

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



A Study on Sensor Node Capture Defence Protocol for Ubiquitous Sensor Network

Yong-Sik Choi and Seung Ho Shin
*Dept. of Computer Science and Engineering,
 University of Incheon
 South Korea*

1. Introduction

Ubiquitous computing is being actively researched and one of the main technologies in ubiquitous computing environments is recognized as RFID and Sensor system. The RFID and Sensor system has much benefit but simultaneously has some problems such as user's privacy violation. USN (Ubiquitous Sensor Network) is a key to build ubiquitous computing environment, and it has been drawing attentions. Sensor node collects data through observation or detection as being installed at various places. Sensor node is to be placed at where an attacker can easily access.

This exposure raises the possibility of sensor node capture attach to dig encrypted secret, to change programming or control with ill intention. Thus, this is to study the basis of technology that USN technology can be actualized by drawing and suggesting fragility and requirements in security that can happen in USN, and by suggesting the direction of security service centered on safe key distribution, certification and node capture defense technology.

For the control of key that can be used for the certification of sensor node or encryption of sensed information, the researchers suggest a key control method and security protocol to defend against sensor node capture by applying PKI (Public Key Infrastructure) method to Hash Lock. For the control of key that can be used for the certification of sensor node or encryption of sensed information, the researcher have used MetaID as a secret key by applying PKI method to Hash Lock based on the difficulties in calculating inverse function of one-way hash function. Sensor is certificated by a registered open key (meta ID) and the meta ID creates the only key (k) of each sensor with meta ID = H(k). At this time, H() is Hash function. To transmit data safely, transmission node transmits data by encrypting data using link key, and the node that receives the data decodes it with its own secret key. All transmission nodes receive the certification of receiving nodes by transmitting their own data. Distributed secret key is re-encrypted on a regular basis regardless of the loss of key to raise safety and provides resilience against sensor capture.

The composition of this paper is as follows: in the second chapter, USN, Key management technology and Hash Lock Approach as related researches. In the third chapter, the design of security protocol using PKI will be presented. In the fourth chapter, experimental environment will be presented and there will be the conclusion in the final chapter.

Source: Convergence and Hybrid Information Technologies, Book edited by: Marius Crisan, ISBN 978-953-307-068-1, pp. 426, March 2010, INTECH, Croatia, downloaded from SCIYO.COM

2. Related work

2.1 Ubiquitous Sensor Network

USN(Ubiquitous Sensor Network) is a wire and wireless network, which consists of several sensor nodes deployed in a certain field. Sensor node should have the functions of computation, sensing and wireless communication. The number of sensor network is about 10~10,000 and its location is flexible depending on its necessity. Because the price per sensor node is not expensive, it is easy to embody sensor network. As sensor network standardizations, sensor interface standardization (IEEE 1451) and sensor network standardization (802.15.4) are being performed. Especially, ZigBee is characterized by low power and low cost, thus, with 2GHz -based wireless home network, it is possible to connect 255 device within 30m radius in a speed of 250kbs [1], [6].

