Journal of The Korea Society of Computer and Information
Vol. 18, No. 11, November 2013
www.ksci.re.kr
http://dx.doi.org/10.9708/jksci.2013.18.11.115

무선 센서네트워크에서 경량화 개인별 암호화를 사용한 멀티캐스트 전송기법

박태현*, 김승영*, 권구인*

Lightweight Individual Encryption for Secure Multicast Dissemination over WSNs

Taehyun Park*, Seung Young Kim*, and Gu-In Kwon*

요 약

본 논문에서는 무선 센서네트워크상에서 Lightweight Individual Encryption Multicast 방식으로 그룹키의 사용대신에 Forward Error Correction을 이용한 개인별 인크립션을 사용하여 안전한 데이터 전송을 제안한다. 무선 센서네트워크에서 센서노드 프로그램을 위한 업데이트 방법으로 싱크 노드는 데이터를 다수의 센서노드에게 멀티캐스트 방식으로 전송이 가능하며, 그룹키 인크립션 방식이 가장 보편적인 안전한 데이터 전송을 위한 방식이라 할 수 있다. 이러한 그룹키 방식은 더 강력하고 안전한 데이터 전송을 위하여 멤버의 가입 및 탈퇴시 키를 재 생성하는 re-key 방식이 필요하다. 그러나 이러한 그룹키 방식을 센서네트워크에서 구현하기에는 제한된 컴퓨팅 자원, 저장 공간, 통신 등으로 인한 많은 제약이 존재한다. 또한 개인별 인크립션을 사용하면 각 노드에 대한 개별적 컨트롤은 가능하지만, 데이터 전송을 위한 개인별 인크립션 비용이 많이 발생하는 문제점이 있다. 멀티캐스트 전송 시 개인별 인크립션 방식이 많이 고려되지 않았지만, 보내고자 하는 전체 데이터의 0.16 %만 개인키를 사용하여 각 노드에게 유니캐스트로 안전하게 전송하고, 나머지 99.84%의 데이터는 멀티캐스트를 이용하여 전송함으로써 무선 센서네트워크 성능을 향상시킨다.

▸ Keywords : 멀티캐스트, 보안, 인트립션, 포워드 에러 코렉션, 전진 오류 수정

Abstract

In this paper, we suggest a secure data dissemination by Lightweight Individual Encryption Multicast scheme over wireless sensor networks using the individual encryption method with

•제1저자 : 박태현, 교신저자 : 권구인, 책임저자 : 권구인
•투고일 : 2013. 9. 5, 심사일 : 2013. 9. 25, 게재확정일 : 2013. 10. 15.
* 인하대학교 컴퓨터정보학부(Dept. of Computer and Information Engineering, Inha University)

※ 이 논문은 인하대학교의 지원에 의하여 연구되었음.

Forward Error Correction instead of the group key encryption method. In wireless sensor networks, a sink node disseminates multicast data to the number of sensor nodes to update the up to date software such as network re-programming and here the group key encryption method is the general approach to provide a secure transmission. This group key encryption approach involves re-key management to provide a strong secure content distribution, however it is complicated to provide group key management services in wireless sensor networks due to limited resources of computing, storage, and communication. Although it is possible to control an individual node, the cost problem about individual encryption comes up and the individual encryption method is difficult to apply in multicast data transmission on wireless sensor networks. Therefore we only use 0.16% of individually encrypted packets to securely transmit data with the unicast to every node and the rest 99.84% non-encrypted encoded packets is transmitted with the multicast for network performance.

▸ Keywords : Multicast, encryption, security, Forward Error Correction(FEC).

