A Novel Cryptographic Algorithm for Data Security in Manets

M .Madhurya[1], Dr.B.Ananda Krishna[2]
M.Tech[1] ECE dept Gudlavalleru Engineering College, professor[2] of ECE dept Gudlavalleru Engineering College,
E-Mail: *madhurya.honey@gmail.com[1], anand_bk@reddiff mail.com[2]*

Abstract:

In modern era, evaluation of networking and wireless networks has come forward to grant communication anywhere at any time. Mobile adhoc networks are the wireless infrastructure less networks can be easily formed or deployed due to its simple infrastructure. So the needs of protecting of such networks are increased by using the different encryption algorithms. Cryptology is a science that deals with codes and passwords. Cryptography provides solutions for four different security areas: confidentiality, authentication, integrity and control of interaction between different parties involved in data exchange finally which leads to security of information. This paper provides a fair performance comparison between various cryptographic algorithms on different settings of data packets. These settings include different data type, block size, key size, number of rounds, encryption/decryption time, CPU processing time, CPU clock cycles and power consumption. A novel cryptographic algorithm for data security is proposed in this paper to prevent the outside attacks to obtain any information from any data exchange in WLAN.This algorithm avoids the key exchange between users and reduces the time taken for encryption, decryption and authentication process. By using NOVEL algorithm we are expecting to generate data security with higher throughput & efficiency than already existing algorithms.

Keywords: *encryption, decryption, network security, adhoc WLANs.*

1. Introduction

Autonomous systems which comprise a collection of mobile nodes that use wireless transmission for communication is known as MANETS is shown in figure1. They are self–organized, self–configured and self-controlled infra-structure less networks. These networks are mainly used by community of users such as military, civilian and emergency services.

The advantages of the MANETS[3] are smaller in size, more convenient, more powerful, support high speed multimedia services, high mobility, device portability and low cost. And the limitations are securing data, link failures, power consumption and limited transmission range.

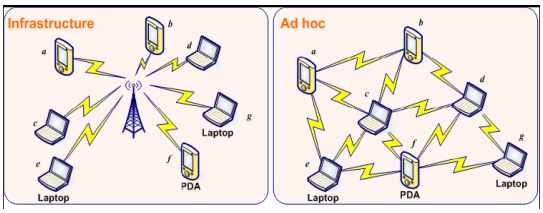

Fig.1: Types of wireless networks

The security in WLAN is based on cryptography [1] the science and art of transforming messages to make them secure and immune to attacks by authenticating the sender to receiver within the WLAN. The cryptography algorithms are divided into two groups[2]: symmetric encryption algorithms and asymmetric encryption algorithms.

Symmetric key encryption use only key to encrypt and decrypt data. Key plays an important role in encryption and decryption. The size of the key determines the strength of Symmetric key encryption. Symmetric algorithms are of two types: block ciphers and stream ciphers. The block ciphers are operating on data in groups or blocks. Stream ciphers are operating on a single bit at a time.

In **Asymmetric key** encryption, two keys are used: private keys and public keys. Public key is used for encryption and private key is used for decryption (e.g Digital Signatures). Public key is known to the public and private key is known only to the user. The most common classification of encryption techniques can be shown in Figure 2.

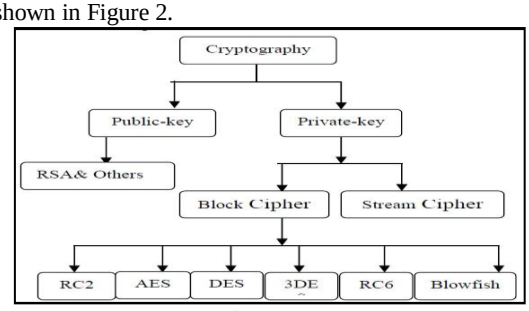

Fig.2: Overview of cryptographic algorithms

1.1 Brief History Of Encryption Techniques

DES: Data Encryption Standard was the first encryption standard published by NIST (National Institute of Standards and Technology)[4].The DES was developed and authorized by the U.S. government in 1977 as an official. It is based on the IBM proposed algorithm called Lucifer.

3DES: In Triple DES encryption method is similar to the one in the original DES but applied 3 times to increase the encryption level and the average safe time. 3DES is slower than other block cipher methods.

AES: Advanced Encryption Standard is a winning algorithm, Rijndael, which was developed by two Belgian cryptologists, Vincent Rijmen and Joan Daemen. AES provides strong encryption and was selected by NIST (FIPS-197) which can be used to protect electronic data.