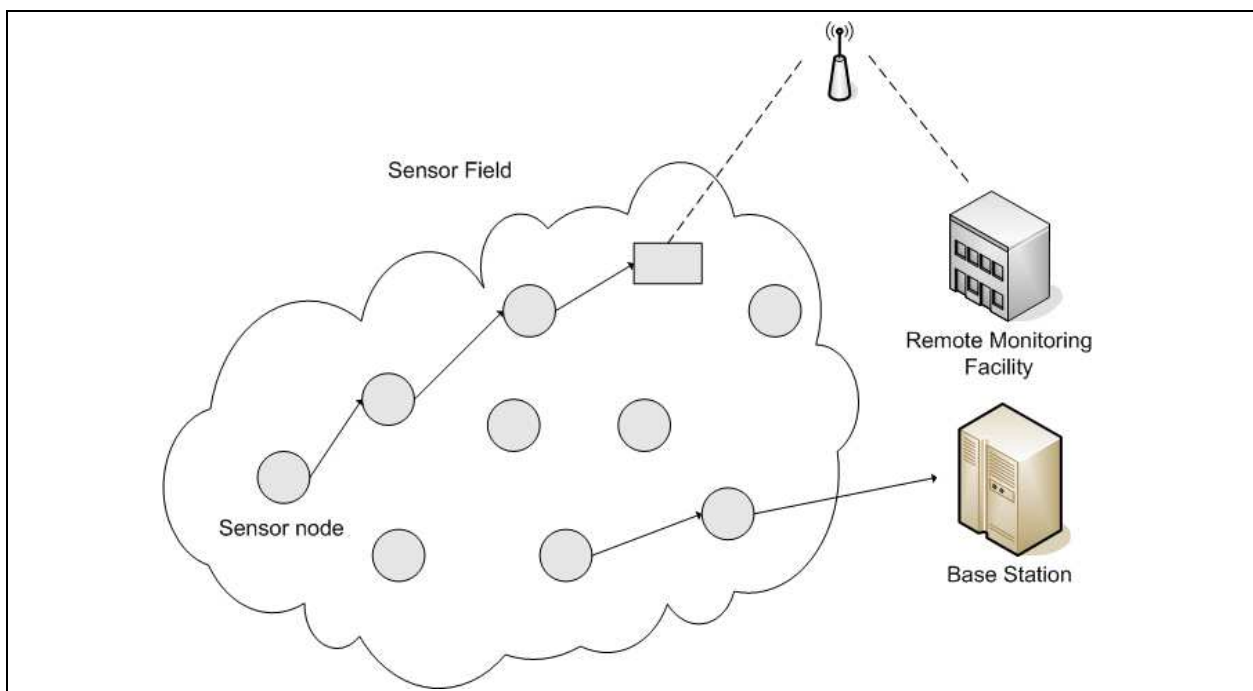


Fig. 1. Ubiquitous Sensor Network Architecture

2.2 Key management

Key control protocol plays the role to construct safe communication infra and to create secret key that are necessary for various security protocols by establishing the reliable relationship between sensor nodes, which are temporarily installed under the condition without reliable sensor or infra [2].

As key control mechanisms, there are pairwise key, random key pre-distribution, public key infrastructure and hierarchical key management [3].

As a shared key in network, pairwise key is used in encoding and authentication, providing effectiveness and simplification. However, the damage caused by node capture may affect entire network, and thus, additional key renewal protocol is required [3], [4].

Before installation random key pre-distributions receive random number of keys from the previously created key pool. It uses the key if neighboring key and common key exist, but it establishes link key through the connected neighboring node if they do not exist. In this method, depending on the density of sensor network, the entire network may not be entirely

connected. Furthermore, because it is a pre-distribution method, it is impossible to provide the newness of key.

Public key infrastructure provides hardware-based public key through the selections of proper algorithm, parameter and optimal embodiment. Although it suggests the application of public key technology in order to distribute low-frequency key, public key-based structure should be established in advance [4], [5].

Hierarchical key management includes hop by hop message encoding / decoding process by the key, which is distributed between nodes in advance according to the characteristic of hierarchical communication structured, suggesting the processing process of sensor addition/deletion. But intermediate node saves key in proportion to the number of child node, it require the capacity to create key. It cannot provide the resilience for sensor capture [3].

2.3 Hash Lock approach

The Hash-Lock approach proposed by Weis et al. Use the concept of locking and unlocking the tag to allow access. The security of the Hash-Lock approach uses the principle based on the difficulty of inverting a one-way hash function. The scheme makes use of a back-end database to provide correct reader to tag identification and the concept of meta-ID stored in each tag. To lock the tag the reader sends a hash of a random key, as the meta-ID, to the tag, i.e. $\text{meta-ID} = \text{hash}(\text{key})$. The reader then stores the meta-ID and key in the back end database. While locked, the tag only responds with the meta-ID when queried. As shown in Fig. 2, to unlock the tag, the reader will query the tag for the meta-ID. The reader will then use the meta-ID to lookup a key and ID for the tag in the database. If the meta-ID is found, the reader then sends the key to the tag in an attempt to unlock the tag. The tag hashes the key and compares the results against the meta-ID stored in the tag [2], [3], [6].

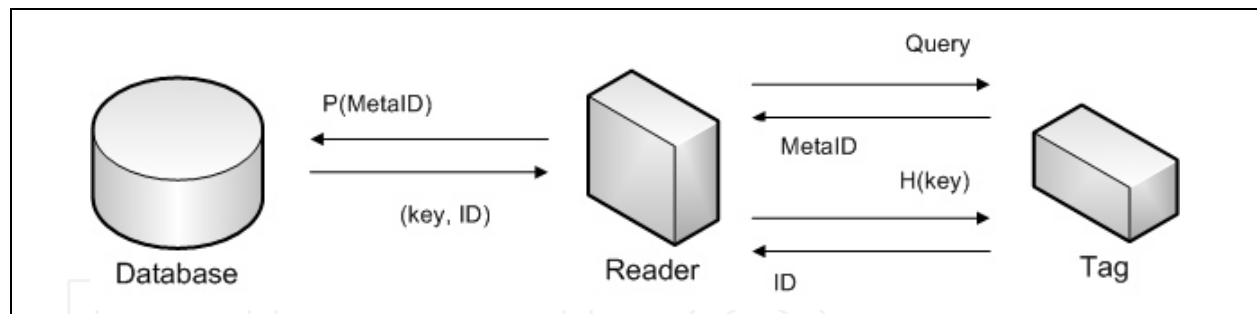


Fig. 2. Hash Locking Protocol

2.4 Hash function algorithm

The Cryptographic hash functions are playing very important roles in modern cryptology, such as checking the integrity of information or increasing the efficiency of authentication code and digital signature. When compared with general hash functions used in non cryptographic computer applications, although both cases are functions from domain to range, they are different from each other in several important aspects. Also, the hash function outputs the value called hash value or have code of fixed length by the input of messages having random length. More strictly, the hash function h correspond text alignment of random length as n bit text alignment of fixed length [5], [7].

