International Journal of Information Security and Privacy
Volume 14 • Issue 1 • January-March 2020

Secured Sharing of Data in Cloud via Dual Authentication, Dynamic Unidirectional PRE, and CPABE

Neha Agarwal, Amity University, Uttar Pradesh, India

Ajay Rana, Amity University, Uttar Pradesh, India

J.P. Pandey, KNIT, Delhi, India

Amit Agarwal, University of Petroleum and Energy Studies, Dehradun, India

https://orcid.org/0000-0002-3933-5014

ABSTRACT

Cloud computing is an emergent computing paradigm; however, data security is a significant issue in recent time while outsourcing the data to the cloud preventing users to upload their data on cloud. The data forwarded to cloud can be protected using some cryptographic techniques based on identity, attributes, and prediction. But these algorithms lack their performance when a revoked user collude with cloud; therefore, it becomes essential to develop a secure data sharing framework with some enhanced cryptographic techniques. The proposed methodology presented a secure privacy preserving data sharing with encryption technique called dynamic unidirectional proxy re-encryption (PRE) with cipher text policy attribute-based encryption. The technique ensures the privacy, integrity, and security of the data while retrieving through the cloud. The framework is implemented in the cloud sim with java language. Experimental results proved that proposed frame work attains reasonable results compared to traditional methods.

KEYWORDS

CipherText Policy ABE, Digital Rights Management (DRM), Efficient Elliptic Curve Public Key Encryption (EECPKE), Proxy Re-encryption, Twofold authentication protocol

1. INTRODUCTION

Cloud computing is an emerging paradigm in which resources are outsourced on rent to the customers by the cloud service providers through internet. It is now acknowledged as utility service after electrical, water and gas services(Ali et al. 2015). It not only saves the capital expenditure of the customer but he can scale out or scale in the request for services provided and pay accordingly. It is not limited for storing and sharing data but is also for managing, monitoring and exploring data in space ground data system (Kaddouri et al., 2018). The four main deployment models are public, private, community and hybrid cloud having variations in cost and security. In cloud stack the services are arranged as layers from the most reduced layer to highest layer where each layer symbolizes one service model. IaaS is the most reduced layer, where the cloud supplier maintains a suite of management resources and services to cope a substantial cloud system (Zhu et al., 2013; Sun et al., 2014) and the user utilizes the infrastructure and resources such as network, storage, computational capacity etc without worrying

DOI: 10.4018/IJISP.2020010104

about the complexity and management (Wang et al., 2012; Wei et al. 2014). The central layer PaaS, offers platform and software to develop applications. SaaS located at top layer, where completely developed software applications are provided as a service (Saouli et al., 2015).

For sharing the data, the cloud model comprises of three entities cloud service supplier, client, owner (Boyang Wang et al. 2015). Cloud service supplier regulates Cloud Storage Server (CSS) which has bigger storage space to shield the clients data and in addition high computation control (Manvi and Shyam 2014). Cloud servers gives a novel service approach where information is stored and its replica is maintained so that information can be acquired by clients anytime and from anyplace over the network (Sood 2012). Owner has colossal information documents for sharing and for this he uploads his data in cloud. The client are authorised by the data owner who can access the shared data. It can be a cloud proprietor itself too (Patel et al. 2013; Rong et al. 2013).

Although Cloud can confirm the client's information security through the thought of firewalls, fundamental private networks and by executing other security policies with in its own particular limits (Bera et al. 2015) yet Security is the most important key concern not only for data at transit but also for data at storage (Yang and Jia, 2012)

While outsourcing the sensitive data to be shared on cloud the owner losses his physical control. The data can be stored anywhere in the cloud as a result it becomes difficult to confirm exact location of storage (Li et al. 2015). The data not only have traditional security risks like (Ahmed et al.,2017), DDOS Attack (Li et al.,2015; Jeyanthi et al., 2013), man in middle attack and several intruder attacks (Boukhlouf et al.,2016) etc but even the third party service provider are semi trustful. As a result the owner needs to ensure the confidentiality, security from intruders, privacy, data availability and accessibility to users according to their access rights (AlZain et al. 2012;Zissis et al. 2012; Jakimoski, 2016).

The most common way to maintain confidentiality and security of the data stored in cloud against semi trusted cloud service provider is to send encrypted data. However there may be several other issues such as preventing the user to access the data for which he is not authorized, preventing the collusion between the revoked user and the semi trustful cloud, revoking away the given access right of the authorized user without re-encrypting the content and redistributing the new keys to the authorized users.

The main features needed while outsourcing data in cloud are privacy, Integrity, confidentiality, fine grained access control, Successful revoking privileges from users without in need of regeneration and distribution of re-encryption key, preventing collusion between third party service provider and revoked clients, Successful joining of new users and rejoining of revoked ones.

To ensure the afore mentioned features, In this paper we have proposed a framework based on DRM mechanism to ensure data security, dynamic authorization, license creation and proxy re-encryption. The proposed DRM scheme involves Ciphertext Policy attribute based encryption to ensure confidentiality of user and proxy re-encryption for preventing revoked user to collude along with One time password and license and other security mechanism. Finally we have compared the encryption and decryption time of proposed scheme with RSA and proved that the proposed scheme take less time in comparison to RSA

The outline of this paper is summarized as follows. In section 2 we have review the existing literature, In section 3 we have presented our framework entitle "Dual Authentication Based Security Framework for Cloud Based Data Sharing Applications" and presented the algorithmic description of proposed approach. Thereafter section 4 represents the experimental result and analysis of our approach and its comparison with the existing approach. Finally in section 5 we have drawn conclusion and presented the future scope.

2. RELATED RESEARCH

Many traditional encryption schemes are available for sharing data through cloud. However they do not provde fine grained access control nor they prevent the data after the collusion between the revoked user and less trustworthy cloud. In our approach we have proposed dual authentication and fine grained access control along with successful revocation of user without in need of redistributing the keys. The owner encrypts data under his public key and send it to third party honest but curious service provider. Along with encrypted data owner also send the access control list specifying the authorization for accessing the attributes corresponding to users to cloud as well as to OTP server which generates one time token for requesting user if he belongs to authorised list . The cloud service provider converts the cipher text under the public key of one authorized user to under the public key of another authorized user and provide it to recipient for decryption. In this way data is securely shared among authorized users using concept called Fine Grained Access Control. Some of the recent research papers which are relevant to sharing data in cloud computing is listed below:-

2.1. Proxy Reencryption

Proxy Re-encryption is public key encryption that allows proxy server to convert ciphertext generated under the public key of sender(owner) to ciphertext under the public key of recipient without proxy being able to know about original message. In this way it helps in sharing data in secured way, it also helps in delegating decryption rights using enforced access controlled mechanism(Shao,2015).

Under this scheme the owner of data outsources the encrypted content in the cloud which is partially reliable along with re-encryption key. The proxy server in turn enforce the access control mechanism through re-encryption process along with maintaining confidentiality of the data from unauthorised users and proxy server itself.

In this scheme there can be set of encryption decryption algorithms defined over set of cipher text spaces. The proxy re-encryption scheme defined over single space are discussed in (Blaze et al. 1998; Canetti and Hohenberg 2007; Xagawa and Tanaka 2010; Aono at al. 2013;Kirshanova 2014;Nunez et al 2015), the scheme with two cipher text spaces are discussed in (Ateniese and Fu 2006; Chu and Tzeng 2007; Deng et al. 2008) some of the schemes are defined over infinite cipher text spaces such as in (Green and Ateniese, 2007).

The main features of proxy re-encryption are directionability, the property is embodied in re-encryption key and specifies about the direction of delegation in case of unidirectional the Sender only delegates the decryption rights to recipient whereas in bidirectional the symmetric trust relationship is established between the sender and recipient. Number of uses, in PRE scheme if cipher text is re-encryptable just once then it is said to be single use however if ciphertext is re-encryptable multiple times then its said to be multiuse. Collusion Safeness, the property states about security of private key of sender in case of collusion between unauthorised recipient and semi trusted proxy server. Transitive, It is said to be transitive if proxy server(cloud) is able to re-delegate the access rights on its own. Interactivity, the PRE is said to be interactive if the sender is able to generate re-encryption key of recipient using his own private key and recipients public key. Other properties of proxy re-encryption are conditional, temporary, Non transferability, proxy invisibility and perfect key switching(Nunez et al., 2016). Due to these properties it is widely used in many emerging areas like encrypted database ZeroDB(Egorov and Wilkison, 2016) and EU H2020 research project CREDENTIAL(Horander et al.,2016) for sharing access key between authorised users.

