International Journal of Innovative Research in Information Security (IJIRIS) ISSN: 2349-7017(O)
Issue 8, Volume 2 (August 2015) ISSN: 2349-7009(P)
www.ijiris.com

Improving Cloud Security Using Multi Level Encryption and Authentication

A.Prakash*
III year Student
Department Of Information Technology
SSN College of Engineering

M.Satish
III year Student
Department Of Information Technology
SSN College of Engineering

T.Sri Sai Bhargav
III year Student
Department Of Information Technology
SSN College of Engineering

Munesswari.G
Associate Professor
Department Of Information Technology
SSN College of Engineering

Abstract— As people have become more social and electronically attached, the concern for information sharing over the internet still persist. As known many powerful cryptographical approaches have been proposed in the past which are practically impossible to break, yet there exists a major concern of total encryption and decryption time taken as a whole. It is a known fact that in encrypting a large chunk of data, traditional asymmetric key algorithm may be slower to symmetric key algorithm by 1000 times or more. Hence this paper proposes a hierarchical structure in which the parties are first authenticated, then exchange keys by asymmetric key algorithm, then do actual encryption and decryption by the symmetric key algorithm. This will be useful to improve the security in cloud applications.

Keywords— Modified Rivest Shamir Adleman Algorithm (MRSA); Modified Triple Data Encryption Standards (M3-DES); Diffie-Hellman Key exchange Algorithm (D-H Algorithm).

I. INTRODUCTION

In Cloud Computing the data is transferred over the internet. Even many classified and important transactions occur over the internet and hence it is important to have some security i.e. some encryption standards to protect the data which is transmitted over the channel. With the increasing numbers of people who are involved in the intrusion side, it is necessary to have a strong approach to security so that all attacks and factors to attack are met and hence avoided. Generally the following approach is applied wherein the user at one end encrypts the data into cipher text and then sends over the channel and the receiver receives the cipher text and then reverts it back to the original data by decryption. So here the intruder sitting in the middle gets only the cipher text and without the knowledge of the key he could not revert it back to the original text, hence this approach is applied which is also called as cryptography.

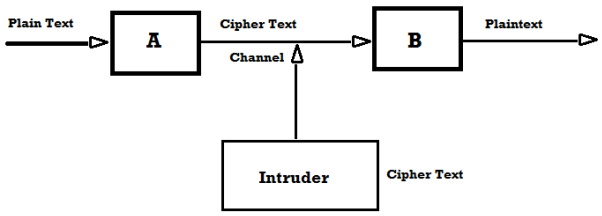

Fig 1. Basic Cryptography with Man-In-The-Middle

Hence the Fig.1 gives a brief idea about the Cryptography.
Tracing the history of cryptographic algorithms we ended up finding many algorithms. Every Algorithm we found had their own pros and cons. These algorithms are broadly classified as

1. Symmetric/Private Key Algorithms
2. Asymmetric/Public Key Algorithms

The Symmetric key Algorithm uses only one key for encryption and decryption of data. Whereas the Asymmetric key Algorithms uses two keys private and public keys .Public key is used for encryption and private key is used for decryption (e.g. RSA). The list of Symmetric Key Algorithms is as follows:

1. Data Encryption Standards(DES)
2. Triple Data Encryption Standards(3-DES)
3. Advanced Encryption Standards(AES)

The list of Asymmetric Key Algorithms is as follows:

1. Rivest Shamir Adleman Algorithm (RSA)
2. Diffie-Hellman Algorithm

There are many such algorithm available, the above are the most popular and most used one's. The main disadvantage of the above algorithms is,

1. DES- Already cracked and the cryptanalysis tool is made available.
2. 3-DES- Very difficult to break, but due to the problem of weak keys at time very rarely can be easily broken.
3. RSA- This is a good algorithm but runs slower to the other private key algorithms due to its exponential raising.

