A Novel Method for Patient Centric Secure and Scalable sharing of PHR in Cloud Computing using Encryption

Shaheen Taj S.A, Prathibha Kiran, Elavarasi

Abstract: PHRs grant patients access to a wide range of health information sources, best medical practices and health knowledge. In patient centric secure sharing, patients will create, manage and control their personal health data from one place using the web. Prior to storing the records in cloud server, they are encrypted using encryption algorithm which ensures the patient's full control over their PHR. In addition to PHR (Medical history, current exams), personal files, insurance details and sensitive information can also be stored and shared. Patients only decide which set of users can access which set of files. All the files stored in clouds which are semi-trusted servers, are in the encrypted form and are confidential to other users. We make use of Attribute Based encryption (ABE) to encrypt the files. In this scheme, users are categorized into personal and professional domains which greatly reduce the key management complexity. There is a structured way to access the files for personal and professional purposes. Patients are able to dynamically modify the access policy and attributes.

Index Terms: Attribute Based Encryption, Cipher, DES, Feistel.

I. INTRODUCTION

A Personal Health Record (PHR) is a health record where health data and information related to the care of a patient is maintained by the patient. Patients control the health information in PHR and can get it anywhere at any time with Internet access. Personal health record makes it easy to gather and manage the medical information in one accessible and secure location. Carrying paper records is a big drawback since patients rarely have paper records when they need. Personal health record systems remedy that problem by making the personal health record accessible anytime via a Web-enabled device, such as computer, phone or tablet. Literally having a personal health record can be a lifesaver. In emergency, patients can quickly give personal vital information, such as information about the disease, medications and drug allergies. A PHR service allows patients to create, manage, and control their personal health data from one place through the web, which has made the storing, retrieving, and sharing of the medical information more secure. Each patient has the full control of their medical records, other files and can share their health data with a wide range of users, including doctors, family members or friends. Researches on PHR's using cloud computing is still underway.

Manuscript received on May, 2013.

Shaheen Taj S.A, M.Tech, Department of ECE, AMC Engineering College, Bangalore, India.

Prathibha Kiran, Associate Professor, Department of ECE AMC Engineering College, Bangalore, India.

Elavarasi, Associate Professor, Department of ECE AMC Engineering College, Bangalore, India.

The main concern is about whether the patients could actually control the sharing of their sensitive personal health records, especially when they are stored on a third-party server such as cloud providers which people cannot fully trust. To ensure patient-centric privacy control over their own PHRs, it is essential to encrypt the data before storing. Basically, the PHR owners themselves will decide how to encrypt their files and to allow which set of users to obtain access to each file. A PHR file should only be available to the users who are given the corresponding decryption key and is confidential to the rest of the users. Furthermore, the patient will always retain the right to grant also revoke access privileges when it is necessary.

Due to high cost of building and maintaining separate data centers, many PHR services are outsourced and provided by third party service providers for example, Microsoft Health Vault (UK). The Health Vault Program of Microsoft will allow users including individuals, health centers, hospitals etc. to gain access to the information on health related issues. The user interface will be simple, that would allow anyone to operate the program easily. But on the other hand, there is still less guarantee to protect one's privacy statement. Many people do not wish to share their private health records and other information universally through the Health Vault.

1.1 Related Work

J. Benaloh, [8] has proposed a scheme in which a file can be uploaded without key distribution and it is highly efficient. This is a single data owner scenario and thus it is not easy to add categories. C. Dong, [10] has explored that the data encryption scheme does not require a trusted data server. The server can perform encrypted searches and updates on encrypted data without knowing the plaintext or the decryption keys. But in this scheme the server knows the access pattern of the users which allows it to infer some information about the queries. To realize fine grained access control, the traditional public key encryption based schemes [8], [10] either incur high key management overhead, or require encrypting multiple copies of a file using different users' keys. To improve upon the scalability of the said solutions, one-to-many encryption methods such as attribute based encryption (ABE) can be used. In Goyal et. al's paper [11] , data is encrypted under a set of attributes so that multiple users who possess proper key can decrypt.

1.2 Disadvantages in Existing System

(1) There is no policy management for file access, so that unauthorized users can also able to access the sensitive data.

(2) There is no encryption and decryption concept, the files stored in the semi-trusted cloud can able to leak the information to others.
(3) There is no structured way to access the file for personal & professional purpose.