I. INTRODUCTION

Multicast communication provides an efficient data delivery from a sink to specific group of sensors over wireless sensor networks (WSN). Multicast security is one of the most important security services in WSN [1-6]. Over the years, multicast in WSN has been the topic of many research areas such as multicast routing, reliable multicast [7], secure multicast, and so on. The severely resource-constrained of sensor networks has posed various challenges to support security with very limited battery power supplies, small size of memory, low computation of CPU and bandwidth. Thus it is obviously challenge to apply efficient secure multicast scheme which is designed for high-performance security system into WSNs. Moreover it is very difficult to control sensor nodes individually in data dissemination protocol.

The general approach to provide the security, especially confidentiality, in multicast is using a shared group key to encrypt the data. Other protocol has been proposed to provide mutual authentication based on the random divided session for the security of medical information in Home-Health [29]. Initially the group key is distributed securely to all clients. The group key is maintained and updated regarding to the group membership change or new software update. In a dynamic membership change environment, this group key encryption becomes complicated because a new group member should not able to access old data and the leaving member cannot access new data, and a new group key has to be delivered to all clients securely. Many methods have been proposed for reducing the number of key change and key distribution [8-18]. Most of these group key management solutions encrypt the 100% of data to provide the security.

In these works, multicast properties, group key management, and 100% of data encryption are all tightly coupled together. In other words, all clients will receive the same encrypted data in multicast. Thus all clients must share the same group key to decrypt the received encrypted data. Due to the dynamic group membership change, the group key must be updated and delivered to all clients securely. Researchers have proposed a new approach

of securing content that significantly lowers the costs of security for both the sink and wireless nodes while still maintaining rigorous security guarantees [19]. This approach considered the tradeoff between the benefit from providing the secure content distribution and the cost to provide the security. This approach enables content providers to consider a lightweight security by reducing the number of encryption to 4%. This method is integrated with efficient forward error correcting codes, such as Tornado codes by using the following property: none of the original content can be recovered whenever a key subset of encoded packets is missing. This approach encrypts only these key code-words which are only 4% of all encoded packets. Our goal is to provide a lightweight secure delivery in WSN multicast with minimal overhead and enable a finer-grained control over each node. While the work [19] provides the theoretical base for the lightweight encryption, we propose a practical solution for the secure and controllable multicast delivery. We propose Lightweight Individual Encryption for secure Multicast (LIEM) dissemination over WSNs which leverages the above work and has the following properties.

① Use an individual key to encrypt data instead of the group key,
② Encrypt partial data (only 0.16% of total data),
③ Remove the group key management problem, and
④ Have finer-grained control over each node.

LIEM encrypts 0.16% of data using the individual key and delivers the encrypted data to each node individually through unicast while the rest 99.84% of data are delivered through multicast. Since the encrypted data are only 0.16% of total data, the total number of data through unicast is minimal. The individual encryption removes the issues of re-key management considered in all previous studies [8-18][20]. Since there is no group key to share among the nodes in our approach, there is no need to generate a new group key when there are frequent membership changes due to joining and leaving a group. While the server might want finer-grained control over the node, such personalized service has not been considered carefully due to technical difficulties. Such personalized service may use any individual encryption and individual transmission, but this approach will not get the benefits of multicast [19].

While there have been various studies on multicast, reliability and security are studied separately. Forward error correction (FEC) codes [21-25] are generally used in multicast to provide reliable transmission. We aim at providing a secure multicast transmission over WSN for soft update or so-called over-the-air programming (OAP) protocols. OAP protocols enable all the nodes in a wireless sensor networks to receive software updates from the sink node. These OAP protocols require both secure delivery to sensor nodes and complete reliability since every packet is crucial to the integrity of the program image. Since LIEM applies the property of FEC, there are clear advantages in terms of reliability and security. The additional cost to provide the above benefits comparing with pure FEC transmission is another encoding of 4% data and encryption of 0.16% data. Comparing with previous 100% encryption approaches, LIEM reduces the complexity over re-key management and the overhead on data encryption and decryption. LIEM we propose provides the enough security with minimal cost in WSN multicast and also enables the sink to control the each node with fine-grained manner.