(Blaze et al. 1998) introduced proxy re-encryption scheme for having bidirectional and multiuse pattern. The scheme is based on Elgamal. Later (Xagawa and Tanaka, 2010 ;Nunez et al., 2015) also represented the same procedure but used lattice based settings. Since it use bidirectional pattern its transitive and interactive but is not collusion resistant. (Canetti and Hohenberger,2007) presents first CCA secure bidirectional scheme by integrating cipher text with one time signature, Its interactive, transitive, but not resistant to collusion.

(Weng et al., 2010) proposed another efficient bidirectional scheme which is not based on bilinear pairing, the scheme has achieved CCA security by integrating PRE version of hashed Elgamal encryption scheme and schnorr signature. It is interactive but not resistant to collusion and is defined for single use as well as multiuse.

(Ateniese et al, 2009) presented bilinear pairing based first unidirectional proxy re-encryption scheme which is collusion resistant, single use, non-transitive, non-interactive and proxy invisible. (Libert and Vergnaud, 2011) presented unidirectional scheme with RCCA security in which he integrated one time signatures. It is unidirectional, single-use, collusion-resistant, but neither interactive nor transitive. However (Seo et al.,2012) detected an error in its security proofs

(Wang et al. 2014) clarified intermediary certificate less proxy re-encryption (CL-PRE) scheme created without pairing. It deals with the key escrow issue in character based public key cryptography. The security of the approach is similar to the security provided . The scheme has same security issue as computational Diffie-Hellman (CDH) issue in the oracle model.

(Lu and Li 2016) introduced the Certificate-based intermediary re-encryption strategy without bilinear pairing. The approach ensures confidentiality and security under computational Diffie-Hellman(Kumar et al., 2016) supposition and is computationally efficient so can be used with power constrained devices.

(Xu et al., 2014) proposed fine-grained and heterogeneous intermediary re-encryption (FHPRE) approach which enables sharing data securely between two heterogenous cloud pursuing different cryptographic techniques

(Liang et al. 2015) introduced the strategy for CP-ABPRE with the integration of double system encryption technology with selective verification technique to enhance the efficiency The CP-ABPRE has much application in network communication for instance information sharing.

(Aono et al., 2013) presented lattice based unidirectional proxy re-encryption scheme it is interactive, non transitive and limited multi use but is not collusion resistant. It was also ensured that scheme is key private but then it was challenged and proved wrong by (Nishimaki and xagawa, 2015 ;Nunez et al., 2016). Further (Kirshanova, 2014) proposed lattice-based PRE scheme by extending the original PKE scheme proposed by (Micciancio and piekert, 2012) to support re-encryptions, using trapdoor delegation. The scheme is single hop, unidirectional, not interactive and resistant to collusions The scheme is single use however (Nunez˜ et al.,2015) described an attack proving that it does not satisfy CCA security, recently (Fan and Liu,2016) has also proved error in security proof of the scheme proposed and extended the work of Kirshanova for constructing multi hop re-encryption.

2.2. Cipher Text Policy Attribute-based Encryption

For ensuring the confidentiality and security of the data being shared (Sahai and waters, 2005) presented Attribute-based encryption(ABE) based on the concept of public-key cryptography. (Goyal et al., 2006) further classify ABE as KP-ABE(Key policy Attribute based encryption) and CP-ABE(Ciphertext Policy attribute based encryption). In KP-ABE secret key is associated to access structure and ciphertext is associated with attribute set as the result the main issue with KP-ABE is owner cannot take a decision on who can decrypt the data. (Bethencourt et al.2007) was the first to propose CP-ABE in which the ciphertext is associated with access structure and secret key is associated with attribute set as a result the owner can define the access policy which has to be satisfied by the recipient for decrypting the data.

Liu Z et al. 2013 has 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.

Tamizharasi GS et al. 2016 has discussed about the issue of client attribute management in cloud computing utilizing the strategy for point-multipoint - Policy Attribute based Encryption (CPABE). In cloud computing the client attribute management was a complex process if the number of information clients increases in correspondence to the number of client attributes. In this way the information clients are assembled under the premise of their common attributes and put away those in client attribute relationship table on the cloud server for simpler cloud management. This system uses five algorithms such as Setup, Grouped, Encrypt, Key Gen and Decrypt.

Due to invariant feature CP-ABE is widely use to share data securely in cloud. Many schemes has been proposed in recent years for improving CP-ABE most of which support either constant size secret key (Guo et al.,2014) or constant size ciphertexts(Zhang et al., 2014 ; Doshi et al., 2014) however few supports both constant size ciphertext as well constant size secret key(Odelu et al., 2017).

2.3. Hybrid Approach for Secure Data Sharing in Cloud

(Yu et al., 2010) proposed a model for secured way of sharing data in cloud using fine grained access control by combining KP-ABE(Key policy Cipher Text Attribute Based Encryption), Proxy re-encryption and lazy re-encryption.. When the data owner wants to revoke certain users from accessing data new key is provided to all authorised users and the encryption has to be performed again. However the model has a flaw as the cloud has to be stateful retaining the history of revoked users also revoked user and the cloud can collude together.

(Park et al.,2011) has provided modification to (Yu et al,2010) as the revoked user and cloud do not collide however he assumed that proxy server is more trust worthy and so replaced cloud with trusted third party provider.

(Yang et al., 2011) proposed a generic solution in which the cloud could be stateless and need not have to maintain list of revoked users. Under this approach the data key is divide into two parts. One part is encrypted using ABE and other part is encrypted using PRE. ABE provides fine grained access control and PRE makes revocation possible, the presence of re-encryption key corresponding to a recipient in cloud states that he is authorized to access data. However the proposal failed when the revoked user rejoins. Also there could be collusion between the revoked user and cloud due to loose coupling between ABE and PRE

(Wang et al., 2011;Hur 2013;Tyoswski and Hasan 2013) have also proposed approaches for secured data sharing in cloud by integrating CP-ABE and proxy re-encryption.

(Xiong et al., 2012) presented a scheme called CloudSeal for flexible access control mechanism by integrating PRE, broad case revocation scheme, symmetric encryption and secret sharing mechanism in cloud. The presented scheme provides end to end scalability, confidentiality and efficiency however the main limitation is It assumes existence of secure channel between the owner and recipient.

(Lin et al., 2012) proposed an approach of combining threshold encryption with proxy re-encryption. Under this approach the proxy server is divided into several servers. Each and every server stores the share of a private key of a owner and so do partial decryption of data. When the recipient sends a request to the server the randomly selected subset of servers re-encrypt the data. The recipient combines all the partially decrypted data to obtain the requested data.

(Liu et al., 2014) proposed another approach by combining ABE and PRE where ABE describes time based access control policies and PRE updates the time attributes.

(Samanthula et al., 2015) proposed a framework in which he has federated the cloud into two and used additive homomorphic encryption and proxy re-encryption for sharing the data securely. According to his framework if the authorised user colludes with revoked user then the revoked user will be able to get information available to authorised user only. For collusion between revoked user and cloud he has assumed that user can collude with only one of the two clouds.

3. DUAL AUTHENTICATION BASED SECURITY FRAMEWORK FOR CLOUD BASED DATA SHARING APPLICATIONS

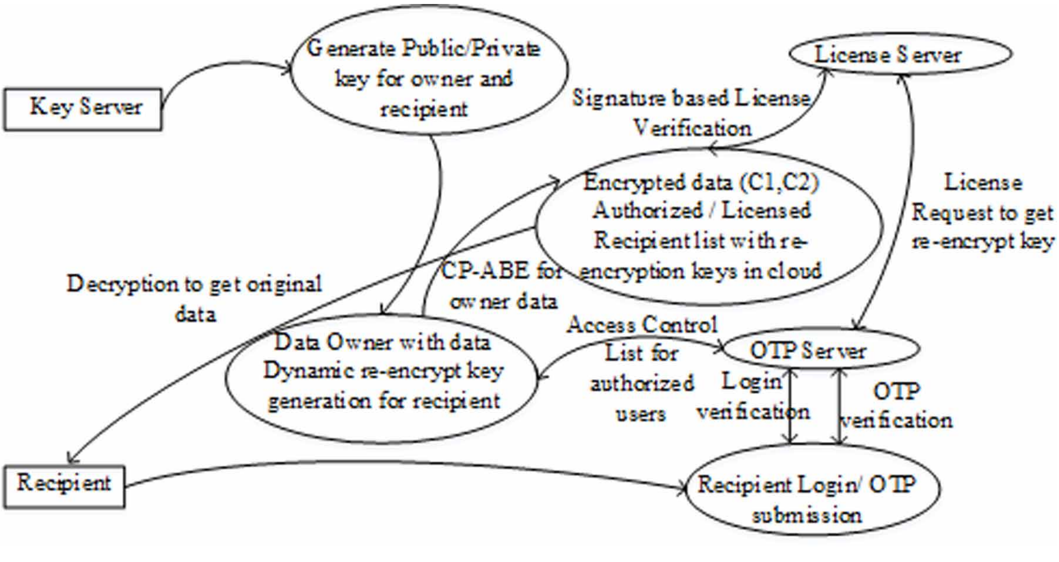
Cloud computing is a recent developing paradigm which is drawing attention due to availability of unlimited infrastructure and resources. It is being widely considered by the organizations to store and share the data however while adopting cloud for sharing file the major issue is security of data during transit and at rest while sharing. Earlier for validating the user the license was represented as alphanumerical serial number which was inserted for launching data. If the entered key matches to license the user got validated to continue however today the product keys are associated with the online activation keys but then its computation cost is very high. To overcome the computation cost we have defined a secured frame work which has dual authentication scheme in which user is identified by his identification and mutual authentication. Further for successful revocation and strong integrity checking dynamic single hop unidirectional proxy re-encryption is used. The Figure 1 below describes the proposed process.

3.1. Problem Formulation

The data is being widely shared in cloud as it can be rapidly and easily access from anywhere across the globe. But at the same time the data kept in cloud is vulnerable to several kinds of attacks which hinders the privacy and security of data. As a result ensuring security of shared data is most challenging and recent area of research. When the owner stores his data in cloud he wants the data to be shared with authentic and authorised users who are listed by him In other words the owner wants to maintain the integrity, confidentiality and security of data being shared. Many cryptographic techniques are available now a days but they lack in their performance in one or the other way. Hence a secured framework needs to be developed based on modern cryptographic technique which ensures the security and privacy of data along with automating the system which lets the minimal intervention of the owner and let him free after uploading the encrypted data. The process of proposed framework is explained below

First the user login and proves himself as an authentic user. Dual authentication protocol is used for validating the user to the cloud as an authorised user. After verification he can upload or access the data stored in cloud. However the file or data is first encrypted before storing in cloud and then the

Figure 1. Data flow diagram of proposed EECPKE security mechanism


Key Server
Generate Public/Private key for owner and recipient
License Server
Signature based License Verification
Encrypted data (C1,C2) Authorized / Licensed Recipient list with re-encryption keys in cloud
License Request to get re-encrypt key
Decryption to get original data
CP-ABE for owner data
Access Control List for authorized users
Data Owner with data Dynamic re-encrypt key generation for recipient
OTP Server
Login verification
OTP verification
Recipient
Recipient Login/ OTP submission

owner who has uploaded the data creates a re-encryption key of the authorised list of users and end it to cloud. When any recipient wants to have an access of the data first he is verified by the license and then in cloud if the re-encryption key for him exist he is verified. The cloud then re-encrypts and transforms it under the public of recipient from public key of sender. The main target of the proposed approach is to ensure the security privacy of data or file being shared, to have fine grained access control and to have successful revocation of user without in need of redistributing the keys and to prevent the data from collusion between the cloud and revoked user.

3.2. Cloud User Authentication via Dual Authentication Protocol (DAP)

When the user logs in he is validated with dual authentication protocol comprising of two phases. For reliable authentication in first phase registration protocol is executed in which user enters user id and password. And in second phase the OTP is generated which is explained below.

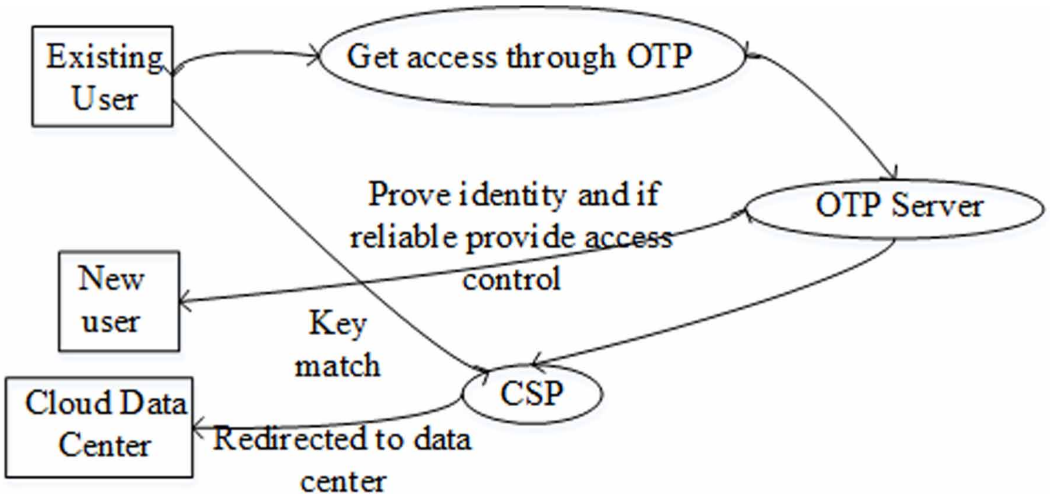
3.2.1 OTP Server for Authentication

OTP server is used as dynamic ticket granting center for the authentic users. For enhancing the reliability of authentication the user is provided with One time token from OTP server. Initially for the first time when the user registers he provide his identity to OTP server. The OTP on receiving the identity of user checks for trust value of the user from the access control list of authorised users provided to it from the owner. If in OTP server the user is found in list, one time token is sent to him. This OTP can be concatenated with IP address. The same OTP is shared with CSP also. When the user wants to access file in cloud he send the OTP received from OTP server to CSP. The CSP checks and matches the OTP received from the user with the OTP received for him from OTP server if the two OTPs matches then the user is allowed to access file in cloud. This is explained in Figure 2.

3.2.2 Login Phase

The login phase initiates the authentication process. In this phase user C_i .nters his identity and password which is converted into ASCII value $ASCII_{UN}$ .Estimates $LV_i = \left[ASCII_{UN}, PW_i\right]$ .nd verifies if LV_i .matches to the document saved in cloud. If LV_i .equals to store RA_i .hen IP address of public cloud service provider is validated, if there is any flaw then session is dismissed else the estimation at authority login page continues

Figure 2. Data flow diagram describing authentication



Computes $OP_i = h_{fc}\left(PW_i \oplus T_c\right)$. where, OP_i .is the one-time password for the estimating variable and T_c .epresents the current for the timestamp.

The public cloud service provider CSP_i .ransmits one time password OTP to the user C_i .hrough RA_i .or additional security over a estimated timestamp. Estimates $S_m = \left[IP_i, OP_i, T_c\right]$. The user C_i .ispatches the appropriate OTP as input and transmits the information to the public cloud; here the channel of the network is unsecure.

Estimates $\left(T_s - T_c\right) > \Delta T_c$. the public cloud service provider CSP_i .orrelate the shape of the time stamp 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.

If the actual time stamp 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.

3.3. Cipher Text Policy Attribute-based Encryption

Once the user is verified as authentic user using dual authentication protocol he can access and send reuest to upload or access file. On receiving request for accessing file he is further guarded to License server. License server checks for the access rights of the requesting user and send license to CSP for the user. At cloud the complete file is re-encrypted under the public key of client who is a requesting user from pubic key of sender and then the re-encrypted file is sent to the requesting user who decrypts it using his secret key.

A. Setup: The system setup phase takes security parameter Par .and generates secret key and master key. It selects a bilinear group $I : \Omega \to \mathfrak{R}$ .of prime order *p* with *q* as generator, and bilinear map Ω .he universe attribute is $c_1, \ldots, c_N$. It selects $\Omega_1, \ldots, \Omega_N$ .or attribute *n*, $\left\{\Omega_i\right\}_{i=1}^{N}$. and a random exponent $\Omega = \bigcup_{i=1}^{N} \Omega_i$. The public key $\Omega_i \cap \Omega_j = 0$ .nd master key $i \neq j$ .s given by

$$\left\{\Omega_i\right\}_{i=1}^{N}. \tag{1}$$

$$\left\{c_i\right\}_{i=1}^{N}. \tag{2}$$

Though A_i .s publicly known to all system parties, $w_i\left(x\right)$ .is kept secretly by trusted authority (T_A).

Algorithm for Setup

Input: i_{th} .Outputs: $I\left(x\right) = \sum_{i=1}^{N} w_i\left(x\right) A_i\left(x\right)$

```
Begin
Select a bilinear group $p_{i,j} = \int \Omega w_i\left(x\right).w_j\left(x\right)dx \,/\, \int \Omega w_i\left(x\right)dx$ .and bilinear map $\Delta A_i$.
if(i>=1) then
 $A_i$ .s selected
End if
```

```
6. Run the setup using $T_A$
7. **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 $\frac{\partial\left(x,y,t\right)}{\partial t} = div\left[g\left(\nabla I\left(x,y,t\right)\right)\nabla I\left(x,y,t\right)\right]$. Then, it calculates the key as

$$I\left(x,y,0\right) \tag{3}$$

Algorithm for key generation
Input: S
Output: $I\left(x,y,t\right)$

```
1. **Begin**
2. Select random number from $g\left(\bullet\right)$
3. Compute the output
4. **End**
```

C. Encryption: The encryption phase takes access control tree *T* as input, a public key $Lim_{x\to\infty} g\left(x\right), g\left(x\right) = 1$. and message *M*. The output of cipher text is as follows. Initially, it describes the random polynomial $Lim_{x\to\infty} g\left(x\right), g\left(x\right) = 0$ .for every node *y* of tree *T* in top down approach beginning from *r*. The node *y* with degree $g_1\left(x\right)$ .s smaller than threshold value $g_1\left(x\right)$ .f that node, that is $g_1\left(x\right) = \exp\left[-\left(\frac{x}{K}\right)^2\right]$. For *r*, $g_2\left(x\right) = \exp\left[-\frac{1}{1+\left(\frac{x}{K}\right)^2}\right]$. that is a random number of $\varphi\left(x\right) = g\left(x\right) * x$. For each and every non-root node *y*, $\varphi\left(x\right)$ .here *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

$$\nabla I = K \tag{4}$$

Algorithm for Encryption
Input: T, $I_t\left(S\right) = I_t\left(S\right) + \lambda/\eta_s \sum_{p\in\eta_s} g_x\left(\left|\nabla I_{s,p}\right|\nabla I_{s,p}\right)$. M
Output: $S = K\sqrt{2}$

```
1. **Begin**
2. Represent a random polynomial $\lambda \in \left(0,1\right)$.
3. **if** (node == root node) **then**
4.     $\eta_s$.
5. **End if**
```

```
6.  **if** (node == non-root node) **then**
7.    $\eta_s = \left[N, S, E, W\right]$.
8.  **End if**
9.  Calculate $\left|\eta_s\right|$.
10. **End**
```

D. Dynamic Unidirectional Proxy Re-Encryption

A single hop uniddirection proxy re-encryption converts the data or file under the public key of one user say owner to public key of another user say reciepient. It comprises of following steps as in (Liang et al., 2015)

Step 1 –Set-up (K) → P_{par}: This phase is executed by trsuted party say owner where a security parameter is taken as an input and the output is a public parameter P_{par} which is a default parameter in remaining all steps.

Step 2- Key generation (K, P_{par}) → (S_k, P_k): Under this phase the public and private key of a user is generated using public parameter P_{par} and security parameter K

Step 3- Dynamic Rekey generation(P_{par}, dS_{ki}, P_{kj}, M' .) → Rij: For a recipient j the owner i generates a reenencrption key R_{ij} using $P_{par,}$ dynamic secret key dS_{ki} of user I, Keyword set M' .nd public key P_{kj} of user j.Using this key the second level ciphertext is re-encrypted from public key of owner to public key of recipient. Since the phase has not involved secret key dS_{kj} of the recipient. The algorithm is unidirectional.

In keyword-set $M' = \left(m_1^{'}, m_2^{'}, ...m_k^{'}\right)$ .for k<n the keywords can be concatenated using "AND" gates.

Step 4- Encryption1 (P_{par},P_i, m) → C: Using probabilistic algorithm and parameters(public parameter P_{par}, public key of sender $P_{i,}$ and message m) first level cipher text is generated which cannot be further re-encrypted.

Step 5- Encryption2 (P_{par},P_k, P) → C: Using randomize algorithm and parameters(public parameter P_{par}, public key of sender $P_{i,}$ and message m) second level cipher text is generated which can be recrypted from public key of owner to public key of recipient,

Step 6- ReEncryption1 (E_i, P_{par}, R_{ij}, C) → C_1: This phase takes as input as P_{par}, a re-encryption key $R_{ij,}$ a second level cipher text C encrypted under public key of user i' and E_i for Re-encryption. Here E_i is a dynamic key set generated for different user and generates a first level ciphertext C_1 which under the public key of recipient.

Step 7- Decryption1 (P_{par}, S_i, C) → P: the algorithm takes as input the public parameter P_{par}, the seceret key S_i of user I and first level ciphertext C and gives an outputs either a plaintext P or a 'invalid' message.

Step 8- Decryption2 (P_{par}, S_k, C_1) → P: the algorithm takes as input the public parameter P_{par}, the seceret key S_i of user k and second level ciphertext C_1 and gives an outputs either a plaintext P or a 'invalid' message.

E. Decryption Phase

The decryption phase takes cipher text $\nabla I_{s,p} = I_t\left(p\right) - I_t\left(s\right), p \in \eta_s = \left\{N, S, E, W\right\}$. public parameter P_{par} and the secret key S_K . It initially defines whether the attribute sets fulfils access structure *T* in $TS = \{\left(\nabla Is_i, p_j\right) \mid j = 1,\ldots\ldots, N \wedge i \in \left[1, M\right]\}$. If the node *y* is leaf node, it is indicated as p_j. and calculate

$$x_1, x_2, x_3, \ldots\ldots., x_d . \tag{20}$$

If $y_1, y_2, y_3, \ldots.., y_m$. Else, *Decrypt Node* $n \le N$. Then, by utilizing polynomial interpolation approach, the pairing results in bottom-up manner were grouped and finally the blind factor $\left(S_1 \ldots\ldots\ldots. S_m\right)$.The phase now decrypts by calculating $m > 1$.The following algorithm deals with the decryption of the message (L, C, and T). The cloud user carry 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 $\left(f, V, p, q, u, h\right)$. private key s_k. Cipher text $\left(L, C, T\right)$.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