II. HARDWARE IMPLEMENTATION

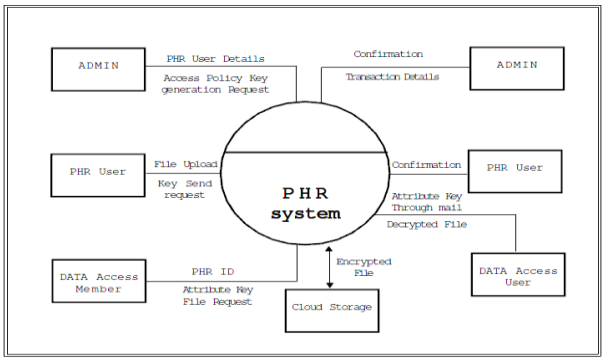

Figure 1: Block diagram

This paper, endeavor's to study the secure sharing of PHRs stored on semi-trusted servers controlled by the patients and focus on the issues of managing the keys. The proposed work for patient-centric secure sharing of PHR is shown in fig (1). To provide security to the personal health data stored on a cloud server, we use attribute-based encryption (ABE). Using ABE, policy management table is prepared based on the attributes of users or data. This allows patients to share their PHR among a set of users selectively by encrypting different files with different attributes. The complexity of maintaining keys is linear with the number of users. To reduce the key management issues, dynamic policy updates and on-demand revocation in large-scale PHR system we make the following main contributions:

(1) We propose ABE-based work for secure sharing of PHRs in cloud computing controlled by patients. To reduce the key management issues, we make a division of users in the system into two types of domains, namely public and personal. Professional users are managed by attribute authorities in the public domain. But in case of personal domain, each owner only needs to manage the keys of a small number of users.

(2) In the public domain, we use multi-authority ABE (MA-ABE) to improve the security and avoid key management problem. Each attribute authority (AA) for example hospitals, insurance companies etc manages a set of users based on role attributes. In the personal domain, patients directly provides access rights for personal users and encrypt the files using attributes. Examples of personal domain are family member and friends and for public domain are doctors, pharmacists and researchers, insurance company people, etc.

Files / Attribute	Friends	Hospitals	Insurance	Emergency
Personal	Yes	No	No	Yes
Medical_History	Yes	Yes	No	Yes
Current_Exams	No	Yes	Yes	Yes
Insurance	No	Yes	Yes	Yes
Sensitive	No	No	No	Yes

Table 1: Access Policy Table

2.1. Module Description

The block diagram of PHR system is shown in fig (1). The PHR system is the one in which encryption and decryption of the health records and other files takes place. PHR owner or patient uploads his/her files which are encrypted in PHR system and then stored in the cloud storage. The PHR owner will send the attributed key through mail to the data access users such as hospitals, friends and insurance using which they will be authorized to access the files according to policy management set by the PHR owner. Hence the decrypted file will be downloaded into the data access user's system. Different users will be given different keys. Admin is responsible for managing PHR owner details, transaction details, policy management etc.

PHR Owners or patients have to upload files on cloud server in encrypted form so that security of data and access rights of the data is achieved. As we discussed earlier various types of users are given access to view the files based on policy management as indicated in the policy management table (1). This system has three types of users Admin, PHR Owner & Data Access Member.

Sample Files used in this system

- Personal File
- Medical History
- Current Medical Examination
- Insurance Details
- Sensitive Details

Attribute Types in this System

- Friends
- Hospitals
- Insurance
- Emergency

Admin Session

- Login Module
- Attribute Values
- Policy Management
- PHR Owners
- Key Generation
- Transaction Details
- Change Password

PHR Owners Session

- Login Module
- File Upload
- Attributed Key Distribution through mail
- Transaction Details