II. LITERATURE SURVEY

In [1], Secured Multiparty computation is been discussed which is a technique that analyses the private data and results are computed by the parties having similar background on minimising the threat of disclosure. In [3],a secured sum is being developed which allows the parties to work on their inputs devoid of sharing with others which is highly secured. On top of secured sum an algorithm was developed to share the simple integer information at all iterations on anonymous id assignment. In [5], we use a secured computation function in which the parties need not disclose their integer information to each other. This function being developed is popular in data mining applications. In [6] id's are generated randomly with no central authority in a network fashion. In [7] we use Anonymous ID Assignment (AIDA) technique which is a sharing algorithm and requires unbounded number of iterations. Dr. James H. Yu [8] in this concluded that many networks are prone to attack and generally people ignore it. As network security increases system performance decreases. J. Daemen [10] concluded that AES is faster and more efficient than other encryption algorithms. Kyung Jun Choi et.al [11] investigated the algorithms suitable for wireless networks and found that MD5 and RC4 stand above all encryption algorithm. Here we have been going through many encryption algorithms including AES, DES, 3DES, RC2, Blowfish, and RC6 and their commonality and performance related aspects is also been spoken of. The results show that AES is faster and more efficient than other encryption algorithms. When the transmission of data as a criteria is taken there is insignificant difference in performance related aspects of different symmetric key schemes. A study in [16] is been performed on various secret key algorithms like DES, 3DES, AES, and Blowfish. Input files of various size are taken and each of the above algorithm is made to be implemented on the input files and results is been compared indicating that Blowfish has high performance when compared to other encrypting algorithms. Also AES stands above 3DES and DES. In addition, the advantage of DES over 3DES is that it has triple throughput. Ruangchaijatupon et al. [4] shows the energy consumption of different symmetric key encryption algorithms on hand-held devices. It is found that only after 600 encryption done on a 5 MB file using 3DES the battery power is been identified as 45% and other encryption algorithm is not so efficient in power consumption as battery dies of rapidly before even completion of 600 trials. A Crypto++ Library [17] is a free C++ class library containing cryptographic schemes. It evaluates the performance between the selected algorithms. Thus it concludes that Blowfish and AES have the better performance than other encryption algorithms .In [16, 18], performance between selected symmetric encryption algorithm is been presented. In case of packet size Blowfish stands above all algorithms followed by RC6. Also 3DES has low performance issue when compared to DES. In terms of time consumption RC2 is weak by consuming more time than the rest of algorithms. Secondly it's been concluded that AES stands above 3DES, DES and RC2. These evaluations are made only on Windows OS. Salama et al. [19] On basis of different settings such as size of blocks, data types used, CPU time and different key size a comparison study is made on encryption algorithms (AES, DES, and 3DES, RC2, Blowfish, and RC6). The results show that Blowfish and AES is more efficient when compared to others. Elkilani et al. [18]Performance of real time video streaming was taken as a criteria and algorithms like RC4, AES, XOR is been tested and AES has less time overhead than the two.

III. ARCHITECTURE FOR IMPLEMENTATION IN CLOUD

The following approach has to applied in the cloud computing scenario when a user say the client gives request to some service which is present in a virtual machine of a cloud so this becomes the server (also the vice versa) see in Fig2. The Architecture used here has three phases as follows:

- Phase 1:
  - Here the User says A wants to connect to Web Service running in Virtual Machine 1 of the Cloud. So First the user is authenticated for connection establishment the so called Long-Term and Short-Term Keys are given to the Virtual Machine 1 by the Hypervisor of the respected Virtual Machine from the Key Schedule. So at the end of Phase1, A is connected to the Web Server.
- Phase 2:
  - Now having the connection established, when A wants to gain access of any data say file transfer or uploading of data or any signs data transfer then Virtual Machine 1 performs the key transfer with A, connects to its Database which may present in some other VM, Encrypts it using the Algorithm proposed below and then finally transfer it to A. A at receiving end decrypts the cypher and reads the data.
- Phase 3:
  - So this phase specifically deals with the scenario when the Virtual Machine1 wants to access the data present in the Virtual Machine say VMx which is in turn present in the other Datacentre of same or different cloud, So here what happens is the VM1 goes through all the phases discussed above and then gets connected to VMx .

