INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS…..

DYNAMIC ENCRYPTION-DECRYPTION ALGORITHM FOR IMPROVING DATA SECURITY

Vijay Kumar[1], Suman Deswal[2]

[1] M.Tech (CSE), [2]Assistant Professor in CSE Department
[1]DCRUST, Murthal
[1]vijaybhattoo@yahool.com, [2]suman_gulia2000@yahoo.co.in

Abstract: *Information security is a necessary part of any organization because all government, business, private and academic authorities are connected to each other through a network to exchange the information. Thus to protect the information from the unauthorized users there is a need of security algorithms. Network security is achieved through the use of cryptography. Many new encryption techniques are introduced day by day in the field of cryptography to protect the information. In this paper a new dynamic encryption-decryption schemes is presented to reduce the cryptanalysis attack. Because dynamic keys are the one time used key, they can surely improve the level of security. In this algorithm Dynamic keys are generated through a random number and S-Box .At last, this new algorithm will be compared with the AES symmetric algorithm.*

Keywords: *Encryption, decryption, dynamic keys, S-Box, WLAN.*

1. INTRODUCTION

Security of a wireless network still ranks as one of the largest concerns of IT professionals planning to roll out an enterprise wireless LAN. In Wireless Local Area Networks (WLAN) major issues are associated with the security problems. Many people erroneously believe that a wireless LAN is inherently insecure. The wireless signal of the WLAN is broadcast through the air in all directions simultaneously. An intruder can easily capture this signal using freeware tools to exploit WLAN vulnerability. WLANs are increasing regularly in home and business environment due to the low cost of wireless devices [1]. WLAN gives mobility and flexibility to users in homes and hot spot environments, such as airports and campuses. The wide range of usage emphasizes the importance of having a secure network and protect from potential breaks. In order to do so, mostly encryptions such as WEP and WPA/WPA2 are used [2]. This allows the transmitted data within the network to be encrypted.

Cryptography is important in the field of Information Security because on the surface it is about making something secret, but it is also about controlling access, specifying who can get to information under what terms. It is also necessary to make sure that secret which is communicated to different users over a communication system remains secret. The cryptography make sure that the information communicated to the legitimate users and also received from the legitimate users and also in future the above said fact would have been proved.

Cryptography is the study of mathematical techniques related to aspects of information security, such as confidentially or privacy, data integrity and entity authentication [3]. Cryptography is not only means of providing information security, but rather one set of techniques. Confidentially means keeping information secret from all but those who authorized to see it[4]. Data integrity means ensuring information has not been altered by unauthorized or unknown means. Entity authentication means corroboration of the identify of an entity.

There are main two types of cryptography: Symmetric key and Asymmetric key [5].

a) Symmetric key cryptography: The number of keys used. If both sender and receiver use the same key, the system is referred to as symmetric key cryptography.

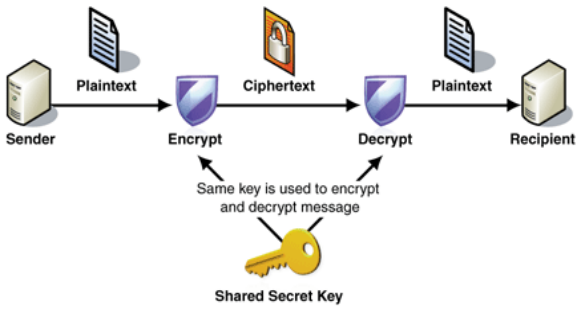

Figure1.1 Symmetric key Cryptography

b) Asymmetric key cryptography: If the sender and receiver each use a different key, the system is referred to as

asymmetric, two-key, or public-key encryption. The method is more secured as compared to private key cryptography but it consumes more power and takes more processing time therefore extra hardware is required. Due to increase in the computational unit the overheads are high in public key cryptography.