In the next Section, we describe a simple encryption scheme and an architecture about the Lightweight Individual Encryption Multicast. Then experiments of our Lightweight Individual Encryption Multicast scheme are compared to previous work with the CPU times in Section Ⅲ. Finally we conclude in Section Ⅳ.

II. LIGHTWEIGHT INDIVIDUAL ENCRYPTION FOR WSN MULTICAST

2.1. Only 4% encryption of all encoding packets

Erasure-resilient codes are widely used to provide the reliable transmission in multicast. One common property to these codes is that each node or receiver can decode data only after receiving a certain number of encoding packets, which is close to the number of original packets. This property is similar to all-or-nothing transforms [26] and encryption about 4% of transmission packets can provide a subsequent encryption after minor modifications to the codes [19]. Since a node cannot decode without these 4% of encoded packets, any partial information cannot be revealed with the rest 96% of packets. They encrypt the key subset of encoded packets, which are 4% of total packets, and provide enough confidentiality. Figure 1 shows the basic procedure of this approach. They minimized the overhead regarding to encrypt and decrypt data while providing precise security guarantees. This lightweight security method enables the content provider to consider the secure data delivery with the minimal additional cost since the cost for encoding is already paid for reliable multicast transmission.

2.2. Lightweight Individual Encryption for Secure Multicast

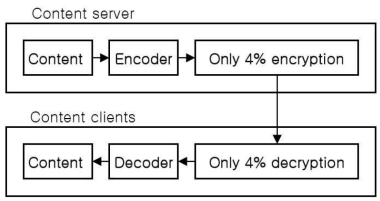

Fig. 1. Only 4% encryption scheme
그림 1. 컨텐츠의 간단한 4% 암호화/복호화 과정

All previous works in multicast security to provide confidentiality use a shared group key. To provide backward and forward secrecy, the group key must be updated based on the group membership change. A new group key must be distributed securely to nodes. While this approach provides a strong secure data delivery, they require a high overhead due to the dynamic membership changes. Content providers may want to employ a light secure data delivery with a minimum cost since the cost for providing a security could be larger than the losses without it. In this section, we propose a new approach for the lightweight multicast security, where the sink uses an individual encryption key per node and the sink distributes the encrypted data to the node using unicast delivery. We describe the procedure of our approach step by step in the following subsections.

2.3 LIEM sink architecture

The sink in our model takes two steps (STEP I, II) for the encoding process, and one step (STEP III) in the encryption process. Figure 2 shows the sink architecture that consists of STEP I, II, and III. STEP I is based on the method described in [19].

An encoder A generates tuned FEC codes which have a key subset of encoded packets. If a key subset of encoded packets is missing, a node cannot perform decoding. Thus original source packets are never recovered by nodes. This key subset of encodes packets, which are marked as a-1, are 4% of total encoded packets. The work in [19] encrypts this key subset of encoded packets to provide the security. Our approach does have one more encoding step with this key subset of packets. These packets, a-1, will be the input for the next encoder in STEP II.

A process of STEP II is similar to that of STEP I. The encoder B generates a key subset of encoded packets from a-1, which are 0.16% of total original packets and marked as b-1. Since b-1 packets are the key subset of encoded packets to generate a-1, without having all b-1 packets none of the a-1

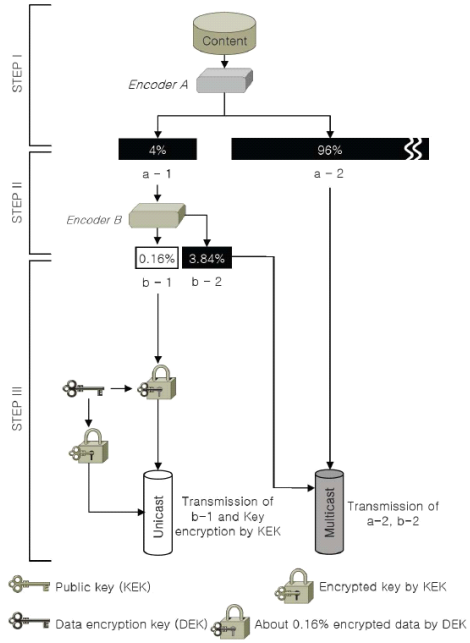

Fig. 2. Lightweight Individual Encryption Multicast architecture of sink
그림 2. 싱크에서의 LIEM architecture

