Volume 3, Issue 9, September 2018 ISSN (online): 2456-0006
International Journal of Science Technology Management and Research
Available online at: www.ijstmr.com

Encryption & Decryption of Data Using Huffman Compression and Exclusive Disjunction

Sangeeta Sambha Uikey
M Tech (CSE) IV Sem.
Lord Krishna College of Technology
Indore M P. India

Vijay Kumar Verma
Asst. Professor CSE department
Lord Krishna College of Technology
Indore M P. India

Abstract: ***Cryptography converts data into a format that is unreadable for an unauthorized user, allowing it to be transmitted without unauthorized entities decoding it back into a readable format, thus compromising the data. Cryptography allows you to have confidence in your electronic transactions. Encryption is used in electronic transactions to protect data such as account numbers and transaction amounts, digital signatures replace handwritten signatures or credit card authorizations, and public-key encryption provides confidentiality. Key management is an important aspect in encryption that allows you to apply common encryption policies across all data on all managed devices. In this paper we proposed new private key cryptography techniques which are based on Huffman compression. We also apply XOR operation to make more effective Ciphertext.***

Keywords: Cryptography, Huffman compression, XOR, private key, public key,

I. INTRODUCTION

Cryptography is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Cryptography is associated with the process of converting ordinary plain text into unintelligible text and vice-versa. It is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Cryptography not only protects data from theft or alteration, but can also be used for user authentication.

Modern cryptography concerns itself with the following four objectives:
1) Confidentiality (the information cannot be understood by anyone for whom it was unintended)
2) Integrity (the information cannot be altered in storage or transit between sender and intended receiver without the alteration being detected)
3) Non-repudiation (the creator/sender of the information cannot deny at a later stage his or her intentions in the creation or transmission of the information)
4) Authentication (the sender and receiver can confirm each others identity and the origin/destination of the information)

TYPES OF CRYPTOGRAPHIC TECHNIQUES

Three types of cryptographic techniques used in general.
1. Symmetric-key cryptography
2. Hash functions.
3. Public-key cryptography

Symmetric-key Cryptography: Both the sender and receiver share a single key. The sender uses this key to encrypt plaintext and send the cipher text to the receiver. On the other side the receiver applies the same key to decrypt the message and recover the plain text.
Public-Key Cryptography: This is the most revolutionary concept in the last 300-400 years. In Public-Key Cryptography two related keys (public and private key) are used. Public key may be freely distributed, while its paired private key, remains a secret. The public key is used for encryption and for decryption private key is used.
Hash Functions: No key is used in this algorithm. A fixed-length hash value is computed as per the plain text that makes it impossible for the contents of the plain text to be recovered. Hash functions are also used by many operating systems to encrypt passwords.

PROBLEM WITH CRYPTOGRAPHY

The keys used in modern cryptography are so large, in fact, that a billion computers working in conjunction with each processing a billion calculations per second would still take a trillion years to definitively crack a key [source: Dartmouth College]. This isn't a problem now, but it soon will be. Current computers will be replaced in the near future with quantum computers, which exploit the properties of physics on the immensely small quantum scale. Since they can operate on the quantum level, these computers are expected to be able

to perform calculations and operate at speeds no computer in use now could possibly achieve. So the codes that would take a trillion years to break with conventional computers could possibly be cracked in much less time with quantum computers. This means that secret-key cryptology (SKC) looks to be the preferred method of transferring ciphers in the future.

But SKC has its problems as well. The chief problem with SKC is how the two users agree on what secret key to use. If you live next door to the person with whom you exchange secret information, this isn't a problem. All you have to do is meet in person and agree on a key. But what if you live in another country? Sure, you could still meet, but if your key was ever compromised, then you would have to meet again and again.

It's possible to send a message concerning which key a user would like to use, but shouldn't that message be encoded, too? And how do the users agree on what secret key to use to encode the message about what secret key to use for the original message? The problem with secret-key cryptology is that there's almost always a place for an unwanted third party to listen in and gain information the users don't want that person to have. This is known in cryptology as the key distribution problem.

