Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.26, NO.6, Dec. 2016 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2016.26.6.1561

IEEE 802.15.4 기반 LoWPAN에서의 디바이스 보안 설정 메커니즘*

이 종 훈,† 박 창 섭‡
단국대학교

Device Security Bootstrapping Mechanism on the IEEE 802.15.4-Based LoWPAN*

Jong-Hoon Lee,† Chang-seop Park‡
Dankook University

요 약

IoT환경에서 센서 디바이스의 사용이 증가함에 따라 보안에 대한 필요성도 중요해지고 있다. 센서 디바이스가 IEEE 802.15.4 기반 LoWPAN에 배치되는 경우 필수적으로 PAN Coordinator와의 가입 과정이 수행되고 후속적으로 디바이스 간 바인딩 과정이 진행된다. 가입 및 바인딩 과정에서는 사전 분배된 네트워크 키 또는 인증서를 이용해 디바이스의 인증 및 키 분배를 한다. 하지만, 기존 방식에서 사용하는 네트워크 키는 그룹인증에 한정된 역할을 하며 인증서의 발급에서도 개별식별이 이루어지지 않는 문제점이 있다. 본 논문에서는 사전 분배된 네트워크 키의 문제점을 보완한 LoWPAN 환경에서 디바이스의 안전한 가입 및 바인딩 프로토콜을 제안한다.

ABSTRACT

As the use of the sensor device increases in IoT environment, the need for device security is becoming more and more important When a sensor device is deployed in IEEE 802.15.4-based LoWPAN, it has to perform the join operation with PAN Coordinator and the binding operation with another device. In the join and binding process, authentication and key distribution of the device are performed using the pre-distributed network key or certificate. However, the network key used in the conventional method has problems that it's role is limited to the group authentication and individual identification is not applied in certificate issuing. In this paper, we propose a secure join and binding protocol in LoWPAN environment that solves the problems of pre-distributed network key.

Keywords: 802.15.4, LoWPAN, Join, Bootstrapping

I. 서 론

최근 사물 인터넷(internet of things) 기술은 경량화 되고 스마트한 센서 디바이스의 다양한 사용을 가능하게 하였다. 사물 인터넷의 센서 네트워크를 IETF(Internet Engineering Task Force)에서는 LoWPAN(Low-Power, Low-data Rate Wireless Personal Area Network)으로 규정하고 있으며 IEEE 802.15.4가 사실상의 표준이다[1]. LoWPAN은 짧은 전송 범위와 손실 네트워크,

Received(08. 17. 2016), Modified(1st: 11. 17. 2016), Accepted(11. 17. 2016)
* 본 연구는 미래창조과학부 및 한국인터넷진흥원의 "고용계약형 정보보호 석사과정 지원사업"의 연구 결과로 수행되었음 (과제번호 H2101-16-1001)
* 이 논문은 2016년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 과학기술인력교류활성화지원사업 성과임.(NRF-2016H1D2A2916919)
† 주저자, skylook4730@naver.com
‡ 교신저자, csp0@dankook.ac.kr(Corresponding author)

센서 디바이스는 작은 저장 공간과 제한된 계산능력을 특징으로 한다[2]. 센서 디바이스가 기존 LoWPAN에 설치되면 PC(PAN Coordinator)와 가입 과정을 수행한 뒤 디바이스 간 센싱 및 액츄에이션 수행을 위해 바인딩(binding)과정을 수행한다. 기존 가입 및 바인딩 과정에서는 사전 분배된 네트워크 키나 인증서를 통한 자격증명으로 신뢰 관계가 구축되는데 현재 대부분의 무선 센서 네트워크에서는 디바이스 자격증명을 위해 키 사전 공유 방식을 사용하고 있다. 하지만, 키 사전 공유 방식은 센서 네트워크의 확장성을 제한하며, 특히 디바이스의 손상으로 인한 공통된 네트워크 키의 노출은 전체 네트워크를 손상시킬 위험이 있다. 대안적으로 인증서 기반의 방식으로 자격증명 및 상호인증을 할 수 있지만 기존 X.509.v3 인증서는 수백 바이트 크기이기 때문에 LoWPAN 환경에서는 적합하지 않고 ECC(Elliptic Curve Cryptography)기반의 수십 바이트 크기의 ECQV(Elliptic Curve Qu-Vans tone)인증서를 사용할 수 있다[3]. 하지만 LoWPAN 환경에서 사용하는 ECQV 인증서도 디바이스의 개별식별 과정 없이 발급된다는 문제점이 존재한다[4]. 본 논문에서는 위 문제점들을 보완한 안전한 가입 및 바인딩 프로토콜을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 IEEE 802.15.4의 표준 가입 과정 및 PAuthKey 프로토콜, 디바이스의 생명주기를 설명하고 3장에서는 기존 프로토콜의 문제점을 보완한 안전한 가입 및 바인딩 프로토콜을 제안한다. 4장에서는 보안 분석을 한다. 5장에서는 기존 방식과 제안 방식을 에너지 소모 관점에서 성능 비교 분석을 하고 마지막으로 6장에서 결론을 내린다.

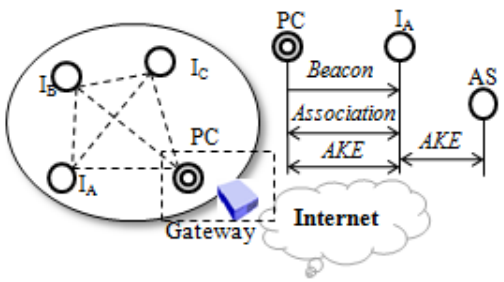

Fig. 1. LoWPAN Block Diagram

II. 관련 연구

2.1 IEEE 802.15.4의 표준 가입 과정

IEEE 802.15.4에서 네트워크는 Fig. 1.에서와 같이 PC의 역할을 하는 디바이스와 몇 개의 센서 디바이스(I_A,I_B,I_C...)로 구성된다. 새로운 디바이스가 LoWPAN에 배치될 때, IEEE 802.15.4의 표준 가입 프로토콜은 PC로부터 *Beacon* 메시지를 수신 후 수행 되며, 두 단계(Association, Authenticated Key Establishment)로 구성된다. 먼저, LoWPAN에 새롭게 진입한 디바이스는 *Association Request* 메시지를 PC에게 보내고, 메시지를 수신한 PC로부터 *Association Response* 메시지를 수신 받으면 가입이 완료된다. 이후 ACL(Access Control List)을 기반으로 가입된 디바이스와 AKE(Authenticated Key Establishment)를 수행해 디바이스에 대한 인증 및 키 설정을 완료한다[5]. AKE는 PC와 디바이스 간이나 AS(Authenticated Server)와 디바이스에서 직접적으로 개별 키를 설정할 수 있다. 하지만 위 가입 프로토콜은 구조적으로 가입 절차와 키 설정을 통한 상호 인증 절차가 분리되어 있어 많은 메시지 교환을 필요로 하는 비효율적인 구조이다.

Table 1. Table of Notations

Notation	Description
PC, I_j	Pan Coordinator and Sensor device ($j=A,B,C,...$)
$\boldsymbol{D}_j, d_j$	ECDH public and private key of I_j
ID_j	secure device identification of I_j or PC
N_j	random number generated by I_j
LK_A, LK_{AB}	pairwise key shared between I_A and PC, I_A and I_B
$kdf(.)$	key derivation function
$H(.)$	2nd-preimage resistant hash function
$MAC64_j$	64bit MAC address of I_j or PC
$MIC(.)$	Message integrity code
K	Network-wide symmetric key
$[1,n]$	Generation range of private key

2.2 PAuthKey 프로토콜

PAuthKey 프로토콜은 ECC 기반의 ECQV 인증서를 발급하여 디바이스 간 상호 인증 및 키 분배에 사용하기 위해 제안되었다. 본 논문에서는 PC가 인증서를 관리하는 인증기관(Certificate Authority)의 역할을 하고 Table 1.에서와 같이 표기한다. ECC는 유한체 E_p(p는 소수)상에 존재하는 타원 곡선 $E(F_p)$ $y^2 \equiv x^3 + ax + b$를 만족하는 포인트들의 집합이다. 타원 곡선 매개 변수는 $(p, a, b, \boldsymbol{G}, n)$으로 표기하며, $\boldsymbol{G}$는 차수가 $n = |E(F_p)|$인 베이스 포인트이다[6].

ECQV 인증서를 이용한 PAuthKey 프로토콜은 먼저 Fig. 2.에서와 같이 PC로부터 공개키를 전송받은 디바이스 I_A가 *Certificate Request* 메시지를 보낸다. *Certificate Request* 메시지에는 인증서 발급을 위해 필요한 $\boldsymbol{R}_A = r_A \cdot \boldsymbol{G}$와 메시지 재생 공격을 방지하기 위한 N_A를 포함한다. 메시지를 수신한 PC는 자체적으로 $r_{PC} \in [1, n]$을 생성하고, $\boldsymbol{Cert}_A$ $(= \boldsymbol{R}_A + r_{PC} \cdot \boldsymbol{G})$와 개인키 복원 데이터 $s (= d_{PC} + r_{PC} \cdot H(\boldsymbol{Cert}_A))$를 계산하여, *Certificate* 메시지에 포함해 디바이스에게 보낸다. 위 메시지들은 사전 분배된 네트워크 키 K로 메시지의 무결성을 보장한다. 인증서를 발급받은 I_A는 자신의 개인키 $d_A (= s + r_A \cdot H(\boldsymbol{Cert}_A))$와 공개키 $\boldsymbol{D}_A (= d_A \cdot \boldsymbol{G})$를 발급받은 인증서로부터 유도할 수 있다. 공개키를 이용한 키 확인 과정은 2-way Handshaking 과정을 통해서 확인되고 I_A는 d_A, $\boldsymbol{D}_A$, $\boldsymbol{Cert}_A$를 소지하게 된다. 이후 I_A는 이웃 디바이스인 I_B와 발급 받은 인증서를 이용하여 상호 인증 및 링크키 설정하고 2-way Handshake과정을 통해 교환된 키 확인 과정을 수행한 뒤 프로토콜이 종료된다[7].