packets can be recovered. Thus if the sink delivers all b-1 packets securely to nodes, the security of all a-1 packets is guaranteed. As we described above, without having all a-1 packets, none of original packets are recovered. Consequently the security of original contents is guaranteed if all b-1 packets are distributed securely. The non-encrypted packets, a-2 and b-2 packets, are distributed through multicast and the encrypted packets are delivered to each node though unicast. The total number of packets in multicast is 99.84% of total transmission packets and the number of packets through unicast is 0.16% of total packets.

Through STEP I and STEP II, the number of packets to encrypt is decreased to 0.16%. By reducing the cost of encryption considerably, it is possible to use an individual encryption in multicast. The individual encryption in multicast has not been studied because the previous approaches use the group key to encrypt and the encrypted data are delivered to all nodes. If the sink uses an individual encryption, the encrypted data must be transmitted through unicast instead of multicast.

STEP III shows the process of individual encryption. The sink encrypts the b-1 packets using an individual data encryption key (DEK), which will then be sent to a node encrypted with the node's public key, used as the key encrypting key (KEK). The content is encrypted using a random DEK per a node. To deliver this individual key to each node securely, the sink encrypts this key using the node's public key. The sender sends the encrypted packet with the encrypted DEK to the node directly through unicast while the other non-encrypted packets will be distributed through multicast. We summarize the delivery of packets as follows:

■ Packets for Unicast transmission
- Encrypted encoding packets
- Encrypted data encryption key, where this key is encrypted by node public key

■ Packets for Multicast transmission
- 96% of encoded packets, which is a-2 packets
- 3.84% of encoded packets, which is b-2 packets

The backward secrecy prevents that a new node should not access the old data. The forward secrecy prevents that a leaving node should not access the data coming from the sink. By making a minor modification from the figure 2, the LIEM can provide the backward and forward secrecy. If the whole content is encrypted once and the encrypted data with an encryption key are distributed to the node, the node can access the whole content even after the node leaves the group. To provide the forward secrecy, a block of content will go through the STEP I, II, and III instead of whole content. If there is no membership change during the delivery of the block, the sink uses the same individual encryption key for the next block. If a node leaves a group, the sink does not perform the encryption for the node in the next block of content encoding time. Thus the node

cannot receive the encrypted data with the encryption key for the next block of data, thus the forward secrecy is guaranteed. In LIEM, the packets are encrypted by an individual symmetric key and the encrypted packets are delivered to the node directly. When a node joins the group, the node cannot recover the previous block of data without receiving the encrypted packets for the previous block. Therefore the new node cannot access the old data in LIEM and the backward secrecy is guaranteed. Many solutions have been proposed to provide the backward and forward secrecy with minimum re-keying cost. The personalized encryption in LIEM gets rid of the complex re-keying schemes and simplifies the multicast security system.

The work in [19] provided the proof of 4% of encoding security. The following is the definition of α-securable. An encoding is called α-securable, $0 <= \alpha <= 1$, if for a randomly selected set of n output symbols, there exists a pi-secure subset of $(1-\alpha)$ output symbols (i.e. the encoding can be secured by securing only a fraction of output symbols). The content delivery scheme in [19] is 0.04-securable. The LIEM inherits the property of 0.04-securable from [19] and please refer to [19] for the mathematical definition and proof of 0.04-securable.

2.4 LIEM sensor node architecture

Figure 3 shows the node architecture that consists of STEP I, II, and III. The STEP I is the process of decryption. The node receives data in two different ways. From unicast delivery, the node receives the encrypted data and the data encryption key used for the data encryption. As we described in figure 2, this key is encrypted by the node's public key. In STEP I, the nodes need to perform 2 types of decryption. First, the node decrypts the encrypted data encryption key (DEK) using the node's private key. After this process, the node acquires the DEK and it uses this key to decrypt the encrypted data. The result of this decryption is marked as b-1 packets, where b-1 packets are the key subset of encoded packets for a-1 packets. STEP II is the first decoding process.

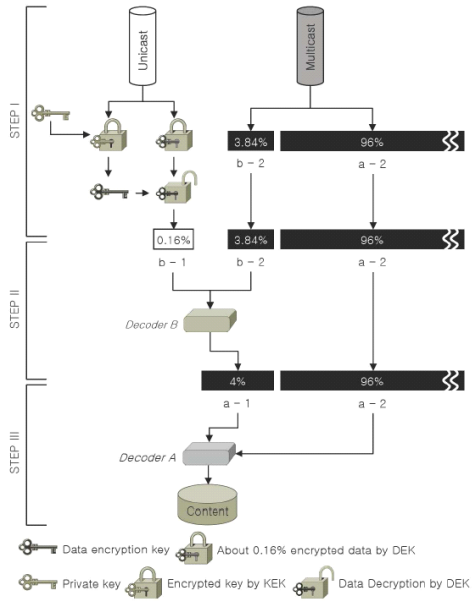

Fig. 3. LIEM sensor node's architecture
그림 3. 센서노드에서의 LIEM architecture

The decoder B takes two parts of encoded packets as an input; b-1 after the decryption and b-2 from multicast delivery. After the first decoding in STEP II, the node generates a-1 packets, where a-1 packets are the key subset of encoded packets for making the original content. The node decodes a-1 and a-2 packets to generate the original content in STEP III. The total number of received packets from unicast delivery is about 0.16% of total transmission packets and the number of packets from multicast delivery is about 99.84%of total packets. The node cannot perform STEP II and STEP III without having the encrypted data and decrypting this data. Since the encrypted packets are delivered through unicast directly from the sink, any non-group member cannot access the data.

2.5 Discussion

In LIEM, the percentage of unicast traffic is 0.16% of total traffic. While this may not considerable, there is an overhead comparing with 100% multicast traffic. There should be contention to acquire the channel between nodes in unicast and it may cause packet collision. The overhead will be worsen as the number of node increases. The work in [27] compared the performance between unicast and multicast in wireless ad hoc network. The throughput drops as the number of receivers increase in unicast, but the multicast drops much more slowly. The multicast algorithm provided in [27] is 7 times better than unicast when the number of receivers is 5, 18 times better when the number of receivers is 30. The performance benefit from multicast may vary depending on wireless multicast protocol and unicast routing protocol, but the performance comparison above may give the brief idea about the overhead in unicast.

In LIEM, b-1 packets are transmitted over unicast and all other 99.84% of traffic are transmitted over multicast. The method in [9] 4% of encrypted packets and other 96% of encoded packets are transmitted over multicast since they use the shared group key. Since 0.16% of encrypted packets in LIEM and 4% of encrypted packets in [9] are crucial for decrypting, 100% of reliability should be guaranteed for these packets. The reliable multicast should consider NACK implosion and duplicated packet reception such as broadcast storm problem [28]. The performance benefit from multicast in [27] does not consider the reliability, thus the performance improvement through multicast may drop since there is overhead to provide reliable multicast in wireless sensor networks.

In our future work, we would like to investigate the performance overhead due to unicast traffic under various unicast routing protocol and multicast protocols.

III. EXPERIMENT

In this section we compare the CPU times of previous work with the LIEM we propose. We extended the experiment results conducted in [19].

Figure 4 shows the results of LIEM with AES cryptography. The meaning of each label is defined as follows.

- Encoding: File is encoded, but not encrypted.
- Encoding *2: File is encoded and 4% of the encoded packets are re-encoded.
- 4% Encrypted: File is encoded and 4% of encoded packets are encrypted.
- 0.16% Encrypted: File is encoded, 4% of them are re-encoded, and then 4% of the re-encoded packets are encrypted. (i.e.LIEM scheme)

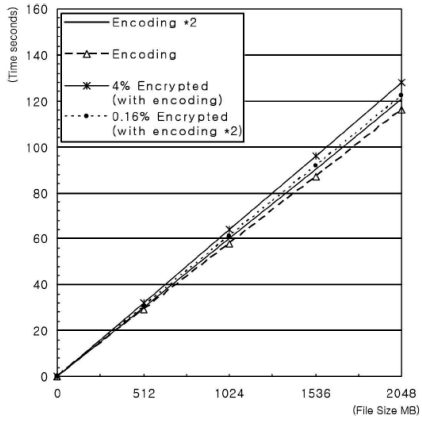

Fig. 4. System evaluation based on CPU time with AES encryption
그림 4. AES 암호 방식에서 파일 크기 증가 시 CPU 시간기반의 시스템 성능평가

Comparison of Encoding*2 with 4% Encrypted show that the encryption process takes much more time than the encoding process. Many comparison results have been shown in [19] and we note the results from [19]. With AES encryption, the encoding time was about twice faster than the

encryption time. 4% encryption after encoding took about half of 100% encryption time.

Figure 4 indicates that the LIEM reduces the whole process time compared to the 4% encryption scheme. The result of figure 4 is conducted with on-line en-coding and online encryption. Many applications use FEC to provide reliable transmission and perform en-coding offline. For these applications, the security cost after encoding is reduced to half in LIEM comparing with 4% scheme. If all encoding processes are done offline, the extra cost for providing security in LIEM is only encryption of 0.16% total packets. We note that the major benefit from LIEM is not only the reduction in the processing time comparing with 4% encryption scheme, but also the reduction in the number of encryption packets. The considerable reduction in the number of encryption packets enables the individual encryption in multicast. Since 4% encryption scheme is a lightweight encryption, the individual encryption might be also applied to this scheme. We plot the encryption time when 4% encryption employs the individual encryption in figure 6 and compare with the LIEM. We use the file size of 128MB and increase the number of nodes. The LIEM reduces the work spent encrypting packets considerably as the number of node increases.

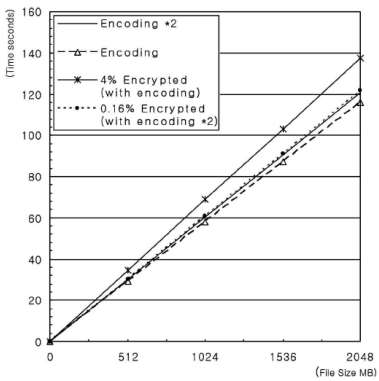

Fig. 5. System evaluation based on CPU time with DES encryption
그림 5. DES 암호 방식에서 파일 크기 증가 시 CPU 시간기반의 시스템 성능평가

In Figure 5, we show the results of LIEM with DES cryptography. As expected, the processing time for encryption is larger than the encryption time with AES. The work in [19] showed that the encoding time is much less than the encryption time with DES. This experiment result also is observed in figure 6. Both Encoding *2 and 4% Encrypted encode the whole con-tent and take 4% of encoded packets. The difference is either encrypting or encoding these 4% of packets. Since the encryption time with DES is larger than the encoding time and the LIEM requires the less number of encryption than 4% scheme, the LIEM reduces the time for providing security to more than half comparing with 4% scheme.

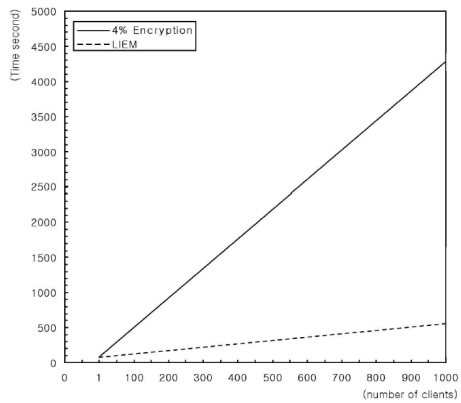

Fig. 6. Comparison of speed if 4% encryption employs individual encryption
그림 6. 클라이언트 증가에 따른 4% 암호방식과 개별 암호방식의 시간 비교

IV. CONCLUSIONS

In this paper, we have proposed the Lightweight Individual Encryption for secure Multicast (LIEM). The number of encrypting packets has been reduced to 0.16% of total packets. This reduction enables the individual encryption in WSN multicast and makes the sink to have finger-grained control over the node. The major distinction from the other studies is that our technique does not use a group key, so

LIEM removes the complex re-keying mechanism proposed in the previous works. Our technique leverage well known properties of certain FEC, so LIEM provides both reliable and secure delivery in multicast over WSN.

References and Notes

〔1〕 D.I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," IEEE Communication Magazine, vol. 40, no. 8, pp. 102-114, Aug. 2002.

〔2〕 H. Chan and A. Perrig, "Security and privacy in sensor networks, Computer," vol. 36, no. 10, pp. 103-105, Oct. 2003.

〔3〕 Q. Yu and C. N. Zhang, "A Secure Multicast Scheme for Wireless Sensor Networks," Mobile, Ubiquitus, and Intelligent Computing (MUSIC), 2012 Third FTRA International Conference on, pp. 158-163, June 2012.

〔4〕 Q. Yu and C. N. Zhang. "A Lightweight Secure Data Transmission Protocol for Resource Constrained Devices," Security and Communication Networks, John Wiley and Sons, Volume 3, Issue 5, pp 362-370, 2010.

〔5〕 Q. Yang and Y. Desmedt, "Secure Communication in Multicast Graphs," Proceedings of 17th International Conference on the Theory and Application of Cryptology and Information Security, pp. 538-555, December 2011.

〔6〕 G. Zeng, B. Wang, Y. Ding, L. Xiao, M. W. Mutka, "Effient multicast algorithms for multichannel wireless mesh networks," IEEE Transactions on Parallel and Distributed Systems 21(1) pp. 86-99, 2010.

〔7〕 G. Kwon and J. Byers, "Roma: Reliable overlay multicast with loosely coupled TCP connections," In Proceedings of IEEE INFOCOM 2004.

〔8〕 R. Varalakshmi,and V. Rhymend Uthariaraj, "A New Secure Multicast Key Distribution Scheme Using Tabulation Method," International Journal of Information Technology and Computer Science, Vol. 4, No. 1, pp. 32-39, February 2012.

〔9〕 A. Ballardie, "Scalable multicast key distribution, Network Working Group," RFC 1949, May 1996.

〔10〕 B. Briscoe, "Marks : Zero side effect multicast key management using arbitrarily revealed key sequences," In 1st International Workshop on Networked Group Communication, Pisa, Italy, November 1999, November 1999.

〔11〕 I. Chang, R. Engel, D. Kandlur, D. Pendarakis, and D. Saha, "Key management for secure internet multicasting boolean function minimization techniques," In Proceeding of IEEE INFOCOM, New York, March 1999.

〔12〕 E. Harder, D. Wallner, and R. Agee, "Key management for multicast: Issues and architectures," RFC 2627, IETF, June 1999.

〔13〕 H. Harney and C. Muckenhirn, "Group key management protocol (gkmp) architecture," Request for Comments (Experimental) 2094, Internet Engineering Task Force, July 1997.

〔14〕 S. Mittra, "Iolus: a framework for scalable secure multicasting," In Proceedings of ACM SIGCOMM Computer Communication Review, vol.27, no.4, pp.277-288, 1977.