II. LITERATURE REVIEW

In 2012 Monika Agrawal and Pradeep Mishra proposed "A Comparative Survey on Symmetric Key Encryption Techniques". Cryptography is an area of computer science which is developed to provide security for the senders and receivers to transmit and receive confidential data through an insecure channel by a means of process called Encryption/ Decryption. Cryptography ensures that the message should be sent without any alterations and only the authorized person can be able to open and read the message. A number of cryptographic techniques are developed for achieving secure communication. There are basically two techniques of cryptography- Symmetric and Asymmetric. They present a detailed study of most of the symmetric encryption techniques with their advantages and limitations over each other. DES, AES, and Blowfish. Symmetric Key algorithms run faster than Asymmetric Key algorithms such as RSA etc. and the memory requirement of Symmetric algorithms is lesser than Asymmetric encryption algorithms. Further, the security aspect of Symmetric key encryption is superior than Asymmetric key encryption. The comparison table of popular encryption algorithms clearly shows the supremacy of Blowfish algorithm over DES, AES and Triple DES on the basis of key size and security. The F function of Blowfish algorithm provides a high level of security to encrypt the 64 bit plaintext data. Also the Blowfish algorithm runs faster than other popular symmetric key encryption algorithms[1] .

In 2013 Obaida Mohammad Awad Al-Hazaimeh proposed "A New Approach for Complex Encrypting and Decrypting Data". They proposed a New Approach for Complex Encrypting and Decrypting Data" which maintains the security on the communication channels by making it difficult for attacker to predicate a pattern as well as speed of the encryption / decryption scheme. They introduced a new approach for complex encrypting and decrypting data. Although there have been many researchers on the cryptography, but most of the existing algorithms have several weaknesses either caused by low security level or increase the delay time due the design of the algorithm itself. The proposed algorithm have been tested against different known attacks and proved to be secure against them. Therefore, it can be consider as a good alternative to some applications because of the high level of security and average time needed to encrypt and decrypt data using a proposed algorithm is much smaller than AES algorithm[2].

In 2013 Mansoor Ebrahim , Shujaat Khan proposed " Symmetric Algorithm Survey: A Comparative Analysis" .They presents a comprehensive comparative analysis of different existing cryptographic algorithms (symmetric) based on their Architecture, Scalability, Flexibility, Reliability, Security and Limitation that are essential for secure communication (Wired or Wireless). In this paper a detailed analysis of symmetric block encryption algorithms is presented on the basis of different parameters. The main objective was to analyze the performance of the most popular symmetric key algorithms in terms of Authentication, Flexibility, Reliability, Robustness, Scalability, Security, and to highlight the major weakness of the mentioned algorithms, making each algorithm's strength and limitation transparent for application. During this analysis it was observed that AES (Rijndael) was the best among all in terms of Security, Flexibility, Memory usage, and Encryption performance. Although the other algorithms were also competent but most of them have a tradeoff between memory usage and encryption performance with few algorithms been compromised[3].

In 2014 Mitali, Vijay Kumar and Arvind Sharma proposed "A Survey on Various Cryptography Techniques". The survey is done on some of the more popular and interesting cryptography algorithms currently in use and their advantages and disadvantages are also discussed. They paper provides a fair performance comparison between the various cryptography algorithms on different settings of data packets. They present the performance evaluation of selected symmetric algorithms. The selected algorithms are AES, 3DES, Blowfish and DES. The presented simulation results show the numerous points. Firstly it was concluded that Blowfish has better performance than other algorithms followed by AES.3DES has least efficient of all the studied algorithms[4].

In 2014 Ashwini R. Tonde, Akshay P. Dhande proposed "Review Paper on FPGA Based Implementation of Advanced Encryption Standard (AES) Algorithm". They proposed FPGA-based implementation of the Advanced Encryption Standard (AES) algorithm. This implementation is compared with other works to show the efficiency. The design uses an iterative looping approach with block and key size of 128 bits, lookup table implementation of S-box. The AES algorithm can be efficiently implemented by software. Software implementations cost the smallest resources, but they offer a limited physical security and the slowest process. Besides, growing requirements for high speed, high volume secure communications combined with physical security, hardware implementation of cryptography takes place. They design FPGA based hardware implementation of AES algorithm. The design is coded using VHDL language. The design is simulated on ModelSim software and synthesize on Xilinx ISE software. Performance parameters of design will be calculated in terms of throughput and latency. Results will be compared with previous work[5].

