. Before encrypting the patient information, the CEASE combines the K_{Att} with a plaintext data to generate an encrypted data for each attribute of U-ID. If patients are registered in the application, the owner collects their personal data and divides the data into several columns (C_1, ..., C_n), for example, the personal data consists of User-ID, Disease, Treatment, Age, and Contact Details. Every Column values of each U-ID, the CEASE system executes the equation (1) for K-1 times. To make each data unique, one fixed-length group of bits called as Initialization vector (IV) is assigned as CT_0. In subsequent iterations the output of the previous iteration, CT_{k-1} is taken as input to encrypt the data. It ensures distinct Cipher Text (CT) generation when encrypting the same plain text for K-1 iterations even with the same K_{Att}. Thus, the CEASE technique enables the proxy server effectively to encrypt the sensitive patient data multiple iterations using AES algorithm and upload the encrypted data to the cloud server.

$$CT_k = K_{Att}\{Data \oplus CT_{k-1}\} \quad (1)$$

where K varies from 1 to K-1.

3.3 CEASE Secure Data Retrieval

There remain two issues in CEASE data retrieval technique. Firstly, there is a possibility to an adversary in the cloud to learn the sensitive data, when that person is also a patient in a hospital. Secondly, even though the encrypted patient data are securely transferred and stored in the cloud server, it is vulnerable to the internal staff malicious behavior of cloud server. Even if the information is accessed by the malicious user or cloud administrator, none of the data is extracted from the personal health information of patients without the secret key. Thus, the CEASE technique needs to encrypt the column names and user id details in user query and retrieve the encrypted data securely. Moreover, it is essential to apply the access control policy on data retrieval.

3.3.1 Access Control Policy Enabled Data Retrieval

The CEASE system executes the authentication scheme to allow the legitimate user access and denies the unauthorized access using the registration process

with different attributes. Not only the unauthorized users, but the authenticated users also attempt to access the data of others due to their curiosity. However, the data owner needs to deny the authenticated user access the data of others. For example, a registered doctor in a hospital environment is likely to try illegally to access the data of a patient with global identity, such as U-ID by logging into the system using own credential. It is essential to apply the access control policy based on user class and strengthen the legitimate users when using the application because curious or malicious operation possibly leads to revealing the confidential record to other illegal parties.

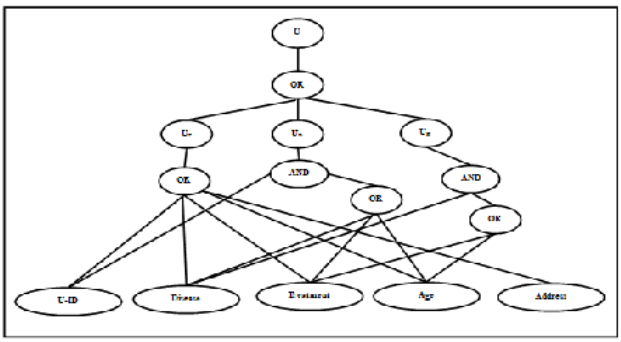

Fig. 3. Access Control Policy Tree

Before encrypting the plain data multiple times, it is essential to authenticate the U-ID to the AA and entitle the attributes for each U-ID according to their class. The U-ID submits Certificate (U-ID) to AA to authenticate the user. If it is a legitimate user, the AA entitles a set of attributes $\{A_{tt}\}$ to the user U-ID and assigns K_{att} according to their class. Initially, the CEASE system defines an access policy in a tree structure for each user class, as shown in fig. 3. The root node is a user of the hospital, which has different types of users such as a patient, doctor, and researcher. The OR indicates that the patient U_P can access any of their data individually. The AND operation shows that the doctor U_D can access the attributes of disease, treatment, and age only along with the access of U-ID, and the researcher can access the data of treatment and age, only along with the disease attribute. Thus, the CEASE technique applies the access control policy along the query encryption during data retrieval.

3.3.2 Query Encryption Enforced Secure Data Retrieval in Cloud

All the legitimate users in a hospital can query the encrypted patient data according to the access control policy. However, the curious or malicious administrator at a hosting can snoop the sensitive data of any user who is logged in the application, when revealing the decryption key to the cloud server. To minimize the amount of secret data leaked to the cloud server when a user is logged in the application, the CEASE exploits encrypted queries over cloud database. When a user queries the application, the proxy server intercepts all the SQL queries and sends an encrypted query to execute on encrypted data in the cloud. Moreover, the CEASE technique never reveals the decryption key to the cloud server, and thus, it ensures that a curious administrator cannot gain access to private information.

