Encryption and Hash based Security in Internet of Things

B.Vinayaga Sundaram
HOD, Department of Information Technology
Madras Institute of Technology, Anna
University Chennai, India
bvsundaram@annauniv.edu

Ramnath.M
Department of Information Technology
Madras Institute of Technology,Anna University
Chennai,India
mramnath94@gmail.com

Prasanth.M
Department of Information Technology
Madras Institute of Technology, Anna
University
Chennai,India
prasanthmit@gmail.com

Varsha Sundaram.J
Department of Information Technology
Madras Institute of Technology, Anna
University
Chennai,India
varsha.joshi02@gmail.com

Abstract— **The Internet of Things (IoT) promises to be the next big revolution of the World Wide Web. It has a very wide range of applications, ranging from smart cities, smart homes, monitoring radiation levels in nuclear plants, animal tracking, health surveillance and a lot more. When nodes in wireless sensor networks are monitored through internet it becomes a part of Internet of Things. This brings in a lot of concerns related to security, privacy, standardization, power management. This paper aims at enhancing security in smart home systems. Devices like thermostat, air conditioners, doors and lighting systems are connected with each other and the internet through the internet of things technologies. Encryption and hash algorithms are proposed in this paper through which devices in the IoT can securely send messages between them. Encryption algorithm is used to ensure confidentiality as the attackers cannot interpret the cipher text that is sent. In order to ensure integrity (cipher text is not changed) hash algorithm is used.**

Keywords— Internet of Things, Wireless Sensor Networks, Security, Smart Homes

I. INTRODUCTION

When objects, people or animals are provided with unique identifiers and are able to communicate with each other without human intervention it is referred as the Internet of Things or Internet of Objects. It has a wide range of applications from automobiles with sensors, biochip transponders on animals, heart monitoring, devices that assist fire fighters etc. Four major challenges in IoT are power management, deployment of IPv6, standardization and security. IoT applications are mostly linked to sensitive infrastructure like electricity or water supply in smart cities or sensitive information like health or wealth monitoring of a person. Hence security mechanisms must be provided such that these information's are not accessed by unauthorized persons. Moreover IoT attacks are a shift from software attacks to hardware attacks. Most of the recent IoT applications are concerned with healthcare and monitoring and hence security is a key concern in IoT.

II. RELATED WORK

The Internet Engineering Task Force (IETF) has taken great efforts to standardize security solutions for the IoT system. Constrained Application protocol (CoAP), a standard security protocol is discusses in [9]. It is an application protocol which was specially designed to be adapted to the constraints in IoT. CoAP operates on top of Datagram Transport layer security (DTLS). Hence standardisation is necessary to adapt and enhance DTLS for IoT ecosystems. This includes the use of a) raw public key in DTLS and b) extending DTLS record Layer to protect group (multicast) communication and c) profiling DTLS for reducing the size and complexity of implementations on embedded devices. Compression schemes to mitigate message fragmentation issues in DTLS, proposed by IETF are also discussed here. If a Wireless Sensor Network (WSN) is integrated into the Internet as a part of the Internet of Things (IoT), new security challenges will appear, such as setup of a secure channel between a sensor node and an Internet host [3].A signcryption scheme that consists of an heterogeneous offline and online mode is used to secure communication between a sensor node and an Internet host. This scheme is secure against adaptive chosen cipher text attacks. Its advantages are, First, it achieves

978-1-4673-6823-015$31.00 2015 IEEE

confidentiality, integrity, authentication, and non-repudiation in a logical single step. Second, it allows a sensor node in an identity-based cryptography to send a message to an Internet host in a public key infrastructure. Third, it splits the signcryption into two phases, offline phase and an online phase. In the offline phase, without the knowledge of the message, heavy computations are done. In the online phase, only light computations are done when a message is available. This scheme is very suitable to provide security solution for integrating WSN into the IoT. This scheme provides security mechanisms only when a node in the Identity based cryptography (WSN) sends a message to a node in the Public key Infrastructure. It doesn't support reverse mechanism and provides security against outsider attacks only (attacks are assumed to be done only by an attacker who is neither the sender not the receiver). The idea of applying IoT technologies to smart home system is discussed [1] . An original architecture of the integrated system is analysed with its detailed introduction. This architecture has great scalability. Based on this proposed architecture many applications can be integrated into the system through uniform interface. Agents are proposed to communicate with appliances through RFID tags. It discusses why existing algorithms like RC-5, AES and Skipjack and RC-5 cannot be used for encryption [11]. Cryptography is essential for the security of wireless sensor networks. There are many constraints for wireless sensor networks like processing speed, memory size and energy. The encryption algorithm that is used for security must take these constraints into considerations. Energy consumption is a key constraint that affects the network lifetime. The encryption algorithm should have basic operations like exclusive-OR, shifting and addition so that it uses less energy and also must be less hardware dependent. But reduction in hardware resource and using simple operations should still ensure good security. By reducing hardware resources, the power consumption of the Wireless sensor networks is also reduced. An encryption algorithm is discussed which processes blocks of 64-bits using master key of 128-bits. . [12] Explains an IoT application named smart community which refers to a paradigmatic class of cyber-physical systems with cooperating objects (i.e., networked smart homes). It explains the architecture and how security mechanisms are achieved in this environment with the help of Neighbourhood watch and Pervasive Healthcare.

III. Proposed Work

Major security issues occur only when nodes are connected to the internet. So instead of connecting the devices separately to the internet, a common access point can be setup from which the nodes can get access to the internet. So network security can be applied to the single access point. Thus unauthorized access can be prevented at the single access point itself as shown in fig. 1. The existing scheme ensures security only when the sender belongs to the Identity based cryptography and receivers belong to public key infrastructure. So using other key and signature mechanisms, the reverse messaging scheme (i.e. sender in PKI and receiver in IBC) can also be done.

The main problem can be now divided into two sub-problems.

1. Security within the network
2. Security when WSN is connected to Internet.

A cryptographic algorithm is devised for ensuring security within the Wireless Sensor Network (Intra network security). This algorithm is devised in such a way that it is suitable for sensor nodes. Sensor devices have limited memory size, processing speed and energy supply. Hence, the cryptographic algorithm should be developed keeping in mind these constraints. The goal of the algorithm must be to ensure encryption and integrity. Since sensor nodes have limited memory and processing power the algorithm should not be more software oriented. The existing algorithms are discussed below

(i) RC-5

The plain text can be of 32,64 or 128 bits and key size can be 0 to 2048 bits. There can be varying number of rounds from 0 to 255. By default it has a block size of 64-bits key size of 128 bits and 12-rounds.12-round RC-5 (with 64-bit blocks) is susceptible to a differential attack using 244 chosen plaintexts. More number of rounds is suggested for better security. It uses more CPU power.

(ii) Skipjack

Plain text is of 64 bits and key size is 80 bits. It is an unbalanced Feistel network with 32 rounds. It is prone to attacks because of its shorter key length of 80 bits.

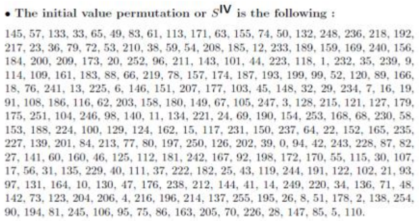
(iii) AES

It has block size of 128 bits and key sizes can be either 128,192 or 256 bits with 10, 12 and 14 rounds each. Shift rows, mix columns, add round key and sub bytes are the operations of each round. It uses complex operations.

The proposed algorithm takes a 64 bit binary plain text and a 128 bit master key as input. It produces a 64 bit cipher text as output. It has 32 rounds of operation. The main functions of the algorithm are initial transformation, whitening key generation, sub-key generation, round function and final transformation. Whitening keys and sub keys are generated from the master key. Totally 128 sub keys are required (4 keys for each round). After each operation changes are made to the plain text. Each operation is followed by a modular division by 8 operations. Hence, the each byte of the cipher text is lesser than 8. Each round function has two functions F0 and F1.

Confidentiality just ensures that messages are encrypted, so that attackers cannot identify what is sent. The attackers still have the option to tap the encrypted messages that are transferred and change the encrypted messages. So integrity algorithms are required to notify the receiver's whether the encrypted messages are changed or not. In the existing RC4 based hash algorithm, initial values are stored (S1V). It requires memory space. So the proposed algorithm doesn't make use of initial values, instead manipulates values based on the input message. Randomness is not achieved because of the stored values. The algorithm becomes slightly predictable.

S = Temp1 . Temp2 . Temp1 is a cost effective function (i.e.) Concatenation requires more CPU cycles. This operation is removed in the proposed algorithm.



- The initial value permutation or S^{IV} is the following :

145, 57, 133, 33, 65, 49, 83, 61, 113, 171, 63, 155, 74, 50, 132, 248, 236, 218, 192, 217, 23, 36, 79, 72, 53, 210, 38, 59, 54, 208, 185, 12, 233, 189, 159, 169, 240, 156, 184, 200, 209, 173, 20, 252, 96, 211, 143, 101, 44, 223, 118, 1, 232, 35, 239, 9, 114, 109, 161, 183, 88, 66, 219, 78, 157, 174, 187, 193, 199, 99, 52, 120, 89, 166, 18, 76, 241, 13, 225, 6, 146, 151, 207, 177, 103, 45, 148, 32, 29, 234, 7, 16, 19, 91, 108, 186, 116, 62, 203, 158, 180, 149, 67, 105, 247, 3, 128, 215, 121, 127, 179, 175, 251, 104, 246, 98, 140, 11, 134, 221, 24, 69, 190, 154, 253, 168, 68, 230, 58, 153, 188, 224, 100, 129, 124, 162, 15, 117, 231, 150, 237, 64, 22, 152, 165, 235, 227, 139, 201, 84, 213, 77, 80, 197, 250, 126, 202, 39, 0, 94, 42, 243, 228, 87, 82, 27, 141, 60, 160, 46, 125, 112, 181, 242, 167, 92, 198, 172, 170, 55, 115, 30, 107, 17, 56, 31, 135, 229, 40, 111, 37, 222, 182, 25, 43, 119, 244, 191, 122, 102, 21, 93, 97, 131, 164, 10, 130, 47, 176, 238, 212, 144, 41, 14, 249, 220, 34, 136, 71, 48, 142, 73, 123, 204, 206, 4, 216, 196, 214, 137, 255, 195, 26, 8, 51, 178, 2, 138, 254, 90, 194, 81, 245, 106, 95, 75, 86, 163, 205, 70, 226, 28, 147, 85, 5, 110.

Fig. 1 Initial permutation

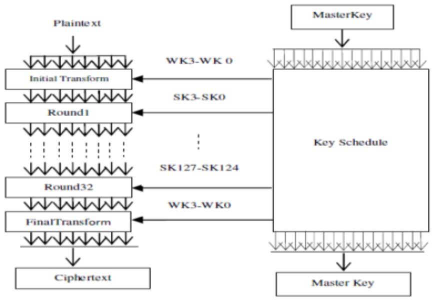


Fig. 2 Confidentiality architecture

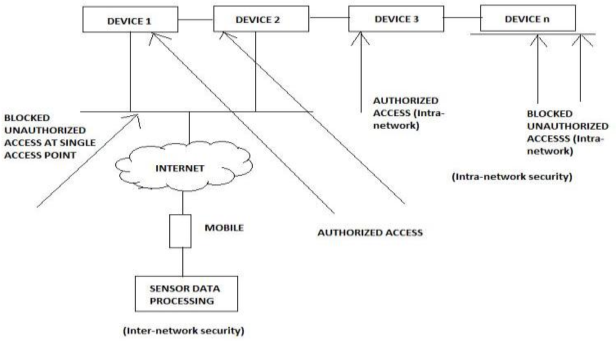


Fig. 3 Single access point architecture

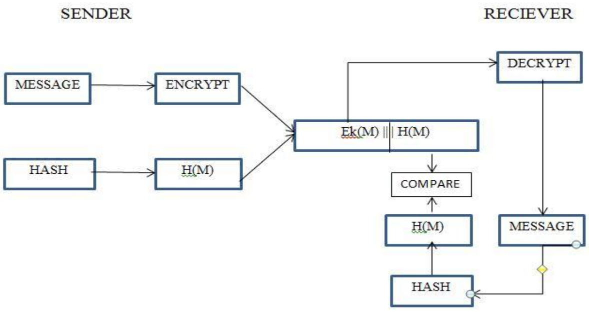


Fig. 4 Confidentiality and Integrity

The hash algorithm has 3 major steps

- Padding
- Compression
- Truncation

PADDING RULE: Message is split into blocks of 512 bytes. If the last block doesn't add up to 512 bytes then padding bits are added to make it 512 bytes. Padding bites are 1 followed by 0's (10000…….)

MESSAGE + PADDING BITS = 0 mod 512

There are 3 steps in Compression

Key Schedule Algorithm (KSA)

Modified Key Schedule Algorithm (KSA*)

Modified Pseudo Random Generation Algorithm

(PRGA*)

KEY SCHEDULE ALGORITHM:

INPUT : Message

OUTPUT : State array

Initialize state array from 1 to 256

Perform J= (j+s[i]+M[I mod 64]) mod 256 from 1 to 256

Swap s[i] and s[j]

MODIFIED KEY SCHEDULE ALGORITHM:

INPUT : Message and State array

OUTPUT : Updated State array

Perform J = (j+s[i]+M[I mod 64]) mod 256 from 1 to 256

Swap s[i] and s[j]

MODIFIED PSEUDO RANDOM GENERATION ALGORITHM (PRGA*)

INPUT : State array

OUTPUT : Updated state array

Do

i=(i+1) mod 256

J=(j+s[i])mod 256

While (length of message)

Swap s[i] and s[j].

The KSA and KSA* algorithm are almost similar except that updated state array and message are given as input to the KSA* algorithm while message is only given as input to KSA. Initialization of state array is done in KSA.

TRUNCATION RULE: One bit is taken from all the 256 bytes (256 bits). Then 16 more index bits are added. Thus the total number of bits are 256+16 = 272 bits (34 bytes).

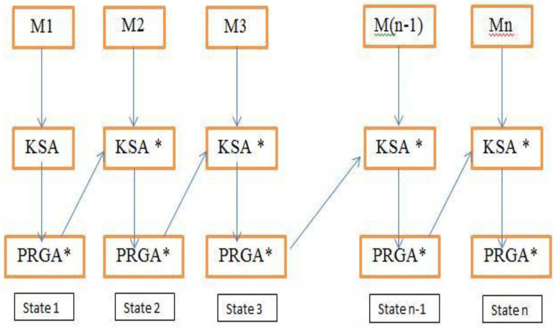


Fig. 5 Hash based algorithm

IV. Implementation and Analysis

One bit change in the plain text changes the cipher text completely. Thus diffusion is achieved.

PLAINTEXT
00000000000000010000001000000011000001000000010100
00011000000011 (0 1 2 3 4 5 6 3)

MASTER KEY
00000000000000010000001000000011000001000000010100
00011000000111000000000000000100000010000000110000
0100000001010000011000000111 (0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7)

CIPHER TEXT
00000100000001100000000000000110000000010000000100
00011100000000 (4 6 0 6 1 1 7 0)

Similarly one bit change in the master key changes the cipher text completely. Thus confusion is achieved.
For example

PLAIN TEXT 0 1 2 3 4 5 6 7
MASTER KEY 0 1 2 3 4 5 6 7 0 1 2 3 4 5 128 1 25
CIPHER TEXT 1 4 0 1 2 6 6 0

Known cipher text attack is not possible because of the large key length (128 bits). The attacker cannot generate 2^128 combinations. Similarly known plaintext and cipher text attack is also not possible.

For example, the attacker has a cipher text say 7 and it corresponds to the alphabet H of the plaintext. He cannot substitute H for every 7 in the cipher text because each time the algorithm generates different cipher text for the same plain text and key.

PLAIN TEXT H 0 1 2 3 4 5 6
MASTER KEY 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
CIPHER TEXT 7 6 4 2 2 4 0 3
PLAIN TEXT 0 1 2 3 4 5 6 H
MASTER KEY 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
CIPHER TEXT 3 4 5 5 7 7 4 3

There are no possible weak keys for this algorithm. Thus the algorithm is secure from several cryptographic attacks.

Table 1
Plain text and Cipher text of similar messages

PLAIN TEXT 1	CIPHER TEXT 1	PLAIN TEXT 2	1-BIT CHANGE	PLAIN TEXT 3	5-BIT CHANGE	PLAIN TEXT 4	10-BIT CHANGE
255	5	255	1	255	5	255	4
254	5	254	4	254	2	254	6
252	7	252	2	252	2	252	7
248	4	248	1	248	3	248	5
240	4	240	5	240	0	240	4
224	6	224	2	224	3	224	3
192	6	192	0	192	1	195	6
128	7	129	4	159	5	127	5

A. SECURITY ANALYSIS OF HASH ALGORITHM

The algorithm is collision free. For a random 'y' it is not possible to find a 'M' such that H(M)=y. It is collision resistant, preimage resistant and second preimage resistant. For a c-bit hash

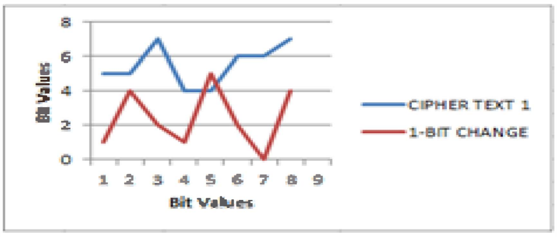


Fig. 6 1-bit change

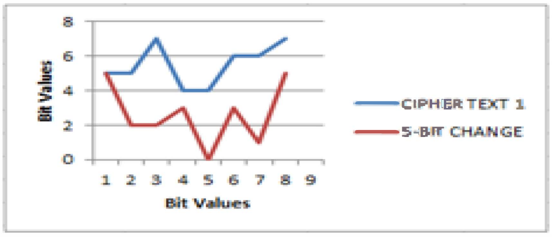


Fig. 7 5 –bit change

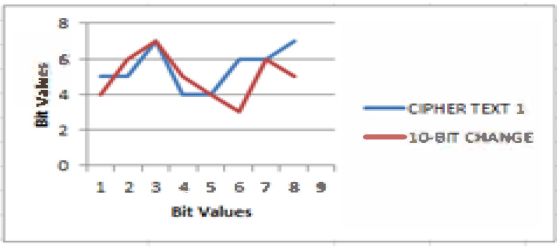


Fig. 8 10-bit change

function, collision attack takes 2^(c/2) complexity. Here c= 2048 and hence it takes 2^1024 complexity for collision attack. 2^c complexity is required for both preimage and second preimage attack. Here the complexity is around 2^2048. Kelsey scheneier second preimage attack is possible in complexity less than 2^(c/2). But if the final hash size is lesser than two times of intermediate state hash size then this attack is not possible.

Here, final hash size = 272 bits
Intermediate State size = 2048 bits
Therefore, w>2c (2048 > 2*272)

B. PERFORMANCE ANALYSIS OF HASH ALGORITHM

It is based on RC4 which itself is very quick. RC 4 based Hash takes only 3/50 times of the time taken by MD 4. This algorithm uses less operations than RC 4 based Hash (Concatenation based operation is not done). Significant increase in the number of output bits (272 bits). The algorithm can also work on 8-bit processors because RC4 based Hash can work on 8-bit processors

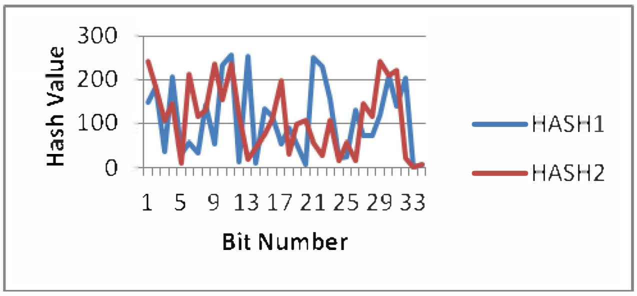


Fig. 9 Hash values of 2 similar messages rotated by 1 byte

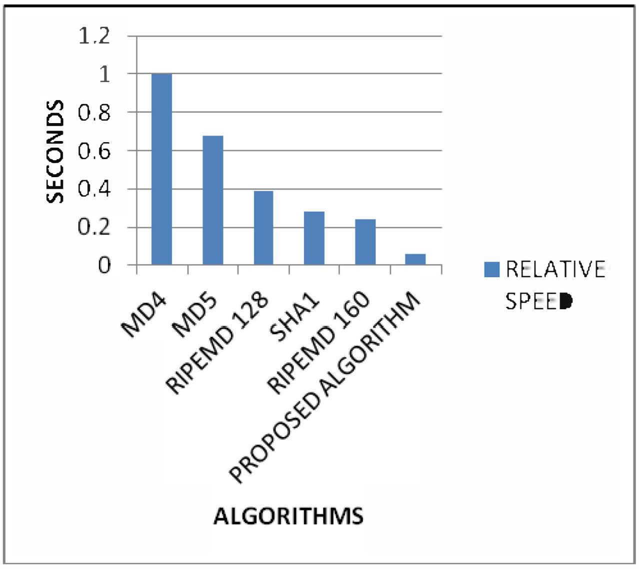


Fig. 10 Relative speed of authentication algorithms

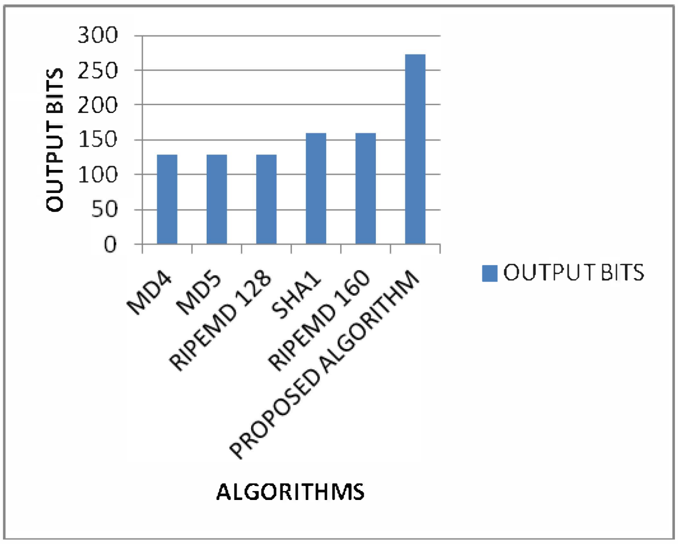


Fig. 11 Output bits of authentication algorithms

V. CONCLUSION

Thus the given cryptographic algorithm can be used for wireless sensor networks as it uses less complex operations. It is not prone to brute force attacks because of its key length (128 bits). It does not have any weak keys. Confusion and diffusion is also achieved through this algorithm. Hence even a single bit change in the plain text or master key changes the cipher text completely. Similarly the integrity algorithm can also be used for ultra-low power devices. It is highly secure, efficient (Collision, pre image and second pre image resistant) and fast (3/50th of time) when compared to the existing algorithms.

REFERENCES

[1] Alberto M.C Souza and Jose R.A. Amazonas, ‘A Novel Smart Home Application Using an Internet of Things Middleware’, European Conference on Smart Objects, Systems and Technologies, pp. 1-7, June 2013.

[2] Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi, ‘Internet of Things for Smart Cities’, IEEE Internet of Things Journal, Vol. 1 No.1, pp. 22-32, February 2014.

[3] Cristina Alcaraz, Pablo Najera, Javier Lopez, Rodrigo Roman,‘Wireless Sensor Networks and the Internet of Things:Do We Need a Complete Integration?’, SecIoT Japan, November 2010.

[4] Chang, K. C. Gupta and M. Nandi, "RC4-Hash: A New Hash Function Based on RC4", Proc. INDOCRYPT, LNCS 4329, pp. 80-94, Springer, 2006.

[5] Fagen Li and Pang Xiong , ‘Practical Secure Communication for Integrating Wireless Sensor Networks Into the Internet of Things’ IEEE SENSORS JOURNAL, Vol .13, No.10, pp. 3677-3684, October 2013.

[6] Kai Kang, Zhibo Pang, Li Da Xu, Liya Ma, and Cong Wang, ‘An Interactive Trust Model for Application Market of the Internet of

Things', IEEE Transaction on Industrial Informatics, Vol. 10 No.2, pp. 1516-1526, May 2014.

[7] Kelly, S.D.T., Suryadevara, N.K., Mukhopadhyay, S.C., 'Towards the implementation of IoT for Environmental Condition Monitoring in Homes', Vol.13 No.10, pp. 3846-3853, August 2013.

[8] M.Tharani, M.Senthilkumar , 'Integrating Wireless Sensor Networks into Internet Of Things For Security' IJIRCCE Vol.2 No.1, March 2014.

[9] Ming Wang, Guiquing Zhang, Jianbin Zhang, Chengdong Li, 'An IoT-based Appliance Control System for Smart Homes' ICICIP , pp. 744-747, 2013.

[10] Dr. Pritam Gajkumar Shah, Javeria Ambareen, 'A Survey of Security Challenges in Internet of Things (IoT) Integration with WSN', AUSJOURNAL, 2014.

[11] Pormante, L. Rinaldi, C. Santic, M.Tennina, S., 'Performance analysis of a lightweight RSSI-based localization algorithm for Wireless Sensor Networks', ISSCS , pp. 1-4, June 2013.

[12] Sang-Eon Lee, Sang-Ho Shin, Geum-Dal Park and Kee-Young Yoo, 'Wireless Sensor Network Protocols for Secure and Energy-Efficient Data Transmission' Proceedings of the CISIM'08 on Computer Information Systems and Industrial Management Applications, pp. 157-162, June 2008.

[13] Sye Loong Keoh, Sandeep S. Kumar, and Hannes Tschofenig,'Securing the Internet of Things A Standardization Perspective', IEEE INTERNET OF THINGS JOURNAL, Vol. 1, No. 3, pp. 265-275, June 2014.

[14] Wade Trappe, Lawrence C. Washington, 'Introduction to Cryptography with Coding Theory.

[15] Woo Kwon Koo, Hwaseong Lee, Yong Ho Kim, Dong Hoon Lee, 'Implementation and Analysis of New Lightweight Cryptographic Algorithm Suitable for Wireless Sensor Networks' International conference on Information assurance and security, pp. 73-76., 2008.

[16] Xu Li, Rongxing Lu, Xiaohui Liang, and Xuemin (Sherman) Shen, 'Smart community An Internet of Things Application', IEEE Communications magazine, Vol. 49 No.11, pp. 68-75, November 2011.