Study & analysis of cryptography algorithms : RSA, AES, DES, T-DES, blowfish

Article *in* International Journal of Engineering & Technology · December 2017

DOI: 10.14419/ijet.v7i1.5.9150

2 authors, including:

Sachin Kumar
Malaviya National Institute of Technology Jaipur
11 PUBLICATIONS **48** CITATIONS

SEE PROFILE

International Journal of Engineering & Technology, 7 (1.5) (2018) 221-225

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Study & analysis of cryptography algorithms : RSA, AES, DES, T-DES, blowfish

Pankaj Singh [1*], Sachin Kumar [2]

[1]M.Tech Scholar, Ajay Kumar Garg Engineering College
Ghaziabad, U.P., India
[2]Professor, Ajay Kumar Garg Engineering College
Ghaziabad, U.P., India
**Corresponding author E-mail: pspankaj02@gmail.com*

Abstract

Cryptography is about protecting the data from third parties or from public to read confidential data. Cryptography mainly focuses on encrypting the data or we can say converting the data and decrypting the actual data or we can say reconverting the data by different methods. These encryption and decryption methods are based on mathematical theories and are implemented by computer science practices. But as cryptography progressed ways were found to decode the secured data and view actual data. This was also done by the use of mathematical theories and computer science practices. Popular algorithms which are used in today's world are, AES (Advance Encryption Standard), Blowfish, DES (Data Encryption Standard), T-DES (Triple Data Encryption Standard), etc. Some of the previously known algorithms were RSA (Rivest–Shamir–Adleman), ECC (Elliptic curve cryptography), etc. These algorithms have their own advantages and drawbacks. But as people were progressing more in breaking them down, these algorithms were supported by digital signatures or hash done by different algorithms like MD5, SHA, etc. By these means data integrity, data confidentiality, and authentication of data are maintained. But as the things are progressing it seems that new advancements are always needed in the field of cryptography to keep the data secure.

Keywords: *Encryption, Decryption, Cryptography, AES, DES, RSA, Blowfish, Triple-DES*

1. Introduction

Cryptography consists of two important methods which help in protecting data and retrieving the protected data. These methods are known as encryption and decryption. In the method of encryption we protect the actual data, by converting it to some other data or protecting it with some key. The data which is encrypted is referred to as plain text, and data which has been protected or comes after applying the steps of encryption is known as cipher text. These cipher texts are always shared between different parties or is stored in data bases to keep the actual information or data protected. In same manner decryption is the methods used to retrieve the actual information from the protected or changed data that is converting cipher text back into plain text.

The method for encrypting the data and decrypting the data is executed by different algorithms which are mainly categorised in two different types; those are Symmetric key algo's and Asymmetric key algo's. In case of Symmetric key cryptography, same keys are used to perform encryption and to perform decryption of data. In this process the problem which persists is that when the data is going to be shared between different personand to be kept protected too, then for the purpose of retrieving the actual data it became necessary to share the key which was then used to provide security to that important data. This gave rise to an important matter that the key is also to be shared securely so that it doesn't goes to somebody else other then the parties which are communicating. While, on the other hand if we talk about asymmetric key cryptography or well known as public key algo's, in this two exponentially matching keys are used for encrypting the data and for decrypting the cipher text and are executed or performed by the means of mathematical properties. These keys are generated in pairs such that they exponentially complement each other from which one is used for encrypting the data while other performs decryption. These keys are denoted as private key and public key. In this way the need of sharing single key was removed. In this procedure the parties who want to share information generate their own public and private keys. [1]

In cryptography the strength of algorithm depends on the key and the number of computations done on the data. Generally the keys used were prime numbers. But as more persons started to break small keys, the keys were used as large prime numbers. The process of breaking and finding these keys is called cryptanalysis.

When cryptography came into existence it was mainly used for securing military related information or military data. But as the use of internet is increasing there are many sites which store the personal information of its users, so today cryptography is used in every field to protect the personal information of their users. [2]

In past keys were used mainly in string form or number form, but now images and audio is also used as a way of protecting data.

2. Cryptography algorithms

Cryptography is well known for protecting sensitive data from unauthorized users by the means of Encryption and allows authorized users to decrypt the data. Cryptography consists of two different types of algorithms namely symmetric or private key algo. or cryptosystem and asymmetric or public key algo. or cryptosystem [3].

2.1 Data Encryption Standard (DES)

Data Encryption Standard algorithm is a symmetric algo. that says that same key should be utilized for encrypting the data and similarly decrypting it. It was used widely before it was not broken and became outdated. It had a small block size and a small key size that's why it is used very less now as it is not secure due to small key size[4].

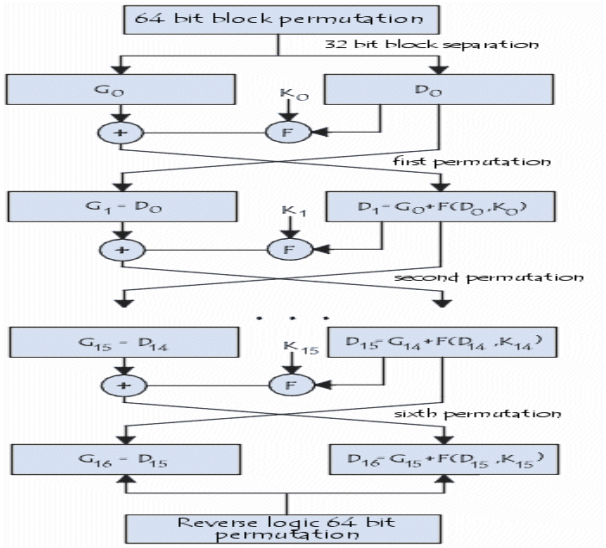


Fig. 1. Block Diagram showing working of DES

DES algorithm consists of the following steps

- Encrypting The Plain Text

o DES acceptsa plaintext and a key which are of size 64 biteach; where in key from 64 bits, 8 bits are referred as parity bits to check whether the key used is similar to the actual key decided to be used. By reducing these 8 bits key remains for 56 bits only which is then used in the algo. further to be executed

o Plain text is gone through permutation.

o Then key permutation is performed which removes the parity bits.

o Then following steps are performed: -

i. The key is divided in two parts of equal size.

ii. Each part of the key is rotated according to the round which is getting performed.

iii. Then the two halves are recombined and then a reducing permutation is applied to reduce the key to 48 bits from 56 bits. This 48 bit key is used to perform encryption on the plaintext of that round.

iv. Then the values we got from rotating each part of the key is used as key for next round.

v. Then the data block is split in two parts of equal size.

vi. Then the left part of the data block undergoes permutation for increase in size which makes it off 48 bits.

vii. Then this 48 bit permutated left part of data block goes into a function XOR operation is performed between this data block and the key.

viii. Then the output from the function is given as input to S-Box which converts it to its actual size that is said to be 32 bit.

ix. Then this substituted output again undergoes permutation performed by P-Box.

x. This permuted output is then given to a function in which XOR operation is executed with inputs as permuted output of previous step and the other equal size part of plaintext. After the complete execution of the function the two parts of the plaintext are swapped to make them input of the next round.

2.2 Triple DES

Triple DES executes DES three times. It was made to increase the key size used in actual DES. But it only happens if different keys are used for different DES executed in Triple DES. If same key is used for all three execution of DES then the key size remains same as actual DES. Triple DES is only beneficial to provide security if all keys used are different. But use of Triple DES is suggested if only if you can compromise with speed of the algorithm. The utilization of key is distributed in two parts in which, one case is that you can use two key k_1 and k_2 in which k_1 is used in first round of Triple – DES and third round of Triple DES. The other case is to use three different keys k_1, k_2 and k_3 for all three rounds [5].Due to use of this method now there was no need to make a new algorithm. Due to this method the same key size was used effectively. Triple DES uses key of 168 bits, 112 bits or 56 bits size. In Triple DES the procedure of executing consecutive rounds of DES is to execute the next round just opposite of the previous one. That is if we are using three different keys say 'k', 'kk' and 'kkk' then the procedure of encryption would be to encrypt the data with 1st key, then decrypt the output of first round with the 2nd key and then again encrypt the output of 2nd round with the 3rd key, so the final output is the cipher text. But Triple-DES is vulnerable to meet in the middle attack. So the key size variants depending upon the number of different keys taken. If a single key is used to perform all three rounds in Triple DES then the key size remains to 56 bits without parity bits that is if only one key leaked the complete procedure gets broken. While if two different keys are used each of 56 bits without parity bits then the total size of key becomes 112 bits[6]. It is also vulnerable as if a single key is determined the other could be determined easily. The best way to use Triple DES is to use three different keys which make the total size of the key as 168 bits without parity. As all the three keys are different, by knowing one key will also not work to break the logic.

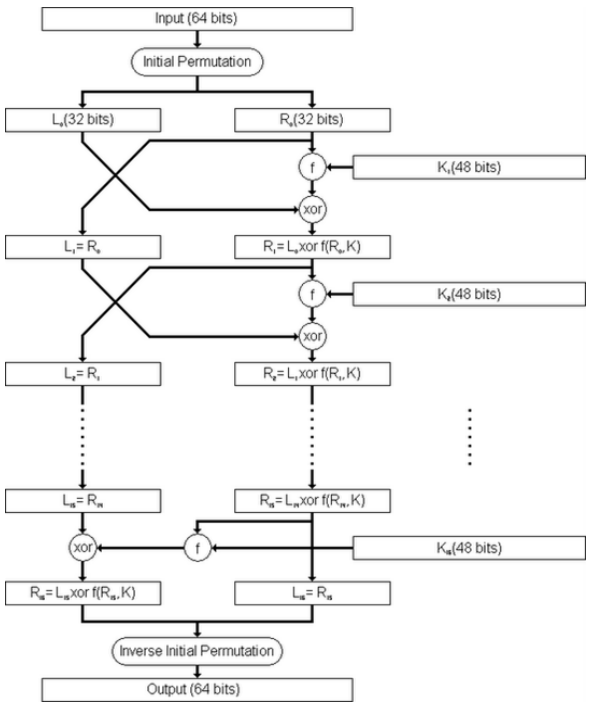


Fig. 2. Block Diagram of Triple DES

So we can formulate that if three different keys are used in Triple DES namely k, kk, kkk, then the encryption and decryption will be carried out in following manner.
Let the plain text be A and cipher text be B.
Encryption is denoted by E and Decryption is denoted by D.
$B = E_{kkk} (D_{kk} (E_k (A)))$ (Encrypting the Plain Text)
Decryption is the reverse:
$A = D_k (E_{kk} (D_{kkk} (B)))$ (Decrypting the Cipher Text)
So the procedure says that while performing encryption first we encrypt the plaintext A with key k, then we decrypt the output of that with the key kk, and finally in third round we encrypt the output of previous by kkk. And similarly we run it in reverse order for decryption, which can be stated as decrypt the cipher text B with key kkk, then encrypt the output of that with key kk, and then in third round we decrypt the output of previous one by k [7].
By this means the complexity of execution of the algorithm increases and makes it much secure then the procedure of using other two ways, which say either use 2 keys in Triple DES or use only one key for Triple DES.

2.3 Advance Encryption Standard (AES)

Advanced Encryption Standard came into existence in 2001 [8]. AES is built to give high security to data. The implementation of AES works fast on both software as well as hardware. Now AES is used in NSIT in place of DES.The number of rounds to be performed for securing the actual data and for retrieving the actual data depends on the size of the key taken for performing these functions. Generally the number of rounds for 128 bit key is 10, 192 bit key is 12 and for 256 bit is 14.

- Algorithm Steps used to secure a 128-bit block size of data

1. The size of cipher key defines how many rounds will be executed.
2. State array is initialized and the initial round key is added to state array.
3. Initial rounds are executed as it is. That is other then final round all round are same.
4. Then Final Round is executed.
5. The output of Final round is the cipher text generated.

Steps followed in different rounds

- Initial Rounds: Following operations are executed in same manner. First function to be executed is Sub Bytes, it is followed by Shift Rows, whose output becomes input to the function Mix Columns, after which final function is executed which is Add Round Key.
- Final Round: Following operations are executed in same sequence. Here also the first step or function to be executed is Sub Bytes, then followed by the function Shift Rows, but then Mix Columns function is not executed , but in place of that directly Add Round Key is performed.
- Encrypting the data: In AES every round executes the following operation other than the final round.

1. Sub Bytes: Sub Bytes is used to substitute a byte with the byte given in fixed matrix as AES designers decided.
2. Shift Rows: In this the rows are shifted in a given order.
3. Mix Columns: Mix Column transformation works on columns, in this it changes each column values by some calculation.
4. Add Round Key: Add Round Key adds that particular rounds key with each value of the matrix. The values are added by the means of matrix addition.

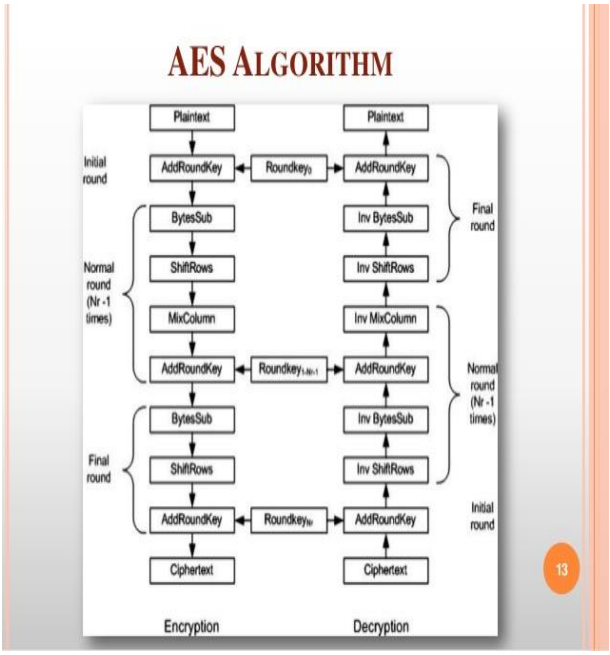
In the final round the only difference from other rounds is that the function mix columns is skipped and directly Add Round Key is executed. The output we get from the final round is the cipher text [9].

- Decrypting the data: Decryption is exact opposite of the Encryption. It involves executing the steps of encryption in reverse order. That is the functions which are performed are as followed

1. First function to be executed is Inverse Shift Rows,
2. This is followed by the function Inverse Sub Bytes,
3. Then we execute the function Add Round Key,
4. The output of the previous undergoes as input to Inverse Mix Columns.

- In process of Decryption also the final round is separately executed similarly to the process of Encryption.
- In the final round the steps followed are same as initial steps but the difference is that Inverse Mix column is not done.
- The output of the final round is the plain text. Similarly as in encryption, in decryption also separate keys are generated for each round.

The keys generated for each round in Advance Encryption Standard are generated as different variation of the actual key. This states that different variations of the actual key in AES become the round keys. These round keys make AES secure as in every round a different variation of actual key noted as round key is added to the data of that round, that was actual data for the first round.



2.4 1RSA

RSA stands for Rivest-Shamir-Adleman. The discoverers of RSA. RSA is the most widely used and known Asymmetric algorithm or we can say public key algorithm. In our proposed work we have used RSA with ECC so that it becomes more secure and faster. RSA is very useful when protected or private data is to be shared between two different parties without the need of any intermediate as it uses one pair of keys, one for encryption and other for decryption[10].
RSA algorithm follows these steps:

1. Generation of pair of keys
2. Encrypting plaintext with public key
3. Decrypting cipher text with private key

I.Generation of pair of keys

In RSA we generate pair of keys named as public key and private key, from which public key is used for encryption and private key is used for decryption. The steps for generation of keys are:

a. Generating public key
1. Choose two distinct prime numbers a and b.
2. Then we compute $x = a \times b$
3. Then we compute $\alpha = (a - 1)(b - 1)$
4. Then we need to select an integer c which should follow the following properties:

i. $1 < c < \alpha$
ii. c and α should be co-prime.

iii.

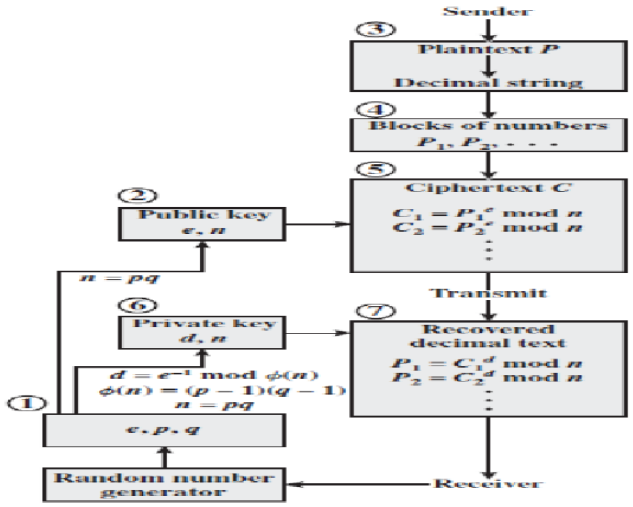


Fig. 4. Working of RSA

The value of c is released as public key exponent, that is public key comprises of c and α.

b. Generating private key

1. Here we compute the value of d by executing c^{-1}(mod α), that corresponds to that cd=1 (mod α).

The value of d is released as private key exponent, that is private key comprises of d and α.

II.Encrypting with public key

In RSA encryption is done by using the public key exponent that is c. The steps are followed as:

1. Let the plaintext to be P, so it is converted in p which belongs to { 0,1,……..,(a×b)-1}
2. Then we compute M = p^c mod n where n=a×b.

III.Decrypting with private key

In RSA decryption is done by using the private key exponent that is d. The steps to be followed are [11]:

Compute M^d = p (mod n)

2.5 Blowfish

Blowfish algorithm is designed by Bruce Schneier. It is widely regarded as the best competitor to AES, and is the most commonly used algorithm after AES. Blowfish uses a variable length key from 32-448 bit. It uses a block size of 64 bit. The total number of rounds remains same in Blowfish irrespective of the key that is it executes in 16 rounds. In blowfish before encryption a key expansion procedure is done. Key expansion converts a key into 448 bits. That is the key submitted should be in multiples of 32 bit[12].Most important part of blowfish algorithm is its key dependent S-boxes.

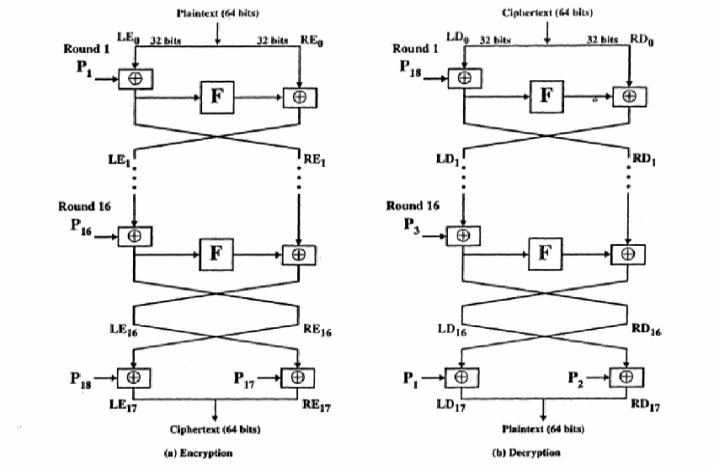


Fig. 5. Block Diagram of Blowfish

I.Key-expansion:

Blowfish converts a 448 bit key into several sub keys making the total size to be 4168 bytes. It is done by the following procedure:

1. First we take a p-array which consists of 18 values consisting of 32-bit sub keys named as p1, p2… p18.
2. Then we take four 32-bit S-boxes are used which have predefined values: -

