# TIME STAMP BASED ACCESS CONTROL FOR PERSONAL HEALTH RECORDS

**Mr. Jaibir Singh, Dr.PrasaduPeddi**

**Research Scholar, SHRI JAGDISHPRASAD JHABARMAL TIBREWALA UNIVERSITY VIDYANAGARI, JHUNJHUNU, RAJASTHAN.**

**Assistant Professor, Dept of CSE, SHRI JAGDISHPRASAD JHABARMAL TIBREWALA UNIVERSITY VIDYANAGARI, JHUNJHUNU, RAJASTHAN**

**ABSTRACT:** *Computing also attracts them with pay-as-you-use method and also features health listing owners. Data owner require rather not keep up their data's backup. As an alternative, "cloud" can keep up with the info. As if Data owner has hands over their information it can end up. As a way to attract more hands-onData owner entry control procedures ought to be invented. Gain control can be just a mechanism that offers avoidance of utilization of tools. The accessibility insurance guidelines can pick also exactly what actions are permitted to do to the info and who are able to see the information. To conquer, a Hierarchical identification Purpose Temporal primarily centered mostly Proxy Re-Encryption (HIRT-PRE) strategy has been suggested. It lets information to be encrypted by virtually any Data owner under people individuality predicated on job in present and expectancy period to assign data direction power into your cloud and, to shield information. This chapter centers on integration of job hierarchical, identity and time to boost the info gain controller.*

**KEYWORDS:** ***Cloud computing, Encryption, Decryption and Access control***

## I. INTRODUCTION

An encryption structure is proposed by considering data confidentiality and access measure to define fine-grained access control over cloud data. The problem of providing different keys for accessing data is difficult. Each user in a real world is associated with set of attributes, which are helpful in devising a new access policy over the remote data. The access policy is the logical combination of different attributes that the user is allowed to access. The fine-graininess is achieved through different access policy that is expressed in the form of logical expression. The proposed system that uses different attributes along with proxy re-encryption scheme achieves,

New file can be inserted or deleted without affecting the access structure.

User adding or revoking will not affect the data privacy and access policy.

Without any knowledge over data, the cloud server can re-encrypt data.

The chapter discusses system structure, where different attributes are identified. The security model is proposed that incorporates attributes for re-encryption and system is justified with a proven security analysis.

## II. DESIGN GOALS

The goal is to achieve fine grained access control on cloud data. The cloud server should not learn anything about the data or the access structure but to allow DATA OWNER to define user set whocan access data from cloud. The addition or deletion of data users should not affect the access policies of other users.

In order to achieve fine grained access control profile-based access control policies are defined. The attributes are expressed in the form of access structure. Initially data is encrypted using secret key known only to Data owner and the same is shared with data user who requests data access. The encrypted data is then placed in cloud. The encrypted data is re-encrypted using proxy re-encryption based on attributes for each data user. When data user sends data access request to cloud server, the server obtains access structure and compares with access policy of the particular user given by owner for that particular user. If access structure matches policy then the encrypted data is passed on to the user and not otherwise passed. The profile-based encryption system is proposed. In the first level data is encrypted using symmetric encryption and second level using proxy re-encryption scheme based on Attribute Based Encryption.

## III. SYSTEM STRUCTURE

The character of cloud computing doesn't let the security strategies to function therefore; the thoughts are focused in media that is static and smaller. The system will probably possess categorized collection of features along with all users along with also access arrangement might be fixed. In scenario that is lively end consumers render or can combine and also set arrangement will likely soon probably change. The secret thing that needs to be invented for user must have legitimacy inside the category should have died before the manhood remains. Additionally, attributes may not be specified unique along with mapping between both features and also client can't be accomplished. Even the architecture will probably get a high jurisdiction that includes control within the composition. The authority underneath the jurisdiction includes got handle of consumers over the domain name and alternative domain names also provides arrangement. The people beneath the domain authority could be user or can be Data owner. The assumption is the info users ' are permitted to truly really possess. Fig 1 reflects the hierarchical arrangement of this procedure utilized. These re Search triggered us to indicate that a brand-new entry assistance mechanism for both cloud computing.