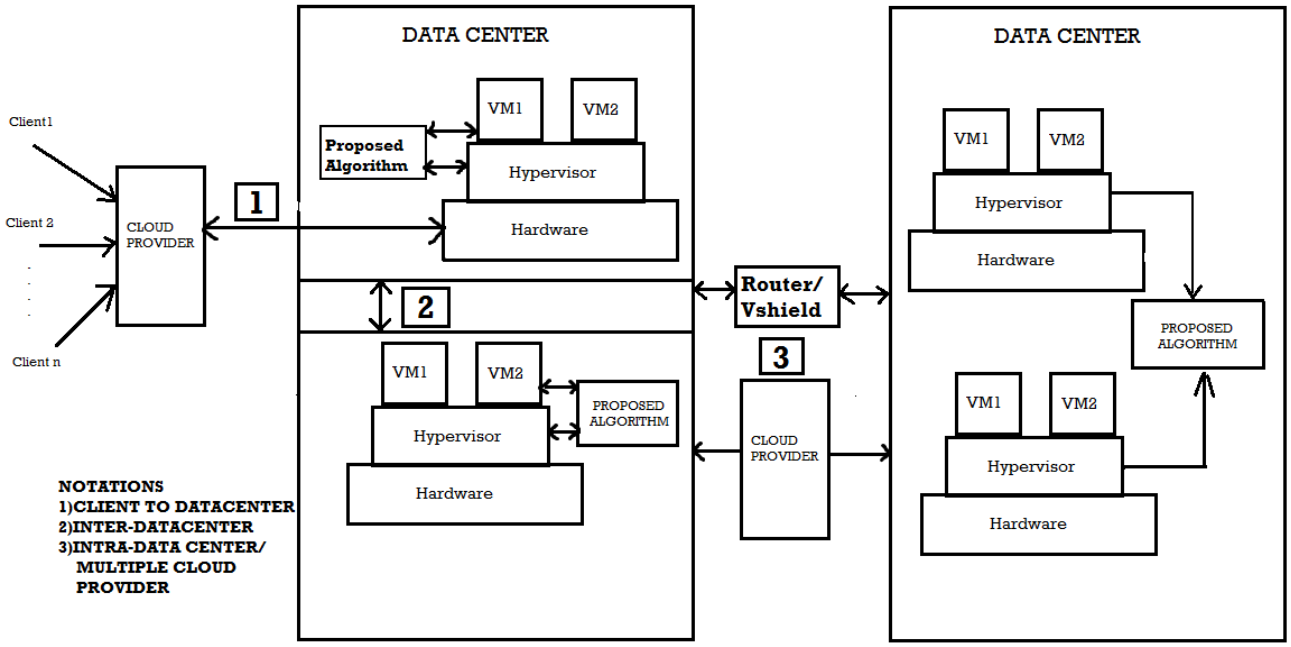

Fig2. Architecture for cloud implementing the proposed system

The proposed approach can be also applied in the scenario where the client and server are independent machines.

IV. PROPOSED LAYERED HIERARCHICAL STRUCTURE

In our proposed hierarchical structure there are three algorithms as a whole are used: Diffe-Hellman Key Exchange algorithm, MRSA algorithm and M3-DES algorithm. The overall architecture diagram of the entire methodology of security is shown in Fig.3

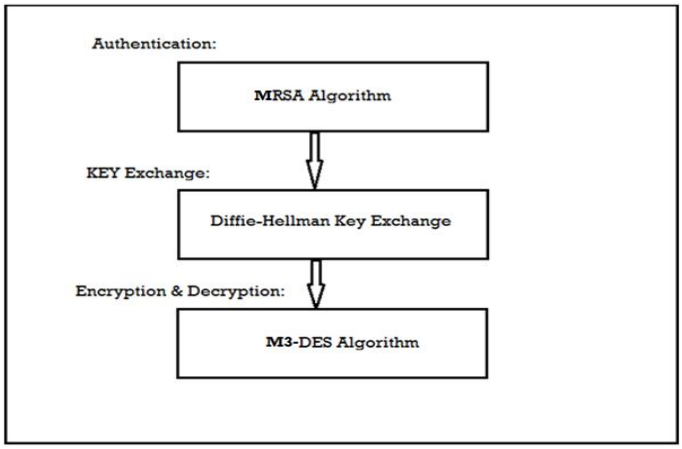

Fig 3. Multilevel Encryption and Authentication

Procedures of algorithms used:

Diffie-Hellman Key Exchange Algorithm:

1. Consider two parties A and B who agree upon the key transfer and a intruder say C whose intension is to know the message which is being transferred.
2. A or B send two values say (g and n) to either of them.
3. The intruder C has values g and n.
4. Now A and B produce a random number each say x and y respectively.
5. Now A transmits X= ((g^x) % n) to B and B transmits Y=((g^y) % n) to A.
6. The intruder C has now got hold of value g, n, X, Y given these values the intruder cannot find the value of x and y because this problem is called Discrete Logarithm Problem (DLP) till today it is considered to be a computationally in-crackable problem. I.e. it cannot be broken in a given reasonable amount of time .Given X, g, n it is not computationally feasible to calculate the value of x.

7. Now A finds KeyA=((Y^x)%n) and B finds KeyB=((X^y)%n).
8. Now both parties agree upon the same key since KeyA==KeyB.

Having this said the D-H algorithm is susceptible to the Man in the middle attack. Wherein the intruder C somehow gets hold of the router and disconnects A from the network and he himself acts as A. So we need some Authentication by which we can verify that we talking to the right person, which is given by MRSA.

MRSA Algorithm:

As far as our approach is concerned, It is crucial that authentication process plays a vital role in the actual algorithm of RSA. Here the public keys of A and B are d1, d2 respectively and the private keys of A and B are e1, e2

1. Parties A and B share with either of them their public key say d1 and d2 respectively.
2. Now A wants to authenticate that the opposite party is B, so what A does is he asks B to encrypt his own private key say e2 as the message using the key d1 and send it to A.
3. Now A having the encrypted message , decrypts it using his private key e1.Which intern gives him e2
4. A checks that
   If ($d2 \cdot e2 \equiv 1 \pmod{\varphi(n)}$)
   The opposite user is B.
   Else
   There is some intruder or say man in the middle.

M3-DES Algorithm:

The Algorithm used for Triple DES is the same, but the problem or a possible micro hole in 3-DES which is Weak, Semi-Weak and Demi-Weak keys is dealt with here in this algorithm.

Weak Keys:

These are keys that cause the encryption mode of DES to act identically to the decryption mode of DES. The Modified Triple Data Encryption Standard Algorithm is as follows:

1. The Key Scheduler generates a 56-bit for a single block of DES as shown below.
2. The First key say K_0 is then passed through the block called Weak Key Detection(WDK) Block
   - If the key is not Weak then it is passed for encryption
   - Else it is resent to the key scheduler indicating the need for generating the new key.

Weak Key Detection (WDK) Block:

In the block a search for weak key is made shown in fig.4, if the search is found then the key is resent to the key scheduler indicating the need for generating the new key. The Weak, Semi-Weak and Demi-Weak keys in DES are as follows:

Weak Keys:

0101010101010101, fefefefefefefefe, 1f1f1f1f1f1f1f1f, e0e0e0e0e0e0e0e0

Semi-Weak:

01fe01fe01fe01fe	fe01fe01fe01fe01
1fe01fe01fe01fe0	e01fe01fe01fe01f
01e001e001e001e0	e001e001e001e001
1ffe1ffe1ffe1ffe	fe1ffe1ffe1ffe1f
011f011f011f011f	1f011f011f011f01
e0fee0fee0fee0fe	fee0fee0fee0fee0

Demi-Weak:

1f1f01010e0e0101	e00101e0f10101f1
011f1f01010e0e01	fe1f01e0fe0e01f1
1f01011f0e01010e	fe011fe0fe010ef1
01011f1f01010e0e	e01f1fe0f10e0ef1
fe0101fefe0101fe	
e0e00101f1f10101	e01f01fef10e01fe
fefe0101fefe0101	e0011ffef1010efe
fee01f01fef10e01	fe1f1ffefe0e0efe

e0fe1f01f1fe0e01
fee0011ffef1010e 1ffe01e00efe01f1
e0fe011ff1fe010e 01fe1fe001fe0ef1
e0e01f1ff1f10e0e 1fe001fe0ef101fe
fefe1f1ffefe0e0e 01e01ffe01f10efe

fe1fe001fe0ef101 0101e0e00101f1f1
e01ffe01f10efe01 1f1fe0e00e0ef1f1
fe01e01ffe01f10e 1f01fee00e0ef1f1
e001fe1ff101fe0e 011ffee0010efef1
1f01e0fe0e01f1fe
01e0e00101e1e101 011fe0fe010ef1fe
1ffee0010efef001 0101fefe0101fefe
1ffee0010ef1fe01 1f1ffefe0e0efefe
01fefe0101fefe01
1fe0e0f10ef1f10e fefee0e0fefef1f1
01fee01f01fef10e e0fefee0f1fefef1
01e0fe1f01f1fe0e fee0e0fefef1f1fe
1ffefe1f0efefe0e e0e0fefef1f1fefe

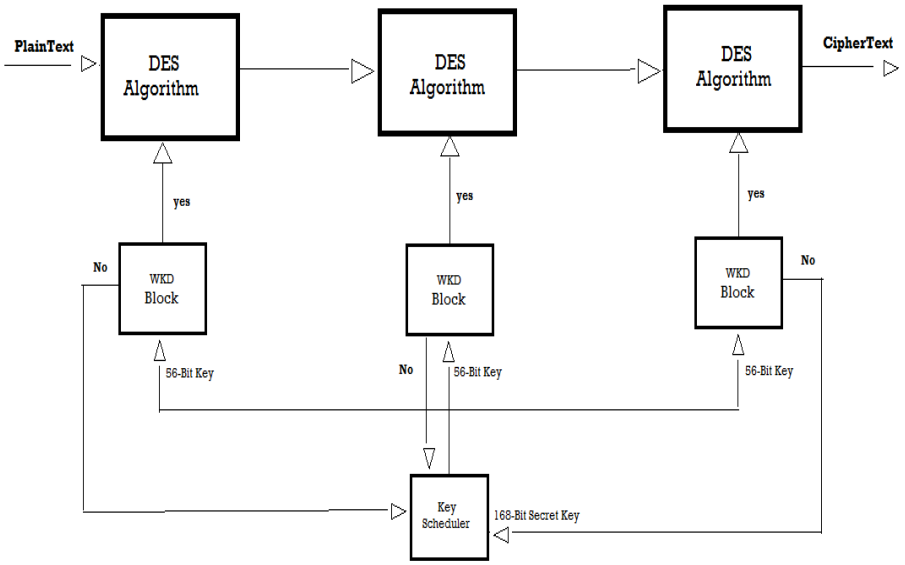

Fig 4. DES with Weak Key detection

Proposed Algorithm:

The main purpose of this approach is to reduce the overall time taken for the entire security process as a whole.
The Algorithm is as follows:

1. The Authentication of users is done by the MRSA approach given above.
2. Each of the users is allocated with two pairs of keys
    1) Long-term keys:
       These keys will be unique and can be there for connection establishment.
    2) Short-term keys:
       These Keys will be allocated to each user after each conversation between two users. This is done because while authentication the users will get to know the private keys of each other, and hence it is not secure to have the same key even after the disconnection.
3. After authentication the users agree upon the key value which will be used for encryption using Diffie-Hellman key exchange algorithm as discussed above.