```
1.  Begin
2.  Perform validation for embedded public key $L$.
3.  $if$ .(Validation fails) then
4.  Result ("Reject the cipher text)
5.  End if
6.  Calculate $Y = h.\left(k.\left(s_k\right)L\right)$.7. $ifY == \infty$ .hen
    8.  Result ("Reject the cipher text")
    9.  End if
    10. $\left(k1, k2\right) \leftarrow KDF\left(xY, L\right)$.here xy is the x-coordinate of Y
    11. Calculate $T_1 = MAC\left(K2\left(C\right)\right)$.
    12. $if\left(T_1 \neq T\right) then$
    13. Result ("{reject the cipher text")
    14. End if
    15. Calculate $m = DEC.\left(k1.\left(C\right)\right)$.
    16. Result$\left(m\right)$.
    17. End
```

4. RESULTS AND DISCUSSION

The proposed framework EECPKE is executed in java in cloud sim. The experimental set up used to implement the proposed framework, the results produced reflects different performance measures and the efficiency of ECC over RSA . Under this proposed framework we have also compared our framework with existing techniques are presented in this section in detail.

4.1. Experimental Setup

The Proposed framework is developed to be suitable for any file. Files of varying sizes are taken to validate the optimization of prposed technique over existing ones. Is is implemented in Cloud Simulator using Java Language. The results generated by this proposed framework with different performance measures are presented in the succeeding 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. The 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. Here the comparison is performed in three phases, they are the Encryption Time of the data, re-encryption 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 are discussed in following manner.

4.3.1. Dual Authentication

By comparing the existing identity based authentication (IBA) schemes with this proposed dual 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 and this proposed scheme.

Since the user is authenticated by dual authentication where we have used OTP which is generated dynamically so the intruder cannot have false login or execute the password guess attack as well as offline or online guessing attack. The proposed scheme has used asymmetric encryption and proxy

Table 1. Security attacks analysis

Security Attacks	IBA Scheme	Proposed Scheme
Password guess attack	✓	✗
Denial-of-service	✓	✗
Offline guessing attack	✓	✗
False login	✓	✗
Online guessing attack	✓	✗
Impersonation attack	✓	✗

re-encryption this will save the file from adversaries fo even if he come to know about identity he cannot guess the secret and public key of legitimated user. This proves that our scheme is saved against the mentioned attacks and is highly secure.

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 IBA It has been proved that our proposed scheme works well than the others

4.3.2. Encryption Time

The encryption time of the data after encrypting it by the proposed optimized EECPKE technique is compared with the encryption techniques for instance RSA. In RSA depends on the Asymmetric key provided by the user encryption is performed. With the results produced by these encryption techniques with varying file sizes in this proposed one is compared and the outcomes are given in Table 3 and in Figure 3.

4.3.3. Re-Encryption Time

The File in cloud id re-encrypted from public key of owner to public key of recipient. The Re-encryption time of proposed technique is compared with the encryption techniques such as RSA. In RSA is based on key provided by the user encryption is performed and. The results are given in the Table 4 as well as in the Figure 4 as follows.

4.3.3. Decryption Time

The decryption time of the data after decrypting it by the proposed technique is compared with the encryption techniques such as RSA. In RSA is based on key provided by the user encryption is performed and. The results are given in the Table 5 as well as in the Figure 5 as follows.

4.3.3. Throughput

Throughput is equal to total plaintext in Kilo bytes encrypted divided by the total time (tencryption time + Reencryption time+Decryption Time). Higher the throughput, higher will be the performance.

Table 2. Features and provision analysis

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

Table 3. Encryption time and comparison with RSA

File Name	File Size (KB)	Encryption Time(ms)	
		RSA	ECC
F1	16	0.075	0.0214
F2	64	2.375	0.0782
F3	256	122.7	0.347
F4	384	439.1	0.657
F5	640	2132.95	1.351
F6	768	3739.95	1.629
F7	896	6186.5	2.187
F8	1024	9104.9	2.545
F9	1152	13058.25	3.012
F10	1280	17947.4	3.555
F11	1408	23702.1	4.104
F12	1536	31864.75	4.707
F13	1664	37659.25	5.295
F14	1792	48770.2	6.281
F15	1920	59248.95	6.283
F16	2048	70662.05	7.581

Figure 3. Comparison graph of encryption

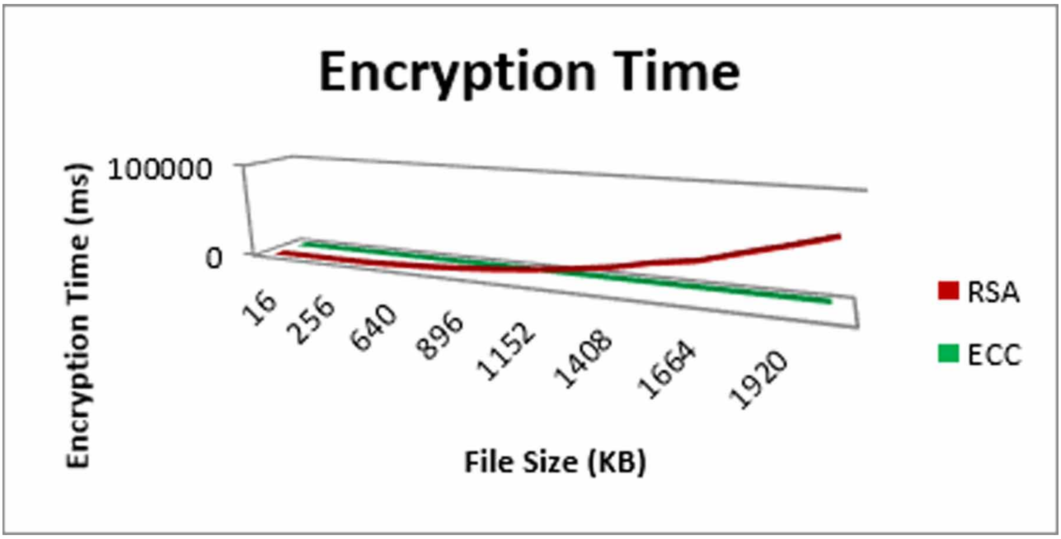


Table 4. Comparison re-encryption time

File Name	File Size (KB)	ReEncryption Time(ms)	
		RSA	ECC
F1	16	0.24	0.0078
F2	64	4.995	0.0329
F3	256	235.35	0.224
F4	384	875.15	0.271
F5	640	4324.65	0.427
F6	768	7532.7	0.564
F7	896	12231.15	0.642
F8	1024	18257.8	0.665
F9	1152	26033.55	0.686
F10	1280	36078.2	0.793
F11	1408	47399.45	0.879
F12	1536	63406.15	0.905
F13	1664	75619.5	1.045
F14	1792	97120.65	1.434
F15	1920	118493	1.961
F16	2048	140209.5	2.277

Figure 4. Comparison graph of re-encryption time

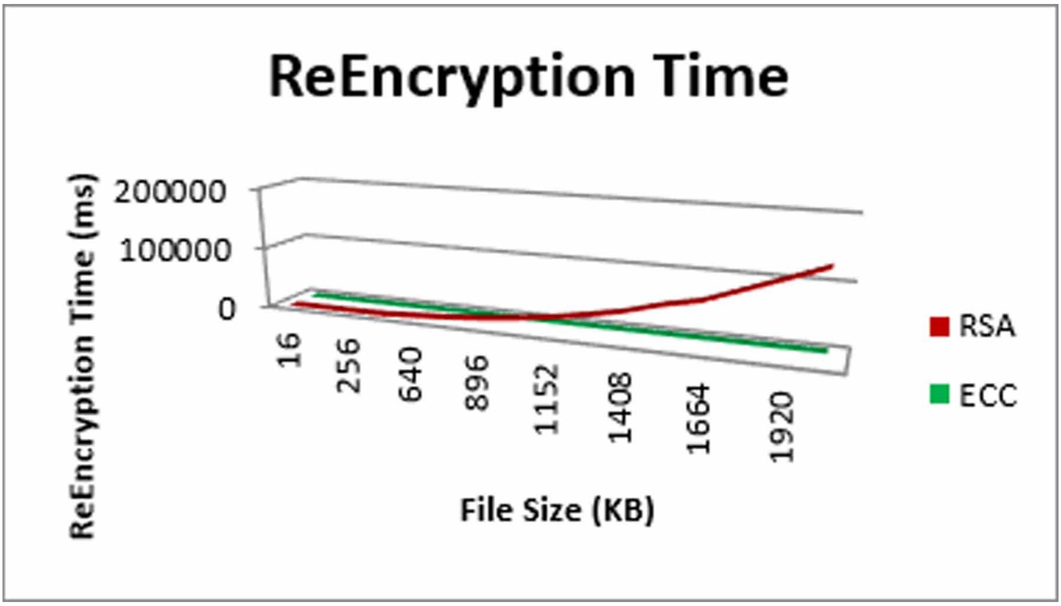


Table 5. Comparison decryption time

File Name	File Size (KB)	Decryption Time(ms)	
		RSA	ECC
F1	16	0.08	0.0088
F2	64	2.535	0.043
F3	256	119.75	0.242
F4	384	438.65	0.329
F5	640	2165.35	0.533
F6	768	3752.35	0.633
F7	896	6174.05	0.694
F8	1024	9121.05	1.099
F9	1152	12989.3	1.224
F10	1280	18105.3	1.343
F11	1408	23653.15	1.601
F12	1536	32396.05	1.907
F13	1664	37597.05	2.082
F14	1792	48461.55	2.127
F15	1920	59515	2.443
F16	2048	69920.35	2.674

Figure 5. Comparison graph of decryption time

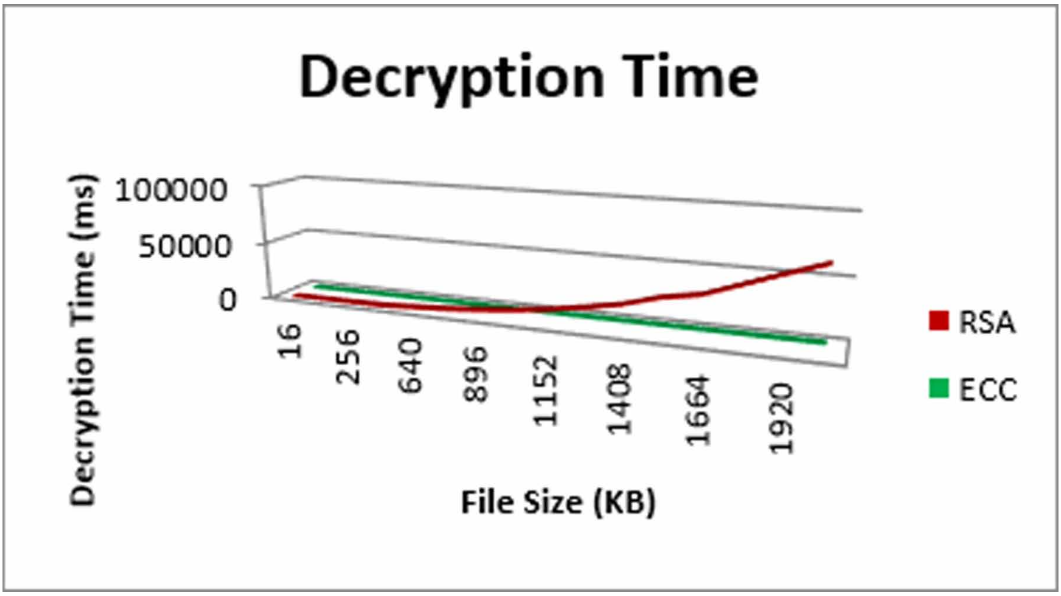
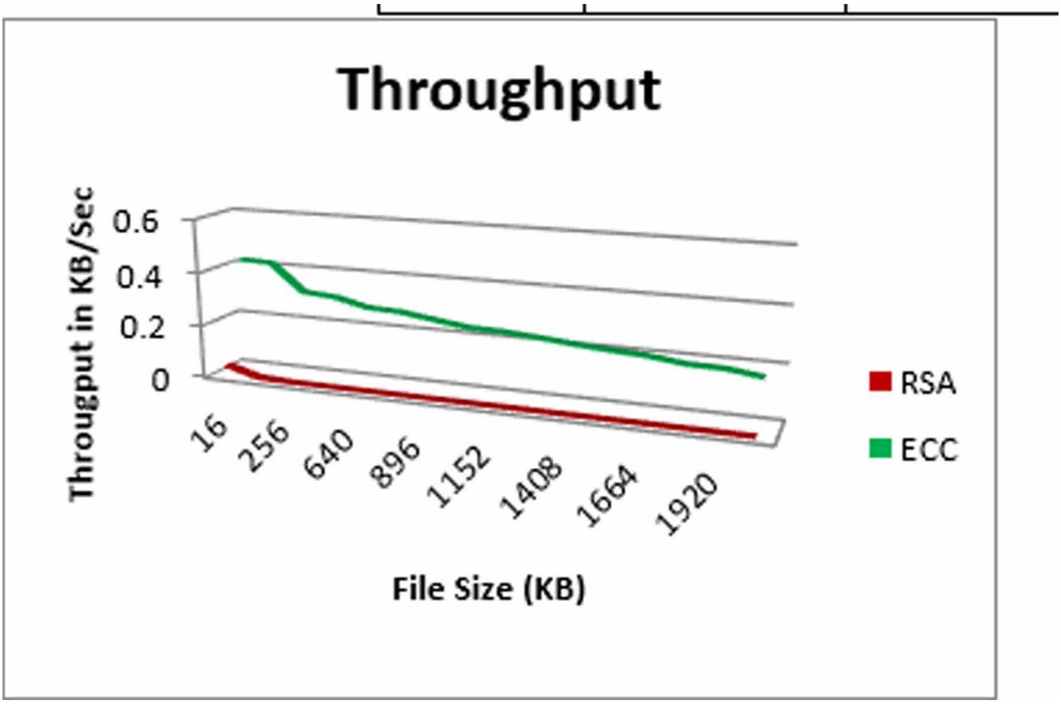


Table 6. Comparison of throughput

File Name	File Size (KB)	Throughput(KB/Sec)	
		RSA	ECC
F1	16	0.040506329	0.421053
F2	64	0.006461383	0.415315
F3	256	0.000535789	0.314883
F4	384	0.000219066	0.305489
F5	640	7.42205E-05	0.276936
F6	768	5.11148E-05	0.271762
F7	896	3.64351E-05	0.254329
F8	1024	2.80673E-05	0.237642
