http://dx.doi.org/10.7840/kics.2014.39C.3.223

멀티 홉 Unattended WSN에서 가변 키 슬롯 기반 μTESLA의 운영

최 진 춘◆, 강 전 일*, 양 대 헌**, 이 경 희°

Operating μTESLA based on Variable Key-Slot in Multi-Hop Unattended WSN

JinChun Choi◆, Jeonil Kang*, DaeHun Nyang**, KyungHee Lee°

요 약

무선 센서 네트워크에서 브로드캐스트 인증 방법 중 하나인 μTESLA는 안전한 센서 네트워크의 운영을 위해 BS(base station)로부터 센서 노드로 전달하는 브로드캐스트 메시지의 인증을 가능하게 한다. 하지만 UWSN 환경에서 매우 짧은 키 슬롯 값을 갖는 μTESLA를 사용할 경우, 센서 노드들은 공개된 비밀키 검증을 위하여 많은 양의 해시값을 계산하여야 한다. 반대로, 센서 노드의 해시 연산량을 줄이기 위하여 μTESLA의 키 슬롯 길이를 지나치게 길게 잡는다면, BS는 키를 공개하기 위하여 그 지나치게 긴 슬롯 길이만큼의 시간을 기다려야 한다. 이 논문에서는 이러한 점을 개선하기 위하여 가변 길이의 키 슬롯을 갖는 μTESLA에 대하여 제안하였으며, 이를 모의 실험한 결과를 보임으로써, 우리의 기법이 센서 노드의 응답 시간을 향상시키고 센서 노드에서 수행되는 해시 연산의 수를 효과적으로 줄였음을 증명한다.

Key Words : WSN Security, μTESLA, Broadcast Authentication, NS-2

ABSTRACT

As a broadcast message authentication method in wireless sensor networks, μTESLA enables sensor nodes efficiently authenticate message from base station (BS). However, if we use μTESLA that has very short length of key slot in unattended wireless sensor network (UWSN), sensors may calculate a huge amount of hashs at once in order to verify the revealed secret key. In contrast, if we set the length of μTESLA's key slot too long in order to reduce the amount of hashs to calculate, BS should wait out the long slot time to release key. In this paper, we suggest variable key slot μTESLA in order to mitigate the problem. As showing experiment results, we prove that our suggestion improve sensor node's response time and decrease of number of hash function calculation.

※ 본 연구는 인하대학교의 지원에 의하여 수행되었습니다.
◆ First Author : 인하대학교 컴퓨터정보공학과, noodlejin@isrl.kr, 학생회원
° Corresponding Author : 수원대학교 전기공학과 부교수, khlee@suwon.ac.kr, 정회원
* 인하대학교 정보통신공학과, dreamx@isrl.kr, 학생회원
** 인하대학교 컴퓨터정보공학과 부교수, nyang@inha.ac.kr, 정회원
논문번호 : KICS2013-09-407, Received September 15, 2013; Reviewed November 15, 2013; Accepted March 3, 2014

Ⅰ. 서 론

무선 센서 네트워크는 마이크로프로세서와 통신 모듈, 센서 모듈을 내장하고 있는 센서들로 구성된 네트워크를 의미한다. 무선 센서 네트워크는 그 특성상 저전력으로 운영되며 한 번 배치되면 이후 추가적인 전력의 공급이 쉽지 않기 때문에 네트워크의 수명을 길게 유지하기 위한 노력이 필요하다. 무선 센서 네트워크는 건물 또는 대중교통 등에서 사람과 통로에 대한 정보를 수집할 수도 있으며, 또한 전쟁 지역이나 국경지대에서 사람 및 기타 장비의 움직임을 감시하기 위해 무선 센서 네트워크를 구성할 수도 있다[1-3].

무선 센서 네트워크를 운영할 때 BS(Base Station)로부터 센서 노드들에게 명령을 전달하거나 데이터를 전달할 때 사용되는 브로드캐스트 메시지는 그 특징 때문에 악의적인 사용자가 BS를 위장하여 센서 노드들의 수명을 단축시키기 위해 센서 노드들에게 무의미한 메시지를 전달하거나, 데이터 취합 명령을 보내서 센서 노드가 수집한 데이터를 탈취해갈 수 있다[4]. 이러한 공격에 대응하기 위해서는 센서 노드가 BS가 보낸 것으로 인증되는 메시지에만 반응하도록 하는 메시지 인증 절차가 필요하다. 일반적으로 사용되는 메시지 인증 방법에는 공개키 암호 알고리즘을 이용한 전자 서명기법이 있는데, 전자 서명 기법에 사용되는 공개키 알고리즘은 센서 네트워크에서 사용되는 센서 노드들이 수행하기에는 에너지 소모가 크며 통신에 사용되는 오버헤드 역시 커서 사용하기에 문제가 있다.

이 논문에서 논의하고자 하는 바는 다음과 같다. 기존의 μTESLA를 좀 더 실용적인 환경에서 실험하기 위해 멀티 홉 환경을 고려하였으며, BS가 네트워크에 항상 참여하여 동작하는 방식이 아니라 주기적으로 네트워크를 방문하여 센서 노드들이 수집한 데이터를 수거해 가는 시나리오를 적용하였다. 또한 기존의 고정된 키 슬롯을 사용한 μTESLA와 가변적인 키 슬롯을 이용한 좀 더 현실적인 환경에 맞춰 구현하고, 비교 실험을 수행하였다.

이 논문의 구성은 다음과 같다. 2장에서는 기존의 μTESLA의 운영에 대한 관련 연구를 알아보며 3장에서는 가변 키 슬롯을 갖는 μTESLA에 대한 내용을 서술하였다. 4장에서는 기존의 고정 키 슬롯을 이용한 μTESLA와 가변 키 슬롯을 이용한 μTESLA를 각각 NS-2를 이용하여 시뮬레이션을 수행한 내용과 실험 결과로부터 알 수 있는 사실들을 서술하였다. 5장에서는 실험 결과에 대한 토의를 하며 6장에서는 결론을 맺는다.