In 2015 Suchita Tayde, Seema Siledar proposed "File Encryption, Decryption Using AES Algorithm in Android Phone" .Nowadays smart gadgets including smart phones and tablets are gaining huge popularity. Comparing with conventional computer, smart phone is

easily carried out and provides much computer functionality, such as processing, communication, data storage as well as many computers services such as web browser, video or audio player, video call, GPS, wireless network. However, smart phone have to come long way in terms of security. Encryption is used for security of information in data storage and transmission process. Various encryption algorithms like DES, 3DES, Blowfish, RSA and others are available to secure the data. In DES, key size is too small. In 3DES, key size is increase but the process is slower than other methods. They have used Advanced Encryption Standard algorithm to overcome above problems. AES algorithm is not only for security but also for great speed. It can be implemented on various platforms especially in small devices like mobile phone. Everyday data is shared, transmitted, stored for many purpose like banking, production, research and development. Hence, we need security for information. Encryption can provide security. This application allows user to run this application on android platform to encrypt the file before it is transmitted over the network. It is used for all type of file encryption such as text, docx, pdf and image encryption. AES algorithm is used for encryption and decryption[6].

In 2015 Zaeniah, Bambang Eka Purnama proposed "An Analysis of Encryption and encryption Application by using One Time Pad Algorithm ".Security of data in a computer is needed to protect critical data and information from other parties. One way to protect data is to apply the science of cryptography to perform data encryption. There is wide variety of algorithms used for encryption of data; this study used a one-time pad algorithm for encrypting data. Algorithm One Time Pad uses the same key in the encryption process and a decryption of the data. An encrypted data will be transformed into cipher text so that the only person who has the key can open that data. Therefore, analysis will be done for an application that implements a one-time pad algorithm for encrypting data. The application that implements the one time pad algorithm can help users to store data securely [7].

In 2016 Mazhar Islam and Mohsin proposed " A New Symmetric Key Encryption Algorithm Using Images as Secret Keys. Symmetric key cryptography is a common cryptographic technique using the same key at both the transmitter and receiver side. The main advantage of symmetric key encryption is its less computational cost compared to its counterpart-public key encryption. In this work a new symmetric key encryption scheme is proposed. Rather than using normal binary secret keys, our technique uses images as secret keys. Message letters are converted into their corresponding 8-bit binary codes. These 8-bit codes are scanned for image pixel values which are represented by the same 8-bit codes. When a match is found between the message 8-bit code and pixel values codes that location of the pixel is saved in a separate file. Instead of saving pixel locations as (x, y) coordinates, their locations are saved as one value column wise. After having all matches, pixels locations are transmitted as cipher text. The receiver side will scan the same image for those locations and will pick values at those locations which represent message codes. The main advantage of this scheme is its high security as its key size is very large[8].

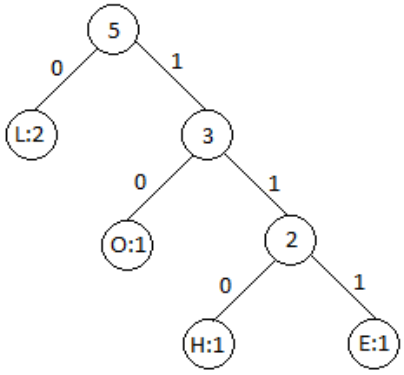
In 2017 M. Ilayaraja, K .Shankar and G. Devika proposed "A Modified Symmetric Key Cryptography Method for Secure Data Transmission" The proposed is a modified Caesar cipher symmetric key encryption method which is developed to convert the actual message into secret information using mathematical principles. Plaintext contains alphabets with case sensitive, numbers and special characters into secret information. The proposed technique produces the ciphertext includes more number of special characters. The result of the proposed method proves that it is very difficult to identify the original message from the encrypted message[9].

III. PROPOSED METHOD

The proposed algorithm used following steps
1. Calculate the frequency of each symbol.
2. Take the two least frequent symbols and assign them to two leaf nodes. And assign the sum of the corresponding frequencies to the parent node.
3. Now select next two least frequency nodes from the rest of the nodes along with the newly created node and form another parent node,
4. Repeat step 3 till a complete binary tree is formed.
5. Starting from the root, assign '0' to the left child and '1' to the right child of every node till you reach the leaf nodes.
6. To assign the code to the symbol, trace the path from root to the corresponding node.

Encryption Algorithm

Step 1. Compress the given text using Huffman coding algorithm as explained above.



Here is an example to do that:

Let's say the data being sent is 'HELLO'.

1. Calculating the frequencies of all the symbols:
H: 1; E: 1; L: 2; O: 1.

2. Constructing the tree

Code:
H: 110
E: 111
L: 0
O: 10

Therefore, the compressed code becomes, say **R = 1101110010.**
Step 2. Take two keys 'A' and 'B'. 'A' is used as public key, not known to the receiver beforehand and 'B' (should be small) is used as the private key previously known by the receiver.
For our example, let's say A=1001 and B= 1000.
Step 3. Take the first 'n' bits of the compressed code where 'n' is the number of bits in 'A'.
Perform XOR operation on these bits with A and shift A by one bit. Repeat this till you reach the end of the code. The new code formed is **R**.
For our example, the operation will be as follows

1	**1**	**0**	**1**	**1**	**1**	**0**	**0**	**1**	**0**

⊕

1	**0**	**0**	**1**

0	**1**	**0**	**0**	**1**	**1**	**0**	**0**	**1**	**0**

⊕

1	**0**	**0**	**1**

0	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**1**	**0**

⊕

1	**0**	**0**	**1**

0	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

⊕

1	**0**	**0**	**1**

0	**0**	**1**	**1**	**0**	**0**	**1**	**0**	**1**	**0**

⊕

1	**0**	**0**	**1**						
0	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**0**

⊕

1	**0**	**0**	**1**

0	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**

⊕

1	**0**	**0**	**1**

0	**0**	**1**	**1**	**1**	**1**	**0**	**1**	**0**	**1**

The encrypted code, **R** now becomes 0011110101
Step 4. Multiply 'R' by 'B' to change R to R*B.
For our example:
0011110101
*1000
0011110101000
Thus the final encrypted data to be sent, R= 0011110101000.
While sending the data 'R', 'A' and the Huffman tree 'T' is sent to the receiver so as to enable him to decrypt the message. Bis known to the receiver previously

Decryption Algorithm

Receiver will now have R, A, the Huffman tree T sent by the sender along with B.
Step 1. Divide R by B.
For our example:
R = 0011110101000 / 1000 = 0011110101.
Step 2. Starting from the last 'n' bits of the obtained code from last step, Apply XOR operation on R and A; where 'n' is thenumber of bits in A. And keep repeating till you reach the first bit of R.
For our example:

0	**0**	**1**	**1**	**1**	**1**	**0**	**1**	**0**	**11**

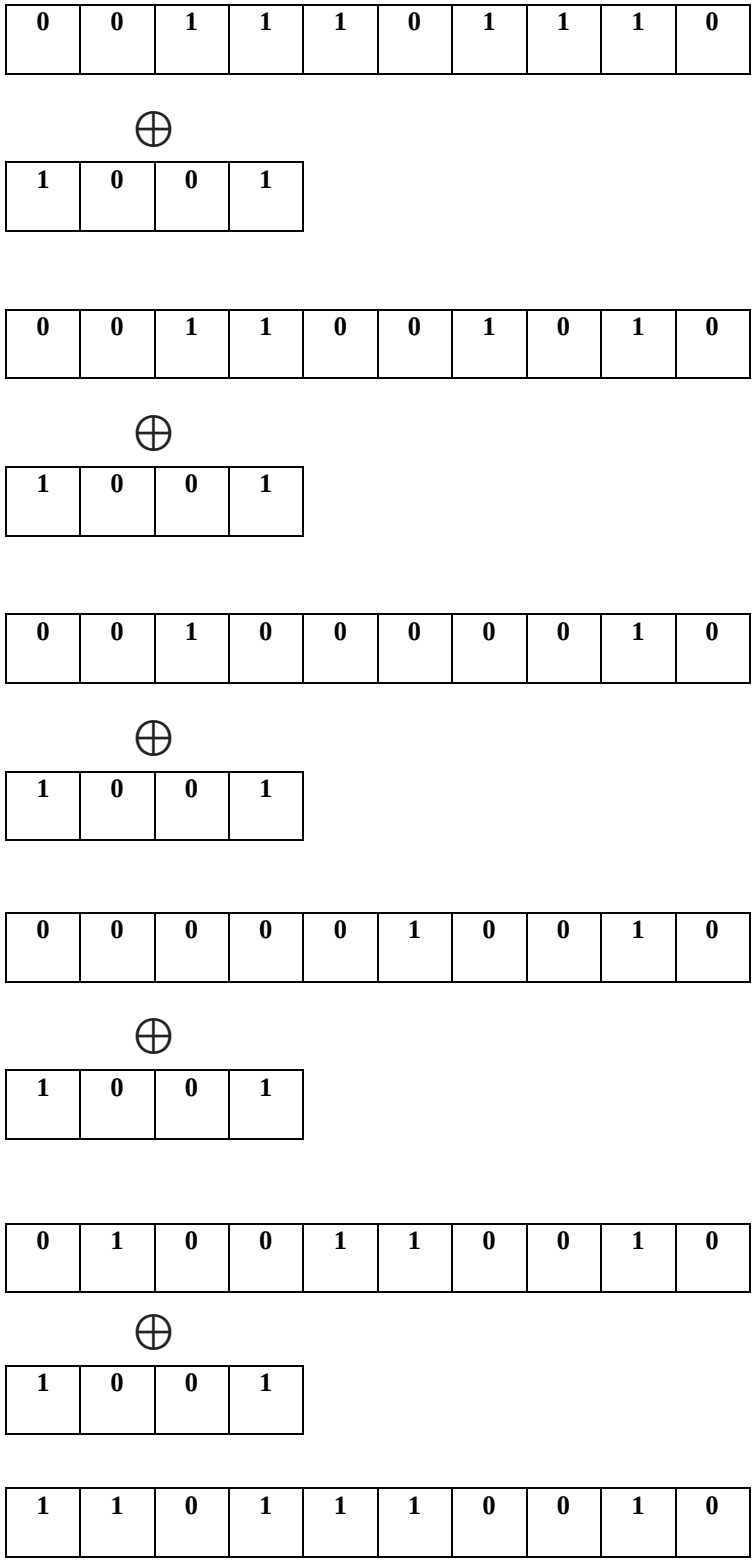
⊕

1	**0**	**0**	**1**

0	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**

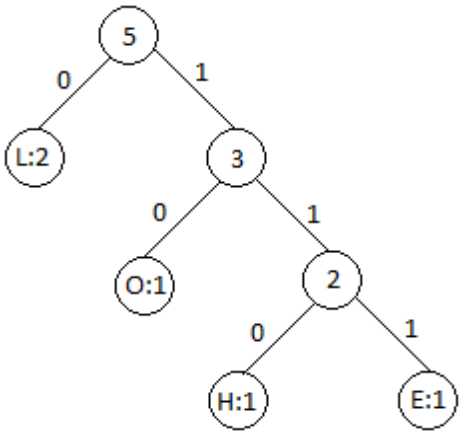
⊕

1	**0**	**0**	**1**

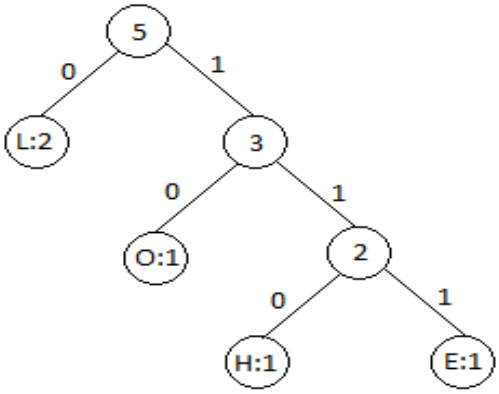


Thus, E" = 1101110010.