F9	1152	2.21193E-05	0.234051
F10	1280	1.77455E-05	0.224917
F11	1408	1.48594E-05	0.213852
F12	1536	1.20313E-05	0.204282
F13	1664	1.10289E-05	0.197578
F14	1792	9.22036E-06	0.182077
F15	1920	8.09249E-06	0.179658
F16	2048	7.29366E-06	0.163422

Figure 6. Comparison of throughput



$$Throughput = \frac{FileSize}{\left(TimeofEncryption + Timeof\,\mathrm{Re}\,Encryption + TimeofDecryption\right)}$$

By the value obtained from the encryption process, the throughput value is calculated for different file sizes. The above Table 6 shows the comparison between the throughput. The results in Figure 6 exposed that the proposed technique attains higher throughput. the proposed approach overtakes the other because of using dual authentication, the reliable algorithm which was used in 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 existing scheme.

5. CONCLUSION

Sharing data through cloud is growing rapidly however privacy and security is the most significant concern while sharing data especially in public cloud. In this paper we have presented a secure data sharing encryption technique called Dynamic Unidirectional Proxy Re-Encryption and Cipher text Policy Attribute based Encryption comprising of five phases Authentication Check, Encryption, Data Integrity Checking, User Confirmation and Data Retrieval. These phases ensures fine grained access of data along with maintaining the confidentially and privacy. The user/client can be revoked without generating and redistributing the keys. Even if the revoked user and cloud colludes the secret key of the owner can not be retrieved. Through results we have proved that our approach is more optimized in comparison to most commonly and widely used encryption algorithms namely: RSA. The throughput is increased by approximately 30%.EECPKE can encrypt the data with same security but with smaller key size and less time. In RSA the time of encryption and decryption increases exponentially with the increase in size which is almost negligible with EECPKE. In future we would be working on secured sharing of files of varying type like image, pdf, etc.

REFERENCES

Ahmed, N. S. S., Acharjya, D. P., & Sanyal, S. (2017). A framework for phishing attack identification using rough set and formal concept analysis. *International Journal of Communication Networks and Distributed Systems*, *18*(2), 186–212. doi:10.1504/IJCNDS.2017.082105

Ali, M., Khan, S. U., & Vasilakos, A. V. (2015). Security in cloud computing: Opportunities and challenges. *Information Sciences*, *305*, 357–383. doi:10.1016/j.ins.2015.01.025

AlZain, M. A., Soh, B., & Pardede, E. (2012). A new model to ensure security in cloud computing services. *Journal of Service Science Research*, *4*(1), 49–70. doi:10.1007/s12927-012-0002-5

Aono, Y., Boyen, X., & Wang, L. (2013, December). Key-private proxy re-encryption under LWE. In *International Conference on Cryptology in India* (pp. 1-18). Springer.

Ateniese, G., Benson, K., & Hohenberger, S. (2009, April). Key-private proxy re-encryption. In *Cryptographers' Track at the RSA Conference* (pp. 279-294). Springer. doi:10.1007/978-3-642-00862-7_19

Ateniese, G., Fu, K., Green, M., & Hohenberger, S. (2006). Improved proxy re-encryption schemes with applications to secure distributed storage. *ACM Transactions on Information and System Security*, *9*(1), 1–30. doi:10.1145/1127345.1127346

Bera, S., Misra, S., & Rodrigues, J. J. (2015). Cloud computing applications for smart grid: A survey. *IEEE Transactions on Parallel and Distributed Systems*, *26*(5), 1477–1494. doi:10.1109/TPDS.2014.2321378

Bethencourt, J., Sahai, A., & Waters, B. (2007, May). Ciphertext-policy attribute-based encryption. In *Security and Privacy, 2007. SP'07. IEEE Symposium on* (pp. 321-334). IEEE. doi:10.1109/SP.2007.11

Blaze, M., Bleumer, G., & Strauss, M. (1998, May). Divertible protocols and atomic proxy cryptography. In *International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 127-144). Springer.

Boukhlouf, D., Kazar, O., & Kahloul, L. (2016). Network security: Distributed intrusion detection system using mobile agent technology. *International Journal of Communication Networks and Distributed Systems*, *17*(4), 335–347. doi:10.1504/IJCNDS.2016.080583

Canetti, R., & Hohenberger, S. (2007, October). Chosen-ciphertext secure proxy re-encryption. In *Proceedings of the 14th ACM conference on Computer and communications security* (pp. 185-194). ACM.

Chu, C. K., & Tzeng, W. G. (2007, October). Identity-based proxy re-encryption without random oracles. In *International Conference on Information Security* (pp. 189-202). Springer. doi:10.1007/978-3-540-75496-1_13

Deng, R. H., Weng, J., Liu, S., & Chen, K. (2008, December). Chosen-ciphertext secure proxy re-encryption without pairings. In *International Conference on Cryptology and Network Security* (pp. 1-17). Springer.

Doshi, N., & Jinwala, D. C. (2014). Fully secure ciphertext policy attribute-based encryption with constant length ciphertext and faster decryption. *Security and Communication Networks*, *7*(11), 1988–2002. doi:10.1002/sec.913

Egorov, M., & Wilkison, M. (2016). *ZeroDB white paper*. arXiv preprint arXiv:1602.07168

Fan, X., & Liu, F. H. (2016). *Various Proxy Re-Encryption Schemes from Lattices*. IACR Cryptology ePrint Archive, 2016, 278.

Goyal, V., Pandey, O., Sahai, A., & Waters, B. (2006, October). 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). ACM. doi:10.1145/1180405.1180418

Green, M., & Ateniese, G. (2007). Identity-based proxy re-encryption. In *Applied Cryptography and Network Security* (pp. 288–306). Berlin: Springer. doi:10.1007/978-3-540-72738-5_19

Guo, F., Mu, Y., Susilo, W., Wong, D. S., & Varadharajan, V. (2014). CP-ABE with constant-size keys for lightweight devices. *IEEE Transactions on Information Forensics and Security*, *9*(5), 763–771. doi:10.1109/TIFS.2014.2309858

Hörandner, F., Krenn, S., Migliavacca, A., Thiemer, F., & Zwattendorfer, B. (2016, August). CREDENTIAL: a framework for privacy-preserving cloud-based data sharing. In *Availability, Reliability and Security (ARES), 2016 11th International Conference on* (pp. 742-749). IEEE. doi:10.1109/ARES.2016.79