Ⅱ. 관련 연구

2.1 μTESLA

기존에 브로드캐스트 메시지를 인증하기 위한 방법으로는 TESLA[5] 등의 방법이 있으나, 이러한 방법들은 인증에 사용되는 전자 서명 등의 과정이 포함되어 있어 무선 센서 네트워크에서 수행하기에는 많은 연산과 그에 따른 에너지 소모를 하게 된다. 이러한 문제를 해결하고, 무선 센서 네트워크에서 브로드캐스트 메시지를 인증하기 위한 방법으로 μTESLA가 있다[1]. μTESLA는 메시지를 인증하는데 사용한 키를 지연시켜 공개함으로써 공개키 암호와 같은 비대칭성을 제공할 수 있다. μTESLA에서 BS와 센서 노드는 BS가 키 체인을 생성할 때 마지막으로 생성된 값 K_0를 공유하고 있다. BS가 브로드캐스트 메시지를 보낼 때 일방향 함수를 이용하여 미리 생성한 해시 키 체인을 이용하여 메시지에 대한 MAC을 생성하여 보낸다. 센서 노드들은 BS로부터 받은 메시지를 바로 확인하지 못하기 때문에 메시지를 버퍼에 저장하며 이후에 BS가 MAC을 검증하기 위한 키를 센서 노드들에게 전송하면 센서 노드들은 해당 키를 이용하여 버퍼에 저장해 둔 메시지들을 인증할 수 있다. 이렇게 키를 지연시켜 공개하는 방식을 사용하기 위해서는 네트워크가 운영되는 시간을 키 슬롯(key slot)이라고 하는 일정한 단위로 나누고 각 키 슬롯마다 하나의 키를 사용해야 한다. 예를 들어 BS가 센서 노드들에게 메시지를 포함한 패킷을 전송하고 하나의 슬롯이 지난 뒤에 키를 센서 노드에게 전송한다. 센서 노드들은 키를 해시 함수에 입력하여 반복 수행하면 K_0 값이 나오는 것을 확인할 수 있으므로 올바른 BS로부터 온 키 값임을 확인할 수 있다. 또한 이 과정에서 계산한 키 값을 이용하여 이전에 받은 메시지를 인증할 수 있다.

이러한 기존의 μTESLA는 무선 센서 네트워크에서 BS와 모든 센서 노드가 1홉 거리에 있는 것을 가정하여 동작하도록 작성되었는데 이는 일반적으로 운영되는 무선 센서 네트워크 환경으로 보기 힘들다. 따라서 센서 노드들이 여러 홉 거리에 떨어져서 운영되는 멀티 홉 환경에서 μTESLA를 운영할 경우에는 문제가 발생할 수 있다. μTESLA의 특성상 키 슬롯의 길이와 센서 노드가 수행해야 하는 해시 함수의 횟수 사이에 트레이드-오프 관계가 발생하게 되는데 이는 μTESLA의 키 슬롯 길이를 짧게 할 경우에는 BS가 네트워크에 방문하여 브로드캐스트 메시지를 모든 센

서 노드에게 전달하는 경우에 센서 노드들이 BS로부터 받은 메시지를 인증하여 BS가 원하는 동작을 하기까지 걸리는 응답 시간을 짧게 가져갈 수 있다. 하지만 이렇게 키 슬롯을 짧게 하는 경우에는 센서 노드가 키를 검증하기 위해 더 많은 양의 해시 함수의 수행이 필요하게 된다.

또는 해시 함수의 수행 횟수를 줄여 네트워크의 수명을 늘리기 위해서는 키 슬롯의 길이를 길게 사용하여야 하는데 이렇게 운영한다면 BS가 방문하여 메시지를 센서 노드들에게 전달한 뒤 센서 노드들이 메시지를 인증하여 BS가 원하는 동작을 수행하는 데 까지 걸리는 시간이 길어지게 되고 BS가 네트워크에 머무르는 시간이 지연되게 된다. 이것은 μTESLA 특성상 키 슬롯 당 하나의 키를 사용해야 하며 센서 노드가 BS의 메시지를 인증하기 위해서는 최소한 하나의 키 슬롯이 지난 뒤에 BS가 키를 전송해야 하기 때문이다.

2.2 μTESLA의 개선

D. Liu 등은 멀티-레벨 μTESLA를 제안하여 μTESLA의 키 체인을 여러 단계로 구성하여 키 체인의 길이가 길어짐에 따라 해시 함수의 연산의 부담이 생기는 단점을 해결하고자 하였다[6]. 상위 단계에서는 긴 키 슬롯을 이용하고, 하위 단계에서는 짧은 키 슬롯을 이용하여 키 검증에 소요되는 시간을 줄였다.

그리고 임채훈 등의 연구에서는 인증 지연 없는 멀티-레벨 μTESLA의 구성을 통해 μTESLA의 운영에 있어서 길어지는 키 체인의 길이를 해결하면서 센서 노드들이 BS의 메시지를 인증하는 데 지연시간이 없도록 하는 방법을 제안하였다[7].

Yuan Wang 등은 기존의 μTESLA에서 사용되는 MAC과 암호키가 고정되어있는 점, 네트워크를 확장하는 경우에 발생하는 문제를 해결하기 위해 기존의 프로토콜에 해시 함수와 공유키를 추가하여 키를 관리하는 방법을 제안하였다. 이 경우 약간의 에너지 소모가 증가하게 되지만, DoS 공격의 가능성을 낮출 수 있다[3].

Ruiying Du 등은 기존 μTESLA의 프로토콜에서 BS가 키를 브로드캐스트 하는 역할과 메시지를 브로드캐스트하는 역할을 분할하였는데 BS 대신 키 서버 역할을 하는 TCP(Trusted Computing Platform)를 사용하여 키를 브로드캐스트하도록 하였다. 이를 통해 센서 노드가 가진 키가 무효화 되더라도 키 서버와 다시 키를 동기화를 할 수 있다. 다만 이 경우 추가적인 키 서버 역할을 하는 TCP가 필요하다는 점과 추가적인 에너지 소모가 생길 수 있다[8].

Yun Zhou 등은 Batch-based 브로드캐스트 인증 기법을 제안하였는데, 이는 기존 μTESLA에서 시간 동기화를 이용하여 동작하기 때문에 발생하는 브로드캐스트 횟수의 제한과 키 체인을 관리해야 하는 문제들을 보완할 수 있다. 이 브로드캐스트 인증 기법에서는 브로드캐스트 시 인덱스, 메시지, 다음 패킷 전송에 사용할 키를 해시 함수로 연산한 값, MAC을 Batch 패킷으로 묶어서 전송하고, 이후에 키를 노출시키는 방법을 사용하여 시간 동기화 없이 메시지 인증을 수행하였다[9].

위와 같은 다양한 연구를 통해 μTESLA가 갖는 단점을 보완하고 있으나 이들 대부분은 μTESLA에서 사용하는 키 슬롯에 고정된 값을 사용하고 있다.