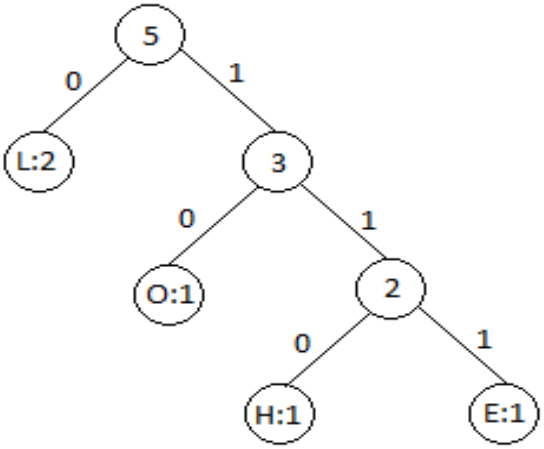
Step 3. Now uncompress the data obtained, for which trace the Huffman tree T, according to the bits of R. As you reach a leaf node, note the symbol associated with it and restart from the root node again. For our example : Tracing from root node



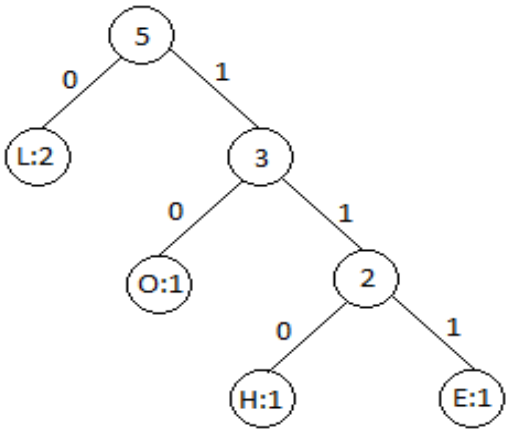
1101110010
H



110**111**0010
H E



1101110**0**10
H E L L



1101110010
H E L L O

Thus, we decrypted the encrypted message.

IV. EXPERIMENTAL ANALYSIS

\

Comparison on the basis of file size and execution time for encryption

Table 1 Shows comparison between Reverse Encryption Algorithm and proposed method on the basis of file size and execution time for encryption. Figure 1 shows the comparisons graph for encryption.

Table 1 comparison on the basis of file size and execution time for encryption

File Size in Bytes	Reverse Encryption Algorithm	Proposed method
50	1438	1282
100	2426	1686
200	3242	2668

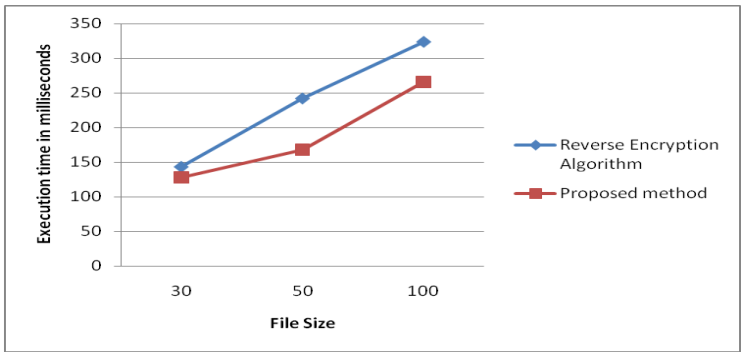


Figure 1 comparisons graph on the basis of file size and execution time for encryption

CONCLUSION

In electronic communication data security is an important issue. There are five different security perimeter which are used in cloud computing. in storage level we can secure the data by applying various encryption based approach. there are several approach have been developed for this purpose. we proposed a new approach which reduces size of the text and provides more strong encryption as compared to reverse encryption approach.