When domain is called D and range is called R , the function $d h: D \rightarrow R (|D| > R)$ is a many-to-one corresponding function. Accordingly, the collision exists for the hash function in

general. For example, assuming function h as the one having input value of t bit and output value of n bit, the number of input values while h has randomness corresponds to each output value. Accordingly, two input values selected at random with probability 2^{-n} comes to have same output value regardless of the t value.

The handling process of most has functions is the repetitive one hashing the input of random length by divided processing of successive fixed blocks. First, the input X becomes padded to become a multiple of block length and divided from X_1 to t number of blocks as X_t . The hash function h is described as follows.

$$\begin{aligned} H_0 &= IV \\ H_i &= f(H_{i-1}, X_i), \\ 1 \leq i \leq t, \\ h(X) &= H_t \end{aligned} \quad (1)$$

Here, f is the compress function), H_i is the chaining variable between $i-1$ and i , while IV is the initial value. The general structure of repetitive hash function using compressed function is in the Fig. 3.

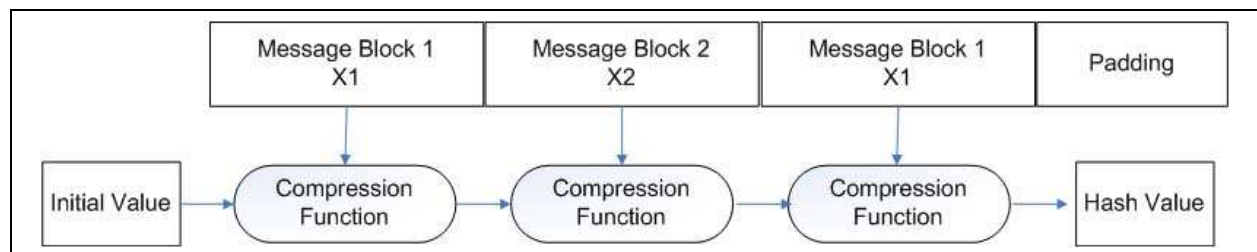


Fig. 3. Structure of the Hash-Function with recurrent

The calculation of hash value is dependent on the chain variable. While starting the hash calculation, this chain variable comes to have the fixed initial value expressed as the part of algorithm. The compressed function renews this chain variable by getting the message block as input until it becomes hashed. This process gets repeated in cycles for all message blocks, and the last value gets output as hash value on the same message. The hash function gets classified into 3 types depending on which structure is used as internal compressed function [8].

1. Hash-Functions based Block Cipher
2. Hash-Functions based Modular Calculation
3. The other Hash-Functions

The exclusive hash function has fast processing speed, and they're the functions specially designed for hashing regardless of other system sub factors. The exclusive hash function proposed until today has the structure based on MD4 designed by Rivest in 1990. There are MD5, SHA-1, RIPEMD-160 and HAVAL for hash functions of MD series being widely used at this time.

When a specific hash function is assigned, although it is ideal to verify the lowest limit on complications attacking the hash function for the establishment of safe hash functions, such method is not known for the most part in reality and the applicable known complication of the attack is considered as the security of hash function for the most part. If hash value is assumed as uniform probability variable, the following are well-known facts.

For the n bit hash function h , the guessing attack to discover preimage and second preimage with 2^n operation. For the attacker that is able to select messages, the birthday attack is able to discover the collision message pair M, M_t with about $2^{n/2}$ operation.

If n bit hash function satisfies the following two characteristics, it serves as an ideal security. Once the hash value is given, the discovery of preimage and second preimage requires 2^n operation.

3. Design of security protocol

In order to transmit data safely, transmission node transmits data by encoding it with link key, and the node that receives data decodes its own secret key. All transmission nodes certify received nodes by transmitting their own data. Distributed secret key is recreated and distributed in a certain cycle regardless of the loss of key.

For the network, which has the recovery power of sensor node capture, it makes copy on network, sends all packets via various independent passes, inspects consistency of the node receiving packet, excludes captured node from network, erases the information contained in sensor, and stops its function if sensor receives the password saved in its own data field.

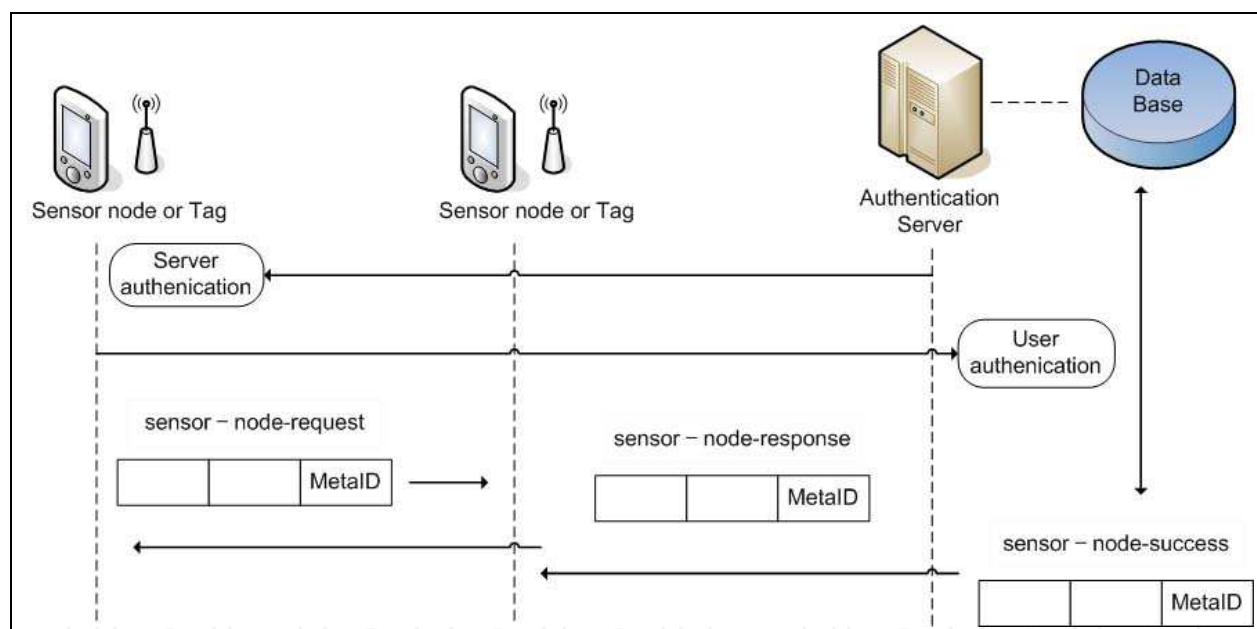


Fig. 4. The security protocol

[Step 1]

Authentication server creates and registers private key and public key using ECC(Elliptic Curve Crypto) about sensor node. Authentication server requires transmission of MetaID packet.

[Step 2]

Sensor node that receipt of message is response sensor-node-response message.

[Step 3]

Reader or sensor node is Authentication key generate by $P(\text{MetaID})$. And send a value.

$\{\text{sep_ID}, P(\text{MetaID})\} = \text{Value}$

[Step 4]

Sensor node is response sensor-node-response packet using hash value.

[Step 5]

Reader or sensor node compare authentication database with receipt value and if value agrees. It is valid sensor node. Transmit key, ID to sensor node.

4. Experimental environment

4.1 Experimental

Implementation controls of Sensor nodes data transmission /reception cycle. The composition of implementation is as follows: Serial communication, node commander, topology view, data log and statistic view.

Fig. 6 sensor data shows the process. Data created in sensor node is moved through its own routing path and is concentrated on base node. Data of base node is passed on middleware that handles RFID/data of sensor node through serial communication. It is stored to database.

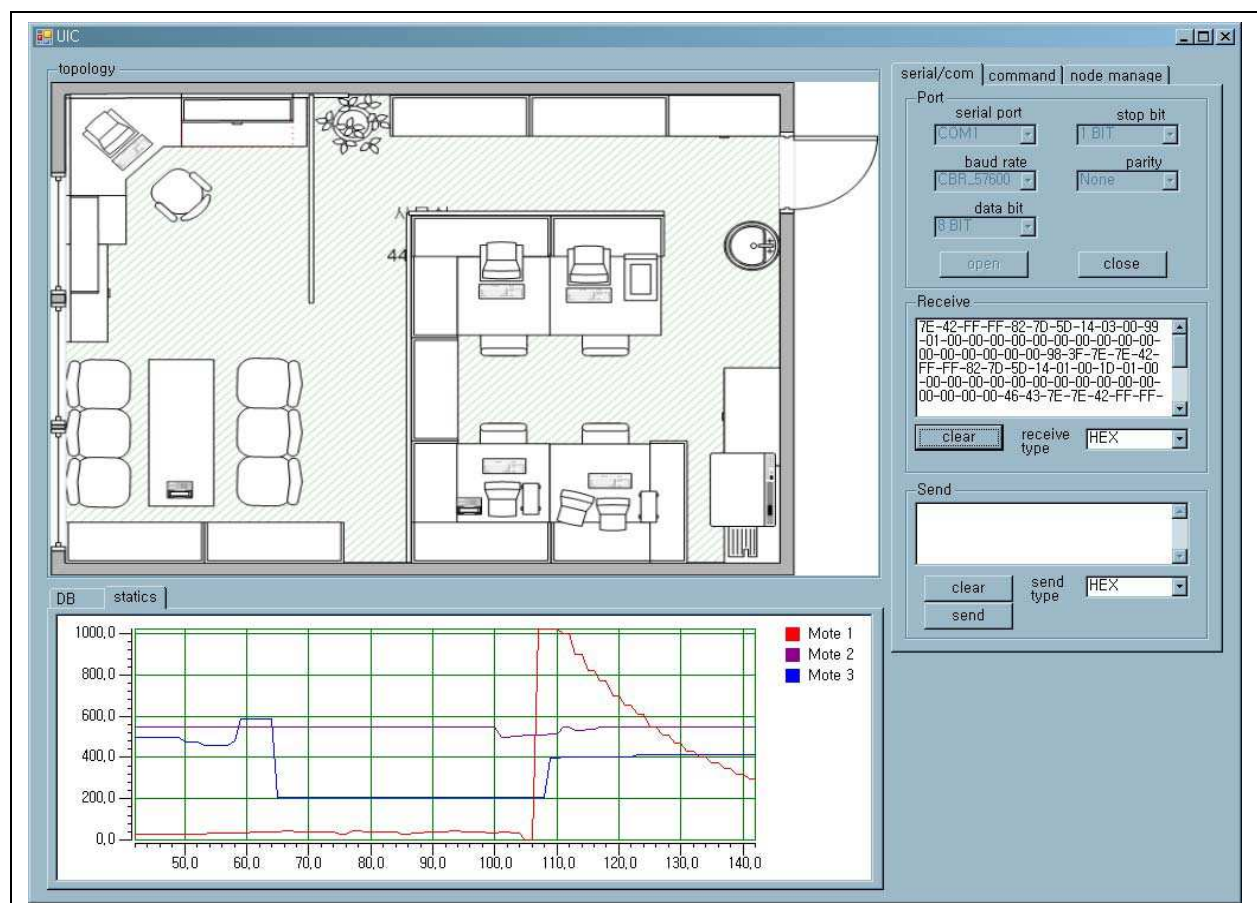


Fig. 5. The implementation architecture

It undergoes filtering to form statistics /analysis module and sensor topology mesh. And then, the user does command in sensor through sensor Commander and changes establishment of sensor.

Fig. 7 shows class diagram. It is RFID/Sensor node data collecting prototype class. For data collecting, we do not consider data pattern escape limit. With Simplex 1:n network satisfied and through application, what is planned is the command with structure simplified to architecture broadcast.

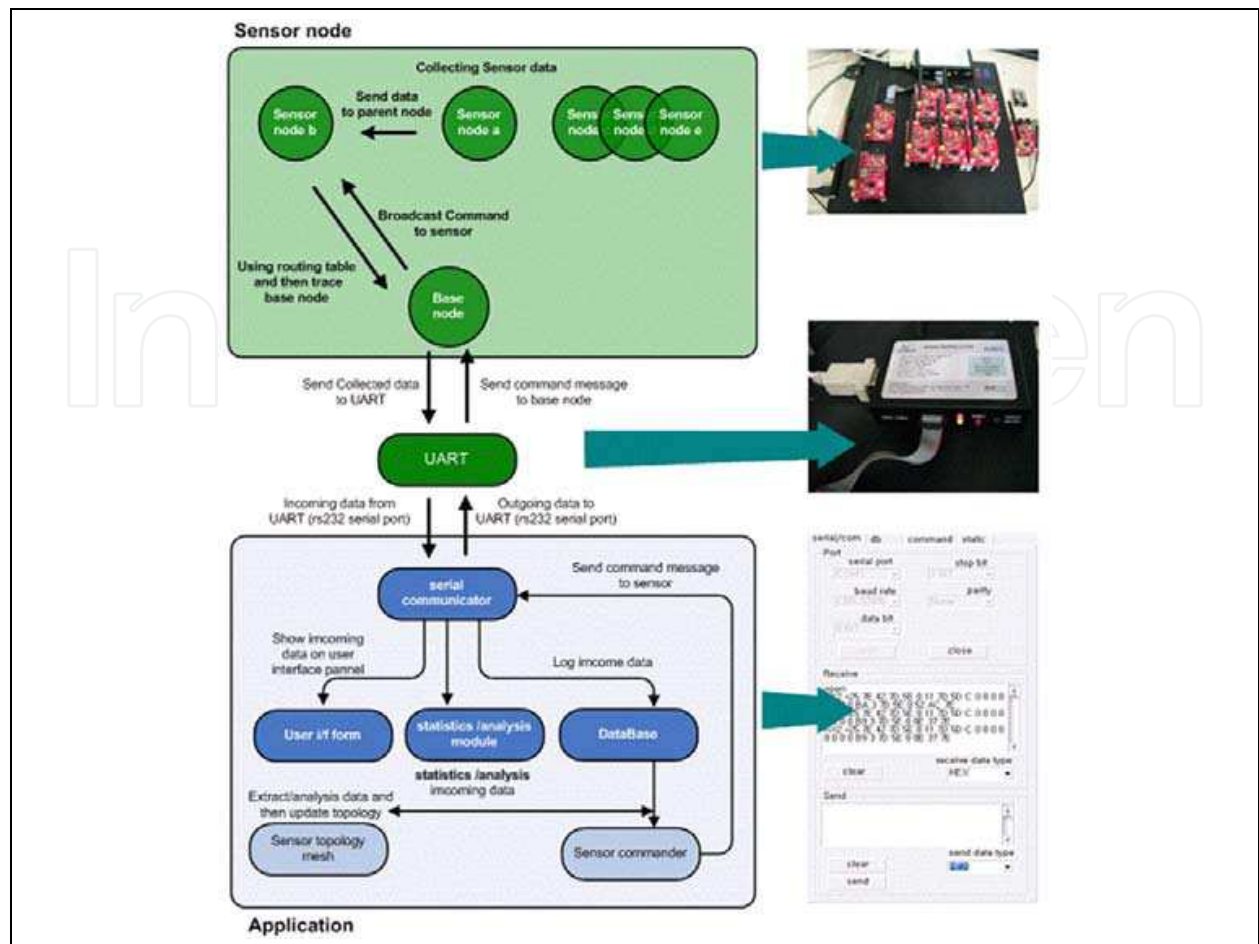


Fig. 6. Data flow

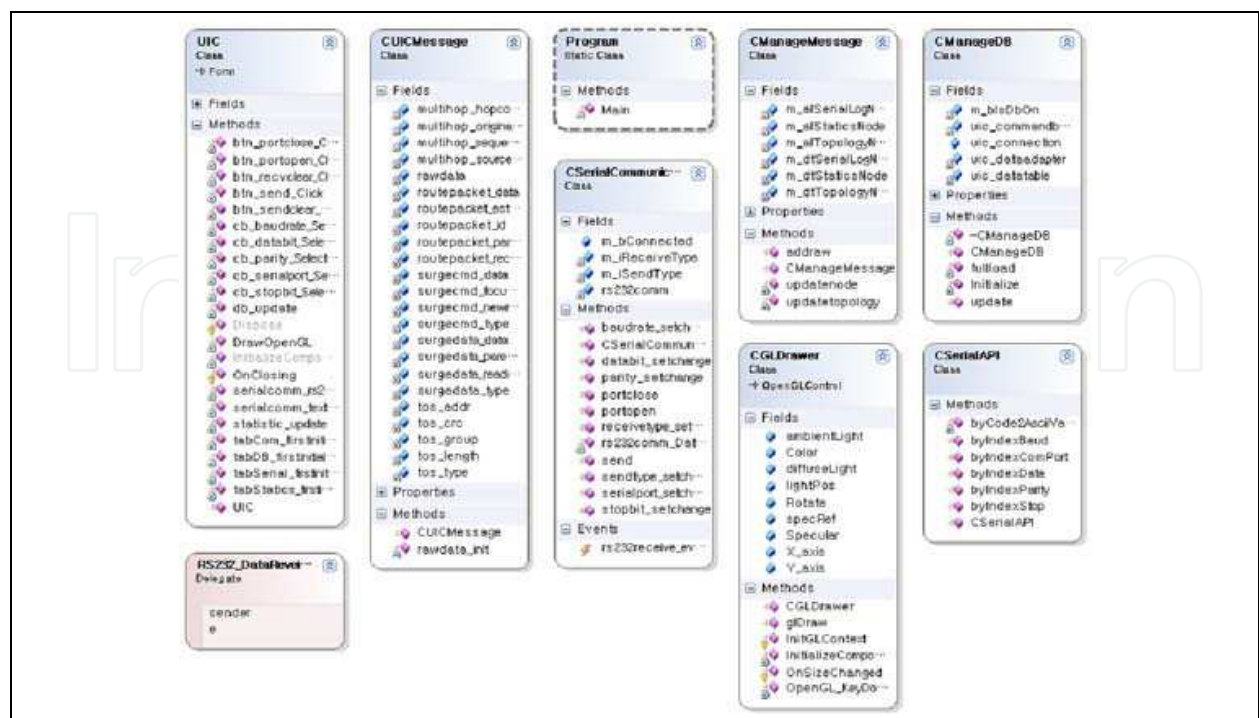
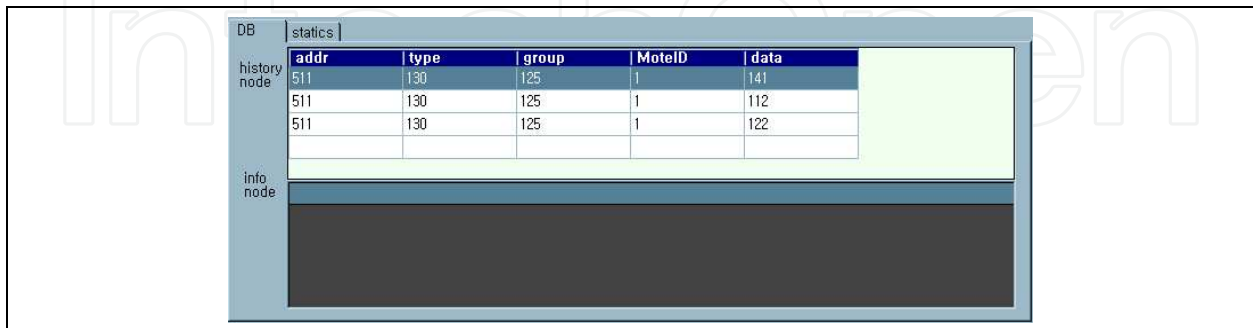


Fig. 7. Class diagram

4.2 Design of sensor node data modules

4.2.1 Data logging

Data, coming from Serial terminal, filters communication codes before inserted to CManageDB instance. Input is divided and use history node for pure log that stores occurrence data, using info node to keep dynamic topology and storing by history node and info node to database. Like Fig. 8. Stored data can be used in data sending/receipt state and network state analysis between sensor nodes.



| DB | | statics | | | | |
|--------------|--|---------|------|-------|---------|------|
| | | addr | type | group | MotelID | data |
| history node | | 511 | 130 | 125 | 1 | 141 |
| | | 511 | 130 | 125 | 1 | 112 |
| | | 511 | 130 | 125 | 1 | 122 |
| info node | | | | | | |

Fig. 8. Data logging

4.2.2 Serial communication

Data collected from sensor nodes, performing serial communication through PC's RS232 cable gathers on middleware. Data collection method of PC is established in serial communication form of Fig. 9. The establishment of sensor module except variable port of PC is 1 stop bit, 57600 baud rates, none parity and 8 data bits.

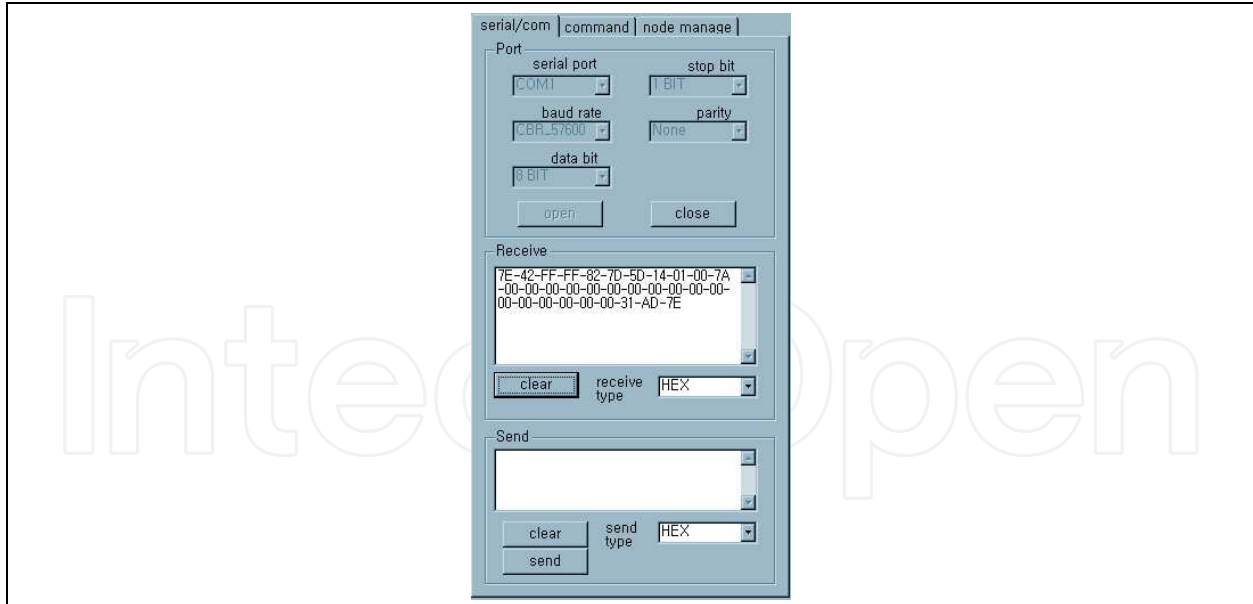


Fig. 9. Serial communication form

4.3 Sensor modules load code

Unnecessary transmission of information is possible because TinyOS mote msg can be commonly used. Therefore, we propose light lite mote msg. We have recorded lite mote msg in each sensor module, developing Mote hex image that is supposed to be loaded in sensor

module to do sending-receive in addition. It is in Fig. 10 and we have composed network in 1:N.

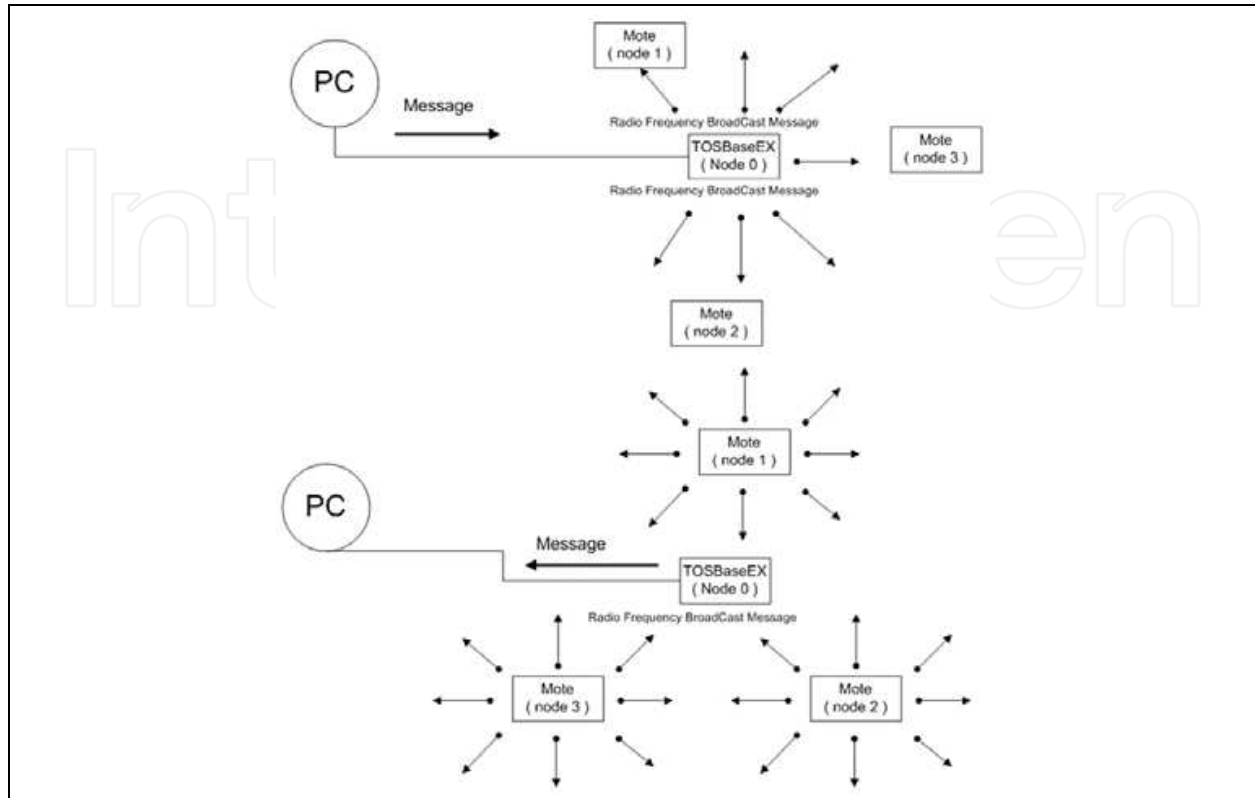


Fig. 10. Mote msg operability

4.4 Arrangement sensor nodes

We have collected data for compose reciprocity network and location of each sensors and pattern in each sensors.

We have arranged sensor nodes with Fig. 11 to collect sensor data. Sensors node compose network and collected data in limited area.

5. Conclusion

Sensor network is applied to various fields from the special application fields such as wild environment monitoring, industrial machine measurement and military-purpose measurement to the daily application fields such as fire monitoring and pollution monitoring.

Ubiquitous computing can be actualized by drawing and suggesting fragility and requirement in security that can happen in ubiquitous environment, and by suggesting the direction of security service centered on safe key distribution, certification and sensor node capture defense technology. In the proposed cryptosystems we use a new security protocol. For the control of key that can be used for the certification of sensor node or encryption of sensed information, the researchers suggest a key control method and security protocol to defend against sensor node capture by applying PKI method to Hash Lock. Drawing security weakness that may occur in the network environment and its requirement and suggesting the direction of security service can realize sensor network technology.

In a nutshell, it measures the target precisely and collects and delivers the safe information.



Fig. 11. Layout sensor and data capture

6. References

- [1] I.F.Akyiliz, W.Su, Y.Sankarasubramaniam and E. Cayirci, "A survey on sensor network", IEEE Communication Magazine, pp 102-114, 2002.
- [2] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Network", Wireless Network Journal(WINE), September 2002.
- [3] W. Du, J. Deng, Y. S. Han and P. Varshney, "A Pairwise Key Pre-distribution Scheme for Wireless Sensor Network", 10th ACM Conference on Computer and Communications Security (CCS), October 2003.
- [4] J. Staddon, S. Miner, and M. Franklin, "Self-Healing Key Distribution with Revocation", Proceedings of 2002 IEEE Symposium on Security and Privacy (S&P2002), May 2002.
- [5] Haiyun Luo, Petros Zefros, Jiejun Kong, Songwu Lu, and Lixia Zhang, "Self-securing Ad Hoc Wireless Networks", 7th IEEE Symposium on Computers and Communication (ISCC '02), July 2002.
- [6] K. Takaragi, M. Usami, R. Imura, R. Itsuki, T. Satoh, "An ultra small individual recognition security chip," Micro, IEEE Nov/Dec 2001 Pages 43 - 49, Volume 21, Issue 6
- [7] ZigBee Alliance Document 03522 : Security Service Specification, December 2004.
- [8] J. Hoffstein, J. Pipher, J. H. Silverman, "NTRU: A Ring-Based Public Key Cryptosystem." Lecture Notes in Computer Science, 1423:267, 1998.



Convergence and Hybrid Information Technologies

Edited by Marius Crisan

ISBN 978-953-307-068-1

Hard cover, 426 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

Starting a journey on the new path of converging information technologies is the aim of the present book. Extended on 27 chapters, the book provides the reader with some leading-edge research results regarding algorithms and information models, software frameworks, multimedia, information security, communication networks, and applications. Information technologies are only at the dawn of a massive transformation and adaptation to the complex demands of the new upcoming information society. It is not possible to achieve a thorough view of the field in one book. Nonetheless, the editor hopes that the book can at least offer the first step into the convergence domain of information technologies, and the reader will find it instructive and stimulating.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Yong-Sik Choi and Seung Ho Shin (2010). A Study on Sensor Node Capture Defence Protocol for Ubiquitous Sensor Network, Convergence and Hybrid Information Technologies, Marius Crisan (Ed.), ISBN: 978-953-307-068-1, InTech, Available from: <http://www.intechopen.com/books/convergence-and-hybrid-information-technologies/a-study-on-sensor-node-capture-defence-protocol-for-ubiquitous-sensor-network>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen