논문 13-38B-09-03

한국통신학회논문지 '13-09 Vol.38B No.09
http://dx.doi.org/10.7840/kics.2013.38B.9.707

사물 인터넷 환경에서 경량화 장치 간 상호 인증 및 세션키 합의 기술

박 지 예◆, 신 새 미*, 강 남 희°

Mutual Authentication and Key Agreement Scheme between Lightweight Devices in Internet of Things

Jiye Park◆, Saemi Shin*, Namhi Kang°

요 약

사물인터넷(IoT) 기술은 M2M 통신의 확장 기술로 구성 장치(사물)들을 인터넷에 연결시켜 사물지능통신을 실체화하기 위해 제안되었다. IoT를 구성하는 다양한 사물들은 일반적으로 자원이 제한적이고, 이기종 장치들은 저용량 네트워크로 상호 연결된다. 이러한 IoT 환경에서 보안 서비스를 제공하기 위해서는 기밀성, 상호인증, 메시지 송신 인증 등이 제공되어야 한다. 그러나 자원이 제한적인 환경 특성상 기존 인터넷 환경에 적용했던 보안 기술들을 그대로 적용하기에는 무리가 있다. IETF 표준화 그룹에서는 안전한 IoT 서비스를 위해 경량화된 DTLS(Datagram TLS) 프로토콜의 적용을 제안하고 있지만 초경량 장치까지 모든 장치를 수용할 수는 없다. 이를 해결하기 위해 본 논문에서는 자원 제약의 이유로 해쉬 함수 혹은 암호 함수와 같은 단일 보안 모듈만을 탑재할 수 있는 경량화 장치들이 상호 인증하고 세션키를 합의할 수 있는 방안을 제안한다. 제안 기술은 세션키 생성 시 사전 계산 방식을 통해 성능을 향상시킬 수 있고 다양한 보안 공격에 대응 할 수 있다.

Key words : IoT, LLN, DTLS, Authentication, Session Key Agreement

ABSTRACT

IoT, which can be regarded as an enhanced version of M2M communication technology, was proposed to realize intelligent thing to thing communications by utilizing Internet connectivity. Things in IoT are generally heterogeneous and resource constrained. Also such things are connected with each other over LLN(low power and lossy Network). Confidentiality, mutual authentication and message origin authentication are required to make a secure service in IoT. Security protocols used in traditional IP Networks cannot be directly adopted to resource constrained devices in IoT. Under the respect, a IETF standard group proposes to use lightweight version of DTLS protocol for supporting security services in IoT environments. However, the protocol can not cover up all of very constrained devices. To solve the problem, we propose a scheme which tends to support mutual authentication and session key agreement between devices that contain only a single crypto primitive module such as hash function or cipher function because of resource constrained property. The proposed scheme enhances performance by pre-computing a session key and is able to defend various attacks.

※ 본 연구는 2012년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2013023700)
※ 본 연구는 미래창조과학부및정보통신산업진흥원의 대학IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2013-H0301-13 - 1003)
◆ 주저자 : 덕성여자대학교 컴퓨터공학부, jiyepark@duksung.ac.kr, 학생회원
° 교신저자 : 덕성여자대학교 디지털미디어학과, kang@duksung.ac.kr, 정회원
* 덕성여자대학교 컴퓨터공학부, apst840@gmail.com, 학생회원
논문번호 : KICS2012-07-289, 접수일자 : 2013년 7월 9일, 최종논문접수일자 : 2013년 9월 2일

Ⅰ. 서 론

스마트폰, 스마트패드, 스마트센서, 스마트TV, 스마트자동차와 같은 다양한 이종 스마트 기기들은 기술의 발전과 더불어 인터넷을 통해 연결되면서 실생활과 더욱 밀접해 지고 있다. 일인당 스마트 장치 소유 개수는 지금보다 더욱 증가하여 2015년에는 최소 7개로 증가하고, 약 250억 개의 장치들이 무선 인터넷을 기반으로 연결될 것이라고 예상된다[1].