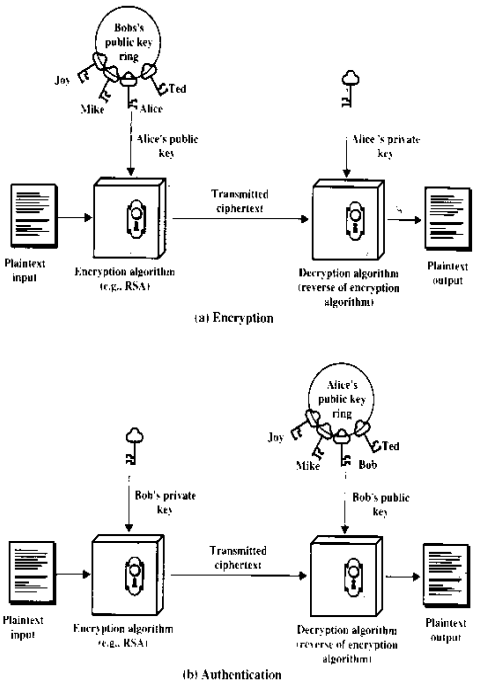

Figure1.2 Asymmetric key Cryptography

2. LITERATURE REVIEW

The strength of any dynamic key generation algorithm will be depend on the two factors first one is less sharing of information between the both communicated parties and second is dynamic key less dependent on other factor (in ideal case it will random)[6].

2.1 One time pads

There is a perfect encryption scheme. It's called a one-time pad, and was invented in 1917 by Major Joseph Mauborgne and AT&T's Gilbert Vernam which was explained in detail by Kahn. Classically, a one-time pad is nothing more than a large non repeating set of truly random key letters, written on sheets of paper, and glued together in a pad. In its original form, it was a one-time tape for tele type writers [7]. The main idea of one time pad is to avoid long term shared cryptographic keys. In other words, when the one-time pad is truly random, it is difficult to break by analyzing successive messages. In one time pad systems, the pads are shared between senders and receivers. To decrypt the messages, the decrypted pads at the receivers must be the same as the encrypted pads at the senders [7]. Therefore, these pads must be distributed between the parties. In practice, the distribution of pads between parties over networks is the weak point in one time pad systems. Similar to current security systems, symmetric cryptographic keys that are used to secure communication messages require secure key exchange among parties before the communication messages are sent. Normally, the key exchange can be performed via public key algorithms like Dife-Hellman or MQV . Every cryptographic key is only secure for a certain amount of time. In addition, larger keys often require higher computational resources, especially in asymmetric cryptography. In practice, excessively large keys may admit denial of service possibilities whereby adversaries can cause excessive cryptographic processing. However, the security of these algorithms relies on long term shared keys that contradict the original idea of one time pad. However, increasing the cryptographic key size is not always the best solution, since no matter how large the key is its cryptography is still ultimately breakable.

2.2 Pseudo random number generators

For the cryptography mechanism lot of random numbers would be needed for the purpose of creating random keys. The best a computer can produce is a pseudo-random-sequence generator [8]. A pseudo-random sequence is one that looks random. The sequence's period should be long enough so that a finite sequence of reasonable length that is, one that is actually used is not periodic. If is has been needed a billion random bits, don't choose a sequence generator that repeats after only sixteen thousand bits. These relatively short non periodic subsequences should be as indistinguishable as possible from random sequences. There are lots of PRNG's recommended for highly sophisticated applications by Campbell and Easter in FIPS which can be used for creating the random keys at the run time.

A PRNG suitable for cryptographic applications is called a cryptographically secure PRNG (CSPRNG)[9].A requirement for a CSPRNG is that an adversary not knowing the seed has only negligible advantage in distinguishing the generator's output sequence from a random sequence. In other words, while a PRNG is only required to pass certain statistical tests, a CSPRNG must pass all statistical tests that are restricted to polynomial time in the size of the seed.

3. NEW PROPOSED SCHEME

The proposed algorithm is based on stream cipher which implements XOR operation to generate cipher text. Dynamic keys are generated using random numbers[10] .In the each type of cryptographic algorithm there is always a key is shared between sender and receiver[11]. In this algorithm, there is no need of sharing any type of keys .This is one of

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS…..

the advantage as compared to other algorithm proposed till now.

In the encryption algorithm, from the 4*4 s-box (key pattern) a randomly key is generated and that key is used to encrypt the message into cipher text. Every time a new key is used to encrypt the part of the message. This key is unknown for everyone. To decrypt the cipher text into plaintext, hacker must have that key. Without knowing the decryption algorithm, hacker can never convert the cipher text into plaintext.

