International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

SECURE DATA IN WIRELESS SENSOR NETWORK VIA DES

Vimal Upadhyay, Pintu Kashyap, Inder Kumar, Jai Balwan , Lalit Choudhary

ABSTRACT

One of the main goals of sensor networks is to provide accurate information about a sensing field for an extended period of time. The emergence of sensor networks as one of the dominant technology trends in the coming decades has posed numerous unique challenges to researchers. Because sensor networks may interact with sensitive data and/or operate in hostile unattended environments, it is imperative that these security concerns be addressed from the beginning of the system design. These networks are likely to be composed of hundreds, and potentially thousands of tiny sensor nodes, functioning autonomously, and in many cases. While the set of challenges in sensor networks are diverse, we focus on security of Wireless Sensor Network in this paper[1,2,3,4,5]. We propose some of the security goal for Wireless Sensor Network. Further, security being vital to the acceptance and use of sensor networks for many applications; we have made in depth threat analysis of Wireless Sensor Network. We also propose some countermeasures against these threats in Wireless Sensor Network. So, in this paper we have implemented Encryption Algorithm like - DES to provide sufficient levels of security for protecting the confidentiality of the data in the WSN network. This paper also analyzes the performance of DES algorithm against Attacks in WSN Network[3,5].

KEYWORDS**:** WSN , Sensor node , Gateway , Security , DES.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

1. INTRODUCTION:

Wireless sensor networks are quickly gaining popularity due to the fact that they are potentially low cost solutions to a variety of real-world challenges. Their low cost provides a means to deploy large sensor arrays in a variety of conditions capable of performing both military and civilian tasks. But sensor networks also introduce severe resource constraints due to their lack of data storage and power. Both of these represent major obstacles to the implementation of traditional computer security techniques in a wireless sensor network[8,9,11]. The unreliable communication channel and unattended operation make the security defenses even harder. Indeed, as pointed out in wireless sensors often have the processing characteristics of machines that are decades old (or longer), and the industrial trend is to reduce the cost of wireless sensors while maintaining similar computing power. With that in mind, many researchers have begun to address the challenges of maximizing the processing capabilities and energy reserves of wireless sensor nodes while also securing them against attackers. All aspects of the wireless sensor network are being examined including secure and efficient routing, data aggregation, group formation, and so on. In addition to those traditional security issues, we observe that many general-purpose sensor network techniques (particularly the early research) assumed that all nodes are cooperative and trustworthy. This is not the case for most, or much of, real-world wireless sensor networking applications, which require a certain amount of trust in the application in order to maintain proper network functionality. Researchers therefore began focusing on building a sensor trust model to solve the problems beyond the capability of cryptographic security. In addition, there are many attacks designed to exploit the unreliable communication channels and unattended operation of wireless sensor networks. Furthermore, due to the inherent unattended feature of wireless sensor networks, we argue that physical attacks to sensors play an important role in the operation of wireless sensor networks. Thus, we include a detailed discussion of the physical attacks and their corresponding

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

defenses, topics typically ignored in most of the current research on sensor security. We classify the main aspects of wireless sensor network security into four major categories: the obstacles to sensor network security, the requirements of a secure wireless sensor network, attacks, and defensive measures. We also give a brief introduction of related security techniques and summarize the obstacles for the sensor network security[13,14,15]. The security requirements of a wireless sensor network are listed as below:

1.1. Obstacles of Sensor Security

A wireless sensor network is a special network which has many constraints compared to a traditional computer network. Due to these constraints it is difficult to directly employ the existing security approaches to the area of wireless sensor networks. Therefore, to develop useful security mechanisms while borrowing the ideas from the current security techniques like (DES, AES).

2. WSN ARCHITECTURE

In a typical WSN we see following network components –

[A]. Sensor motes (Field devices) – Field devices are mounted in the process and must be capable of routing packets on behalf of other devices. In most cases they characterize or control the process or process equipment. A router is a special type of field device that does not have process sensor or control equipment and as such does not interface with the process itself.

[B]. Gateway or Access points – A Gateway enables communication between Host application and field devices.

[C].Network manager – A Network Manager is responsible for configuration of the network, scheduling communication between devices (i.e., configuring super frames), management of the routing tables and monitoring and reporting the health of the network.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

[D].Security manager – The Security Manager is responsible for the generation, storage, and management of keys[5,18,19].

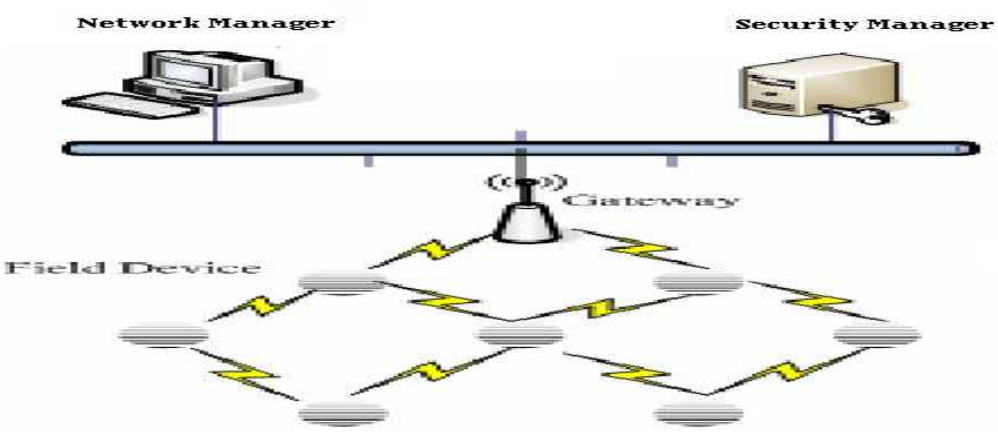


Figure 1 WSN Architecture

3. WSN SECURITY ANALYSIS

Simplicity in Wireless Sensor Network with resource constrained nodes makes them extremely vulnerable to variety of attacks. Attackers can eavesdrop on our radio transmissions, inject bits in the channel, replay previously heard packets and many more. Securing the Wireless Sensor Network needs to make the network support all security properties: confidentiality, integrity, authenticity and availability. Attackers may deploy a few malicious nodes with similar hardware capabilities as the legitimate nodes that might collude to attack the system cooperatively. The attacker may come upon these malicious nodes by purchasing them separately, or by "turning" a few legitimate nodes by capturing them and physically overwriting their memory. Also, in some

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

cases colluding nodes might have high-quality communications links available for coordinating their attack. Sensor nodes may not be tamper resistant and if an adversary compromises a node, she can extract all key material, data, and code stored on that node. While tamper resistance might be a viable defense for physical node compromise for some networks, we do not see it as a general purpose solution. Extremely effective tamper resistance tends to add significant per-unit cost, and sensor nodes are intended to be very inexpensive

3.1Overview

The Data Encryption Standard (DES) is the name of the Federal Information Processing Standard (FIPS) 46-3, which describes the data encryption algorithm (DEA). The DEA is also defined in the ANSI standard X3.92. DEA is an improvement of the algorithm Lucifer developed by IBM in the early 1970s. IBM, the National Security Agency (NSA now National Institute of Standards and technology NIST) developed the algorithm. The DES has been extensively studied since its publication and is the most widely used symmetric algorithm in the world. [15,5,19]The DES has a 64-bit block size and uses a 56-bit key during execution (8 parity bits are stripped off from the full 64- bit key). DES is a symmetric cryptosystem, specifically a 16-round Feistel cipher. When used for communication, both sender and receiver must know the same secret key, which can be used to encrypt and decrypt the message, or to generate and verify a Message Authentication Code (MAC). The DES can also be used for single-user encryption, such as to store files on a hard disk in encrypted form.

3.2 In Depth

DES encrypts and decrypts data in 64-bit blocks, using a 64-bit key (although the effective key strength is only 56 bits, as explained below). It takes a 64-bit block of plaintext as input and outputs a 64-bit block of cipher text. Since it always operates on blocks of equal size and it uses both permutations and substitutions in the algorithm, DES is both a block cipher and a product cipher.DES has 64-bit rounds, meaning the main algorithm is repeated 16 times to produce the

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

cipher text. It has been found that the number of rounds is exponentially proportional to the amount of time required to find a key using a brute-force attack. So as the number of rounds increases, the security of the algorithm increases exponentially[3,4,5].