2.3 Unattended Wireless Sensor Network와 BS의 방문

UWSN(Unattended Wireless Sensor Network)은 센서 노드들이 특정한 공간에 배치되어 주변 데이터를 수집하며 수집한 데이터를 수거하기 위해 BS가 센서 노드들을 방문하는 방식으로 동작한다. 이는 센서 노드들과 BS가 같은 장소에 위치하여 동작하는 WSN(Wireless Sensor Network)과는 BS의 동작에 있어서 차이가 있다. 이러한 UWSN이 사용되는 환경은 다음과 같이 생각해 볼 수 있다. 사람이 쉽게 다가갈 수 없는 장소나 사람이 상주하여 데이터 수집을 하기 힘든 장소에 센서 노드를 배치해두고 주기적으로 네트워크를 방문하는 BS를 이용하여 센서 노드가 수집한 데이터를 수거해 가는 환경이나, 군사적으로 지속적인 감시가 요구되는 장소에서 센서 노드들을 배치해 두어 비정상적인 행동이나 현상이 있었는지를 분석하는 환경 등을 고려할 수 있다.

또한 매우 넓은 구역에서 기상 정보를 측정하는 경우, 농업에서 토양의 상태를 관리하는 경우 등에는 센서 노드를 Sparse하게 배치하게 되는데, 이러한 경우 네트워크 내의 센서 노드 밀도가 낮아지게 되며 센서 노드 사이의 통신이 힘들게 된다. Mario Di Francesco 등은 이러한 경우에 MDC(Mobile Data Collector)를 두어 센서 노드들이 측정한 데이터를 수거하는 데에 사용하였다. 이는 UWSN에서 네트워크를 방문하는 BS 역할과 유사하다고 볼 수 있다. 이들의 논문에서 수행한 실험에서는 MDC의 이동 패턴을 3가지로 분류하였다. 이 중 Gaussian Mobility는 MDC가 네트워크에 주어진 평균과 분산을 따르는 정규분포에 따라 정기적으로 방문하는 것을 가정하고 있다. 이러한 방문은 완벽한 주기를 따르지는 않지만 예측 가능하다[10].

그리고 장기적인 치료를 필요로 하는 환자를 대상으로 하여 센서를 환자 몸에 장착한 후, 센서가 환자의 건강 관련 데이터를 수집하게 한다. 이후 환자가 병원이나 모니터링 BS를 방문하는 때에 병원에서 데이터를 수거해갈 수 있다. 또는 병원에 입원중인 환자들에게 센서를 장착한 뒤, 담당 의사가 정기적으로 Mobile Sink를 가지고 돌아다니며 데이터를 수거하고 확인하여 환자들의 검진에 사용할 수 있다[11].

Ⅲ. 가변 키 슬롯을 갖는 μTESLA

이 논문에서는 가변 키 슬롯을 이용한 μTESLA를 제안하여 기존의 μTESLA에서 고정으로 사용하는 키 슬롯 때문에 발생하는 센서 노드의 응답 시간과 수행하는 해시 함수의 횟수 사이의 트레이드-오프 문제를 완화시킬 수 있다. 만약 BS가 UWSN 환경처럼 주기적으로 방문한다면, BS의 방문주기를 통계적으로 분석하여 통계 데이터에 기반을 둔 BS 방문 확률을 계산할 수 있을 것이다. 이를 이용하여 BS의 방문 확률이 높은 시간대에는 짧은 키 슬롯을 이용하고 이와 반대로 BS의 방문 확률이 낮은 시간대에는 긴 키 슬롯을 적용할 수 있다. BS의 방문 확률에 따라 가변 키 슬롯을 이용하여 μTESLA를 운영한다면 BS가 방문할 확률이 높은 시간에는 짧은 키 슬롯을 이용하기 때문에 BS의 메시지를 받아서 빠르게 브로드캐스트 메시지를 인증하고, BS가 요청하는 작업을 수행할 수 있다. 또한 BS의 방문 확률이 낮은 시간에는 긴 키 슬롯을 사용함으로써 이후에 BS가 방문하여 전송하는 메시지를 인증하는 데 수행해야 하는 해시 함수의 횟수를 줄일 수 있고, 따라서 무선 센서 네트워크에서 센서 노드가 BS의 메시지를 인증하는 데 사용하는 에너지의 양을 줄일 수 있다.

그러나 이 때 사용하는 키 슬롯의 길이를 너무 길거나 짧게 사용하는 경우에는 문제가 생길 수 있다. 키 슬롯이 너무 긴 경우에는 BS가 낮은 확률로 평소와 다른 시간에 방문하는 경우 센서 노드는 BS의 브로드캐스트 메시지를 받고 나서 그 메시지에 대한 MAC을 검증하는 키를 매우 긴 키 슬롯 시간 이후에 받고 인증하여야 하며, 다음 메시지 또한 긴 키 슬롯 시간 이후에 전달하여야 하기 때문에 네트워크의 전체적인 응답 시간이 느려지게 된다.

반대로 키 슬롯의 길이가 너무 짧은 경우에는 센서 노드가 키를 인증하기 위해 센서 노드의 마이크로프로세서에서 해시 함수를 반복적으로 수행하는 시간이 주어진 키 슬롯의 길이보다 길어질 수 있다. 이러한 경우가 발생한다면 센서 노드는 BS의 브로드캐스트 메시지를 인증하지 못하게 되며 BS에서 요청하는 작업을 정상적으로 수행할 수 없게 된다.

```
1  while STATE = IDLE do
2    MSG ← Receive()
3    if MSG != NULL
4      Flood(MSG)
5      /* message */
6      if type(MSG) = MESSAGE
7        PushTaskQ(MSG)
8      else if type(MSG) = KEY /* key */
9        GapSlot ← CurrentSlot - LastTrustedSlot
10       K ← MSG
11       for i = 1 to GapSlot
12         K ← Hash(K)
13       end for
14       if K = TK /* Trusted Key */
15         TK ← K
16         LastTrustedSlot ← CurrentSlot
17         while MSG ← PopTaskQ() do
18           (Data, MAC) ← MSG
19           if HMAC(Data, K) = MAC
20             /* do process tasks */
21             ProcessTask(Data)
22           end if
23         end while
24       end if
25     end if
26   end if
27 end while
```

그림 1. 센서 노드의 동작 알고리즘
Fig 1. Operation algorithm of sensor nodes

만약 가변적인 키 슬롯을 사용하는 환경이 UWSN에서 BS의 방문 확률이 불규칙하여 통계적으로 예측하기 어렵다면 키 슬롯의 길이를 BS의 방문 확률에 기반하여 결정하기 힘들기 때문에 가변 키 슬롯을 운영하여 얻는 장점이 줄어들게 된다. 따라서 BS의 방문 패턴에 따라 적당한 최소 길이와 최대 길이의 설정이 무엇보다 중요하다고 할 수 있다.