Hur, J. (2013). Improving security and efficiency in attribute-based data sharing. *IEEE Transactions on Knowledge and Data Engineering*, *25*(10), 2271–2282. doi:10.1109/TKDE.2011.78

Jakimoski, K. (2016). Security techniques for data protection in cloud computing. *International Journal of Grid and Distributed Computing*, *9*(1), 49–56. doi:10.14257/ijgdc.2016.9.1.05

Jeyanthi, N., Barde, U., Sravani, M., Tiwari, V., & Iyengar, N. C. S. N. (2013). Detection of distributed denial of service attacks in cloud computing by identifying spoofed IP. *International Journal of Communication Networks and Distributed Systems*, *11*(3), 262–279. doi:10.1504/IJCNDS.2013.056223

Kaddouri, A., Guezouri, M., & Mbarek, N. (2018). A new proposed cloud computing based architecture for space ground data systems. *International Journal of Communication Networks and Distributed Systems*, *20*(2), 244–262. doi:10.1504/IJCNDS.2018.089772

Kirshanova, E. (2014, March). Proxy re-encryption from lattices. In *International Workshop on Public Key Cryptography* (pp. 77-94). Springer.

Kumar, S., & Singh, R. K. (2016). Secure authentication approach using Diffie-Hellman key exchange algorithm for WSN. *International Journal of Communication Networks and Distributed Systems*, *17*(2), 189–201. doi:10.1504/IJCNDS.2016.079102

Li, J., Li, Y. K., Chen, X., Lee, P. P., & Lou, W. (2015). A hybrid cloud approach for secure authorized deduplication. *IEEE Transactions on Parallel and Distributed Systems*, *26*(5), 1206–1216. doi:10.1109/TPDS.2014.2318320

Li, Q., Wei, W., Tao, M., & Chen, Q. (2014). A DDOS defence scheme based on two-stage traffic flow control. *International Journal of Communication Networks and Distributed Systems*, *13*(3-4), 290–300. doi:10.1504/IJCNDS.2014.064638

Liang, K., Au, M. H., Liu, J. K., Susilo, W., Wong, D. S., Yang, G., & Yang, A. et al. (2015). A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. *Future Generation Computer Systems*, *52*, 95–108. doi:10.1016/j.future.2014.11.016

Libert, B., & Vergnaud, D. (2011). Unidirectional chosen-ciphertext secure proxy re-encryption. *IEEE Transactions on Information Theory*, *57*(3), 1786–1802. doi:10.1109/TIT.2011.2104470

Lin, H. Y., & Tzeng, W. G. (2012). A secure erasure code-based cloud storage system with secure data forwarding. *IEEE Transactions on Parallel and Distributed Systems*, *23*(6), 995–1003. doi:10.1109/TPDS.2011.252

Liu, Q., Wang, G., & Wu, J. (2014). Time-based proxy re-encryption scheme for secure data sharing in a cloud environment. *Information Sciences*, *258*, 355–370. doi:10.1016/j.ins.2012.09.034

Liu, Z., Cao, Z., & Wong, D. S. (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., & Li, J. (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

Manvi, S. S., & Shyam, G. K. (2014). Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. *Journal of Network and Computer Applications*, *41*, 424–440. doi:10.1016/j.jnca.2013.10.004

Micciancio, D., & Peikert, C. (2012, April). Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 700-718). Springer. doi:10.1007/978-3-642-29011-4_41

Mitra, A., Kundu, A., Chattopadhyay, M., & Chattopadhyay, S. (2017). A cost-efficient one time password-based authentication in cloud environment using equal length cellular automata. *Journal of Industrial Information Integration*, *5*, 17–25. doi:10.1016/j.jii.2016.11.002

Nishimaki, R., & Xagawa, K. (2015). Key-private proxy re-encryption from lattices, revisited. IEICE TRANSACTIONS on Fundamentals of Electronics. *Communications and Computer Sciences*, *98*(1), 100–116.

Nunez, D., Agudo, I., & Lopez, J. (2016). On the application of generic CCA-secure transformations to proxy re-encryption. *Security and Communication Networks*, *9*(12), 1769–1785. doi:10.1002/sec.1434

Odelu, V., Das, A. K., Rao, Y. S., Kumari, S., Khan, M. K., & Choo, K. K. R. (2017). Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment. *Computer Standards & Interfaces*, *54*, 3–9. doi:10.1016/j.csi.2016.05.002

Park, N. (2011). Secure data access control scheme using type-based re-encryption in cloud environment. In *Semantic methods for knowledge management and communication* (pp. 319–327). Berlin: Springer. doi:10.1007/978-3-642-23418-7_28

Patel, A., Taghavi, M., Bakhtiyari, K., & Celestino Júnior, J. (2013). An intrusion detection and prevention system in cloud computing: A systematic review. *Journal of Network and Computer Applications*, *36*(1), 25–41. doi:10.1016/j.jnca.2012.08.007

Rong, C., Nguyen, S. T., & Jaatun, M. G. (2013). Beyond lightning: A survey on security challenges in cloud computing. *Computers & Electrical Engineering*, *39*(1), 47–54. doi:10.1016/j.compeleceng.2012.04.015

Sahai, A., & Waters, B. (2005, May). Fuzzy identity-based encryption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 457-473). Springer.

Samanthula, B. K., Elmehdwi, Y., Howser, G., & Madria, S. (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., & Bhathal, E. G. S. (2016). *To Enhance the OTP Generation Process for Cloud Data Security using Diffie-Hellman and HMAC. Global Journal of Computer Science and Technology.*

Saouli, H., Kazar, O., & Benharkat, A. C. N. (2015). SaaS-DCS: Software-as-a-service discovery and composition system-based existence degree. *International Journal of Communication Networks and Distributed Systems*, *14*(4), 339–378. doi:10.1504/IJCNDS.2015.069670

Seo, J. W., Yum, D. H., & Lee, P. J. (2013). Comments on" Unidirectional Chosen-Ciphertext Secure Proxy Re-Encryption. *IEEE Transactions on Information Theory*, *59*(5), 3256. doi:10.1109/TIT.2012.2236606

Shao, J. (2015). *Bibliography on proxy re-cryptography*. Academic Press.

Sood, S. K. (2012). A 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

Sun, L., Dong, H., Hussain, F. K., Hussain, O. K., & Chang, E. (2014). Cloud service selection: State-of-the-art and future research directions. *Journal of Network and Computer Applications*, *45*, 134–150. doi:10.1016/j.jnca.2014.07.019

Tamizharasi, G. S., Balamurugan, B., & Aarthy, S. L. (2016, August). Scalable and efficient attribute based encryption scheme for point to multi-point communication in cloud computing. In *Inventive Computation Technologies (ICICT), International Conference on* (Vol. 1, pp. 1-4). IEEE. doi:10.1109/INVENTIVE.2016.7823292