RC2: "RC" stands for "Rivest Cipher" alternatively "Ron's Code". It is a variable-key-length cipher, Designed by Ronald Rivest in 1987 for RSA Data Security

RC6: It was designed by four analysts named Ron Rivest, Matt Robshaw, Ray Sidney, and Yiqun Lisa Yin. RC6 is block cipher derived from RC5. It was designed to meet the necessities of the AES competition.

Blowfish: Blowfish was designed in 1993 by Bruce Schneier as a fast alternative to existing encryption algorithms [4]. Blowfish can be used as a replacement for the DES algorithm. Blowfish is unpatented, license-free, and is available free for all uses. Blowfish is successor to Twofish. Though there is a complex initialization segment required before any encryption can

take place, the actual encryption of data is very efficient on large microprocessors.

This paper organized as follows: Related work in section 2. Analysis of existing algorithm in section 3. Proposed algorithm in section 4.Implementation and Expected results in section 5. The future scope in section 6 and the conclusion in section 7.

2. Related work

To give more prospective about the performance of the compared algorithms, this section discusses the results obtained from other resources.

In [5] energy consumption of different common symmetric key encryptions on handheld devices. It is found that after only 600 encryptions of a 5 MB file using Triple-DES the remaining battery power is 45% and subsequent encryptions are not possible as the battery dies rapidly.

In [6] AES is faster and more efficient than other encryption algorithms. When the transmission of data is considered there is insignificant difference in performance of different symmetric key schemes. Increasing the key size by 64 bits of AES leads to increase in energy consumption about 8% without any data transfer. The difference is not noticeable.

In [7] a study of security measure level has been proposed for a web programming language to analyze four Web browsers. This study consider of measuring the performances of encryption process at the programming language's script with the Web browsers. This is followed by conducting tests simulation in order to obtain the best encryption algorithm versus Web browser.

In [8] consider the performance of encryption algorithm for text files. AES, DES and RSA algorithms has been evaluated from the parameters like Computation time, Memory usage, and output bytes. Comparing these three algorithms they found RSA takes more time for computation process. The memory usage of each algorithm is considered as memory byte level. RSA takes more memory than AES and DES. Finally, the output byte is calculated by the size of output byte of each algorithm. The level of output byte is equal for AES and DES, but RSA algorithm produces low level of output byte.

In [9] it is conducted for different popular secret key algorithms such as DES, 3DES, AES, and Blowfish. They were implemented, and their performance was compared by encrypting input less of varying contents and sizes. The results showed that BlowFish had a very good performance compared to other algorithms. Also it showed that AES had a better performance than 3DES and DES. It also shows that 3DES has almost 1/3 throughput of DES, or in other words it needs 3 times than DES to process the same amount of data.

3.Analysis of existing algorithms

Longer key lengths mean more effort must be put forward to break the encrypted data security. So, here Blowfish has the longer key size as compared to other algorithms. Since the evaluation test is meant to evaluate the results when using block cipher, due to the memory constraints on the test machine (1 GB) the test will break the load data blocks into smaller sizes. The load data are divided into the data blocks and they are created using the Random Number Generator class available in System. This section we discusses the result obtained from added resources, to give more potential about the performance of the all the above described algorithms shown in table 1.

Table 1: performance comparison of symmetric algorithms

Algorithm	Key size (bits)	Block size(bits)	Number of Rounds
DES	64	64	16
3DES	168	64	48
AES	128,192 or 256	128	18
RC-2	64	64	18
RC-6	128,192 or256	128	16
BLOWFISH	From 8 To 448	64	16

To asses all the algorithms in this paper, the performance data is collected using a laptop with Pentium IV of 2.4 GHz CPU Speed, by which we encrypt a different range of file size from 49 K byte to 7.310 Mega Byte which were displayed in table 2. A number of performance metrics are calculated based on the following :

(a) Encryption/decryption time.

(b)CPU processing time: In the form of throughput

(c) CPU clock cycles and battery power.

The Encryption time is the time taken by an Encryption algorithm to generate a cipher text from a given plaintext. Encryption time is used to calculate the throughput of an encryption method which indicates the speed of encryption and similarly the decryption in table 3.

$$Throughput = \frac{Total\ plain\ text\ in\ bytes}{encryption\ time}$$

Throughput (Mega bytes/second)						
Input size (KB)	DES	3DES	AES	RC2	RC6	BLOW FISH
49	29	54	56	57	41	36
59	33	48	38	60	24	36
100	49	81	90	91	60	37
247	47	111	112	121	77	45
321	82	167	164	168	109	45
694	144	226	210	262	123	46
899	240	299	258	268	162	64
963	250	283	208	295	125	66
5345.28	1296	1466	1237	1570	695	122
7310.336	1695	1786	1366	1915	756	107
Average time	389	452	374	480.7	217	60.3
Throughput (MB/s)	4.01	3.45	4.174	3.25	7.19	25.89

Table 2: Throughput for encryption algorithms

Throughput (Megabytes/second)						
Input size (KB)	DES	3DES	AES	RC2	RC6	BLOW FISH
49	50	53	63	65	35	38

59	42	51	58	59	28	26
100	57	57	60	90	58	52
247	72	77	76	95	66	66
321	74	87	149	161	100	92
694	120	147	142	165	119	89
899	152	171	171	183	150	102
963	157	177	164	194	116	80
5345.28	783	835	655	904	684	149
7310.336	953	1101	882	1216	745	140
Average time	246	275.6	242	313.2	210	83.4
Throughput (MB/s)	6.35	5.67	6.45	4.985	7.43	18.72

Table 3: Throughput for decryption algorithms

The CPU process time is the time that a CPU is committed only to the particular process of calculations. It reflects the load of the CPU. The more CPU time is used in the encryption process, the higher is the load of the CPU. The CPU clock cycles are a metric, reflecting the energy consumption of the CPU while operating on encryption operations. Each cycle of CPU will consume a small amount of energy[10].

Key generation:

Select **p , q** — **p** and **q** both prime, **p ≠ q**

Calculate **n = p*q**

Calculate **Φ (n) = (p-1)(q-1)**

Select integer **'e'** — gcd (**Φ (n)**,**e**)=1; 1< **e < Φ(n)**

Calculate **'d'** — $d=e^{-1} \bmod \Phi(n)$

Public key — KU= **{e, n}**

Private key — KR= **{d, n}**

The encryption time is considered the time that an encryption algorithm takes to produce a cipher text from a plaintext. It reflects the load of the CPU. The CPU clock cycles are a metric, reflecting the energy consumption of the CPU while operating on encryption operations. Each cycle of CPU will consume a small amount of energy.

4. Proposed Algorithm

The novel cryptographic algorithm consists of modified RSA algorithm, which gives the data security with authentication. The encryption and decryption of this novel algorithm is shown in figure 4 and 5.

Encryption: The figure 4 shows the encryption algorithm considering a small example which can be described in the following steps:

- Here the plain text can be considered as **n** number of bits (i.e., 8,16,32,64,128,256). For example consider a 16-bit block of plain text (ie,1011110100011001).

Decryption:

Cipher text: **C**

Plain text: $\mathbf{M} = C^d \pmod{\mathbf{n}}$

- Perform circular left shift or rotation by **m** number of bits on the plain text (ie., if **m=2** then the resultant plain text is 1111010001100110) .
- Perform symmetric encryption with a symmetric key which yields cipher text 1 ie., C^1. The key must be the same size of the plain text.
- This C^1 is divided into two blocks (**a** & **b**) and then performs the initial permutation on both the sides.
- The input to the permutation block is an 8-bit block of cipher text 1, which we perform using IP function. Here the first 8-bits are assigned to '**a**' block and next 8-bits are assigned to '**b**' block.
- By considering this IP block we do the permutation to each block.

IP
2 6 3 1 4 8 5 7

a-block = 8 bits:

1 2 3 4 5 6 7 8: 1 1 1 1 0 1 0 0
2 6 3 1 4 8 5 7: 1 1 1 1 1 0 0 0
b-block = 8 bits:
1 2 3 4 5 6 7 8: 0 1 1 0 0 1 1 0
2 6 3 1 4 8 5 7: 1 1 0 0 0 0 0 1

- Again perform left shift for '**a**' and right shift for '**b**' by **S** number of bits (ie., if **S** =3 then the resultant is as follows.

a-block = 8 bits:
Permuted bits: 1 1 1 1 1 0 0 0
Left shift by 3 bits: 1 1 0 0 0 1 1 1
b-block = 8 bits:
Permuted bits: 1 1 0 0 0 0 0 1
Right shift by 3 bits: 1 0 0 1 1 0 0 0

- Then combine both the blocks and perform the function of RSA algorithm to generate the final cipher text 2 by adding a public key to the function block. Public key is the key which is the known key to all users in the network. The key generation, encryption and decryption process of RSA algorithm steps are shown in figure [3].

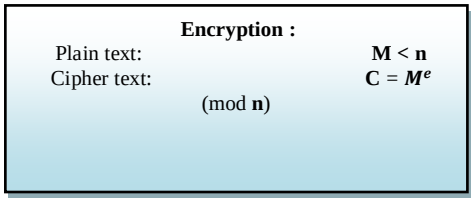

Fig3: The RSA algorithm at function block

This gives the cipher text 2 which is the data to be transmitted over the network. This entire process is shown in the figure [4] that gives the encryption of the pain text which is to be send in a network.

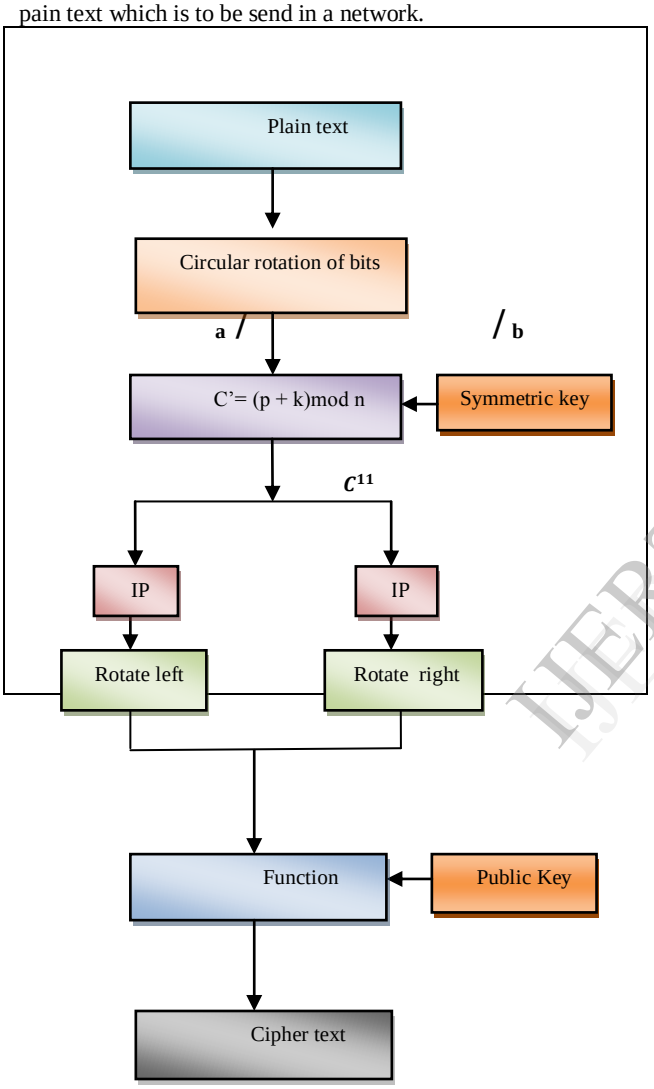

Fig4: Encryption of plain text

Decryption:

The figure [5] shows decryption algorithm, in which the private key is generated and decrypted the received cipher text2, produces cipher text 1. Decryption is the reverse process of encryption algorithm.

Plain text is a readable message or data that is fed into the algorithm as input.

Cipher text is a scrambled message produced as output. It depends on the plain text and the key. For a given message two different keys will produce two different cipher texts.