Ⅳ. 성능 평가 실험

4.1 실험 환경

이 논문에서 수행된 실험은 NS-2(버전 2.35)를 사용하였고 실험에 사용된 센서 모델은 MICAz[12]로 가정하였으며, 해당 센서의 정보를 이용하여 시뮬레이션에 사용하였다. 또한 μTESLA의 구현에서 사용된 해시 함수는 SHA-1을 사용한다고 가정하였다. 각 실험에 사용하는 매개변수는 표 1과 같다.

노드 배치는 격자 모양 정렬로 하였으며, 센서 노드

표 1. 실험에 사용할 매개변수
Table 1. Simulation Parameters

Param.	Description	Value	
		Ours	mTesla
	sensor node	MICAz	
n	the number of sensor nodes in network	100	
	network topology	100m × 100m	
	hash function	SHA-1	
	time to compute hash for one block (512-bit)	0.008s[7]	
	period of hibernation	3s (idle), 5~7s (hibernation)	
v	variant of normal distribution used to calculate BS visiting probability	0.05. 0.1, 0.6	-
$\ell_{\mathrm{slot},i,v}$	length of i-th unit time according to v	10 ~ 300	10, 82, 100, 138, 300
T	unit time (time period to change key slot length)	1800s	-
$p_{\mathrm{slot},i,v}$	BS visiting probability in i-th unit time according to v	0.005 ~	-

는 에너지 효율을 위해 동면(hibernation)과 휴식(idle) 상태를 반복하도록 하였다. 각 절에서는 응답 시간의 차이와 해시 함수의 수행 횟수 차이를 보였다. 응답 시간은 BS가 정해진 키 슬롯이 시작되는 시간에 메시지를 보낸 시간부터 센서 노드가 메시지를 받은 후 그에 해당하는 키를 받은 후 인증을 마칠 때 까지 걸린 시간을 측정하였다. 해시 함수의 수행 횟수는 BS로부터 받은 메시지에 대한 키를 인증하기 위해 수행하는 해시 함수의 횟수를 측정하였다. 각각의 경우에 대해서 50회의 반복 실험을 수행하였다.

MICAz 센서 노드에서 SHA-1을 수행하기 위한 시간은 다음과 같은 공식에 따라 구할 수 있으며[13], MICAz 센서 노드에 해당하는 각각의 상수는 <α =60980, β=458660, block size(bits)=512>와 같다.

$$t_{exec} = \frac{a + \beta \times [t/b]}{Freq \times Bus} \tag{1}$$

여기서 t는 해시 함수로 입력 받는 문자의 길이이고, b는 해당하는 해시 함수의 블록의 크기이다. $Freq$는 MICAz 프로세서의 클럭 수이고, Bus는 프로세서의 버스 크기이다.

가변 키 슬롯을 사용하는 경우 BS의 방문 확률 $p_{\mathrm{slot},i,v}$에 따른 키 슬롯의 길이 $\ell_{\mathrm{slot},i,v}$를 계산하기 위해 다음과 같이 직선의 방정식을 사용하였다.

$$p_{\mathrm{slot},i,v} = 0.5 + 0.76 \int_{0.5\times i}^{0.5\times(i+1)} N(12, v)\, dx \tag{2}$$

$$p_{\max,v} = \max\left(\{p_{\mathrm{slot},i,v}\}_{i\in\{0,1,\cdots,47\}}\right) \tag{3}$$

$$\ell_{\mathrm{slot},i,v} = (p_{\mathrm{slot},i,v} - p_{\max}) \frac{\ell_{\min} - \ell_{\max}}{p_{\min} - p_{\max,v}} + \ell_{\min} \tag{4}$$

여기서 $\ell_{\min} = 10$, $\ell_{\max} = 300$, $p_{\min} = 0.005$ 이다. $N(12, v)$는 평균이 12이고, 분산이 v인 정규분포를 의미한다. 실험에 사용된 분산 $v = \{0.05, 0.1, 0.6\}$에 대하여 100만개 샘플링을 통해 구한 $p_{\mathrm{slot},i,v}$의 분포는 그림 2와 같으며, $p_{\max,v}$는 각 {0.3754, 0.34181 0.1880}와 같았다.

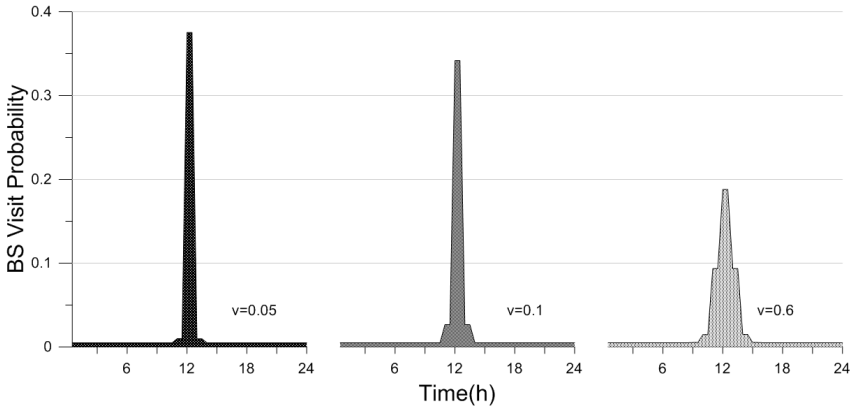


그림 2. 분산 v에 따른 BS의 방문 확률 p_{slot} (30분 기준)
Fig. 2. BS visiting probability p_{slot} by variant v (in 30 minutes)

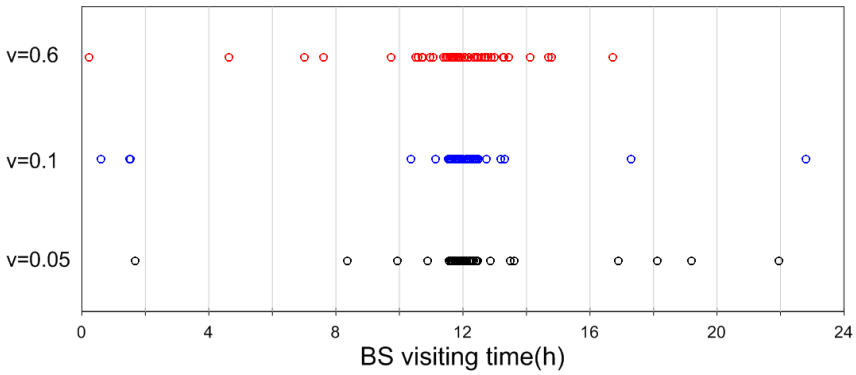


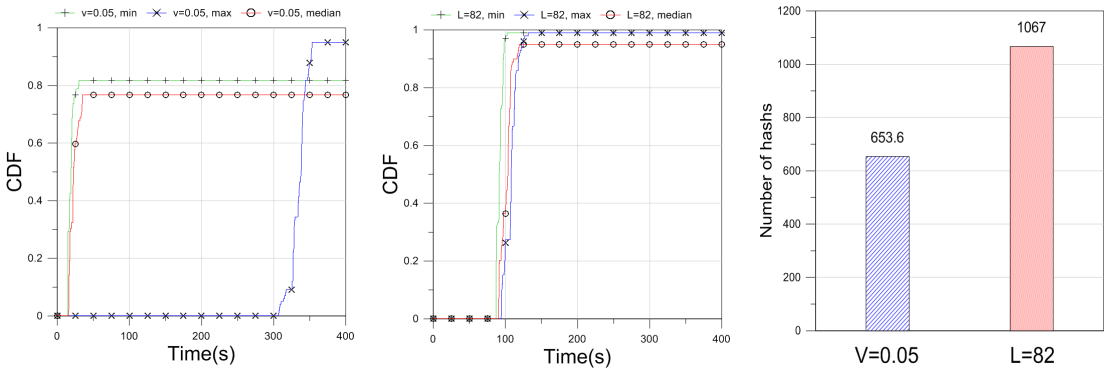
그림 3. 분산 v에 따른 BS의 방문 시간 분포 (50회 기준)
Fig. 3. BS visiting time distribution by variant v (50 times)

또한 이 장에서 실험을 수행할 때에는 계산한 수식에 따라 BS의 방문 확률을 결정하였으며, 그에 따른 BS의 방문 시간 분포는 그림 3과 같다. BS의 방문 확률이 평균이 12이고 분산 $v=\{0.05, 0.1, 0.6\}$인 정규분포를 따르는 경우에 50회의 반복 수행을 하였을 때 BS의 방문 시간은 12시를 중심으로 분포되어 있음을 확인 할 수 있다.

4.2 평균 기대 키 슬롯의 시간이 유사할 때 수행하는 해시 수행 횟수의 차이

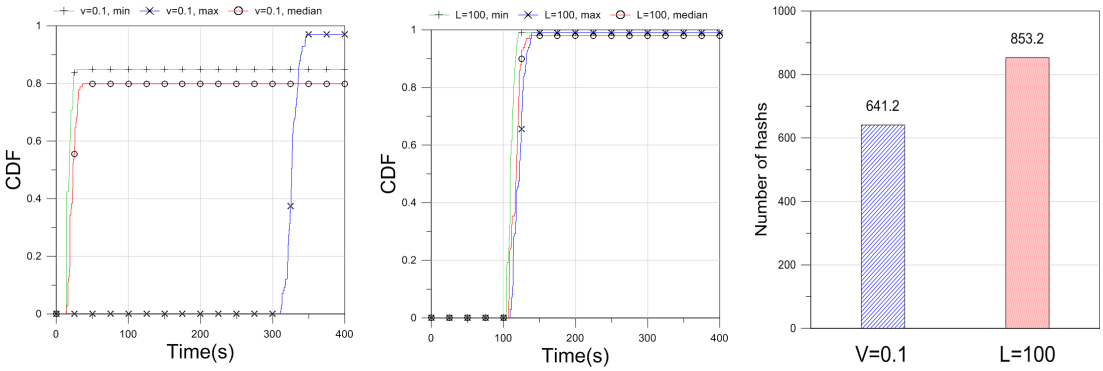
이 절에서는 고정 키 슬롯과 가변 키 슬롯을 이용한 μTESLA에서 평균 기대 키 슬롯의 시간이 유사한 경우에 고정 키 슬롯과 가변 키 슬롯을 이용한 μTESLA에서 수행하는 해시 함수의 차이를 보인다. 여기서 평균 기대 키 슬롯은 BS가 24시간 동안 BS의 방문 확률 p_{slot}에 근거하여 도착 하였을 때 기대할 수 있는 평균 키 슬롯의 길이를 말한다. 가변 키 슬롯의 경우 v가 0.05인 경우에 평균 기대 키 슬롯의 길이는 82.13이며, v가 0.1인 경우는 99.77, v가 0.6인 경우 137.56이었다. 이에 따른 고정 키 슬롯은 길이를 각각 82, 100, 138로 사용하여 실험을 수행하였다. 그림 4(a), 그림 5(a), 그림 6(a)에서는 가변 키 슬롯의 경우 50번의 수행 중 반응 시간의 평균이 가장 작을 경우와 클 경우, 그리고 중간값의 CDF를 보여준다. 그림 4(b), 그림 5(b), 그림 6(b)의 경우는 고정 키 슬롯의 경우를 보여준다. 그림 4(c), 그림 5(c), 그림 6(c)는 앞선 두 경우의 평균 해시 함수 수행 횟수의 비교를 보인다.

가변 키 슬롯의 경우, v가 0.05인 경우 약 50%의 노드가 29.92초 안에 정상적인 메시지 인증을 수행할 수 있었지만, 메시지 인증을 수행하지 못한 노드를 제외한 경우에는 평균 138.68초가 걸렸다. 응답율의 경



(a) v=0.05 in variable key slot μTESLA
(b) L=82 in fixed key slot μTESLA
(c) number of hashs in variable and fixed key slot μTESLA

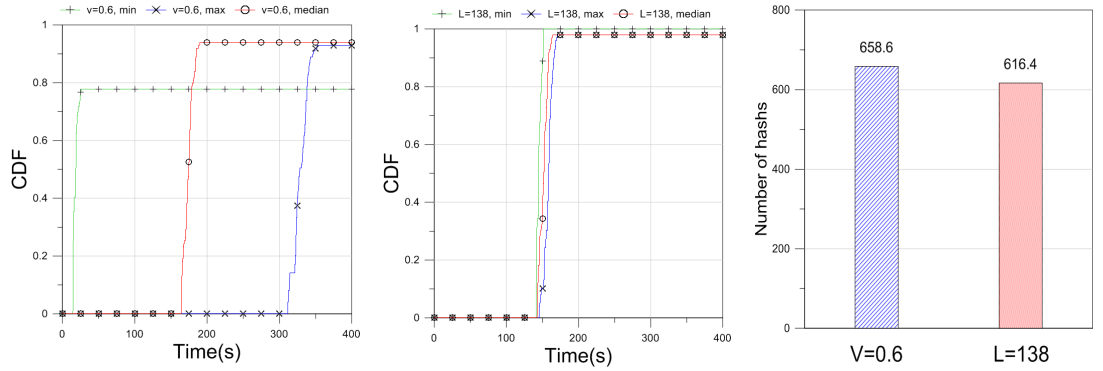
그림 4. 평균 기대 키 슬롯 시간이 유사한 경우 해시 함수 수행 횟수의 비교(가변 키 슬롯 v=0.05와 고정 키 슬롯 L=82)
Fig. 4. Comparison of the number of hashs on similar expecting key slot time on average (variable key slot v=0.05 and fixed key slot L=82)