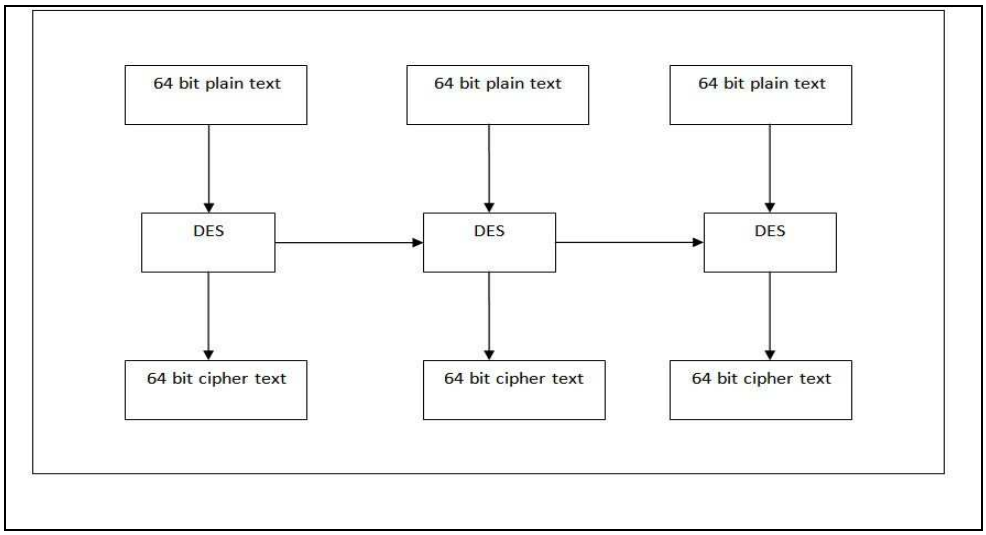


Figure2. Conceptual working of DES

3.3 Key Scheduling

Although the input key for DES is 64-bits long, the actual key used by DES is only 56-bits in length. The least significant (right-most) bit in each byte is a parity bit, and should be set so that there are always an odd number of 1s in every byte. These parity bits are ignored, so only the seven most significant bits of each byte are used; resulting in a key length of 56-bits.The first step is to pass the 64-bit key through a permutation called Permuted Choice 1, or PC-1 for short. The table for this is given below. Note that in all subsequent descriptions of bit numbers, 1 is the left-most bit in the number, and n is the rightmost bit.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

PC -1: Permuted Choice 1							
Bit	0	1	2	3	4	5	6
1	57	49	41	33	25	17	9
8	1	58	50	42	34	26	18
15	10	2	59	51	43	35	27
22	19	11	3	60	52	44	36
29	63	55	47	39	31	23	15
36	7	62	54	46	38	30	22
43	14	6	61	53	45	37	29
50	**21**	**13**	**5**	**28**	**20**	**12**	**4**

Table 1: Permuted Choice 1

For example, we can use the PC-1 table to figure out how bit 30 of the original 64-bit key transforms to a bit in the new 56-bit key. Find the number 30 in the table, and notice that it belongs to the column label 5 and the row labeled 36. Add up the value of the row and column to find the new position of the bit within the key. For bit 30, 36+5=41, so bit 30 becomes bit 41 of the new 56-bit key. Note that bit 8,16,24,32,40,48,56 and 64 of the original key are not in the table. These are the unused parity bits that are discarded when the final 56-bit key is created [17,19,5].

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64

Figure3. Discarding of every 8th bit of Original Key (Shaded Bit Position are Discarded)

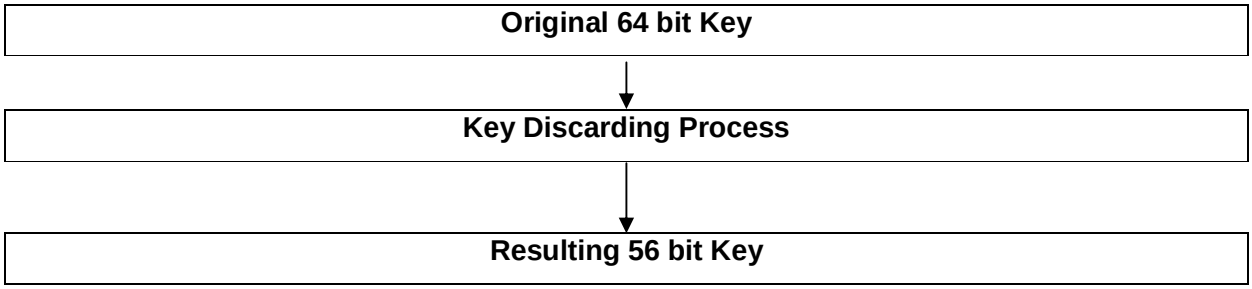


Figure 4. Key Discarding Process

Now that we have the 56-bit key, the next step is to use this key to generate 16 48-bit sub keys, called K [1] – K [16], which is used in the 16 rounds of DES for encryption and decryption. The procedure for generating the sub keys – known as key scheduling – is fairly simple:

1. Set the round number R to 1.
2. Split the current 56-bit key, K, up into two 28-bit blocks, L (the left-hand half) and R (the right-hand half).
3. Rotate L left by the number of bits specified in the table below, and rotate R left by the same number of bits as well.
4. Join L and R together to get the new K.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

5. Apply Permuted Choice 2 (PC-2) to K to get the final K[R], where R is the round number we are on.

6. Increment R by 1 and repeat the procedure until we have all 16 sub keys K [1] – K [16].

Here are the tables involved in these operations:

Sub Key Rotation Table:

Round Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Number of Bits to Rotate	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Table 2: Sub Key Rotation Table

P C 2 : Permuted Choice 2						
Bit	**0**	**1**	**2**	**3**	**4**	**5**
1	14	17	11	24	1	5
7	3	28	15	6	21	10
13	23	19	12	4	26	8
19	16	7	27	20	13	2
25	41	52	31	37	47	55
31	30	40	51	45	33	48
37	44	49	39	56	34	53
43	46	42	50	36	29	32

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

Table 3: P C 2 : Permuted Choice 2

Plaintext Preparation

Once the key scheduling has been performed, the next step is to prepare the plaintext for the actual encryption. This is done by passing the plaintext through a permutation called the Initial Permutation, or IP for short. This table also has an inverse, called the Inverse Initial Permutation, or IP^ (-1) is also called the Final Permutation. Both of these are shown below.

IP: Initial Permutation								
Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	58	50	42	34	26	18	10	2
9	60	52	44	36	28	20	12	4
17	62	54	46	38	30	22	14	6
25	64	56	48	40	32	24	16	8
33	57	49	41	33	25	17	9	1
41	59	51	43	35	27	19	11	3
49	61	53	45	37	29	21	13	5
57	**63**	**55**	**47**	**39**	**31**	**23**	**15**	**7**

Table 4: IP(Initial Permutation)

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

IP^(-1):Inverse Initial Permutation								
Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	40	8	48	16	56	24	64	32
9	39	7	47	15	55	23	63	31
17	38	6	46	14	54	22	62	30
25	37	5	45	13	53	21	61	29
33	36	4	44	12	52	20	60	28
41	35	3	43	11	51	19	59	27
49	34	2	42	10	50	18	58	26
57	**33**	**1**	**41**	**9**	**49**	**17**	**57**	**25**

Table 5: IP^(-1) Inverse Initial Permutation

These tables are used just like PC-1 and PC-2 were for the key scheduling. By looking at the tables is becomes apparent why one permutation is called the inverse of the order. For example, let's examine how bits 32 are transformed under IP. In the table, bit 32 is located at the intersection of the column labeled 4 and the row labeled 25. So this bit becomes 29(25+4) of the 64-bit block after the permutation. Now let's apply IP^ (-1). In IP^ (-1), bit 29 is located at the intersection of the column labeled 7 and the row labeled 25. So this bit becomes bit 32(25+7) after the permutation. And this is the bit position that we started with before the first permutation.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

So IP^ (-1) really is the inverse of IP. It does the exact opposite of IP. We will end up with the original block.

DES Core Function

Once the key scheduling and plaintext preparation have been completed, the actual encryption or decryption is performed by the main DES algorithm. The 64-bit block of input data is first split into two halves, L and R. L is the left-most 32 bits, and R is the right-most 32 bits.

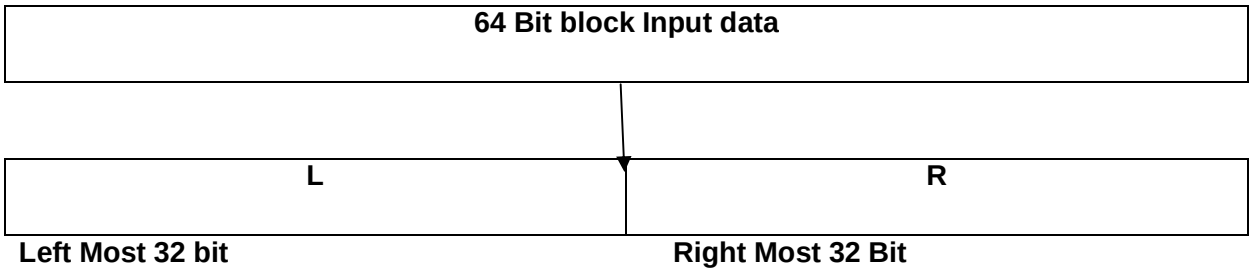
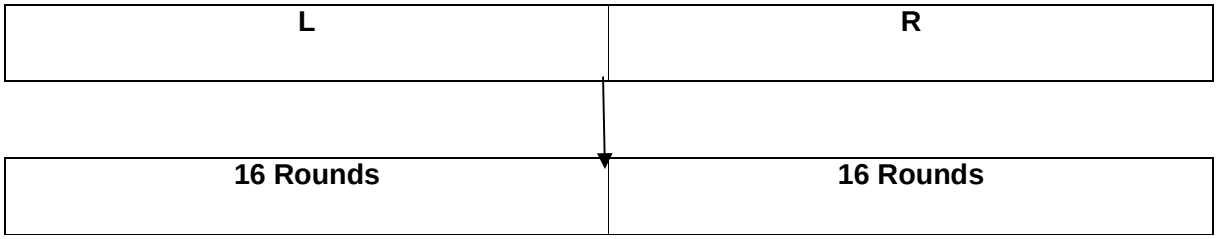


Figure 5. Splitting 64-bit block input data

The following is repeated 16 times, making up the 16 rounds of standard. We call the 16 sets of halves L[0]- L[15] and R[0]-R[15].



International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

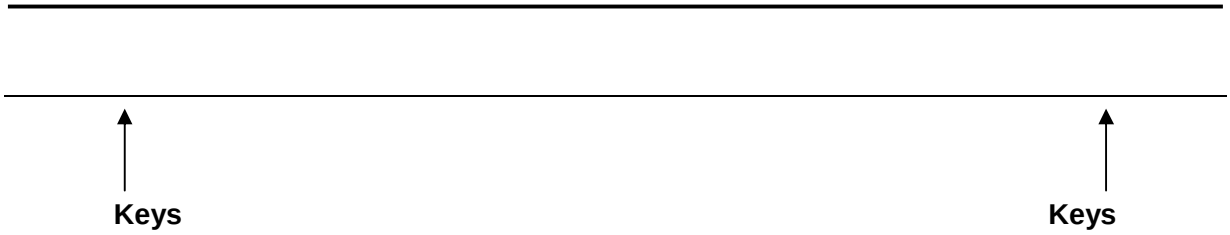


Figure 6. 16 Rounds

The following process is repeated 16 times, making up the 16 rounds of standard DES. We call the 16 sets of halves L [0] – L [15] and R [0] – R [15].

1. R[I-1] – where I is the round number, starting at 1 – is taken and fed into the E-Bit Selection Table, which is like a permutation, except that some of the bits are used more than once. This expands the number R [I-1] from 32 to 48 bits to prepare for the next step.

E-Bit Selection Table						
Bit	0	1	2	3	4	5
1	32	1	2	3	4	5
7	4	5	6	7	8	9
13	8	9	10	11	12	13
19	12	13	14	15	16	17
25	16	17	18	19	20	21
31	20	21	22	23	24	25
37	24	25	26	27	28	29
43	28	29	30	31	32	1

Table 6: E-Bit Selection Table

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

2. The 48-bit R [I-1] is XORed with K [I] and stored in a temporary buffer so that R [I-1] is not modified.

3. The Result from the previous step is now split into 8 segments of 6 bits each. The left-most 6 bits are B [1], and the right-most 6 bits are B [8]. These blocks from the index into the S-boxes, which are used in the next step? The Substitution boxes, known as S-boxes, are a set of 8 two-dimensional arrays, each with 4 rows and 16 columns. The numbers in the boxes are always 4 bits in length, so their values range from 0-15. The S-boxes are numbered S [1] – S [8].

4. Starting with B [1], the first and last bits of the 6-bit block are taken and used as an index into the row number of S [1], which can range from 0 to 3, and the middle four bits are used as an index into the column number, which can range from 0-15. The number from this position in the S-box is retrieved and stored away. This is repeated with B [2] and S [8]. At the point, you now have 8 4-bit numbers, which when strung together one after the other in the order of the retrieval, give a 32-bit result.

5. The result from the previous stage is now passed into the Permutation.

6. This number is now XORed with L [I-1], and moved into R [I]. R [I-1] is moved into L [I].

7. At this point we have a new L [T] and R [I]. Here, we increment I and repeat the core function until I = 17, which means that 16 rounds have been executed and keys K[I] – K[16] have all been used.

8. When L [16] and R [16] have been obtained, they are joined back together in the same fashion they were split apart (L [16] is the left-hand half, R [16] is the right-half hand), then the two halves are swapped, R [16] becomes the left-most 32 bits and L [16] becomes the right-most 32 bits of the pre-output block and the resultant 64- bit number is called the pre-output.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

P- Permutation				
Bit	0	1	2	3
1	16	7	20	21
5	29	12	28	17
9	1	15	23	26
13	5	18	31	10
17	2	8	24	14
21	32	27	3	9
25	19	13	30	6
29	22	11	4	25

S-Box 1:Substitution Box 1																
Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	**15**	**12**	**8**	**2**	**4**	**9**	**1**	**7**	**5**	**11**	**3**	**14**	**10**	**0**	**6**	**13**

S-Box 2:Substitution Box 2

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
3	**13**	**8**	**10**	**1**	**3**	**15**	**4**	**2**	**11**	**6**	**7**	**12**	**0**	**5**	**1**	**4**

S-Box 3:Substitution Box 3

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
3	**1**	**10**	**13**	**0**	**6**	**9**	**8**	**7**	**4**	**15**	**14**	**3**	**11**	**15**	**2**	**12**

S-Box 4:Substitution Box 4

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	**3**	**15**	**0**	**6**	**10**	**1**	**13**	**8**	**9**	**4**	**5**	**11**	**12**	**7**	**2**	**14**

S-Box 5:Substitution Box 5

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
3	**11**	**8**	**12**	**7**	**1**	**14**	**2**	**13**	**6**	**15**	**0**	**9**	**10**	**4**	**5**	**3**

S-Box 6:Substitution Box 6

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	12	1	10	15	9	2	6	8	0	3	13	4	14	7	5	11
1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

3	**4**	**3**	**2**	**12**	**9**	**5**	**15**	**10**	**11**	**14**	**1**	**7**	**6**	**0**	**8**	**13**

S-Box 7:Substitution Box 7

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
3	**6**	**11**	**13**	**8**	**1**	**4**	**10**	**7**	**9**	**5**	**0**	**15**	**14**	**2**	**3**	**12**

S-Box 8:Substitution Box 8

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
3	**2**	**1**	**14**	**7**	**4**	**10**	**8**	**13**	**15**	**12**	**9**	**0**	**3**	**5**	**6**	**11**

Table 7: S-Boxes

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

How to use the S-boxes

The purpose of this example is to clarify how the S-boxes work. Suppose we have the following 48-bit binary number:

011101000101110101000111101000011100101101011101

In order to pass this through steps 3 and 4 of the Core Function as outlined above, the number is split up into 8 6-bit blocks, labeled B[1] to B[8] from left to right:

011101 000101 110101 000111 101000 011100 101101 011101

Now, eight numbers are extracted from the S- boxes – one from each box:

B[1] = S[1](01,1110) = S[1][1][14] = 3 = 0011

B[2] = S[2](01,0010) = S[2][1][2] = 4 = 0100

B[3] = S[3](11,1010) = S[3][3][10] = 14 = 1110

B[4] = S[4](01,0011) = S[4][1][3] = 5 =0101

B[5] = S[5](10,0100) = S[5][2][4] = 10 = 1010

B[6] = S[6](00,1110) = S[6][0][14] = 5 = 0101

B[7] = S[7](11,0110) = S[7][3][6] = 10 = 1010

B[8] = S[8](01,1110) = S[8][1][14] = 9 = 1001

In each case of S[n][row][column] , the first and last bits of the current B[n] are used as the row index , and the middle four bits as the column index.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

The results are now joined together to form a 32 – bit number which serves as the input to stage 5 of the Core Function (the P Permutation):

00110100111001011010010110101001

Conclusion:

Data encryption is utilized in various applications and environments. The specific utilization of encryption and the implementation of the DES will be based on many factors particularly to the WSN and its associated components like sensor node , Gateway ,Routing in WSN etc. Cryptography is used to protect data while it is being communicating between two points or while it stored in a medium vulnerable to physical theft.

References:

[1]. International Journal of Technology and Applied Science, Vol. 2, pp. 5-11, 2011. ISSN: 2230-9004 © 2011 IJTAS 5 Repairing the Gaps in Connectivity of Wireless Sensor Network & WiMAX Using Robots: Jagbir Dhillon, Krishna Parsad, Rajesh Kumar, R.S Sikarwar, Vimal Upadhyay.

[2]. Secure and Efficient Broadcast Authentication in Wireless Sensor Networks Taekyoung Kwon, Member, IEEE, and Jin Hong.

[3]. Distributed Recovery from Network Partitioning in Movable Sensor/Actor Networks via Controlled Mobility, Kemal Akkaya, Member, IEEE, Fatih Senel, Aravind Thimmapuram, and Suleyman Uludag, Member, IEEE, IEEE transactions on computers, vol. 59, no. 2, february 2010.

[4]. IEEE journal on selected areas in communications, vol. 28, no. 5, june 2010 Cross Layer QoS-Aware Communication for Ultra Wide Band Wireless Multimedia Sensor Networks Tommaso Melodia, Member, IEEE, and Ian F. Akyildiz, Fellow, IEEE.

[5]. IEEE journal on selected areas in communications, vol. 28, no. 7, september 2010. Handling Inelastic Traffic in Wireless Sensor Networks Jiong Jin, Student Member, IEEE, Avinash Sridharan, Bhaskar Krishnamachari, Member, IEEE and Marimuthu Palaniswami, Senior Member, IEEE.

[6]. Constrained Relay Node Placement in Wireless Sensor Networks: Formulation and Approximations, Satyajayant Misra, Member, IEEE, Seung Don Hong, Guoliang (Larry) Xue, Senior Member, IEEE, and Jian Tang, Member, IEEE; IEEE/ACM transactions on networking, vol. 18, no. 2, April 2010.

[7]. Deploying Sensor Networks With Guaranteed Fault Tolerance; Jonathan L. Bredin, Erik D. Demaine, Mohammad Taghi Hajiaghayi, and Daniela Rus; IEEE/ACM transactions on networking, vol. 18, no. 1, february 2010.

[8]. A Distributed Node Localization Scheme for Wireless Sensor Networks; Qinqin Shi, Hong Huo, Tao Fang, Deren Li; Published online: 26 March 2009 © Springer Science+Business Media, LLC. 2009.

[9]. High Reliable In-Network Data Verification in Wireless Sensor Networks; Dong-Wook Lee, Jai-Hoon Kim; Published online: 29 May 2009 © Springer Science+Business Media, LLC. 2009.

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

[10]. Secure and Efficient Localization Scheme in Ultra-Wideband Sensor Networks; Daojing He, Lin Cui, Hejiao Huang, Maode Ma; Published online: 4 November 2008 © Springer Science+Business Media, LLC. 2008.

[11]. TMH Book of Cryptography & System Security.

[12]. Phalguni Gupta Internet & Protection Security Book.

[13] Adrian Perrig, John Stankovic, David Wagner, “Security in Wireless Sensor Networks” Communications of the ACM, Page53-57, year 2004

[14] Al-Sakib Khan Pathan, Hyung-Woo Lee, Choong Seon Hong, “Security in Wireless Sensor Networks: Issues and Challenges”, International conference on Advanced Computing Technologies, Page1043-1045, year 2006

[15] A. S. Wander, N. Gura, H. Eberle, V. Gupta, and S. C. Shantz, "Energy analysis of public-key cryptography

for wireless sensor networks," in Third IEEE International Conference on Pervasive Computing and

Communications (PERCOM'05). IEEE Computer Society Press, 2005, pp. 324-328.

[16] D. C. Schleher, Electronic Warfare in the Information Age. Artech.

[17] D. Djenouri, L. Khelladi, and N. Badache, “A Survey of Security Issues in Mobile ad hoc and Sensor

Networks," IEEE Commun. Surveys Tutorials, vol. 7, pp. 2–28, year 2005.

[18] D.Ganesan, R.Govindan, S.Shenker, and D.Estrin, "Highly resilient, energy efficient multipath routing in

wireless sensor networks," Mobile Computing and Communications Review (MC2R), vol. 1, no. 2, 2002.

[19] F. Nait-Abdesselam, B. Bensaou, T. Taleb, "Detecting and avoiding wormhole attacks in wireless Ad hoc

networks," IEEE Communication Magazine, Vol.46, Issue 4, pp. 127-133, April 2008.

[20] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: challenges and solutions," IEEE Wireless Communications, vol. 11, no. 1, pp. 38- 47, Feb. 2004

[21] Ian F. Akykildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci, "A Survey on Sensor Networks", IEEE Communication Magazine, year. 2002.

[22] John Paul Walters, Zhengqiang Liang, Weisong Shi, Vipin Chaudhary, "Wireless Sensor Network Security:

A Survey", Security in Distributed, Grid and Pervasive Computing Yang Xiao (Eds), Page3-5, 10-15, year 2006.

[23]Mohit Saxena, "Security In Wireless Sensor Networks - A Layer Based Classification", Cerias Tech Report

2007-04.

[24]S. Khan, K-k. Loo, T. Naeem, M.A. Khan, “Denial of service attacks and challenges in broadband wireless

network,” International Journal of Computer Science and Network Security, Vol. 8, No. 7, pp.1-6, July 2008.

[25]Wang, B-T. and Schulzrinne, H., “An IP trace back mechanism for reflective DoS attacks”, Canadian M. Conference on Electrical and Computer Engineering, Volume 2, 2-5 May 2004, pp. 901 – 904. ISSN : 0975-

3397 1835

[26]Y.Wang, G.Attebury, and B. Ramamurthy, “A Survey of Security Issues in Wireless Sensor Networks,” IEEE Commun. Surveys Tutorials, vol. 8, pp. 2–23, year 2006.

[27]Y.Mun and C. Shin, “Secure routing in sensor networks: Security problem analysis and countermeasures,” in International Conference on Computational Science and Its Applications - ICCSA 2005, May 9- 12 2005, vol.

3480 of Lecture Notes in Computer Science, (Singapore), pp. 459–467, Springer Verlag, Heidelberg, D-69121, Germany, 2005.

[28]Thomas Haenselmann (2006-04-05). Sensornetworks. GFDL Wireless Sensor Network textbook

International Journal of Enterprise Computing and Business Systems

ISSN (Online) : 2230-8849

http://www.ijecbs.com

Vol. 1 Issue 2 July 2011

Author's Profile

Vimal Upadhyay is working as Assistant Professor, in Deptt. of Computer Science Engineering, St. Margaret Engineering College, Neemrana, NH-8, Delhi-Jaipur. He has done M.Tech. in CS From M.D.U, Rohtak. He has a number of publications in some journal of well repute. His area of interest is security in WSN/WiMax.