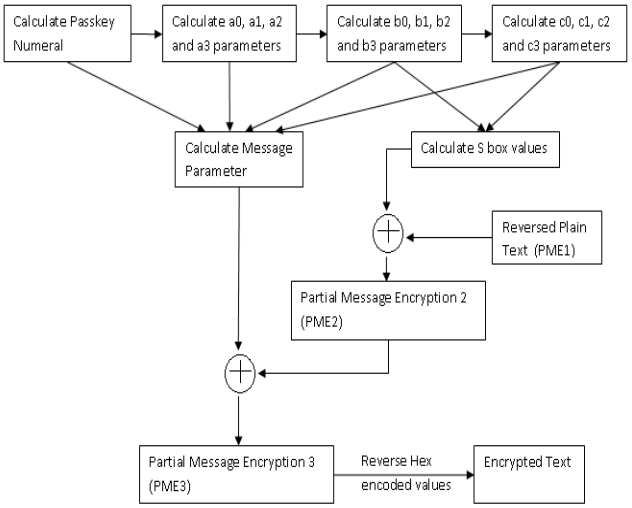

Figure 3.1 Encryption Overview

3.1 Encryption Algorithm

The steps of encryption algorithm are:

Step 1: Calculate Passkey Numeral:
a. Random number is generated between 1024 and 999999.
b. Length of number is calculated.
c. Sum of ASCII value of digits of number are calculated
And, Passkey Numeral= Numeral Length (obtained in step b.) + Sum of ASCII value of digits (obtained in step c.)

Step 2: Calculate a0, a1, a2 and a3 parameters
a. a0= Sum of digits at even positions of passkey numeral
b. a1= Sum of digits at odd positions of passkey numeral
c. a2= Product of digits of passkey numeral
d. a3= (Passkey numeral) mod (256)

Step 3: Calculate b0, b1 and b2 parameters:
In order to compute b0 value, encryption parameters EP1, EP2, EP3 and EP4 are required:
a. EP1= a0 XOR a1
EP2= EP1 + 15
EP3= a2 XOR a3
EP4= EP3 + 55
And, b0= EP1 + EP2 + EP3 + EP4
b. EP1= a0 XOR a2
EP2= EP1 + 25
EP3= a1 XOR a3
EP4= EP3 + 65
And, b1= EP1 + EP2 + EP3 + EP4
c. EP1= a0 XOR a3
EP2= EP1 + 35
EP3= a1 XOR a2
EP4= EP3 + 75
And, b2= EP1 + EP2 + EP3 + EP4

Step 4: Calculate c0, c1, c2 and c3 parameters:
c0 = ((EP1[b2] XOR EP2[b2]) * a0) + b2
c1 = ((EP1[b1] XOR EP3[b1]) * a1) + b1
c2 = ((EP1[b0] XOR EP4[b0]) * a2) + b0
c3 = ((EP2[b2] XOR EP3[b2]) * a3) + b2

Step 5: Calculate s-box values:

(EP1 XOR c0) * c0	(EP1 XOR c1) * c0	(EP1 XOR c2) * c0	(EP1 XOR c3) * c0
(EP2 XOR c0) * c1	(EP2 XOR c1) * c1	(EP2 XOR c2) * c1	(EP2 XOR c3) * c1
(EP3 XOR c0) * c2	(EP3 XOR c1) * c2	(EP3 XOR c2) * c2	(EP3 XOR c3) * c2
(EP4 XOR c0) * c3	(EP4 XOR c1) * c3	(EP4 XOR c2) * c3	(EP4 XOR c3) * c3

Figure3.2: 4*4 S-Box

Step 6: Calculate Message Parameter:
Message Parameter =Passkey Numeral (obtained in step 1) + Randomly generated key between 1024 and 9999 + Average of a0, a1, a2 and a3 parameters (obtained in step 2) + Average of b0, b1 and b2 parameters (obtained in step 3) + Average of c0, c1, c2 and c3 parameters (obtained in step 4)

Step 7: Message Encryption:
a. Reverse the plaintext to be encrypted to obtain Partial Message Encryption 1 (PME1).
b. Perform PME1 XOR S-box [index1][index2] (obtained in step 5) operation to obtain Partial Message Encryption 2 (PME2).
c. Perform PME2 XOR Message parameter (obtained in step 6) operation to compute Partial Message Encryption 3 (PME3).
d. Reverse binary/ hexadecimal value of PME3 to compute Partial Message Encryption 4 (PME4).
e. PME4 is split into group of 8 bits and converted into corresponding ASCII character to form encrypted text and is appended to file along with text delimiter.

