JANT 한국항행학회논문지
Journal of Advanced Navigation Technology

응용 기술

J. Adv. Navig. Technol. 28(1): 142-148, Feb. 2024

IoT-EC 환경에서 일회용 생체정보와 ECC를 이용한 인증 관리

An Authentication Management using Biometric Information and ECC in IoT-Edge Computing Environments

한 승 진
경인여자대학교 소프트웨어융합학과

Seungjin Han
Department of Software Convergence, KyungIn Women's University, Incheon, 21041, Korea

[요 약]

IoT (Internet of Things)장치들은 열악한 환경, 저용량, 저성능 프로세서로 인해 기존의 유선망이나 무선망의 인증 방법을 적용하기가 어렵다. 특히 블록체인과 같은 방법을 IoT 환경에 적용하기에는 많은 문제점이 있다. 본 논문에서는 IoT 환경에서 생체정보 중 일회용 템플릿의 인증을 수행하는 서버 역할을 위해 엣지 컴퓨팅을 이용한다. 이와 같은 환경에서 ECC (elliptic curve cryptographic)를 기반으로 IoT-EC(edge computing) 시스템을 활용하여 가볍고 강한 인증 절차를 제안하고 이에 대한 안전성을 평가한다.

[Abstract]

It is difficult to apply authentication methods of existing wired or wireless networks to Internet of Things (IoT) devices due to their poor environment, low capacity, and low-performance processor. In particular, there are many problems in applying methods such as blockchain to the IoT environment. In this paper, edge computing is used to serve as a server that authenticates disposable templates among biometric information in an IoT environment. In this environment, we propose a lightweight and strong authentication procedure using the IoT-edge computing (IoT-EC) system based on elliptic curve cryptographic (ECC) and evaluate its safety.

Key word : Internet of Things (IoT), Elliptic curve cryptographic (ECC), Biometric, One-time template (OTT), Authentication.

http://dx.doi.org/10.12673/jant.2024.28.1.142

Received 5 January 2024; **Revised** 22 February 2024
Accepted (Publication) 24 February 2024 (29 February 2024)

***Corresponding Author; Seungjin Han**

Tel: +82-32-540-0136
E-mail: softman@kiwu.ac.kr

Ⅰ. 서 론

IoT는 많은 기기들을 연결하는 방법이자 열악한 환경에서 정보를 획득하고 이를 전달함으로써 다양한 분야에서 활용되고 있다. 그러나 IoT 장치의 특성상 저용량, 낮은 프로세서 성능으로 인해 다양한 공격에 대한 취약점이 보고되고 있고 이와 같은 공격에 대한 대처로 ECC[1]-[4], EC[4]-[6] 그리고 블록체인을 이용하여 대상을 인증하도록 하는 방법이 제안되었다[7]-[9]. 또한 최근에는 생체 정보를 이용하여 보다 간단하면서 강력한 사용자 인증 방법이 제안되었다[8],[10],[11].

본 논문에서는 IoT-EC 환경에서 [2]를 기반으로 하면서 제 3자 기관이나 등록센터를 이용하지 않고 일회용 생체정보와 ECC를 이용하여 IoT 장치에서도 동작이 가능한 경량 인증 프로토콜을 제안한다. 제안한 프로토콜은 제 3자 기관이나 중앙의 통제가 힘든 재난 지역이나 인프라가 미흡한 분쟁 지역에서 적용이 가능하다.

Ⅱ. 관련 연구

IoT 환경에서 다양한 공격을 완화하기 위한 방법으로 사용되는 방법과 이에 대한 문제점 및 보완점에 대해서 기술한다.

2-1 ECC

ECC는 IoT 네트워크에서 보안 공격을 완화하기 위한 새로운 접근 방식으로 연구되고 있다[1]-[4]. ECC는 공개 키를 교환하여 공용 네트워크를 통해 데이터를 안전하게 전송하는 데 사용되는 비대칭 암호화 기술이다. 근본적인 복잡한 문제를 해결하기 위한 알고리즘의 개선으로 인해 RSA(Rivest Shamir Adleman) 피연산자의 길이는 128비트의 보안 수준에 대해 3072비트로 증가하지만 ECC는 동일한 보안 수준에 대해 256비트로 달성할 수 있다. 표 1은 ECC와 RSA를 비교한 표이다. ECC는 데이터 전송 기밀성, 데이터 인증 및 신뢰성, 부인 방지 측면에서 강력한 정보통신 솔루션을 제공한다 [12], [13].

타원곡선 암호 시스템에서는 다음과 같은 3가지 수학적 어려움을 이용하는 암호 기술이다[14]. 여기서 P는 타원곡선에서의 포인트 값이다.

1) ECDLP (elliptic curve discrete logarithm problem)는 $z=xP$에서 z와 P를 알더라도 x를 알아낼 수 없는 특징

2) ECCDHP (elliptic curve computational Diffie-Hellman problem)는 xP, yP 값을 알아도 xyP 값을 알아낼 수 없는 특징

3) ECDDHP (elliptic curve decisional Diffie-Hellman problem)는 $zP = xyP$ 일 때, zP로부터 xP, yP를 알 수 없는 특징이 있다.

본 논문에서는 xP에서 x와 P 간 타원곡선 연산을 ・로 표시한다. 예를 들어, ECDDHP는 타원곡선연산 ・ 상에서 $z \cdot P = x \cdot y \cdot P$에서 공격자가 $z \cdot P$ 값을 알더라도 $x \cdot P$ 값과 $y \cdot P$ 값을 알아낼 수 없다는 것을 뜻한다.

표 1. ECC와 RSA 비교
Table 1. Comparison between ECC and RSA.

Method	ECC	RSA
Infra	WPKI(Wireless Public Key Infrastructure)	PKI(Wire)
Speed	Superior to RSA	Slower than ECC
Size of Key	Smaller key than RSA	Larger height compared to ECC (160 ECC = 1024 RSA)
Apply	Compact mobile environment	Environment with some infrastructure implemented

참고문헌 [1]은 IoT-Edge와 클라우드 서버 환경에서, 참고문헌 [3]은 스마트 그리드에서 계층적으로 IoT 장치와 안전하게 통신할 수 있도록 ECC 적용을 제안하였다.

참고문헌 [2]에서 제안하는 내용 중 ECCbAP[2]의 절차에서 3군데의 오류가 있다. 본 논문에서는 참고문헌 [2]의 내용을 기반으로 하되 오류 부분을 수정하고 일회용 생체 정보를 추가로 적용하여 새로운 인증 방법을 제안한다.

참고문헌 [4]에서는 CPA (chosen plaintext attack) 및 CCA (chosen ciphertext attack)와 같은 여러 암호화 공격에 저항하기 위해 평문의 타원 곡선에 대한 매핑 단계를 향상하여 인증된 암호화 (AE; authenticated encryption)를 제공하는 테스트되고 신뢰할 수 있는 체계를 제안하였다.

2-2 EC 기반의 IoT에서의 블록체인 동작

IoT 장치들은 트랜잭션 생산자가 되며, IoT 게이트웨이와 엣지 컴퓨팅 장치들은 트랜잭션의 유효성 검증 및 블록을 생성하는 피어로 동작한다. 컴퓨팅 파워가 충분하지 못한 IoT 장치들은 블록체인의 노드로 동작하기 힘들다. 따라서 충분한 컴퓨팅 자원을 가진 엣지 컴퓨팅 장치는 블록 채굴을 위한 합의 알고리즘을 수행하고, 채굴한 블록을 블록체인 네트워크에 배포하여 최종적으로 블록체인을 생성하고 있다[7]-[9].

IoT 장치들이 접속하여 이를 관리하는 네트워크에 배치되는 엣지 컴퓨팅 장치로부터 컴퓨팅 자원을 임대하여 채굴을 할 수 있도록 하고 있다.

2-3 IoT 환경에서 생체 정보를 이용한 인증 관리

참고문헌 [8]에서는 Shamir's secrete sharing (SSS) scheme과 생체 정보를 이용하여 개인 키의 백업 및 복구 방안을 제안하였고, 사용자의 생체 정보를 사용하여 OTT (one time template)를 생성하고 이를 이용하여 블록체인에 백업된 개인 키 복구를 제안하였다. 또한 공격자가 한 개의 저장소에서 원본 데이터를 확보해도 임계치 이상의 데이터를 확보해야 복원이 가능하기 때문에 임계치 미만의 데이터로는 개인 키 복원은 원천적으로 불

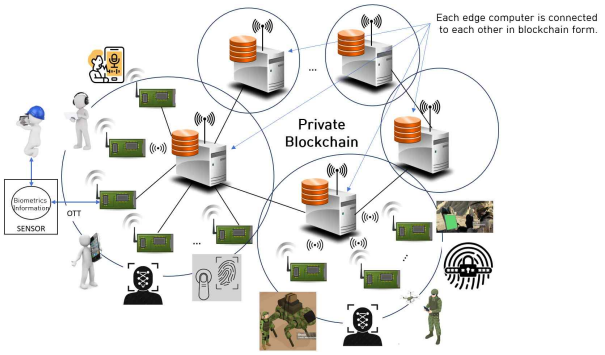

그림 1. 제안하는 시스템의 개략도
Fig. 1. Overview of proposed system.

가능하다는 것을 보였다.

참고문헌 [10]에서는 웨어러블 IoT 기기의 보안 취약점을 인식하고, 신체를 통해 생체 신호 값을 가지고 PUF (physical unclonable function))의 동작을 구현하여 인증키를 관리하는 보안 인증 방식에 대해 제안하고 동적 인증키 관리 구조를 설명하였다.

Ⅲ. 일회용 생체정보와 ECC를 이용한 인증 관리

그림 1은 본 논문에서 제안하는 방법으로 사용자는 최초 한 번 EC에 자신의 생체 정보와 사용자 ID를 등록한다.

이후 사용자는 적법하게 등록 과정이 완료된 IoT 장치를 통해 자신의 생체 정보를 인식시킨다. IoT 장치는 획득한 생체 정보를 통해 OTT를 생성한 후 즉시 생체 정보 원본을 삭제한다. OTT는 ECC를 이용하여 공격자가 알 수 없게 암호화되어 EC로 전달되고 EC는 프라이빗 블록체인으로 구성된 EC의 각 저장소에서 사용자의 OTT를 검색한 후 비교하여 사용자를 인증한다. 사용자가 적법하다면 OTT 토큰을 갱신하여 사용자가 인증을 요청한 IoT 장치를 통해 갱신한다. 여기서 OTT 생성 및 갱신 방법은 [11]을 따른다.

사용자 인증으로 사용하는 생체 정보는 원본이 노출이 된다면 대체가 불가능하기 때문에 본 논문에서는 [8]에서 기술한 취소가능한(cancellable) 생체 정보를 위해 단방향 (one-way)이면서, 되돌릴 수 없는(irreversible) 함수를 사용하여 특징을 변환하기 위해 카르테시안 변환(Cartesian transformatuon)[15]을 적용한다고 가정한다.

본 논문에서 제안하는 내용을 간단하고 명확하게 하기 위해 표 2와 같이 기호를 정의한다.

3-1 등록 단계

알고리즘 1은 ED_i가 자신이 속한 $EC(ED_i)$에 등록하는 과정이다. 난수를 발생시키고 $PEID_i, PUID_i$를 생성한 후 $EC(ED_i)$로 전송한다.

표 2. 기호
Table 2. Notations

기호	설명
$\{X \rightarrow Y: M\}$	X sends the message M to Y
$RANDINT()$	Generate integer random number
$H(Y)$	Hashes Y using a strong one-way hash function(H).
$A \oplus B$	exclusive-or A and B
$\bullet$	Operators on Elliptic Curves
$A \overset{?}{\equiv} B$	compare A with B
P	Point values of elliptic curve
ED_i	i-th embedded device
ST_i	Session timer between i-th embedded device and edge computer, which manages i-th ED
UID_i	Unique ID of user registering on i-th ED
EID_i	Unique ID of i-th ED
$EC(ED_i)$	EC to which the i-th ED belongs
ECR_i	Random number generated by EC for I-th ED
$EC(ED_i)_{EK}$	Unique Identifier
B_i	Biometric Information provided by the user to the i-th ED
OTT_i	OTT created by the user in the i-th ED
R_i	Random number to update the OTT token of the authenticated user on the i-th ED

알고리즘 2는 ED_i로부터 $PEID_i, PUID_i$를 수신한 $EC(ED_i)$는 난수를 발생시킨 후 $EC(ED_i)$만의 식별자 $EC(ED_i)_{EK}$(본 논문에서는 특정 블록체인의 키)와 타원곡선의 포인트 값 P를 이용하여 CK_i를 생성한다. EID_i와 세션 타이머 ST_i를 이용하여 CK_i'를 생성한 후 ED_i에 전송한다. 이후 $EC(ED_i)$는 $PEID_i, PUID_i, ST_i, CK_i$를 저장한다. 여기서, ED_i와 $EC(ED_i)$ 간에 세션 타이머(ST_i)을 두어 주기적으로 ED_i와 $EC(ED_i)$ 간에 상호 인증을 하도록 한다.

