ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 2, February 2015

Implementation of LiSP using Different Random Number Generator as a Dynamic Key for Wireless Sensor Network

Ankita Ojha[1], Kusumlata Jain[2]

Department of Information Technology, Banasthali University, Jaipur, India[1]

Department of Computer Science, Banasthali University, Jaipur, India[2]

Abstract: Wireless sensor networks are a current highly developed technology of networks and are employed in many application areas like in military for surveillance purpose. WSN gather data from sensor node and send it to another sensor node. Due to the distributed nature, restricted computing power, memory space and use in some sensitive application, security is a major problem, so security is very important to transfer the data from one node to another node. To provide security in this LiSP protocol is used. This paper include how much energy is consumed for secure communication between sensor nodes and base station. In this paper, encryption and decryption operation use the block cipher rather than stream cipher and key is used to create security protocol. Encryption algorithm contains two inputs, one is the user input another is the key. To produce lightweight security protocol dynamic key is used. In this paper dynamic key is generated by using different types of random number generator like RC4, Blum Blum shub and wichman random number generator. After that find out how much time and energy is consumed by this protocol. This paper contains protocol which provide secure and efficient data transmission withminimum energy and time consumption so it makes it harder and difficult to break the security of data.

Keywords: Wireless sensor networks, dynamic key, RC4, blum blum shub, whichman, encryption, decryption, one time pad, lightweight protocol.

I. Introduction

A sensor network is the collection of multiple tiny nodes which has the capability of sensing, computing and transmitting information to the base station or we can say that it sense the environmental condition and send to the network. Wireless sensor network is created by distributed independent devices which are mainly used to sense the environmental condition such as temperature. Network model of wireless sensor network consist of sensor field, sensor node, base station and user. Sensor field is the area in which all the sensor nodes are placed. Sensor node is the tiny nodes which sense the environmental condition and routing these information to the main network. Base station is the central part of the network which extracts the information from the all nodes and distribute to all other receivers. User which extract the information from the base station do desire task and take decision from these information.

This paper consist how to create lightweight security protocol by using different types of random number generator algorithm. in this paper lightweight security protocol using park-miller, RC4, Blum Blum Shub and Wichman – Hill. Security is a big issue in wireless sensor network. In WSN sensor nodes are communicate through wireless medium, it has the distributed nature all these reason and deployment in remote area these networks are vulnerable to security attacks which affect the proper functionality of the sensor network. Major application of sensor network is in military and civilian application, which contains some sensitive information if proper security algorithm is not applied over it than authenticate data is accessed by intruders and attackers. Any algorithm which is used in any security method contains two parts. one is the user input and second part is the key. Key plays an important function in encryption. To produce the key pseudo random number is used.

Park-Miller random number generator algorithm is used to generate 32-bit sequences of key. Park – miller uses the stream cipher rather than block cipher. This method use the asymmetric key for the encryption and decryption means one key is used for encryption and another key is use for decryption.

RC4 algorithm generates a pseudo random key stream that is used to generate the ciphertext. This algorithm called pseudo random number generator because it generates a sequence of numbers that satisfy the properties of random numbers. This algorithm contains two parts first is key scheduling algorithm and second one is pseudo random number generation algorithm. The keystrem contains 256 bytes array. When array has been initialized and shuffled with the key scheduling algorithm, it is used and modified in the pseudo random generation algorithm to generate the keystream.

Blum Blum Shub algorithm is also used to generate the pseudo random number generator. This algorithm proposed by Lenore Blum, Manuel Blum and Michael Shub so it named as Blum Blum Shub random number

DOI 10.17148/IJARCCE.2015.4295

generator. This algorithm uses two large prime numbers. The generator is not appropriate for use in simulation because it is not very fast it take large time when large volume of data is used.

The wichman – Hill algorithim is a three seed pseudo random number generator. The algorithm generates numbers between 0.0 and 1.0 with a cycle of. According to Bevington the algorithm is free of correlations and is a good generator for most purpose .

II. PROPOSED METHOD/ ALGORITHIM

In this paper proposed a method based on the dynamic key encryption. In the proposed work every message is encrypted and decrypted by using the same key and main feature of this key is that this is dynamic means which is not same for every pair of encryption and decryption which is removed after every pair of encryption and decryption than new key is generated for new block of message.

A key generating function is called after every decryption operation [4] and a new key is produced to perform the encryption and decryption. This concept is similar to one time pad. The proposed algorithm is implemented on MATLAB 7.0.

A. *RC4 Algorithm*

RC4 algorithm use symmetric key encryption. It is a stream cipher. This algorithm used in SSL and TLS between web browsers and servers.

This algorithm divided into two parts:

1) Key Scheduling Algorithm (KSA)
2) Pseudo Random Number Generation Algorithm (PRNG)

Key Scheduling Algorithm which mainly used to generate state array. It uses 256 bytes array, the values in the array are equal to their index. Use the secret key to initialize and permutation of state vector S and it uses 8 – bit index state pointer.

```
for   m = 0 to 255 do

S[m] = m;

T[m] = K [m mod K ];

n = 0;

for  m = 0 to 255 do

n = ( ( n + S(m) + T(m) )  mod 256);

swap ( S[m] , S[n]);
```

where

S = S is a set equal to the values from 0 to 255
T = Temporary vector
K = Key length

Pseudo Random Number Generator which generate key stream than XOR key stream with the data to generated encrypted stream.

- Generate key stream k, one by one
- XOR S[k] with next byte of message to encrypt and decrypt.

```
m = n = 0;

whie ( more byte to encrypt )

m = ( m + 1 ) mod 256 ;

n = ( n + S[m] ) mod 256;

swap  ( S[m] , S[n] );

k= ( S[m] + S[n] ) mod 256;

C = M XOR S[k] ;
```

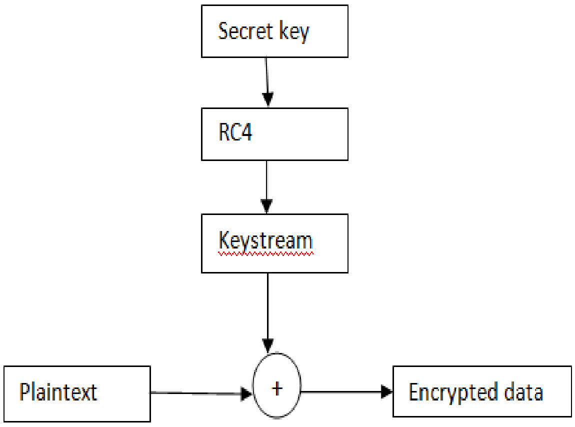

Figure 1: RC4 method

B. *Blum Blum Shub Algorithm*

This algorithm generate pseudo random number generator. It contains two odd prime number p and q.
Let a and b be two odd primes

m = p * q

x_0 = seed value

$x_{n+1} = x_n{}^2$ mod m

This algorithm is provably secure because using

- Euler's criterion
- Legendre symbol
- Jacobi symbol,
- Composite quadratic residues.

DOI 10.17148/IJARCCE.2015.4295

The basic application of Blum used in probabilistic public key encryption. It mainly generate the key stream during encryption and decryption.

C. *Wichman – Hill Algorithm*

This algorithm produce the output in the range of 0 and 1 but it may also produce the negative number.

Algorithm

ia, ib, ic should be set to integer values between 1 and 30000

```
ia = 171 * mod ( ia , 177);

ib = 172 * mod ( ib , 176);

ic = 170 * mod ( ic , 178);

if ( ia < 0)
ia = ia + 30269 ;

if ( ib < 0)
ib = ib + 30307 ;

if ( ic < 0)
ic = ic + 30323 ;

then
ia = mod (171 *  ia , 30269);

ib = mod (172 * ib , 30307);

ic = mod ( 170 * ic , 30323);
```

The result generate the random numbers.

Rand = mod (float (ia) / 30269.0 + float (ib) / 30307.0 + float (ic) / 30323.0);

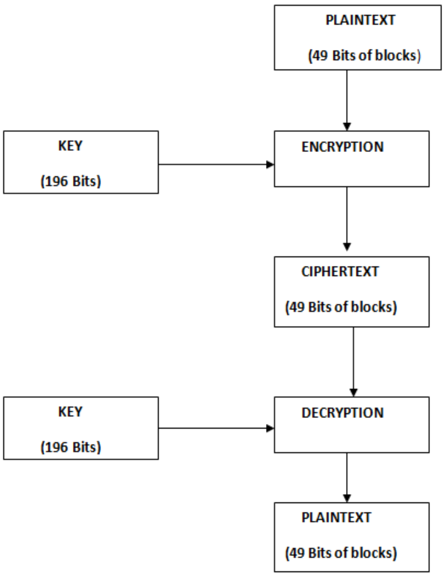

Figure 2: Planned Method

D. *Park – Miller Algorithm*

Park miller algorithm is also used to generate random numbers. It generats 32 – bit random sequence. This sequence is used in security algorithm to perform encryption and decryption. This sequence of key paerfom encryption operation with plaintext and ciphertext to produce secure ouput to user.

Park-Miller metod [2] is based on the LCG:

$Y_{n+1} = A\ Y_n \bmod (2^{31}-1)$.

Where [1] the constant value “P” chosen as 15,997. Constant

P=157557; (2 < P < M-1)

M=2197483747; $(2^{31}-1)$

Q=13877; (M div P)

R=158805; (M mod P)

```
Var

High_val: =I_seed div Q;

Low_val: =I_seed mod Q;

Test_caes: =P*low_val – R*high-val;

if test_case > 0 then

O_seed:= test;

Random_num: =O_seed / M;

Else

O_seed:= test_case + M;

Random_num: =O_seed / M;

end;
```

E. *Dynamic Key*

In the proposed work key is an important part for any security algorithm. In this lightweight security algorithm dynamic key is used which is created by using Blum Blum Shub, Wichman-Hill, RC4 random number generator. Dynamic key is also use the concept of one time pad. This key has important feature is that for each couple of the encryption and decryption new key is generated and last one is discarded. In proposed work user enter a key. The entered key is divided into 49 bits of block. An inbuilt key ‘ibk’ is concatenated with the key which is entered by user for producing a matrix of 14X14. A random function (FR_r) is used to produce the key. A random function contains various matrix operations like addition, multiplication, substraction, XOR and shifting etc. In which the random number X is added to the final matrix to produce the dynamic key. [6][7]

DOI 10.17148/IJARCCE.2015.4295

- *Key Generation Algorithm*

Steps of the algorithm are as follows:

1) *MATLAB Implementation for Randomize Function (FR_r)*

This randomize function use the different array to calculate the the dynamic key KEYDY.

```
I [si] [1] = EK;
% Text entered by user.

J [1] [si] = EK;
%Text entered by user.

K [14] [1] = find(ibk,14,'first') ;
% find first 14 character of ibk.

L [1] [14] = find(ibk,14,'last') ;
% last 14 character of ibk.

M [14] [si] = K [14] [1] * J [1] [si];

N [si] [14] = I [si] [1] * L [14] [1] ;

O [14] [14] = M [14] [si] * N [si] [14] ;

P [14] [14] = O [14] [14] + $Z_1$ ;

KEYDY  = P [14] [14] ;

%  KEYDY is a dynamic secret key.
```

2) *Determine the number of plaintext blocks*

```
   PText = Text enter by users

Si = length (PT)
% Length of the plaintext enter by user.

KEY = Si mod 49 ;

IPad = 49 –KI ;

 % number of padding bits

NPText = (PText - KEY) / 49;
% number of plaintext blocks of 49 bits.

If (KEY==0)

NPText = NPText

Else

NPText = NPText + 1

For (j = 1 to N)

{

CALL ENCRYPTION PROCESS

}
```

III. ENERGY CONSUMPTION FOR SENDING AND RECEIVING

Energy consumption is also an issue in wireless sensor network because of limited energy supply to sensor nodes. Enery available at nodes should play an important role in protocols. Energy consumption is also depends on distance between sensor nodes and base station. Nodes which has maximum distance between them consume maximum energy and node which has less distance between them consume less energy.

Energy consumption formula for transmitting a packet with L length and d distance. E_{elec} is the amount of energy utilized per bit to run the transmission and receiving.

$$E_{tx} = \begin{cases} L * E_{elec} + E_{fs} * L * d^2 & , \text{if } d < = d_0 \\ L * E_{elec} + E_{fs} * L * d^4 & , \text{if } d > = d_0 \end{cases}$$

Here
E_{fs} , E_{mp} itis the amount of energy per bit disspated in RF amplifier according to the distance .

Distance is determined from equation :

$$d_0 = \sqrt{E_{fs}}\, \sqrt{E_{mp}}$$

Energy utilized for receiving a packet with L bits is calculte from the following equation :

$$E_{rx} = L * E_{elec}$$

IV. ENCRYPTION PROCESS

" This paper proposes an encryption process which contains four rounds. In each round different key is applied to different parts of the plain text. The encryption process contains many operations likes Addition, XOR, Transpose, operation. The block size of the text enter by user for the encryption is 49 bits. The size of the dynamic key which is used to convert the plaintext into ciphertext is 196 bits.[4]

Initially the text enters for encryption is divided into 49 bits of blocks. Dynamic key size which is used for encryption and decryption is 196 bits which is the 14X14 matrix. The dynamic key for first round is divided into four parts: DP1, DP2, DP3, and DP4. [4]

DP1 the first part of the key is applied to the first part of the text which is entered by user to perform the encryption operation and at the end of the round 1 second part of the key DP2 is applied over the last part of the plaintext. The operations which are performed over the plaintext is ADD, XOR and TRANSPOSE. The output produces by round 1 is called round1 ciphertext (RP1).

In the next round of encryption process dynamic key is divided into two parts: DKF and DKL. DKF contains the first 64 bits of the key and DKL contains the last 147 bits of the key. Futher DKL is divided into two parts DKL1 and DKL2. Further DKF is divided into 3 parts these are DKF1, DKF2, DKF3.

In the second round only one key is used to perform the encryption and decryption operation. The output of the second round is known as round 2 ciphertext (RP2).

In the next round or we can say that in third round two keys are used one is DKL2 and another is DKF1 is applied over the output of the round 2.This round contain the same operation like Addition, subtraction , XOR, Transpose.

In last or we can say that in fourth round also contains the two keys one is DKF2, and another one is DKF3 which are applied over the output of the second round. The output produces by last round is known as round 4 ciphertext. “

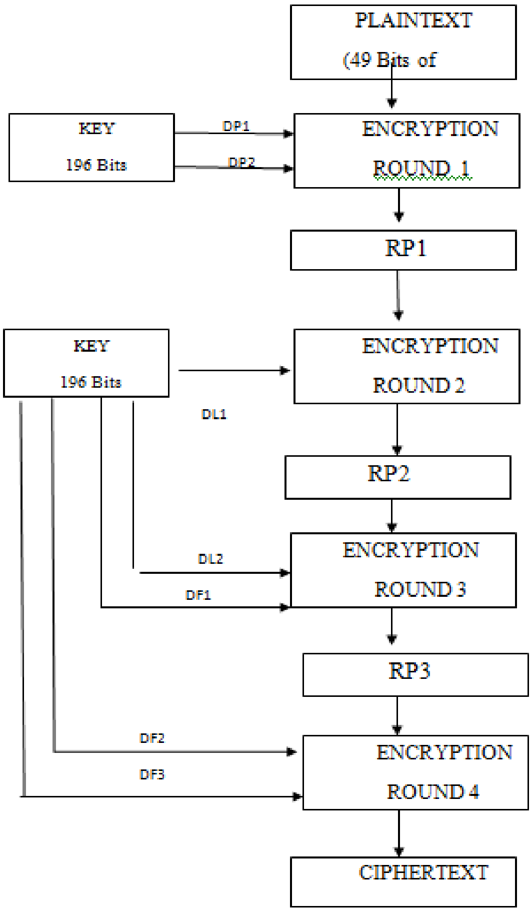

Figure. 3 Encryption process for lightweight protocol

V. Decryption Process

“ The decryption method is inverse of the encryption method. This process converts the unreadable message into the original message. As the encryption process decryption method contains the four rounds to perform the decryption method. Decryption method contains the same operations which are used in the encryption process like subtractions, XOR, Transpose. In the first round of the decipherment process operations are applied over the ciphertext which produce the first round ciphertext which is denoted by RCT1.

In the next round of the decipherment process operations are applied over the ciphertext which produce the second round ciphertext which is denoted by RCT2. .

In the third round of the decipherment process operations are applied over the ciphertext which produce the second round ciphertext which is denoted by RCT3. .

In the last round of the decipherment process or we can say that at fourth round, operations are applied over the ciphertext which produce the fourth round output which is the plaintext or original message “ .[4]

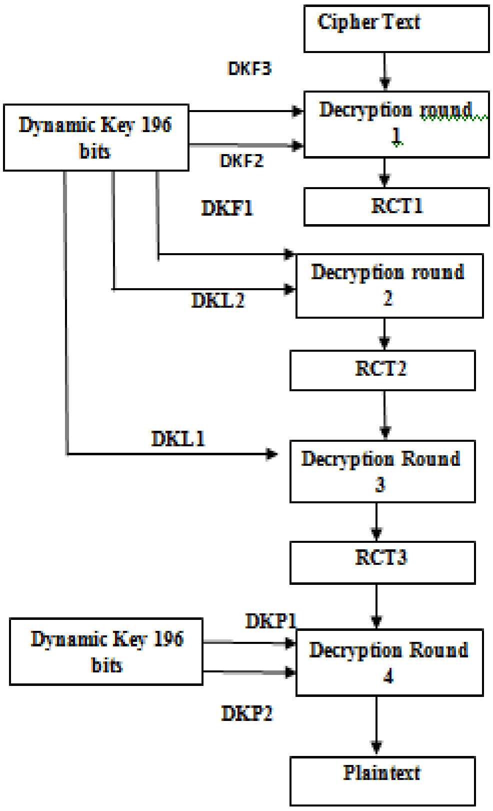

Figure. 4 Decryption process for lightweight protocol

VI. COMPARE THE PERFORMANCE AND TIME COMPLEXITY

This section describes the key size, block size of different algorithm which is used to perform encryption and decryption. It also describe the time taken by various algorithm to perform encryption and decryption. In this paper lightweight protocol using LCG and lightweight protocol using park miller both implemented in MATLAB 7.0 and specification of computer in which algorithm is run is Intel Core 2 Duo CPU, 2.96 GB of RAM. When we run protocols on this system than time taken by both the lightweight protocols have time difference so we can say that LiSP using park-miller is more efficient than LiSP using LCG.

DOI 10.17148/IJARCCE.2015.4295

TABLE 1
KEY AND BLOCKSIZE OF ALGORITHIM

Algorithm	Block Size(Bits)	Key Size (Bits)
DES	64	64
LiSP using LCG	196	49
Proposed Algorithm	196	49

TABLE 2
TIME COMPLEXITY OF DIFFERENT ALGORITHIM

Algorithm	Kilobytes processed	Time
LiSP using LCG	12	24.094
LiSP using Park-Miller algorithm	12	20.250
Lisp using Blum Blum Shub	12	36.953
Lisp using RC4	12	22.641
Lisp using Wichman - Hill	12	25.844

VII. CONCLUSION

In this paper lightweight protocol is created using dynamic key for wireless sensor network. In this dynamic key is generated by using different random number generator algorithm. The main strong point of this method is dynamic key which generate every time after encryption and decryption process. Par miller algorithm provides the best authentication mechanism. In the proposed work find out time to execute the algorithm with using different random number genarator algorithm and which algorithm is energy eficient. The conclusion of the implementation is that which algorithm execute in less time consume less energy and which take large time to ececute has high energy consumption. Energy consumption is big issue because sensor network has limited energy resource so which algorithm has less energy consumption and efficiently provide security is energy efficient. So proposed method is more secure than existing. This type of lightweight protocol can meet our security requirement.

REFERENCES

[1] William Stalling, “Applied Cryptography” 4th ed.

[2] Bharatesh N, Rohith S, “FPGA Implementation of Park-Miller Algorithm to Generate Sequence of 32-Bit Pseudo Random Key for Encryption and Decryption of plain text”, International Journal of Reconfigurable and Embedded Systems (IJRES), Vol. 2, No. 3, November 2013, pp. 99~105.

[3] Bo Sun, Chung-Chih Li, Kui Wu, Yang Xiao, “A lightweight secure protocol for wireless sensor networks”, ELSEVIER, computer communication, 29 (2006) 2556–2568.

[4] Zeenat mahmood, Anurag jain, Chetan agrawal,” Hybridize Dynamic Symmetric Key Cryptography using LCG”, In International Journal of Computer Applications (0975 – 8887) , vol. 60-no. 17 ,December 2012 .

[5] SK Park, KW Miller, Random number generators: Good ones are hard to find. Communications of the, *ACM*. 32(10): 1192-1201.

[6] Che-Chens Lin, Shiuhyng Shieh, Jia-Chun Lin, “Distributed Key Agreement Protocol for Wireless Sensor Network”, The Second International Conference on Secure System Integration and Reliability Improvement.

[7] Suman Bala, Gaurav Sharma and Anil K. Verma, “Classification of Symmetric Key Management Schemes for Wireless Sensor Network”, International Journal of Security and Its Applications Vol. 7, No. 2, March, 2013.

[8] Priya L C, Shantal Devi Patil, “ A survey on Sensor Aunthication in Dynamic Wireless Sensor Networks ”, International Journal of Computer Science and Information Technology Research,Vol. 2, Issue 2, pp: (454-461), Month: April-June 2014.

[9] Delan Alsoufi, Khaled Elleithy, Tariq Abuzaghleh and Ahmad Nassar “ Security in wireless Sensor Network”, International journal of computer Science and Engineering Survey, vol 3, no. 3, June 2012.

[10] Anupma Sangwan1, Deepti Sindhu2, Kulbir Singh, ” A Review of various security protocols in wireless sensor networks”, Anupma Sangwan et al, int. j. comp. tech. Appl., Vol 2 (4), 790-797.

[11] Ankita ojha, Kusumlata jain “ A Survey on Lightweight Security protocol using Dynamic Key for Wireless Sensor Network” , International journal of advance research in computer science and communication engineering, vol. 3, Issue 9, September 2014.

[12] Ankita ojha, Kusumlata jain “ Implementation of LiSP using Park – Miller for Wireless Sensor Network” , International journal of Computer Applications, vol. 110, No. 8, January 2015.

DOI 10.17148/IJARCCE.2015.4295