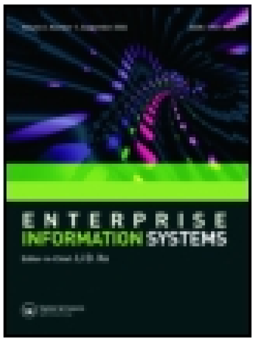

Enterprise Information Systems

ISSN: 1751-7575 (Print) 1751-7583 (Online) Journal homepage: http://www.tandfonline.com/loi/teis20

Guarded dual authentication based DRM with resurgence dynamic encryption techniques

Neha Agarwal, Ajay Rana & J.P. Pandey

To cite this article: Neha Agarwal, Ajay Rana & J.P. Pandey (2018): Guarded dual authentication based DRM with resurgence dynamic encryption techniques, Enterprise Information Systems, DOI: 10.1080/17517575.2018.1527041

To link to this article: https://doi.org/10.1080/17517575.2018.1527041

Published online: 27 Nov 2018.

Submit your article to this journal

Article views: 11

View Crossmark data

ENTERPRISE INFORMATION SYSTEMS
https://doi.org/10.1080/17517575.2018.1527041

Taylor & Francis
Taylor & Francis Group

Guarded dual authentication based DRM with resurgence dynamic encryption techniques

Neha Agarwal[a], Ajay Rana[a] and J.P. Pandey[b]

[a]Department of Computer Science Engineering, Amity University, Noida, India; [b]Department of Electrical Engineering, KNIT, Sultanpur, India

ABSTRACT

Cloud computing is the emergent technology that face one of the significant issues time with data security while outsourcing the data onto the cloud in recent. Some cryptographic techniques have been used for protection in form of identity, attributes and prediction algorithms nonetheless these algorithms lack their performance and becomes are very prone to attackers when an unauthorized user reunited the system with dissimilar way for privileges to the similar data files. The essential need of this data security solved by some enhanced cryptographic techniques in DRM utilizing a secure privacy preserving data sharing with encryption techniques of Dynamic Unidirectional Proxy Re-Encryption. This technique is based on Cipher text Policy Attribute by providing the privacy, integrity and security of the data while retrieving.

ARTICLE HISTORY
Received 8 July 2017
Accepted 18 September 2018

KEYWORDS
Twofold authentication protocol; Efficient Elliptic Curve Public Key Encryption (EECPKE); integrity checking protocol; Covariance matrix Adaptation Evolution Strategies (CMA-ES)

1. Introduction

Cloud computing deliver all the practicality of existing data services at the same time as it dramatically reduces the direct prices of computing that deter several organizations from deploying several with-it EIS services (Li et al. 2012). In EISs, there are two main trends for cloud computing specifically informatics potency and business legerity, whereby EISs will be used as a competitive tool through fast preparation, parallel instruction execution, use of compute-intensive business analytic and mobile interactive applications that respond in period of time to user needs (Da and Li 2011). By exploitation the rising technologies, cloud computing makes it potential for brand spanking new categories of applications and delivers services that weren't potential before. Recently in EIS, cloud computing is one among the quickly prospering fields for the reason that gives adaptable and on request services to the clients (Da Xu 2011). Without capitalizing significantly on infrastructure and maintenance, associations with low financial arrangement can be also used high computing and storage services (Yong et al. 2014). Conversely, lack of control on data and computation hoists greater security issues for the associations, ruining the broad adaptability of the public cloud (Aldossary and Allen 2016).

CONTACT Neha Agarwal ✉ agarwalnehajain@gmail.com Department of Computer Science Engineering, Amity University, Noida, India

The loss of control over data and the storage stage also motivates cloud customers to maintain the access control over data (SandeepSood et al. 2012). In addition to that, the customers essential cloud service suppliers has to maintain confidentiality and privacy of the data (Uddin et al. 2015). The supervision of confidentiality by a customer may not acquire any data with respect to the customer data. Cryptography is abused as a typical apparatus to bear the cost of confidentiality and privacy services to the data (Rafeeq et al. 2015).

The data are typically encrypted before storing to the cloud. The access control, key management, encryption, and decryption processes are managed by the customers to guarantee data security [9]. In any case, when the data are to be shared among a group, the cryptographic services should be sufficiently adaptable to deal with different clients, exercise the access control, and deal with the keys in an effective way to defend data confidentiality (Ali et al. 2017). The data handling among a group has certain additional characteristics rather than two-party communication or the data dealing with a single client. The current, leaving, and recently joining grouped members can turn out to be an insider threat damaging data confidentiality and privacy (Tung, Tseng, and Kuo 2015), (Patil, Khatawkar, and Dange 2017). Insider threat can turn out to be all the more disturbing because of the fact that they are launched by trusted elements (Chang and Hu 2017). Because of the fact that individuals trust insider entities, the research community focuses more on outside attackers (Singh, Jeong, and Park 2016).

A single key shared between all group individuals will result in the access to past data onto a recently joining member. The aforementioned circumstance abuses the confidentiality and the principle of least benefit (Mapoka, Shepherd, and Abd-Alhameed 2015). In this manner, in group shared data, insiders may generate the issue of backward access control and forward access control (Spandana and P Sunitha 2016; Choi, Choi, and Lee 2015). The basic solution to rekeying (producing another key, decrypting all data, and re-encrypting with the new key) does not prove to be scalable for frequent changes in the group membership (Hur, Koo, Shin, and Kang 2016).

A separate key for each client is a cumbersome solution. The data must be separately encrypted for each client in such a scenario (Blasco, Tapiador, Peris-Lopez, and Suarez-Tangil 2015). The existing and authentic group individuals may demonstrate ill-conceived conduct to control the data (Taylor 2017; Fabian, Ermakova, and Junghanns 2015). The presence of the whole symmetric key with a client enables a malicious client to transform into an insider threat (Kumari and Khan 2014). The data can be decrypted, modified, and re encrypted by a malicious insider within a group. Consequently, an authentic client in the group may access certain unapproved records within the group (Catuogno, Löhr, Winandy, and Sadeghi 2014). In any case, these strategies lacks their performance to some level when an unapproved client rejoined the system with disparate path in benefits to the comparative data records as they are exceptionally inclined to attackers and in this way it becomes basic to build up a secure data imparting structure to some enhanced cryptographic techniques (Samanthula, Elmehdwi, Howser and Madria 2015).

Considering the issues in the cryptographic techniques of DRM(Digital Right Management) and to improve the security, identity for authenticating users through cloud and to enhance the computation time, the paper proposed the two fold authentication protocol for identification and validation of the cloud user and for checking the security,trustworthiness of the data . Then for a strongest access control, Dynamic Unidirectional Proxy Re-Encryption is proposed which follows an integrity checking protocol. At the end, a novel algorithm is utilized to store the data in cloud server and remote server so that if there is any loss of main file there is always a backup in the remote server. The upcoming sections detail the process and the paper is composed as follows. Section 2 comprises of some of the recent research work. Section 3 gives the proposed authentication mechanism follows encryption technique and a novel algorithm for cloud through data sharing. Section 4 incorporates the simulation results and conclusion in section 5.

2. Related research

Some of the recent research papers which are relevant to the data distribution guaranteed in cloud computing is listed below:-

Sandhu and Bhathal (2016) discussed about a symmetric key understanding algorithm to tackle the issue of Key management and Key Sharing because it decreased the unwavering quality. To deal with that issue, measured design for key sharing and key management in completely Homomorphic Encryption plan was created. In their method, Diffie Hellman symmetric key algorithm was utilized to create session key between two groups and HMAC was utilized to create OTP (One Time Password) for greater security. Their approach shared session key among client and cloud. For each process, new key was created between two preceding correspondence selected node assume user1. Because of this, the issue of dealing with the key was ousted and information was more secured. Results showed that completely homomorphic encryption system was more effective than full disk encryption.

Balu and Kuppusamy (2010) recommended the strategy to save the privacy of the encryptor. In cloud computing the access policies are the logical combinations of the attributes, which were sent alongside the cipher text. The decryption process is done just when any of the attribute matches with the access policy. Accordingly this makes the process complicated, so they introduced the strategy for Ciphertext Policy Attribute-Based Encryption (CPABE) without sending access policy with the cipher text. The Diffe-Hellman presumption also utilized for secure encryption and decryption

Liu Z et al. (2013) introduced the strategy for traceable CP-ABE (T-CP-ABE) systems with any monotone access structures to trace the issue policy expressiveness in the system. In CP-ABE system, the decryption keys with attributes are shared by various clients. During decryption process the decryption benefits are not generally predict the first key proprietors; it also traces the malicious clients because the malicious client also consist of same arrangement of attributes. They share these attributes to outsiders for cash aspect or escape from the hazard. In this manner this issue affects the application of CP-ABE. Along these lines this work was designed without debilitating the security of CP-ABE system utilizing traceable mechanism.