(a) v=0.1 in variable key slot μTESLA (b) L=100 in fixed key slot μTESLA (c) number of hashs in variable and fixed key slot μTESLA

그림 5. 평균 기대 키 슬롯 시간이 유사한 경우 해시 함수 수행 횟수의 비교 (가변 키 슬롯 v=0.1와 고정 키 슬롯 L=100)

Fig. 5. Comparison of the number of hashs on similar expecting key slot time on average (variable key slot v=0.1 and fixed key slot L=100)



(a) v=0.6 in variable key slot μTESLA (b) L=138 in fixed key slot μTESLA (c) number of hashs in variable and fixed key slot μTESLA

그림 6. 평균 기대 키 슬롯 시간이 유사한 경우 해시 함수 수행 횟수의 비교 (가변 키 슬롯 v=0.6와 고정 키 슬롯 L=138)

Fig. 6. Comparison of the number of hashs on similar expecting key slot time on average (variable key slot v=0.6 and fixed key slot L=138)

우 86.09%으로 약 13.91%의 노드는 BS의 메시지에 정상적으로 응답하지 못했다. 중간값의 경우 반응 시간의 평균이 가장 작을 경우와 거의 비슷한 반응 시간과 반응율을 보임을 알 수 있다. 고정 슬롯의 길이가 82인 경우, 평균 100.88초가 소요되었다. 이러한 결과는 슬롯의 길이가 상대적으로 짧은 경우 메시지 인증을 정상적으로 마치지 못하는 노드의 수가 늘어났기 때문으로 추정되며 이러한 경우가 가변 키 슬롯의 경우, 전체 실험의 75%에 해당하기 때문으로 보인다. 해시의 수행 횟수에서는 슬롯이 가변적으로 조정될 때에 비해 고정일 때 63%정도 더 많은 것을 확인할 수 있었다.

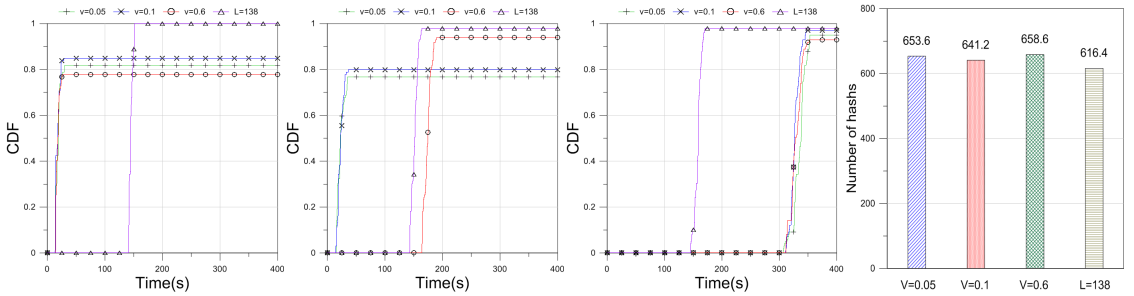
가변 키 슬롯의 경우에서, v가 0.1인 경우는 v가 0.05인 경우와 비교할 때 응답율에 있어서는 약간 더 나아짐을 보였으며, 반응시간은 비슷하였다. 약 46%의 노드가 31초 안에 메시지 인증을 마쳤으나, 약 12.2%의 노드는 메시지 인증을 정상적으로 수행하지 못하였다. 고정 키 슬롯의 길이가 100인 경우에는 82인 경우에 비해서 16초 정도 메시지 인증이 지연되었다. 해시 함수는 고정 키 슬롯의 경우가 33%정도 더 수행하여야 했다.

가변 키 슬롯의 경우, v가 0.6일 때, 약 34%의 노드가 32.80초에, 약 67%의 노드가 194.29초에 메시지 인증을 마쳤다. 메시지 인증을 마치지 못한 노드는 11.7%정도로 다른 경우에 비해서 낮아졌다. 고정 키 슬롯의 길이가 138인 경우 평균 인증 시간은 151.48초가 필요했다. 이 두 경우에는 수행해야 하는 해시 함수의 횟수에는 큰 차이가 벌어지지 않음을 알 수 있다.

앞서 확인할 수 있다시피, 키 슬롯의 길이가 너무 짧은 경우, 네트워크가 불안정하게 동작하게 됨을 알 수 있었다. 이는 고정 키 슬롯의 경우에도 마찬가지임을 뒤의 실험에서 확인할 수 있다. 즉, 키 슬롯의 길이와 응답율 사이에 약간의 트레이드-오프 문제가 성립함을 확인할 수 있다.

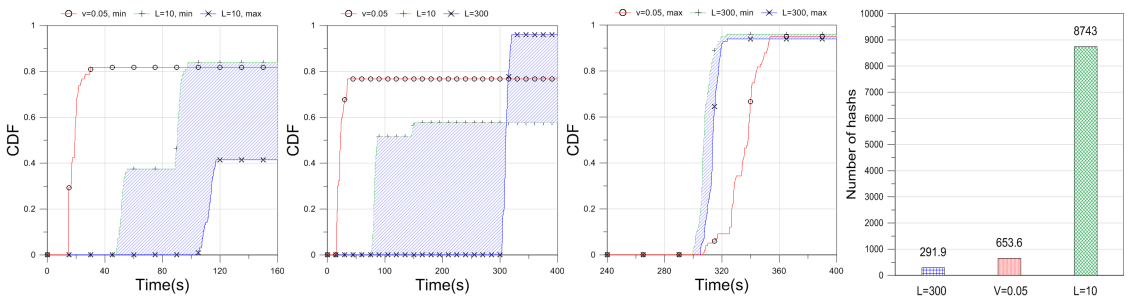
4.3 평균 수행한 해시 함수의 횟수가 유사할 때의 반응 시간의 차이

가변 키 슬롯을 이용한 μTESLA에서 v가 각각 0.05, 0.1, 0.6인 경우와 고정 키 슬롯을 이용한 경우 키 슬롯의 길이를 138로 한 경우를 비교하였다. 그림 7(a)는 각 실험의 결과 최소 반응 시간이 나온 경우를 비교하였는데, 가변 키 슬롯을 이용한 경우에는 v가 0.05일 때 81%의 노드가 29.94초 동안 메시지 인증을 성공 하였다. 고정 키 슬롯을 이용한 경우에는 81%의 노드가 인증되는 데 148.53초가 걸렸다. 그림 7(b)는 50회 수행한 실험의 평균 반응 시간의 중간값을 비교하였다. v가 0.1인 경우 약 80%의 노드가 36.17초에 걸쳐 인증을 하였고, 고정 키 슬롯을 이용한 경우에는 약 81%의 노드가 메시지를 인증하는 데 156.76초가 필요하였다. 그림 7(c)는 각 실험에서 최대 반응 시간이 관찰된 경우를 비교하였다. 가변 키 슬롯을 이용할 때 v가 0.1인 경우에 91%의 노드가 메시지 인증을 완료하는 데 339.88초가 걸렸다. 이는 BS의 방문이 통계적으로 예측한 BS의 방문 시간 분포를 따르지 않은 경우로, 긴 키 슬롯을 사용하는 시간에 BS가 방문한 경우에 측정된 반응 시간이다. 그림 7(d)는 v가 0.05, 0.1, 0.6인 가변 키 슬롯과 고정 키 슬롯에서 키 슬롯의 길이를 138으로 사용한 경우에 각각 수행하는 해시 함수의 횟수를 측정하였다. 이렇게 평균 수행하는 해시 함수의 횟수가 큰 차이를 보이지 않는 경우에 가변 키 슬롯을 이용하는 경우 BS의 방문이 키 슬롯이 짧은 구간에 집중 된다면, 즉 BS의 방문이 v값이



(a) minimum response time in variable and fixed key slot μTESLA (b) median response time in variable and fixed key slot μTESLA (c) maximum response time in variable and fixed key slot μTESLA (d) average of number of hash function operation

그림 7. 해시 함수 수행 횟수가 유사한 경우 반응 시간의 비교(가변 키 슬롯 v=0.05, 0.1, 0.6과 고정 키 슬롯 L=138)
Fig. 7. Comparison of response time on similar hash operation (variable key slot v=0.05, 0.1, 0.6 and fixed key slot L=138)



(a) minimum response time in v=0.05 variable key slot and L=10 fixed key slot μTESLA (b) median response time in v=0.05 variable key slot and L=10, 300 fixed key slot μTESLA (c) maximum response time in v=0.05 variable key slot and L=300 fixed key slot μTESLA (d) number of hashs in variable and fixed key slot μTESLA

그림 8. 가변 키 슬롯 v=0.05와 고정 키 슬롯 L=10, 300에서 해시 함수 수행 횟수의 비교
Fig. 8. Comparison of the number of hashs in variable key slot v=0.05 and fixed key slot L=10, 300

작은 정규분포를 따른다면 더 빠른 반응 시간을 보임을 알 수 있다.

4.4 키 슬롯의 차이에 따른 반응 시간과 해시 함수의 수행 횟수 비교

가변 키 슬롯을 이용한 μTESLA에서 v가 0.05인 경우와 고정 키 슬롯을 이용한 경우 L이 각각 10, 300인 경우를 비교하였다. 그림 8(a)는 v가 0.05인 경우 반응 시간이 최소인 경우와 L이 10인 경우 반응 시간이 최소, 최대인 경우를 비교하였다. v가 0.05인 경우 약 80%의 노드들이 29.94초 동안 인증을 마쳤으며 L이 10이고 반응 시간이 최소인 경우에 약 37%의 노드가 인증을 마치는 데 55.32초가 걸렸고, 약 81%의 노드들이 인증을 마치는 데에는 95.85초가 걸렸다. 그림 8(b)는 고정 키 슬롯과 가변 키 슬롯의 경우 각각 반응 시간이 중간값인 경우를 비교하였다. v가 0.05인 가변 키 슬롯을 이용한 경우 키 슬롯의 길이가 10, 300인 고정 키 슬롯을 이용한 경우보다 반응 시간이 짧음을 볼 수 있다. 그림 8(c)에서는 v가 0.05인 경우 반응 시간이 최대인 경우와 L이 300인 경우 반응 시간이 최소와 최대인 경우를 각각 비교하였다. 가변 키 슬롯의 경우 약 82%의 노드가 인증을 마치기까지 344.56초가 필요했고, 고정 키 슬롯에서 L이 300인 경우에 반응 시간이 최대일 때 약 91%의 노드가 319.5초 동안 인증을 마칠 수 있었다. 그림 8(d)는 각각의 경우 수행하는 해시 함수의 횟수를 나타낸다. 가변 키 슬롯을 이용한 경우는 고정 키 슬롯에서 키 슬롯의 길이를 300으로 한 경우보다는 약 2배 정도 많은 해시 함수의 수행을 하는 것으로 보이고, 키 슬롯의 길이가 10인 경우와 비교하였을 때는 고정 키 슬롯의 경우가 약 13배 정도 해시 함수의 수행을 더 하는 것을 볼 수 있다.

이상의 결과를 통해 볼 때, 가변 키 슬롯을 이용한 경우에 더 빠른 응답 시간을 보이면서, 추가적인 해시 함수의 수행은 많지 않음을 알 수 있다.

V. 토 의

이 논문의 4장 4절에서 수행한 해시 함수 수행 횟수의 비교 실험에서 시뮬레이션을 통해 얻은 결과값과 시뮬레이션과 유사한 시나리오로 계산한 결과를 정리하면 다음과 같다.

계산을 위해 설정한 시나리오는 네트워크에 배치된 센서 노드들이 24시간 지난 뒤에 BS가 방문하여 브로드캐스트 메시지를 보낸 경우로 시뮬레이션에서 사용한 것과 같다. 이 시나리오에서 각각 고정 키 슬롯과 가변 키 슬롯을 이용한 경우 수행된 해시 함수의 횟수는 다음과 같이 계산할 수 있다.

표 2. 계산과 실험으로 얻은 해시 연산 수행 횟수 비교
Table 2. Comparison of the number of hashs in Simulation and Calculation

Type	Calculation	Simulation
fixed, L=10	8640	8743
fixed, L=300	288	291.9
variable, v=0.05	636.305	653.6

고정 키 슬롯 길이 300인 경우 수행된 해시 함수의 수는 $24\times 3600/300=288$

고정 키 슬롯 길이 10인 경우 수행된 해시 함수의 수는 $24\times 3600/10=8640$

가변 키 길이 평균 12, 분산이 0.05인 통계를 따르는 BS방문의 경우 해시 함수의 수행 횟수는 $\sum_{i=0}^{47}(\frac{T}{\ell_{slot,i,0.05}})=636.305$ 과 같이 계산할 수 있다.

이를 실제 실험과 비교하면 표 2와 같으며, 이를 통해 계산을 통해 얻은 해시 함수의 수행 횟수와 시뮬레이션을 통해 얻은 고정 키 슬롯 및 가변 키 슬롯을 이용한 μTESLA에서의 해시 함수 수행 횟수가 거의 같음을 알 수 있다.

VI. 결 론

이 논문에서는 기존에 고정 키 슬롯을 이용한 μTESLA와 가변 키 슬롯을 이용하는 μTESLA를 다양한 매개변수를 이용하여 실험하고 비교하였다. 이를 통해 고정 키 슬롯을 이용한 μTESLA에서 발생할 수 있는 문제점을 지적하였으며, 문제점들을 보완한 가변 키 슬롯을 이용한 μTESLA를 제안하였다. 가변 키 슬롯을 이용한 μTESLA를 이용할 때, 기존의 고정 키 슬롯을 이용한 μTESLA에서 발생하는 응답 반응시간과 해시 함수 수행 횟수간의 균형점 문제를 완화할 수 있었다. 다만 이 논문에서 제안하는 가변 키 슬롯을 이용하여 μTESLA를 효율적으로 운영하려면 UWSN 환경에서 BS의 방문이 통계적으로 예측할 수 있어야 한다. 이것은 BS의 방문에 대한 통계 데이터를 기반으로 하여 μTESLA의 키 슬롯 길이를 조절하기 때문이다. 가변 키 슬롯을 이용하는 경우 고정 키 슬롯을 이용할 때 보다 해시함수의 수행 횟수를 줄일 수 있고 이는 전체 네트워크에서 소모되는 에너지를 줄일 수 있게 된다. 또한 BS의 메시지를 받은 뒤 인증 완료까

지의 걸리는 시간 역시 줄여서 네트워크의 응답성을 높일 수 있었다.

그러나 키 슬롯의 길이가 너무 짧은 경우에는 센서 노드들이 정해진 동면-활동을 할 때 BS로부터 받는 메시지의 수신 기회가 낮아지기 때문에 메시지 응답율이 떨어지게 된다. 향후 연구에서는 추가적인 실험을 통해 네트워크의 정상적인 동작을 보장하는 응답율과 그에 따른 키 슬롯의 길이를 설정하여 최적의 값을 알아낼 수 있을 것이다.

References

[1] A. Perrig, R. Szewczyk, J. Tygar, V. Wen, and D. Culler, "SPINS: Security protocols for sensor networks," *J. Wirel. netw.*, vol. 8, no. 5, pp. 521- 534, Sept. 2002.

[2] Y. Cho and S. Lee, "An IDE based hierarchical node authentication protocol for secure data transmission in WSN environment," *The Korean Inst. Commun. Inf. Sci.*, vol. 37B, no. 3, pp. 149-157, 2012.

[3] Y. Wang, L. Hu, J. F. Chu, and X. B. Xu, "Analysis and improvement for SPINS," *J. Netw.*, vol. 8, no. 1, pp. 229- 236, Jan. 2013.

[4] Z. S. Bojkovic, B. M. Bakmaz, and M. R. Bakmaz, "Security issues in wireless sensor networks," *Int. J. Commun.*, vol. 2, no. 1, pp. 106-115, 2008.

[5] A. Perrig and R. Canetti, "Efficient and secure source authentication for multicast," *Netw. Distributed Syst. Security Symp., NDSS*, vol. 1, pp. 35-46, Feb. 2001.

[6] D. Liu and P. Ning, "Multilevel μ TESLA: Broadcast authentication for distributed sensor networks," *ACM Trans. Embedded Computing Syst.*, vol. 3, no. 4, pp. 800- 836, 2004.

[7] C. H. Lim, "New constructions of multi-level μTESLA with immediate authentication," *Korea Inst. Inf. Security & Cryptography*, vol. 16, no. 6, pp. 163-167, 2006.

[8] R. Du and S. Wen, "An improved scheme of μTESLA authentication based trusted computing platform," *Int. Conf. Wirel. Commun., Netw. Mobile Comput., 2008 (WiCOM'08)*, pp. 1-4, 2008.

[9] Y. Z. Y. Zhou and Y. F. Y. Fang, "WSN09-1: BABRA: Batch-based broadcast authentication in wireless sensor networks," *IEEE GLOBECOM 2006*, pp. 1-5, San Francisco, CA, 2006.

[10] M. Di Francesco and K. Shah, "An adaptive strategy for energy-efficient data collection in sparse wireless sensor networks," *Wirel. Sensor Netw.*, pp. 322-337, 2010.

[11] F. S. Babamir and a. Norouzi, "Achieving key privacy and invisibility for unattended wireless sensor networks in healthcare," *The Comput. J.*, May 2013.

[12] Crossbow, MICAz Data Sheet, 6020-0060-04 Rev A, from http://www.openautomation.net/uploadsproductos/micaz_datasheet.pdf

[13] P. Ganesan, R. Venugopalan, P. Peddabachagari, A. Dean, F. Mueller, and M. Sichitiu, "Analyzing and modeling encryption overhead for sensor network nodes," in *Proc. ACM Int. conf. Wirel. sensor netw. appl. - WSNA '03*, pp. 151-159, 2003

최 진 춘 (JinChun Choi)

2011년 2월 : 인하대학교 컴퓨터 정보공학과 졸업

2014년 2월 : 인하대학교 컴퓨터 정보공학과 석사

2014년 3월~현재 : 인하대학교 컴퓨터 정보공학과 박사 과정

<관심분야> 네트워크 보안, 무선 센서 네트워크 보안

강 전 일 (Jeonil Kang)

2003년 2월 : 인하대학교 컴퓨터 정보공학과 졸업
2006년 2월 : 인하대학교 정보통신대학원 석사
2006년 3월~현재 : 인하대학교 정보공학과 박사 과정

<관심분야> RFID 보안, 생체 인식 보안, 무선 센서 네트워크 보안, 무선 인터넷 보안, 웹 인증 보안

양 대 헌 (DaeHun Nyang)

1994년 2월 : 한국과학기술원 과학기술 대학 전기 및 전자공학과 졸업
1996년 2월 : 연세대학교 컴퓨터과학과 석사
2000년 8월 : 연세대학교 컴퓨터과학과 박사
2000년 9월~2003년 2월 : 한국전자통신연구원 정보보호연구본부 선임연구원
2003년 2월~현재 : 인하대학교 컴퓨터정보공학과 부교수
<관심분야> 암호이론, 암호프로토콜, 인증프로토콜, 무선 인터넷 보안, 네트워크 보안

이 경 희 (KyungHee Lee)

1993년 2월 : 연세대학교 컴퓨터과학과 학사
1998년 8월 : 연세대학교 컴퓨터과학과 석사
2004년 2월 : 연세대학교 컴퓨터과학과 박사
1993년 1월~1996년 5월 : LG소프트(주) 연구원
2000년 12월~2005년 2월 : 한국전자통신연구원 선임연구원
2005년 3월~현재 : 수원대학교 전기공학과 부교수
<관심분야> 바이오인식, 정보보호, 컴퓨터비전, 인공지능, 패턴인식