3.2 Decryption Algorithm

Step 1: Calculate the length of decrypted message.
Step 2: In order to calculate the decrypted_msgparameter
a. Replace the "\r\n" by " ".
b. Replace ":" by " " .
Step 3: In order to calculate the decrypted_sbox
In the decrypted S_box Replace ":" by " ".
Step 4: In order to calculate msg_encryption5 after steps 1, 2, 3 reverse the Decrypted message.
msg_encryption5 is a decrypted message array.
Step 5: Calculate the msg_encryption6

INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS…..

msg_encryption6=parse the msg_encryption5 into Hex i.e 16.

Step 6: Calculate msg_encrypion7

msg_encryption6=msg_encryption6 XOR decrypted_msgparameter

msg_encryption7=msg_encryption6 XOR decrypted_sbox

Convert the msg_encryption7 into character.

Step 7: Reverse the decrypted message character.

Step 8: Concatenate the decrypted character into string and this will be the original message.

4. IMPLEMENTATION RESULTS

4.1 Algorithm Efficiency

The proposed algorithm is compared with the AES symmetric algorithm.

The algorithm efficiency has been measured on basis of:

1. Execution time

Total execution time for AES algorithm and proposed algorithm is respectively 231ms and 109ms.Thus the proposed algorithm has faster processing than AES algorithm.

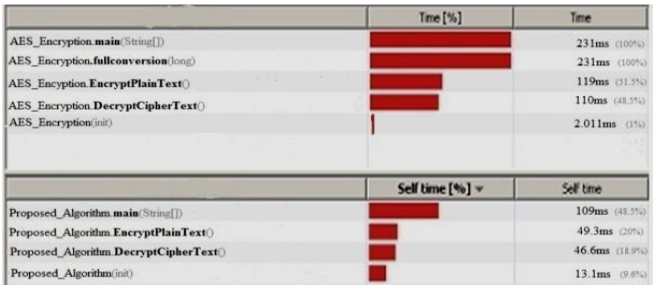

Figure4.1: Comparison between Execution time of AES and Proposed Algorithm

Table 4.1 Comparison between execution time of AES algorithm and proposed algorithm

Algoritm	Full conversion time	Encryption time	Decryption time	others
AES algorithm	231ms	119ms	110ms	2.011 ms
Proposed algorithm	109ms	49.3ms	46.6ms	13.1 ms

2. Memory requirements

Total and available heap size has been computed.

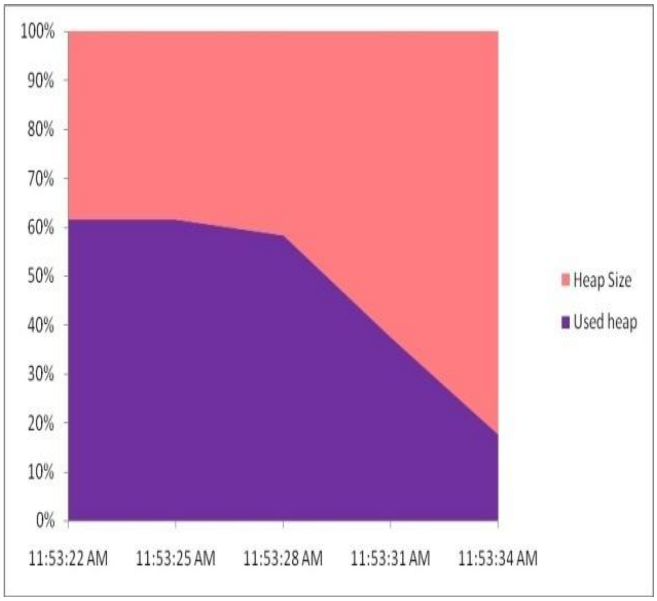

Figure4.2: Heap Size during AES Encryption/Decryption Algorithm

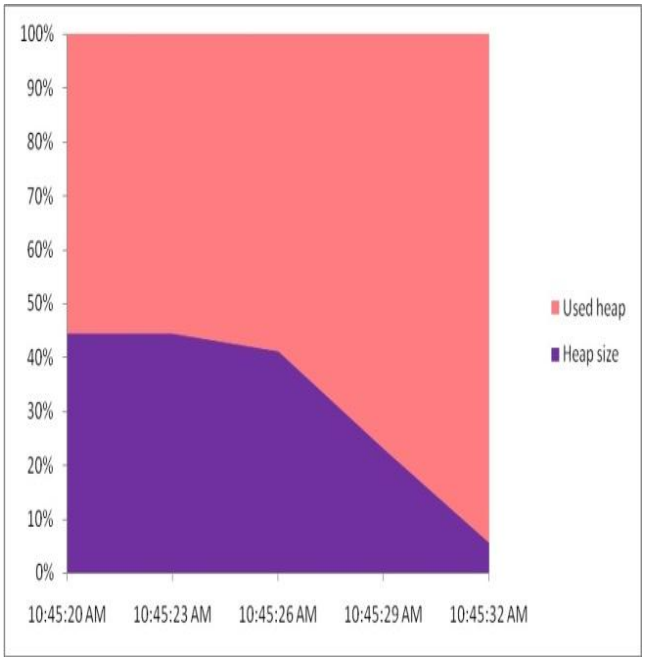

Figure 4.3: Heap Size during proposed algorithm Encryption/Decryption Algorithm

In terms of memory requirement, the proposed algorithm utilizes very less memory as compared to AES algorithm.

Table 4.2 Comparison between memory used by AES and Proposed algorithm

Algorithm	Heap Size	Used Heap

AES Algorithm	100%	47.5%
Proposed Algorithm	100%	34.75%

4.2 Advantages of Proposed Algorithm

a. The algorithm has a faster processing time of 109 ms for plain text of 300KB size.
b. Being a stream cipher; it works on only a few bits at a time, thus consuming less memory.
c. Bytes are individually encrypted without association with other chunks of data, thus are less susceptible to transmission noise because in case of erroneous modification of one part of data, rest of data is still recoverable.
d. This algorithm provides easy integration with mobile simulator without performance impact.

4.3 Limitations of Proposed Algorithm

a. Larger S-box matrix impacts performance when integrated with mobile simulator.
b. This algorithm being a stream cipher can only work in forward direction unlike block ciphers which can work in both directions.

5. CONCLUSIONS

From the result it is clear that our "proposed technique" is better result producing as compared "Symmetric Data Encryption through AES Methodology". Dynamic keys can reduce the cryptanalysis attack because every time a new key is used for encryption process. In this algorithm, there is no need of sharing any type of keys .This is one of the advantage as compared to other algorithm proposed till now. This approach provides a huge amount of keys, which increases the security level and makes it almost impossible for hacker to predict the key.

REFERENCES

[1] **Arash Habibi Lashkari, Masood Mansoori, Amir Seyed Danesh, "Wired Equivalent Privacy (WEP) versus Wi-Fi Protected Access (WPA)", in ICCDA Singapore Conference, 2009**

[2] **Jiang li. Moses Garuba "Encryption as an Effective Tool in Reducing Wireless LANVulnerabilities "Fifth International Conference on Information Technology 2008IEEEMarc Parenthood, Patrick Rainier, Jacques Tissue.**

[3] **Ayushi, " A Symmetric Key Cryptographic Algorithm "International Journal of Computer Applications (0975 8887) Volume 1 – No. 15**

[4] **Ajay Kakkar,M.L. Singh,P.K. Bansal: "Comparison of various Encryption Algorithms" IJET, Jan-2012.**

[5] **Behrouz A. Forouzan,"Cryptography & Network Security" Tata McGraw Hill, ISBN 13-978-0-07-066046-5.**

[6] **N.Yuvaraj,D.Manikandan,Dr.V.Parthasaray:Generate Dynamic key on Asymmetric Key Cryptography Infrastructure,IJERT,Dec-2012.**

[7] **N.Nagaraj, V. Vaidya and P.G. vaidya, "Revisiting the one-time pad," International Journal of Network Security,Vol. 6,no. 1 ,pp.94-102,2008**

[8] **Matsumoto, M. and Nishimura, T. Dynamic Creation of "Pseudorandom number generator. in Niederreiter, E.H. and Spanier, J. eds. Monte Carlo and Quasi-Monte Carlo Methods",Springer,2000,56-59**

[9] **Matsumoto,M.and Nishimura,T. Mersenne twister"a 623- dimensionally equidistributed uniform pseudo-random number generator" ACM Transactions on Modeling and Computer Simulation(TOMACS) 8(1). 3-30.**

[10] **Hamid Mirvaziri,Kasmiran Jumari mahamod Ismail and Zurina Mohd Hanapi, "Message Based Random Lenght Key Encryption Algorrithm" Journal of Computer Scinece,2009.**

[11] **Naim Ajlouni,Asim EL-Sheikh, and Abdullah Abdali Rashed "A New Approach in Key generation and Expansion in Rijndael Algorithm",IAJIT,Vol.3,Jan-2006.**