Data Access Member Session

- File Download

The module require the Following software

- J2EE – Servlet, JSP
- JDK 1.6
- Database Server My-SQL Server
- TomCat 6.0

III. SOFTWARE IMPLEMENTATION

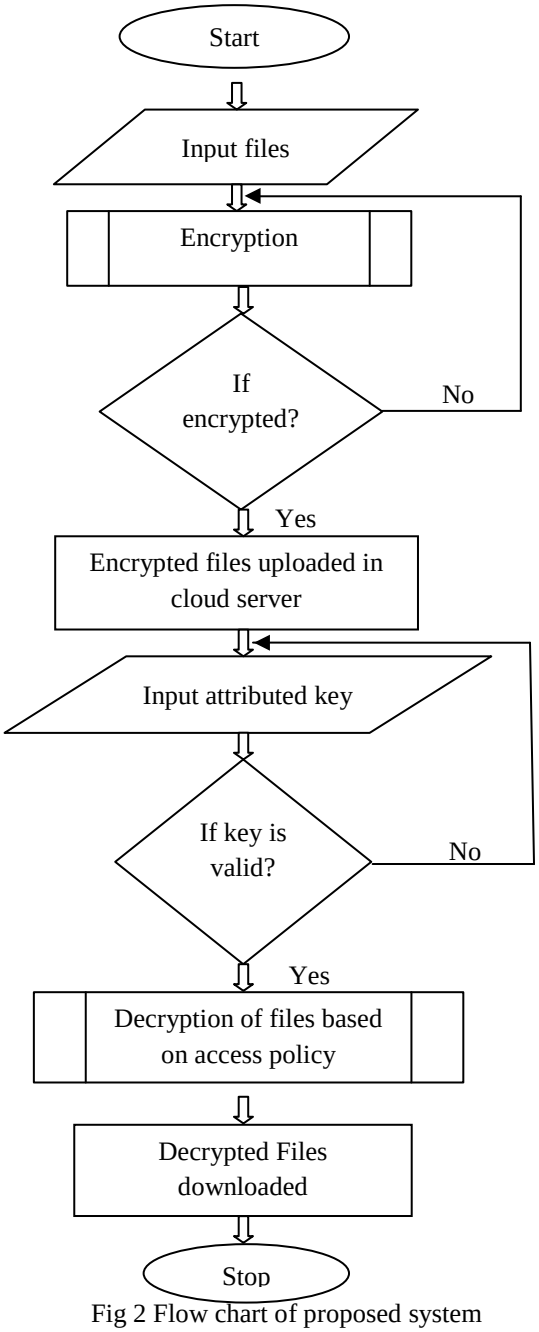

Fig 2 Flow chart of proposed system

3.1 Algorithm for proposed system

1. PHR owner will input the files he/she wants to share.
2. Encryption of input files is done using DES algorithm.
3. Encrypted files uploaded into the cloud server.
4. User or data access member input the attributed key.
5. Verification of the key for its validity.
6. If the key is invalid, files will not decrypt.
7. If the key is valid, files will be decrypted.
8. Decrypted files will be downloaded into the user's local system.

IV. ENCRYPTION AND DECRYPTION

Encryption is the process in which, a message in its original form (plaintext) is converted (encrypted) into an unintelligible form (cipher text) by a set of procedures known as an encryption algorithm (cipher) and a variable, called a key. The cipher text is transformed (decrypted) back into plaintext using the encryption algorithm and a key. In this scheme, using Data Encryption Standard (DES) algorithm data is encrypted and decrypted. DES is the block cipher, an algorithm that takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another cipher text bit string.

In DES, the block size is 64 bits. DES also uses a key to customize the transformation, so that decryption can supposedly only be performed by those who know the particular key used to encrypt. The key consists of 64 bits; however, only 56 of these are actually used by the algorithm. Eight bits are used solely for checking parity, and are thereafter discarded. Hence the effective key length is 56 bits, and it is always quoted as such. DES uses the two basic techniques of cryptography - confusion and diffusion. At the simplest level, diffusion is achieved through numerous permutations and confusion is achieved through the XOR operation.

Fig (3) gives the general description of DES encryption algorithm. While fig (5) shows the simplified one for the same.

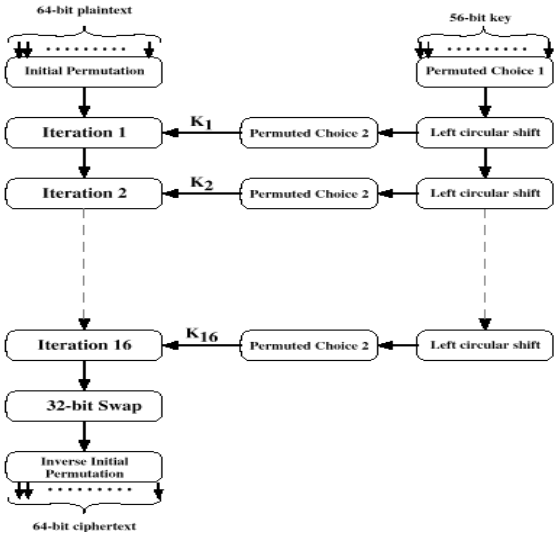

Fig 3. General description of DES algorithm

The basic process in enciphering a 64-bit data block and a 56-bit key using the DES consists of:

- An initial permutation (IP)
- 16 rounds of a complex key dependent calculation f
- A final permutation, being the inverse of IP

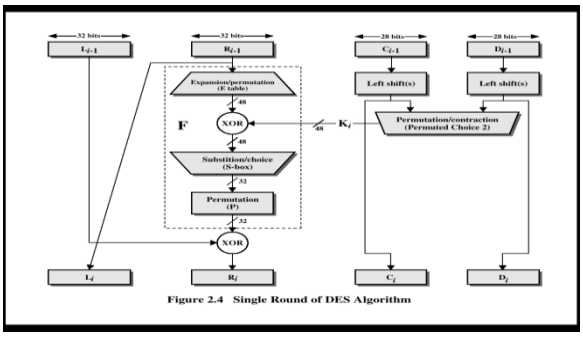

Fig 4 Single Round of DES algorithm

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Fig (4) shows the single round of DES algorithm.
The overall processing at each iteration:

- Li = Ri-1
- Ri = Li-1 ⊗ F(Ri-1, Ki)

Where Li is the left half of 64 bit block and Ri is the right half of the same 64 bit block and F(Ri-1, Ki) is the fiestal function of Ri-1 and the 56-bit key Ki. ⊗ is the XOR operation.

The plain text is encrypted by the following steps in DES algorithm.
1. The plaintext is divided into 64-bit blocks. Each block is worked independently through 16 iterations of the algorithm.
2. At the same time, the 56-bit key is divided in half. In each iteration, the bits in each half are shifted to the left to change the key values.
3. A 64-bit block is divided in half and the right half is combined with the two key halves created in step 2.
4. The results of step 3 are converted again using permutation, then the results are combined with the left half of the 64-bit block.

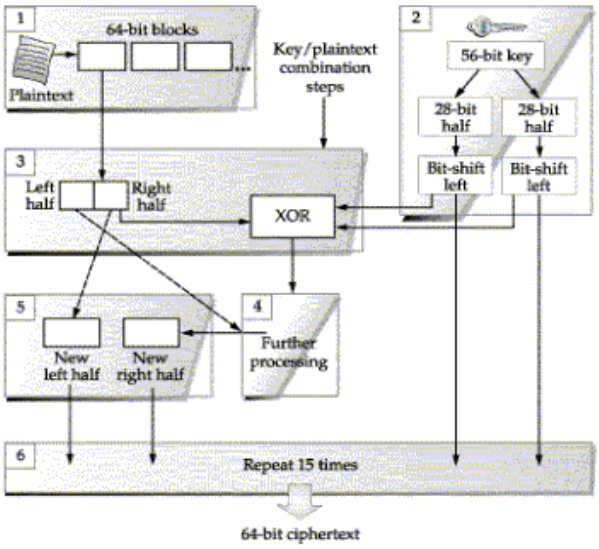

Figure 5. DES encryption algorithm (simplified)

5. The results of the above steps become the new right half. Now the next iteration for the same 64-bit block is ready to start. The right half from the previous iteration is brought down to become the new left half. Also, the left and right halves of the key are bit-shifted left and combined to create a new key..
6. The process repeats again using the new left half and new right half for 15 more iterations. This produces the first 64-bit block of cipher text.

The next 64-bit block is processed using the same procedure. There are 16 identical stages of processing, termed rounds. Before the main rounds, the block is divided into two 32-bit halves and processed alternately; this crisscrossing is known as the Feistel scheme. The Feistel structure ensures that decryption and encryption are very similar processes — the only difference is that the sub-keys are applied in the reverse order when decrypting. The rest of the algorithm is identical. This greatly simplifies implementation, particularly in hardware, as there is no need for separate encryption and decryption algorithms. The *F-function* scrambles half a block together with some of the key. The output from the F-function is then combined with the other half of the block, and the halves are swapped before the next round. After the final round, the halves are swapped; this is a feature of the Feistel structure which makes encryption and decryption similar processes.

Encryption and decryption keys are generated using the following formulas.

Encryption: $C = E_K(P)$

Decryption: $P = E_K^{-1}(C)$

E_K is chosen from a family of transformations known as a cryptographic system. The parameter that selects the individual transformation is called the key K, selected from a keyspace K. For a K-bit key the keyspace size is 2^K

V. DATA FLOW CHARTS

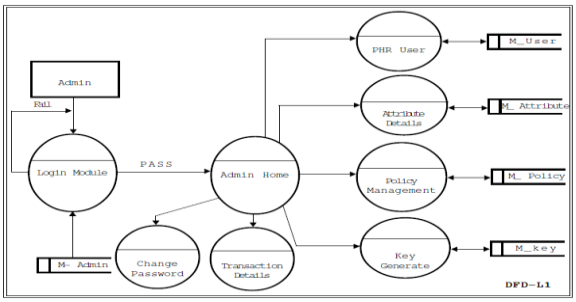

Figure 6. Admin module

Fig (6) shows the data flow diagram of admin module, which shows that the admin is responsible for managing PHR user details, transaction details, policy management, attribute value etc. He can change the access policy any time.

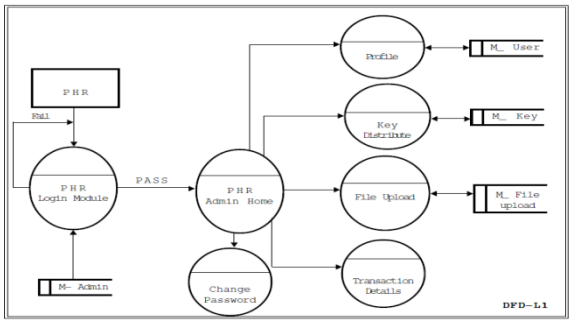

Figure 7. PHR Owner module

Figures (7) give the data flow chart of PHR owner module. The Owner will upload the PHR and other files in the encrypted form into the cloud server and distributes the attributed keys to different users.

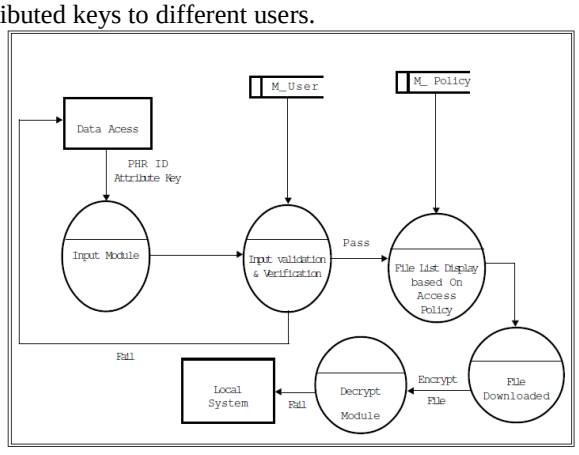

Figure 8. Data Access member module

Figures (8) give the data flow chart of data access member module. The data access member or the user has to enter the attributed key, if it is valid the decrypted files are downloaded into his local system.

VI. SIMULATION RESULTS

Fig (9) shows the profile of the admin which includes the name, email id, contact number and address of the admin.

Fig 9. Admin Profile

Fig 10. Attribute Values

Fig(10) gives the attribute values such as friend, hospital, insurance and emergency.

Fig 11. Policy Management

Fig 12. PHR Owner Details

Fig (12) shows different PHR owner details which includes name, id, address, contact number and email id.

Fig 13. Key Generation

Fig (13) shows the key generation for different PHR owners. Encryption and decryption keys are generated using the following formulas.

Encryption: $C = E_K(P)$

Decryption: $P = E_K^{-1}(C)$

Fig 14. Transaction Details

Fig (14) shows the transaction details which includes the user name, transaction type (file upload or download), file type (text or image), time and date of transaction. Fig (15) shows the PHR owner profile, it consists of owner id, name, email id etc.

Fig 15. PHR Owner Details

Fig 16. File upload

Fig 17. File upload confirmation

16) shows the uploading of a file and fig (17) shows file uploading confirmation

Fig 18. Key distribution

Fig (18) shows the key distribution to friend. The attributed key is distributed to respective users through e-mail.

VII. CONCLUSION

In the proposed scheme, it is possible to achieve secure sharing of personal health records and other files in cloud computing. Patients can have complete control of their own privacy through encrypting their Personal Health Record (PHR) and other files to allow access to selective users. The unique challenge introduced by multiple PHR owners and users such as security and key management complexities are greatly reduced by using DES encryption algorithm that has a key size of 56-bits. As Attribute Based Encryption (ABE) is used to encrypt the PHR data, so that patients can allow access not only to personal users, but also various users from public domains with different professional roles. On-demand user revocation with security is also achieved. Through implementation and simulation, we show that our solution is scalable.

REFERENCES

[1] M. Li, S. Yu, K. Ren, and W. Lou, "Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings," in SecureComm'10, Sept. 2010, pp. 89–106.

[2] H. Lo¨ hr, A.-R. Sadeghi, and M. Winandy, "Securing the e-health cloud," in Proceedings of the 1st ACM International Health Informatics Symposium, ser. IHI '10, 2010, pp. 220– 229.

[3] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized private keyword search over encrypted Personal health records in cloud computing," in ICDCS '11, Jun. 2011.

[4] "The health insurance portability and accountability act."[Online]. Available: http://www.cms.hhs.gov/HIPAAGenInfo/01 Overview.asp

[5] "Google, microsoft say hipaa stimulus rule doesn't apply to them," http://www.ihealthbeat.org/Articles/2009/4/8/.

[6] "At risk of exposure in the push for electronic medical records, concern is growing about how well privacy can be safeguarded," 2006. [Online]. Available: http://articles.latimes.com/2006/jun/26/health/he-privacy26

[7] K. D. Mandl, P. Szolovits, and I. S. Kohane, "Public standards and patients' control: how to keep electronic medical records accessible but private," BMJ, vol. 322, no. 7281, p. 283, Feb. 2001.

[8] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient controlled encryption: ensuring privacy of electronic medical records," in CCSW '09, 2009, pp. 103–114.

[9] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in IEEE INFOCOM'10, 2010.

[10] C. Dong, G. Russello, and N. Dulay, "Shared and searchable encrypted data for untrusted servers," in Journal of Computer Security, 2010.

[11] V. Goyal, O. Pandey, A. Sahai and B. Waters, " Attribute-based encryption for a fine grained access control of encrypted data'' in CCS 06, 2006, pp. 89-98.

[12] M. Li, W. Lou and K. Ren, "Data security and Privacy in wireless body area network," IEEE Wireless Communication Magazine, Feb,

[13] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *ACM CCS*, ser. CCS '08, 2008, pp. 417–426.

[14] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, W. Jonker,"Ciphertext-policy attribute- bas- d threshold decryption with flexible delegation and revocation of user attributes," 2009.

[15] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in

Mrs.Shaheen Taj S.A received her **B.E** degree in **Electronics & Communication Engineering** from Visvesvaraya Technological University, Belgaum, Karnataka State, India in the year 2006. Presently she is pursuing her **M.Tech** in **Digital Electronics and Communication** from AMC Engineering College, Bangalore, which is affiliated to VTU, Belgaum. Her area of interest includes wireless communication, networking and cloud computing,

Prof.PrathibhaKiran. was awarded with **M.Tech** in **Biomedical Signal Processing and Instrumentation** during 2011 from DayanandSagar College, which is affiliated to VTU, Belgaum. Presently working as **Assistant Professor** in AMC Engineering College, Bangalore in the department of **Electronics & Communication Engineering** with one and half year experience. She secured **2'nd rank in M.Tech in the stream of Biomedical Signal Processing and Instrumentation** from **Visvesvaraya Technological University**. It is to her credit that she has already contributed papers at National conference, International conference and International journals. Her Area of research interest includes Signal processing, Wavelets, Advanced digital image processing, Neural Network and Fuzzy logic, Bio-medical Instrumentation and Wireless Communication.

Prof. E Elavarasi received her **B.E** degree in **Electronics & Communication Engineering** from Visvesvaraya Technological University, Belgaum, Karnataka State, India in the year 2004 and **M.Tech** in **Digital Electronics and Communication** from the same University in the year 2007. She has also received **MBA** degree in **HR & Finance** from Bangalore University in the year 2011. She has 6.3 years of teaching experience and presently she is working as Assistant Professor in the Dept. of ECE, AMC Engineering College, Bangalore. She has published several research papers in International and National conferences. Her areas of interest include Signal Processing, Communication, Multirate, Error Control Coding, Digital Circuits and Logic Design etc.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication