

ANALYSIS AND COMPARISON OF SYMMETRIC KEY CRYPTOGRAPHIC ALGORITHMS BASED ON VARIOUS FILE FEATURES

Ranjeet Masram, Vivek Shahare, Jibi Abraham, Rajni Moona
Department of Computer Engineering and Information Technology,
College of Engineering Pune, India

ABSTRACT

For achieving faster communication most of confidential data is circulated through networks as electronic data. Cryptographic ciphers have an important role for providing security to these confidential data against unauthorized attacks. Though security is an important factor, there are various factors that can affect the performance and selection of cryptographic algorithms during the practical implementation of these cryptographic ciphers for various applications. This paper provides analysis and comparison of some symmetric key cryptographic ciphers (RC4, AES, Blowfish, RC2, DES, Skipjack, and Triple DES) on the basis of encryption time with the variation of various file features like different data types, data size, data density and key sizes.

KEYWORDS

Cipher Algorithms, data type, data size, data density and encryption time

1. INTRODUCTION

Cryptography is a powerful tool used to protect the information in computer systems. When a browser is used for home banking, numbers of cryptographic algorithms are being used to protect data send to the bank. When someone login to computer, the password is protected by cryptographic hash functions. An email it is often encrypted using SSL while sending it. [9]. In cryptography original message is basically encoded in some non readable format. This process is called encryption. The only person who knows how to decode the message can get the original information. This process is called decryption. On the basis of key used, cipher algorithms are classified as asymmetric key algorithms, in which encryption and decryption is done by two different keys and symmetric key algorithms, where the same key is used for encryption and decryption [8]. On the basis of the input data, cipher algorithms are classified as block ciphers, in which the size of the block is of fixed size for encryption and stream ciphers in which a continuous stream is passed for encryption and decryption [9].

A data file formats represents the standard for encoding the information to be stored in computer file. There are file formats like textual, image, audio and video data file formats. Textual data formats are ANSII, UNICODE (16 & 32 bit little and big Endian and UTF-8). ANSII is encoding scheme for 128 characters basically made for English alphabets. It contains alphabets a-z and A-Z, numbers 0-9 and some special characters. In Unicode standard unique numbers are provided for every character independent of platform. Image file formats are JPEG, TIFF, BMP, GIF and PNG [10]. JPEG is image file format for digital images that uses lossy compression method. TIFF and BMP are image file format that are used to store images of raster graphics. GIF image is similar to image format of bitmap images. GIF uses LZW (Lempel-Ziv-Welch) technique of

compression and for each image it can support up to 8 bits/pixel. PNG is alternative to GIF image file format and allows more compression than GIF. Audio file formats are WAV, AIFF, M4A, MP3 and WMA. WAV and AIFF are usually uncompressed audio file format. M4A (audio) uses Apple Lossless compression format but often it is compressed with Advance audio coding (lossy). MP3 and WMA are lossy compression audio formats. Video file formats are AVI, M4V, MPEG and WMV etc. AVI format contains the data (audio and video data) file container; which allows audio-with-video playback synchronously. M4V and MP4 are very similar format, but M4v can be protected by Digital Rights Management copy protection. MPEG contains compressed audio and visual digital data. WMV is compressed video format developed by Microsoft.

Density of data represents the amount of different information present in the data file [10]. File is said to be dense file if file size is less and content is more. For example if there are two file X and Y both containing 2000 words and having sizes 50kb and 200kb respectively, then file X is denser. The more the information, the dense is the data and lesser the information, sparse is the data. Sparse file is a file that contains most of the empty spaces and attempts to use the computer space more effectively.

Data size is space occupied by a file on a disk. Audio, video takes more space on disk than textual files as they contain multimedia information. Key size in cryptography represents the size of key file in bits. For example AES is having key sizes 128, 192 and 256 bits.

The main objective of this paper is to analyze time taken for encryption by various cryptographic algorithms for parameters like data type, data size, data density and key size.