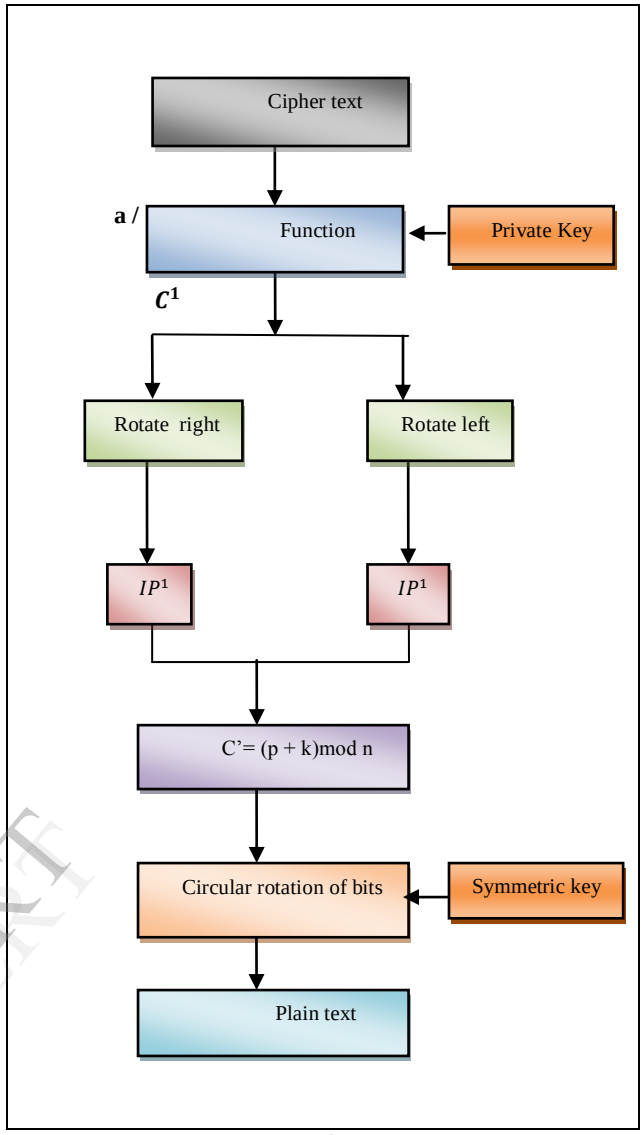

Fig 5:Decryption of cipher text

At the end of the algorithm inverse permutation is used.

IP^{-1}
4 1 3 5 7 2 8 6

it is easy to show by example that the second permutation is the reverse of the first permutation ie.,

[IP^{-1} (IP (X)) = X]

5. Implementation and expected results

The proposed algorithm is going to simulate in GLOMOSIM. We are going to develop a stronger Encryption Algorithm which gives high throughput, less processing time and low power consumption. The algorithm will help the authentication protocols to reduce the delay taken by them, and gives the channel the data security wanted. So that the experiments can be performed on data, image & audio.

6. Future Scope

We need a way to make decisions about energy consumption and security to reduce the consumption of battery powered devices. We examine a method for analyzing trade-offs between energy and security. The goal is to aid the design of energy efficient secure communication with authentication schemes for the wireless environment in the future.

7. Conclusion

We are expecting the results to show the superiority of proposed algorithm with others in terms of the throughput, processing time and power consumption. More the throughput, more the speed of the algorithm & less will be the power consumption. The algorithm will help the authentication protocols to reduce the delay taken by them, and gives the channel the data security wanted. The programs ensure the key updated without any problem on the decryption of the text or the image, and show that the algorithm reduce the time used in the encryption or decryption process. It is efficient and useable for the security in the WLAN systems.

References

[1] Diaa Salama, Hatem Abdual Kader, and Mohiy Hadhoud, "Studying the effects of Most Common Encryption Algorithms", *International Arab Journal of e-Technology, Vol. 2, No. 1, January 2011*

[2] W.Stallings, "Cryptography and Network Security 4th Ed," *Prentice Hall, 2005, PP. 58-309.*

[3]Subir kumar sharkar, T.G.Basavaraju, C.Puttamadappa, "Adhoc mobile wireless networks*". Principles, protocols and applications.*

[4] Diaa Salama, Hatem Abdual Kader, and Mohiy 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, Sept. 2010*

[5] El-Fishawy.N(2007)," Quality of Encryption Measurement of Bitmap Images with RC6,MRzC6, and Rijndael Block Cipher Algorithms*", International Journal of Network Security, PP.241–251.*

[6] Ruangchaijatupon.P, Krishnamurthy.P (2001), "Encryption and Power Consumption in Wireless LANs-N". The Third IEEE Workshop on Wireless LANs - Newton, Massachusetts.

[7] Shih.E, Cho.S, Ickes.N, Min.R, Sinha.A, Wang.A, and Chandrakasan.A(2001), "Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks," in Proceedings of *The 7th ACM Annual International Conference on Mobile Computing and Networking (Mobi Com), Rome, Italy,pp.272-287.*

[8] Shashi Mehrotra Seth, Rajan Mishra on "Comparative Analysis Of Encryption Algorithms For Data communication " *in IJCST Vol. 2, Issue 2, June 2011 I ,pp. 292-294*

[9] A. Nadeem and M. Y. Javed, "A performance comparison of data encryption algorithms," *Information and Communication Technologies, ICICT 2005,* pp.84-89, 2005.

[10] Gurjeevan Singh, Ashwani Kumar Singla, K.S. Sandha," Performance Evaluation of Symmetric Cryptography Algorithms".