인증 및 경량화 암호알고리즘 기반 IoT 보안 서비스

김선집*
한세대학교 IT학부 교수

A IoT Security Service based on Authentication and Lightweight Cryptography Algorithm

Sun-Jib Kim*
Professor, Div. of Information Technology, Hansei University

요 약 IoT 시장은 지속적으로 확대 성장하고 있지만, IoT 기기에 대한 보안 위협 또한 증가하고 있다. 그러나 자원 한정의 문제점을 가지고 있는 IoT 기기에 기존 시스템에 적용되었던 보안기술을 적용하는 것은 어렵다. 이에 본 논문에서는 IoT 기기의 자원 한정이라는 특성 하에서 보안 기능 적용에 따른 오버헤드를 줄일 수 있는 인증 및 경량 암호알고리즘 적용하여 IoT 기기의 보안성을 향상시킬 수 있는 서비스를 제시하여 IoT 기기가 제공되는 홈네트워크 등에 보안성을 제공하고자 한다. 이에 본 논문에서 제시하고 있는 인증 및 경량 암호알고리즘 적용 서비스는 기존 연구에서 증명되었던 IoT 환경에서 적용 가능한 LEA 암호화 알고리즘의 이용과 더불어 비밀키 생성에 있어 이용자, IoT 기기와 서버가 참여하여 3자의 상호인증기반 비밀키 생성을 통해 서비스의 안전성을 확보하였으며 실험에서 랜덤하게 비밀키를 생성하는 방식과 속도의 차이가 없으나, 부가적으로 비밀키 공유를 위한 로직을 IoT 기기에 적용하지 않음으로써 IoT 기기의 자원 한정의 문제점을 해결할 수 있음을 검증하였다.

주제어 : IoT, 경량화 암호알고리즘, 접근 통제, 보안

Abstract The IoT market continues to expand and grow, but the security threat to IoT devices is also increasing. However, it is difficult to apply the security technology applied to the existing system to IoT devices that have a problem of resource limitation. Therefore, in this paper, we present a service that can improve the security of IoT devices by presenting authentication and lightweight cryptographic algorithms that can reduce the overhead of applying security features, taking into account the nature of resource limitations of IoT devices. We want to apply these service to home network IoT equipment to provide security. The authentication and lightweight cryptographic algorithm application protocols presented in this paper have secured the safety of the service through the use of LEA encryption algorithms and secret key generation by users, IoT devices and server in the IoT environment. Although there is no difference in speed from randomly generating secret keys in experiments, we verify that the problem of resource limitation of IoT devices can be solved by additionally not applying logic for secret key sharing to IoT devices.

Key Words : IoT, Lightweight encryption algorithm, Access control, Security.

*교신저자 : 김선집(kimsj@hansei.ac.kr)
접수일 2020년 12월 15일 수정일 2021년 1월 14일 심사완료일 2021년 1월 18일

1. 서론

전 세계 IoT 시장 규모는 2017년 이후 산업 전 분야에서 IoT 기기에 대한 수요로 인하여, 두 자릿수의 연간 성장률을 유지하고 있다[1].

최근 정부 및 산업 분야를 포함하여 일반 소비자의 주변 생활 환경에도 IoT 기기의 도입이 증가하고 있다. 산업 분야의 경우 기존의 노동 중심에서 IoT 기기들 간의 연결을 통해 시스템을 자동화함으로써 생산의 원가를 절감하는데 활용되고 있다. 연결된 IoT 기기들 간에 생성된 데이터를 통해, 실시간 의사결정에 도움을 줄 수 있을 뿐만 아니라 새로운 서비스와 비즈니스를 만들어 내고 있다. 일반 소비자의 경우 헬스 등 건강 관련 서비스와 홈네트워크, 가족 구성원들에 대한 정보 교류에도 활용되고 있다.

이러한 IoT 분야의 성장세가 이어지고 있는 반면에 IoT 기기를 대상으로 하는 사이버 위협 또한 증가하고 있다. 그러나 낮은 전력, 대역폭 등 기존 시스템과 비교해서 낮은 성능을 가지는 IoT 기기에 기존 시스템에 적용되던 보안기술을 적용하기에는 한계가 있다. 그 결과 기존 위협요소와 IoT의 특성을 반영한 위협요소에도 노출되어 있다.

IoT 기기의 제한적인 환경에서 사용하는 CoAP와 MQTT 같은 경량 메시징 프로토콜도 네트워크상의 보안 위협요소들에 노출될 수 있는 취약성이 크다[2]. 이러한 경량 메시징 프로토콜은 메시지 전송에만 초점이 맞추어 설계된 프로토콜로서 표준 보안기술이나 보안정책의 적용이 부족하다[3].

IoT 기기의 보안 문제점을 해결하기 위해 한국인터넷진흥원(KISA)에서는 IoT 보안인증 대상 및 등급에 관련된 시험인증기준을 수립하고 이에 따라 등급별 보안성 확보기준을 제시하여 관련 요구사항을 만족하는 제품에 대해 인증을 부여함으로써 IoT 제품의 보안 신뢰성을 확보하고 있다.

본 연구에서는 KISA 시험인증기준을 고려하여 IoT 환경에 맞는 경량 암호알고리즘 기반의 IoT 보안 서비스를 제안 및 구현하고 시험을 통해 그 성능을 평가하였다.

본 논문의 2장에서는 관련 연구를, 3장에서는 본 논문에서 제안 및 구현한 보안 서비스를 설명하였다. 4장에서는 기존 방식과 본 논문에서 제시한 방안에 대한 성능을 비교하였다, 5장에서는 결론 및 시사점을 제시한다.

2. 관련연구

2.1 IoT 응용 프로토콜

MQTT는 OASIS에 의해 2013년에 IoT표준 메시지 전송 프로토콜로 지정, 2016년도에 ISO표준(ISO/IEC PRF 20922)으로 채택된 TCP 기반의 경량 메시지 전송 프로토콜이다[4]. MQTT는 IoT 통신을 위한 신뢰성을 제공하는 프로토콜로 주목받고 있으며 다양한 IoT 플랫폼 및 서비스에서 지원되고 있다. 다양한 분야에서 IoT 통신에 적용하고 있으며, 최근에는 클라우드 서비스와도 연동되고 있다.

MQTT는 Publish/Subscribe 구조로 비동기화 방식을 지원한다. MQTT는 Broker가 존재하여 Publisher와 Subscriber 사이에 메시지를 중개한다. 메시지는 Topic이라는 메시지 통신 채널을 이용하여 전달되며 메시지 전송 신뢰성을 보장하기 위해 QoS 레벨을 설정할 수 있다. 그러나 이러한 MQTT 프로토콜은 IoT 환경에서 평문의 메시지를 전송함에 따라 기밀성을 제공하지 못하여 이에 대한 해결책을 제공하기 위해 메시지 전송에 있어 암호화 방식을 적용하여 전송하는 방식이 연구되고 있다[5].

CoAP은 2014년 IETF(Internet Engineering Task Force)에 의해 제정된 UDP 기반의 유니캐스트와 멀티캐스트 양쪽을 지원하는 경량 REST 프로토콜로서 6LoWPAN의 응용계층에서 저전력과 저사양의 IoT 기기에 적합하게 설계한 M2M(Machine-to-Machine)을 위한 경량 메시지 전송 프로토콜이다.

CoAP은 서버와 클라이언트 구조로 되어 있으며, 기본적으로 일대일 '요청·보고' 인터렉티브 모델을 제공하며, IPv6 환경에서는 멀티캐스트를 지원할 수 있도록 설계되었다. HTTP 및 RESTful웹과 상호 운용이 가능하도록 설계되어 있어 기존의 인터넷에 적합하다. 이러한 CoAP는 일관된 연결 대신 반복적 메시징에 의존해 신뢰성을 제공하며 일대다 또는 다대다 멀티캐스트 요구사항을 지원하고, UDP 전송 프로토콜 상에서 DTLS를 사용하여 보안성을 제공 한다[6].