Lu and Li (2016) introduced the Certificate-based intermediary re-encryption strategy without bilinear blending. The information proprietors should encrypt the information while storing them into the cloud storage. There are various dangers in effective sharing of delicate information in public clouds. Hence the intermediary re-encryption gives the perfect solution to information sharing. This re-encryption process enables the information proprietors to decrypt the encrypted information in the public cloud with any direct interaction to the unapproved client. The information confidentiality and security is demonstrated utilizing classic computational Diffie-Hellman supposition.

Shao et al. (2016) investigated the bidirectional intermediary encryption technique with three attractive properties such as size of constant cipher text without considering the number of change, master secret security, re playable chosen cipher text(RCCA) security. The properties utilized as a part of bidirectional intermediary re-encryption was an adaptable apparatus which is utilized as a part of numerous condition such as cryptographic cloud storage that exchanges the cipher text between the source and destination through semi-trusted intermediary. The master secret security technique gives the privileges of decryption to the information proprietors.

Wang et al. (2014) clarified intermediary re-encryption (PRE) as the cloud data encryption technique. In a PRE system, a semi-trusted intermediary can change a ciphertext under one public key into a ciphertext of a comparative message under another public key, however the intermediary cannot increase any information about the message. In this work, a certificate less PRE (CL-PRE) scheme without pairings is created. The security of this scheme can be ended up being identical to the computational Diffie-Hellman (CDH) issue in the oracle model. The new scheme does not require the public key certificates to guarantee legitimacy of public keys and deals with the key escrow issue in character based public key cryptography.

Zhou, Huang, and Wang (2015) discussed about privacy protecting constant Cipher text Policy Attribute Based Encryption (PP-CPABE) technique which maintains the constant cipher text estimate for any number of attributes and furthermore introduced privacy saving attribute based broadcast encryption (PP-AB-BE)method for maintaining expressive hidden access policy. In this method, a potential malicious attacker does not reflect on both the access policies and user's secrecy. Moreover the past privacy saving method also saves the secrecy but it requires higher cipher text size.

Yibin et.al (2017) discussed Associate in nursing intelligent cryptography approach, by that the cloud service operators cannot directly reach partial information. The approach divides the file and individually stores the info within the distributed cloud servers. another approach is intended to see whether or not the info packets would like a split so as to shorten the operation time. it had been entitled Security-Aware economical Distributed Storage (SA-EDS) model, that is especially supported by algorithms, together with various information Distribution (AD2) algorithm, Secure economical information Distributions (SED2) algorithmic program and economic data Conflation (ED Con) algorithmic program. The theme may effectively defend major threats from cloud-side and also the computation time was shorter than current active approaches however it might fails to deal with securing information duplications so as to extend the extent {of information of knowledge| of information} handiness since any of knowledge center's down can cause the failure of data retrievals.

According to the Literature survey, Sandhu and Bhathal (2016) discussed about a symmetric key understanding algorithm to tackle the issue of Key management and Key Sharing because it decreased the unwavering quality. The work of Zhou, Huang, and Wang (2015) and Shao et al. (2016) investigated encryption technique with three attractive properties such as size of constant cipher text without considering the number of change, master secret security, re playable chosen cipher text(RCCA) security. Also as an overall suggestion Li et al. (2017) discussed an intelligent cryptography approach, by which the cloud service operators cannot directly reach partial data. The approach divides the file and separately stores the data in the distributed cloud servers. An alternative approach is designed to determine whether the data packets need a split in order to shorten the operation time. By the overall work, it portrays that many researches have suggested the solutions for the problems facing in data security, user authentication, protection of data from the internal attacks etc. But there are no clear solutions yet. Hence it is important to design a secure data preserving approach for user authorization with aid of backup mechanism to store and recover large number of data's.

3. Dual authentication based security framework for cloud based data sharing applications

Nowadays internet-based computing is one of the developing techniques with a high infrastructure. To share the file securely there are several mature and powerful options, among them one of the leading concept is cloud which shares the data, stores and retrieves the data from cloud through network by still creating a problem of security lacking, sharing efficiency etc. For data security, in earlier stages a product key with a typically alphanumerical serial number is used to represent a license for a particular piece of data, serves a similar function. During the installation process or launch for the data, the user is asked to input the key. If the key correctly corresponds to a valid license (typically via internal algorithms),the key is accepted, then the user who bought can continue. In modern practice, product keys are typically combined with other DRM practices (such as online 'activation'), as the data could be cracked to run without a product key, or 'keygen' programs could be developed to generate keys that would be accepted. But the computation time and other similar process meets a great risk. So to overcome the needs, we adopt a concept of security framework with two fold authentication in DRM. The two fold authentication protocol works based on the identification of the cloud user and validation based on the identified user through mutual authentication. Then for a strongest access control, Dynamic Unidirectional Proxy Re-Encryption is used which follows an integrity checking protocol. At the end a novel algorithm is utilized to store the data in cloud server and remote server so that if there is any loss of main file there is always a backup in the remote server. The Figure 1 displays the process of this proposed method.

3.1. Problem formulation

The certain attributes of the cloud computing had impact on its security and privacy issues such as the network security and attackers in the cloud. The security of data is one of the important issues while retrieving the data from cloud to user panel. The data storage and retrieval for the application data or any other data can done by cloud users who are all authenticated through cloud data owners in storing as well as retrieving the data, maintaining integrity and confidentiality. The security of the data stored in cloud

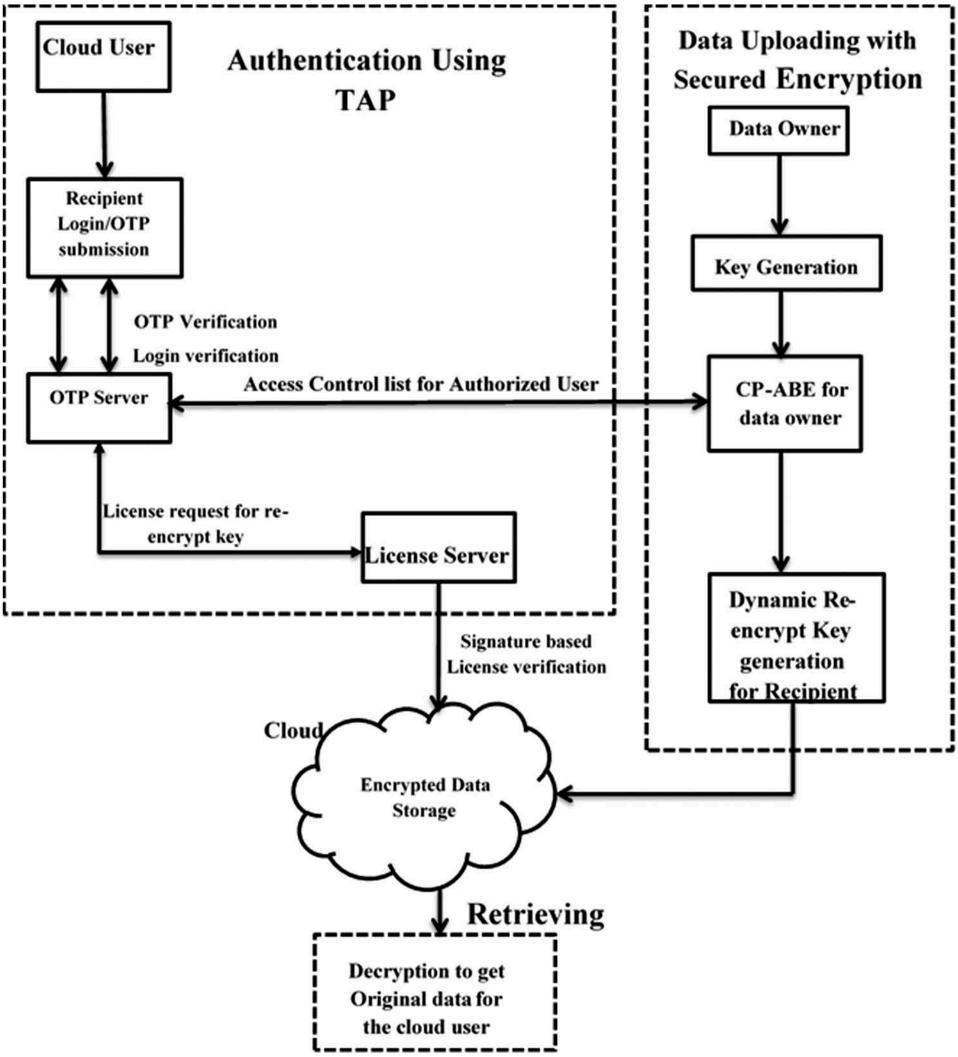
Cloud User
Authentication Using TAP
Recipient Login/OTP submission
OTP Verification
Login verification
OTP Server
Access Control list for Authorized User
License request for re-encrypt key
License Server
Data Uploading with Secured Encryption
Data Owner
Key Generation
CP-ABE for data owner
Dynamic Re-encrypt Key generation for Recipient
Signature based License verification
Cloud
Encrypted Data Storage
Retrieving
Decryption to get Original data for the cloud user

Figure 1. Architecture of proposed method.

can be protected using some cryptographic techniques and this can done based on identity, attributes and prediction. But these techniques are lacks in their performance to some level and very inclined to attackers while encrypting these data. Thus it is necessary to develop a secure cloud data sharing framework with enhanced cryptographic techniques to automate the system. The remaining process in the proposed framework is explained through the following steps.

