ISSN 2383-6318(Print) / ISSN 2383-6326(Online)
KIISE Transactions on Computing Practices, Vol. 26, No. 5, pp. 244-249, 2020. 5
https://doi.org/10.5626/KTCP.2020.26.5.244

KHU-TEE: ARM PSA 기반 IoT 보안 플랫폼

(KHU-TEE: A Security Platform for IoT Devices based on ARM PSA)

정 준 영 [†]　　조 진 성 [††]
(Junyoung Jung)　　(Jinsung Cho)

요 약 IoT 서비스 개발과 적용 사례가 늘어남에 따라 IoT를 구성하는 기기의 취약점에 관한 관심이 높아지고 있다. 이러한 요구에 따라 하드웨어 보안 모듈, Security SoC (System on Chip), TrustZone과 같은 IoT 기기에 적용 가능한 보안 플랫폼이 개발되었다. 특히, ARM은 저사양 IoT 기기 리소스의 안전한 보관 및 보호, 운영을 위해 실행환경의 격리를 제공하는 보안 아키텍처인 PSA (Platform Security Architecture)를 개발하여 상용화 단계에 접어들고 있다. 하지만 PSA 지원 하드웨어의 이해가 부족한 어플리케이션 개발자는 PSA 기반 서비스 개발의 어려움을 겪는 문제를 가지고 있다. 본 논문에서는 이러한 문제를 해결하기 위해 IoT 플랫폼의 보안 요구사항을 분석하고, 이를 기반으로 KHU-TEE라는 PSA 기반의 보안 플랫폼을 제안한다. 또한, PoC (Proof of Concept) 구현 및 사례 연구를 수행하여 제안하는 플랫폼의 보안성과 개발자의 편의성을 검증한다.

키워드: 사물인터넷, 보안, 보안 플랫폼, 보안 서비스

Abstract Advances in computer technology and internet communication have lead to the emergence of a new communication paradigm called IoT, which is rapidly evolving in the form of various services. Accordingly, security requirements of IoT devices are paramount. In order to satisfy the security requirements of IoT devices, various studies have been conducted into technology such as HSM, Security SoC, and TrustZone. PSA can ensure confidentiality and integrity of IoT devices based on its structural features, but conversely, using the security functions of PSA involves the trade-off of increasing development difficulty. To solve this problem, this paper analyzes the security requirements of IoT platform and proposes KHU-TEE, a security platform based on a PSA. In addition, this paper verifies the security and the developer convenience of KHU-TEE by a case study a PoC implementation.

Keywords: IoT, security, security platform, security service

· 이 논문은 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구결과임(NRF-2017R1D1A1B04035914)
· 이 논문은 2019 한국컴퓨터종합학술대회에서 'IoT 디바이스를 위한 Arm PSA 기반 보안 플랫폼 설계'의 제목으로 발표된 논문을 확장한 것임

† 비 회 원 : 경희대학교 컴퓨터공학과 학생
jjy920517@khu.ac.kr
†† 종신회원 : 경희대학교 컴퓨터공학과 교수(Kyung Hee Univ.)
chojs@khu.ac.kr
(Corresponding author임)

논문접수 : 2019년 10월 1일
(Received 1 October 2019)
논문수정 : 2020년 3월 16일
(Revised 16 March 2020)
심사완료 : 2020년 3월 17일
(Accepted 17 March 2020)

정보과학회 컴퓨팅의 실제 논문지 제26권 제5호(2020. 5)

1. 서 론

IoT를 구성하는 대부분의 기기는 저성능으로 개발되기 때문에, 서버와 PC 위주의 기존 보안 솔루션을 적용하기엔 어려움이 있다. 이를 해결하기 위해 ARM에서는 하드웨어 및 소프트웨어 리소스를 두 개의 영역 (Secure world, Non-secure world)으로 격리하여 IoT 기기의 보안성을 향상시키는 ARM 프로세서 기반의 상용 TEE (Trusted Execution Environment)인 TrustZone을 제안하였다[1,2]. 더불어 ARM은 최근 저사양 마이크로컨트롤러를 위한 보안 플랫폼인 PSA (Platform Security Architecture)를 새롭게 공개했다[3].

ARM의 파트너 기업은 PSA를 통해 보안이 강화된 새로운 IoT 플랫폼 및 서비스의 개발을 할 수 있으며, 이를 통해 안전한 IoT 생태계를 구축할 수 있다. 그러나 PSA 적용이 가능한 IoT 서비스를 개발하기 위해 플랫폼 제조사는 ARM과 NDA (Non-Disclosure Agreements)를 맺어야 하며, PSA적용이 가능한 하드웨어 사용이 익숙하지 못한 어플리케이션 개발자는 안전한 IoT 서비스 개발에 어려움을 느낄 수 있다. 따라서 보안 강화에 대한 가능성이 있음에도 불구하고, PSA를 활용한 IoT 서비스의 연구와 개발이 상대적으로 이뤄지지 않고 있다[4].