인터넷을 기반으로 한 사물 지능통신 기술인 IoT (Internet of Things)는 자원제한적인 센서를 포함한 이기종 스마트 기기 간의 상호 접속 네트워크를 제공하고자 한다. 따라서 IoT 환경에는 컴퓨팅 파워, 메모리의 가용성, 배터리 파워, 통신 대역폭 등 다양한 환경적 특수성을 고려해야한다[2]. 이에 IETF 표준화 기구의 LWIG(Light-Weight Implementation Guidance) 워킹 그룹에서는 IoT 환경을 구성하는 장치들을 자원의 제한적인 정도에 따라 클래스 0부터 클래스 2까지 구분하고 있다. 특히 Class 0에는 메모리가 10KiB 이하이고, 최대 적재 가능한 코드 크기가 100KiB 이하의 초경량화 장치들이 포함된다[3]. Class 0의 장치들은 비용이나 효율성을 고려하여 LLN (Low Power Lossy Network)으로 분류되는 IEEE 802.15.4나 저전력(Low Power) Wifi 등의 접속 기술을 사용한다.

서로 다른 성능을 가진 이기종 장치들과 LLN 환경이 인터넷과 결합된 IoT는 빌딩 자동화, 환경 모니터링, 에너지 관리 등 다양한 영역에 적용될 수 있다. 특히 LLN은 BAN(Body Area Network), CAN(Car Area Network)등에 적용하여 헬스 케어, 스마트 카와 같은 서비스를 제공할 수 있다. 상기 서비스를 제공하기 위해서는 IoT 환경을 구성하고 있는 장치 간 상호 인증, 메시지 송신 인증 및 정보의 기밀성 등이 필수적으로 제공되어야 한다.

IETF CORE 그룹에서는 IoT 환경을 위해 CoAP (Constrained Application Protocol)을 표준화 하고 있다. 특히 안전한 서비스 제공을 위해 기존 인터넷 환경에서 사용하던 보안 프로토콜인 DTLS(Datagram Transport Layer Security), HIP등을 자원 제한적인 환경에 맞게 경량화 하여 적용하는 방안을 모색하고 있다[4].

DTLS는 UDP와 같은 데이터 그램 프로토콜을 사용하는 응용 서비스에 데이터의 기밀성, 무결성, 사용자 인증 등을 제공하는 보안 프로토콜이다. 그러나 DTLS의 경우 전송되어야 하는 총 6번의 메시지 패킷은 fate-sharing 특성을 가지므로, 한 패킷이라도 손실될 경우 전체 메시지를 다시 전송해야 한다. 메시지 패킷의 재전송은 전송량을 증가시켜 LLN 환경에 부담을 주고, 자원 제한적인 장치의 성능이 저하되는 결과를 가져온다. HIP 기술에서는 암호학적 해쉬 함수인 HMAC을 기본으로 사용한다. 이를 경량화 한 Diet HIP 기술은 메모리가 제한된 장치에서도 HIP을 사용할 수 있도록 해준다.

그러나 표준화에서 진행하고 있는 보안프로토콜의 경량화 방안들은 이기종 IoT 환경에서 암호화 모듈을 탑재하지 못하는 초경량 장치들을 모두 수용할 수 없다. IoT 환경을 구성하고 있는 장치들 중 Class 0으로 분류된 기기들은 메모리나, 저장용량, 배터리와 같은 자원이 매우 제한적인 특성으로 인해 보안 기능을 제공하는 다양한 모듈들을 모두 탑재하는 것이 불가능하다. 다음 표는 [5]에 기술된 내용으로 DTLS의 보안 모듈들을 경량화 하여 적재할 경우를 고려한 코드 크기의 예이다.

표 1. 보안 모듈별 코드 사이즈
Fig 1. Cryptographic module code size

Library	Code Size
MD5	4,856 bytes
SHA1	2,432 bytes
HMAC	2,928 bytes
RSA	3,984 bytes
Big Integer Implementation	8,328 bytes
AES	7,096 bytes
RC4	1,496 bytes
Random Number Generator	4,840 bytes

표 1을 참조하면 DTLS 프로토콜을 경량화 하더라도 필요한 보안 모듈들을 모두 탑재할 경우, 총 35,960 bytes의 적재 공간이 필요하다. 본 논문의 4.2절에 기술된 동작시험에 사용한 Arduino uno R3는 상대적으로 큰 저장 장치를 갖고 있음에도 (플래시메모리의 크기는 32KiB임) 모든 암호화 모듈을 탑재 할 수 없다. 따라서 적용 시나리오별 보안 요구사항에 따라, 장치 특성에 따라 서로 다른 인증 방법 및 세션 키 분배 방법이 제공되어야 한다.