The opening phase starts with the basis of authentication which means, it authenticates that the requested user is an authorized user or not. For that a twofold authentication protocol is used for the reorganization of the user regarding his/her details and validates the identified user of the cloud. After the verification of the users they can access the cloud applications like share, store or retrieve the content from cloud server. It is an essential one while accessing these features, the data should not be in a plaintext format. For that an encryption technique of elliptic curve cryptographic encryption is used. After the completion of these process to check the integrity of the data an integrity checking protocol is used and at last a novel algorithm is used to store the data in two different servers as i) cloud server and ii) remote server. The main motto of this algorithm is to get the backup of the data even though it is dropped. By using this algorithm the data which is dropped from the main file can able to retrieve it from the remote server. Adopting these methods, the control measures of cloud can be controlled.

3.2. Cloud user authentication via Twofold Authentication Protocol (TAP)

In this phase the authentication of the cloud user is validated with the help of the Twofold Authentication Protocol (TAP) which means the authentication is done by two phases namely identification and onetime password verification (OTP). The intention rear of this protocol is to provide and expand a security aware authentication, where the end user's credential performs as a distinct identity, which is utilized to verify the validity of the user through secure hash algorithm.To accomplish high reliable authentication, an authentication protocol named as registration protocol is initiated. The input factors of TAP is i) user id and password ii) tokens with a one-time password are explained below section.

3.2.1. OTP server for authentication

A Onetime Password (OTP) like a ticket granting server is introduced here. If a user want to access the data stored in a cloud server or a data center, the user must get one time token from the OTP server, after interact with the Cloud, the OTP token will be verified. Then only the user will be allowed to refer the knowledge which is stored in the datacenter. The end user must get the authentication one time token for each and every time. By this we can avoid the misbehaved users. If a user wants to join into the cloud, first step the user have to prove their identity. Here user first communicates to the OTP server and reveals their identity. Then the OTP server checks with the identity provided by the user and verify for the trust value of the user. If it found the reliable value then it gives an access control. The OTP may be combined with the system IP address, and the key given to the user will also be informed to the CSP and Cloud Data Center. Then the user has to enter into the cloud data center through the CSP with the secret key which was given by the License Server. If the key match with the key given by OTP server to the CSP, then the user will be allowed to access the cloud data. By attaining two time verification with the reliable protocol, the cloud user can be authenticated with more reliable.

3.2.2. Login phase and password reset phase

This method extended from initial registration protocol into Login Protocol, which is going to be utilized throughout the authentication process. In this phase, user C_i visits the authority Login page and input credential username and cloud system generated password through public internet (i.e. Social Access Media such as e-mail and other web based service in the public internet requires user registration and password), the user C_i inputted the username alphanumeric values are automatically converted into ASCII value denoted as U_A. From the cloud service provider CSP_i of the public the credentials are authorized.

(a) Estimates $LV_i = [U_A, P_i]$ and checks whether LV_i is identical to the document saved by the user C_i in the public cloud service provider. If LV_i equals to store RA_i then validate the IP address of public cloud service provider, the authority login page continues the estimation else the session will be dismissed.
(b) Computes $OP_i = h_{fc}(P_i \oplus T_c)$, where, OP_i is the one-time password for estimating variable and T_c represents the current for the timestamp.
(c) The public cloud service provider CSP_i transmits one time password OTP to the user C_i through RA_i for additional security over an estimated timestamp. Estimates $S_m = [IP_i, OP_i, T_c]$. The user C_i dispatches the appropriate OTP as input and transmits the information to the public cloud; here the channel of the network is unsecure.
(d) Estimates $(T_s - T_c) > \Delta T_c$, the public cloud service provider CSP_i correlate the shape of the timestamp T_c, but if it results in erroneous format, then the cloud excludes the login request. If the subtracted value of the current time stamp from T_c of the cloud server T_s is larger than that of assumed time interval ΔT_c of the system, at this situation also the system excludes the login request.
(e) If the actual timestamp is contained in a correct format and under the necessary time interval then the system permits to contact the user C_i and use the cloud application.

The password-reset phase is the next general protocol in this paper. This stage begins at the time during the new password requested by the user C_i. The login process is similar as the above mentioned Login Phase; hence the login protocols not discussed. The user C_i keeps contact with his smart application and the information such as username and password is given as input. Then the login authentication from the public cloud will request the user to provide his new password after that the user device executes as follows.

(a) Estimates $LV_i = [U_A, P_i]$, where, LV_i is the login computing variable.
(b) Verifies whether LV_i is identical as the saved P_i or not. If the sign-in is accomplished then the authority login page continues to estimate, else the session will be discarded.
(c) An estimate $NP_i = P_n \oplus h_{fc}(U_A \oplus k)$, where, NP_i is the new password computing variable, P_n is the recent password. Later successful verification, the cloud creates new password integrated with hash chain and save it to the cloud database.
(d) After that the system saves and substitutes the new password P_n with the previous P_i and the phase is discarded effectively.

3.3. Cipher text policy attribute based encryption

Once the cloud user is authenticated by means of TAP to perform secure data access, the encryption of the data is necessary. In this encryption system, the data owner encrypts both the index and files by generating an index to the file collection. Finally, the official user generates a request and delivers to the server. When the cloud server receives a request, it provides tsshe decrypt key to the authorized user. Then, the user decrypts the files to extracts the original data. For encryption process the optimized algorithm Cipher text Policy Attribute based Encryption (CT-ABE) scheme is proposed

In this, the cipher text, public key, secret key of user and re-encryption keys are all named with information of the version indicating the master key version of system. In attribute revocation, it does not have to the change of information. For that, one secret attribute S_A for the need of key administration is portrayed. Every user's attribute set contain S_A in addition to significant attributes. The S_A never be updated which ensures cloud servers cannot acquire the whole secret key components of a user.

A. Setup: The setup phase obtains input as a security parameter K. It selects a bilinear group b_o of prime order p with b as generator, and bilinear map $e : B_0 \times B_0 \rightarrow B_1$. The universe attribute is $v = \{1, 2, ...i\}$. It selects $t_n \in Z_p$ for attribute n, $n, 1 \leq n \leq i$, and a random exponent $\beta \in Z_p$. The public key P_K and master key M_K is given by

$$P_K = \left\{B_0, e, b, e(b, b)^a, \{T_n = b^{t_n}\}_{n=1}^{i}\right\} \tag{1}$$

$$M_K = \left\{\beta, \{t_n\}_{n=1}^{i}\right\} \tag{2}$$

Though P_K is publicly known to all system parties, M_K is kept secretly by trusted authority (T_A).

