저 사양 IoT 장치간의 암호화 알고리즘 성능 비교

박정규[1], 김재호[2*]
[1]창신대학교 컴퓨터소프트웨어공학과 교수, [2]경상국립대학교 항공우주및소프트웨어공학부 & AI융합공학과 교수

Comparison of encryption algorithm performance between low-spec IoT devices

Jung Kyu Park[1], Jaeho Kim[2*]
[1]Professor, Department of Computer Software Engineering, Changshin University
[2]Professor, Dept. of Aerospace and Software Engineering & AI Convergence Engineering, Gyeongsang National University

요 약 사물인터넷(IoT)은 다양한 플랫폼, 컴퓨팅 성능, 기능을 가지는 장치를 연결한다. 네트워크의 다양성과 IoT 장치의 편재로 인해 보안 및 개인 정보 보호에 대한 요구가 증가하고 있다. 따라서 암호화 메커니즘은 이러한 증가된 요구 사항을 충족할 만큼 충분히 강력해야 하고 동시에 저 사양의 장치에 구현될 수 있을 만큼 충분히 효과적이어야 한다. 논문에서는 IoT에서 사용할 수 있는 다양한 유형의 장치에 대한 최신 암호화 기본 요소 및 체계의 성능 및 메모리 제한 사항을 제시한다. 또한, IoT 네트워크에 자주 사용되는 저 사양의 장치에서 가장 일반적으로 사용되는 암호화 알고리즘의 성능에 대한 자세한 성능 평가를 수행한다. 데이터 보호 기능을 제공하기 위해 바이너리 링에서 암호화 비대칭 완전 동형 암호화와 대칭 암호화 AES 128비트를 사용했다. 실험 결과 IoT 장치는 대칭 암호를 구현하는데 충분한 성능을 가지고 있었으나 비대칭 암호 구현에서는 성능이 저하되는 것을 알 수 있다.

주제어 : 사물인터넷, BLE, mbed 플랫폼, 암호학, 바이너리 숫자 링

Abstract Internet of Things (IoT) connects devices with various platforms, computing power, and functions. Due to the diversity of networks and the ubiquity of IoT devices, demands for security and privacy are increasing. Therefore, cryptographic mechanisms must be strong enough to meet these increased requirements, while at the same time effective enough to be implemented in devices with long-range specifications. In this paper, we present the performance and memory limitations of modern cryptographic primitives and schemes for different types of devices that can be used in IoT. In addition, detailed performance evaluation of the performance of the most commonly used encryption algorithms in low-spec devices frequently used in IoT networks is performed. To provide data protection, the binary ring uses encryption asymmetric fully homomorphic encryption and symmetric encryption AES 128-bit. As a result of the experiment, it can be seen that the IoT device had sufficient performance to implement a symmetric encryption, but the performance deteriorated in the asymmetric encryption implementation.

Key Words : IoT, BLE, mbed Platform, Cryptography, Binary Number Ring,

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2021R1F1A1052129)
*교신저자 : 김재호(jaeho.kim@gnu.ac.kr)
접수일 2021년 12월 22일 수정일 2022년 2월 9일 심사완료일 2022년 2월 12일

1. 서론

사물인터넷(Internet of Things)은 우리 삶에 밀접하게 들어와 있으며 전 세계 수십억 명의 사람들이 사용하고 있다. 그러나 연결된 장치의 수가 증가하면서 인명에 대한 물리적 피해에서 가동 중지 시간 및 장비 손상에 이르기까지 보안 위험이 증가하고 있다. 예로 전기를 생산하는 발전소 또는 급수를 위한 정화 시스템이 문제가 될 수 있다. 이러한 IoT 시설이나 시스템은 이미 공격받고 상당한 피해를 보았기 때문에 시스템 보호가 최우선이다. IoT 기술이 발전함에 따라 서로 연결되는 장치의 수도 증가하고 있다. 이러한 장치는 환경을 식별하고 제어하는 기존에 없었던 기능을 제공한다. 장치 간에 전송되는 센서 데이터는 분석을 위해서 서버로 전송될 수 있다. 서버로 전송되는 기술의 보안은 본질적으로 안전하지 않기 때문에 큰 문제가 될 수 있다. 임베디드 시스템의 경우 이것은 새로운 보안 문제가 아니다 [1-5]. 도청과 같은 보안 문제는 사람들의 일상 활동에 불법적으로 영향을 미칠 수 있으며 합법적인 주체라도 사용자 동의 없이 데이터를 수집할 수 있다. 데이터 프라이버시를 보장하는 가장 쉬운 방법은 데이터에 암호화를 적용하는 것이다 [6-8].

IoT 개념의 개발과 다양한 분야에서의 구현은 수백억 개의 독립형 장치를 제공하고 있다. Statista 포털에 따르면 2017년에는 이미 200억 개 이상이 사용되고 있으며 2025년에는 최소 750억 개에 이를 것으로 예상된다. 장치를 모두 네트워크에 연결되어 네트워크를 통해 기능에 해당하는 데이터를 전송한다. 장치에서 생성된 데이터와 기능 모두 공격자의 대상이므로 보호해야 한다.

IoT에서는 기본 보안 기능 외에도 기밀성도 고려해야 한다. 많은 IoT 서비스 및 애플리케이션은 공격자가 공개할 수 있는 기밀 및 개인 정보를 제공한다. 보호되지 않은 기밀 데이터는 제3자에게 전송될 수 있다. 많은 개인 정보 보호 솔루션은 강력한 컴퓨터와 인터넷 사이트용으로 설계되었다. 기밀성 솔루션은 일반적으로 계산 비용이 많이 드는 암호화 기본 요소를 기반으로 한다. 이와 관련하여 주로 액세스가 제한된 장치에서 작동하는 IoT에 대한 안전하고 효과적이며 기밀성 솔루션을 개발하는 것은 미결 과제로 남아 있다.

본 연구에서 주요 목표는 다양한 장치에서 일반적인 암호화 기본 요소가 어떻게 필요한지 보여주고 IoT에서 기밀성을 유지하는 몇 가지 방법의 관점을 보여주는 것이다.

본 논문에서는 다양한 장치에서 널리 사용되는 암호화 알고리즘의 성능을 제시하고 메모리 제한에 대해 논의한다. 또한, 다양한 플랫폼에서 암호화 작업을 구현하고 측정한다. 연구에서 목적은 mbed 플랫폼의 BLE Nano 키트 및 BLE Nano 1.5 마이크로컨트롤러, MULTOS 플랫폼의 Smartcard ML3-36k-R1에서 이진 숫자 링의 완전 동형 암호화 및 블록 암호와 같은 최신 암호화 알고리즘을 성능 암호화 및 해독하는 것이다. 다음으로 리소스가 제한된 IoT 장치와 고성능 IoT 장치에서 다양한 대칭 및 비대칭 알고리즘의 실행 시간을 비교한다 [9-13].

마이크로컨트롤러는 전자 장치 모음을 제어하기 위해 일련의 기능을 수행하도록 프로그래밍할 수 있는 집적 회로(IC)이다. 프로그래밍을 통해 마이크로컨트롤러를 원하는 기능을 수행할 수 있게 만들 수 있다. 마이크로컨트롤러는 작은 크기, 간단한 아키텍처의 장점으로 인해 다양한 주변 장치와 함께 다양한 응용 분야에 쉽게 채택될 수 있다. BLE(Bluetooth Low Energy) Nano는 오픈 소스 플랫폼을 갖춘 마이크로컨트롤러의 일종이다. BLE는 개인 영역 네트워크에 대한 무선 기술 표준이다 [14-17]. BLE는 몇 달 또는 몇 년 동안 코인 셀 배터리로 실행할 수 있는 초저전력 장치이다. BLE를 사용하는 일반적인 애플리케이션은 건강 관리, 피트니스 추적기, 비콘, 스마트 홈, 보안, 엔터테인먼트, 근접 센서, 산업 및 자동차 등이다. 본 연구에서는 BLE Nano Kit와 BLE Nano 1.5의 두 가지 종류의 마이크로컨트롤러를 사용한다. BLE Nano Kit 및 BLE Nano 1.5는 시장에서 가장 작은 BLE 개발 보드이다. 코어는 초저전력 소비로 16MHz에서 실행되는 Nordic nRF51822(ARM Cortex-M0 SoC 및 BLE 기능)입니다. IoT 및 기타 흥미로운 프로젝트를 위한 프로토타입 및 데모 타겟을 빠르게 생성할 수 있다. BLE Nano Kit와 BLE Nano 1.5는 저비용, 크로스 OS 확장성, 오픈 소스 및 손쉬운 사용 기능으로 인해 다양한 개발환경에 사용될 수 있다. 결과적으로 이 플랫폼에서 다양한 다기능 응용 프로그램을 개발할 수 있다. 스마트 카드로 작업을 수행하기 위해 ML3-36k-R1 스마트 카드를 선택했다.

논문은 다음과 같이 구성된다. 2장에서는 이진 숫자 링에서 블록 암호와 완전 동형 암호화의 주요 기능과 적용 가능성을 개괄적으로 설명한다. 3장에서는 마이크로컨트롤러, 스마트 카드 및 PC 플랫폼을 설명하고, 4장에서 다양한 알고리즘의 실행 시간과 문제를 제시하고 접근 방식의 채택에 대해 논의한다. 마지막으로 5장에서는 논문의 최종 결론을 설명한다.

2. 관련 연구

이 장에서는 실험에서 사용되는 암호화 방법과 프로세스에 대해 자세히 설명한다. 그리고 제안된 두 기술인 이진수 링에서 블록 암호와 완전 동형 암호화의 주요 특징과 장점을 간략하게 소개한다.

2.1 블록 암호화

보안을 보장하고 데이터의 기밀성을 유지하기 위해 대칭 블록 암호 알고리즘을 사용하며 널리 사용되는 AES(Advanced Encryption Standard) 암호화를 선택하였다 [10-12]. 또한, AES는 데이터의 시간 확보 분석에 사용된다. 암호화에서 블록 암호는 블록이라고 하는 고정 길이 비트 그룹에서 작동하는 결정론적 알고리즘이며 대칭 키로 지정되는 불변 변환을 사용한다. 블록 암호는 많은 암호화 프로토콜 설계에서 중요한 기본 구성 요소이며 대량 데이터 암호화를 구현하는 데 널리 사용된다. 본 연구에서는 128비트 키가 있는 AES를 사용하여 데이터를 암호화하고 복호화했다. AES-128 암호화 및 복호화 데이터 흐름도는 그림 1과 같다.

AES는 2개의 주요 부분으로 구성된다.

1) 암호화 또는 복호화 과정

각 라운드에서 블록 암호는 다음 네 가지 연산을 사용한다.

a) SubBytes: 배열의 각 바이트는 AES SBox라고 하는 비선형 대체 상자를 사용하여 변환한다. 블록 암호의 S-Box는 신중하게 구성되었으며 암호는 암호화 전체에 하나의 S-Box만 사용한다.

b) ShiftRows: 배열의 마지막 세 행이 다른 수의 바이트 위치만큼 이동되도록 하는 전치 단계이다.

c) MixColumns: 더 많은 확산을 만들기 위해 어레이의 각 열을 혼합한다.

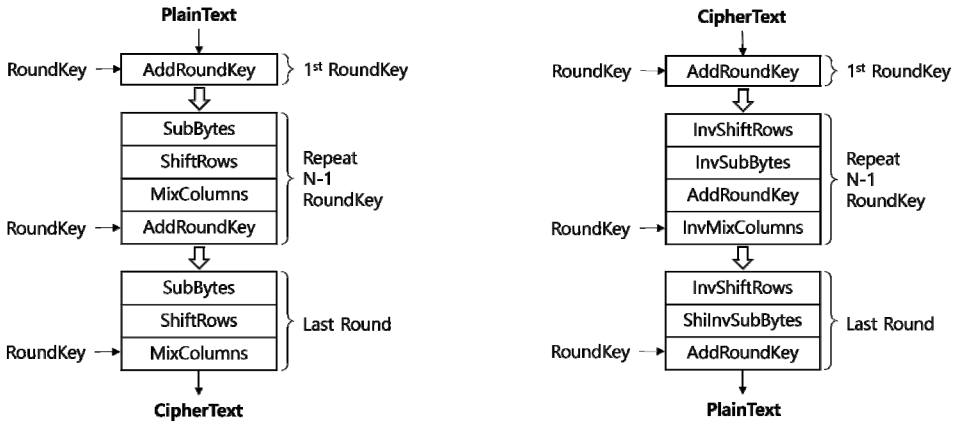
d) 키 추가: 비트별 XOR을 사용하여 배열의 각 바이트는 라운드 키라고도 하는 하위 키 자료의 바이트와 혼합한다. 하위 키는 "키 확장"으로 만들어지며 Rijndael 키 일정을 사용하여 기본 암호 키에서 파생된다.

e) 암호화 과정: Key0이 있는 AddKey로 시작한다. 그런 다음 루프로 이동하여 SubBytes, ShiftRows, MixColumns, Addkey를 순서대로 9개의 원에 대해 서로 다른 단계 키를 사용한다. 그런 다음 마지막 단계 (서클 10)으로 이동하고 MixColumns를 제외하고 루프에서 동일한 이전 기능을 반복한다. 암호 해독 프로세스: 모든 단계에서 암호화 프로세스의 역순으로 구성된다.

2) 키 교환

암호화, 복호화 과정에서 각 단계에 대해 충분한 키를 생성하기 위해 RotWord, SubBytes 및 XOR 비트 연산이 사용된다. 각 단계의 키 생성 프로세스에서 생성된 다른 키로 작동하기 때문이다. 이 암호화는 정보와 키가 계속 사용자의 영향을 받고 저장 또는 전송 중에 노출될 수 있다. AES 알고리즘은 광범위한 응용 프로그램 개발에서 사용되는 데이터 암호화 표준 알고리즘이다.

2.2 이진수 링의 완전 동형 암호화



[Fig. 1] AES encryption and decryption

본 연구에서 데이터 암호화를 위해 이진 숫자 링 기술의 완전 동형 암호화를 사용한다. 왜냐하면 데이터 판독에서 진화된 수학적 계산을 가장 잘 처리할 수 있기 때문이다. 동형 암호화는 암호화된 데이터에 대해 임의의 계산을 수행하는 암호화의 한 형태이다. 클라우드 컴퓨팅에서 민감한 데이터를 암호화된 형식으로 유지할 수 있지만 암호 텍스트에 대한 계산을 수행하려면 데이터를 악용할 수 있는 클라우드 서비스 제공업체와 키를 공유해야 한다. 따라서 클라우드 서비스 공급자와 키를 공유하는 것을 방지하기 위해 대신 동형 암호화 기술을 사용한다. 계산에는 검색, 정렬, 더하기, 암호문에 대해 수행되는 곱셈이 포함된다 [16, 17].

완전 동형 암호화(Fully homomorphic encryption, FHE)는 단일 당사자의 암호화된 데이터에 대한 안전한 계산이 가능하다. Craig Gentry는 이상적인 격자를 사용하여 부분 동형 암호화에 대한 부트스트랩 기반 동형 암호화를 구현했다. 그러나 다수의 연산 후에 에러가 증폭되어 그 크기가 일정 수준을 넘어서면 정확한 복호화가 불가능한 한계가 있다 [18-20].

1) 암호화

데이터 암호화 프로세스는 다음과 같다.

a) 비밀 매개변수인 임의의 홀수를 선택한다.

$p = 2k+1,\ m \in \{0,1\}$이라 가정한다

b) 숫자 $z \in Z_2$ 은 다음과 같이 $z = 2r + m$ 컴파일 된다. 여기서 r은 임의의 숫자이다. z는 다음과 같다. $z = m \bmod 2$.

c) 암호화 과정에서 각 m은 $c = z + pq$와 연결되며 이때 q는 임의로 선택된다. 그리고 $c = 2r + m + (2k+1)q$ 로 표현된다.

정리하면 $cmod2 = (m+q) \bmod 2$ 와 같고 따라서 공격자는 암호화 출력의 패리티만 결정할 수 있다.

2) 복호화

암호화된 번호 c와 비밀 번호 p가 있으면 그다음 데이터 암호 해독 프로세스에는 다음 작업이 필요하다.

a) 비밀 매개변수를 사용여 복호화 한다.

$$p : r = cmodp = (z + pq) \bmod p$$
$$= zmodp + (pq) \bmod p$$

이때 $r = cmodp$ 은 노이즈라 표현한다.

3) 근거

$m_1, m_2 \in Z_2$ 두 개의 비트가 있고 숫자 쌍 $z_1 = 2r_1 + m_1,\ z_2 = 2r_2 + m_2$과 연결되어 있다고 가정한다. 그리고 비밀 매개변수 $p = 2k+1$을 사용하여 암호화 데이터 $c_1 = z_1 + pq_1,\ c_2 = z_2 + pq_2$를 구할 수 있다.

이 숫자의 합은 다음과 같이 계산할 수 있다.

$$\begin{aligned} c_1 + c_2 &= z_1 + pq_1 + z_2 + pq_2 \\ &= z_1 + z_2 + p(q_1 + q_2) \\ &= 2r_1 + m_1 + 2r_2 + m_2 \\ &\quad + (2k+1)(q_1 + q_2) \end{aligned}$$

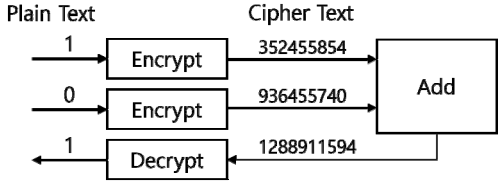
이 숫자의 합계의 복호화된 메시지는 원본 비트의 합이다.

$$\begin{aligned} m_1 m_2 : &((c_1 + c_2) \bmod p) \bmod 2 \\ &= (2(r_1 + r_2) + m_1 + m_2) \bmod 2 \\ &= m_1 + m_2 \end{aligned}$$

그러나 p를 모를 경우에는 데이터 복호화가 불가능하다.

$((c_1 + c_2) \bmod p) \bmod 2 = m_1 + m_2 + q_1 + q_2$

그림 2는 바이너리 링의 덧셈 연산 결과를 표시한다.



[Fig. 2] Result of homomorphic encryption with add operation.

3. IoT 플랫폼

이 장에서는 IoT 환경에서 사용되는 마이크로컨트롤러, 칩 카드, PC와 같은 다양한 플랫폼에서 실험을 수행하고 결과를 설명한다. 실험을 위해서 바이너리 링의 완전 동형 암호화 및 AES 암호화와 같이 가장 일반적으로 사용되는 암호화 기본 및 체계를 구현하였다. 이러한 암호화 알고리즘은 많은 IoT 보안 솔루션에서 사용된다.

리소스가 제한된 장치는 IoT 인프라에서 가장 많이 사용되는 장치로 사용된다. 제한된 리소스, 즉 BLE Nano 키트 마이크로컨트롤러, BLE Nano 1.5 마이크로컨트롤러, 33MHz Multos 카드에 구현된 암호화 기본 요소의 성능 결과를 보여준다. 또한 IoT 장치에서 비해서 고성능인 PC에서 암호화 성능도 측정하였다.

실험에서 선택한 암호화 알고리즘은 장치에 따라 두 가지 프로그래밍 언어로 구현하였다. 마이크로컨트롤러의 암호화 기능 구현은 C 프로그래밍 언어로 작성하였다. Multos 카드용 테스트 애플리케이션은 C로 작성되

었습니다. C++ 언어를 이용하여 PC 장치의 구현 및 테스트 프로그램을 제작하였다. 실험에 사용한 모든 장치는 개인 정보 보호 실험을 위한 충분한 메모리와 저장공간을 제공하고 있다. 또한, 개별 장치는 자체 응용 프로그램 및 라이브러리를 로드하기 위한 프로그래밍 가능한 플랫폼을 제공한다. 선택한 개인 정보 보호 체계를 구현하여 장치에 업로드하여 실험을 수행하였다. 실험에 사용된 BLE Nano 키트 마이크로컨트롤러, BLE Nano 1.5 마이크로컨트롤러, Multos 카드 장치, PC의 세부사항은 표 1에 정리하였다.

〈Table 1〉 Technical specification of devices

Device	Processor	Frequency	RAM	Storage
BLE Nano Kit	16bit	16MHz	16kB	256kB
BLE Nano 1.5	16bit	16MHz	32kB	256kB
SmartCard ML3-35k-R1	16bit	33MHz	1088 +960B	280 +60kB
PC	64bit Intel Celeron	P4600 2.0GGz	4GB	32GB

mbed 플랫폼은 마이크로 컨트롤러 장치의 구현 및 테스트에 사용된다. mbed는 32비트 ARM Cortex-M 마이크로컨트롤러를 기반으로 하는 인터넷 연결 장치용 플랫폼 및 운영체제이다. 이 프로젝트는 Arm과 기술 파트너가 공동으로 개발하였다. USB를 사용하여 mbed 마이크로 컨트롤러를 PC에 연결하면 USB 플래시 드라이브처럼 표시된다. 이 작은 디스크는 mbed 인터페이스로 표시되며 드라이버 없이도 mbed에서 직접 실행하려는 BLE Nano 마이크로컨트롤러의 16진수 파일을 저장할 수 있다. mbed 디스크에 .hex 파일을 저장할 때 내부 마이크로컨트롤러의 플래시 메모리에 즉시 로드되지 않는다. 재설정을 누르면 mbed 인터페이스는 디스크에서 찾을 수 있는 최신 .hex 파일을 찾는다. 새 파일이 있으면 JTAG 인터페이스를 사용하여 마이크로컨트롤러의 내부 FLASH 메모리에 파일을 로드한다. 최신 바이너리가 이미 로드되어 있으면 다시 로드하지 않는다. 그런 다음 마이크로 컨트롤러가 실행되기 시작한다.

MULTOS 플랫폼은 스마트 카드 장치의 구현 및 테스트에 사용된다. MULTOS 플랫폼의 목적은 스마트 카드를 실행하기 위한 안전하고 장치 독립적인 플랫폼을 제공하는 것이다. 이를 위해 모든 MULTOS 구현이 제공해야 하는 실행 및 메모리 모델에 대한 사양을 개발되었다. 사양의 필수 부분 외에도 주로 암호화 기능과 관련된 많은 추가 요소가 있으며 특정 하드웨어 플랫폼에서 사용 가능하다.

4. 실험 결과

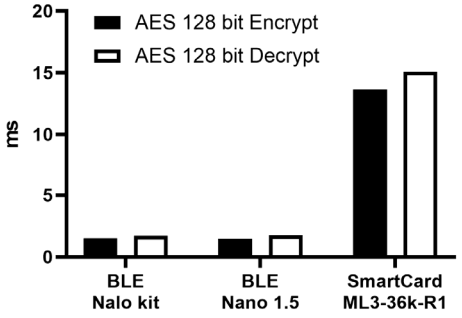
표 2는 실험에 사용된 장치 및 PC 기술에 대한 실험 결과를 표시한다. 사용된 장치의 성능 오버헤드를 측정하고 메모리/통신 오버헤드를 추정하였다. 암호화 시간 및 복호화 시간과 같은 주요 작업/단계의 시간에 중점을 두었다. 마이크로컨트롤러의 성능 암호화 작업은 사이클 수로 측정되며 그 수는 런타임에 변환하였다(클럭 주파수가 1MHz인 프로세서에서 1사이클은 1마이크로초 소요). 스마트 카드, BLE Nano Kit, BLE Nano 1.5 및 PC 트랜잭션 시간은 10회 반복하여 계산한 평균값이다.

〈Table 2〉 Encryption/Decryption test results of algorithms

Algorithm	BLE Nano Kit	BLE Nano 1.5	SmartCard ML3-35k-R1	PC
AES 128bit Encrypt	1.53ms	1.49ms	14.62ms	0.11ms
AES 128bit Decrypt	1.76.ms	1.72ms	16.05ms	0.13ms
FHE in a binary ring Encrypt	391.7ms	390.6.ms	289.2ms	21ms
FHE in a binary ring Decrypt	625.6ms	623.3ms	545.3ms	46ms

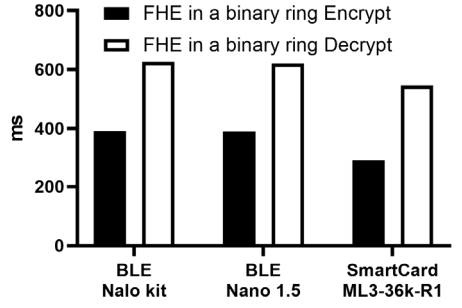
마이크로컨트롤러의 암호화를 검증하기 위해서 컴퓨터의 터미널을 이용하여 데이터와 암호화 키를 전송하였다. 그런 다음 여러 번의 테스트를 수행하였다. 그림 3은 리소스가 제한된 장치에서 AES-128 bit을 이용한 암호화/복호화 작업을 수행한 결과를 보여주고 있다.

수행 결과 암호화/복호화 작업은 장치에서 몇 밀리초만 소요되었다. 이 정보를 기반으 보면 해당 장치는 실시간 건강 모니터링 시스템으로 활용될 수 있다. 또한, 하나의 128비트 데이터 블록에 대한 AES 암호화 작업은 스마트 카드보다 마이크로컨트롤러에서 더 효율적임을 알 수 있었다. 칩 카드와 리더기 간의 카드 운영 체제 및 APDU 통신을 초기화하면 시간 지연이 발생하였다. 따라서 마이크로컨트롤러보다 프로세서 주파수가 높은 칩 카드는 AES 암호화 작업에 더 긴 실행 시간이 필요하다. 실험 결과를 종합하면 AES 기능이 효과적이며 IoT 보안 솔루션에서 구현할 수 있음을 보여준다.



[Fig. 3] Test Result of AES 128bit encryption/ decryption

그림 4는 리소스가 제한된 장치에서 바이너리 링 암호화/복호화 FHE의 수행 결과 시간을 표시하고 있다. 바이너리 링 공개 키 작업의 FHE는 마이크로컨트롤러에서 수백 밀리초가 소요된다. 개인 키를 사용한 바이너리 링 작업의 FHE는 마이크로컨트롤러에서 몇 초가 소요된다. Multos 카드는 최적화된 바이너리 링 API에서 직접 FHE를 제공한다. 이와 관련하여 이러한 스마트 카드의 바이너리 링 작업에서 FHE는 수십에서 수백 밀리초가 걸린다. 따라서 마이크로컨트롤러보다 프로세서 주파수가 높은 칩 카드는 AES 암호화 작업에 더 긴 실행 시간이 필요하다. 실험 결과를 종합하면 AES 기능이 효과적이며 IoT 보안 솔루션에서 구현할 수 있음을 보여준다.



[Fig. 4] Test result of FHE encryption/decryption

5. 결론

본 논문에서는 IoT 장치에서 사용할 수 있는 다양한 유형의 암호화 알고리즘 제시하고 실험을 통해서 검증하였다. 검증된 알고리즘은 제한된 자원을 가지는 장치를 사용하는 IoT 서비스에서 대칭 암호와 비대칭 암호를 활용할 수 있다. 가장 많이 사용되는 BLE Nano kit와 BLE Nano 1.5는 다양한 보안 기능을 수행하기 위해서 사용된다. AES 128bit와 같은 많은 대칭 암호는 BLE Nano Kit, BLE Nano와 같은 IoT 장치에 구현하기에 충분히 빠른 특징을 가지고 있다. 반면에 바이너리 링 FHE와 같은 비대칭 암호화 작업을 기반으로 하는 보안 솔루션은 실행하는 데 더 많은 시간이 필요하다. 바이너리 링의 FHE는 IoT 장치에서 수십에서 수백 밀리초가 걸리는 단점이 있다. 예를 들어 바이너리 링의 FHE는 마이크로컨트롤러에서 수백 초가 소요되며 실시간(의료) IoT 애플리케이션에는 적합하지 않다.

REFERENCES

[1] S.N.Swamy and S.R.Kota, "An Empirical Study on System Level Aspects of Internet of Things (IoT)," IEEE Access, Vol.8, pp.188082-188134, 2020.

[2] J.Hwang, L.Nkenyereye, N.Sung, J.Kim and J.Song, "IoT Service Slicing and Task Offloading for Edge Computing," IEEE Internet of Things Journal, Vol.8, No.14, pp.11526-11547, 2021.

[3] A.M.Zarca, J.B.Bernabe, A.Skarmeta and J.M.Alcaraz Calero, "Virtual IoT HoneyNets to Mitigate Cyberattacks in SDN/NFV-Enabled IoT Networks," IEEE Journal on Selected Areas in Communications, Vol.38, No.6, pp.1262-1277, 2020.

[4] H.Lee, "Intrusion Artifact Acquisition Method based on IoT Botnet Malware," Journal of The Korea Internet of Things Society, Vol.7, No.3, pp.1-8, 2021.

[5] J.K.Park, J.Kim, "Real-Time Monitoring and Control System of Server Room based on IoT," Journal of The Korea Internet of Things Society, Vol.6, No.3, pp.7-13, 2020.

[6] J.Hao, J.Liu, W.Wu, F.Tang and M.Xian, "Secure and Fine-Grained Self-Controlled Outsourced Data Deletion in Cloud-Based IoT," IEEE Internet of Things Journal, Vol.7, No.2, pp.1140-1153, 2020.

[7] S.Ramesh and M.Govindarasu, "An Efficient Framework for Privacy-Preserving Computations on Encrypted IoT Data," IEEE Internet of Things Journal, Vol.7, No.9, pp.8700-8708, 2020.

[8] S.Kim, "A IoT Security Service based on Authentication and Lightweight Cryptography Algorithm," Vol.7, No.1, pp.1-7, 2021.

[9] H.Seo, J.K.Park, "The prevent method of data loss due to differences in bit rate between heterogeneous IoT devices," Journal of the Korea Institute of Information and Communication Engineering, Vol.23, No.7, pp.829-836, 2019.

[10] K.Tsai, Y.Huang, F.Leu, I.You, Y.Huang and C.Tsai, "AES-128 Based Secure Low Power Communication for LoRaWAN IoT Environments," IEEE Access, Vol.6, pp.45325-45334, 2018.

[11] L. E. Kane, J. J. Chen, R. Thomas, V. Liu and M. Mckague, "Security and Performance in IoT: A Balancing Act," IEEE Access, Vol.8, pp.121969-121986, 2020.

[12] K.Tsai, F.Leu, I.You, S.Chang, S.Hu and H.Park, "Low-Power AES Data Encryption Architecture for a LoRaWAN," IEEE Access, Vol.7, pp.146348-146357, 2019.

[13] M.Lee, "Performance Evaluation of Smoothing Algorithm for Efficient Use of Network Resources in IoT environments," Journal of The Korea Internet of Things Society, Vol.7, No.2, pp.47-53, 2021.

[14] S.C.Cha, M.S.Chuang, K.H.Yeh, Z.J.Huang and C.Su, "A User-Friendly Privacy Framework for Users to Achieve Consents With Nearby BLE Devices," IEEE Access, Vol.6, pp.20779-20787, 2018.

[15] C.Huang, H.Liu, W.Wang and J.Li, "A Compact and Cost-Effective BLE Beacon With Multiprotocol and Dynamic Content Advertising for IoT Application," IEEE Internet of Things Journal, Vol.7, No.3, pp.2309-2320, 2020.

[16] K.E.Jeon, J.She, P.Soonsawad and P.C.Ng, "BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities," IEEE Internet of Things Journal, Vol.5, No.2, pp.811-828, 2018.

[17] B.Luo, F.Xiang, Z.Sun and Y.Yao, "BLE Neighbor Discovery Parameter Configuration for IoT Applications," IEEE Access, Vol.7, pp.54097-54105, 2019.

[18] J.Kim and A.Yun, "Secure Fully Homomorphic Authenticated Encryption," IEEE Access, Vol.9, pp.107279-107297, 2021.

[19] Y.Su, B.Yang, C.Yang and L.Tian, "FPGA-Based Hardware Accelerator for Leveled Ring-LWE Fully Homomorphic Encryption," IEEE Access, Vol.8, pp.168008-168025, 2020.

[20] Y.Ke, M.Q.Zhang, J.Liu, T.T.Su and X.Y. Yang, "Fully Homomorphic Encryption Encapsulated Difference Expansion for Reversible Data Hiding in Encrypted Domain," IEEE Transactions on Circuits and Systems for Video Technology, Vol.30, No.8, pp.2353-2365, 2020.

박 정 규(Jung Kyu Park) [종신회원]

- 2002년 2월 : 홍익대학교 컴퓨터공학과 (공학석사)
- 2013년 8월 : 홍익대학교 컴퓨터공학과 (공학박사)
- 2016년 3월 ~ 2007년 2월 : 서울여자대학교 초빙교수
- 2018년 3월 ~ 현재 : 창신대학교 컴퓨터소프트웨어공학과 교수

〈관심분야〉
사물인터넷, 로보틱스, 임베디드 시스템

김 재 호(Jaeho Kim) [정회원]

- 2010년 2월 : 서울시립대학교 컴퓨터통계학과 (공학석사)
- 2015년 2월 : 서울시립대학교 컴퓨터과학과 (공학박사)
- 2017년 10월 ~ 2019년 10월 : 버지니아공대 박사후연구원
- 2019년 11월 ~ 2020년 8월 : Huawei Germany 연구원
- 2020년 9월 ~ 현재 : 경상국립대학교 항공우주및소프트웨어공학부 & AI융합공학과 교수

〈관심분야〉
스토리지 시스템, 운영체제, 컴퓨터구조