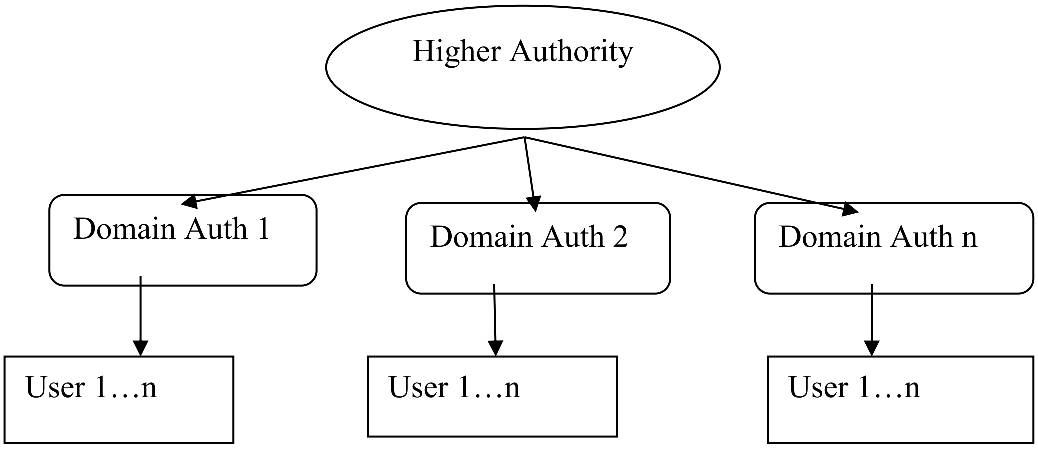


Figure 1 System Structure

### 3.1 Novelty of the proposed system

- The attributes incorporating hierarchy, identity, role and time combined with proxy re-encryption scheme is not explored in the existing schemes.
- The attribute matrix is incorporated to identify authenticated attribute set.
- The binary representation of attributes is used to form access key for the authorized users.
- Information table which consists of proxy key and ciphertext listing is given to cloud server which helps to verify the authorized users.