이를 해결하기 위해 본 논문에서는 ARM PSA를 기반으로 하는 보안 플랫폼인 KHU-TEE를 제안한다. KHU-TEE는 IoT 기기에 필수적인 보안 서비스들로 구성되며, 이를 통해 IoT 기기의 무결성과 기밀성을 보장할 수 있다. 또한, 보안 서비스를 적용할 수 있는 인터페이스를 제공함으로써 IoT 서비스 어플리케이션 개발자의 개발 편의성을 제공한다. 제안하는 KHU-TEE의 검증을 위해 본 논문에서는 PSA 적용 가능한 개발보드를 사용하여 PoC (Proof of Concept) 구현을 한다.

논문의 구성은 다음과 같다. 2장에서는 PSA를 살펴보며, 3장에서는 IoT 보안 플랫폼의 요구사항을 분석하고, 이를 기반으로 PSA 기반의 신뢰 플랫폼인 KHU-TEE와 KHU-TEE의 기능 및 API를 설계한다. 4장에서는 사례 기반 PoC 구현 및 사례 연구를 통해 제안하는 플랫폼을 검증하고, 마지막으로 5장에서는 결론 및 향후 연구 방향을 제시한다.

2. ARM PSA

ARM은 IoT 디바이스의 위협 모델과 보안 분석에 대한 사전 조사를 반영하여 PSA 보안 모델을 설계했다[5,6]. PSA 보안 모델은 실행환경의 격리를 제공함으로써 상대적으로 신뢰할 수 없는 플랫폼 리소스가 신뢰할 수 있는 플랫폼 리소스를 손상하지 않도록 보장하는 것을 목표로 한다[7]. 신뢰할 수 있는 플랫폼 리소스를 PSA Root of Trust (RoT)라 하며, PSA RoT 중 가장 안전한 격리 방법은 TrustZone-based RoT이다[8,9]. TrustZoone-based RoT가 가능한 프로세서는 ARMv8-M 패밀리의 Cortex-M23, Cortex-M33이 있으며, 이 프로세서들은 IoT 보안 서비스를 제공하는 Secure state 혹은 IoT 서비스 어플리케이션이 동작하는 Non-secure state로 실행할 수 있다. 두 실행 상태의 구분은 메모리 맵을 통해 이뤄지며, 서로 격리되어 기밀성을 유지한다. 하지만 Secure state의 소프트웨어 개발은 PSA 지원 하드웨어의 이해가 필요하므로 가용성 측면의 어려움이 있다.

ARMv8-M 프로세서의 상태 전환에는 두 가지 경우가 있다. Non-secure state에서 Secure state로의 전환은 비보안 영역의 코드가 보안 영역의 함수를 호출할 때 수행된다. 보안 영역의 함수 호출이 수행될 때, 비보안 영역으로의 올바른 리턴을 위한 분기점(Entry point)을 안전하게 저장해야 하며, 이를 위해 ARMv8-M 프로세서는 NSC (Non-Secure Callable)라는 영역을 보안 영역에 할당하고, NSC에 분기점을 저장하기 위한 SG (Secure Gateway) 명령어를 제공한다. 보안 영역의 함수 동작이 완료되면, NSC에 저장된 분기점으로 돌아가기 위해 BXNS 명령어를 사용하며 Secure state가 Non-secure state로 전환된다. 반대로, 보안 영역의 코드가 비보안 영역의 함수를 호출하면 Secure state에서 Non-secure state로 상태가 전환된다. ARMv8-M 프로세서는 상태 전환 중에 보안 영역으로의 리턴 주소 및 프로세서 상태 정보를 보안 스택으로 푸시하며 LR (Link Register)의 주소를 FUNC_RETURN이라는 특수 값으로 설정한다. 이를 위해 ARMv8-M 프로세서는 BLXNS 명령어를 실행하고, 현재 시스템 상태를 Non-secure state로 전환한다. 비보안 영역의 함수 실행이 완료되면 FUNC_RETURN으로 설정된 LR의 주소로 분기가 실행되고, 이로 인해 보안 스택에 저장된 프로세서 상태 정보를 언스택하여 반환하게 된다.

3. PSA 기반 보안 플랫폼: KHU-TEE

3.1 보안 플랫폼 요구사항

본 논문에서 제안하는 보안 플랫폼은 IoT 기술의 요구사항을 정하는 글로벌 표준 단체, oneM2M의 표준을 준수하여 설계한다. oneM2M의 TS-0002-Requirements 표준 문서에 의하면, M2M (Machine To Machine) 디바이스는 표 1과 같은 보안 요구사항을 만족해야 한다[10].

3.2 KHU-TEE 설계 및 기능

제안하는 KHU-TEE는 그림 1과 같이 ARMv8-M 프로세서 기반의 SoC와 외부 주변 장치로 구성된다.