하지만, PAuthKey 프로토콜은 메시지 교환에서 네트워크 키의 사용과 인증서 발급 시 그룹 인증된 디바이스에게 인증서를 발급해주는 보안 취약점이 있다. 공격자가 감염된 디바이스로부터 획득한 네트워크 키를 사용해 PC에게 그룹 인증을 받고 유효한 인증서를 발급받을 수 있는 가장공격에 취약하고 단방향성의 난수 사용은 메시지 재생공격에 취약 하다.

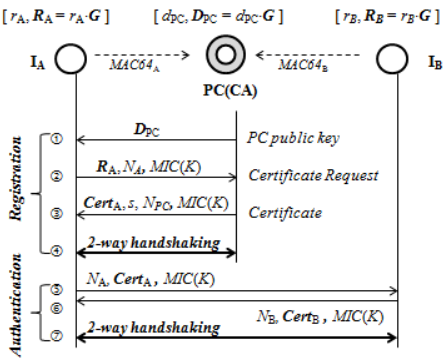

Fig. 2. PAuthKey Protocol

2.3 센서 디바이스의 생명 주기

Fig. 3.은 센서 디바이스의 제작 단계, 설치 및 시운전 단계, 운영 단계로 구성된 생명주기를 보여준다. 제작단계는 펌웨어와 기본적인 매개변수가 저장되며, 디바이스 식별자가 생성되어 NFC(Near Field Communication) 태그에 삽입된다. NFC 태그는 I^2C(Inter Integrated Circuit)포트를 이용해 전기적으로 디바이스와 연결된다. 디바이스 식별과 바인딩 과정은 설치 및 시운전 단계에서 수행된다.

식별과정은 휴대용 시운전 기기를 이용, 디바이스의 물리적 위치와 64bit MAC주소로 구성된 디바이스 식별자를 맵핑시켜 관리 목적으로 관리자의 데이터베이스에 저장한다. 디바이스 바인딩은 동일 네트워크의 두 센서 디바이스 간의 용이한 통신을 위해 제공된다. 바인딩은 직접 결합 의사를 결정할 수 있도록 디바이스에 제공되는 버튼을 조작하거나, 상대 디바이스의 식별자를 삽입해 자동적으로 활성화 시킬 수 있다. 이후 운영 단계에서는 LoWPAN 가입 및 센싱, 액츄에이션 작업을 수행한다.

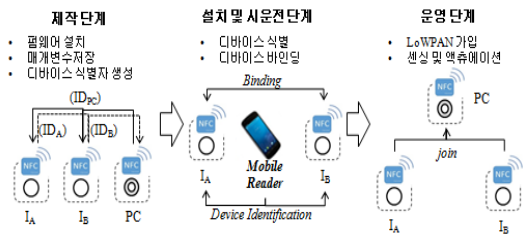

Fig. 3. Device Life Cycle

III. 제안 프로토콜

3.1 설계 원리

본 논문에서 제안한 안전한 가입 및 바인딩 프로토콜의 암호학적 원리는 디바이스 식별자를 ECDH (Elliptic Curve Diffie-Hellman) 공개키의 한 쌍을 이용하여 구성한다는 점이다. 기존 디바이스 식별자와 구별하기 위해 본 논문에서는 보안 디바이스 식별자라고 지칭한다. ECDH 인증은 인증 주체가 각각 자신들의 개인키 $d_A, d_B \in [1,n]$를 생성하고 다음의 식 $\boldsymbol{D}_A = d_A \cdot \boldsymbol{G}, \boldsymbol{D}_B = d_B \cdot \boldsymbol{G}$으로 공개키를 계산한다. 이후 인증 개체 간 공통의 비밀 키를 유도하기 위해 디피헬만 방식으로 ECDH 공개키가 교환된다[8]. 본 논문에서는 ECDH 공개키를 인증과정 뿐만 아니라, 다음 식 $ID_j = H(x_j \| MAC64_j)$, $\boldsymbol{D}_j = (x_j, y_j)$으로 보안 디바이스 식별자의 구성요소로도 사용해 기존 물리적 주소로만 구성된 디바이스 식별자의 문제점을 보완하였다. 만약 보안 디바이스 식별자가 사전에 디바이스들에게 분배되어있다면, 단순히 ECDH 공개키의 교환만으로 상대방의 보안 디바이스 식별자를 유도할 수 있고 사전에 분배된 보안 디바이스 식별자와 비교해 공개키에 대한 무결성을 확인 할 수 있다.

3.2 제안 프로토콜의 설계 가정

안전한 가입 및 바인딩 프로토콜은 다음의 가정을 기반으로 설계된다. 첫 번째, EC 매개변수는 제조과정에서 디바이스에 사전 설치된다. 두 번째, 각 디바이스의 보안 디바이스 식별자는 시운전 단계 중 장치 식별 과정에서 각각의 장치에 계산/저장 된다. PC의 보안 디바이스 식별자는 NFC 태그를 통해 각각 디바이스에 정상적인 설치자에 의해 설치되고 디바이스들의 보안 디바이스 식별자도 PC의 NFC 태그에 설치된다. 세 번째, 설치자는 디바이스 바인딩 과정에 참여하지 않고 바인딩 정보를 PC의 바인딩 테이블에 저장한다. 디바이스가 성공적으로 PC와 가입과정을 수행하면, 바인딩 정보가 디바이스에게 제공된다. 네 번째, PC는 게이트웨이와 동일 위치에 있고, 물리적으로 보호된다. 또한 안전하게 디바이스들의 정보를 대량으로 저장하고 정보 결합의 목적을 위해 게이트웨이 내의 저장 공간에 접근 할 수 있다.

3.3 디바이스 시운전 제안

제조 단계를 수행한 디바이스는 자체 초기화 과정에서 ECDH 개인키 $d_j \in [1,n]$를 생성하고 공개키 $\boldsymbol{D}_j = d_j \cdot \boldsymbol{G}$를 계산한다. 이후 다음의 식 $ID_j = H(x_j \| MAC64_j)$, $\boldsymbol{D}_j = (x_j, y_j)$으로 보안 디바이스 식별자를 계산하고 PC도 같은 식으로 ECDH 개인키, 공개키, 보안 디바이스 식별자를 계산한다. 마지막으로 보안 디바이스 식별자를 각 디바이스의 NFC 태그에 저장함으로써 자체 초기화 과정이 마무리 된다.

과정 1. 디바이스 자체 초기화
- $d_j \in [1,n]$ 계산 후 $\boldsymbol{D}_j = d_j \cdot \boldsymbol{G}$계산.
- $ID_j = H(x_j \| MAC64_j)$, $\boldsymbol{D}_j = (x_j, y_j)$의 식으로 보안 디바이스 식별자를 계산한다.
- 각각의 디바이스 NFC 태그에 ID_j가 저장된다.

자체 초기화 단계가 완료되면 보안 디바이스 식별자는 휴대용 시운전 도구를 이용해 접근할 수 있다. 디바이스는 미리 지정된 물리적인 위치(Loc_j)에 배치되고, 보안 디바이스 식별자와 물리적인 위치를 맵핑시켜 관리목적으로 네트워크 관리자의 데이터베이스에 저장된다. 보안 디바이스 식별자들은 PC의 디바이스 테이블에 저장되고 PC의 보안 디바이스 식별자는 각 디바이스에 저장된다. 마지막으로 디바이스 바인딩 정보도 생성되어 PC의 바인딩 테이블에 저장된다.

과정 2. 디바이스 식별 및 바인딩
- $mapping\_table \leftarrow (ID_j, Loc_j)$
- $device\_table \leftarrow (ID_j, \_, \_)$
  '_'은 안전한 가입 과정에서 설정된다.
- $I_A, I_B, \ldots \leftarrow ID_{PC}$

[과정 2]까지 완료되면 각각의 디바이스는 EC 매개변수 $(p, a, b, \boldsymbol{G}, n)$, ECDH 공개키 $\boldsymbol{D}_j$, 개인키 d_j, 보안 디바이스 식별자 ID_j, PC의 보안 디바이스 식별자 ID_{PC}를 저장한다. PC는 위 보안 요소와 바인딩 테이블을 관리한다. [과정 2]를 통해 안전한 가입을 위한 PC와 각 디바이스 간의 신뢰 관계는 확립되었다.

3.4 안전한 가입 및 바인딩 제안

Fig. 4.에서와 같이 제안 프로토콜은 LoWPAN에서 디바이스가 브로드캐스팅으로 *Beacon Request* 메시지를 보낼 때, PC는 ECDH 공개키 $\boldsymbol{D}_{PC}$와 난수 N_{PC}를 포함한 *Beacon* 메시지를 보낸다. PC에게 메시지를 수신 받은 디바이스는 공개키와 보안 디바이스 식별자에 대한 유효성을 검증하기 위해 자체적으로 $ID_{PC}=H(x_{PC}\|MAC64_{PC})$식을 계산, 시운전 단계에서 저장한 PC의 보안 디바이스 식별자와 비교한다. 보안 디바이스 식별자의 검증이 성공하면, 디바이스는 $\boldsymbol{D}_{PC}$가 PC의 공개키임을 확신 할 수 있고 디바이스는 PC와 사용할 수 있는 링크키를 $LK_A=kdf(\hat{x}, MAC64_{PC}, MAC64_A, N_{PC}, N_A)$, $d_A\cdot \boldsymbol{D}_{PC}=d_A\cdot d_{PC}\cdot \boldsymbol{G}=(\hat{x},\hat{y})$식으로 도출 할 수 있다. 링크키 설정 이후 PC와 디바이스 간 주고받는 메시지의 무결성은 $MIC(LK_A)$으로 보장되고 디바이스는 자신의 ECDH 공개키 $\boldsymbol{D}_A$와 난수 N_A를 포함한 *Association Request* 메시지를 PC에게 전송한다.

과정 3. 안전한 가입

- $I_A \leftarrow PC$: $Beacon(Field_1, \boldsymbol{D}_{PC}, N_{PC})$
- $I_A \rightarrow PC$: $AssociationRequest(Field_2, \boldsymbol{D}_A$ $N_A, MIC(LK_A))$
- $I_A \leftarrow PC$: $AssociationResponse(Field_3, binding(ID_A), N_{PC}, MIC(LK_A))$

Association Request 메시지를 수신한 PC는

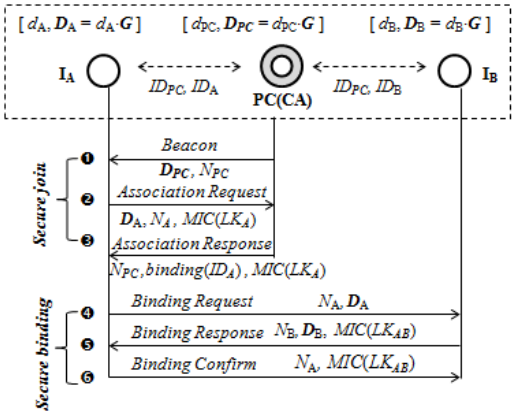

Fig. 4. Proposed Protocol

디바이스의 I_A의 공개키를 이용하여 LK_A를 생성한다. $MIC(LK_A)$값으로 메시지의 유효성을 검사하고 정상적인 메시지일 경우 디바이스 테이블에 디바이스 I_A에 대한 항목을 추가하고 관리한다. 이후 ID_j와 $MAC64_j$로 구성된 바인딩 테이블($binding(ID_A)$)에도 I_A에 대한 정보를 저장하고 관리한다. 바인딩 테이블은 *Association Response* 메시지에 포함되어 I_A로 돌려보내지고 I_A에서도 동일한 방법으로 메시지의 유효성 검증을 한다. 유효성 검증에 성공하게 되면 정상적으로 LoWPAN에 가입하게 된다.

과정 4. 안전한 바인딩

- $I_A \rightarrow I_B$: $BindingRequest(Field_4, \boldsymbol{D}_A, N_A)$
- $I_A \leftarrow I_B$: $BindingResponse(Field_5, \boldsymbol{D}_B, N_B$ $, MIC(LK_{AB}))$
- $I_A \rightarrow I_B$: $BindingConfirm(Field_6, N_A, MIC$ $(LK_{AB}))$

디바이스 간 수행되는 안전한 바인딩은 개시자와 응답자로 구성된다. 안전한 바인딩의 개시자는 상대방의 64bit MAC주소를 보유해야하며, PC의 디바이스 테이블에는 두 디바이스의 항목이 포함되어 있어야 한다. 개시자는 PC로부터 상대방의 보안 디바이스 식별자와 64bit MAC주소를 알고 응답자는 오직 상대방의 보안 디바이스 식별자만 알고 있다. 개시자는 *Binding Request* 메시지에 자신의 $\boldsymbol{D}_A, N_A$를 보내고 응답자는 다음의 식 $ID_A=H(x_A\|MAC64_A)$으로 공개키와 보안 디바이스 식별자에 대한 유효성을 검증한 뒤 $LK_{AB}=kdf(\hat{x}, MAC64_A, MAC64_B, N_A, N_B)$, $d_B\cdot \boldsymbol{D}_A=d_B\cdot d_A\cdot \boldsymbol{G}=(\hat{x},\hat{y})$를 계산한다. 이후 응답자는 *Binding Response* 메시지에 $\boldsymbol{D}_B, N_B, MIC(LK_{AB})$를 포함해 전송한다. 메시지를 수신한 개시자는 동일 방법으로 응답자의 공개키와 보안 디바이스 식별자 검증을 수행하고 *Binding Confirm* 메시지를 통해 응답자와 링크키 LK_{AB}를 공유하게 된다. 안전한 바인딩 과정을 통해 응답자와 개시자는 상호 인증 및 링크키를 설정하였다.

IV. 보안 분석

제안한 프로토콜은 보안 디바이스의 위조가 불가

능하다는 것을 전제로 한다. 보안 디바이스 식별자의 위조는 공격자가 임의의 ECDH 공개키와 개인키의 쌍($\boldsymbol{D}_V', d_V'$)을 생성하고 $ID_V' = H(x_V' \| MAC64_V')$ 식으로 유효한 보안 디바이스 식별자를 생성할 수 있음을 의미한다. 하지만 공격자의 위조 공격의 성공 가능성은 보안 디바이스 식별자의 길이에 기인하여 안전성이 보장된다. 다항식 l과 안전한 매개변수 k로 구성된 $|H(.)| = l(k)$식에서 공격자는 유효한 보안 디바이스 식별자를 생성하기 위해 $2^{l(k)}$번의 계산을 수행해야하며 본 논문에서 사용하는 SHA-1을 기준으로 2^{160}번의 계산을 수행해야 한다. 더욱이 본 논문에서는 제 2 역상 저항성을 만족하는 해쉬 함수를 사용하기 때문에 공격자가 임의의 값을 사용하여 유효한 보안 디바이스 식별자를 생성할 확률은 매우 낮다.

Formal Proof를 위해 해쉬 함수를 다음과 같이 정의 한다. $\boldsymbol{Gen}_{\mathrm{H}}$를 보안 매개 변수 1^n을 입력으로 하고 키 s를 출력하는 Probabilistic Algorithm이라 할 때, 해쉬 함수는 Probabilistic Polynomial Time(PPT) 알고리즘 $\Pi = (\boldsymbol{Gen}_{\mathrm{H}}, H^s)$으로 정의된다. 그리고 $H^s: 0,1^* \rightarrow 0,1^{l(n)}$ 는 알려진 키 s를 사용하는 해쉬 함수이다. Π 알고리즘을 위한 Experiment는 공격자를 A, 보안 매개변수를 k로 정의한다.

〈제 2역상 찾기 Experiment $2PR_{\Pi,A}(k)$〉

• $s \leftarrow \boldsymbol{Gen}_{\mathrm{H}}(1^k)$

• 공격자 A에게 (s, x)가 주어지고, 공격자는 x'를 출력 할 수 있다. $x, x' \in \{0,1\}^*$

• $2PR_{\Pi,A}(k) = 1$ if and only if $x \neq x'$ and $H^s(x) = H^s(x')$

정의 1. 모든 PPT공격자 A에 대해 $Pr[2PR_{\Pi,A}(k) = 1] \leq \mathrm{negl}(k)$을 만족하는 negligible 함수 negl(.)이 존재한다면, 해쉬 함수 $\Pi = (\boldsymbol{Gen}_{\mathrm{H}}, H^s)$는 제 2역상 저항성을 만족한다.

다음과 같이 보안 디바이스 식별자의 생성은 공식화 된다.

구성 1. 보안 디바이스 식별자의 생성

• $(\boldsymbol{D}_V, d_V) \leftarrow \boldsymbol{Gen}_{\mathrm{s}}(1^k)$, $\boldsymbol{Gen}_{\mathrm{s}}$는 보안 매개 변수 1^k을 입력으로 하는 Probabilistic algorithm이고 디바이스를 위한 공개키와 개인키를 출력한다.

• $MAC64_V$가 주어질 때 $ID_V := H^s(x_V \| MAC64_V)$, $\boldsymbol{D}_V = (x_V, y_V)$의 식으로 계산된다.

정리 1. $\Pi = (\boldsymbol{Gen}_{\mathrm{H}}, H^s)$가 제 2역상 저항성을 만족하는 해쉬 함수이면 〔구성 1〕은 위조 공격에 대해 안전하다.

증명. Π'는 $ID_V := H^s(x_V \| MAC64_V)$ 을 나타내고 B는 특정 Π'를 위조하려는 공격자라고 할 때, 다음의 Experiment를 정의한다. □

〈디바이스 식별자 위조를 위한 Experiment, $\mathbf{FORGE}_{\Pi',B}(k)$〉

• 공격자 B에게는 $(s, \boldsymbol{D}_V, MAC64_V)$가 주어지고 $\boldsymbol{D}_V'$를 출력한다.

• $\mathbf{FORGE}_{\Pi',B}(k) = 1$ if and only if $\boldsymbol{D}_V' \neq \boldsymbol{D}_V$ and $H^s(x_V' \| MAC64_V) = H^s(x_V \| MAC64_V)$, $\boldsymbol{D}_V' = (x_V', y_V')$, $\boldsymbol{D}_V = (x_V, y_V)$

공격자 B의 서브루틴을 사용하여 다음의 공격자 A의 $2PR_{\Pi,A}(k)$에서의 Π공격을 고려한다.

〈공격자 A〉

• (s, x)를 입력 받는다. $x = x_V \| MAC64_V$ and $\boldsymbol{D}_V = (x_V, y_V)$.

• $z := x$

• B는 $B(s, z)$를 실행하고, $z' = x_V' \| MAC64_V$를 반환한다. $\boldsymbol{D}_V' = (x_V', y_V')$.

• $x' := z'$

• A는 x'를 출력한다.

$H^s(.)$는 제 2역상 저항성을 만족하기 때문에 $Pr[2PR_{\Pi,A}(k) = 1] \geq Pr[\mathbf{FORGE}_{\Pi',B}(k)]$식과 $Pr[2PR_{\Pi,A}(k) = 1] \leq \mathrm{negl}(k)$식이 성립한다. 결론적으로 $Pr[\mathbf{FORGE}_{\Pi',B}(k)] \leq Pr[2PR_{\Pi,A}(k) = 1] \leq \mathrm{negl}(k)$식이 성립하기 때문에 〔구성 1〕은 위조 공격에 안전하다.

V. 성능 비교 분석

성능 분석은 기존 PAuthKey 프로토콜에 비해 제안 프로토콜에서 변경된 암호 함수 계산과 ECDH

공개키 탑재를 위한 프레임 수정에 따른 패킷 송·수신의 에너지 소모 값을 계산하였다. 실험 환경에 사용된 TelosB는 10KByte RAM, 48KByte ROM을 포함하는 MSP430 16bit 마이크로컨트롤러와 2.4GHz RF주파수 대역, 250kbps 전송비율의 하드웨어 성능을 가진 CC2420으로 구성된다. Fig. 5.은 제안 프로토콜에서 필요한 기능들의 시간 소모 값이다. EC domain은 secp192k1를 사용하였으며, ① Nonce, ② AES-CTR, ③ SHA-1, ④ EC-Scalar, ⑤ HMAC- SHA-1, ⑥ CBC-MAC은 메시지 재생공격 방지, 메시지 암호화, 암호화 해시 함수, 스칼라 곱셈, 키 유도 함수, 메시지 무결성 코드에 각각 사용된다.

TelosB 센서 디바이스에서 측정된 각 함수들의 실행 시간에 기초하여 제안기법과 기존 기법을 비교하기 위해 다음 식$E(Juels) = V \bullet I \bullet T$ 으로 에너지 소모 값을 계산하였다. V는 전압, I는 전류, T는 소비 시간을 나타내고 TelosB 데이터 시트[9]에 따르면 암호 계산에 소비되는 전압과 전류는 각각 3V, 1.8mA를 소비하고 메시지 송·수신에서는 전압 3V, 송신전류 19.5mA, 수신전류 21.8mA를 소비한다. Table 2.1과 Table 2.2는 디바이스 I_A에서 암호 연산과 메시지 송·수신에서의 에너지 소모 값을 프로토콜 진행에 따라 정리한 표이다.

Fig. 6.은 기존 방식과 전체 적인 에너지 소모 값의 비교를 나타내며 메시지 송·수신 부분에서는 메시지 교환횟수를 줄여 283uJ만큼 에너지 소모 값을 절약하였으며 암호 연산 부분에서는 디바이스 간 링크키를 설정을 위한 HMAC 연산 때문에 기존 방식보다 21uJ 만큼 에너지를 더 소모한다. 하지만 전체

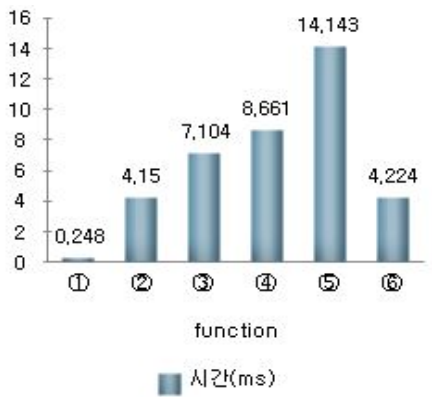

Fig. 5. Time consuming of functions

Table 2.1 Energy Consumption for sending (receiving) the message

	Proposed Protocol						(uJ)
	❶	❷	❸	❹	❺	❻	Total
I_A	85	117	163	87	145	108	705
	PAuthKey						(uJ)
	①	②	③	④	⑤	⑥	⑦	Total
I_A	68	128	168	175	129	145	175	988

Table 3.2 Energy Consumption for function

	Proposed Protocol						(uJ)
	❶	❷	❸	❹	❺	❻	Total
I_A	163	25	22	1	186	24	421
	PAuthKey						(uJ)
	①	②	③	④	⑤	⑥	⑦	Total
I_A		72	107	45	24	107	45	400

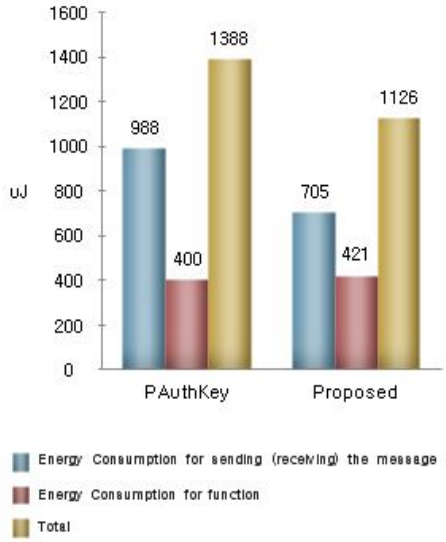

Fig. 6. Energy consumption comparison

적으로 기존방식보다 제안방식이 262uJ만큼 낮은 에너지 소모 값을 보이며 그 이유는 다음과 같다. 첫 번째, 시운전 단계에서 개인키, 공개키, 보안 디바이스 식별자를 사전에 계산하여 프로토콜 진행 중에 연산 부담을 줄였고 두 번째, 기존의 분리되어있던 가입 및 키 설정 단계를 가입 단계로 통합하여 메시지 교환 횟수를 줄여 전체적으로 기존 방법보다 에너지

소모 값을 감소시킬 수 있었다.

VI. 결 론

LoWPAN에 배치되는 센서 디바이스는 필수적으로 가입 과정을 수행한다. 가입 과정에서는 네트워크를 보호하기 위해 PAN Coordinator로부터 인증받은 디바이스만 가입하여야 한다. 하지만 기존 연구에서는 네트워크 키를 이용한 그룹인증과 인증서 발급 과정에서 개별식별이 이루어지지 않는 문제점이 존재하였다. 본 논문에서는 ECDH 공개키의 한 쌍을 이용한 보안 디바이스 식별자를 프로토콜에 적용시켜 가입 단계에서 디바이스 인증 및 PC와 디바이스 간에만 사용하는 링크키를 설정해 그룹인증과 키 사전 분배 문제점을 보완하였다. 이후 안전한 바인딩 과정에서는 디바이스 간 인증서 대신 ECDH 공개키 교환으로 상호 인증과 개별 키를 설정할 수 있다. 또한 제안 프로토콜의 실효성을 검증하기 위해 암호 계산과 프레임 수정에 따른 송·수신에 소비되는 에너지를 계산해 기존 프로토콜과 비교 분석하였다. 분석 결과 디바이스와 PC, 디바이스와 디바이스 간 개별 키를 설정하기 때문에 기존 기법보다 보다 암호 계산에서 에너지 소모 값이 21uJ 증가하였고 분리되어 있던 가입 및 인증과정을 통합시켜 메시지 송·수신에서 에너지 소비 값을 283uJ 감소 시켰다. 전체적인 에너지 소모 값에서는 기존 프로토콜보다 262uJ이 감소되었다. 결론적으로 기존 PAuthKey 프로토콜의 디바이스 그룹인증 및 키 사전 분배의 문제점을 보완하였고 전체 에너지 소모에서 작은 값을 가지기 때문에 자원 제약적인 센서 디바이스 환경에서 적용 가능한 프로토콜을 제시하였다.

References

〔1〕 IEEE std. 802.15.4-2015, Part 15.4:Low-Rate WPANs, pp. 87-92, Dec, 2015.

〔2〕 S. Sciancalepore, S. G. Piro, E. Vogli, G. Boggia, and L. A. Grieco, "On Securing IEEE 802.15.4 Networks through a Standard Compliant Framework," In Proc. of the 2014 Euro Med Telco Conference, pp. 1-6, Naples, Nov. 12-15, 2014.

〔3〕 Certicom Research, "Standard for Efficient Cryptography," SEC 4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV) Version 0.97. pp. 5-12, Aug, 2013.

〔4〕 P. Porambage, C. Schmitt, P. Kumar, A. Gurtov, and M. Ylianttila, "PAuthKey: A Pervasive Authentication Protocol and Key Establishment Scheme for Wireless Sensor Networks in Distributed IoT Applications," International Journal of Distributed Sensor Networks, vol. 2014, Article ID 357430.

〔5〕 G. Piro, G. Boggia, and L.A. Grieco, "A Standard Compliant Security Framework for IEEE 802.15.4 Networks," in Proc. of the IEEE World Forum on Internet of Things, pp. 27-30, Seoul, Mar. 6-8, 2014.

〔6〕 Certicom Research, "Standard for Efficient Cryptography," SEC 1: Elliptic Curve Cryptography Version 2.0. pp. 15-21, May 21, 2009.

〔7〕 S. Sciancalepore, A. Capossele, G. Piro, G. Boggia, and G. Bianchi, "Key Management Protocol with Implicit Certificates for IoT systems," In Proc. of the 2015 Workshop on IoT challenges in Mobile and Industrial Systems, pp. 37-42, Florence, May 18-22, 2015.

〔8〕 C. Lederer, R. Mader, M. Koschuch, J. Großschadl, A. Szekely, and S. Tillich, "Energy-Efficient Implementation of ECDH Key Exchange for Wireless Sensor Networks," in *Information Security Theory and Practice – WISTP 2009*, LNCS, vol. 5746. Springer Verlag, pp. 112 - 127, 2009.

〔9〕 Instrument and Technology, "K-mote Data sheet," pp. 7-8, Oct 5, 2008.

〈저자소개〉

이 종 훈 (Jong-Hoon Lee) 학생회원
2016년 2월: 공주대학교 정보통신학과 졸업
2016년 3월~현재: 단국대학교 소프트웨어보안 석사 과정
〈관심분야〉 정보보호, 네트워크 보안

박 창 섭 (Chang-seop Park) 종신회원
1983년 2월: 연세대학교 경제학과 졸업
1987년 2월: Lehigh University 컴퓨터과학과 석사
1990년 2월: Lehigh University 컴퓨터과학과 박사
1990년 3월~현재: 단국대학교 소프트웨어학과 교수
〈관심분야〉 정보보호, 네트워크 보안, 무선인터넷 및 모바일 컴퓨팅 보안