알고리즘 1에서 ED_i는 $EC(ED_i)$로부터 CK_i'를 수신한 후 $PEID_i, PUID_i$와 함께 저장한다.

표 3. $EC(ED_i)$에 ED_i 등록 - ED_i
Table 3. A ED_i registers with $EC(ED_i)$ - ED_i

Algorithm 1: ED_i registration – ED_i
Input: UID_i
Output: $PEID_i, PUID_i$
1: X_1 = RANDINT()
2: X_2 = RANDINT() /* X_1 and X_2 must be different */
3: $PEID_i = (X_1 \oplus EID_i)$
4: $PUID_i = (X_2 \oplus UID_i)$
5: $\{ED_i \rightarrow EC(ED_i): PEID_i, PUID_i\}$
6: STORE($PEID_i, PUID_i$)
7: RECEIVE(CK_i') FROM $EC(ED_i)$
8: STORE(CK_i')

표 4. $EC(ED_i)$에 ED_i 등록 - $EC(ED_i)$
Table 4. A ED_i registers with $EC(ED_i)$ - $EC(ED_i)$

Algorithm 2: ED_i registration – $EC(ED_i)$
Input: $PEID_i, PUID_i$,
Output: CK_i'
1: **RECEIVE**($PEID_i, PUID_i$) **FROM** ED_i
2: Y_1 = **RANDINT()**
3: $CK_i = (Y_1 \oplus EC(ED_i)_{EK} \oplus ST_i \oplus PEID_i \oplus PUID_i)$
4: $CK_i' = CK_i \cdot P$
5: $\{EC(ED_i) \rightarrow ED_i : CK_i'\}$
6: **STORE**($PEID_i, PUID_i, ST_i, CK_i$)

3-2 OTT를 이용한 사용자 인증 단계

OTT_i를 이용하여 사용자를 인증하는 과정을 ED_i와 $EC(ED_i)$ 측에서 각각 알고리즘 3과 4를 통해 설명한다.

여기서, OTT 생성, 사용자 등록 및 초기화, OTT를 이용한 사용자 인증의 서버 측 과정, 클라이언트의 OTT 탬플릿 갱신, OTT token 갱신은 [11]을 따른다.

표 5. OTT_i를 이용한 인증 - ED_i
Table 5. Authentication using OTT_i - ED_i

Algorithm 3: Authentication step at ED_i
Input: , B_i, CK_i', Q_3, Q_5
Output: $PEID_i, PUID_i PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$
1: X_3 = **RANDINT()** /* X_1, X_2 and X_3 must be different each other */
2: $Q_1 = X_3 \cdot P$
3: $Q_2 = X_3 \cdot CK_i'$
4: $PPEID_i = PEID_i \cdot P$
5: $PPEID_i = PUID_i \cdot P$
6: **GET** B_i **FROM** sensor
7: $OTT_i = OTTGen(B_i)$
8: **DELETE**(B_i)
9: $POTT_i = OTT_i \cdot P$
10: $\{ED_i \rightarrow EC(ED_i) : PPEID_i, PPUID_i, POTT_i, Q_1, Q_2\}$
11:
12: **RECEIVE**(Q_3, Q_5) **FROM** $EC(ED_i)$
13: **GET** $Q_5' = ((CK_i' \cdot PEID_i) + (CK_i' \cdot PUID_i) + (X_3 \cdot Q_3))$ **FROM** Q_5 **EXCEPT** Q_4
14: **GET** R_i **FROM** Q_4
15: **IF** $H(K_n \oplus R_n') \stackrel{?}{=} H(K_n \oplus R_n)$ **THEN**
16: $K_{n+1} = H(K_n \oplus R_i)$
17: **END IF**

알고리즘 3에서 사용자는 ED_i의 센서를 통해 생체 정보(B_i를 제공하고 ED_i는 이를 이용하여 OTT_i를 생성한 후 즉시 획득한 B_i를 삭제한다. $EC(ED_i)$로 $PPEID_i, PPUID_i$, Q_1, Q_2를 생성하여 전송한다.

표 6. OTT_i를 이용한 인증 - $EC(ED_i)$
Table 6. Authentication using OTT_i - $EC(ED_i)$

Algorithm 4: Authentication step at $EC(ED_i)$
Input: $PEID_i, PUID_i, PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$
Output: CK_i', Q_3, Q_5
1: **RECEIVE**($PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$) **FROM** ED_i
2: **FIND**($PPEID_i, PPUID_i, CK_i$)
3: **VERIFY**(ST_i)
4: $Q_2' = CK_i \cdot Q_1$
5: **IF** $Q_2' \stackrel{?}{=} Q_2$ **THEN**
6: **FIND**($PUID_i(OTT_i)$)
7: **IF** $POTT_i \stackrel{?}{=} OTT_i \cdot G$ **THEN**
8: R_i = **RANDINT()**
9: X_4 = **RANDINT()**
10: $Q_3 = X_4 \cdot G$
11: $Q_4 = R_i \cdot G$
1 2 : $Q_5 = ((CK_i \cdot PEID_i) + (CK_i \cdot PUID_i) + (X_4 \cdot Q_1)) \oplus Q_4$
13: $\{EC(ED_i) \rightarrow ED_i : Q_3, Q_5\}$
14: **END IF**
15: **END IF**

ED_i로부터 $PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$를 수신한 $EC(ED_i)$는 알고리즘 4에서 각각의 조건에 맞는 $PPEID_i, PPUID_i, CK_i$를 검색하고, 세션 타이머가 만료가 되었는지 검증한다. $Q_2' \stackrel{?}{=} Q_2$를 만족하면 UID_i의 OTT_i를 검색한다. $POTT_i \stackrel{?}{=} OTT_i \cdot G$를 만족하면 ED_i로 OTT 토큰을 갱신하기 위한 난수(Q_3)와 Q_5를 전송한다.

Q_3와 Q_5를 수신한 ED_i는 Q_4를 제외한 Q_5를 계산하여 적법한 $EC(ED_i)$가 전송한지를 검증한다. Q_4로부터 R_i를 계산한 후 이를 이용하여 OTT 토큰을 갱신한다.

Ⅳ. 보안성 평가

본 논문에서 제안하는 방법의 안전성 검증은 그림 2의 각 단계에서 사용자가 사용하는 ED_i와 $EC(ED_i)$에서 주고 받는 메시지의 보안성을 [2]와 비교하여 재생(replay) 공격, 가장(impersonation) 공격, 추적 가능성(traceability) 공격, 메시지 무결성(message integrity) 공격, 위·변조(tampering) 공격 형태로 평가한다.

그림 2에서 ①과 ② 단계에 대해서 보안성 평가를 하고 ③ 단계는 사용자에게 OTT 토큰 갱신을 위한 난수를 전달하는 단계이므로 보안성 평가에서는 제외한다.

4-1 재생 공격(Replay attack)

재생 공격을 위해 공격자는 성공적인 세션에서 ED_i와 $EC(ED_i)$간에 전달된 메시지를 저장하고 이를 이용하여 ED_i 또는 $EC(ED_i)$와 직접 접촉을 시도한다.

ED_i와 $EC(ED_i)$는 각 세션에 무작위로 난수를 생성하고 도청된 세션은 이후 ST_i로 인해 이후 세션에서 쓸모가 없으므로 현실적으로 불가능하다. 그러나 [2]에서는 경우에 따라 $V_i = PID_i \times P_3$를 통해 V_i 재생이 가능하다. 이는 이전의 세션을 통해서 얻은 V_i이지만 $V_i = V_i'$인 경우가 발생할 수 있다. 이런 경우라면 재생 공격이 가능하다.

또한 본 논문에서는 인증 수단인 생체 정보는 OTT를 사용하고 있기 때문에 재사용이 불가능하다.

4-2 가장 공격(Impersonation attack)

ED_i 가장 공격을 하기 위해서는 공격자는 $EC(ED_i)$와의 세션에서 유효한 $PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$를 보내야 한다. 따라서 ED_i가 성공적으로 가장(impersonation)하기 위해서는 공격자가 유효한 $PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$를 계산해야 하는데 이는 난수 X_3, P, OTT_i를 모르면 실현 불가능하며 $PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$를 결정하기 위해 공격자는 타원 곡선을 통해 xP에서 x를 추출하는 hard problem을 해결해야 하는데 이는 실현 불가능하다. 따라서 adversary는 Di를 가장할 수 없다.

$EC(ED_i)$를 가장하려면 ED_i가 $PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$를 전송할 때 공격자는 유효한 Q_3, Q_5를 반환해야 한다. 공격자가 알고리즘 4의 12번 라인에서처럼 $CK_i, PEID_i, PUID_i, X_4, X_3, P$를 모른다면 유효한 Q_3, Q_5를 반환하는 것은 불가능하다.

4-3 추적성 공격(Traceability attack)

추적성 공격을 구현하려면 공격자는 $EC(ED_i)$ 또는 ED_i에서 전송된 메시지에 사용되는 값이 필요하다. 본 논문의 모든 단계에서 사용하는 메시지들은 X_1, Y_1과 같이 새로운 값으로 무작위화되기(randomized) 때문에 추적성 공격이 불가능하다.

4-4 메시지 무결성 공격(Message integrity attack)

ECC는 비대칭 암호 방식이기 때문에 암호화 후 공격자가 메시지를 변경한다면 암호문의 그림의 단계 ②에서 조건이 성립되지 않는다.

4-5 중간자 공격(Man-In-The-Middle attack)

가장 공격에서 설명한 것처럼 중간자 공격을 위해서는 공격자가 $PPEID_i, PPUID_i, POTT_i, Q_1, Q_2$와 $CK_i, PEID_i, PUID_i, X_4, X_3, P$를 생성해야 한다. 공격자가 올바른 각 항목을 재생산하는 것은 불가능하고, 특히 OTT를 재생산하는 것이 불가능하다.

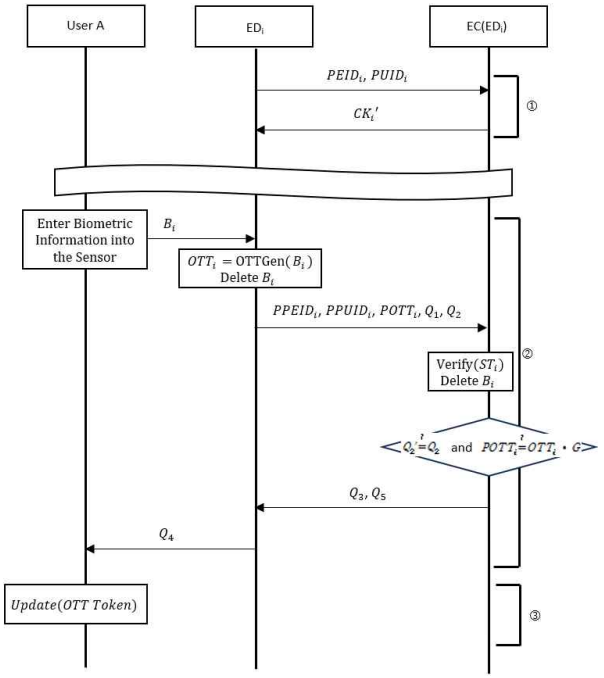

그림 2. 안전성 검증
Fig. 2. Safety verification.

4-6 위·변조 공격(Tampering attack)

사용자의 원본 생체 정보(B_i)는 알고리즘 3의 8번처럼 OTT를 생성한 후 사용자 단말기에서 바로 삭제하기 때문에 공격자는 정보를 위·변조 하더라도 인증에 필수적인 생체 정보(B_i)를 복구할 수 없다.

표 6. 제안 방식과 타 프로토콜과의 암호 기능과 보안성 비교[2]
Table 6. Encryption function and security comparison of the improved protocol to other protocols[2].

Protocol	Encryption function	P1	P2	P3	P4	P5	P6
[16]	ECC	✖	✖	✓	✖	✖	✖
[17]	Hash+ECC	✖	✖	✖	✖	✖	✖
[18]	Hash+ECC	✖	✖	✓	✓	✖	✖
[19]	Hash+ECC	✖	✖	✓	✓	✖	✖
[20]	Hash+ECC	✖	✖	✓	✓	✖	✖
[2]	ECC	✓	✓	▲	✓	▲	▲
This Paper	ECC	✓	✓	✓	✓	✓	✓

P1: Traceability attack; P2: Impersonation attack; P3: Replay attack;
P4: Message Integrity attack; P5: Man-in-the-middle attack;
P6: Tampering attack
✓ Resistant ✖: Non-resistant ▲: Partially resistant

그림 2의 각 단계를 공격하기 위해서는 공격자는 $PPEID_i$, $PPUID_i$, $POTT_i$, Q_1, Q_2와 CK_i, $PEID_i$, $PUID_i$, X_4, X_3, P를 알아야 한다. 따라서, 공격자가 전송 중인 메시지를 중간에 가로챈다 하더라도 각 단계의 메시지를 위·변조할 수 없다.

표 6은 참고문헌 [2]의 결과를 토대로 위·변조 공격을 추가했고 본 논문의 결과는 개선된 프로토콜이 허용 가능한 수준의 보안을 가지고 있으며 IoT 응용 프로그램에 대한 인증 프로토콜의 보안 요구 사항을 충족할 수 있음을 보여준다.

Ⅴ. 결론 및 추후 연구

본 논문은 ECC의 특징을 이용하여 저성능의 IoT 환경에서 장치를 안전하고 가볍게 등록하고, 사용자의 OTT를 이용하여 강력하면서 가볍게 인증할 수 있는 방법을 제안하였다.

본 논문이 비록 [2]의 알고리즘을 일부 이용하였으나 [2]의 내용 중 잘못된 내용을 바로 잡으면서 사용자의 일회용 생체 정보를 추가하여 보다 안전한 인증 메커니즘을 제안하였다.

보안성 분석을 통해 보안 특성이 우수하고, 다양한 공격에 대해 안전함을 보였다. 또한 공격자가 ED_i와 $EC(ED_i)$가 주고 받는 메시지의 내용을 안다고 해도 인증 때마다 바뀌는 사용자의 일회용 생체 정보를 알 수 없기 때문에 기존의 방법에 비해 안전하다.

추후 연구과제로는 ECC와 생체 정보를 이용하여 Pervasive Computing 환경에서 블록체인에 참여하는 노드가 다른 지역으로 이동하거나 새롭게 들어왔을 때 블록체인 기능이 가능하도록 하는 방법을 연구할 계획이다. 이는 소규모 선단이나 인프라가 열악한 지역에서 블록체인의 노드가 빈번하게 변경되는 경우에 적용이 가능할 것이다.

Acknowledgments

이 논문은 2021년도 정부(교육부)의 재원으로 한국연구재단 기본연구사업의 지원을 받아 수행된 연구임 (No.2021R1F1 A1048973)

References

[1] E. Gyamfi, J. A. Ansere, and L. Xu, "ECC based lightweight cybersecurity solution for IoT networks utilising multi-access mobile edge computing," in *Proceeding of the 38th Annual International Symposium on Computer Architecture*, Rome: Italy, pp. 10-13, June 2019.

[2] S. Rostampour, M. Safkhani, Y. Bendavid, and N. Bagheri, "ECCbAP: A secure ECC-based authentication protocol for IoT edge devices," *Pervasive and Mobile Computing, Elsevier*, Vol. 67, pp. 1-16, Sep. 2020.

[3] A. Das, M. Wazid, A. R. Yannam, J. Rodrigues, and Y. Park, "Provably secure ECC-based device access control and key agreement protocol for IoT environment," *IEEE Access*, Vol. 7, pp. 55382-55397, Apr. 2019.

[4] H. AlMajed and A. AlMogrenet, "A secure and efficient ECC-based scheme for edge computing and internet of things," *MDPI Sensors*, Vol. 20, No. 21, pp. 1-31. [Online] Available: https:// www.mdpi.com/1424-8220/20/21/6158

[5] S. Garg, K. Kaur, G. Kaddoum, P. Garigipati, and G. S. Aujlaet, "Security in IoT-driven mobile edge computing: new paradigms, challenges, and opportunities," *IEEE Network*, Vol. 35, Issue 5, pp. 293-305, Sep/Oct. 2021.

[6] J.-Y. Choi, "A study on the application of blockchain to the edge computing-based Internet of Things," *Journal of Digital Convergence*, Vol. 17, No. 12, pp. 219-228, Dec. 2019.

[7] W. Choi, S Kim, and K Han, "Blockchain-based lightweight mutual authentication protocol for IoT systems," *Journal of the Korea Society of Computer and Information*, Vol. 25, No. 1, pp. 87-92, Jan. 2020.

[8] S. Han, "A study on the private key backup and restoration using biometric information in blockchain environment," *Journal of the Korea Society of Computer and Information*, Vol. 28, No. 3, pp. 59-66, Mar. 2023.

[9] H. Park, M Kim, and J Seo, "IoT multi-phase authentication system using token based blockchain," *KIPS Transactions on Computer and Communication Systems*, Vol. 8, No. 6, pp. 139-150, Jun. 2019.

[10] G. J. Yoon et. al, "Advanced IoT platform using bio-inspired dynamic authentication key management," *in Proceeding of KICS Winter Conference 2020*, Yongpyong: Korea, pp. 832-833, Oct. 2020.

[11] TTA, Biometric authentication framework based on one-time template, Telecommunications Technology Association: Korea, TTA Standard TTAK.KO-12.0098, 2008.

[12] C. -C. Chang and H. -D. Le, "A provably secure, efficient, and flexible authentication scheme for Ad Hoc wireless sensor networks," *IEEE Transactions on Wireless Communications*, Vol. 15, No. 1, pp. 357–366, Jan, 2016.

[13] A. G. Reddy, A. K. Das, E.-J. Yoon, and K.-Y. Yoo, "A secure anonymous authentication protocol for mobile services on elliptic curve cryptography," *IEEE Access*, Vol. 4, pp. 4394–4407, 2016.

[14] K. Lauter, "The advantages of elliptic curve cryptography for wireless security," *IEEE Wireless communications*,

Vol. 11, No. 1, pp. 62–67, Feb, 2004.

[15] M. Aydar, S. C. Cetin, S. Ayvaz, and B. Aygun, “Private key encryption and recovery in blockchain,” arXiv, 1907.04156v1 [cs.CR], Jul 2019.

[16] Y. P. Liao and C.-M. Hsiao, “A secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol,” *Ad Hoc Networks,* Vol. 18, pp. 133-146, Mar. 2013.

[17] S. Kalra amd S. K. Sood, “Secure authentication scheme for IoT and cloud servers”, *Pervasive and Mobile Computing, Elsevier*, Vol. 24, pp. 210-223, Dec. 2015.

[18] C. Chang, H.-L Wu and C.-Y. Sun, Notes on ‘‘Secure authentication scheme for IoT and cloud servers,’’ *Pervasive and Mobile Computing, Elsevier*, Vol. 38, Part 1, pp. 275-278, Jul. 2017.

[19] S. Kumari, M. Karuppiah, A. Das, X. Li, F. Wu, and N. Kumaret, “A secure authentication scheme based on elliptic curve cryptography for IoT and cloud servers,” *The Journal of Supercomputing, Springer*, Vol. 74, pp. 6428-6453, Apr. 2017.

[20] K. Wang, C.-M. Chen, W. Fang, and T.-Y. Wu, “A secure authentication scheme for Internet of Things,” *Pervasive and Mobile Computing, Elsevier*, Vol. 42, pp. 15-26, Dec. 2017.

한 승 진 (Seungjin Han)

1985~1990 인하대학교 이과대학 전자계산학과 (학사), 1990~1992 인하대학교 일반대학원 전자계산공학과 (공학석사)
1999~2002 인하대학교 전자계산공학과 (공학박사). 1992~1996 대우통신 종합연구소
1996~1996 한국전산원 초고속사업단, 1996~1998 SKTelecom 디지틀사업본부
2002~2004 인하대학교 컴퓨터공학부 강의조교수, 2004~현재 경인여자대학교 소프트웨어융합학과 부교수
2007~현재: TTA PG505 표준화위원, 2012~현재 TTA PG505 간사
※관심분야 : Wired/Wireless Security, Security Protocol, Biometric, Computer Network, USN, MANET