Algorithm for Setup

```
Input: $K$
Outputs: $P_K, M_K$
Begin
Setup $K, v$
Select a bilinear group $b_0$ and bilinear map $e : B_0 \times B_0 \rightarrow B_1$
if(i ≥ 1) then
    $t_n \in Z_p$ is selected
End if
Run the setup using $T_A$
End
```

***B. Key Generation*:** The key generation phase takes set of attributes S as input and the secret key equivalent to S is produced as output. Initially, it selects a random number from $r \in Z_p^*$. Then, it calculates the key as

$$S_K = \left(D = b^{a-r}, \forall n \in S : D_n = b^{r - t_i^{-n}}\right) \tag{3}$$

Algorithm for key generation

```
Input: S
Output: $S_K$ $S_K = \left(D = b^{a-r}, \forall n \in S : D_n = b^{r - t_i^{-n}}\right)$
Begin
Select random number from $r \in Z_P^*$
Compute the output
End
```

***C. Encryption*:** The encryption phase takes access control tree *T* as input, a public key P_K, and message *M*. The output of cipher text is as follows. Initially, it describes the random polynomial q_y for every node *y* of tree *T* in top down approach beginning from *r*. The node *y* with degree d_y is smaller than the threshold value k_y of that node, that is $d_y = k_y - 1$. For *r*, $q_r(0) = s$, that is a random number of Z_p. For each and every non-root node *y*, $q_y(0) = q_{par(y)}(ind(y))$ where *par(y)* represents *y's* parent and *Ind(y)* represents *y's* exceptional index specified by its parent. *Y* is the set of nodes of leaf in *T*. The cipher text is then created by giving the access structure of tree *T* and calculates

$$C_T = \left(T, \widetilde{C} = M.e(b,b)^{as}, C = b^s, \forall x \in Y : C_x = T_x^{q_x(0)}\right) \tag{4}$$

Algorithm for Encryption

```
Input: T, $P_K$, M
Output: $C_T$
Begin
    Represent a random polynomial $q_y$
    if(node = = root node) then
        $q_r(0) = S$
    End if
    if (node = = non-root node) then
        $q_y(0) = q_{par(y)}(ind(y))$
    End if
    Calculate $C_T$
End
```

The randomly generated cipher texts are optimized with constraint to the parameters called Encryption Quality, correlation coefficient of the cipher texts and differential attack. The definitions and calculation of these parameters, the objective function formulation and the optimization of the cipher texts are clearly illustrated as follows.

i) Encryption Quality

The Encryption Quality is denoted as Q_E is calculated for measuring the Bitmap image encryption quality and modified the calculation for the cipher texts from the plain texts. Then, Q_E for the generated cipher texts is calculated as mentioned below.

- Measure the deviation among the plaintext and the cipher text in how many places they are differing and this is calculated using the following Equation (5).

$$d = \left|pl - c_2^n\right|, \;\; n = 1, 2, ...N \tag{5}$$

- Compute the average value of bits deviation as given in Equation (6).

$$\bar{d} = \frac{1}{pl_{Len}}(d) \tag{6}$$

- Calculate Encryption Quality Q_E as in Equation (7).

$$Q_E = \left|d - \bar{d}\right| \tag{7}$$

This is the first objective of this optimization technique and this aim is to maximize the Encryption Quality and hence the first part of objective function is given as in Equation (8).

$$f_1 = \max(Q_E) \tag{8}$$

The second objective function called correlation coefficient is measured and formulated as follows.

ii) Correlation coefficient

The correlation coefficient is denoted as $r_{\rho c_2^n}$ and this is also one of the parameter used for measuring the correlation between pixels of the encrypted and original image and similarly the correlation between the cipher text and the plain text is calculated and the calculation is given as follows.

- Calculate the mean of both ρ and c_2^n using the Equations (9) and (10) given below

$$E(pl) = \frac{1}{pl_{Len}} \sum_{i=1}^{pl_{Len}} pl_i \tag{9}$$

$$E(c_2^n) = \frac{1}{pl_{Len}} \sum_{i=1}^{pl_{Len}} c_{2i}^n \tag{10}$$

- Measure covariance between pl and c_2^n using Equation (11) as given below.

$$\mathrm{cov}(pl, c_2^n) = E\left[(pl - E(pl))\left(c_2^n - E\left(c_2^n\right)\right)\right] \tag{11}$$

- Then the standard deviations of pl and c_2^n is calculated using the Equations (12) and (13) as given below.

$$std(pl) = \frac{1}{pl_{Len}} \sum_{i=1}^{pl_{Len}} (pl_i - E(pl))^2 \tag{12}$$

$$std(c_2^n) = \frac{1}{pl_{Len}} \sum_{i=1}^{pl_{Len}} (c_{2i}^n - E(c_2^n))^2 \tag{13}$$

- Finally $r_{\rho c_2^n}$ is calculated via Equation (15) as given below.

$$r_{\rho c_2^n} = \frac{\mathrm{cov}(pl, c_2^n)}{\sqrt{std(pl)}\sqrt{std(c_2^n)}} \tag{14}$$

It is to be noted that depending on the value of $r_{\rho c_2^n}$, the relationship between pl and c_2^n is decided and for the encryption should be a successful one the relation should be minimum. The value of $r_{\rho c_2^n}$ and the relationship between the original and encrypted texts is given by the following condition in (15).

$$r_{\rho c_2^n} = \begin{cases} >0, & \textit{Strong positive relationship pl and } c_2^n \\ 0, & \textit{No relationship pl and } c_2^n \\ <0, & \textit{Strong negative relationship between pl and } c_2^n \end{cases} \tag{15}$$

As seen from the condition (16) the objective is that $r_{\rho c_2^n}$ should be less than or equal to zero and the second part of the objective function is given as in Equation (16).

$$f_2 = \min(r_{\rho c_2^n}) \tag{16}$$

Then the third objective function called Differential Attack is formulated as given below.

iii) Differential attack

Differential Attack is the one in which the attacker will try to observe the change in the encrypted data by means of modifying some bit values in the original information. There are two measures are used to detect the impact of the single bit value on the whole encrypted image and this also suits for this proposed framework where analyze the impact of single bit variation in the plaintext to that of the cipher text. The measures are (a) Information Entropy factor and (b) Avalanche Effect (AE) and the calculation of these measures are given in Equations (17) and (18).

$$H(m) = -\sum\{0 \le i \le n-1\}p(m_i)\log_2 p(m_i) \tag{17}$$

Where $p(m_i)$ indicates probability of m_i.

$$AE = \frac{Ham\min g\, Dis\tan ce}{File\, Size} \tag{18}$$

The measures given in Equations (17) and (18) should also be maximum to avoid the Differential Attack and thus the final part of this objective function is formulated as given in Equation (19).

$$f_3 = \max(H(m)) + \max(AE) \tag{19}$$

The overall objective function of the proposed method and the optimization of the best cipher text is given in the next section.

iv) Objective function Formulation and cipher text optimization

The formation of the objective function of this proposed framework is thus given by combining the Equations (8), (16) and (19) and the overall objective function f is given in the Equation (20).

$$\left.\begin{aligned} f &= f_1 + f_2 + f_3 \\ f &= \max(Q_E) + \min(r_{\rho c_2^n}) + \max(H(m)) + \max(AE) \end{aligned}\right\} \tag{20}$$

Based on the objective function given in Equation (15) the optimization of the cipher text is performed.

3.4. Dynamic unidirectional proxy re-encryption

A unidirectional conditional PRE scheme consists of a number of steps such as Set-up, Key generation, Dynamic Rekey generation, Encryption1, Encryption2, ReEncryption1, ReEncryption2, Decryption1, and Decryption 2.

Step 1–Set-up (*K*) → P_{par}: The dynamic Unidirectional Proxy Re-encryption algorithm is running by a trusted party. Set input as *K(*security parameter) to produce the outputP_{par} (public parameters). So, that it will be used for all the parties in the scheme.

Step 2- Key generation (*K,* P_{par}) → (S_k, P_k): Byusing the randomized algorithm, generateprivate key(S_k)/public key (P_k) pair for all the parties using *K* andP_{par}.

Step 3- Dynamic Rekey generation(P_{par}, $dS_{k\mathrm{i}}$, P_{kj}, M') → Rijn: For the given public parameters P_{par}, a keyword sets M', User *i*'s dynamic private key $dS_{k\mathrm{i}}$ and User *j*'s public key P_{kj}, the randomized algorithm produce a key $R_{ij.}$ For a unique keyword *i*'s second level cipher texts are translated into *j*'s encrypted first level cipher texts.

Let us consider the condition that the keywords are concatenated with 'AND' gates. For the given dynamic private key $dS_{k\mathrm{i}}$ of user i, the public key P_{kj} of user j and its keyword-set $M' = (m'_1, m'_2, ...m'_k)$ for $k < n$, randomized algorithm generates the proxy re-encryption key *rk*, where *k* denotes keywords in *M*′. For different users, the keyword datasets will create a pair of key at each unique keyword.

Step 4- Encryption1 (P_{par}, P_k,P) → C: first level cipher text is generated using the randomized algorithm for the given parameter of P_{par}, P_k P. Third party cannot re-encrypt this cipher text.

Step 5- Encryption 2 (P_{par},P_k, P) → C: The second level ciphertext is generated using the randomized algorithm for the given parameter of P_{par}, P_k P that can be re-encrypted into a first level cipher text using the suitable re-encryption key.

Step 6- Re-Encryption 1 (E_i, P_{par}, $\mathrm{R_{ij}}$, C) → $\mathrm{C_1}$.this randomized algorithm takes as input as P_{par}, a re-encryption key $\mathrm{R_{ij,}}$a second level cipher text C encrypted for user *i*'s public key and also introduce an auxiliary variable E_i for Re-encryption. The auxiliary variable E_i captures the dynamic key set generated with respect to the different users keyword sets. For a given ciphertext as input, $E_i$re-encrypts it to C1 using $\mathrm{R_{ij}}$ rk_1*→j1 and decrypts $\mathrm{C_1}$ using S_{kj1}.It repeats the re-encryption and the decryption by using ($\mathrm{R_{ij}}$ rk_2*→j2,S_{kj2}). It rejects if the two executions have variable results .The first level ciphertext's output, $\mathrm{C_1}$ re-encrypted for user j. In a single hop scheme, $\mathrm{C_1}$ cannot be further re-encrypted.

Step 7- Decryption 1 (P_{par}, S_k, C) → P: the randomized algorithm outputs either a plaintext P∈ {0, 1} * or a 'invalid' message using the given P_{par}, S_k, C parameters.

Step 8- Decryption 2 (P_{par}, S_k, C) → P: the algorithm returns either a plaintext P∈ {0, 1} * or a 'invalid' message using the given P_{par}, S_k, C parameters.

3.5 Decryption phase

The decryption phase takes cipher text C_T, public key P_K and the secret key S_K. It initially defines whether the attribute sets fulfills access structure T in C_T. If the node y is leaf node, it is indicated as $n = arr(y)$, and calculates

$$e(C_n, D_n) = e\left(b^{t_n.q_y(0)}, b^{r.t_n^{-1}}\right) = e(b, b)^{r.q_y(0)} \tag{21}$$

If $n \in S$,$Decrypt\ \ Node\ \ (C_T, S_K, y) = \perp$. Then, by utilizing polynomial interpolation approach, the pairing results in bottom-up manner were grouped and finally the blind factor $A = e(b, b)^{rs}$. The phase now decrypts by calculating

$$\tilde{C}/(e(C, D).A) = M \tag{22}$$

The following algorithm deals with the decryption of the message (L, C, and T). The cloud user carries out the verification of the embedded public key L. If the public key L flops then the information will be ignored. So the recipient makes use of key establishment protocol the shared secret value of Y. The shared secret key is estimated by the S_k of the recipient and fixed public key L. If the estimated value of the Y is infinitude then the recipient discards the content. The recipient makes use of KDF which is set up between transmitter and receiver to produce the keying data K which is integration of k1 and K2. The length of K can be calculated from the Encryption key lengthk1 and MAC key length K2. Encryption key length is employed for symmetric key decryption procedure and MAC key length is employed as MAC key. K1 is utilized to decode the cipher text C into m and MAC algorithm makes use of key K2 to estimate tag T1. If T1 is identical to T, then it agrees to acquire the cipher text or, the cipher text gets ignored.

Algorithm 3: Decryption for cloud user

```
Input: Domain parameters $(f,\ V,\ p,\ q,\ u,\ h)$, private key $s_k$, Cipher text $(L\,, C,\ T)$
Output: plaintext m or cipher text rejection
Assumption: Both the sender and receiver has their respective key pairs such as public and private keys and also other public key
Begin
Perform validation for embedded public key $L$
$if$ (Validation fails) then
    Result ("Reject the cipher text)
End if
Calculate $Y =\ h.\,(k.\,(s_k)\,L)$
$if\ \ Y == \infty$ then
    Result ('Reject the cipher text')
End if
$(k1, k2) \leftarrow KDF(xY, L)$ Where xy is the x-coordinate of Y
Calculate $T_1 = MAC(\ K2(C))$
$if\ \ (T_1 \neq T)\ then$
    Result ('{reject the cipher text')
End if
```

```
15. Calculate $m = DEC.(k1.(C))$
16. Result $(m)$
17. End
```

By employing the Dynamic Unidirectional Proxy Re-Encryption techniques, a proxy re-encryption to reveals the information and encrypted with public key to a 3rd party . Also it becomes attainable to art incoming encrypted knowledge to Associate in Nursing external filtering contractor at the initial entree, while not risking exposure of plaintexts at the entrée itself.

4. Results and discussion

The implementation of this new secure data storage with optimized EECPKE cryptographic technique is executed in the cloud sim tool. The experimental set up was implemented in Java and the results produced with different performance measures. The efficiency of the proposed framework on comparing with the existing techniques are presented in detail.

4.1. Experimental setup

The Proposed framework is developed to be suitable for any type of application so it is implemented with Cloud Simulator using Java Language. The data size to be stored and recovered and the number of cloud user is considered as one in this proposed framework and later the number is varied to validate the efficiency of this proposed framework on compared to other methods. The results generated by this proposed framework with different performance measures are presented in this section.

4.2. Dataset

The dataset for the data storage is collected from the medical dataset which is the Drug and Health plan data of the year 2015 obtained from the official US government site for medicine. The dataset contains Cost Benefit Report Structure, Geography, Local Contract Service Areas, Plan Cobrand Names, Plan Drugs Cost Sharing, Plan Drug Tier Cost, Regional Contract Service Areas and Plan Services. Among those, this method uses Plan Services for the store and retrieval from the cloud user side to the CSP and the corresponding experimentation results are given in the following sections.

4.3. Performance evaluation with comparison

The efficiency of this proposed framework can be proved further by comparing the results produced by this proposed one with other conventional techniques like IBA [33],EPDR [34],TEES [35]. Here the comparison is performed in two phases, they are the Encryption Time of the data with different encryption techniques also the number of valid Decryption Time with different techniques. The comparison is made between this proposed method and the existing techniques were discussed in following manner.

4.3.1. Twofold authentication

By comparing the existing identity based authentication(IBA) schemes, EPPDR scheme, TEES scheme with this proposed twofold authentication scheme provides additional features to ignore external attacks. Table 1 presents the compressed correlation of the identity based scheme and this new technique's comparison to several damages.

The Table 1 given below displays the feature confrontation between IBA, EPPDR, TEES and this proposed scheme.

For the evaluation of the proposed scheme with the other, in the Table 1 the performance of the proposed scheme with five attacks namely Denial-of-service, Password guess attack, Offline guessing attack, Insider attack, False login,Online guessing attack, Impersonation attack. The performance of the proposed scheme is compared with other scheme like EPPDR, TEES, IBM by their ability of securing the data's from the attacks. By evaluating it shows that, in IBM there is a chance for all the attacks .While in EPPDR, Offline guessing attack, False login, Impersonation attack only solved. As on another scheme of TEES, it can able to protect from the internal attacks such as Denial-of-service, False login, Online guessing attack only not the other attacks. But as an emerging trend, our proposed scheme can protect the files from all the attacks such as Denial-of-service, Password guess attack, Offline guessing attack, Insider attack, False login, Online guessing attack, Impersonation attack.

The algorithm mentioned above in Registration and Login step expend 2.7 KB and 3.0 KB of the entire memory concurrently and this registration stage authentication protocol algorithm exploits single-way hash functions. In windows operating system the hash function h_{fc} spend about 24 KB of the entire ROM.

The initial values of the chain may lead the cellular phones to execute the hash functions several times. This method makes use of single-way collision free hash functions which is appropriately faster. This system also needs common authentication, which helps to make less complicated. Another major performance of this schemes is, not even a bit of data is saved within a user system like smart phones then tablets, simultaneously it makes the system less affected but more processing efficient and utilize less memory. Taking the above said advantages of our proposed scheme, it is tabulated in the Table 2 by comparing Operational mechanism, Forward secrecy, Scheme efficiency, Password change, Lost or stolen, Cloud based,authentication, Cost efficient (cloud computing) with the other scheme like EPPDR,IBA,TEES. It has been proved that our proposed scheme works well than the others.

Table 1. Security attacks analysis.

Security Attacks	IBA Scheme	EPPDR scheme	TEES Scheme	Proposed Scheme
Password guess attack	✓	✓	✓	×
Denial-of-service	✓	×	×	×
Offline guessing attack	✓	×	✓	×
Insider attack	✓	✓	✓	×
False login	✓	×	×	×
Online guessing attack	✓	✓	×	×
Impersonation attack	✓	×	✓	×

Table 2. Features and provision analysis.