2. CRYPTOGRAPHIC ALGORITHMS

This section provides information about the various symmetric key cryptographic algorithms to be analyzed for performance evaluation, to select the best algorithm with appropriate parameter suitable to provide security for data.

Symmetric key cryptographic ciphers come in two varieties, stream and block ciphers. Stream ciphers works on a stream of bits or bytes. Stream ciphers are used for securing data of terminal and wireless applications. Block ciphers performs encryption or decryption on fixed size block of data. The plaintext is not always in multiple of block size, therefore padding bits are needed to compensate partially filled block. The padding scheme defines how the plaintext is filled with data for last block. In network applications block ciphers are used for transmission of files of huge sizes which require high security. Deciphering ciphertext without knowing the key is called cryptanalysis. Cryptanalysis of block ciphers is difficult compared to stream ciphers [9]. Hence in most of the applications, block ciphers are used for providing better security than stream ciphers.

Mathematically linked series of operations are used in Substitution-permutation network in cryptography to construct the block of the Symmetric key cryptographic block cipher. In SP network key and plain text are taken as an input and number of alternating rounds of S-Box substitution and permutation are applied to get a single ciphertext block. The reverse process is done for decryption of the blocks. Symmetric key cryptographic ciphers have different structures that are used to construct the block of the different Symmetric key block ciphers. There are symmetric key structures like Feistel network, Substitution-permutation network etc. In the case of Fiestel network, the encryption and decryption process of the block are almost similar to each other, except it requires the reversal of key schedule. Iteration is a characteristic feature of Fiestel network cipher as an internal function knows as round function.

There are different symmetric cryptographic algorithms in the literature [8] [9]. Out of them, the algorithms listed in the Table 1 are selected for detailed study in case of data from health field. Except RC4, all the selected algorithms are of block type, since most of the health related data are to be provided with high security.

The various features of the cryptographic algorithm are listed in Table 1.

Table 1. Cryptographic Algorithms Information

Algorithm Name	Structure	Key Size (In bits)	Rounds	Cipher Type
AES	Substitution-permutation network	128, 192, 256	10, 12, 14	Block
DES	Balanced Feistel network	56	16	Block
Triple DES	Feistel network	112, 168	48	Block
RC2	Source-heavy Feistel network	40 to 1024	18	Block
Blowfish	Feistel network	32 to 448	16	Block
Skipjack	Unbalanced Feistel network	80	32	Block
RC4	----	40 to 2048	256	Stream

3. RELATED WORK

This section provides the information and results which are obtained from the numerous sources. Cryptographic algorithms have been compared with each other for performance evaluation on basis of throughput, CPU Memory utilization, energy consumption, attacks, Encryption time, Decryption time etc.

In [3] the author compared AES and RC4 algorithm and the performance metrics were encryption throughput, CPU work load, memory utilization, and key size variation and encryption and decryption time. Results show that the RC4 is fast and energy saving for encryption and decryption. RC4 proved to be better than AES for larger size data. In [2] author compared AES and DES algorithms on image file, MATLAB software platform was used for implementation of these two cipher algorithms. AES took less encryption and decryption time than DES. In [4] the author compared cipher algorithms (AES, DES, Blowfish) for different cipher block modes (ECB, CBC, CFB, OFB) on different file sizes varying from 3kb to 203kb. Blowfish algorithm yield better performance for all block cipher modes that were tested and OFB block mode gives better performance than other block modes. In [7] the author talks about comparison between three algorithms (DES, Triple DES, Blowfish) on processing time. They found, that the key generation time for all these three algorithms is almost same but there is a difference in time taken by CPU for encryption. On SunOS platform Blowfish seem to be fastest, followed by DES and Triple DES respectively. They analyzed CPU execution time for generating the secret key, encryption and decryption time on 10MB file. In [6] the author compared cipher algorithms (AES, DES, 3-DES and Blowfish) for varying file size and compared the encryption time on two different machines Pentium-4, 2.4 GHz and Pentium-II 266 MHz in ECB and CFB Mode. The author concluded that Blowfish is fastest followed by DES and Triple DES and CFB takes more time than ECB cipher block mode.

4. PROPOSED WORK

From the related works, it is realized that none of the work did a very detailed analysis of the performance of various symmetric algorithms on various parameters on different type of files, especially the files which are used for medical health related data.

The main objective of this paper is to analyze the time taken for encryption by various cryptographic algorithms for parameters like data type, data size, data density and key size in order to select the most suitable cryptographic algorithm for encryption.

5. EXPERIMENTAL SETUP AND TESTING

The execution results are taken on machine having Intel® Core™ i7-2600 (3.40 GHz) processor with Intel® Q65 Express 4 GB 1333 MHz DDR3 (RAM) and Ubuntu 12.04 LTS operating System. The java platform (openjdk1.6.0_14) is used for implementation. JCA (Java Cryptography Architecture) and JCE (Java Cryptography Extension) are used for cipher algorithm implementation. The JCA is a major platform that contains "provider" architecture and the set of APIs for encryption (symmetric ciphers, asymmetric ciphers, block ciphers, stream ciphers), message digests (hash), digital signatures, certificates and certificate validation, key generation and secure random number generation. Here we have used sun and Bouncy Castel provider for implementing cryptographic algorithms.

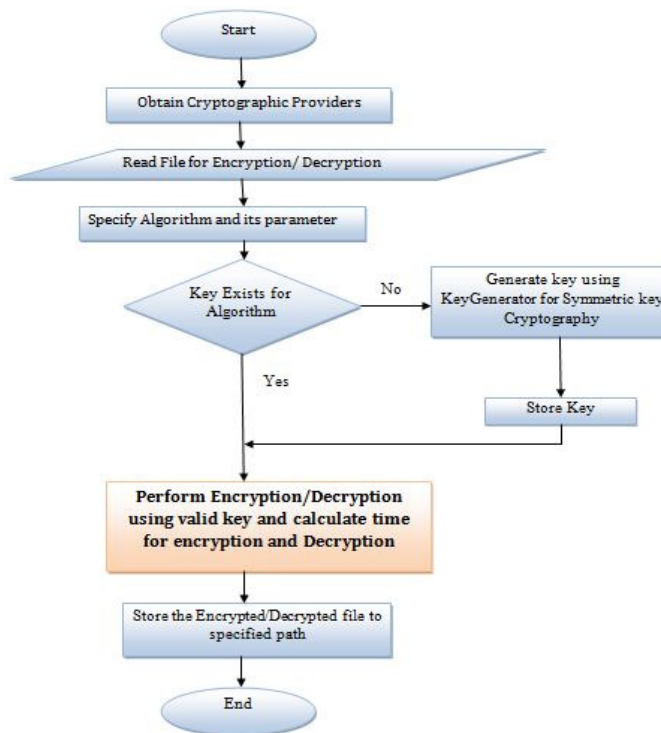


Figure 1: Symmetric key cryptography flowchart

The flow chart in Figure 1 gives the brief idea about the execution of the program flow control. There are various security providers available in java like sun, bouncy castle, cryptix etc. Initially obtain one of these security providers. Read the input file for encryption or decryption. The program checks that the key is present for algorithm or not. If the key is not present,

KeyGenerator is used to generate the key for performing the cipher operation. KeyGenerator is a class provided by java security API for generating the key of the symmetric ciphers. The generated key is stored to the specified path. If the key is already present at the specified path, that key is used. A timer is started before encrypting or decrypting a file. After encryption or decryption is performed, the timer is stopped. The output file is then written to specified location. And the results are written to result sheet. During any of the cipher operation the I/O time is not taken. Due to this pure encryption time is obtained for evaluating the performance of a particular cipher algorithm.

The brief analysis of different symmetric key cryptographic algorithm for various parameters is as follows:

Case Study 1: Files with different Data types.

This case study has taken to check whether the encryption has dependency on type of data. Different data type files like audio, image, textual and video of nearly 50MB in size are chosen and encryption time of different cipher algorithms is calculated for these data types. For all executions of a specific cipher algorithm, varying parameter is data type and constant parameters are key size and block cipher mode. Key size and block mode are at kept at bare minimal parameters. The key size of AES, DES, 3-DES, RC2, Blowfish, Skipjack, and RC4 are kept at minimum values as 128, 56, 112, 40, 32, 80 and 40 bits respectively. Block cipher mode used is ECB with PKCS#5 padding scheme. Figure 2 shows the execution time of the algorithms for different data type files.

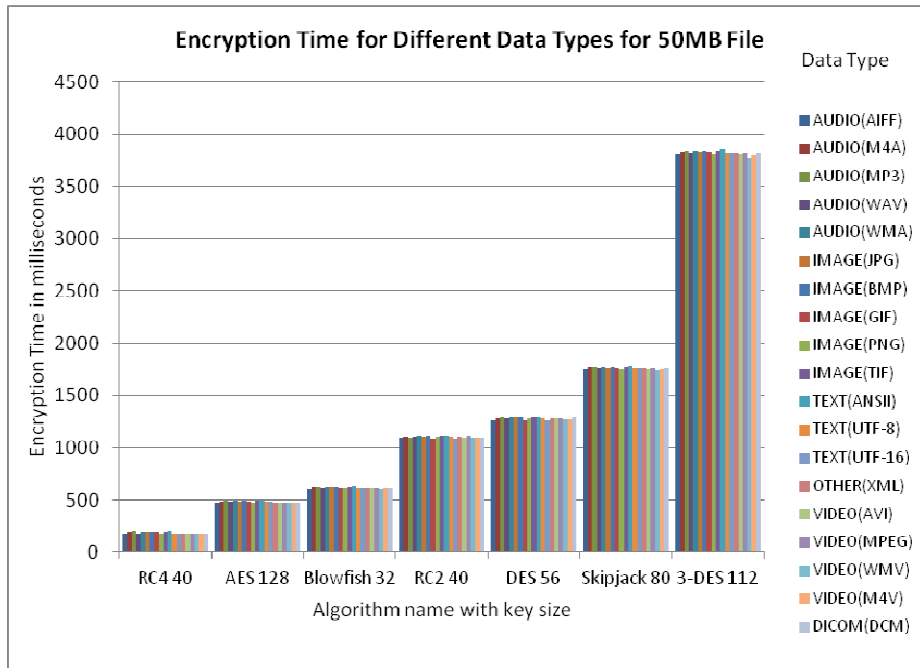


Figure 2. Encryption time Vs Cipher Algorithm for files of different data type

Observation: In Figure 2 it can be clearly seen that encryption time for all the data type is almost same. The result shows that the encryption time does not vary according to the type of the data. Encryption depends only on the number of bytes in the file and not on the type of file. Encryption time of AES is quiet low compared to other block ciphers. RC4 with key size 40 is fastest among the cipher algorithms tested.

Case Study 2: Data files of same type with different sizes.

This case study is taken to ensure once again the observations obtained in case study 1. Case study 1 revealed that encryption time depends on number of bytes in the file. To ensure this another study is made in which different files (BMP and FLV) of same types but different sizes are given for encryption and their encryption time is calculated. For all executions key size and block mode are kept at bare minimal parameters. Table 2 gives the details about the files used for all executions and Figure 3 and 4 show the execution results for BMP and FLV file formats of different sizes respectively.

Table 2. Execution Parameters for files of different size.

File Type	Varying Parameters (Data Size)	Constant Parameters
BMP	10.7MB, 50MB, 100MB	Data Type, Key size
FLV	50MB, 100MB, 482MB	

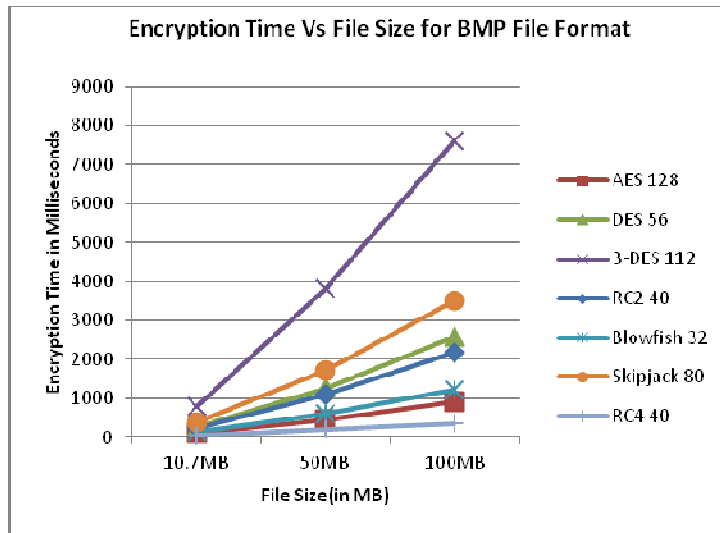


Figure 3. File size Vs Encryption time for BMP file of different sizes.

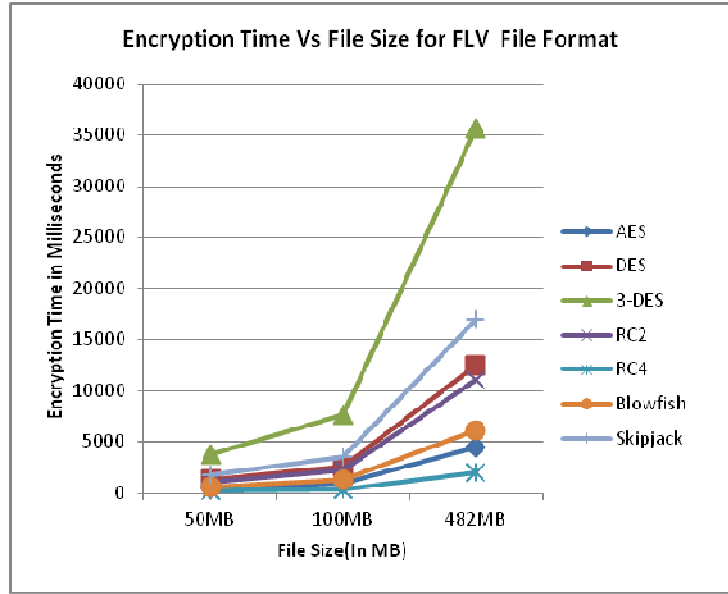


Figure 4. File size Vs Encryption time for FLV file of different sizes.

Table 3. Encryption time for files of different sizes

File Type	Size (In MB)	Encryption Time in Millisecond						
		AES	DES	3-DES	RC2	Blowfish	Skipjack	RC4
		128	56	112	40	32	80	40
BMP	10.7	101	272	788	238	133	381	40
	50	455	1253	3804	1095	614	1729	198
	100	909	2595	7628	2189	1223	3505	372
FLV	50	456	1268	3810	1112	629	1731	196
	100	918	2586	7631	2224	1267	3515	360
	482	4518	12529	35654	11038	6087	16941	1972

Observation: For each encryption algorithm same parameters are used for files of different sizes. Table 3 shows encryption time of different sizes of files of same type. From the results in Table 3 and Figure 3 and 4 we can find that the result for different size of data varies proportional to the size of data file. Encryption time increases as file size increases in multiples of data size.

Case Study 3: File with different data densities.

This case study is taken to check whether the encryption depends on density of data or not. Encryption rate is evaluated for the two different data density file; a sparse file of 69MB and a dense file of 58.5MB. For a cipher algorithm, key size and block mode are kept at bare minimal parameters. The results of execution are shown in Table 4.

Table 4. Execution rate for sparse and dense data file

Algorithm Name	Dense (61392454 Bytes) AIFF file		Sparse (72000118 Bytes) AIFF file	
	Encrypt Time(ms)	Encryption Rate(MB/s)	Encrypt Time(ms)	Encryption Rate(MB/s)
AES 128	539	108.62	632	108.64
DES 56	1535	38.14	1800	38.14
3-DES 112	4363	13.42	5074	13.53
RC2 128	1283	45.63	1518	45.23
Blowfish 128	725	80.75	852	80.59
Skipjack 128	2040	28.70	2384	28.80
RC4 128	216	271.05	251	271.56

Observation: Encryption rate for sparse and dense file has been calculated. The Table 4 shows that the encryption time is not affected by density of data in a file. The variation in time with respect to different algorithms follows the same pattern for both sparse and dense files. The encryption rate for a particular cipher algorithm remains the same, even if the file is sparse or dense. It depends on only the number of bytes in the file.

Case Study 4: Encryption Algorithms with different key sizes

This case study is to analyze the effect of changing the size of encryption key on encryption time. BMP file of 50.5MB is taken and different cipher algorithms are executed for different size of keys supported by them in ECB mode with PKCS#5 padding scheme. The various key sizes mentioned in Table 1 are used during experimentation. Figure 5 shows the result of execution for key size variation.

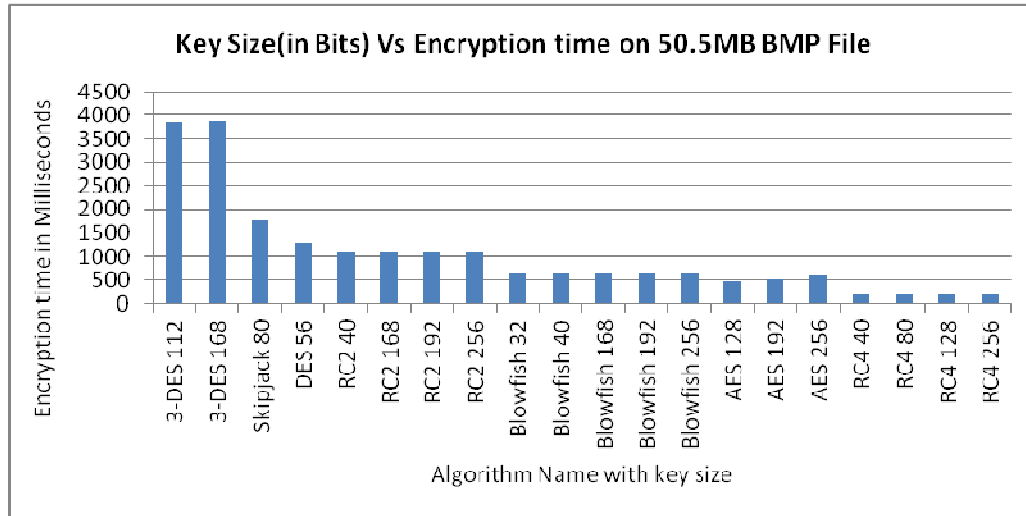


Figure 5. Variation of key sizes for different cipher Algorithms

Obsevation: The execution results show that for all ciphers algorithms, the encryption time varies with the change in the size of the of the key. Encryption time increases with increase in key size for block ciphers. The variation in time is very small. AES dominates in the block cipher. RC4 is fastest among all algorithms tested.

6. CONCLUSION

In this paper different symmetric key algorithm have been analyzed for various file features like different data type, data density, data size and key size, and analyzed the variation of encryption time for different selected cipher algorithms. From the simulated results it is concluded that encryption time is does not dependent upon data type and date density of the file. The research reveled that; encryption only depends upon the number of bytes present in the file. It also reveled that encryption time and data size is proportional to each other. As the size of data increase the encryption time also increase proportional to data size and vice versa. For all block cipher algorithms that are analyzed, with increase in key size, encryption time also increases, but reduces with increase in key size for stream cipher like RC4. AES is appears to be fastest block cipher with encryption rate of 108MB/sec at bare minimal parameter, but RC4 stream cipher with encryption rate of 270MB/sec comes out to be fastest among all analyzed cipher algorithms.

ACKNOWLEDGEMENT

We would like to thanks C-DAC, Pune for all providing funding to the project. We would also like to thanks Dr. P.K. Sinha, Mrs Gaur Sunder, Mr. Prashant Bendale and Mrs. Sayali Pophalkar for their support during the project.

REFERENCES

- [1] AL.Jeeva1, Dr.V.Palanisamy and K.Kanagaram, "Comparative Analysis of Performance Efficiency and Security Measures of Some Encryption Algorithms", *International Journal of Engineering Research and Applications (IJERA)*, Vol. 2, Issue 3, May-Jun 2012, pp.3033-3037.
- [2] S. Soni, H. Agrawal, M. Sharma, "Analysis and comparison between AES and DES Cryptographic Algorithm", *International Journal of Engineering and Innovative Technology*, Vol 2, Issue 6, December 2012, pp.362-365.
- [3] Nidhi Singhal and J.P.S.Raina, "Comparative Analysis of AES and RC4 Algorithms for Better Utilization", *International Journal of Computer Trends and Technology*, Vol 2, Issue 6, July-Aug 2011, pp.177-181.
- [4] Jawahar Thakur and Nagesh Kumar, "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis", *International Journal of Emerging Technology and Advanced Engineering*, Vol 1, Issue 2, December 2011, pp.6-12.
- [5] Allam Mousa and Ahmad Hamad, "Evaluation of the RC4 Algorithm for Data Encryption", *International Journal of Computer Science & Applications*, Vol 3, Issue 2, June 2006, pp.44-56.
- [6] Aamer Nadeem, Dr M. Younus Javed, "A Performance Comparison of Data Encryption Algorithms", *First International Conference on IEEE Information and Communication Technologies (ICICT)*, Vol 1, Issue 6, 27-28 Aug. 2005, pp 84-89.
- [7] Kofahi, N.A., Turki Al-Somani, Khalid Al-Zamil, "Performance evaluation of three Encryption/Decryption Algorithms", *IEEE 46th Midwest Symposium on Circuits and Systems*, Vol 2, Issue 1, 30-30 Dec. 2003, pp. 790-793.
- [8] Jonathan Knudsen, *Java Cryptography*, 2nd Edition, O'Reilly, 2011.
- [9] Behrouz A. Forouzan, Debdeep Mukhopadhyay, *Cryptography and Network Security*, 2nd Edition, Tata McGraw Hill, 2012.
- [10] John Miano, *Compressed Image File Formats*, 1st Edition, Addison Wesley Longman, Inc, 1999.

Authors

Ranjeet Masram is M tech Student in Computer Engineering from College of Engineering, Pune (India). He is appointed as JRF for Joint project on Medical data Security between C-DAC, Pune and College of Engineering, Pune for a period of one year



Vivek Shahare received his Bachelor Degree in Computer Science and Engineering from Government College of Engineering, Amravati (India). He is appointed as JRF for Joint project on Medical data Security between C-DAC, Pune and College of Engineering, Pune for a period of one year.



Dr. Jibi Abraham is Professor at College of Engineering, Pune. She received her Doctor of Philosophy (PhD) in Computer Engineering from Visvesvaraya Technological University. She is the Principal Investigator from COEP for Joint project on Medical data Security between C-DAC, Pune and College of Engineering



Mrs. Rajni Moona was project engineer at IIT Kanpur. She was Visiting faculty at International Institute of Information Technology. She is the co-investigator for Joint project on Medical data Security between C-DAC, Pune and College of Engineering