〔15〕 M. Naor, D. Naor, and L. Lotspiech, "Revocation and tracing schemes for stateless receivers," In Proceedings of Crypto 2001, 2001.

〔16〕 C. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," In Proceedings of SIGCOMM, September 1998.

〔17〕 A.P. Singh, S. M. Potey, F. A. Barbhuiya and S. Nandi, "A Scalable and Secure Key Distribution Mechanism for Multicast Networks," Advances in Computing and Communications (ICACC), 2012 International Conference on, pp. 211-214, Aug. 2012.

〔18〕 K. Gomathi and B. Parvathavarthini, "An efficient cluster based key management scheme for manet with authentication," In Trendz in

Information Sciences Computing (TISC), 2010, pp. 202-205, Dec. 2010.
〔19〕 J. Byers, J. Considine, G. Itkis, M. Cheng, and A. Yeung, "Securing bulk content almost for free," Computer Communications, vol.29, no.3, pp. 280-290, February 2006.
〔20〕 D. Balenson, D. McGrew, and A. "Sherman, Key management for large dynamic groups: One-way function trees and amortized initialization," Draft-balenson-groupkeymgmt-oft-00.txt, IETF, Feb. 1999.
〔21〕 J. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," IEEE Journal on Selected Areas in Communications, vol.20, no.8, pp.1528-1540, 2002.
〔22〕 M. Luby, "LT codes," In Proceedings of 43rd symposium on Foundations of Computer Science, November 2001.
〔23〕 M. Luby, M. Mitzenmacher, "A. Shokrollahi, and D. Spielman, Efficient erasure correction codes," IEEE Transactions on Information Theory, vol.47, no.2, 2001.
〔24〕 P. Maymounkov and D. Mazieres, "Relateless codes and big downloads," In Proceedings of 2nd International Workshop on Peer-to-Peer Systems, February 2003.
〔25〕 Q. Shuang, G. Feng and Y. Zhang, "Cooperative Communications for Reliable Data Transport with Fountain Codes," Journal of Communications 5, no. 4, pp. 340-347, 2010.
〔26〕 R. L. Rivest, "All-or-nothing encryption," In Proceedings Fast Software Encryption, pp. 210-218, 1997.
〔27〕 R. Bhatia and Li Erran Li. Characterizing achievable multicast rates in multi-hop wireless networks. Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing. ACM, 2005.
〔28〕 S. Ni, Y. Tseng, Y. Chen, and J. Sheu. "The broadcast storm problem in a mobile ad hoc network". In Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99), pp.151-162.
〔29〕 HC. Lim, TH. Park, and GI. Kwon, " Mutual Authentication Protocol based on the Random Divided Session for the Security of Medical Information in Home-Health," Journal of the Korea Society of Computer and Information, 17(10):79-88, 2012.

저 자 소 개

박 태 현

2003: 홍익대학교 컴퓨터공학과 공학사.
2005: 울산대학교 컴퓨터정보공학과 공학석사.
현 재: 인하대학교 컴퓨터정보공학과 공학 박사과정
관심분야: 센서네트워크, 무선 에드혹 네트워크, 멀티홉 무선 센서 네트워크
Email : th_park@naver.com

김 승 영

1998: 인하대학교 전기공학과 공학사.
2002: 인하대학교 컴퓨터정보공학과 공학석사.
현 재: 인하대학교 컴퓨터정보공학과 공학 박사과정
관심분야: 센서네트워크, 멀티홉 무선 센서 네트워크
Email : seankim811@gmail.com

권 구 인

1995: 인하대학교 컴퓨터공학과 공학사.
1998: City University of NewYork 컴퓨터공학과 공학석사.
2005: Boston University Computer Science 공학박사.
현 재: 인하대학교 컴퓨터정보공학부 교수
관심분야: Software Defined Network, 센서네트워크
Email : gikwon@inha.ac.kr