2.2 암호화 알고리즘

AES는 미국의 NIST에 의해 표준화되어 널리 사용되는 블록 암호화 알고리즘으로, 128비트 블록 크기를 바탕으로 128, 192, 256비트의 키 길이의 사용을 지원한다. 라운드 함수 내에서 8비트 단위의 연산을 통한 암호화와 복호화를 지원한다[7].

LEA는 Hong 등이 제안하였으며, Feistel 구조를 적용하고 있다. 키 스케쥴 특성에 의한 이론적 취약성이 존재하지 않으며, 128비트 블록 크기를 바탕으로 128, 192, 256비트 키 길이를 지원한다. LEA는 ARX (Additon Rotation XOR) 구조를 채택하고 있으며, LEA의 라운드 함수는 32비트 단위의 ARX 연산만으로 구성되어 있다. 이러한 LEA는 IoT, 클라우드 등 경량 환경에서 기밀성을 제공하기 위한 블록 암호알고리즘으로 활용되고 있다. 경량, 저전력의 특성이 있는 IoT 기기들을 고려하여 계산 자원을 효율적으로 사용하면서 충분한 성능과 보안성을 제공하는 데 적합하게 설계되었다[8,9].

RSA는 공개키와 개인키의 한 쌍의 키로 암호화 및 복호화하는 공개키 기반 대표 암호화 알고리즘으로 Ron Rivest, Adi Shamir, Leonard ADleman에 의해 개발되었으며, 큰 소수(prime number)의 곱으로 이루어진 정수의 소인수 분해가 어렵다는 사실 기반, 암·복호화와 더불어 전자서명이 가능한 암호방식이다[10].

ECC는 Victor Miller와 Neil Koblitz에 의해 제안된 공개키 암호방식으로 타원곡선 상의 임의의 두 점 P와 Q를 알더라도 비밀키로 사용된 임의 정수 k를 알아내기 어렵다는 타원곡선 군(group)의 이산대수 문제에 안전성을 기반으로 설계되었다. 이는 다른 암호화 알고리즘과 비교 시 짧은 키의 길이도 대등한 안전도를 제공하는 장점이 있다. ECC 암호 시스템을 구현하기 위해서는 키분배 알고리즘과 메시지 암호알고리즘으로 구성되어야 한다. 이에 난수와 결합한 공개키를 각 단말에 공유하여 공격자가 유추할 수 없는 비밀키로 동기화하고 암호화하는 순서로 진행된다. 이에 키 분배 대표 방식은 ECDH (Elliptic Curve Diffie-Hellman)알고리즘이다. 메시지 암호화 방식은 비밀키를 계산한 후 이를 송신자와 수신자가 진행한다[11].

2.3 IoT 보안인증

한국인터넷진흥원(KISA)에서는 IoT 시장의 활성화와 더불어 보안 문제점을 해결하기 위해 IoT 보안인증 대상 및 등급에 관련된 시험인증기준을 수립하고 이에 따라 등급별 보안성 확보기준을 제시하여 관련 요구사항을 만족하는 제품에 대해 인증을 부여함으로써 IoT 제품의 보안 신뢰성을 확보하고 있다. 이에 〈Table. 1〉은 한국인터넷진흥원의 IoT 보안 시험인증기준에 명시되어 있는 등급별 요구되는 보안사항에 대한 내용이다[13].

펌웨어 기반의 소형제품의 경우 암호화 부분에 있어 IoT 제품에는 Lite등급을 모바일 앱은 Basic등급을 적용받도록 권장하고 있다. 이에 Lite등급과 Basic등급에서의 보안 요구사항은 중요정보 전송 또는 저장 시 안전한 암호알고리즘을 사용하면 인증 조건을 만족하며, 안전한 키 관리와 난수 생성은 적용받지 않고 있다.

〈Table 1〉 Feature comparison between Lite, Basic and Standard Level(Lite, Basic, Standard feature comparison)

Rating	Contents	Apply
Lite	Minimum action items to maintain product security certification: Users, products Application of encryption algorithm when storing important information, Data in transit protection	Suitable for small firmware-based products such as sensors
Basic	Key measures required to improve the reported vulnerabilities of hacking cases, etc. certification : Users, products Application of encryption algorithm when storing important information, Data in transit protection	Suitable for small and medium-sized products with low specification OS
Standard	Comprehensive security measures items at the level of international requirements	Suitable for medium to large sized smart home appliances

3. 제안하는 서비스

3.1 제안하는 서비스 구조

본 논문에서 제안하는 보안 기능이 적용되는 IoT 서비스는 이용자와 IoT 기기의 인증 및 전송 데이터의 암호화에 중점을 두고 설계하였다. 이는 IoT 기기의 제안된 자원의 현황에서, 최근 IoT 보안의 이슈를 고려함과 더불어 한국인터넷진흥원의 IoT 보안 시험 인증기준에서 중요시되고 있는 인증 및 전송 데이터 보호에 암호알고리즘을 적용하여 인증기준에 준하는 보안 성능을 제공하기 위함이다.

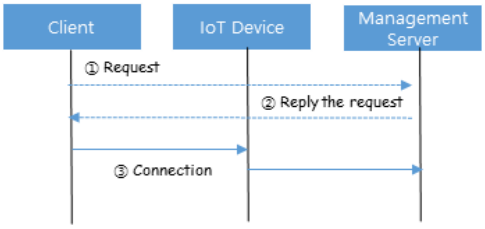
이러한 요구사항에 본 논문에서 제안하는 서비스는 경량 암호알고리즘인 LEA를 적용하고 인증과 암호화를 위한 비밀키 생성에 있어 이용자(Client), IoT 기기(IoT Device) 및 서비스 관리 서버(MS)간 상호 신뢰성 기반 프로세스를 구성 반영하였다.

[Fig 1]은 본 논문에서 제안하는 서비스에 대한 대략적인 구성도이다.

일반적으로 IoT 기기를 활용하고자 하는 이용자는 스마트폰과 같은 무선기기를 소장하고 있으며, IoT 기기는 기존의 기계적인 장비에 탑재되어 사물을 네트워크에 연

결하여 특정한 서비스를 제공할 수 있으며, 관리 서버는 데이터베이스 및 웹 기반 서비스를 통해 인가된 이용자가 IoT 기기를 이용할 수 있게 한다.

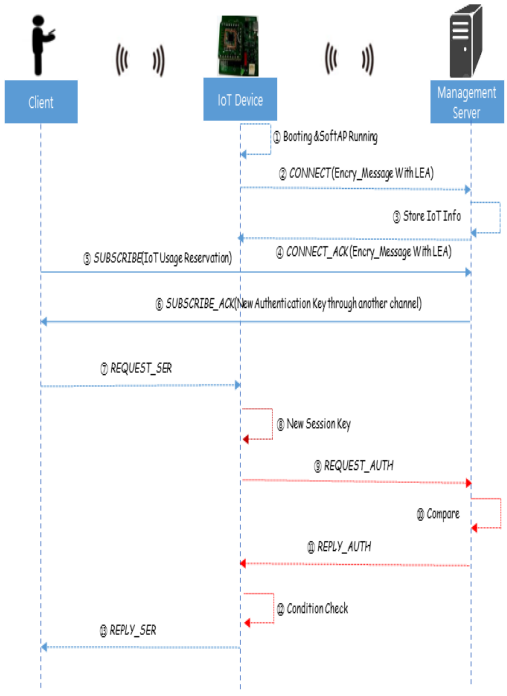
[Fig. 1]는 본 논문에서 제시하는 서비스 구조로 이용자가 관리 서버에 서비스를 요청하고 이에 관리 서버가 서비스를 허용하면, 이용자는 IoT 기기와 관리 서버가 제공하는 IoT 서비스를 이용할 수 있게 된다.



[Fig. 1] A structure of IoT service

3.2 제안하는 서비스의 메시지 구조

제안하는 프로토콜의 작동 구성은 [Fig. 2]와 같다.



[Fig. 2] A Process of IoT service

제안하는 서비스의 경우 중앙에서 IoT 기기에 대한 서비스를 관리하는 관리 서버가 존재하여 일반 이용자가 모바일 디바이스를 통해 IoT 기기가 부착되어 있는 장비를 쉽고 안전하게 사용할 수 있는 환경을 제공하도록 하였다.

이에 기본적으로 이용자, IoT 기기 및 관리 서버간 신뢰 기반의 인증을 활용하여 비밀키를 생성하는 방법을 제안하였으며, 서로 상호 간 데이터 전송과 더불어 데이터의 암호화 저장에 경량 암호화 알고리즘인 LEA를 적용하였다.

서비스의 구현 프로세스는 다음과 같다.

① 1단계 : IoT 기기가 부팅되며, 이때 사전의 관리자에 의해 세팅된 pre-session key를 관리 서버와 공유한다. 또한, SoftAP의 기능이 작동됨으로써 이용자가 핸드폰과 같은 모바일 디바이스를 이용하여 무선랜을 통해 IoT 기기가 지원하는 사물과 연결된다.

② 2단계 : LEA 암호알고리즘에 pre-session key를 활용하여 관리 서버에 접속 요청과 IoT 기기의 MAC주소, SoftAP의 SSID의 값을 전송한다.

③ 3~4단계 : 관리 서버에서 IoT 기기에서 암호화되어 전송된 패킷을 사전에 공유하고 있던 pre-session key를 활용하여 복호화한 후에 MAC 주소와 SoftAP의 SSID를 저장하고 IoT 기기에 응답 메시지를 전송한다.

③ 3~4단계 : 관리 서버에서는 IoT 기기에서 암호화하여 전송한 패킷을 사전에 공유하고 있던 pre-session key를 활용하여 복호화한 후에 MAC 주소와 SoftAP의 SSID를 저장하고 IoT 기기에 응답 메시지를 전송한다.

④ 5단계 : 이용자가 IoT 기기를 제어할 수 있는 서비스를 관리 서버가 별도로 제공하는 웹 서비스에 요청한다.

⑤ 6단계 : 관리 서버가 다른 통신 채널(예 : 문자 메시지)을 이용하여 서비스를 이용할 수 있는 OTP 값(Random 값)을 이용자에게 전송한다.

⑥ 7단계 : 이용자가 관리 서버로부터 전송받은 OTP 값을 이용하여 IoT 기기의 SoftAP를 통해 서비스에 접속한다.

⑦ 8~9단계 : IoT 기기에서 new-session key를 생성하고, LEA 암호화 알고리즘을 이용하여 이용자의 요청을 관리 서버에 전달한다.

⑧ 10~11단계 : 관리 서버에서 이용자로부터 요청한 사항에 대해 new-session key를 생성하고, LEA

로 복호화하여 요구사항을 복구한다. 이에 해당 메시지가 복호화되면 IoT 기기에 관련 응답 메시지를 보낸다.

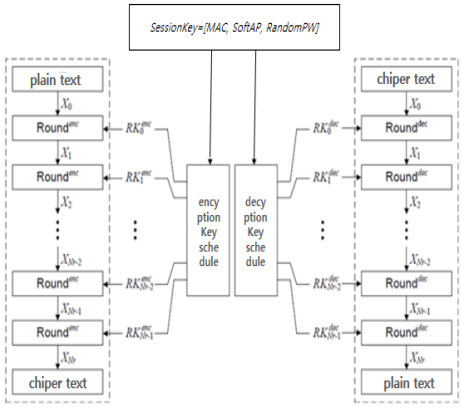
⑨ 12~13단계 : IoT 기기에서 현재 IoT 기기 상태에 따라 서비스 가용 여부를 이용자에게 응답한다.

[Fig. 2]에서 사용되는 메시지는 〈Table. 2〉와 같이 정의할 수 있다. 〈Table. 2〉의 1, 2의 메시지는 이용자가 IoT 기기 서비스를 이용하기 전에 IoT 기기와 관리 서버 간의 서비스가 연계시 사용되는 메시지다. 이용자가 IoT 기기와 관리 서버가 지원해 주는 서비스를 이용하기 위해 이용자를 인증하기 위하여 다른 통신 채널을 통해 이용자와 관리 서버간의 이용자에 대한 랜덤 인증값 전송에 3, 4의 메시지가 사용되며, 이후 이용자와 IoT 기기 인증을 통한 비밀키 과정에서 관리 서버간의 암호화 통신을 통한 이용자 및 IoT 기기에 대한 인증에는 5~8까지의 메시지가 사용된다.

〈Table 2〉 A definition of Actions or Messages

No	Name	Description
1	CONNECT	Request to connect
2	CONNECT_ACK	Connect Acknowledgement
3	SUBSCRIBE	Subscribe request
4	SUBSCRIBE_ACK	Subscribe Acknowledgement
5	REQUEST_SER	Request Connect Service
6	REQUEST_AUTH	Request Authentication
7	REPLY_AUTH	Reply Authentication
8	REPLY_SER	Reply Connect Service

본 서비스에서의 중요정보의 저장 및 데이터 전송시의 암호화 알고리즘은 IoT의 특성을 고려하여 LEA를 사용한다. 다만 IoT 기기의 특성상 암호화 알고리즘을 최소화하기 위해 LEA를 사용한 만큼 비밀키를 지속적으로 변경해 주기 위해 사용하는 알고리즘은 두 가지로 구분한다. 첫 번째로 사용되는 pre-session key는 본 서비스의 관리자가 IoT 기기를 설치 시 초기화하여 셋팅을 하는 방식을 선택한다. 이후 서비스별 IoT MAC address(6byte), SoftAP Password(4byte)와 Random(6byte)를 활용 프로그램화된 배열함수에 의해 128bit new-session key를 계산하여 [Fig. 3]과 같이 암호화와 복호화에 적용하여 사용한다. 이에 새로운 서비스별 new-session key를 이용함으로써 서로 다른 이용자를 구분하고 이용자, IoT 기기와 관리 서버간 상호 인증기반으로 비밀키가 생성될 수 있도록 구현하였다[14].



[Fig. 3] A structure of Encyption, Decryption relationship with Secret key

본 논문에서 제안한 비밀키는 LEA 128bit 암호화 방식에 맞추어 설계되었으며, 암호화 강도를 강화하기 위해 192, 256비트 비밀키도 생성할 수 있다.

암호화와 복호화를 위해서는 신뢰 관계를 기반으로 이용자, IoT 기기, 관리 서버 중 하나라도 그 신뢰 관계가 형성되지 않아 비밀키 값이 일치하지 않으면 암호화된 메시지가 복호화되지 않아 상호 간의 통신이 인가되지 않는다.

4. 성능 비교

본 논문에서는 대표적인 대칭키 알고리즘인 AES 기반 랜덤 키 생성방식을 사용하여 짧은 평문을 128 비트의 크기로 변환하는 방식과 LEA기반 본 논문에서 제안하는 키 생성 스킴을 반영한 방식에 대해 키 생성시간, 암호화 시간, 복호화 시간 및 총 암호화에 걸린 시간을 측정하여 비교 분석하였다.

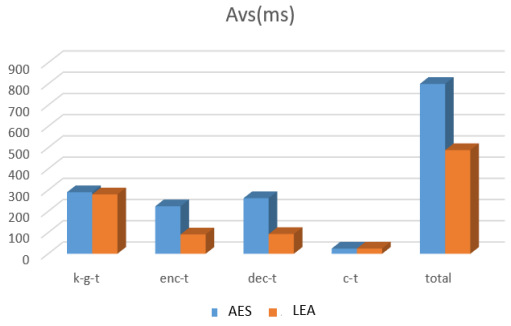
또한, 통신 구간에서의 지연 시간을 측정하였다. 다만, 관리 서버 측에서 메시지의 암호화 및 복호화는 서버의 성능에 따라 그 속도가 좌우되게 됨에 따라, IoT 기기에서만 그 값을 측정하였다. 시험에 사용된 IoT 기기의 규격은 〈Table. 3〉과 같다[15].

[Fig. 4]는 성능 비교 대상항목에 대해 30회씩을 수행한 시간의 평균값을 나타낸다. *k-g-t*는 키를 생성하는 시간의 평균값을 의미하며, *enc-t*는 암호화 시간의 평균값, *dec-t*는 복호화 시간의 평균값, *c-t*는 통신에 걸린 한 방향의 평균값을 의미한다. 이에, 특정 서비스의 통신 또는

〈Table 3〉 ESP 8266 SPEC

Spec.	Detail
Operation Voltage	3.3V
CPU	Tensilica L106 32bit running at 80 MHz or 160MHz
Current Consumption	10uA ~ 170mA
RAM	32K+80K
Input	17 GPIO pins
Network support	802.11
Maximum concurrent TCP connection	5

인증을 위해서 특정 데이터를 암호화하여 관리 서버로 전송하고 다시 이에 대한 응답 값을 받아야 하므로 IoT 기기 입장에서 데이터의 전송과 지연 시간은 본 측정값의 최소 2배 이상 소요된다고 볼 수 있다.



[Fig. 4] Compare of Average times for AES & LEA encryption, decryption and key generation(*ms*)

[Fig. 4]와 같이 기존의 방식이 본 논문에서 제시하는 방식에 비해, 암호화 시 2.4배 정도의 시간이 더 소요되었으며, 복호화 시 2.8배가 더 소요되었고, 비밀키 생성에서는 본 성능평가가 하나의 IoT 기기만을 한정하여 측정함에 따라 기존 랜덤 방식과 본 연구에서 제시한 방식과의 차이점은 거의 없었으나 5% 정도의 감소 효과를 확인하였으며, 전체 소요시간은 본 연구에서 제안한 방식이 AES와 랜덤 비밀키 생성 방식에 비해 40%정도의 시간을 줄일 수 있었음을 확인하였다. 또한 비밀키 생성을 위한 암호화 알고리즘을 IoT 기기에 적용하지 않아 IoT 기기의 자원 한정의 문제점도 해결할 수 있었다.

5. 결론

본 연구에서는 자원의 제안사항을 가지고 있는 IoT 기기에 한국인터넷진흥원의 IoT 보안 시험인증기준을 충족하기 위하여 암호화 알고리즘에 LEA을 활용하였으며, 비밀키 생성에 있어 이용자, IoT 기기 및 서비스 제공 서버 간의 신뢰를 기반으로 비밀키 생성 방식을 제안 적용하였으며 그 성능을 평가하였다.

실험 결과, 특정 IoT 기기에서 기존 방식보다 40% 이상 속도가 빨라짐을 확인하였다. 이는 제한된 자원을 가진 IoT 기기에서 인증과 데이터의 암호화 등의 기밀성을 확보할 수 있음을 의미한다.

또한, 본 연구에서 제시한 비밀키 생성 방식은 기존의 방법과 비교하여 특정 환경에 적합한 비밀키 생성 패턴을 제시함으로써 인증과 비밀키 생성의 두 가지 목표를 달성할 수 있었다. 이는 비밀키 생성의 편리성과 키 관리의 효율성 그리고 악의적인 의도를 가지고 접근하는 공격자에 대한 보안성을 확보할 수 있을 것으로 생각된다. 최근 저가격 고성능의 다양한 IoT 기기들이 개발되고 있는 상황에서 기존 유 무선 네트워크에서 지원되었던 강도의 보안성을 IoT 기기에 확보하기 어려운 IoT 기기 기반 객실관리 시스템 및 사물함 관리 시스템 등에 본 논문의 연구 결과를 적용한다면 다양한 IoT 시스템에도 한국인터넷진흥원의 IoT 보안 시험인증기준에서 요구하는 보안 성능을 제공할 수 있을 것으로 기대한다.

REFERENCES

[1] IDC [Internet] https://www.idc.com/getdoc.jsp?containerId=IDC_P29475, Worldwide Internet of Things Spending Guide.

[2] N.H.Kang, "Standard technology trends for IoT security", Information and Communication Magazine, Vol.21, No.9, pp.40-45, 2014.

[3] S.R.Oh and Y.G.Kim, "Security Analysis of MQTT and CoAP protocols in the IoT Environment" Proceeding of the Korea Information Processing Society Conference, Vol.23, No.1, pp.297-299, 2016.

[4] ISO/IEC, "Information technology - Message Queuing Telemetry Transport(MQTT) V3.1.1. "ISO/IEC 20922:2016, 2016.

[5] H.Y.Kim and J.N.Kim, "A Study of End-to-End Message Security Protocol Based on Lighweight Ciphers for Smart IoT Devices", Journal of The Korea Institute of Information Security & Cryptology, Vol.28, No.6, 2018.

[6] Zach Shelby, Klaus Hartke, and Carsten Bormann, "The Constrained Application Protocol(CoAP)", IETF RFC 7252, 2014.

[7] J. Daemen and V. Rijmen, "The design of Rijndel:AES-the advanced encryption standard", Springer, 2013.

[8] D. Hong, J. Lee, D. Kim, D. Kwon, K. Ryu. and D. Lee, "LEA: A 128-Bit Block Cipher for Fast Encryption on Common Processors", ISA. LNCS Vol.8267, pp.3-27, Springer, 2013.

[9] J.M.Jeong, P.H.Kim, KY.Jung, E.J.Yoon and K.Y.Yoo, "Key Management Method for LEA Lightweight Block Cipher", Proceedings of Symposium of the Korean Institute of communications and Information Sciences, pp.959-960, 2017.

[10] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining Digital Signatures and Public-Key Crypto-systems", Communications of the ACM, Vol.21, No.2, pp.120-126, 1978.

[11] N. Koblitz, "Elliptic curve cryptosystems", Mathematics of Computation, Vol.48, No.177, pp.203-209, 1987.

[12] K.H.Lee, "A Scheme on Anomaly Prevention for Systems in IoT Environment", Journal of The Korea Internet of Things Society, Vol.5, No.2, pp.8195-101, 2019.

[13] KISIS [Internet] https://www.kisis.or.kr/kisis/subIndex/307.do

[14] Namuwiki [Internet] https://namu.wiki/w/LEA

[15] Wikipedia [Internet] https://en.wikipedia.org/wiki/ESP8266

김 선 집(Sun-Jib Kim) [정회원]

■ 2001년 2월 : 숭실대학교 컴퓨터학과 (공학석사)
■ 2010년 2월 : 한세대학교 IT학과 (공학박사)
■ 2014년 3월 ~ 현재 : 한세대학교 IT학부 ICT 융합학과 교수

〈관심분야〉
정보보안, 사물인터넷, 클라우드, 환경시스템