International Journal of Grid Distribution Computing
Vol.7, No.5 (2014), pp.43-52
http://dx.doi.org/10.14257/ijgdc.2014.7.5.04

Evaluation of Encryption Method of SNEP for Implementing Security in Wireless Sensor Network using SPINS Framework

Vartika Shah and Sanjiv Sharma

Department Of Computer ScienceEngineering & Information Technology
Madhav Institute of Technology and Science, Gwalior (M.P.)
vartika.shah@gmail.com, dr.s.sanjiv@gmail.com

Abstract

Wireless sensor network (WSN) is a collection of densely deployed sensor nodes. These nodes are prone to attack and also resource constrained. Resource accountability is also required for Security implementation in WSN. Some researchers proposed different methods, algorithms and frameworks for WSN security implementation. SPINS is a framework for implementing overall security in WSN using SNEP (Sensor Network Encryption Protocol) and µTESLA (the "micro" version of the Timed, Efficient, Streaming, Loss-tolerant Authentication Protocol) protocol. Existing research works reveal that SNEP employed RC5 encryption algorithm for WSN. This paper analyzed efficient encryption algorithm XTEA for SNEP & evaluate that it is better as compare to RC5 in terms of energy, storage and time.

Keywords- *Wireless sensor network, security, SPINS, SNEP, block cipher, RC5, XTEA*

1. Introduction

Recently wireless communication has been accelerated the deployment of large-scale wireless sensor networks. WSNs are changing the way people observe and interact with their environment. The development of Wireless Sensor Networks was motivated by military applications such as battlefield surveillance. Today such networks are used in many industrial and consumer applications. Wireless sensor nodes are used to collect the information from different types of applications such as tracking critical facilities, monitoring animal habitats, and surveillance. These are spatially distributed resource constrained sensing devices having small batteries and are usually capable of measuring physical phenomenon like temperature, sound, vibration, pressure etc. A Wireless Sensor Network (WSN) can be defined as a group of spatially distributed autonomous sensor nodes, communicating wirelessly over limited frequency and bandwidth. Sensor nodes are limited in computation, memory, power resources, and bandwidth. Due to these unique properties and different behavior of WSNs, achievement of security services is a challenging task. Only designing a framework for implementing security is not sufficient in WSN, it should also provide efficient utilization of resources.

Perrig, *et al.*, [1] has proposed a set of security framework, SPINS, for WSN. SPINS is a combination of SNEP (Sensor Network Encryption Protocol) and µTESLA (the "micro" version of the Timed, Efficient, Streaming, Loss-tolerant Authentication Protocol). SNEP provides data confidentiality, data freshness and two-party authentication where µTESLA provides authenticated broadcast for severely resource-constrained environments. SPINS is a model that propose feasible ideas for both base station to node and node to node communication. So far, SPINS framework is one of the best defined and known WSN security model. The main downside of SPINS is that it uses RC5 algorithm for encryption

ISSN: 2005-4262 IJGDC

that increases the energy cost of SNEP. Existing work shows that RC5 encryption algorithm is used in SNEP. In this paper, it is shown that XTEA is better than RC5 using some parameters.

This paper propose new type of secure routing protocol that is suitable for sensor networks in reference to energy efficiency, storage, speed without compromising with security. In section 2, Existing Encryption Method for SNEP Method in SPINS Framework is presented. Next in section 3, the proposed routing protocol is presented. Next in section 4, the detailed analysis of proposed protocol with comparison to the old one is shown. Finally, section 5 offers conclusion and issues for the future work.

2. Existing Encryption Method for SNEP in SPINS Framework

SNEP provides many advantages. First, it adds only 8 bytes per message which creates low communication overhead. Second, it uses counters like many other cryptographic protocols but avoids transmitting it by keeping state at both end points. Third, it achieves semantic security which prevents eavesdroppers from inferring the message content from the encrypted message. Finally, gives replay protection, data authentication and weak message freshness. For achieving data confidentiality with semantic security, it uses randomization before encryption. To achieve randomization it uses DES-CBC. For encryption, RC5 encryption algorithm is used. RC5 [2] was designed by Professor Ronald Rivest of MIT and first published in December 1994. It has variable key size (0 to 2040 bits), block size (32, 64 or 128-bits) and no of rounds (0 to 255). Due to trade-off between security and efficiency, RC5-64/12/16 is suggested. In which, 64 bits word length, 12 no. of rounds and 16 bytes key is used. The encryption and decryption can be done in few lines of code but the key schedule is more complex.

2.1. RC5 Key Expansion Method

In key schedule, User's key k is expanded to fill the key table S which resembles an array of t=2(r+1) random binary words determined by k. where, r is number of rounds. The key expansion function has certain amount of "one-wayness": it is not so easy to determine k from S. Key expansion is performed using the magic constants Pw and Qw. Where, w is word length. Magic constants are calculated as follows:

$Pw = Odd((e - 2)2^w)$

$Qw = Odd((\phi - 1)2^w)$

Where,

e = 2.718281828459…. (base of natural logarithms)

ϕ = 1.618033988749….. (golden ratio)

Odd(x) is the odd inter nearest to x.

In first step of key expansion the secret key K[0, 1,……, b-1] is copied into an array L[0,……c-1] of c= ceiling(b/u) words, where, u = w/8 is number of bytes/word. Any unfilled byte positions of L are zeroed.

In second step, array S is initialized to a particular fixed pseudo-random bit pattern, using magic constants Pw and Qw.

S[0] = Pw;

For i=1 to t-1 do

S[i] = S[i-1] + Qw;

In third step, mixing of user's secret key performed in threepasses over the array of S and L as follows:-

i=j=0;

A=B=0;
do 3* max(t, c)times:
A = S[i] = (S[i] + A + B) <<< 3;
B = L[j] = (L[j] + A + B) <<<(A + B);
i = (i + 1) mod(t);
j = (j+1) mod(c);

2.2. RC5 Encryption Method

Encryption is done by using 3 simple operations, addition of words modulo 2w, denoted by +, bit-wise exclusive-OR of words, denoted by⊕and left rotation , denoted by <<<. The pseudo-code of encryption on 2 w-bit registers A and B are as follows:-

A = A + S[0];
B = B + S[1];
for i = 1 to r do
A = ((A ⊕ B) <<< B) + S[2*i];
B = ((B ⊕ A) <<< A) + S[2*i+1];

2.3. RC5 Decryption Method

The decryption steps are easily derived from the encryption steps as follows:-
The two w-bit word inputs are denoted as A and B.
for i = r down to 1 do
B = ((B – S[2 * i + 1]) >>> A) ⊕ A;
A = ((A – S[2 * i] >>> B) ⊕ B;
B = B - S[1];
A = A - S[0];

3. Proposed Methodology for SNEP in SPINS Framework of WSN

Implementing security in sensor network is different as compared to traditional systems because limited battery and storage is available. Many researchers have studied the security and energy efficiency separately. These issues are trade-off. When security is implemented the energy efficiency is minimized and vice versa. Here, a security protocol is proposed in which the energy efficiency is better than the old one and the security is also not compromised. In SPINS for security RC5 is used by UC Berkeley but after analyzing RC5 with XTEA, XTEA was found better in all the main aspects. The main goal is to maximize the security by minimizing the resource requirements. A light weight, secure and fast encryption algorithm, extended Tiny Encryption Algorithm (XTEA) is available. XTEA [3] is a Feistel cipher proposed by Needham and Wheeler as an enhanced version of TEA. TEA [4] is a very fast block cipher that does not require much initialization time but Related-key attack was proposed by Kelsey, *et al.*, [11]. Hence, the authors proposed XTEA to avoid this attack. Some researches show that XTEA is one of the fastest cryptographic algorithms in existence. XTEA use modular addition, shift and XOR functions in their round function to provide Shannon's properties of diffusion and confusion which is necessary for a secure block cipher without the need for P-boxes and S-boxes. The authors estimated that being quite fast, it has a security comparable with the DES (Data Encryption Standard).

3.1. XTEAEncryption Method

XTEA is a 64-bit block ciphers with 128-bit key-length. The key K = (K0, K1, K2, K3) where, Ki(i=0,…..,3) are 32-bit words. Plain text is in vector form v[0] and v[1] and cipher text comes in w[0] and w[1]. Number of rounds n is 32 and a constant delta is used in encryption. Cipher text is calculated by performing left shift (<<), right shift (>>) and XOR (^) operations on the inputs and adding sum to the key. To select the key, binary AND (&) operation is used. The encryption process is as follows:-

```
For n= 1 to 32
{
 y +=(((z <<4)^(z>>5))+z)^(sum+k[sum&3]);
sum +=  delta;
z +=(((y<<4) ^ (y>>5))+y)^(sum+k[sum>>ll&3]);
}
w[0]=y;  w[l]=z;
```

3.2. XTEA Decryption Method

Decryption steps are the reverse of the encryption steps.

```
For n=32 down to 1
{
z -= (((y<< 4)^ (y >> 5)) + y)^(sum + key[(sum>>11) & 3]);
sum -= delta;
y -= (((z << 4) ^ (z >> 5)) + z) ^ (sum + key[sum & 3]);
}
v[0]=y;  v[l]=z;
}
```

4. Performance Evaluation of Proposed and Existing Encryption Methodsin SPINS Framework for WSN

To compare the performance of the existing method with the proposed one, many parameters likes speed, energy consumption and security are used. For performance evaluation MATLAB tool is used. 100 sensor nodes are densely deployed in the area of 100×100 meters. Encryption is performed on node 5. Node 5 transmits encrypted data to node 10 and decryption is performed there.

4.1 No. of Rounds, Key Length and Block Size

Here, same length of plaintext block is taken for both the algorithms. The length of plaintext and the cipher text is same so the transmission cost of both the algorithms is same and very low. Key lengthsare alsochosen same but in XTEA the key works as 4 key blocks of 32 bits were in RC5 it works as a whole S-table. This creates the difference in computation speed also. As shown in table, the encryption and decryption time of XTEA is far better than that of RC5. This key size also makes the big difference in storage requirements.

Table 1. Comparison Table of Algorithms based on Chosen no of Rounds, Key Length and Block Size

Name of encryption methods	Measurements			
	No of rounds	Key length	Length of plaintext	Length of cipher text
RC5	12	128 bits	64 bits	64 bits
XTEA	32	4, 32 bits blocks	64 bits	64 bits

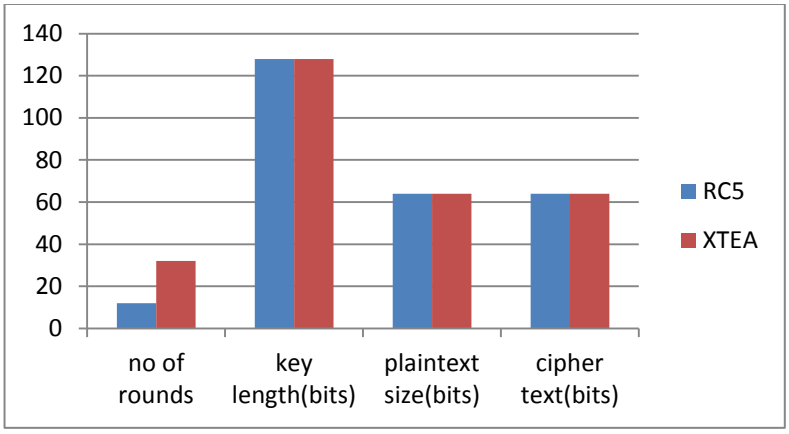

Figure 1. Graphical Representation on the Basis of no of Rounds, Key Length and Block Size

4.2. Storage Requirement

In RC5 while key expansion many variables are used. Hence, RAM requirement of RC5 is very high as compared to XTEA. For XTEA, RAM requirement is just 11 bytes where for RC5 it is 72 bytes. Similarly, different types of values are to be stored and the line of code is also very high in RC5 as compared to XTEA. For which ROM requirement is also very high. 1394 bytes ROM is required for code in XTEA and it is 3188 bytes for RC5.

Table 2. Comparison Table of Algorithms Based on Storage Required

Name of encryption methods	Storage Requirement	
	RAM required	ROM required
RC5	72 bytes	3188 bytes
XTEA	11bytes	1394 bytes

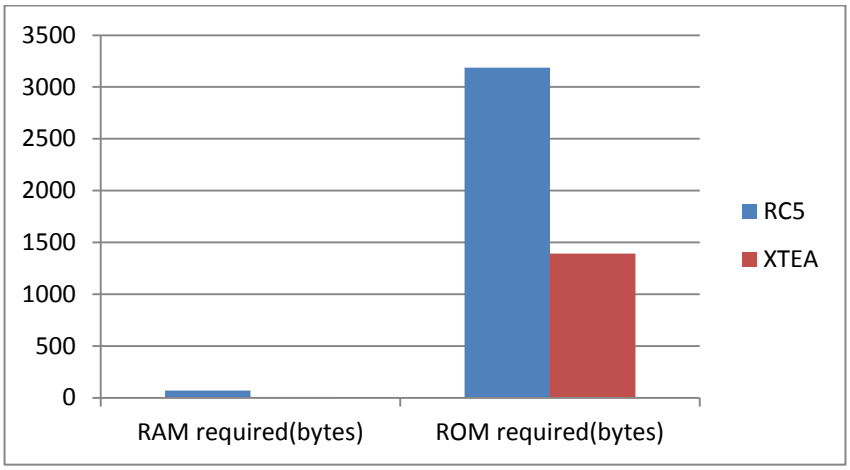

Figure 2. Graphical Representation of the Algorithms on the Basis of Storage Required

4.3. Execution Time

No of rounds of RC5 is less than XTEA but the calculations of RC5 are more time consuming than XTEA. The key expansion is also time consuming work for RC5. As the Table shows the initialization work takes 2410 µsec in RC5 which is just 10 µsec in XTEA.

Table 3. Comparison Table of Algorithms Based on Execution Time in µsec

Name of encryption methods	Time Required		
	Initialization	Encoding	Decoding
RC5	2410	829	817
XTEA	10	171	174

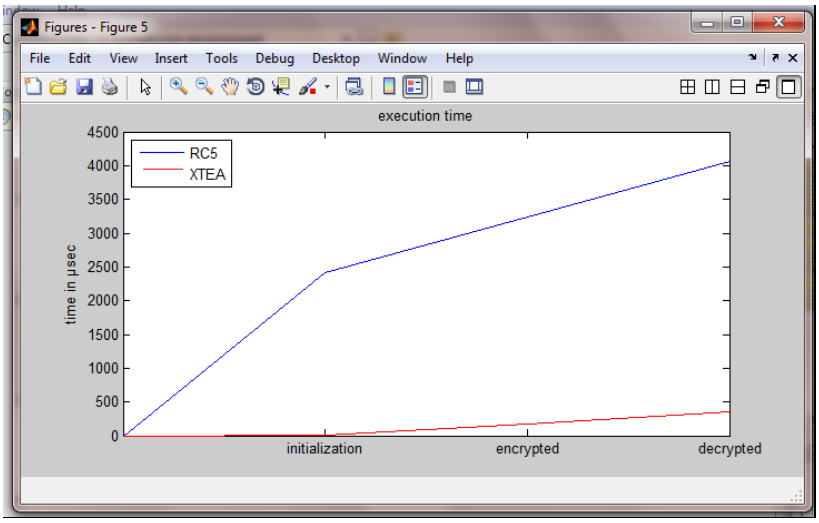

Figure 3. Graphical Representation of the Algorithms on the Basis of Execution Time in µsec

4.4. Energy Consumption

Energy consumption is directly dependent on computation complexity. It is assumed that .0001 joule energy is consumed in every computation. The energy of the node i, Energy (i), for N_C computations is calculated with the help of following formula:-

Energy(i) = Energy(i) - 0.0001* N_C

Decrease in energy for XTEA is shown by red color and that of RC5 is in blue color. As shown by the Figure 4, constant amount of energy is consumed in every round of encryption. Sharp declination shows that the initialization energy consumption is very high in RC5 and very low in XTEA and also the energy required while the rounds of encryption is low in XTEA than RC5.

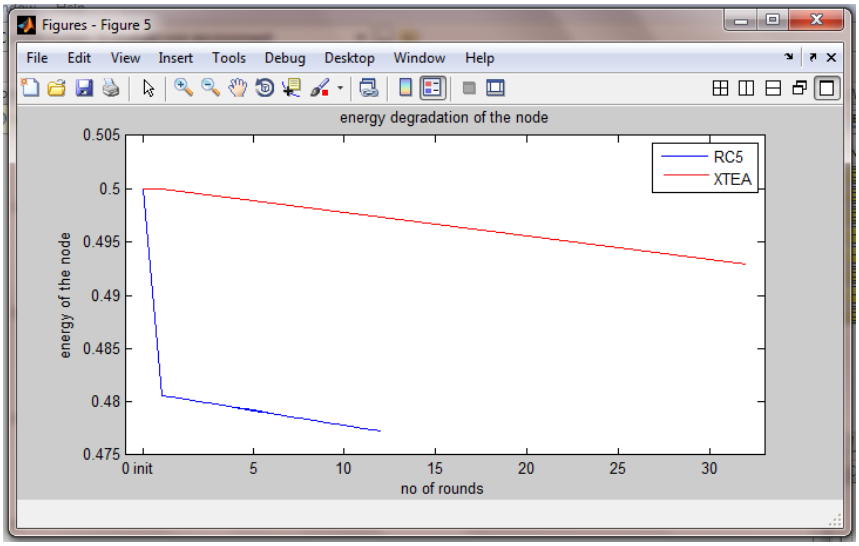

Figure 4. Graphical Representation of the Algorithms on the Basis of Energy of the Node after Initialization and Rounds of Encryption

4.5. Cryptanalysis on XTEA

Many researchers have performed cryptanalysis on XTEA and found it very secure. Sasaki, *et al.*, [16] have presented an impossible differential attacks on 23-round XTEA. Chen, *et al.*, [5] have shown how the three-subset meet-in-middle attack is applied against 25-rounds of XTEA with 9 known plaintexts and $2^{120.4}$ XTEA computations. In the single-key setting, Moon, *et al.*, gave impossible differential attacks on 14-round XTEA [9] based on 12-round impossible differentials. Hong, *et al.*, [6] proposed truncated differential attacks that can break XTEA reduced to 23 rounds with $2^{120.65}$ encryptions. Later, Sekar, *et al.*, [10] presented a meet-in-the-middle attack on 23-round XTEA with complexity 2^{117}. Very recently, Bogdanov, *et al.*, [13] proposed attacks on XTEA with a new technique named zero correlation linear cryptanalysis [12]; these attacks are best attacks on XTEA in terms of the number of rounds to date, which can break 27 rounds of XTEA using the whole code book. There are also attacks on XTEA in the related-key setting, which are given in [14, 7, 15, 8]. Moon, *et al.*, [9] have worked on 12-round impossible differential of XTEA, and result in improved impossible differential attacks on 23-round XTEA. Although, the attacksproposed by A. Bogdanov, *et al.*, [13] in XTEA are not very good, they greatly improve the corresponding impossible differential attacks [9].

5. Conclusion

After critical analysis, this paper shows that XTEA is more efficient and secure than other existing encryption algorithms for WSN. The key evaluation, encryption and decryption are less complex than other methods and are very simple to implement. This paper also analyzed the XTEA and RC5 encryption algorithms in WSN environment withthe help of many parameters. MATLAB simulated WSN environment shows the results are more efficient in terms of storage, energy and speed.Consequently, XTEA consume less energy & time as compare to RC5 encryption algorithm for implementing security in SNEP Method of SPINS framework for WSN.