REFERENCE

1. Monika Agrawal & Pradeep Mishra "A Comparative Survey on Symmetric Key Encryption Techniques" International Journal on Computer Science and Engineering (IJCSE)ISSN : 0975-3397 Vol. 4 No. 05 May 2012.
2. Obaida Mohammad Awad Al-Hazaimeh" A New Approach for Complex Encrypting and Decrypting Data" International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.2, March 2013.
3. Mansoor Ebrahim & Shujaat Khan Symmetric Algorithm Survey: A Comparative Analysis International Journal of Computer Applications (0975 – 8887) Volume 61– No.20, January 2013
4. Mitali, Vijay Kumar & Arvind Sharma "A Survey on Various Cryptography Techniques". International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)Web Site: www.ijettcs.org Email: editor@ijettcs.orgVolume 3, Issue 4, July-August 2014

5. Ashwini R. Tonde&, Akshay P. Dhande "Review Paper On FPGA Based Implementation Of Advanced Encryption Standard (AES) Algorithm "International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 1, January 2014.
6. Suchita Tayde & Seema Siledar "File Encryption, Decryption Using AES Algorithm in Android Phone". Volume 5, Issue 5, May 2015 ISSN: 2277 128X International Journal of Advanced Research in Computer Science and Software Engineering Research Paper.
7. Zaeniah & Bambang Eka Purnama "An Analysis of Encryption and Decryption Application by using One Time Pad Algorithm International Journal of Advanced Computer Science and Applications, Vol. 6, No. 9, 2015.
8. Mazhar Islam ; Mohsin Shah "A New Symmetric Key Encryption Algorithm Using Images as Secret Keys " 2015 13th International Conference on Frontiers of Information Technology (FIT) IEEE Xplore: 29 February 2016.
9. Dr. M. Ilayaraja1, Dr. K.Shankar2, Dr. G. Devika3 "A Modified Symmetric Key Cryptography Method for Secure Data Transmission" International Journal of Pure and Applied Mathematics Volume 116 No. 10 2017, 301-308 ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version) url: http://www.ijpam.eu Special Issue
10. H. Brown, Considerations in Implementing A Database Management System Encryption Security Solution , A Research Report presented to The Department of Computer Science at the University of Cape Town, 2003.
11. A. Ceselli, E. Damiani, S. D. C. D. Vimercati, S. Jajodia, S. Paraboschi, and P. Samarati, "Modeling and assessing inference exposure in encrypted databases," ACM Transactions on Information System Security, vol. 8, no. 1, pp. 119–152, 2005.
12. J. Daemen and V. Rijmen, "Rijndael: The advanced encryption standard (AES)," Dr. Dobb's Journal, vol. 26, no. 3, pp. 137-139, Mar. 2001.
13. E. Damiani, S. D. C. D. Vimercati, M. Finetti, S. Paraboschi, P. Samarati, and S. Jajodia, "Implementation of a storage mechanism for untrusted dbmss," IEE Security in Storage Workshop 2003, pp. 38-46, 2003.
14. E. Damiani, S. D. C. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Metadata management in outsourced encrypted databases," in The Second VLDB Workshop on Secure Data Management, Lecture Notes in Computer Science, pp. 16–32, Springer, 2005.
15. G. Davida, D. L. Wells, and J. B. Kam, "A database encryption system with subkeys," ACM Transactions on Database Systems, vol. 6, no. 2, pp. 312–328, 1981.
16. M. R. Doomun and K. M. S. Soyjaudah, "Analytical comparison of cryptographic techniques for resource-constrained wireless security" International Journal of Network Security, vol. 9, no. 1, pp. 82-94, 2009.
17. D. S. A. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Evaluating the performance of symmetric encryption algorithms,"
18. International Journal of Network Security, vol. 10, no. 3, pp. 213-219, 2010.
19.] D. S. A. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Evaluating the effects of symmetric cryptography algorithms on power consumption for different data types" International Journal of Network Security, vol. 11, no. 2, pp. 78-87, Sep. 2010.
20. C. Gu and Y. Zhu, "New efficient searchable encryption schemes from bilinear pairings," International Journal of Network Security, vol. 10, no. 1, pp. 25-31, Jan. 2010.