이러한 제한사항을 극복하기 위해 본 논문에서는 단일 보안 모듈만 탑재될 수 있는 경량화 장치들을 위한 상호 개체 인증 및 세션키 합의 방식을 제안한다. 각 보안 모듈은 다음의 보안 서비스를 제공하기 위해

장치에 적재될 수 있다.

- 안전한 서비스(기밀성)에 적용되기 위해 대칭키 기반 암호화 모듈만을 가지고 있는 장치
- 무결성 및 데이터 송신 인증을 제공하기 위해 해쉬 함수만이 탑재된 장치

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대해서 기술하고 3장에서는 상호 인증 및 세션키 합의 기술을 제안한다. 또한 합의된 키를 기반으로 암호화 통신 및 데이터 송신 인증 시스템을 제공할 수 있는 방안을 기술한다. 4장에서는 제안한 시스템의 보안 분석과 동작 시험 내용을 기술한다. 마지막으로 5장에서 결론을 맺는다.

Ⅱ. 관련 연구

사물 간 통신기술인 M2M(machine to machine)은 각각의 구성 장치(사물)에 인터넷과의 연결성을 제공하면서 IoT 기술로 확장 발전되고 있다. IoT는 수없이 많은 종류의 장치들이 인터넷 프로토콜을 이용하여 통신 서비스를 제공하는 하나의 트렌드를 나타낸다[2]. 이러한 환경에서 다양한 응용 및 서비스를 창출하기 위해 보안은 중요한 요소가 된다.

M2M이나 센서 네트워크 환경에서 보안 서비스를 제공하기 위해 다양한 인증 기술들과 세션키 분배 및 합의 기술들이 제안되었다. M2M이 IoT로 발전되면서 IETF CoRE 워킹그룹 표준화 작업에서는 IoT환경을 위한 경량화 DTLS에 관한 연구가 활발히 진행되고 있다. 본 장에서는 문헌을 통해 제안된 인증 및 세션키 분배 기술의 예와 표준화 그룹에서 제안하는 방안을 간략하게 살펴본다.

2.1. 구성 장치 간 인증 및 세션키 분배 방안

센서 네트워크 환경과 IoT는 경량화된 센서들의 네트워크 환경이라는 점에서 유사점을 가진다. 반면 기존 센서 네트워크 환경에서의 통신은 센서 노드와 인프라 노드(베이스 스테이션) 간 통신이라는 점에서 센서 간 통신이 주를 이루는 IoT환경과 차별성을 가진다. 특히, IoT의 범주에 포함되는 WoT를 구성하는 장치는 웹 클라이언트와 웹 서버의 기능을 동시에 수행할 수 있다. 이와 같은 차이로 기존센서 네트워크 환경을 기반으로 제안되었던 일반적 인 방안을 그대로 IoT환경에 적용하기 어렵다.

[6]에서는 베이스스테이션과 키 분배 센터를 통한 공유키 설정 및 세션키 분배 방안을 제안하였다. 그러나 제안 기술은 센서 간 비밀 키 공유를 위해 베이스 스테이션과 통신을 하고 베이스 스테이션과 키 분배 센터와 다시 통신을 해야 하는 번거로움이 있다. 또한 암호화와 해쉬함수를 모두 사용하므로 모든 모듈을 탑재할 수 없는 자원제한적인 센서에 부담이 될 수 있다. [7]에서는 센서 네트워크 환경에서 공개키를 이용한 키 분배 방안을 제안하였다. 상기 지적한 이유로 [7]에서 제안된 방식은 RSA나 ECC와 같은 암호화 모듈을 탑재 할 수 없는 경량 센서에는 적용이 어렵다.

2.2. IoT 환경을 위한 경량화 보안 기술

IoT 기술을 표준화하고 있는 IETF의 CORE 작업그룹에서는 TCP/IP 기반 인터넷 환경에 적용되었던 TLS, DTLS, IPSec, HIP, PANA 등을 적용할 수 있는 방안을 고려하고 있다[8]. 특히, 핵심 프로토콜인 CoAP에 DTLS의 적용을 기본 방향으로 설정하고 있다[4]. 기존 IP 기반 보안 프로토콜을 적용하기 위해서는 장치의 계산 능력과 메모리 공간을 고려하여 경량화 시킬 방안이 필요하다. 또한 LLN의 통신 능력을 고려하여 전송하는 메시지를 최소화 해야 한다. 이를 위해 다양한 기술들이 제안되고 있다[9-11].

DTLS 프로토콜을 경량화 하는 방법에는 핸드 쉐이크 메시지의 패킷 개수를 줄이거나 인증서에 대한 검증과정을 간단히 하는 방법이 있다. [9]에서는 핸드쉐이크 메시지 패킷을 줄이기 위해 자원 제한적인 장치의 소유자에게 초기 핸드쉐이킹 과정을 위임한다. 장치 소유자와 각 장치는 사전에 비밀키를 안전하게 공유하고 서버 측과 DTLS 세션을 맺는다. 장치 소유자는 서버와 성공적으로 맺어진 DTLS 세션 정보를 장치에 사전에 공유한 비밀키로 암호화하여 전송하고 세션을 종료한다. 그 후 장치와 서버 간 세션을 재개할 수 있게 하여 복잡한 초기 핸드쉐이킹 과정을 장치 소유자가 대신 할 수 있게 한다. 하지만 제안된 시스템은 결국 장치와 서버간 DTLS 세션 재개를 위해서 장치에 DTLS 프로토콜을 전부 올려야 하는 부담이 있어 암호화 모듈을 탑재하지 못하는 초경량 센서에는 적용할 수 없다.

[9]에서는 계산량의 오버헤드를 줄이기 위해 OCPS를 통한 인증서 검증을 게이트웨이에서 하는 방안을 제안 하였다. 하지만 암호화 모듈을 이용할 수 있는 장치라고 가정한다 하더라도 게이트웨이를 통한 인증서 검증은 신뢰 할 수 없다. 게이트웨이에서의 사전 인증서 검증은 MTM(Man in the Middle)공격의 가능

성을 배재 할 수 없다.

이와 달리 [10]은 CoAP에 DTLS를 적용할 경우 전송되는 메시지의 크기를 주로 고려했다. LLN의 적용으로 고려되고 있는 IEEE 802.15.4의 최대 전송 프레임 크기 (즉, MTU)를 고려할 경우 단일 메시지에 DTLS 메시지를 전송하기는 어렵다. 이를 해결하기 위해 [10]에서는 6LoWPAN에서 제공하는 헤더 압축 기술을 적용하여 DTLS Record 영역을 최소화할 수 있는 방안을 제안했다. [11]에서 제안된 기술도 LLN에 전송되는 메시지를 최소화할 수 있는 방안을 다룬다. 차이점은 [11]에서는 종단간의 보안을 제공하기 위해 압축된 IPSec을 적용했다.

Ⅲ. 제안 시스템

본 논문에서 제안하는 시스템은 LWIG 워킹그룹에서 제시한 WoT환경을 구성하고 있는 자원 제한적인 센서들 중 Class 0에 해당하는 초경량 센서와 Class 1 이상의 센서로 나누어 각 장치 성능에 맞는 인증방법 및 세션키 공유방법을 제안한다.

다음은 본 논문의 제안 시스템에서 사용되어 지는 파라미터들을 나타낸다.

- I는 인증 개시자를 나타낸다.
- R은 인증 개시자로부터 전송 받은 메시지에 응답하는 응답자를 나타낸다.
- $r1$, $r2$는 랜덤 넌스값을 나타낸다.
- kIR은 인증자와 응답자 간 사전에 안전하게 공유된 키를 의미한다.
- sk는 인증자와 응답자가 공유한 세션키를 의미한다.

3.1. 대칭키 기반 인증 및 세션키 동의 시스템

제안 시스템은 대칭키 기반 암호화 모듈만 사용할 수 있는 자원 제한적인 장치에서 장치 간 인증 및 세션키 공유 방안을 제공한다. 본 방안에서 인증 요청자인 I와 응답자 R사이에는 비밀키 값인 kIR과 암호화 함수인 E_k(x)이 장치 설정 단계에서 사전에 안전하게 저장되어 있다고 가정한다. 인증 및 세션키 동의 절차는 다음과 같다.

(1) I는 랜덤수 $r1$을 생성하고, 자신의 ID값인 I_ID와 같이 R에게 전송한다.
(2) R은 전송받은 $r1$과 자신이 생성한 $r2$를 연접한 값을 kIR로 암호한 값과, 자신의 ID인 R_ID를 I에게 응답으로 전송한다.
(3) I와 R은 $r1$과 $r2$를 XOR연산 한 후, kIR로 암호화 하여 세션키 sk를 생성한다.
(4) I는 생성한 세션키 sk를 이용하여 전송받은 $r2$를 암호화 하여 R에게 재전송 하여 인증 받는다.
(5) R은 (4)메시지를 통해 I를 인증하고, 데이터는 sk를 이용하여 암호화한 후 전송하는 것으로 기밀성을 제공한다.
(6) I도 데이터 암호화에 sk를 이용하게 된다.
(7) 암호화 통신이 끝나면 세션이 종료 되고, 두 주체는 사용했던 sk를 폐기한다.

3.1.1. 상호 인증 및 세션 키 공유 절차

(1) $I \rightarrow R$: { I_ID, r1 }
(2) $R \rightarrow I$: { R_ID, E_kIR(r1 ‖ r2) }
(3) I , R: sk = E_kIR(r1 $\oplus$ r2)
(4) $I \rightarrow R$: { E_sk(r2) }

3.1.2. 데이터 암호화 통신

(5) $R \rightarrow I$: { E_sk(Data1)}
(6) $I \rightarrow R$: { E_sk(Data2)}
(7) Session 종료

본 방안을 통해 인증 요청자 I와 응답자 R이 상호 인증을 할 수 있고, 인증에 사용했던 랜덤 값을 세션키 동의에 이용하므로 세션 키 공유를 위한 새로운 파라미터의 교환이나 추가적인 방법 없이 인증 시 마다 서로 다른 세션 키를 공유 할 수 있다. 또한 응답자 R은 (2) 메시지 전송 후 자신이 가지고 있는 $r1$, $r2$, E_kIR을 이용하여 세션 키를 미리 계산하여 (3) 메시지가 전송되는 즉시 검증 할 수 있어 계산 성능 상 효율적이다. R은 (2)에 표시된 메시지를 전송할 때 $r1$과 $r2$을 연접하여 암호화한다. 따라서 세션키 생성 시 사용한 $r1$과 $r2$의 XOR연산 방안과 입력값이 다르다. 이를 통해 동일한 $r1$과 $r2$를 사용하더라도 (2)과 (3)의 값은 달라지므로 제 3자에 의한 메시지 변조 공격에 대응할 수 있다.

3.2. 해시 기반 인증 및 세션키 동의 시스템

본 절에서는 Class 0으로 분류된 장치가 자원의 제

한으로 해시 함수만 탑재된 경우를 다룬다. 제안 시스템은 장치 간 인증, 세션키 공유 및 데이터 송신 인증 방안을 제공한다. 3.1과 같이 인증 요청자 I와 응답자 R사이에는 키 kIR과 해쉬 함수인 h(x)이 장치 설정 단계에서 사전에 안전하게 저장되어 있다고 가정한다. 또한 암호 강도는 AES 128bit를 기본으로 설명한다. (즉, 해시 출력은 256bit를 가정하고 랜덤 수의 길이는 128bit를 가정한다.) 인증 및 데이터 송신 인증 방안 절차는 다음과 같다.

(1) I는 자신의 ID, r1을 인증대상 장치 R에게 전송한다.
(2) R은 사전에 I와 공유하고 있는 256bit 키 kIR를 이용하여 128bit씩 나눈 앞부분을 kIR_1, 뒷부분을 kIR_2로 나눈다. 위 표에 명시된 KH(data)의 형식을 따름으로써 HMAC과 같은 효과를 낼 수 있다. 또한 kIR를 반으로 나누어 kIR_1, kIR_2로 사용함으로써 두 개의 키를 사용하는 효과를 낼 수 있다.

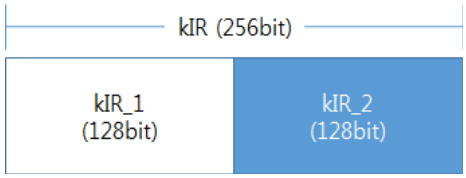


그림 1. 키 분리
Fig. 1. Pre-Shared key division

kIR_2와 $r2$을 XOR 연산한 후 $r1$을 연접하여 해쉬 연산 한 결과 값과 키의 앞부분 kIR_1과 $r1$을 XOR 연산한 값을 연접하여 해쉬 함수의 입력 값으로 사용하고, 해쉬 연산한 결과 값에 R의 ID와 생성한 nonce값 $r2$를 연접하여 전송한다.

(3) I는 (2)에서 전송받은 메시지를 검증하여 (1)에서 보낸 메시지에 대한 응답이 맞는지 확인 후, kIR_2와 $r2$을 XOR한 후 $r2$을 연접하여 해쉬 연산을 수행한다. 그 결과 값을 kIR_1과 $r1$이 XOR 한 값과 연접하여 해쉬한 값을 전송한다.
(4) I와 R은 상호 인증에 사용했던 랜덤 nonce값인 $r1$과 $r2$를 연접한 후, 공유된 키 kIR과 XOR 연산 하여 해쉬한 결과값에 0번째 bit부터 127번째 bit 까지를 세션 키 sk로 사용한다.

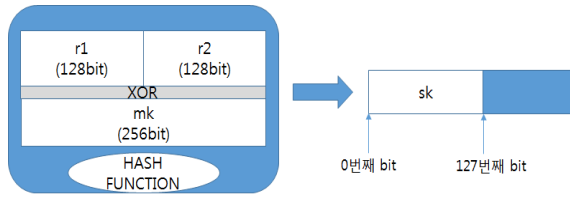


그림 2. 세션키 생성 과정
Fig. 2. Session key generation

(5) R은 생성한 세션 키 sk에 $r2$을 XOR 하고 보내고자하는 데이터를 연접하여 해쉬 하고 결과 값을 세션 키와 $r1$를 XOR 값과 연접하여 해쉬 하여 Data1에 대한 송신 인증 값을 만들어 Data1 과 연접하여 전송한다.
(6) I는 생성한 세션 키 sk에 $r2$을 XOR 하고 보내고자하는 데이터를 연접하여 해쉬 하고 결과 값을 세션 키와 $r1$를 XOR 값과 연접하여 해쉬 하여 Data2에 대한 송신 인증 값을 만들어 Data2와 연접하여 전송한다.
(7) 데이터 전송이 완료되면 세션을 종료한다.

3.2.1. 상호 인증 및 세션 키 절차

※ KH(data)=h(kIR_1⊕r1||h(kIR_2⊕r2|| data))

(1) $I \rightarrow R$: { I_ID, r1 }
(2) $R \rightarrow I$: { KH(r1), r2 }
(3) $I \rightarrow R$: { KH(r2) }
(4) I , R: sk = { |h((r1 || r2) ⊕ kIR)|128 }

3.2.2. 송신 인증 메시지를 통한 데이터 전송

※ SKH(data)=h(sk⊕r1||h(sk⊕r2||data))

(5) $R \rightarrow I$: { Data1 || SKH(Data1) }
(6) $I \rightarrow R$: { Data2 || SKH(Data2) }
(7) Session 종료

본 제안 시스템은 암호화 프로토콜이 제공되지 않고, $r1$과 $r2$를 HMAC의 ipad, opad와 같은 용도로 사용함으로써 HMAC이 구현되어 있지 않은 초경량 장치에서의 상호 인증 및 전송하는 데이터에 대한 데이터 송신 인증을 제공 할 수 있다. 제공되는 해쉬 함수의 결과 bit 길이보다 긴 256bit 키 길이를 사용하여 HMAC과 같은 보안 강도를 제공한다. 또한 랜덤

nonce값을 이용하여 세션 키를 생성하므로 인증 시마다 새로운 세션 키를 공유할 수 있다. 또한 응답자 R은 (2) 메시지 전송 후 세션 키를 미리 생성해 놓을 수 있어 보내고자 하는 데이터에 대한 송신 인증 메시지를 빠르게 생성하여 전송 할 수 있다.

Ⅳ. 보안 분석 및 동작 시험

4.1. 제안 시스템의 보안 분석

본 논문에서 제안 하는 시스템은 재전송 공격 및 중간자 공격, 도청에 의한 비밀키 공격에 대응 할 수 있다. 공격자는 센서노드 간 인증 및 세션키 분배 과정 중 전송되는 패킷을 가지고 있다가 일정시간 이후 재전송 할 수 있다. 본 논문에서 제안 하는 시스템 3.1절과 3.2절의 방안에서는 인증을 시도 할 때마다 새로운 난수를 생성하여 인증에 사용하므로 공격자가 일정시간 이후 재전송 공격을 한다면 인증이 성립되지 않는다. 3.1절 시스템의 경우 일정시간 이후에는 새로운 세션키를 사용하므로 재전송 공격을 할 수가 없고 3.2절 시스템의 경우 또한 새로운 세션키의 사용으로 메시지 송신 인증 값이 변경되어 재전송 공격을 막을 수 있다.

공격자는 제 3자가 센서노드 간 인증 및 세션키 분배 과정 중에 참여하여 인증과정을 통과하거나 숨겨진 정보를 획득하는 중간자 공격을 시도 할 수 있다. 본 제안 시스템에서 제3자는 단순 메시지를 포워딩하는 방법으로 인증과정을 통과할 수 있지만 공격자는 사전에 공유된 비밀키를 알지 못하므로 세션키를 생성할 수 없다. 3.1절 시스템의 경우 공격자는 암호화 되어 전송되는 데이터를 복호화 할 수 없고, 암호화 데이터를 생성하지 못한다. 3.2절 시스템의 경우 메시지 송신 인증 값을 변조하거나 생성할 수 없으므로 중간자 공격에 대응 할 수 있다.

공격자는 세션 키 생성을 위해 주고받는 정보를 수집하여 세션 키 생성을 시도 할 수 있다. 본 논문의 제안 시스템 중 3.1절에 제안된 시스템에서는 세션 키 생성에 사용되는 랜덤 넌스 값 $r2$를 암호화 하여 전송하므로 공격자는 세션키 생성을 위해 사전에 공유되어 있는 마스터 키 값과 $r2$값을 전부 알아야 한다. 3.2절 시스템의 경우 공격자는 마스터 키 값을 알지 못하므로 세션키를 생성하지 못한다. 따라서 본 논문의 제안 시스템은 세션키를 통한 데이터 기밀성 및 데이터 송신 인증을 제공 할 수 있다.

4.2. 동작시험

본 절에서는 3.2절에서 제안한 시스템을 구현하여 시험한 결과를 기술한다. 구현 시스템으로 인증 개시자의 경우 안드로이드 기반의 스마트 폰을 사용하였다. 응답자의 경우 소형 기기 플랫폼으로 많이 사용되는 Arduino Uno을 이용했다. 동작 시스템의 규격은 다음과 같다. 개시자로 사용된 스마트 기기의 운영체제는 Android 4.0.4이다. 시스템에 탑재된 메모리는 1GB RAM이고 CPU는 듀얼코어 1.5GHz이다. 응답자로 사용된 기기는 ATmega328 마이크로컨트롤러인 Arduino UNO R3로 32KB의 플래시메모리와 2KB의 SRAM으로 구성되었다.

그림3은 동작화면을 캡쳐한 것이다. 그림3의 (a)는 개시자이고 (b)는 응답자이다. 본 시험의 Arduino 장치에 사용된 해쉬 함수는 [12]에서 공개한 MD5를 사용 했다. 센서 장치(Arduino)에 제안시스템을 구현한 총 코드 크기는 20,540 바이트였고 MD5연산을 위한 코드 크기는 3,260 바이트였다.

동작 시험 결과, 안드로이드 기기에서 수행된 (1)의 계산시간은 평균 1msec(milli-sec)였고 (3)의 수행시간은 84 msec였다. 이에 반해 자원이 제한적인 Arduino Uno 기기의 경우 (2)의 동작을 수행하는데 평균 574msec이 소요됐다(표2 참조).

표 2. 동작시험 결과
Table 2. Performance test

Device	KH(d) Processing time (측정단위 milli-sec)
Android Smart Device	84
Arduino Uno	574

*KH(d): 데이터 d를 입력받은 keyed-hash 함수

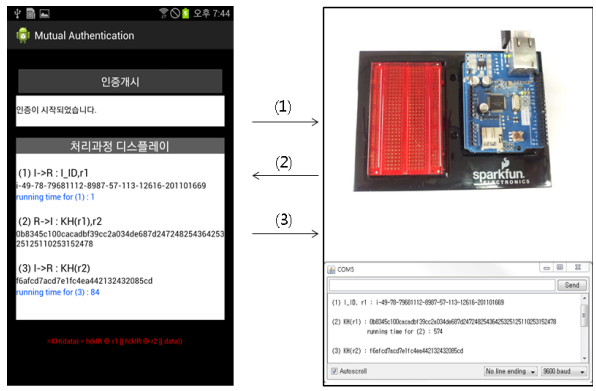


그림 3. 3.2절 제안시스템 동작시험
Fig. 3. 3.2 system operational test

V. 결 론

본 논문에서는 자원이 제한되어 해쉬 모듈만을 탑재한 장치 간 인증 및 데이터 송신 인증 시스템과 암호화 모듈만을 탑재한 장치 간 인증 및 세션키 분배 시스템을 제안 하였다. 제안한 시스템은 키분배 센터에서 키를 분배하던 기존 센서 네트워크 환경을 벗어나 각 센서 노드가 세션키 생성에 합의할 수 있게 하였다. 인증이 완료된 장치는 세션키를 선 계산 할 수 있게 하여 성능을 개선하였다. 제안한 상호 인증 및 세션키 분배 시스템은 재전송 공격, 중간자 공격, 도청에 의한 비밀 키 공격에 대응 할 수 있다. 다만, 본 연구는 인증의 참여 기기들이 비밀키 kIR을 안전하게 공유하고 있다고 가정한다. kIR이 불법 노출될 경우 제안 시스템은 보안 제공에 한계를 갖는다. 기존에 제안된 많은 사전 설정된 비밀키 기반 방식과 동일하게 제안시스템도 안전한 채널과 저장 방식을 가정하고 있다. 그러나 kIR의 안전한 공유 방식은 추가 연구될 필요성이 있다.

References

[1] J. Park and N. Kang, “Entity authentication scheme for secure WEB of Things applications,” *J. KICS*, vol. 38B, no. 5, pp. 394-400, May 2013.

[2] H. Tschofenig, J. Arkko, “Report from the smart object workshop(2012),” Retrieved June, 29, 2013, from http://tools.ietf.org/html/rfc6574.

[3] C. Bormann, M. Ersue, and A. Keranen, “Terminology for constrained node networks(2013),” Retrieved June, 30, 2013, from http://tools.ietf.org/html/draft-ietf-lwig-terminology-03.

[4] Z. Shelby, K. Hartke, and C. Bormann, “Constrained application protocol (CoAP)(2013),” Retrieved June, 1, 2013, from http://tools.ietf.org/html/draft-ietf-core-coap-17.

[5] H. Tschofenig, “Smart Object Security: Considerations for Transport Layer Security Implementations,” in *Proc, Smart Object Security Workshop*, pp. 3, Paris, France, Mar. 2012.

[6] W.S Juang, “Efficient user authentication and key agreement in wireless sensor networks,” *Lecture Notes Comput. Sci.*, vol. 4298, pp. 15-29, 8 2006.

[7] K. Oh, T. Kim, and H. Kim, “Implementation of publickey-based key distribution in wireless sensor network,” in *Proc. KOSBE*, pp. 95-98, Seoul, Korea, Feb. 2008.

[8] T. Heer, O. Garcia-Morchon, R. Hummen, S. Keoh, S. Kumar, and K. Wehrle, “Security challenges in the IP-based Internet of Things,” *Wireless Personal Commun.*, vol. 61, no. 3, pp. 527–542, Dec. 2011.

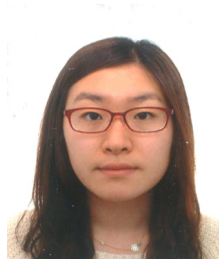
[9] R. Hummen, J. H. Ziegeldorf, H. Shafagh, S. Raza, and K. Wehrle, “Towards viable certificate-based authentication for the Internet of Things,” in *Proc, ACM HotWiSec ‘13*, pp. 37-42, Budapest, Hungary, Apr. 2013.

[10] S. Raza, D. Trabalza, and T. Voigt, “6LoWPAN compressed DTLS for CoAP,” in *Proc. IEEE DCOSS*, pp. 287-289, Hangzhou, China, May 2012.

[11] S. Raza, S. Duquennoy, T. Chung, D. Yazar, T. Voigt, and U. Roedig, “Securing communication in 6LoWPAN with compressed IPsec,” in *Proc. IEEE DCOSS*, pp. 1-8, Barcelona, Spain, June 2011.

[12] Arduino Forum, *MD5 Hash function*, Retrieved 6, 3, 2013, from http://forum.arduino.cc/.

박 지 예 (JiYe Park)



2013년 2월 덕성여자대학교 컴퓨터공학부 졸업

2013년 3월~현재 덕성여자대학원 전산정보통신학과 석사과정

<관심분야> 네트워크 보안, Web of Things>

신 새 미 (SaeMi Shin)

2013년 8월 덕성여자대학교 컴퓨터공학부 졸업

<관심분야> Wireless Network, Cloud Computing>

강 남 희 (Namhi Kang)

2001년 2월 숭실대학교 정보통신대학원 공학석사

2004년 12월 University of Siegen 컴퓨터공학과 공학박사

2009년 3월~현재 덕성여자대학교 디지털미디어학과 조교수

<관심분야> 유무선 인터넷통신, 통신보안, 시스템 보안>