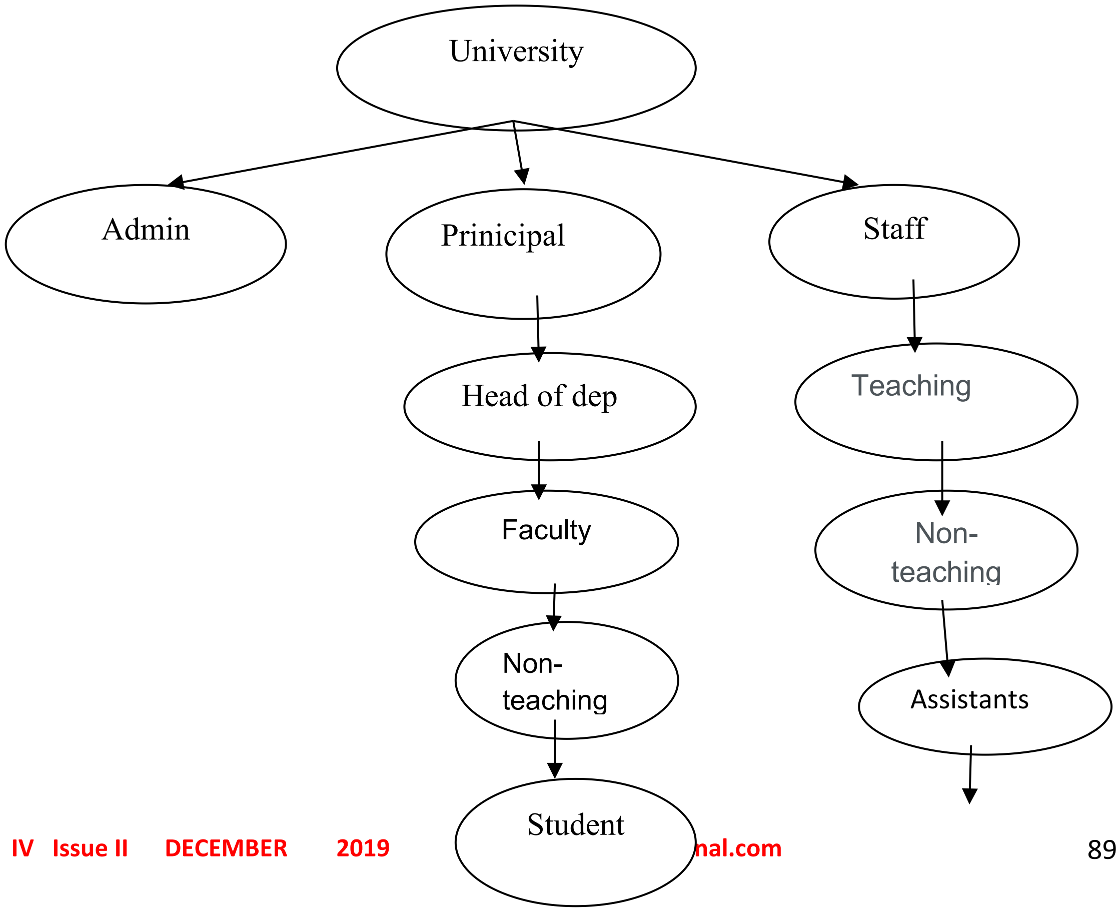
- The security of the system is analyzed based on data privacy, confidentiality and access control.

## IV. ACCESS STRUCTURE

There will be number of attributes available in the system. The Data owner selects the essential attributes and represents them in the form of access structure. The authorization is accessed in the form of bottom-up approach from access tree and is represented in access matrix. The row in an access matrix represents the authorized access set.

**Definition 4.1 (Access Structure):** Any access structure can be represented in the form of access tree. The access tree consists of threshold gates and attributes in the leaf node. The threshold gates consist of AND/OR gates. The threshold gates will take value between 0 and numattr, where numattr is equal to the number of attributes. The OR gate will take a maximum value of 1 and gates takes maximum value equal to the number of attributes.

Any access structure can be represented as access tree and each data is associated with access structure. Figure2 shows an example of hierarchical access structure over the result dataset within a college. The structure insists that controller of Examination, Principal and Head of Departments of various departments can access the result of all departments at different level but professors and students can view results based on their privileges



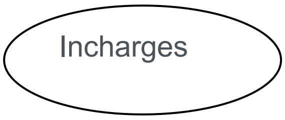


Figure 2 an example of Hierarchical Access Structure

## SYSTEM OUTLINE

It lets proprietor to possess their own data. Just about every data will be shared according to. It works by using proxy strategy that cloud services to gives accent protection dependent around the info supplied from the proprietor. To attain this, Hierarchical identification part Temporal primarily centered mostly Proxy Re-encryption Strategy (HIRT-PRE) has been suggested. It helps Data owner to detach data utilizing covert key at an identical point, to assign data direction power to cloud and, even to shield information. Moreover that the user can assign access command power into the cloud, that might grant the accessibility to a user below the character which he plays with, taking into consideration the devote consumer's toolbox also from altering the cipher text encoded with all the Data owner's identification into this individual with all an individual's individuality.

Is offered with the entire key. The cloud us for proxy strategy the consumer's features. For that reason, data could be decrypted by the customers at the foreseeable long run with their secret based in their own individuality into their own hierarchy with all character that they play around the moment. The Data owner constructs the access structure by identifying different user’s attributes during initial registration. The cloud server builds the proxy re-encryption key using access structure provided by the Data owner while placing his data in cloud. When the Data owner provides the access policies to the cloud, it can convert the cipher text outsourced byData owner to the ciphertext that can be decrypted by the user.

In this the Data owner uploads the encrypted file to the cloud. Then the cloud performs the Proxy Re-Encryption using the user’s access policy and stores it in the storage. Whenever the user wants to access the file, user retrieves it by decrypting the file using secret identity.

## IV. HIRT-PRE SCHEME

The Data owner initially processes the data before placing it in the cloud. The profile of each user is given to the cloud server in the form of attribute matrix. The cloud performs proxy re-encryption using the access structure defined by the Data owner.

### 5.1 HIRT-PRE Framework

The HIRT-PRE framework consists of five polynomial time algorithms.

**System Setup(AT,$\lambda$,A)→ (Attribute matrix(AS)):** The System Setup algorithm takes attributes, security parameter and access tree as input and outputs attribute matrix.

**KeyGen($\lambda$,PK)→ (pxk,shk):** The KeyGen algorithm takes security parameter and public key PK as input and outputs secret key shk and proxy key pxk for decrypting data which are obtained from cloud and Secret Hash key which are generated based on the attributes of different users and are sent to data users.

**Encrypt(M,shk)→ CT:** The random algorithm takes secret key shk and Message M as input and outputs ciphertext CT as output.

**Re-Encrypt(CT, Tc, Attribute matrix)→ CTT:** The server after receiving encrypted data from cloud re-encrypts the same using attribute matrix given by the Data owner considering current time Tc. The server re-encrypts the data and maintains the ciphertext CTT. This CTT consists of Ciphertext header CH and body. The CH consists of validity period which are verified by incorporating current time during data request. If Tc<Tv, then the encrypted data is passed on to the data user.

**Decrypt (CT, CH, Shk) → M:** The secret key Shk , Ciphertext header CH and Ciphertext CT as input the decryption algorithm outputs message M.

The Algorithm 5.1 and 5.2 explains the steps needed to be performed during system setup and sharing data access.

**Algorithm 5.1 System Setup**

```
Begin;
1. Access Matrix is constructed using Access tree AT, Security Parameter and Set of attributes;
2. Generate Proxy key 'pxk' nd secret key 'shk' using security parameter and public key 'pk';
3. The message 'M' is encrypted using secret key and outputs the ciphertext 'CT';
End;
```

**Algorithm 5.2 Data sharing Access**

```
Begin;
1. The ciphertext 'CT' is re-encrypted using attribute matrix and current time Tc to output the re-encrypted ciphertext 'CTT';
2. CTT is comprised of ciphertext header CH and cipher text body 'CT';
3. Validity period 'Tv' is incorporated within CH;
4. If Tc<Tv
5. Then
6. Return Cipher text CT to user;
7. Else
8. Return null;
End;
```

The workflow of HIRT-PRE scheme is show in Figure 4.4. The Data owner uses symmetric key encryption scheme to encrypt the data before placing data in cloud.

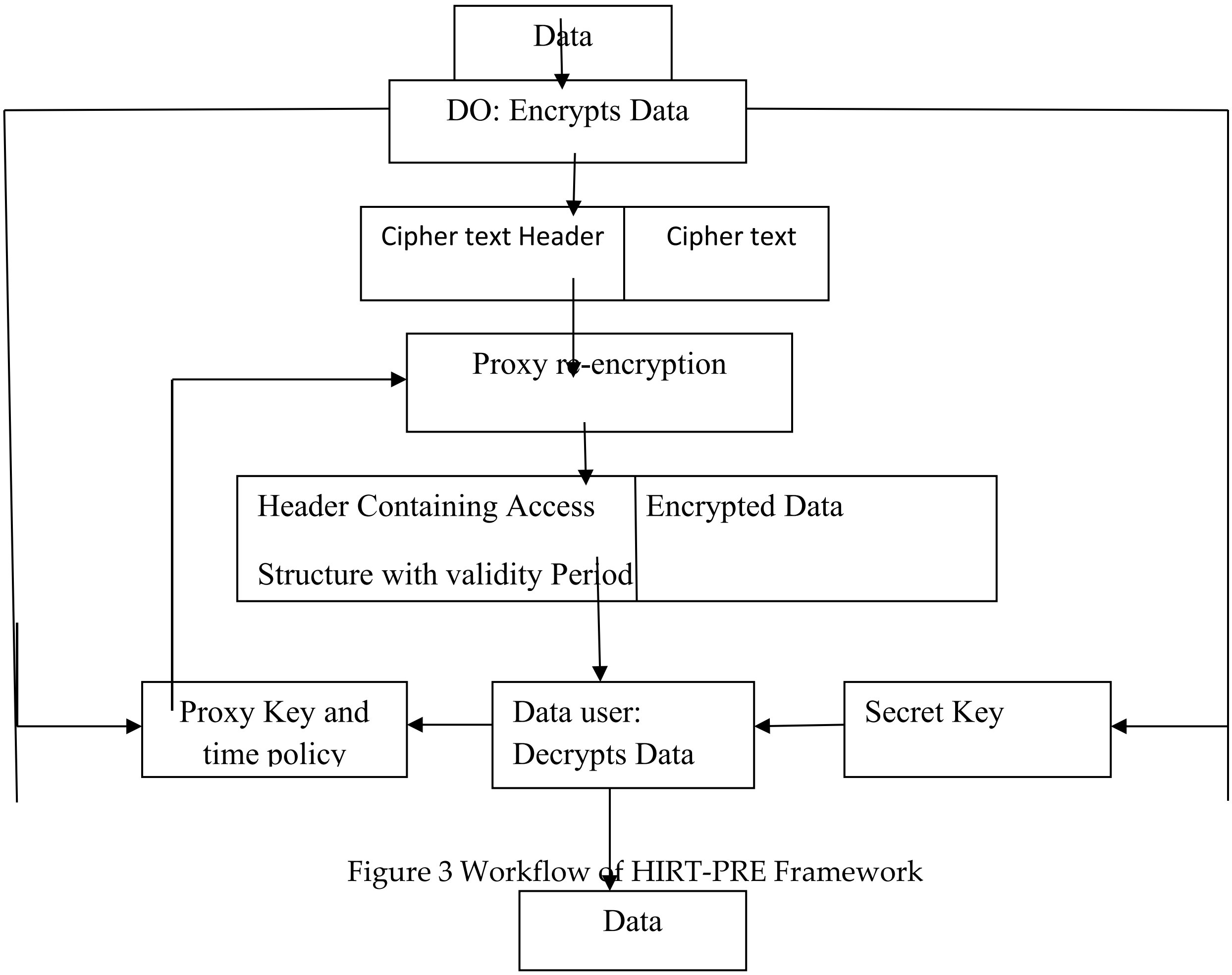


Figure 3 Workflow of HIRT-PRE Framework

## VI. EXPERIMENTAL ANALYSIS

The proposed system and existing schemes follows the same experimental procedure but the framework for authenticating user is different from the existing work. The procedure consists of system startup, key generation, encryption and decryption procedure which offers fine-grained access control for remote data.The scheme proposed by HASBE uses tree based access structure; so, the time generation of key based on access structure increases as number of attributes increases. In the method proposed by only temporal policies are considered so the time taken for key generation is less whereas security is also less. In Yu‟s scheme Master key, secret key and public key are devised based on bilinear mapping that increases the key generation time when compared with methods. Figure 4 shows the key generation time of HIRT-PRE scheme and the same is compared with different existing works.

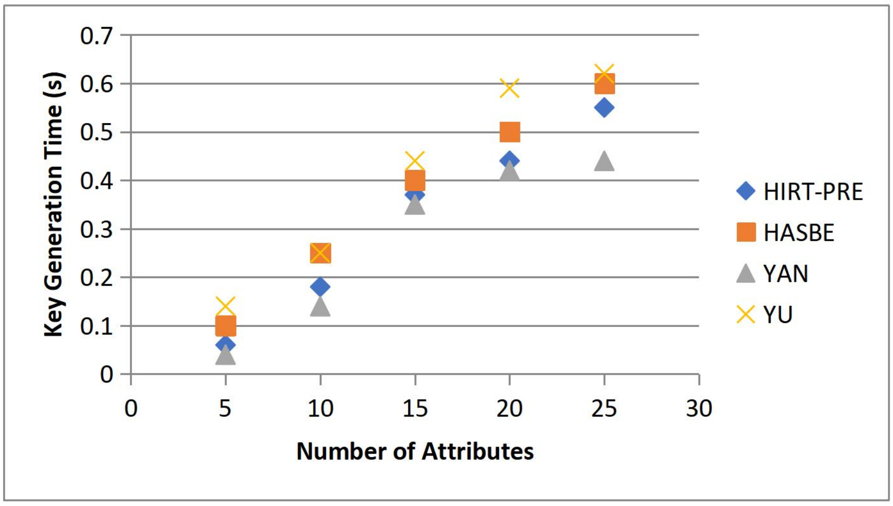


Figure 4 Key Generation Time

Figure 5 compares decryption time taken for different levels in the access tree. The decryption time remains uniform for different number of access levels in the access tree. In HASBE scheme the decryption time varies with the access structure. The number of attributes at different level in the access structure will not impact the decryption time in the proposed system as the secret key are generated while user registration and are generated based on the entire access structure.

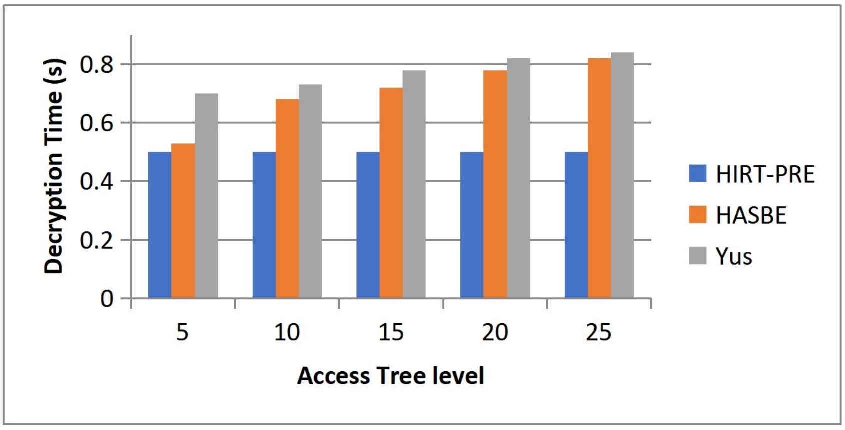


Figure 5 Decryption Time

In the proposed HIRT-PRE scheme the computation remains constant as each file is encrypted with proxy key by the server along with time policy. In HASBE scheme re-encryption is done by the Data owner and less effort is done by server. The computation effort of Data owner is higher in HASBE scheme when compared to proposed scheme. In Wang's scheme the re-encryption depends on number of attributes in the access structure. Figure 6 compares the computation effort of cloud server.