Finally the M3-DES is run for encryption and decryption based on the key decided in step 3.

V. EVALUATION RESULTS

To justify our fact as given above that time taken for encryption is 100 to 1000 times more for large data set if we use asymmetric key algorithms to symmetric key algorithms, we use a simulation tool to run both the symmetric and asymmetric key algorithms which are Triple DES and RSA respectively.

Here in our approach we have used MRSA for authentication while for encryption and decryption we have used Modified Triple Data Encryption Standards. The following is a table which shows the time taken for Encryption for M3-DES and RSA for packets or data of different size.

Input Size(KB)	M3-DES	RSA
45	51	55
55	45	49
96	77	90
236	114	120
319	155	165
560	170	185
899	300	350
5400	1167	1445
Throughput (MB/sec)	2.01	1.54

Table1. Time Taken for encryption in milliseconds

The following is a table which shows the time taken for Decryption for M3-DES and RSA for packets or data of different size in milliseconds.

Input Size(KB)	M3-DES	RSA
45	51	62
55	45	58
96	65	56
236	70	65
319	85	156
560	160	170
899	175	200
5400	830	905
Throughput (MB/sec)	4.12	2.21

Table2. Time Taken for decryption in milliseconds

The encryption and decryption throughputs are shown in fig.5 and fig.6.

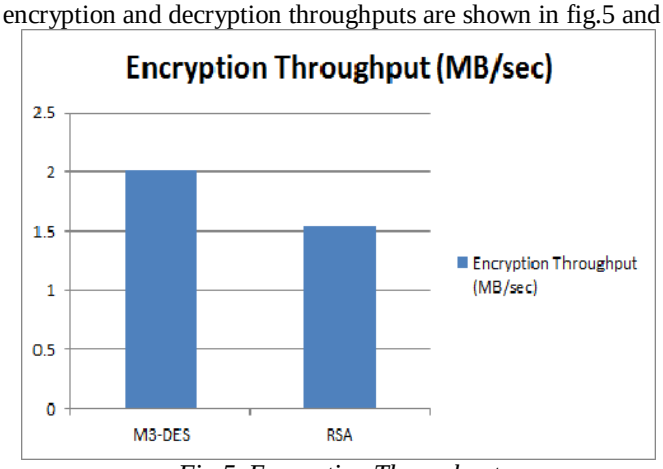

Fig 5. Encryption Throughput

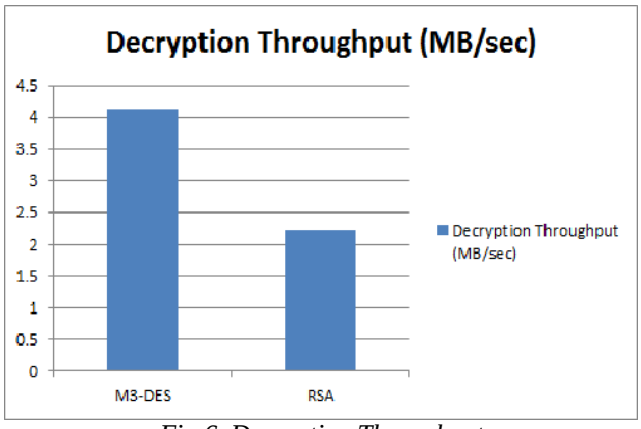

Fig 6. Decryption Throughput

Hence the results shows that asymmetric key algorithms are very slow when compared to the symmetric key algorithms. We have used very small data sets but if we use very large data set say of 100 MB file then for encrypting any asymmetric key algorithm will increase time taken in an exponential order and but when we use symmetric key algorithms whatever may be the file almost same time is used because the rounds in the M3-DES is just going to permute the same text and hence the time will not increase in an exponential order. So based on these results we have used M3-DES rather than using RSA.

VI. CONCLUSIONS

This layered approach gives a total stability to the user, wherein the hosts are first authenticated, exchange keys and finally encrypt and transfer the cipher text. The receiver receives the cipher, decrypts using the agreed key and reverts or gets the message without intrusion. We have also shown the results for encryption and decryption throughputs. So, we argue that this multilevel algorithm improves security in public clouds.

REFERENCES

[1] Dr. Durgesh Kumar, Neha Koria, Nikhil Kapoor, Ravish Bahety (2009), "A Secure Multi-Party Computation Protocol for Malicious Computation Prevention for Preserving Privacy During Data Mining", International Journal of Computer Science and Information Security,Vol. 3.

[2] A.Nadeem , MYJ" A performance comparison of data encryption algorithms." First International Conference on Information and Communication Technologies.2005, pp: 84- 89.

[3] Akheel Mohammed, Sajjad Ahmed Md , Ayesha (2013), "Confidentiality And Anonymity Strengthening in Computational Services", IJRRECS,Volume-1,Issue-6,1006-1011.

[4] N. Ruangchaijatupon and P. Krishnamurthy "Encryption and power consumption in wireless LANs-N" The Third IEEE Workshop onWireless LANs, pp. 148-152, Newton, Massachusetts, Sep. 27-28,2001.

[5] Swathi, P.Jyothi, and Anil Kumar(2014), "Assigning Privacy Ids For Each Data That Have Been Sharing In Wireless Networks", International Journal of Communication Network and Security (IJCNS) ISSN: Volume-2, Issue-3.

[6] Ayswarya R Kurup, Simi Lukose (2014), "Security Enhanced Privacy Preserving Data sharing With Random ID Generation",IJSRE Volume 2 Issue 8.

[7] Larry A. Dunning, And Ray Kresman (2013), "Privacy Preserving Data Sharing With Anonymous ID Assignment ",IEEE Transactions on Information Forensics and Security, Vol. 8, NO. 2.

[8] *Dr. James H. Yu & Mr. Tom K. Le, "Internet and Network Security", "Journal of industrial technology",Volume 17, Number 1 - November 2000 to January 2001*.

[9] T.Muthumanickam,"PERFORMANCE ANALYSIS OF CRYPTOGRAPHIC VLSI DATA", IRACST – International Journal of Computer Networks and Wireless Communications (IJCNWC), ISSN: 2250- 3501 Vol. 2, No. 1, 2012.

[10] J. Daemen, and V. Rijmen, \Rijndael: The advanced encryption standard", Dr. Dobb's Journal, pp. 137- 139, Mar. 2001.

[11] Kyung Jun Choi, John –In Song, "Investigation of feasible cryptographic Algorithm For wireles sensor network", International conference on ICACT Feb 20- 22,2006.

[12] B. Schneier, J. Kelsey, " Unbalancedfeistelnetworks and block cipher design. In: Proceedings of the Third International ", Workshop on Fast 12 Software Encryption. London, UK: Springer-Verlag. ISBN 3-540-60865-6; 1996:121-144.

[13] N.Ferguson, D.Whiting, B.Schneier, J. Kelsey, S.Lucks, T .Kohno. "Helix fast encryption and authentication in a single cryptographic primitive". In: Proc. Fast Software Encryption 2003, volume 2887 of LNCS. Springer-Verlag; 2003:330-346.
[14] A. Nadeem and M. Y. Javed, "A performance comparison of data encryption algorithms," Information and Communication Technologies, ICICT 2005, pp.84-89, 2005.
[15] W.S.Elkilani, H.m.Abdul-Kader, "Performance of Encryption Techniques for Real Time Video Streaming", IBIMA Conference, Jan 2009, PP 1846-1850.
[16] D. Salama, A. Elminaam and etal, "Evaluating The Performance of Symmetric Encryption Algorithms", International Journal of Network Security, Vol.10, No.3, PP.216–222, May 2010