Features	IBA Scheme	EPPDR Scheme	TEES Scheme	Proposed Scheme
Operational mechanism	×	×	×	✓
Forward secrecy	×	✓	×	✓
Scheme efficiency	×	×	✓	✓
Password change	×	✓	✓	✓
Lost or stolen	×	✓	✓	✓
Cloud based authentication	×	×	×	✓
Cost efficient (cloud computing)	×	✓	✓	✓

4.3.2. Encryption time

It is the time taken by the encryption algorithm to convert the plain text into the cipher text. The encryption time of the data after by the proposed optimized EECPKE technique is compared with the other encryption techniques for instance EPPDR,TEES.

$$Encryption\ time = \frac{computation\ time}{Response\ time}$$

For encryption calculation,the proposed scheme is evaluated with five different file sizes namely $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6,}$ with the size of 160,187,203,465,1586,3873 Kb that is stated in Table 3. The encryption time obtained for each file are 0.10, 0.11, 0.124, 0.239, 0.789, 2.126 sec. The following Figure 2 shows the details of encryption time.

The above Figure 2 shows the encryption time for different files for the proposed scheme. Encryption time is estimated based on the computation time divided by response time. Encrypted text is evaluated by the cipher text policy attribute based encryption. For increase in file size, there is a gradual increase in encryption. This shows that the encryption time for the large files increase's slightly, there no far variation or increase in encryption time.

The above Table 4 shows the comparison between the encryption time of proposed method EPPDR, TEES encryption time with different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6,}$ with the size of 160, 187, 203, 465, 1586, 3873 Kb. The results exposed that the proposed encryption technique attains better performance with an encryption time of 0.10000, 0.11000, 0.12400, 0.23900, 0.78600, 2.12600 sec for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ F_6. While the other like EPPDR, TEES attains 0.19000, 0.21700, 0.25000, 0.34000, 0.96400, 2.35000 sec and 0.185, 0.207, 0.217, 0.304, 0.814, 2.1570 with different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ F_6 that is depicted in Figure 3.

With the results generated by these encryption techniques with varying file sizes .In this the proposed one is compared and the outcomes are given in above figure shows that the proposed method had low encryption time.

Table 3. Encryption time for proposed scheme.

		Encryption Time(Sec)
File Name	Size (Kb)	Proposed
F1	160	0.10000
F2	187	0.11000
F3	203	0.12400
F4	465	0.23900
F5	1586	0.78600
F6	3873	2.12600

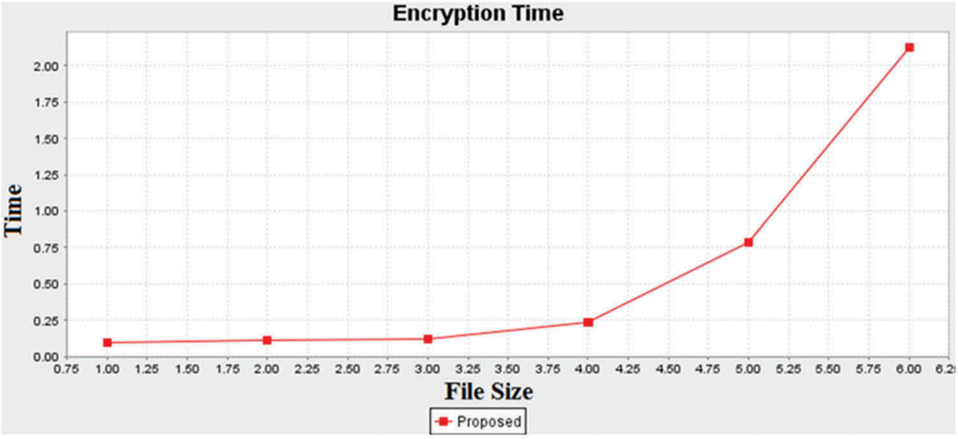

Figure 2. Encryption graph for proposed scheme.

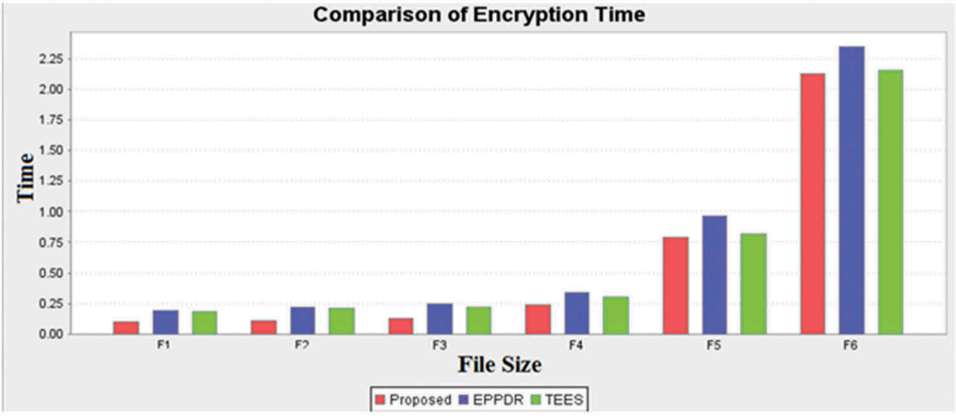

Figure 3. Comparison graph of encryption process.

Table 4. Comparison of encryption time.

		Encryption Time(Sec)		
File Name	Size (Kb)	Proposed	EPPDR	TEES
F1	160	0.10000	0.19000	0.185
F2	187	0.11000	0.21700	0.207
F3	203	0.12400	0.25000	0.217
F4	465	0.23900	0.34000	0.304
F5	1586	0.78600	0.96400	0.814
F6	3873	2.12600	2.35000	2.1570

4.3.3. *Decryption time*

It is the time taken by the decryption algorithm to convert the cipher text into the plain text. The decryption time of the data for the proposed technique is compared with the other techniques such as EPPDR, TEES

Table 5. Decryption time for proposed scheme.

		Decryption Time(Sec)
File Name	Size(Kb)	Proposed
F1	160	0.15000
F2	187	0.25800
F3	203	0.30000
F4	465	0.89600
F5	1586	2.4600
F6	3873	7.13700

$$Decryption\ time = \frac{computation\ time}{Response\ time}$$

Same as encryption,the proposed scheme is evaluated for decryption time with five different file sizes namely $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6,}$ with the size of 60,187,203,465,1586,3873 Kb that is shown in Table 5. The decryption time obtained for each file are 0.15000, 0.25800,0.30000,
0.89600,2.4600,7.13700sec. The following Figure 2 shows the details of decryption time.

The above Figure 4 shows the decryption time for different files for the proposed scheme. Decryption time is estimated based on decryption algorithm to convert the cipher text into the plain text.

The above Table 6 shows the comparison between the decryption time for proposed method with EPPDR, TEES for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6,}$ with the size of 160, 187, 203, 465, 1586, 3873 Kb. The results exposed that the proposed encryption technique attains better performance with an decryption time of 0.15000, 0.25800, 0.30000, 0.89600, 2.46000, 7.13700 sec for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ F_6.While the other like EPPDR, TEES attains 00.24000, 0.31700, 0.40300, 0.95300, 2.51000, 8.62000sec and 0.235, 0.2940, 0.4200, 0.9300, 2.1460, 8.1900 sec with different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6.}$

Same as encryption time Figure 5 shows the decryption time for each file. The decryption time for proposed method is compared with EPPDR, TEES. Also, it has proved that the proposed approach have better performance than others.

4.3.4. Throughput for encryption and decryption time

Throughput is equal to total plaintext in bytes encrypted divided by the encryption time. Higher the throughput, higher will be the performance.

$$Throughput = \frac{Total\ plain\ text}{Time}$$

By the value obtained from the encryption process, the throughput value is calculated for different file sizes. The above Table 7 shows the comparison between the throughput of encryption process for proposed method with EPPDR, TEES for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6,}$ with the size of 160, 187, 203, 465, 1586, 3873 Kb. The results exposed that the proposed technique attains higher with throughput value of 1600,1700, 1637.096774, 1945.606695, 2017.811705, 1821.73095 kb/sec for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ F_6.While the other like EPPDR,TEES attains 842.1053, 861.7512,

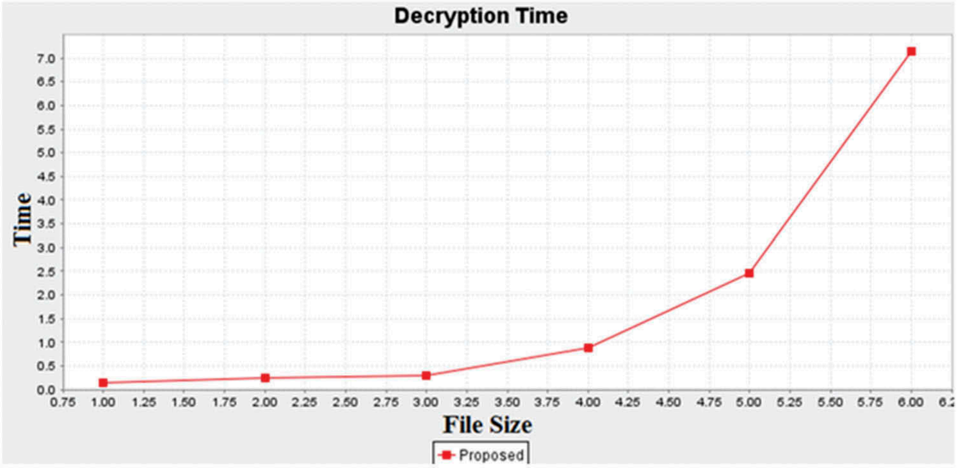

Figure 4. Decryption time graph for proposed scheme.