Tysowski, P. K., & Hasan, M. A. (2013). Hybrid attribute-and re-encryption-based key management for secure and scalable mobile applications in clouds. *IEEE Transactions on Cloud Computing*, *1*(2), 172–186. doi:10.1109/TCC.2013.11

Wang, B., Li, B., & Li, H. (2015). Panda: Public auditing for shared data with efficient user revocation in the cloud. *IEEE Transactions on Services Computing*, *8*(1), 92–106. doi:10.1109/TSC.2013.2295611

Wang, C., Wang, Q., Ren, K., Cao, N., & Lou, W. (2012). Toward secure and dependable storage services in cloud computing. *IEEE Transactions on Services Computing*, *5*(2), 220–232. doi:10.1109/TSC.2011.24

Wang, G., Liu, Q., Wu, J., & Guo, M. (2011). Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers. *Computers & Security, 30*(5), 320-331.

Wang, L. L., Chen, K. F., Mao, X. P., & Wang, Y. T. (2014). Efficient and provably-secure certificateless proxy re-encryption scheme for secure cloud data sharing. *Journal of Shanghai Jiaotong University (Science)*, *19*(4), 398–405. doi:10.1007/s12204-014-1514-6

Wei, L., Zhu, H., Cao, Z., Dong, X., Jia, W., Chen, Y., & Vasilakos, A. V. (2014). Security and privacy for storage and computation in cloud computing. *Information Sciences*, *258*, 371–386. doi:10.1016/j.ins.2013.04.028

Weng, J., Deng, R. H., Liu, S., & Chen, K. (2010). Chosen-ciphertext secure bidirectional proxy re-encryption schemes without pairings. *Information Sciences*, *180*(24), 5077–5089. doi:10.1016/j.ins.2010.08.017

Xagawa, K., & Tanaka, K. (2010). *Proxy Re-Encryption based on Learning with Errors*. Mathematical Foundation of Algorithms and Computer Science.

Xiong, H., Zhang, X., Yao, D., Wu, X., & Wen, Y. (2012, February). Towards end-to-end secure content storage and delivery with public cloud. In *Proceedings of the second ACM conference on Data and Application Security and Privacy* (pp. 257-266). ACM. doi:10.1145/2133601.2133633

Xu, P., Chen, H., Zou, D., & Jin, H. (2014). Fine-grained and heterogeneous proxy re-encryption for secure cloud storage. *Chinese Science Bulletin*, *59*(32), 4201–4209. doi:10.1007/s11434-014-0521-1

Yang, K., & Jia, X. (2012). Data storage auditing service in cloud computing: Challenges, methods and opportunities. *World Wide Web (Bussum)*, *15*(4), 409–428. doi:10.1007/s11280-011-0138-0

Yang, Y., & Zhang, Y. (2011, September). A generic scheme for secure data sharing in cloud. In *Parallel Processing Workshops (ICPPW), 2011 40th International Conference on* (pp. 145-153). IEEE. doi:10.1109/ICPPW.2011.51

Yu, S., Wang, C., Ren, K., & Lou, W. (2010, March). Achieving secure, scalable, and fine-grained data access control in cloud computing. In *Infocom, 2010 proceedings IEEE* (pp. 1–9). Ieee. doi:10.1109/INFCOM.2010.5462174

Zhang, Y., Zheng, D., Chen, X., Li, J., & Li, H. (2014, October). Computationally efficient ciphertext-policy attribute-based encryption with constant-size ciphertexts. In *International Conference on Provable Security* (pp. 259-273). Springer. doi:10.1007/978-3-319-12475-9_18

Zhu, Y., Ahn, G. J., Hu, H., Yau, S. S., An, H. G., & Hu, C. J. (2013). Dynamic audit services for outsourced storages in clouds. *IEEE Transactions on Services Computing*, *6*(2), 227–238. doi:10.1109/TSC.2011.51

Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues. *Future Generation Computer Systems*, *28*(3), 583–592. doi:10.1016/j.future.2010.12.006

Neha Agarwal is currently working as an Assistant Professor in Amity School of Engineering and technology, Amity University, Uttar Pradesh. She is pursuing her PhD from Dr. A.P.J. Abdul Kalam Technical University (APJAKTU) (UP) in the area of Cloud Computing. She received her M.Tech in Computer Science Engineering from Amity University Noida, Uttar Pradesh.

Ajay Rana is a director at Amity University. He received his M.Tech degree in Computer Science Engineering from Kurukshetra university, Haryana, India. He obtained hid Ph.D degree from UP Technical University, Lucknow (UP) India.. He has published more than 200 Research Papers in reputed Journals and Proceedings of International and National Conferences. He has co-authored 06 Books and co-edited 36 Conference Proceedings. He is Editor in Chief, Technical Committee Member, Advisory Board Member for 18 Plus Technical Journals and Conferences at National and International Levels.

Jai Prakash Pandey is currently working as Professor and Director in the Department of electrical engineering at Kamala Nehru Institute of Technology, Sultanpur, (UP), India. He has received his B. Tech. and M. Tech. degree in Electrical Engineering from Kamala Nehru Institute of Technology, Sultanpur (UP), India. He obtained his Ph.D degree from UP Technical University, Lucknow (UP) India. His research interests include applications of artificial techniques to electrical engineering problems in power system, estate estimation and power quality.

Neha Agarwal is currently associated with University of Petroleum and Energy Studies, Dehradun as Professor and Head of Department of Virtualization under School of Computer Science. He has completed Ph.D in Computer Science & Engineering from Indian Institute of Technology, Roorkee. He is having more than 18+ Years of academic, administrative and research experience. His area of research includes Cloud Computing, Cloud Security, Internet of Things and GIS. He has completed two international consultancy projects and published more than 70 research publications in reputed journals and conferences. He has organized several IEEE conferences and workshops like NGCT-2015, NGCT-2016, IEEE Mini POCO-2016, Doctoral conference-2017 and so on. He has received Best Researcher Award from UPES in 2017 and Distinguished Academician Award from Pentagram Research Centre, Hyderabad in 2017. He has delivered several keynote sessions in international and national conferences and workshops. He is the Vice President of Next Generation Computing Technology Society, Dehradun. He is the guest editors for several international journals like IJCNDS, IJIRR, IJITWE, IJRSDA and IJITPM from Inderscience and IGI Global. He is the member of IEEE.