JPNT 15(2), 137-144 (2026)
https://doi.org/10.11003/JPNT.2026.15.2.137

JPNT Journal of Positioning, Navigation, and Timing

메시지-전자서명 중첩 전송을 이용한 개선된 QZSS-CLAS 항법 메시지 인증 프로토콜

송승준[1], 노재희[2], 이문규[3†]

Improved QZSS-CLAS Navigation Message Authentication Protocol Using Message-Signature Overlapping Transmission

Seung-Jun Song[1], JaeHee Noh[2], Mun-Kyu Lee[3†]
[1]Department of Electrical and Computer Engineering, Inha University, Incheon 22212, Korea
[2]Korea Aerospace Research Institute, Daejeon 34133, Korea
[3]Department of Computer Engineering, Inha University, Incheon 22212, Korea

ABSTRACT

With recent advances in location-based services, satellite navigation systems that guarantee high precision and high integrity have become increasingly important. The Japanese Quasi-Zenith Satellite System (QZSS) provides the Centimeter-Level Augmentation Service (CLAS) via the L6 signal. However, unlike standard QZSS navigation messages such as LNAV and CNAV, CLAS augmentation messages currently lack an authentication mechanism and are consequently vulnerable to spoofing attacks. Furthermore, since CLAS is already an operational service, incorporating authentication by modifying the existing message specification is not possible. To address this issue, in 2024 Jeon et al. proposed an authentication method that uses the reserved field in the navigation messages of the L62 signal. However, this method requires significant Time To Authentication (TTA) to deal with multiple fragments of an Elliptic Curve Digital Signature Algorithm (ECDSA) signature transmitted over multiple frames. In this paper, we propose a novel authentication protocol to mitigate this issue. The proposed protocol exploits the property of the ECDSA signature pair (r, s), where the r component can be generated independently of the message content. By transmitting r concurrently with the data, the receiver can initiate signature verification earlier. Analytical results demonstrate that the proposed protocol reduces TTA by up to 25% compared to the previous method. Experimental results on both a PC and a Raspberry Pi confirm that the computational overhead of the proposed scheme is negligible.

Keywords: QZSS, CLAS, digital signature, navigation message authentication
주요어: QZSS, CLAS, 전자서명, 항법 메시지 인증

1. 서론

위치 기반 서비스(location based service)가 고도화됨에 따라, 고정밀도 및 고신뢰성을 보장하는 위성 항법 시스템의 중요성이 증대되고 있다. 위성 항법 시스템은 서비스 제공 범위에 따라 전 지구를 커버하는 Global Navigation Satellite System (GNSS)과 특정 권역을 대상으로 하는 Regional Navigation Satellite System (RNSS)으로 구분되며, 이들은 공통적으로 위성에서 다수의 수신기로 단방향 방송되는 무선 신호를 기반으로 동작한다.

그러나 이러한 개방형 단방향 통신 구조에는 공격자가 실제 신호와 유사한 위조 신호를 생성 및 송출해 수신기가 이를 획득 및 추적하도록 유도함으로써 수신기의 위치 및 시각 정보를 교란하는 스푸핑(spoofing) 위협이 존재한다. 스푸핑은 위조 신호 생성 외에도, 정상 신호를 기록 및 재전송(record and replay)하는 형태

Received Feb 13, 2026 **Revised** Feb 27, 2026 **Accepted** Mar 17, 2026

†**Corresponding Author** E-mail: mklee@inha.ac.kr

로도 구현될 수 있다. 이러한 위협에 대응하기 위해, 여러 위성 항법 시스템에서 위성 신호에 대한 인증(authentication) 체계를 도입하고 있다. 위성 신호 인증은 적용 계층에 따라 대역 확산 코드 인증(Spreading Code Authentication, SCA)과 항법 메시지 인증(Navigation Message Authentication, NMA)으로 구분할 수 있다. SCA는 확산 코드(spreading code) 자체 또는 코드에 결합된 마커(marker) 등을 이용해 신호 및 코드 수준에서 인증 정보를 제공하는 방식이며, NMA는 항법 메시지에 인증 정보를 결합하여 수신기가 메시지의 무결성과 출처를 검증하도록 하는 방식이다.

현재 NMA의 접근 방식은 크게 전자서명 기반과 Timed Efficient Stream Loss-tolerant Authentication (TESLA) (Perrig et al. 2000) 프로토콜 기반으로 구분된다. 그 중, 전자서명 기반의 NMA 방식에는 주로 Elliptic Curve Digital Signature Algorithm (ECDSA)가 사용된다. 예를 들어, Global Positioning System (GPS)의 Chips-Message Robust Authentication (Chimera) (Air Force Research Laboratory 2024a) 및 일본의 Quasi-Zenith Satellite System (QZSS)의 Quasi-Zenith Satellite Navigation Message Authentication (QZNMA) (Cabinet Office 2024)에서는 ECDSA 전자서명 기반 NMA 방식이 제시되어 있으며, Wu et al. (2019)에서는 BeiDou의 항법 메시지에 대한 ECDSA 기반 NMA 방법을 제안하였다.

반면, Galileo의 Open Service Navigation Message Authentication (OSNMA)에서는 TESLA 기반의 NMA를 제공하며 (Galileo ICD 2023), GPS Chimera의 확장 규격인 TESLA Chimera에서도 GPS 항법 메시지에 대한 TESLA 기반 NMA를 제시하였다 (Air Force Research Laboratory 2024b).

Wu et al. (2020)에서는 BeiDou 항법 메시지에 대한 TESLA 기반 인증 방법을 제안하였다. 다만, TESLA 기반 NMA에서도 인증을 위한 해시 체인(hash chain)의 루트 키(root key) 검증을 위해 ECDSA 전자서명을 사용한다.

위성 항법 시스템은 기본적인 측위 서비스 외에도, 정확도 향상을 위한 보정(augmentation) 서비스를 제공할 수 있다. 예를 들어 Galileo는 High Accuracy Service (HAS)를 통해 고정밀 측위를 위한 보정 정보를 제공하며 (Galileo ICD 2022), QZSS의 Centimeter-Level Augmentation Service (CLAS)에서는 L6 대역의 L62 신호를 통해 센티미터 수준 정밀도 향상을 위한 보정 정보를 제공한다 (Cabinet Office 2025).

현재 QZSS는 LNAV나 CNAV 등 기존 항법 신호에 대해서는 메시지 인증 기법을 적용하여 스푸핑 공격에 대한 방어 체계를 갖추고 있으나 (Cabinet Office 2024), 고정밀 보정 정보를 다루는 CLAS 신호에는 별도의 메시지 인증 기능이 적용되지 않아 보안상 취약점이 존재한다. 특히 CLAS는 이미 상용 서비스가 진행중인 시스템이므로, 기존 인프라와의 호환성을 유지하기 위해서는 메시지 규격을 변경하거나 대규모 인증 데이터를 추가하는 방식의 접근이 제한된다.

위와 같은 제약 사항을 고려하여, Jeon et al. (2024)에서는 L62 신호 메시지의 예비 영역(reserved field)을 인증에 활용하는 방안을 제시하였다. 해당 연구는 프레임 내 서명 전송에 할당할 수 있는 비트 공간이 매우 제한되어 있는 상황을 가정하여, ECDSA 전자서명을 다수의 프레임에 걸쳐 분할 전송하는 구조를 설계하였다. 그러나 이 방식은 실시간으로 변동하는 보정 데이터의 특성상 서명 값을 미리 계산할 수 없다는 한계를 가진다. 즉, 데이터 생성 및 전송 후 서명 연산과 순차적인 분할 전송이 진행되어야 하므로, 메시지 송신 시작 시점부터 전자서명 수신 완료 시점까지의 인증 지연 시간인 Time To Authentication (TTA)이 크다는 문제가 있다 (Anderson 2024, 2025).

이 논문에서는 인증에 할당된 자원이 제한적인 환경에서도 TTA를 최소화할 수 있는 기법을 제안한다. 제안 기법은 ECDSA 서명 쌍 (r, s) 중 r 값이 메시지 내용과 관계없이 생성될 수 있다는 알고리즘적 특성을 활용한다. 구체적으로는 서명의 r 값을 인증 대상 보정 데이터의 방송 시점에 미리 계산하여 전송하고, 해당 데이터 방송 이후에 서명의 s 값을 전송하도록 하여, 수신자 측이 보정 데이터 수신 완료 후 인증을 위해 전송받아야 할 데이터의 크기를 경감시킨다. 이를 통해 기존의 방식 대비 수신자 측의 TTA를 최대 25%까지 단축시킬 수 있다.

이 논문은 Song et al. (2025)의 내용을 확장한 논문이다. Song et al. (2025)에서는 송신자 측 프로토콜만을 제안하였으나, 이 논문에서는 수신자 측의 상세한 프로토콜도 제시하며, 실제 수신 단말 환경을 가정한 연산 부하 분석을 수행한다. 이 연구에서는 실험을 통해 기존 방식 (Jeon et al. 2024)과 제안 기법의 실행 시간을 측정 및 비교하였으며, 결과적으로 제안 기법이 시스템의 연산 오버헤드를 증가시키지 않음을 확인함으로써 실제 시스템 적용에 대한 타당성을 제시한다.

이 논문의 구성은 다음과 같다. 2장에서는 CLAS 보정 데이터가 전송되는 QZSS L62 신호의 메시지 구조를 설명한다. 3장에서는 제안된 방식의 송신자 측, 수신자 측 프로토콜을 정의한다. 4장에서는 제안된 방식의 TTA를 분석하고 실제 실행 시간을 측정하여 기존 방식과 비교한다. 5장에서는 제안 프로토콜이 실제로 적용될 수 있는 조건 및 표준 ECDSA 전자서명 검증에 줄 수 있는 영향에 대해 설명하고, 6장에서는 결론을 제시한다.

2. QZSS의 메시지 구조

QZSS는 L6 대역을 통해 고정밀 측위 정보를 전송하며, 해당 대역의 물리적 신호 명칭은 L62로 정의된다 (Cabinet Office 2024). L62 신호는 전송하는 데이터의 목적에 따라 L6D와 L6E의 두 데이터 채널로 구분되어 운용된다. 이 중 L6D 메시지는 CLAS에 사용되며, L6E 메시지는 Multi-GNSS Advanced Orbit and Clock Augmentation - Precise Point Positioning 서비스 등에 활용된다.

CLAS에 사용되는 L62 신호의 메시지는 1초에 1개씩 전송되는 고정 길이 구조를 가지며, 총 길이는 2,000 비트이다. Fig. 1과 같이, 메시지는 헤더 49 비트, 데이터 파트 1695 비트, Reed-Solomon 코드 256 비트로 구성된다.

CLAS의 전송 프레임은 연속된 30개의 데이터 파트로 이루어지며, 이때 5개의 연속 데이터 파트가 1개 서브프레임, 6개의 서브프레임이 1개 프레임을 구성한다. 이때 L6 메시지 하나의 전송에 1초가 소요되므로, 하나의 프레임 전송에는 30초가 소요된다.

https://doi.org/10.11003/JPNT.2026.15.2.137

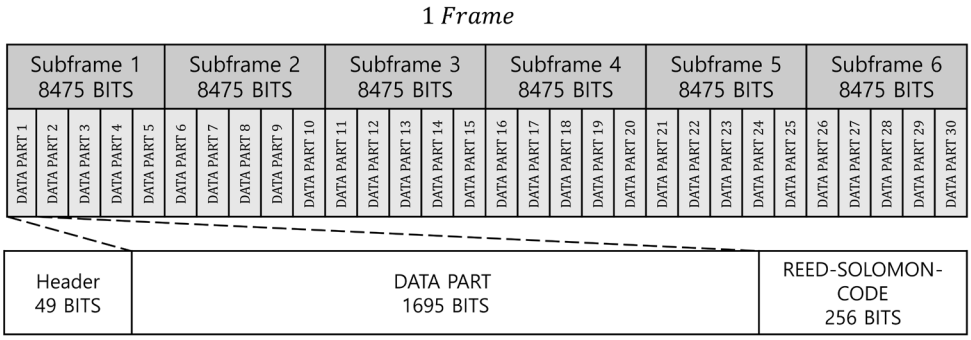

Fig. 1. Structure of QZSS L62 signal message (Cabinet Office 2025).

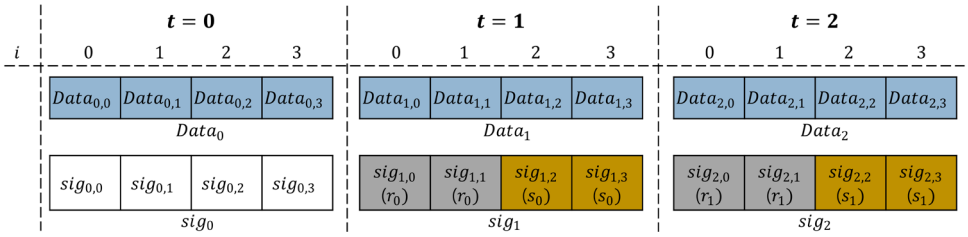

Fig. 2. Transmitter data configuration in Jeon et al. (2024) (N=4).

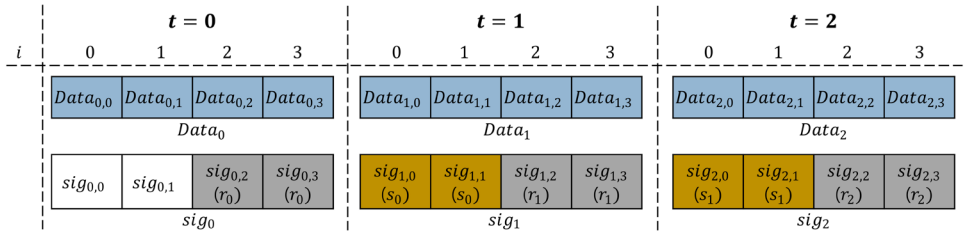

Fig. 3. Transmitter data configuration in the proposed scheme (N=4).

3. 제안 방법

이 논문에서는 Jeon et al. (2024)의 연구와 같이 QZSS L62 신호의 구조적 특징을 반영하며, 기존 인증 체계와의 호환성 및 미국 국립표준기술연구소(National Institute of Standards and Technology, NIST)의 ECDSA 안전성 권고를 고려하여 (Barker 2020) ECDSA P-256 곡선을 기반으로 인증 프로토콜을 설계한다. 구체적으로, 각 전송 프레임 내의 마지막 서브프레임에 위치한 데이터 파트에 존재하는 예비 영역을 인증 데이터 전송 채널로 활용하는 상황을 가정한다. 이때, 할당 가능한 비트 수는 전체 메시지 크기에 비해 매우 제한적이므로, 수백 비트의 크기를 갖는 표준 ECDSA 전자서명을 단일 프레임에 실어 보내는 것은 어렵다. 따라서 서명 데이터를 다수의 프레임에 걸쳐 나누어 전송하는 분할 전송의 방식을 고려한다.

송신자가 특정 t 시점에 생성하여 방송하는 L6D 메시지는 실시간 보정 정보를 포함하고 있으며, 이를 $Data_t$로 표기한다. $Data_t$가 전송되는 프레임에 포함되는 서명 데이터는 sig_t로 표기한다. 여기서 sig_t를 여러 번에 걸쳐 분할 전송하기 위해 필요한 프레임의 개수를 N으로 정의하며, 그 중 서명 r이 포함된 프레임의 수를 N_r, 서명 s가 포함된 프레임의 수를 N_S로 정의한다. 즉, $N=N_r+N_s$이며, N이 짝수일 때는 $N_r=N_s=N/2$이 된다. 이 절에서는 설명의 편의상 N이 짝수인 경우만 설명하나, N이 홀수인 경우도 유사하게 설계 가능하다. 또한, t 시점의 서명 sig_t를 N개의 조각으로 분할하여 연속된 N개 프레임에 실어 보낼 때 각 조각을 $sig_{t,i}$로 표기한다. 여기서 인덱스 i는 0부터 $N-1$까지의 값을 가지며, i번째 조각은 각 t 시점의 i번째 프레임에 포함되어 전송된다고 가정한다. 마찬가지로, $Data_t$ 자체도 N개의 조각으로 구분될 수 있으며, $Data_t$의 i번째 조각을 $Data_{t,i}$로 표기한다.

Jeon et al. (2024)에서는 Fig. 2와 같이 t 시점의 N개의 데이터 조각 $Data_{t,0},\cdots,Data_{t,N-1}$이 모두 전송된 후 서명 r_t,s_t가 생성되며, 각각 $sig_{t+1,0},\cdots,sig_{t+1,Nr-1}$ 및 $sig_{t+1,Nr},\cdots,sig_{t+1,N-1}$로 분할되어 $t+1$ 시점에 데이터 조각 $Data_{t+1,0},\cdots,Data_{t+1,N/2-1}$ 및 $Data_{t+1,N/2},\cdots,Data_{t+1,N-1}$과 함께 전송된다 ($N_r=N_s=N/2$).

한편, 제안 방법에서는 r_t가 $Data_t$와 무관하게 생성될 수 있다는 점을 이용하여 미리 r_t를 계산하여 Fig. 3과 같이 $Data_{t,N/2},\cdots,Data_{t,N-1}$과 함께 전송하고, r_t 및 $Data_t$로부터 계산되는 s_t를 $Data_{t+1,0},\cdots,Data_{t+1,N/2-1}$과 함께 전송한다. 3.1절과 3.2절에서는 제안한 방법의 송신자 및 수신자 측의 동작을 자세히 설명한다.

3.1 송신자 측 프로토콜

송신자는 실시간으로 생성되는 L6D 메시지와 함께 인증 정보를 제공하기 위해, 매 전송 주기마다 난수 생성, 데이터 패킷 구성, 그리고 ECDSA 기반 서명 생성 연산을 수행한다. 데이터 전송 이전에 필요한 암호학적 도메인 파라미터와 개인키를 입력받고 초기 상태 값을 설정하며, 이후 각 시점마다 새로운 서명 요소를 생성하여 메시지와 함께 방송한다. 송신자의 전체적인 동작 절차는 Protocol 1과 같다. 도메인 파라미터 D=(q,FR,S,a,b,P,n,h)에서, q는 타원곡선이 정의되는 유한체(finite field)의 위수(order), FR은 유한체 원소의 표현 형식, S는 타원곡선을 무작위로 생성하기 위한 시드(seed), a,b는 시드로부터 생성된 타원곡선 $y^2=x^3+ax+b$의 계수, P는 기저점(base point), n은 P에 의해 정의되는 타원곡선 군의 위수(order), h는 여인자(cofactor)이다. 이 논문에서 고려하는 NIST P-256 타원곡선 (Chen et al. 2023)의 경우, h=1이며 n은 256비트 소수(prime number)이다.

Protocol 1. Transmitter operation in the proposed scheme

Input: Domain parameters $D = (q, FR, S, a, b, P, n, h)$, private key d, message $Data_t$.

1. $s_{-1} = 0^{\lceil \log_2 n \rceil}$ // A bit string consisting of $\lceil \log_2 n \rceil$ zeros.
2. For $t \in \{0,1,2,\dots\}$ do:
   - 2.1 Select $k_t \in_R [1, n-1]$, and save k_t.
   - 2.2 Compute $k_t P = (x_1, y_1)$, and convert x_1 to an integer $\overline{x_1}$.
   - 2.3 Compute $r_t = \overline{x_1} \bmod n$, and save r_t.If $r_t = 0$ then go to step 2.1.
   - 2.4 Compute $sig_t = s_{t-1} \| r_t$, and split sig_t into $\{sig_{t,i} \mid i = 0, \dots, N-1\}$.
   - 2.5 For $i \in \{0, \dots, N-1\}$ do:
     - 2.5.1 Construct $Data_{t,i}$, and save $Data_{t,i}$.
     - 2.5.2 Transmit ($Data_{t,i}$,$sig_{t,i}$).
   - 2.6 Compute $Data_t = Data_{t,0} \| Data_{t,1} \| \cdots \| Data_{t,N-1}$.
   - 2.7 Compute $e = H(Data_t)$.
   - 2.8 Compute $s_t = k_t^{-1}(e + dr_t) \bmod n$.

먼저 프로토콜 시작 지점 t=0에서는 구조적으로 참조해야 할 이전 주기의 서명 데이터가 존재하지 않는 상황이 발생한다. 이를 위해 단계 1에서 t=−1 시점의 서명 값 s_{-1}을 $\lceil \log_2 n \rceil$ 비트 길이의 0으로 채워진 비트열로 초기화한다.

이후 매 시점 t=0,1,2,… 마다 송신자는 반복적인 서명 생성과 서명 및 데이터 전송 사이클을 수행한다. 우선 메시지 내용과 독립적으로 서명 요소 r_t를 생성하기 위해 단계 2.1에서 [1,n−1] 범위의 난수 k_t를 선택하고, 단계 2.2, 2.3에서 이에 대응하는 r_t를 계산한다. 여기서 r_t가 0으로 계산될 경우 단계 2.1로 돌아가며, 표준 ECDSA 서명 생성 절차 (Hankerson et al. 2004)와 같이 k_t부터 다시 선택하여 서명 생성을 진행한다. 생성된 k_t와 r_t는 이후 s_t 계산에 사용하기 위해 메모리에 저장한다. 단계 2.1 ~ 2.3의 과정은 t 시점의 $Data_t$가 아직 생성되지 않은 상태에서도 수행될 수 있으므로, 송신자 측에서 미리 연산해 두는 것이 가능하다.

단계 2.4에서는 위와 같이 구성된 현재 t 시점의 r_t와, 직전 t−1 주기에서 계산되어 메모리에 저장되어 있던 s_{t-1}을 연접(concatenate)하여 현재 시점의 sig_t를 구성하며, 단계 2.5에서 송신자는 실시간으로 생성된 $Data_{t,i}$와, 인증 정보 sig_t를 N개의 조각으로 분할한 $sig_{t,i}$를 프레임 단위로 방송한다.

t 시점의 N개 프레임을 모두 전송한 이후, 단계 2.6에서 송신자는 분할 전송되었던 $Data_{t,i}$ 조각들을 연접하여 $Data_t$로 구성한다. 이후 단계 2.7, 2.8에서 $Data_t$, 개인키 d, 그리고 메모리에 저장된 k_t 및 r_t를 사용하여 서명 s_t를 생성하며, 생성된 s_t는 다음 t+1 시점에 전송된다. 여기서 H는 암호학적 해시(cryptographic hash) 함수이다.

3.2 수신자 측 프로토콜

수신자는 L6D 신호를 지속적으로 청취하며, 실시간으로 수신되는 메시지와 분할된 서명 데이터를 재 조합하여 서명 검증을 수행한다. 제안하는 프로토콜에서 송신자가 서명 요소를 sig_t=$s_{t-1} \| r_t$ 형태로 구성하여 전송하므로, 수신자는 각 t 시점의 전송 주기에서 직전 주기의 메시지 $Data_{t-1}$에 대한 서명 s_{t-1} 부분을 먼저 획득하게 된다. 이를 통해 현재 주기 $Data_t$의 수신이 완료되기 이전에 직전 메시지 $Data_{t-1}$에 대한 서명 검증을 조기에 완료할 수 있다. 수신자의 구체적인 동작 절차는 Protocol 2와 같다.

Protocol 2. Receiver operation in the proposed scheme

Input: Domain parameters $D = (q, FR, S, a, b, P, n, h)$, public key Q, message $Data_t$.

1. $Data_{temp}$ = NULL, sig = NULL
2. For $t \in \{0,1,2,\dots\}$ do:
   - 2.1 For $i \in \{0, \dots, N_S - 1\}$ do:
     - 2.1.1 Receive $Data_{t,i}$, $sig_{t,i}$.
     - 2.1.2 Compute $Data_{temp} = Data_{temp} \| Data_{t,i}$.
     - 2.1.3 Compute $sig = sig \| sig_{t,i}$.
   - 2.2 If $t > 0$ then Compute $s_{t-1} = \mathrm{MSB}_{\lceil \log_2 n \rceil}(sig)$
   - 2.3 If $t > 0$ then:
     - 2.3.1 If r_{t-1} or s_{t-1} not in $[1, n-1]$ then: return “Reject the signature”.
     - 2.3.2 Compute $e = H(Data_{t-1})$.
     - 2.3.3 Compute $w = s_{t-1}^{-1} \bmod n$.
     - 2.3.4 Compute $u_1 = ew \bmod n, u_2 = r_{t-1} w \bmod n$.
     - 2.3.5 Compute $X = u_1 P + u_2 Q$.
     - 2.3.6 If $X = \infty$ then: return “Reject the signature”.
     - 2.3.7 Convert the x-coordinate x_1 of X to an integer $\overline{x_1}$; compute $v = \overline{x_1} \bmod n$.
     - 2.3.8 If $v = r_{t-1}$ then: return “Accept the signature”; else return “Reject the signature”.
   - 2.4 Save $Data_t = Data_{temp}$.
   - 2.5 For $i \in \{N_S, \dots, N-1\}$ do:
     - 2.5.1 Receive $Data_{t,i}$, $sig_{t,i}$.
     - 2.5.2 Compute $Data_t = Data_t \| Data_{t,i}$.
     - 2.5.3 Compute $sig = sig \| sig_{t,i}$.
   - 2.6 $r_t = \mathrm{LSB}_{\lceil \log_2 n \rceil}(sig)$
   - 2.7 Save $Data_{temp}$ = NULL, sig = NULL.

수신자는 타원곡선 파라미터 D 및 서명 검증에 필요한 공개키 Q를 입력받고, 단계 1에서 수신된 데이터 조각과 서명 조각을 임시로 저장하기 위한 버퍼 $Data_{temp}$, sig를 $NULL$로 초기화한다. 이후 매 시점 t=0,1,2,… 마다 수신자는 반복적인 서명 및 데이터 수신과 서명 검증을 수행한다. 이때 각 t 시점에서 N개의 프레임을

수신하는 과정은 단계 2.1의 서명 s_{t-1}이 포함된 인덱스 i=0,⋯,N_S−1 프레임을 수신하는 부분과, 단계 2.5의 서명 r_t가 포함된 인덱스 i=N_S,⋯,N−1 프레임을 수신하는 부분으로 구분되어 처리된다.

먼저 단계 2.1에서, 수신자는 t 시점의 0번째부터 N_S−1번째 프레임까지 전송되는 $Data_{t,i}$와 $sig_{t,i}$를 수신하여 각각 $Data_{temp}$, sig 버퍼에 연접한다. 단계 2.2에서는 수신된 서명 버퍼 sig의 상위 $\lceil \log_2 n \rceil$개 비트들을 직전 t−1 시점 메시지의 전자서명 부분인 s_{t-1}에 저장한다. 만약 현재 수신 시점이 t=0이 아니라면, 수신자는 단계 2.3과 같이 이미 메모리에 저장하고 있는 $Data_{t-1}$과 서명 r_{t-1}, 그리고 s_{t-1}을 이용하여 서명 검증을 수행할 수 있다. 만약 서명 검증 과정에서 계산된 점 X가 타원곡선의 무한원점(point at infinity)이거나 서명이 유효하지 않다고 판단되는 경우, 수신자는 단계 2.3.1, 2.3.6, 2.3.8과 같이 "Reject the signature"를 반환한다.

서명 검증 완료된 후, 단계 2.4에서 수신자는 현재까지 $Data_{temp}$에 저장된 데이터들을 현재 시점의 메시지를 위한 버퍼 $Data_t$로 복사하여 저장한다. 단계 2.5에서, 수신자는 단계 2.1에서의 프레임 수신에 이어서 나머지 N_S번째부터 N−1번째 프레임까지 계속해서 수신한다. 이 과정에서 수신된 $Data_{t,i}$ 조각들은 앞서 저장된 $Data_t$버퍼의 뒤에 순차적으로 연접되어 온전한 메시지를 구성하게 되며, $sig_{t,i}$ 조각들은 기존 sig 버퍼에 계속 연접된다.

i=N−1까지 모든 프레임의 수신이 완료되면, 단계 2.6에서 수신자는 sig 버퍼의 하위 $\lceil \log_2 n \rceil$개 비트들을 서명 r_t에 저장한다. 해당 r_t는 다음 t+1 시점의 서명 검증에 사용하기 위해 수신자의 메모리에 저장해둔다. 이후 단계 2.7에서는 다음 t+1 시점에서의 프로토콜 진행을 위해 임시 버퍼 $Data_{temp}$와 sig를 초기화한다.

4. 성능 분석

이번 장에서는 제안하는 프로토콜에 대하여 TTA를 분석하고, PC 및 Raspberry pi 환경에서의 프로토콜 주요 동작에 대한 속도 측정으로 기존 방식과의 성능을 비교한다.

4.1 TTA 성능 분석

Jeon et al. (2024)의 인증 방식에서는 수신자가 N개의 프레임에 걸쳐 $Data_t$의 조각들을 모두 수신하며, 그것에 대한 서명 sig_t의 조각들은 그 다음 N개의 프레임에 걸쳐 수신하므로, $Data_t$ 및 $Data_t$에 대한 서명 정보 sig_{t+1}을 모두 수신하기 위해 총 2N개의 프레임을 수신해야 한다.

제안하는 기법은 Protocol 1과 Protocol 2에서 확인되는 바와 같이, 서명 대상 메시지 $Data_t$와 서명 r_t 부분을 동시에 전송함으로써 TTA를 구조적으로 단축시켰다. 송신자는 $Data_t$의 각 분할 조각 $Data_{t,i}$를 전송할 때 r_t의 분할 조각을 프레임에 함께 실어 보내며, 메시지에 종속적인 서명 요소 s_t는 $Data_t$ 전송이 완료된 이후인 t+1 시점의 프레임들에 할당하여 전송한다. 이에 따라 수신자는 $Data_t$의 수신이 완료되는 시점에 이미 검증에 필요한 r_t를 확보하게 되며, t+1 시점에서 서명 s_t가 포함된 N_S개의 프레임만 추가로 수신하면 즉시 서명 검증을 수행할 수 있다.

Table 1은 제안 방식에서의 TTA를 Jeon et al. (2024)의 TTA와 비교하고 있다. Table 1에서 X는 프레임에 할당된 예비 영역의 비트 수, L은 보안 강도, X_{SP}는 서명 조각 $sig_{t,i}$의 비트 수를 나타낸다. 보안 강도가 L일 경우, ECDSA 전자서명은 4L 비트로 구성된다. 서명을 N개로 분할하여 전송하므로, 수신자 측에서는 특정 시점에 수신한 서명 조각이 전체 서명 중 몇번째 분할인지 식별이 필요하며, 이를 위해 0부터 N−1까지의 인덱스를 부여하여야 한다. 이 인덱스 역시 전송 프레임의 비트 영역 일부를 차지하며, X_{SN}은 이 인덱스가 차지하는 비트 수를 나타낸다. 즉, X_{SN}=[$\log_2 N$]이다. 각 서명 조각은 실제 서명 내용의 일부를 포함하는 X_{SP} 비트와 이 서명 조각의 인덱스를 나타내는 X_{SN} 비트를 모두 포함하여야 한다. 따라서, 총 전송 시간을 최소화하기 위해, 인증을 위해 각 프레임에 할당되어야 할 총 비트 개수 X_{SN}+X_{SP}의 크기는 주어진 X 범위 내에서 N이 최소가 되도록 설정하였다. 한편, sig_t의 N개 분할 중 서명 s_{t-1}이 포함된 분할의 개수를 나타내는 N_S는 Eq. (1)과 같이 계산할 수 있다.

Table 1. Comparison of TTA between Jeon et al. (2024) and proposed scheme (unit: s).

X	L	N	X_{SN}	X_{SP}	N_S	TTA (Jeon et al. 2024)	TTA (proposed)
50	128	12	4	43	6	720	540
100	128	6	3	86	3	360	270
200	128	3	2	171	2	180	150
50	192	18	5	43	9	1080	810
100	192	8	3	96	4	480	360
200	192	4	2	192	2	240	180

$$N_S = \left\lceil \frac{2L}{X_{SP}} \right\rceil \tag{1}$$

QZSS L62 신호의 프레임 하나가 전송되는 데 30초의 시간이 소요되므로, Jeon et al. (2024)과 제안 방식에서의 TTA는 Eqs. (2–3)과 같이 계산할 수 있다.

$$\text{TTA (Jeon et al. 2024)} = 30 \times 2N \tag{2}$$

$$\text{TTA (proposed)} = 30 \times (N + N_S) \tag{3}$$

예를 들어, Table 1의 첫번째 행과 같은 세팅에서는 프레임에 할당된 예비 영역의 비트 수 X가 50 비트일 때, ECDSA 서명 512 비트를 X_{SP}=43 비트씩 N=12 프레임에 걸쳐 분할 전송하게 된다. 이때 제안 방식에서는 첫 N_S=6개 프레임에 서명 s_{t-1} 부분이 포함되며, TTA는 30×(12+6)=540초로 계산된다. 반면에, Jeon et al. (2024)의 방식에 의하면, TTA는 30×(2×12)=720초로 계산된다. 다양한 파라미터를 기반으로 한 Table 1의 분석 결과에 의하면, 제안 방식은 Jeon et al. (2024) 대비 TTA가 최대 25% 감소하는 것을 확인할 수 있다.

실제 QZSS CLAS 운용 환경에서는 QZSS L62 신호 각 데이터 파트에 존재하는 예비 영역의 크기가 고정되어 있지 않으므로, 인증에 사용할 예비 영역의 비트 수 X를 Jeon et al. (2024)과 같이 보수적으로 데이터 파트 크기 1695 비트의 5 퍼센트 미만으로 설정할 수 있다. 이에 따라 P-256 곡선을 사용하는 환경에서 이 논문의 프로토콜을 실제로 적용할 경우, 파라미터를 Table 1의 첫번째 행과 같이 X=50, N=12로 설정하는 것이 적합하다.

Table 2. Experimental environment.

	PC	Raspberry Pi
CPU	Intel(R) Core(TM) i7-10700F CPU @ 2.90 GHz	ARM Cortex A53 MP4 @ 1.2 GHz
OS	Ubuntu 24.04.1 LTS (WSL2 2.3.24.0)	Debian GNU/Linux 12 (bookworm)
RAM	32 GB	1 GB
Compiler	GCC 13.3.0	GCC 12.2.2

Table 3. Measured execution time (mean (standard deviation) over 30 batches; each batch is the average of 1,000 iterations).

Operation		PC		Raspberry Pi	
		Jeon et al. (2024)	Proposed	Jeon et al. (2024)	Proposed
Transmitter	ECDSA signature generation (ms)	3.42 (0.03)	3.44 (0.04)	48.57 (0.04)	48.56 (0.05)
	Storing $Data_{t,i}$, $sig_{t,i}$ in frame (μs)	0.18 (0.01)	0.17 (0.01)	2.98 (0.03)	3.00 (0.03)
Receiver	ECDSA signature verification (ms)	6.37 (0.03)		86.13 (0.08)	
	Extracting $Data_{t,i}$, $sig_{t,i}$ from frame (μs)	0.18 (0.01)	0.19 (0.02)	2.78 (0.05)	2.74 (0.01)
	Copying $Data_{temp}$ to $Data_t$ (μs)	1.33 (0.06)	0.72 (0.03)	19.85 (0.80)	8.49 (0.44)

4.2 실행 시간 분석

이번 절에서는 제안하는 프로토콜이 실제 시스템에 적용되었을 때 발생 가능한 연산 부하를 검증하기 위해, 송신자와 수신자의 주요 동작에 대한 실행 시간을 측정하였다. 각 방식의 구현은 C언어로 진행하였으며, ECDSA 서명 생성 및 검증 구현에는 Mbed-TLS 3.1.0 라이브러리를 사용하였다 (Mbed TLS 3.1.0 2026).

실험 환경은 PC 환경과 실제 수신 단말의 제한된 자원을 모사하기 위한 Raspberry Pi 환경으로 나누어 진행하였으며, 구체적인 실험 환경은 Table 2와 같다. Table 3은 Table 2의 실험 환경에서 제안 방식의 수행 시간을 측정하여 Jeon et al. (2024)과 비교한 것이다. 실험 결과에 의하면, ECDSA 서명 생성 및 검증 동작이 전체 실행 시간의 대부분을 차지하는 것으로 나타났다. 제안 방식에서 송신자의 서명 생성 시간은 PC 환경에서 약 3.44 ms, Raspberry Pi 환경에서 약 48.56 ms로 측정되었으며, 이는 Jeon et al. (2024)의 수치와 거의 차이가 없었다. 수신자 측면에서 기존 방식과 제안 방식의 서명 검증 방식은 동일하며, PC에서 약 6.37 ms, Raspberry Pi에서 약 86.13 ms로 측정되었다. 서명 관련 동작(서명 생성 및 검증)을 제외한 나머지 데이터 처리와 관련된 동작들(데이터 및 서명 저장, 추출, 복사 등)은 대부분 μs 단위의 매우 낮은 소요 시간을 보였다. 수신자의 임시 버퍼 $Data_{temp}$로부터 $Data_t$로 복사하는 동작은 두 환경에서 모두 제안 방식이 대략 두 배 빠른 수치를 보였다. 기존 방식에서는 수신자가 t 시점에 받은 $Data_t$에 대한 서명 검증을 위해 그 다음 t+1 시점의 sig_{t+1}를 모두 받아야 가능하며, 이 시점에 수신하는 프레임에 포함된 $Data_{t+1}$의 N개 분할들을 모두 $Data_{temp}$에 저장해 두어야 한다. 그러나 제안 방식에서는 수신자가 $Data_t$ 수신 이후, t+1 시점에서 sig_{t+1}의 분할 중 서명 s_t가 포함된 부분들을 수신한 뒤 서명 검증을 진행하므로, $Data_{temp}$에는 $Data_{t+1}$의 첫 N_S개 분할만 저장하게 되어 복사할 $Data_{temp}$의 크기가 기존 방식에 비해 줄어들며, 따라서 이에 비례하여 복사 동작 시간도 줄어들게 된다.

5. 적용 조건 및 고려사항

이번 장에서는, 이 논문에서 제안한 방식이 실제 CLAS 보정 정보의 검증에 적용될 수 있는 조건에 대해 논의한다. QZSS 규격 (Cabinet Office 2025)에 따르면, CLAS 실시간 보정 정보들 중 대부분은 갱신 주기가 30초, 유효기간이 60초로 정의되어 있으며, 위성시계 보정의 경우 갱신 주기 5초, 유효기간은 10초로 정의되어 있다 (Cabinet Office 2025). 한편, QZSS L62 신호의 프레임당 전송 시간은 30초이며, 이 논문의 제안 방법은 전자서명을 여러 프레임에 분할 전송하므로 전자서명 전송 시간이 위의 갱신 주기나 유효기간에 비해 더 길다. 따라서, 수신자가 보정 정보를 인증 완료된 후 사용하면 보정 정보의 품질을 보장하기 어려우므로, 이 논문에서는 수신자가 실시간 보정 정보를 인증 완료 전에 우선 적용하고, 이후 인증 정보를 통해 사후적으로 검증하는 제한적인 상황을 가정한다. 이러한 가정은 L62 신호의 메시지 중 최소한의 예비 영역을 사용하는 상황을 고려한 Jeon et al. (2024)의 접근 방식에 따른 것이며, 충분한 예비 영역을 인증에 사용할 수 있을 경우, 10초 이내의 TTA를 달성할 수 있을 것으로 예상된다. 다만, 이러한 상황은 이 논문의 연구 범위를 벗어나므로, 수신자가 보정 정보를 반드시 인증 완료 후에만 적용해야 하는 운용에 대해서는 향후 추가적인 연구가 필요할 것으로 보인다.

다음으로, 이 논문에서 제안한 방식과 표준 ECDSA 전자서명의 관계를 설명한다. Protocol 1의 ECDSA 서명 생성 과정에서, 서명 s_t가 0으로 계산되는 예외 상황이 발생할 수 있는데, 표준 ECDSA 서명 생성 과정 (Hankerson et al. 2004)에 따르면, Protocol 1의 단계 2.8에서 생성된 s_t가 0일 경우 난수 k_t를 생성하는 단계 2.1부터 다시 진행해야 한다. 그러나 이 논문에서 제안하는 Protocol 1은 서명의 r_t 부분을 $Data_t$와 함께 사전에 전송하는 구조를 가진다. 따라서 서명 s_t가 계산되는 시점에는 이미 r_t가 수신자에게 전송 완료된 상태이므로, 송신자가 k_t를 변경하여 r_t를 수정하는 것이 구조적으로 불가능하다.

위와 같은 상황에서, 송신자는 s_t가 0이므로 수신자의 서명 검증이 실패할 것이라는 사실을 인지하면서도 해당 값을 그대로 전송하여야 하며, 수신자는 Protocol 2에 따라 "Reject the signature"를 반환하게 된다. 따라서, s_t=0인 상황에는 표준 ECDSA 전자서명과 이 논문의 제안 방식의 동작이 일치하지 않는다. 그러나, P-256 타원곡선을 사용할 때 위와 같이 s_t=0일 확률은 약 2^{-256}으로, 거의 0에 가까운 값이다.

6. 결론

이 논문에서는 QZSS CLAS 보정 메시지에 대한 인증을 도입함에 있어, 제한된 할당 자원과 실시간성의 제약으로 인해 발생하는 TTA를 최소화하기 위한 최적화된 메시지 인증 프로토콜을 제안하였다. 제안 기법은 ECDSA 알고리즘의 구조적 특성을 활용하여, 서명 대상 보정 데이터와 독립적으로 생성할 수 있는 서명 요소 r을 데이터와 함께 사전에 전송하고 s 부분만 해당 데이터 송신 이후에 전송함으로써, 수신자의 서명 검증 시점을 앞당길 수 있도록 설계되었다.

성능 분석 결과, 제안 프로토콜은 기존의 방식 대비 TTA를 최대 25% 단축시킬 수 있다. 또한 PC 및 Raspberry Pi 환경에서의 실험을 통해 제안 기법이 추가적인 연산 오버헤드를 발생시키지 않음을 확인하였으며, 부수적으로 수신자의 임시 저장 버퍼 사용량이 감소되는 이점을 관찰하였다.

이 논문은 CLAS 보정 정보의 짧은 갱신 주기 및 유효기간을 고려하여, 수신자가 보정 정보를 우선 적용하고 이후 인증 정보를 통해 사후적으로 검증하는 운용을 가정하였다. 따라서 만약 수신자가 보정 정보를 반드시 인증 완료 후에만 적용해야 하는 운용을 전제한다면, 현재와 같이 제한된 예비 영역과 기존 메시지 구조 하에서는 이러한 시간 제약을 만족시키기 어렵다. 이 경우에는 인증 정보에 할당되는 예비 영역의 비트 수를 확대하거나, 메시지 구조 자체를 재설계하여 인증 데이터를 더 빠르게 전송할 수 있도록 하는 추가적인 접근이 필요하며, 이러한 구조적 확장 방안은 향후 연구를 통해 다룰 예정이다.

결과적으로 이 논문의 제안 기법은 자원이 제한된 상황에서 수신자가 보정 데이터를 인증하기까지의 지연 시간을 감소시키며, 이러한 지연 시간의 단축은 스푸핑 공격에 대한 노출 시간을 줄이고 수신기의 서비스 가용성을 향상시킬 수 있다. 또한, 이 제안 기법은 QZSS CLAS 뿐만 아니라, 향후 고정밀 측위 보정 서비스를 제공하는 타 위성 항법 시스템에 있어서도 실용적인 저지연 인증 프로토콜의 설계 방안으로 활용될 수 있을 것으로 기대된다.

ACKNOWLEDGMENTS

본 연구는 우주항공청의 재원으로 지원을 받아 수행된 것임 (과제번호: RS-2022-00165802).

AUTHOR CONTRIBUTIONS

Conceptualization, S.-J.Song and M.-K.Lee; methodology, S.-J.Song and M.-K.Lee; formal analysis, S.-J.Song and M.-K.Lee; investigation, S.-J.Song; resources, J.Noh; data curation, S.-J.Song; writing-original draft preparation, S.-J. Song; writing-review and editing, J.Noh and M.-K.Lee; visualization, S.-J.Song; supervision, M.-K.Lee.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

Air Force Research Laboratory 2024a, Chips Message Robust Authentication (Chimera) Enhancement for the L1C Signal: Space Segment/User Segment Interface, Interface Specification, IS-AGT-100 Rev A. https://drive.google.com/file/d/1L9S9EW4KHVpSEpOsDSUcK2rw7ElMLXk5/view

Air Force Research Laboratory 2024b, Timed Efficient Stream Loss-tolerant Authentication (TESLA) Chips Message Robust Authentication (Chimera) Enhancement for the L1C Signal: Space Segment/User Segment Interface, Interface Specification, IS-AGT-101. https://drive.google.com/file/d/1HrIx3hT1T_ebiIHEuyNG7fvPi1v3irKG/view

Anderson, J. 2024, Designing Cryptography Systems for GNSS Data and Ranging Authentication, PhD Dissertation, Stanford University.

Anderson, J. 2025, World's First Authenticated Satellite Pseudorange from Orbit, In Proceedings of the 38th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2025), Baltimore, Maryland, 8-12 September 2025, pp.738-748. https://doi.org/10.33012/2025.20365

Barker, E. 2020, NIST Special Publication 800-57 Part 1 Revision 5, Recommendation for Key Management: Part 1 – General. NIST, Tech. Rep, pp.1-171. https://doi.org/10.6028/NIST.SP.800-57pt1r5

Cabinet Office 2024, Quasi-Zenith Satellite System Interface Specification Signal Authentication Service (IS-QZSS-SAS-001), Cabinet Office. https://qzss.go.jp/en/technical/ps-is-qzss/is_qzss_sas_agree.html

Cabinet Office 2025, Quasi-Zenith Satellite System Interface Specification Centimeter Level Augmentation Service (IS-QZSS-L6-007), Cabinet Office. https://qzss.go.jp/en/technical/ps-is-qzss/is_qzss_l6_007_agree.html

Chen, L., Moody, D., Regenscheid, A., Robinson, A., & Randall, K. 2023, NIST Special Publication 800-186, Recommendations for Discrete Logarithm-Based Cryptography: Elliptic Curve Domain Parameters. NIST, Tech. Rep, pp.1-75. https://doi.org/10.6028/NIST.SP.800-186

Galileo ICD 2022, European GNSS (GALILEO) High Accuracy Service, HAS SIS ICD, Issue 1.0, 2022, European GNSS Service Centre. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_HAS_SIS_ICD_v1.0.pdf

Galileo ICD 2023, European GNSS (GALILEO) Open Service Navigation Message Authentication, OSNMA SIS ICD,

Issue 1.1, 2023, European GNSS Service Centre. https://www.gsc-europa.eu/sites/default/files/sites/all/files/Galileo_OSNMA_SIS_ICD_v1.1.pdf

Hankerson, D., Menezes, A. J., & Vanstone, S. A. 2004, Guide to Elliptic Curve Cryptography (New York: Springer)

Jeon, Y., Kwon, H.-Y., Noh, J., & Lee, M.-K. 2024, Design of Compact Navigation Message Authentication Protocol for Japanese QZSS Centimeter Level Augmentation Service, Journal of Positioning, Navigation, and Timing, 13, 409-423. https://doi.org/10.11003/JPNT.2024.13.4.409

Mbed TLS 3.1.0, The Mbed TLS Contributors, cited 2026 Feb 9, available from: https://github.com/Mbed-TLS/mbedtls/releases/tag/v3.1.0

Perrig, A., Canetti, R., Tygar, J. D., & Song, D. 2000, Efficient authentication and signing of multicast streams over lossy channels, In Proceedings of the 2000 IEEE Symposium on Security and Privacy. S&P 2000, Berkeley, CA, USA, 14-17 May 2000, pp.56-73. https://doi.org/10.1109/SECPRI.2000.848446

Song, S.-J., Noh, J., & Lee, M.-K. 2025, Message–Signature Overlapping Transmission Protocol for Faster Authentication in QZSS-CLAS, In Proceedings of the Korea Software Congress 2025 (KSC 2025), Yeosu, Korea, 16-19 December 2025, pp.911-913

Wu, Z., Liu, R., & Cao, H. 2019, ECDSA-based message authentication scheme for BeiDou-II navigation satellite system, IEEE Transactions on Aerospace and Electronic Systems, 55, 1666-1682. https://doi.org/10.1109/TAES.2018.2874151

Wu, Z., Zhang, Y., Liu, L., & Yue, M. 2020, TESLA-based authentication for BeiDou civil navigation message, China Communications, 17, 194-218. https://doi.org/10.23919/JCC.2020.11.016

Seung-Jun Song received the B.S. in Computer Engineering from Anyang University, in 2024. He is currently pursuing an M.S. in the Department of Electrical and Computer Engineering at Inha University, Korea. His research interests include information security, homomorphic encryption, and AI security.

JaeHee Noh recently works as a senior researcher at Korea Aerospace Research Institute (KARI). She received B.S., M.S. and Ph.D. degrees from Chungnam National University, Department of Electronic Engineering in 2017, 2019 and 2022. Her research interests include designing the satellite navigation signal, GNSS receiver, anti-spoofing techniques and navigation message authentication.

Mun-Kyu Lee received the B.S. and M.S. degrees in Computer Engineering from Seoul National University in 1996 and 1998, respectively, and the Ph.D. degree in Electrical Engineering and Computer Science from Seoul National University in 2003. From 2003 to 2005, he was a senior engineer at Electronics and Telecommunications Research Institute, Korea. He is currently a professor in the Depart-ment of Computer Engineering at Inha University, Korea. His research interests are in authentication, privacy enhancing technology, applied cryptography, blockchain security, and AI security.