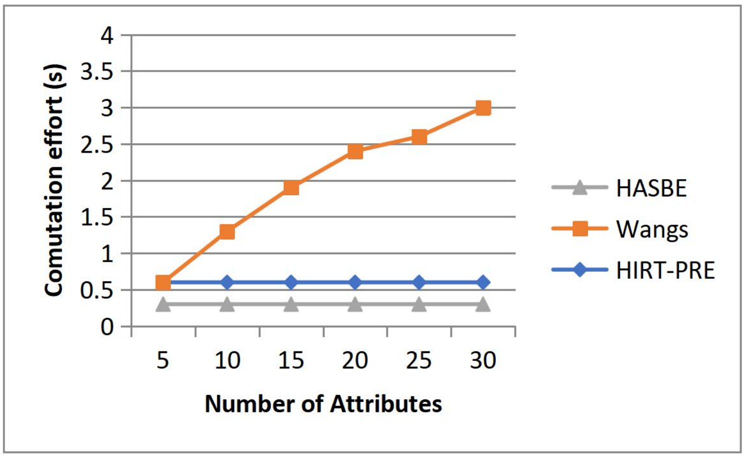


Figure 6 Computation Effort of cloud server

The Data owner has to identify the attributes possessed by data users and based on those attributes secret key has to be designed which increases the computation time of the Data owner. The information table is an additional storage needed in cloud server that needs to perform proxy re-encryption over the ciphertext.

## CONCLUSION

As there are very different access control models based on the current models like HASBE and HIRT-PRE, a comparison is made with few models like Hierarchical RBAC and constrained RBAC to check proposed version provides capacities offered by those models. A comparison can be made with Characteristic based access control. But the suggested model doesn't include different encryption methods which can be employed with the HASBE and HIRT-PRE such as attribute-based encryption and different similar techniques utilized in cloud environment for securing the information. The proposed work concentrates only on giving a basic model-based object relation. The upcoming job would take directly into account all demands of an access controller model and offers a standard methodology incorporating it directly into almost any application. A deeper study role mixing calculation is required that might be helpful using application scenarios.

## REFERENCES

*1. A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption," pp. 62–91.*

*2. Anna et.al, 2007, "Ontology-Based Approach for Managing Personal Health and Wellness Information", ISSN: 1094-687X, PP: 569-571.*

*3. PrasaduPeddi (2017) "Design of Simulators for Job Group Resource Allocation Scheduling In Grid and Cloud Computing Environments", ISSN: 2319- 8753 volume 6 issue 8 pp: 17805-17811.*

*4. A. Stolyar; et.al, 2006, "A Patient-Centered Health Record in a Demonstration Regional Health Information Network", PP: 160-163.*

*5. Amin Fallahi; et.al, 2017, "Towards Secure Public Directory for Privacy-Preserving Data Sharing", ISSN: 1063-6927, PP: 2577-2578.*

*6. PrasaduPeddi, 2018, Data sharing Privacy in Mobile cloud using AES, ISSN 2319-1953, volume 7, issue 4.*

*7. Chien Liu, 2013, "Developing IHE-Based PHR Cloud Systems", PP: 1022-1025.*

*8. Chia Liu; et.al, 2013, "Secure PHR Access Control Scheme for Healthcare Application Clouds", ISSN: 0190-3918, PP: 1067-1076.*

*9. Danwei Chen; et.al, 2014, "Securing patient-centric personal health records sharing system in cloud computing", ISSN: 1673-5447, Volume: 11, Issue: 13, PP: 121-127.*

10. *Prasadu Peddi (2019), Data Pull out and facts unearthing in biological Databases, International Journal of Techno-Engineering, Vol. 11, issue 1, pp: 25-32.*

11. *Imad Ghoubach; et.al, 2016, "Efficient secure and privacy preserving data access control scheme for multi-authority personal health record systems in cloud computing", PP: 174-179.*

12. *Linke Guo; et.al, 2012, "User-centric private matching for eHealth networks - A social perspective", ISSN: 1930-529X, PP: 732-737.*

13. *Prasadu Peddi (2018), "A Study For Big Data Using Disseminated Fuzzy Decision Trees", ISSN: 2366- 1313, Vol 3, issue 2, pp:46-57.*