Table 6. Comparison decryption time.

File Name	Size (Kb)	Decryption Time(Sec)		
		Proposed	EPPDR	TEES
F1	160	0.15000	0.24000	0.235
F2	187	0.25800	0.31700	0.2940
F3	203	0.30000	0.40300	0.4200
F4	465	0.89600	0.95300	0.9300
F5	1586	2.46000	2.51000	2.1460
F6	3873	7.13700	8.62000	8.1900

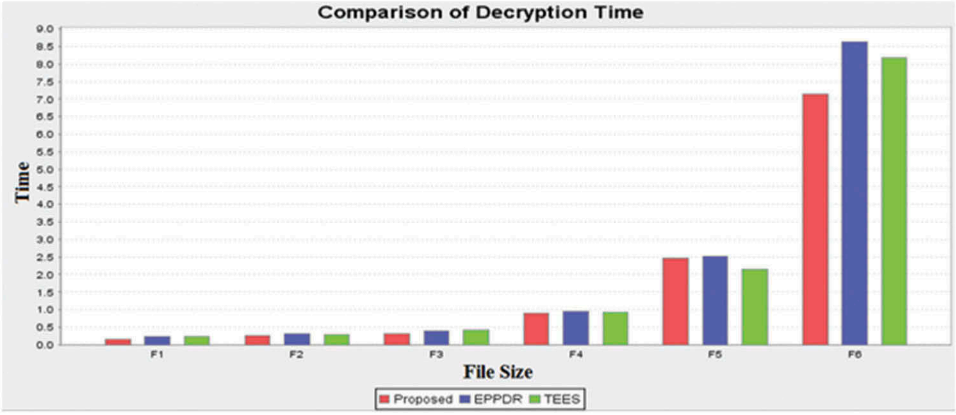

Figure 5. Comparison graph of decryption process.

812,1367.647, 1645.228, 1648.085 kb/sec and 864. 8649, 903.3816, 935.4839, 1529.605, 1948.403, 1795.549kb/sec with different file size F_1, F_2, F_3, F_4, F_5, F_6.

Table 7. Comparison of throughput encryption process.

	Throughput for Encryption Time		
File Name	proposed	EPPDR	TEES
F1	1600	842.1053	864.8649
F2	1700	861.7512	903.3816
F3	1637.096774	812	935.4839
F4	1945.606695	1367.647	1529.605
F5	2017.811705	1645.228	1948.403
F6	1821.73095	1648.085	1795.549

The above Figure 6 shows the throughput of encryption process and demonstrates the high throughput compared to the others scheme. The average throughput value for encryption process is 1787.041.

Same as throughput for encryption, the value of throughput for decryption process is calculated for different file sizes. The above Table 8 shows the comparison between the throughput of decryption process for proposed method with EPPDR, TEES for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ $F_{6,}$ with the size of 160, 187, 203, 465, 1586, 3873 Kb. The results exposed that the proposed technique attains higher with throughput value of 1066.666667, 724.8062016, 676.6666667, 518.9732143, 644.7154472, 542.6649853 kb/sec for different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ F_6.While the other like EPPDR, TEES attains 666.6667, 589.9054, 503.7221, 487.9328, 631.8725, 449.3039 kb/sec and 680.8511, 636.0544, 483.3333, 500, 739.0494, 472.8938 kb/sec with different file size $F_{1,}$ $F_{2,}$ $F_{3,}$ $F_{4,}$ $F_{5,}$ F_6.

Figure 7 shows the performance analysis of throughput for the process of decryption in EPPDR, TEES and the proposed optimized EECPKE. We can observe that the average throughput of decryption process of the proposed scheme is 695.749kb/sec.

By comparing the proposed scheme with EPPDR, TEES in their security from the attacks, encryption and decryption time,the proposed EECPKE scheme overtakes the others because of using dual authentication, the reliable algorithm which was used in

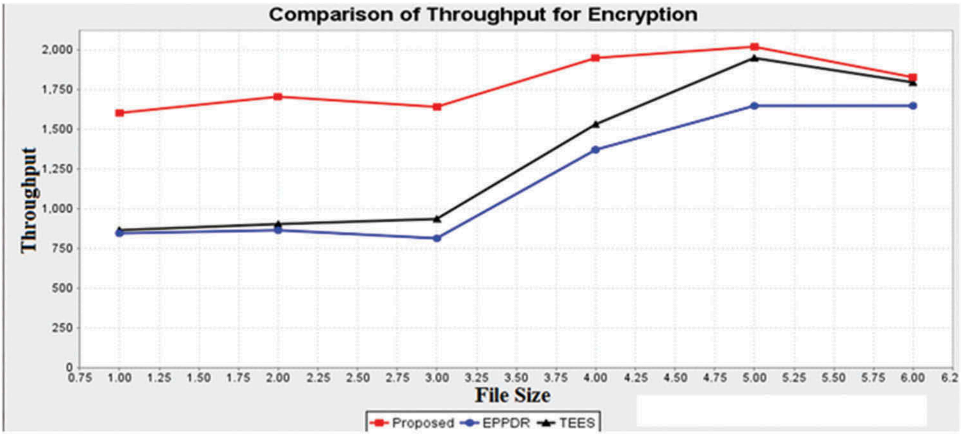

Figure 6. Throughput of encryption process.

Table 8. Comparison of throughput for decryption process.

	Throughput for Decryption Time		
File Name	Proposed	EPDR	TEES
F1	1066.666667	666.6667	680.8511
F2	724.8062016	589.9054	636.0544
F3	676.6666,667	503.7221	483.3333
F4	518.9732143	487.9328	500
F5	644.7154472	631.8725	739.0494
F6	542.6649853	449.3039	472.8938

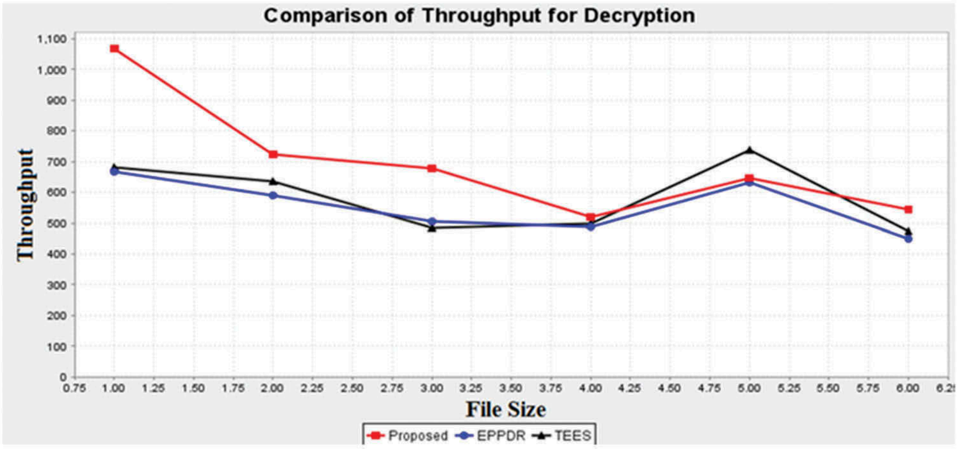

Figure 7. Throughput of decryption process.

the re-encryption process as well as recovering. The overall scheme optimized the overall time for computation. Hence it shows the proposed scheme works well compared to others.

5. Conclusion

The advances in cloud computing technologies makes possible to develop an integrated scheme which can seamlessly integrate the new technologies into existing enterprise information systems(EIS).The proposed DRM method of secured data sharing with Dynamic Unidirectional Proxy Re-Encryption and Cipher text Policy Attribute based Encryption techniques includes five phases such as authentication check, encryption, data integrity checking, user confirmation and data retrieval. Evaluating the efficiency of the proposed work based on optimized EECPKE with the most common existing schemes exhibits rapid encryption and decryption operation with high security from the attacks. The overall results for the proposed scheme of EECPKE are 1) an efficient method which protects the data files from the attackers, 2) the throughput for both encryption and decryption process is increased to33% than other techniques like EPPDR and TEES and 3) the overall time consumption for encryption and decryption process is reduced to 25%than other existing techniques like IBA, EPPDR, TEES etc.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Aldossary, S., and W. Allen. 2016. "Data Security, Privacy, Availability and Integrity in Cloud Computing: Issues and Current Solutions." *International Journal of Advanced Computer Science and Applications* 7 (4): 485–498. doi:10.14569/issn.2156-5570.

Ali, M., R. Dhamotharan, E. Khan, S. U. Khan, A. V. Vasilakos, K. Li, and A. Y. Zomaya. 2017. "SeDaSC: Secure Data Sharing in Clouds." *IEEE Systems Journal* 11 (2): 395–404. doi:10.1109/JSYST.2014.2379646.

Balu, A., and K. Kuppusamy. 2010. "Privacy Preserving Ciphertext Policy Attribute Based Encryption." In *International Conference on Network Security and Applications, 402–409. Springer, Berlin, Heidelberg.*

Blasco, J., J. E. Tapiador, P. Peris-Lopez, and G. Suarez-Tangil. 2015. "Hindering Data Theft with Encrypted Data Trees." *Journal of Systems and Software* 101: 147–158. doi:10.1016/j.jss.2014.11.050.

Catuogno, L., H. Löhr, M. Winandy, and A. R. Sadeghi. 2014. "A Trusted Versioning File System for Passive Mobile Storage Devices." *Journal of Network and Computer Applications* 38: 65–75. doi:10.1016/j.jnca.2013.05.006.

Chang, S. Y., and Y. C. Hu. 2017. "Secure MAC: Securing Wireless Medium Access Control against Insider Denial-of-Service Attacks." *IEEE Transactions on Mobile Computing*. doi:10.1109/TMC.2017.2693990.

Choi, D., H. K. Choi, and S. Y. Lee. 2015. "A Group-Based Security Protocol for Machine-Type Communications in LTE-advanced." *Wireless Networks* 21 (2): 405–419. doi:10.1007/s11276-014-0788-9.

Da Xu, L. 2011. "Enterprise Systems: State-Of-The-Art and Future Trends." *IEEE Transactions on Industrial Informatics* 7 (4): 630–640. doi:10.1109/TII.2011.2167156.

Fabian, B., T. Ermakova, and P. Junghanns. 2015. "Collaborative and Secure Sharing of Healthcare Data in Multi-Clouds." *Information Systems* 48: 132–150. doi:10.1016/j.is.2014.05.004.

Hur, J., D. Koo, Y. Shin, and K. Kang. 2016. "Secure Data Deduplication with Dynamic Ownership Management in Cloud Storage." *IEEE Transactions on Knowledge and Data Engineering* 28 (11): 3113–3125. doi:10.1109/TKDE.2016.2580139.

Kumari, S., and M. K. Khan. 2014. "More Secure Smart Card-Based Remote User Password Authentication Scheme with User Anonymity." *Security and Communication Networks* 7 (11): 2039–2053. doi:10.1002/sec.v7.11.

Li, J., M. Ruhui, and H. Guan. 2014. "Tees: An Efficient Search Scheme over Encrypted Data on Mobile Cloud." *IEEE Transactions on Cloud Computing* 5 (1): 126–139. doi:10.1109/TCC.2015.2398426.

Li, S., L. Xu, X. Wang, and J. Wang. 2012. "Integration of Hybrid Wireless Networks in Cloud Services Oriented Enterprise Information Systems." *Enterprise Information Systems* 6 (2): 165–187. doi:10.1080/17517575.2011.654266.

Li, Yibin., K. Gai., L. Qi., M. Qiu., and H. Zhao. 2017. "Intelligent Cryptography Approach for Secure Distributed Big Data Storage in Cloud Computing". *Information Sciences* 387: 103–115. doi:10.1016/j.ins.2016.09.005.

Liu, Z., Z. Cao, and D. S. Wong. 2013. "White-Box Traceable Ciphertext-Policy Attribute-Based Encryption Supporting Any Monotone Access Structures." *IEEE Transactions on Information Forensics and Security* 8 (1): 76–88. doi:10.1109/TIFS.2012.2223683.

Lu, Y., and J. Li. 2016. "A Pairing-Free Certificate-Based Proxy Re-Encryption Scheme for Secure Data Sharing in Public Clouds." *Future Generation Computer Systems* 62: 140–147. doi:10.1016/j.future.2015.11.012.

Mapoka, T. T., S. J. Shepherd, and R. A. Abd-Alhameed. 2015. "A New Multiple Service Key Management Scheme for Secure Wireless Mobile Multicast." *IEEE Transactions on Mobile Computing* 14 (8): 1545–1559. doi:10.1109/TMC.2014.2362760.

Patil, K., S. D. Khatawkar, and A. Dange. “Secure Data Sharing in Cloud through Limiting Trust in Third Party/Server.” *International Research Journal of Engineering and Technology (IRJET)* 4: 1101–1106.

Rafeeq, M. D., and C. S. Kumar. 2015. “Reliable Secure Data Storage in the Cloud Environments and De Duplication.” *International Journal of Computer Science and Engineering* 3 (3): 1086–1091.

Samanthula, B. K., Y. Elmehdwi, G. Howser, and S. Madria. 2015. “A Secure Data Sharing and Query Processing Framework via Federation of Cloud Computing.” *Information Systems* 48: 196–212. doi:10.1016/j.is.2013.08.004.

Sandhu, G. K., and E. G. Bhathal. 2016. “To Enhance the OTP Generation Process for Cloud Data Security Using Diffie-Hellman and HMA.” *Global Journal of Computer Science and Technology* 6: 2.

Shao, J., R. Lu, X. Lin, and K. Liang. 2016. “Secure Bidirectional Proxy Re-Encryption for Cryptographic Cloud Storage.” *Pervasive and Mobile Computing* 28: 113–121. doi:10.1016/j.pmcj.2015.06.016.

Singh, S., Y. S. Jeong, and J. H. Park. 2016. “A Survey on Cloud Computing Security: Issues, Threats, and Solutions.” *Journal of Network and Computer Applications* 75: 200–222. doi:10.1016/j.jnca.2016.09.002.

Sood, S. K. 2012. “Combined Approach to Ensure Data Security in Cloud Computing.” *Journal of Network and Computer Applications* 35 (6): 1831–1838. doi:10.1016/j.jnca.2012.07.007.

Spandana, B., and D. R. P Sunitha. 2016. “Focusing in the Security Constraints of Cloud Computing.” *International Journal of Advanced Technology and Innovative Research* 08: 3600–3604.

Taylor, L. 2017. “Safety in Numbers? Group Privacy and Big Data Analytics in the Developing World.” In *Group Privacy*, 13–36. Springer, Cham.

Tung, Y. H., S. S. Tseng, and Y. Y. Kuo. 2015. “A Testing-Based Approach to SLA Evaluation on Cloud Environment.” In: *Network Operations and Management Symposium (APNOMS) 2015 17th Asia-Pacific*, 495–498. IEEE.

Uddin, M., J. Memon, R. Alsaqour, A. Shah, and M. Z. Rozan. 2015. “Mobile Agent Based Multi-Layer Security Framework for Cloud Data Centers.” *Indian Journal of Science and Technology* 8: 12. doi:10.17485/ijst/2015/v8i12/52923.

Wang, L. L., K. F. Chen, X. P. Mao, and Y. T. Wang. 2014. “Efficient and Provably-Secure Certificate Less Proxy Re-Encryption Scheme for Secure Cloud Data Sharing.” *Journal of Shanghai Jiao Tong University (Science)* 19: 398–405. doi:10.1007/s12204-014-1514-6.

Yong, Y., N. Jianbing, M. H. Au, H. Liu, H. Wang, and C. Xu. 2014. “Improved Security of a Dynamic Remote Data Possession Checking Protocol for Cloud Storage.” *Expert Systems with Applications* 41 (17): 7789–7796. doi:10.1016/j.eswa.2014.06.027.

Zhou, Z., D. Huang, and Z. Wang. 2015. “Efficient Privacy-Preserving Cipher Text-Policy Attribute Based-Encryption and Broadcast Encryption.” *IEEE Transactions on Computers* 64 (1): 126–138. doi:10.1109/TC.2013.200.