/* Decryption Algorithm */
Input: A CT block $\{CT_1, ..., CT_k\}$ and keys $\{K_{att1}, ..., K_{attk}\}$
Output: Plain-Text
1: **Assign** k=1 and i=|CT|
2. **If** k≤|CT| **do**
3. **for** each k **do**
4: **execute** equation (2)
5: **Assign** k=k+1 and i=|CT|-1
6: **else**
Plain-Text = PT_k

Algorithm 1: Decryption Algorithm

Consider, a user sends a query, SELECT Disease FROM Table-Name, WHERE (User-Id = "125") to the proxy server via data owner. The proxy server exploits the same encryption keys used for encrypting the column names such as disease and user-id and rewrites the query as SELECT XYZ FROM abc1234, WHERE (uvw= "wxy123"). The cloud server determines the column, which matches with the encrypted query terms, but not the actual content of column name and value. Then it sends the matched encrypted data to the proxy server. Finally, the proxy server applies the decryption algorithm with the same symmetric key to obtain the Plain Text (PT) of received encrypted data.

$$PT_k = K_{Att}\{CT_i \oplus CT_{i-1}\} \quad (2)$$

where K varies from k to 1.

3.4 Data Confidentiality Protection of Hot Health Data via Optimized Shuffling

In the cloud, a malicious administrator can derive sensitive information about user queries, by observing the access patterns, e.g. the records retrieved frequently called as hot records. Such a threat aggravates the encrypted data storage model whereby a data owner outsources the encrypted data to an untrusted cloud service provider. The data encryption does not entirely solve the confidentiality

problem because access patterns leak the frequency of data access with the encrypted column and row index, the hot health records are likely to be deleted by the malicious administrators. The conventional techniques apply full or partial shuffling of the database to avoid this. However, due to shuffling, they do not support the optimal query search on databases. It leads to scanning the entire database for querying, resulting in high time complexity. Thus, the CEASE technique exploits the partial shuffling within a restricted data block using indexing.

3.4.1 Single Level Data Block Index and Partial Shuffling

Indexing significantly reduces the response time of querying the database. For every query, the server looks in all the rows from the beginning of the table to the end of it for determining the matched data. The query processing time linearly increases with database size. For this reason, creating an index for the database allows the server to get the result without searching the whole data. However, indexing is a complex process, when the data are shuffled periodically. Thus, the CEASE technique introduces the single level data block index on partially shuffled database records.

Table 1. Alphabetically Sorted Database Records with single level data block index

Block	First Key
A	AA452
B	BHU8750
.	.
X	XDRG895

S.No	C_1	C_2	....	C_n
1	AA452	FRDG45	.	TYUHNHG6
2	ABC589	HYTGR6	.	RTUYI74
3	AGTY678	GTY90	.	RET567
4	BHU8750	FRTYB	.	AWERT56
.	.	.	.	.
$\|U_p\|$	XDRG895	DRT567	.	ACRET89

Initially, the records in the database are alphabetically sorted based on the encrypted U-ID value. When including a new row into the database, it is inserted into a specific position to retain the sorting. Table 1 demonstrates the single level data block index on the database and the C_1 act as a search key. Based on the search key, the records are sorted. Database records that appear for every search key in the database are named as a block. For example, first three records start with the search key of A and these records are considered as a single block. The table 1 shows that the database contains a pointer to the first record that starts with the particular search key value. Initially, all the records in a database are labeled as white. When a record is frequently fetched, it is labeled as black. The full database shuffle is not indispensable, and the white record does not leak any information since there is no access pattern involving it. However, black records shuffling over full database lead to index the records after every shuffling. The CEASE applies partial shuffling within its corresponding data block. Thus, the proposed CEASE technique ensures data confidentiality as well as indexing based fast querying.

4. Experimental Evaluation

The CEASE technique is implemented using Java 1.8 to evaluate the performance of the proposed protocol. The CEASE is implemented in three steps such as encrypted database creation, access control policy, and data shuffling. The performance of CEASE technique is compared with the Encryption scheme Integrated with an Access Control (EIAC) [FCM13] mechanism.

4.1 Evaluation Settings

The user and the server side experiments are conducted on an Intel® Pentium(R) Dual CPU G2030 @ 3.00 GHz × 2 with 4GB of RAM. The database contains 75000 records**.** The AAs generates secret key $K_{id(j)}$ for each attribute j with the size of 128-bit long. The records are encrypted by the PS using advanced encryption standard mechanism. By dividing the encrypted records into blocks using alphabetical letters and numbers, the records are arranged. When a record is accessed more than one time, the record is shuffled in its corresponding block. A set of queries such as SELECT, UPDATE, and INSERT are created to evaluate the performance of CEASE technique.

4.2 Evaluation Results

To evaluate the data confidentiality, the database size varied from 500 ... 75 000 records.

Querying Cost:
The time is taken by the query to retrieve the result from the encrypted database.

Storage Overhead:
The memory used by the server to store the secret key values and user credential information.

Hot Data Confidentiality:
The percentage of protecting the confidentiality of critical data and data access pattern.

a. Querying Cost:
The percentage of protecting the confidentiality of critical data and data access pattern.

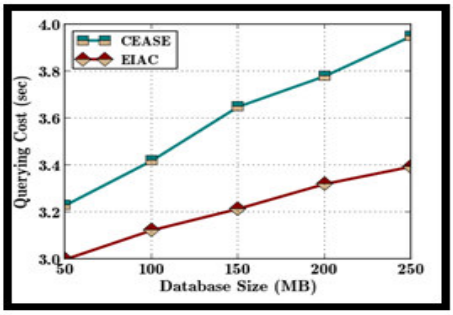

Fig. 4. Database Size vs. Querying Cost

Fig 4 shows the comparative results of Querying cost between CEASE and EIAC mechanisms. The querying cost includes the time of query encryption, data retrieval time, and data decryption time. As the figure 4 illustrates, the querying cost increases linearly with the size of database. As the CEASE implements the query encryption, the querying cost of CEASE increases more than that of EIAC. Even when the querying cost increases with the database size, still CEASE achieves the data confidentiality significantly. For instance, the querying cost of CEASE is 3.25 Sec when the size of the database is 50 MB. However, it increases to 3.95 Sec when the database size reaches 250 MB. There is a chance for increasing the querying cost dramatically due to the encryption of every query in CEASE, the shuffling over restricted database block effectively achieves the data confidentiality, also considerably reduces the data retrieval time from the database. It results in a slight increment of overall querying cost even when the database size is 250 MB.

b. Storage Overhead:

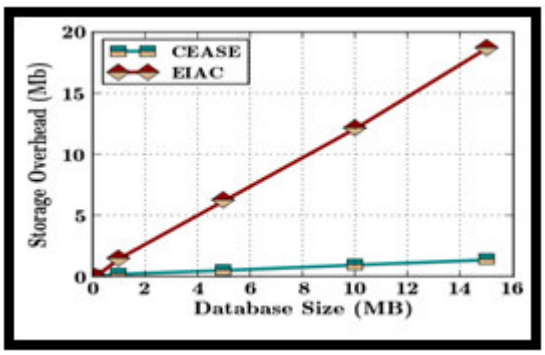

Fig. 5. Number of Attributes vs. Storage overhead

The storage overhead on each user in the PS of CEASE scheme is equal to the size of the secret key issued by the AAs to each attribute and credential information. The results of storage overhead for both CEASE and EIAC are compared in fig. 5. It is observed that the storage overhead of CEASE scheme has increased moderately with the database size. The CEASE technique attains low storage overhead and better data confidentiality by maintaining the secret keys only for each attribute disregarding the user. In contrast, the EIAC mechanism generates a secret key value for each attribute and each user separately, and thus, the storage overhead gets increased dramatically with the database size. For instance, the storage overhead of CEASE is 0.08MB with 0.1 MB database size, but it increases to 0.25MB when the database size is 1Mb. By sharing the attribute keys between different categories of users, the CEASE implements the access control policies without incurring additional storage overhead.

c. Hot Data Confidentiality:

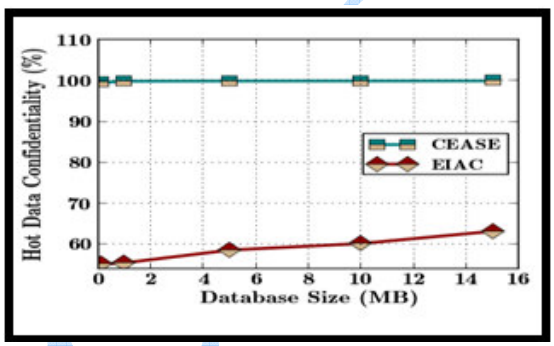

Fig. 6. Database Size vs. Hot Data Confidentiality

The results of hot data confidentiality of CEASE and EIAC mechanisms are shown in the fig 6. The hot data confidentiality of EIAC has improved moderately with the database size. Because, the adversaries identify the hot or repeatedly accessed data of field in the database, by observing the user queries continuously. However, in CEASE technique, the queries are encrypted, and the hot records in the database are shuffled within a restricted area. Thus, the hot data confidentiality of CEASE technique has always been better with any size of the database. With small database size, the attack on hot data confidentiality is possible in the EIAC. However, in CEASE, the hot data shuffling in a single block protects the hot data confidentiality successfully. For instance, the hot data confidentiality is 99.5% with small database size, whereas, the EIAC attains only 55% of hot data confidentiality.

5. CONCLUSIONS

This work proposed a Cryptographically Enforced Access control for Securing Electronic health records (CEASE) that provide data confidentiality in the cloud. The data owner encrypts the health data using advanced encryption standard and uploads the encrypted data to the cloud server. Thus, it ensures the data confidentiality in a cloud server. The CEASE exploits the proxy server who is responsible for attribute management and applies access control policy on encrypted health data using a set of

attributes. By enabling the encrypted queries on the encrypted data from the cloud, the CEASE delivers the data to an end user without disclosing the critical data to the malicious user or administrator. The CEASE technique introduces the partial shuffling within a restricted data block and avoids the malicious administrator to infer the hot health records which are repeatedly accessed. Finally, the CEASE technique is evaluated and compared with the EIAC mechanism in terms of Querying cost, storage overhead, and Hot Data Confidentiality.

REFERENCES

[A+11] **J. A. Akinyele, M. W. Pagano, M. D. Green, C. U. Lehmann, Z. N. J. Peterson, A. D. Rubin** - *Securing electronic medical records using attribute-based encryption on mobile devices*, In Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices, pp. 75-86, 2011.

[Beh11] **A. Behl** - *Emerging Security Challenges in Cloud Computing: An insight to cloud security challenges and their mitigation*, World congress on Information and Communication Technologies, PP. 217-222, 2011.

[BGK08] **A. Boldyreva, V. Goyal, V. Kumar** - *Identity-based encryption with efficient revocation*, In Proceedings of the 15th ACM conference on Computer and communications security, pp. (417-426), 2008.

[BAW07] **John Bethencourt, Sahai Amit, Brent Waters** - *Ciphertext-policy attribute-based encryption*." IEEE Symposium on Security and Privacy, pp. 321-334, 2007.

[B+09] **J. Benaloh, M. Chase, E. Horvitz, K. Lauter** - *Patient controlled encryption: ensuring privacy of electronic medical records*, In Proceedings of the ACM workshop on Cloud computing security, pp. 103–114, 2009.

[B+11] **M. Barua, X. Liang, R. Lu, X. Shen** - *PEACE: An efficient and secure patient-centric access control scheme for eHealth care system*, IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. (970-975), 2011.

[CS12] **A. Clarke, R. Steele** - *Secure and reliable distributed health records: Achieving query assurance across repositories of encrypted health data*, IEEE 45th Hawaii International Conference on System Science (HICSS), pp. 3021-3029, 2012.

[CRD11] **Dong Changyu, Giovanni Russello, Naranker Dulay** - *Shared and searchable encrypted data for untrusted servers*. Journal of Computer Security, Vol.19, No. 3, pp. 367-397, 2011.

[CS+11] **Sabrina De Capitani di Vimercati, Sara Foresti, Stefano Paraboschi, Gerardo Pelosi, Pierangela Samarati** - *Efficient and private access to outsourced data*, 31st IEEE International Conference on Distributed Computing Systems (ICDCS), pp. 710-719, 2011.

[C+13] **S. De Capitani di Vimercati, S. Foresti, S. Paraboschi, G. Pelosi, P. Samarati** - *Distributed shuffling for preserving access confidentiality*, In Proc. of the European Symp. on Research in Computer Security (ESORICS), Vol. 8134, pp.(628-645), 2013.

[C+14] **S. De Capitani di Vimercati, S. Foresti, S. Paraboschi, G. Pelosi, P. Samarati** - *Protecting access confidentiality with data distribution and swapping*, IEEE Fourth International Conference on Big Data and Cloud Computing (BdCloud), pp. 167-174, 2014.

[FCM13] **L. Ferretti, M. Colajanni, M. Marchetti** - *Access control enforcement on query-aware encrypted cloud databases*, IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom), Vol. 2, pp. 219-219, 2013.

[G+06] **V. Goyal, O. Pandey, A. Sahai, B. Waters** - *Attribute-based encryption for fine-grained access control of encrypted data*, In Proceedings of the 13th ACM conference on Computer and communications security, pp. (89-98), 2006.

[G+08] **V. Goyal, A. Jain, O. Pandey, A. Sahai** - *Bounded ciphertext policy attribute based encryption*, In Automata, languages and programming, Springer, pp. (579-591), 2008.

[I+09] **L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, W. Jonker** - *Ciphertext-policy attribute-based threshold decryption with flexible delegation and revocation of user attributes*, 2009.

[LW12] **A. B. Lewko, B. Waters** - *New Proof Methods for Attribute-Based Encryption: Achieving Full Security through Selective Techniques*, in Proc. 32nd Ann. Int'l Cryptology Conf.: Advances in Cryptology - CRYPTO'12, Vol. 7417, pp. 180-198, 2012.

[LLR10] **M. Li, W. Lou, K. Ren** - *Data security and privacy in wireless body area networks*", IEEE Wireless Communications, Vol. 17, No.1, pp.51-58, 2010.

[L+10a] **Allison Lewko et all.** - *Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption*. Advances in Cryptology–EUROCRYPT, Springer Berlin Heidelberg, pp.62-91, 2010.

[L+10b] **M. Li, S. Yu, K. Ren, W. Lou** - *Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings*, In Security and Privacy in Communication Networks, Springer, pp. (89-106), 2010.

[L+13] **M. Li, S. Yu, Y. Zheng, K. Ren, W. Lou** - *Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption*", IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 1, pp.131-143, 2013.

[L+15] **C. H. Liu, F. Q. Lin, C. S. Chen, T. S. Chen** - *Design of secure access control scheme for personal health record-based cloud healthcare service*, Security and Communication Networks, Vol.8, No.7, pp.1332-1346, 2015.

[NGS10] **S. Narayan, M. Gagné, R. Safavi-Naini** - *Privacy preserving EHR system using attribute-based infrastructure*, In Proceedings of the ACM workshop on Cloud computing security, pp. (47-52), 2010.

[OSW07] **R. Ostrovsky, A. Sahai, B. Waters** - *Attribute-based encryption with non-monotonic access structures*, In Proceedings of the 14th ACM conference on Computer and communications security, pp. (195-203), 2007.

[Rya11] **Mark D. Ryan** - *Cloud computing privacy concerns on our doorstep*, Communications of the ACM, Vol. 54, No.1, pp.36–38, 2011.

[Rya13] **M. D. Ryan** - *Cloud computing security: The scientific challenge, and a survey of solutions*, Journal of Systems and Software, Vol.86, No.9, pp.2263-2268, 2013.

[Wat11] **Brent Waters** - *Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization*, Public Key Cryptography–PKC, Springer Berlin Heidelberg, pp. (53-70), 2011.

[Y+10a] **Shucheng Yu, Cong Wang, Kui Ren, Wenjing Lou** - *Attribute based data sharing with attribute revocation*. In Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, pp. 261-270, 2010.

[Y+10b] **S. Yu, C. Wang, K. Ren, W. Lou** - *Achieving secure, scalable, and fine-grained data access control in cloud computing*, IEEE Infocom proceedings, pp. 1-9, 2010.

[Y+11] **K. Yang, J. Zhang, W. Zhang, D. Qiao** - *A light-weight solution to preservation of access pattern privacy in un-trusted clouds*. In Proc. of the European Conf. On Research in Computer Security (ESORICS), pp.528-547, 2011.