Reference

[1] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks", Proceedings of Mobi Com, **(2001)** July, pp. 189-199, Rome, Italy.
[2] R. Rivest, "The RC5 Encryption Algorithm", in Fast Software Encryption of Lecture Notes in Computer Science, Preneel, B., Ed.,Springer-Verlag, vol. 1008, **(1995),** pp. 86-96.
[3] R. Needham and D. Wheeler, "TEA extensions, Technical report", Computer Laboratory, University of Cambridge, **(1997)**.
[4] R. Needham and D. Wheeler, "TEA, a Tiny Encryption Algorithm", in Fast Software Encryption: Second International Workshop, Lecture Notes in Computer Science, Springer, vol. 1008, **(1994)**, pp. 14-16.
[5] J. Chen, M. Wang and B. Preneel, "Impossible Differential Cryptanalysis of the Lightweight Block Ciphers TEA, XTEA and HIGH", in Progress in Cryptology – AFRICACRYPT, LectureNotes in Computer Science, Springer, **v**ol. 7374, **(2012),** pp. 117-137.
[6] S. Hong, D. Hong, Y. Ko, D. Chang, W. Lee and S. Lee, "Differential Crypt analysis of TEA and XTEA", in Lim, J. I., Lee, D. H. (eds.) ICISC, Lecture Notes in Computer Science, Springer, vol. 2971, **(2003),** pp. 402-417.
[7] Y. Ko, S. Hong, W. Lee, S. Lee and J. S. Kang, "Related Key Differential Attacks on 27 Rounds of XTEA and Full-Round GOST,in Roy, B.K., Meier, W. (eds.) FSE", Lecture Notes in Computer Science, Springer, **v**ol. 3017, **(2004),** pp. 299-316.
[8] J. Lu, "Related-key Rectangle Attack on 36 Rounds of the XTEA Block Cipher", Int. J. Inf. Sec., vol. 8, no. 1, **(2009),** pp.1-11.
[9] D. Moon, K. Hwang, W. Lee, S. Lee and J. Lim, "Impossible Differential Cryptanalysis of Reduced Round XTEA and TEA", In Daemen, J., Rijmen, V. (eds.) FSE, Lecture Notes in Computer Science, Springer, vol. 2365, **(2002),** pp. 49-60.
[10] G. Sekar, N. Mouha, V. Velichkov and B. Preneel, "Meet-in-the-Middle Attacks on Reduced-Round XTEA", in Kiayias, A. (ed.) Topics in Cryptology - CT-RSA,Lecture Notes in Computer Science, Springer, vol. 6558, **(2011),** pp. 250-267.
[11] J. Kelsey, B. Schneier and D. Wagner, "Related-key Cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA", in Han, Y., Okamoto, T., Qing,S. (eds.) ICICS, Lecture Notes in Computer Science, Springer, vol. 1334, **(1997),** pp. 233-246.
[12] A. Bogdanov and V. Rijmen, "Zero-Correlation Linear Cryptanalysis of Block Ciphers", IACR CryptologyePrint Archive, vol. 123, **(2011)**.
[13] A. Bogdanov and M. Wang, "Zero Correlation Linear Cryptanalysis with ReducedData Complexity", Pre-proceedings of FSE, **(2012)**.
[14] C. Bouillaguet, O. Dunkelman, G. Leurent and P. A. Fouque, "Another Look at Complementation Properties", in Hong, S., Iwata, T. (eds.) FSE. Lecture Notes in Computer Science, Springer, vol. 6147, **(2010),** pp. 347-364.
[15] E. Lee, D. Hong, D. Chang, S. Hong and J. Lim, "A Weak Key Class of XTEA for a Related-Key Rectangle Attack", In: Nguyen, P.Q. (ed.) VIETCRYPT. LectureNotes in Computer Science, Springer, vol. 4341, **(2006),** pp. 286-297.
[16] Y. Sasaki, L. Wang, Y. Sakai, K. Sakiyama and K. Ohta, "Three-subset meet-in-the-middle attack on reduced xtea", in Progress in Cryptology - AFRICACRYPT. Lecture Notes in Computer Science, Springer, vol. 7374, **(2012).** pp. 138-154.

Authors

Vartika Shah, she received B.Tech (Computer Science and Engineering) from BSA College of Engineering and Technology, Mathura in 2010, pursuing M.Tech (Information Technology) from Madhav Institute of Technoly and Science, Gwalior.

Sanjiv Sharma, he works as an assistant professor in Department of Computer science Engineering and Information Technology in Madhav Institute of Technology and Science, Gwalior. He have 7 year of teaching and research experience. He received his B.E from Pt. Ravishankar University, Raipur, and M.Tech. From RGPV, Bhopal and Ph. D for BansthaliVidhyaPeeth, Jaipur. His area of research is Network security, Data Mining and Social Network Analysis.