S1,0, S1,1,..., S1,255
S2,0, S2,1,..,, S2,255
S3,0, S3,1,..., S3,255
S4,0, S4,1,..,, S4,255

The sub keys are generated by the following method:

1. p-array and the S-boxes are initialized with the hexadecimal values of pi.
2. Then we need to iterate the following operations till all p-array values are updated, the operation is XOR the value of p-array with the simultaneous 32 bits of the key till all the values of p-array are updated.
3. Then we encrypt a string of size 64 bits containing all zero with blowfish algorithm using the sub keys generated in p-array.
4. Then we replace the values of p1 and p2 with the output we got by encrypting the all zero string.
5. Now we again encrypt the output we gain by encrypting the string of 64-bit which contained all zero by using the updated p-array, which also contains the new values of p1 and p2.
6. Now we replace p3 and p4 with output we gained from the previous step.
7. We continue this same procedure until all values of p-array have been replaced.

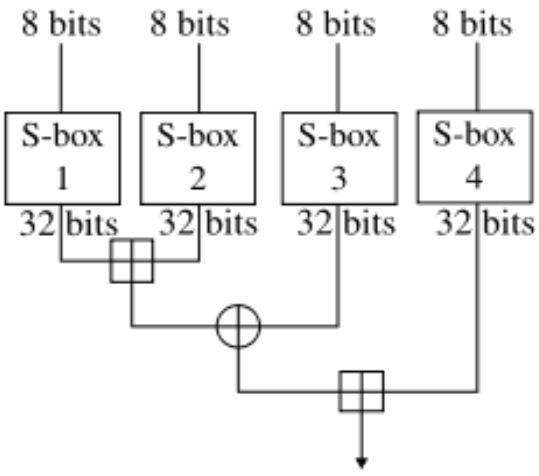


Fig. 6. Computation Performed by S-Box

II.Encrypting the plain text:

The steps to perform encryption are as follows:

1. Let the plain text be A of 64 bit.
2. We divide the plain text A into two parts each of 32 bits named as LA and RA.
3. Now we perform the following steps 16 times that corresponds to the 16 rounds of Blowfish algorithm:

a) First we XOR the left 32-bits of A named as LA with the value of p-array corresponding to that round and store it in LA.
b) Then we break LA into four parts of 8 bits each and process them from the 4 S-boxes shown in the figure 6. The final output which comes from this is named as X (LA).
c) Then we XOR the value of X (LA) with the value of RA and store it in RA.
d) Then we swap the value of LA and RA.
e) Then we XOR the value of LA with the value of p17.

4. Then after execution of these 16 rounds we XOR the value of p18 with the current value of LA.
5. At last we recombine the value of LA and RA.
6. The final cipher text is the combination of LA and RA.

III.Data decryption:

Decryption procedure is the opposite of encryption. The steps to perform decryption are as follows:

i. Let the cipher text be B of 64 bit.
ii. In the same manner as in Encryption the value of p-array and S-boxes are calculated.
iii. Here also we split the cipher text in two equal half named as LB and RB.
iv. Then we swap the value of LB and RB
v. Now we XOR LB with p18 and store it in LB.
vi. Then we follow the following iterations 16 times same as 16 rounds of Blowfish. The iterations are:

a. Here also we break LB into four parts of 8 bits each and process them from the 4 S-boxes. The final output which comes from this is named as X (LB).
b. Then we swap the value of LB and RB.
c. Then we XOR the value of LB with X (LB).

vii. After completion of following iterations we combine LB and RB to gain ciphertext.

3. Comparison

Table 1: Difference between different algorithms

Factors	AES	DES	T-DES	RSA	Blowfish
Developed	2000	1977	1978	1978	1993
Key Length	128,192,256 bits	56 bits	168 bits, 112 bits	>1024 bits	32-448 bits
Block Size	128 bits	64 bits	64 bits	Min. 512 bits	64 bits
Algorithm	Symmetric	Symmetric	Symmetric	Asymmetric	Symmetric
Security	Considered Secure	Weak	Weak	Not Secure	Considered Secure
Rounds	10/12/14	16	48	1	16
Keys Used	Same	Same	Same	Different	Same
Vulnerabilities	Not known	Differential and Linear Cryptanalysis	Brute force, Differential Cryptanalysis	Brute Force	Not Known

4. Conclusion

AES and Blowfish are the algorithms which are very secure till now but they have a restriction that they can only be used to encrypt and decrypt data by yourself that is the data cannot be shared between two people as then there will a need to share the key too which will make it insecure. While in case of RSA there is no need of sharing the keys as the keys used for securing the data and for decoding the data are separate and are known as Public key and Private Key, so it is the best method to share data between different parties. If RSA is combined with Elliptic curve cryptography then it will become the most secure asymmetric algorithm. As ECC is a lot faster asymmetric algorithm while RSA uses a key of very large size which ECC does not do. That is we combine RSA and ECC then they will become a very secure Asymmetric algorithm as they will hide the drawbacks of each other.

References

[1] Idrizi, Florim, Dalipi, Fisnik & Rustemi, Ejup. "Analyzing the speed of combined cryptographic algorithms with secret and public key". International Journal of Engineering Research and Development, August 2013.
[2] Padmapriya, Dr. A, Subhasri, "Cloud Computing: Security Challenges & Encryption Practices",IJARC, March 2013.
[3] Abdul.Mina, D.S, Kader, H.M., Abdual & Hadhoud, M.M. "Performance Analysis of SymmetricCryptography".
[4] Madhumita Panda "Performance analysis of encryption algorithms for security", IEEE SCOPES, Oct. 2016
[5] Chehal Ritika, Singh Kuldeep, "Efficiency and Security of Data with Symmetric EncryptionAlgorithms", IJARC, August 2012.
[6] Pooja Dixit, Avadhesh Kumar Gupta, Munesh Chandra Trivediand Virendra Kumar Yadav "Traditional and Hybrid Encryption Techniques: A Survey", Springer LNDECT, 2017
[7] Prashanti.G, Deepthi.S & Sandhya Rani.K., "A Novel Approach for Data Encryption Standard Algorithm". IJEAT, June 2013.
[8] R. Ahmad, W.Ismail"Performance Comparison of the Improved Power-Throughput AES and Blowfish Algorithms on FPGA", Springer LNEE, volume 398, 2016
[9] Sunitha K, Prashanth K.S., "Enhancing Privacy in Cloud Service Provider Using CryptographicAlgorithm", IOSR Journal of Computer Engineering ,2013.
[10] Das Debasis, Misra Rajiv, "Programmable Cellular Automata Based Efficient Parallel AES Encryption Algorithm", IJNSA, November 2011.
[11] Kalpana Parsi, Singaraju Sudha, "Data Security in Cloud Computing using RSA Algorithm", IJRCCT, September 2012.
[12] Praveen Kumar, Seema Rawat, Tanupriya Choudhury, Sanju Pradhan. "A performance based comparison of various symmetric cryptographic algorithms in run-time scenario", IEEE: System Modelling & Advancement in Research Trends, Nov. 2016.
[13] M. Rajesh, Manikanthan, "ANNOYED REALM OUTLOOK TAXONOMY USING TWIN TRANSFER LEARNING", International Journal of Pure and Applied Mathematics, ISSN NO:1314-3395, Vol-116, No. 21, Oct 2017.
[14] S.V.Manikanthan and K.srividhya "An Android based secure access control using ARM and cloud computing", Published in: Electronics and Communication Systems (ICECS), 2015 2nd International Conference on 26-27 Feb. 2015,Publisher: IEEE,DOI: 10.1109/ECS.2015.7124833.
[15] T. Padmapriya and V.Saminadan, "Improving Performance of Downlink LTE-Advanced Networks Using Advanced Networks Using Advanced feedback Mechanisms and SINR Model", International Conference on Emerging Technology (ICET), vol.7, no.1, pp: 93, March 2014.
[16] S.V.Manikanthan and T.Padmapriya "Recent Trends In M2m Communications In 4g Networks And Evolution Towards 5g", International Journal of Pure and Applied Mathematics, ISSN NO:1314-3395, Vol-115, Issue -8, Sep 2017.