SoC는 펌웨어가 저장되는 NVM (Non Volatile Memory)과 디바이스 제조 과정에서 주입된 프로비저닝 데이터

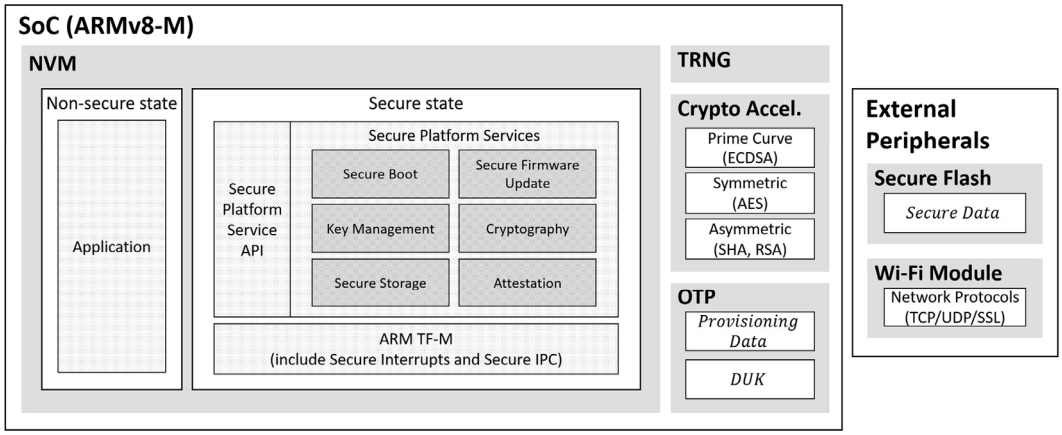

그림 1 KHU-TEE의 시스템 아키텍처
Fig. 1 System Architecture of KHU-TEE

표 1 oneM2M의 TS-0002-Requirements
Table 1 TS-0002-Requirements of oneM2M

Code	Requirement
SER-002	The oneM2M system shall be able to ensure the confidentiality of data.
SER-003	The oneM2M system shall be able to ensure the integrity of data.
SER-013	The oneM2M system shall be able to provide the mechanism for integrity-checking on boot, periodically on run-time, and on software upgrades for software/hardware/firmware component(s) on M2M Device(s).
SER-064	The M2M devices shall provide a mechanism to prevent installation or modification of the software/middleware/firmware which run on the M2M devices, unless it is authorized by an allowed stakeholder.
SER-065	The oneM2M system shall be able to detect installation or modification of the software/middleware/firmware of M2M Devices that has not been authorized by an allowed stakeholder.
SER-068	The information exchanged within the oneM2M system shall use cryptographic technology to ensure information authentication and information integrity.
SER-069	The oneM2M system shall be able to securely transfer information by using an appropriate method such as digital signature.

(Provisioning Data) 및 DUK (Device Unique Key)를 저장하는 OTP (One-Time Programmable) 메모리가 있다. 그리고 TRNG (True Random Number Generator)와 타원 곡선 암호 알고리즘, 대칭/비대칭 키 암호화 알고리즘을 포함하는 암호화 가속기(Crypto Accel.)로 구성된다. 이 암호화 가속기는 외부 주변 장치 중, 암호화된 데이터(Secure Data)를 저장하는 Secure Flash 메모리에 직접 접근할 수 있지만, 메인 프로세서는 Secure Flash 메모리에 직접 접근이 불가하다. 또 다른 외부 주변 장치로는 TCP, UDP, SSL과 같은 네트워크 프로토콜을 통해 외부와 통신을 할 수 있는 Wi-Fi Module이 있다.

SoC의 보안 상태는 NVM의 메모리 맵 구성에 따라 Non-Secure state와 Secure state로 구분된다. Non-secure state는 IoT 서비스를 제공하는 어플리케이션 (Application)이 있고, Secure state는 안전한 인터럽트 (Interrupt) 및 IPC (Inter-Process Communication)를 제공하는 ARM TF-M과 ARM TF-M을 기반으로 하는 6가지 *Secure Platform Services*가 있다. Secure Platform Service에 대한 세부 사항은 아래와 같다.

- **Secure Boot**: KHU-TEE의 부팅 중 부트 구성 요소의 무결성을 검증한다. Reset 신호가 메인 프로세서에 작용하면, 기기가 부팅되고 Secure state와 Non-secure state의 무결성을 확인한다. Secure Boot는 [10]의 SER-003, SER-013을 만족한다.
- **Secure Firmware Update**: IoT 서비스 어플리케이션 개발자가 개발한 현재 Non-secure state의 어플리케이션과 새로운 어플리케이션의 무결성을 검사한다. Secure Firmware Update는 [10]의 SER-003, SER-064를 만족한다.
- **Attestation**: KHU-TEE의 구성 요소를 검사한다. Attestation 결과는 런타임 및 부팅 때 기기 검증에 사용되며, 대표적인 Attestation의 사용 사례는 Remote Attestation이다[11]. Attestation은 [10]의 SER-003, SER-013, SER-064, SER-069를 만족한다.
- **Cryptography**: 암호화 가속기의 타원 곡선 알고리즘 및 대칭/비대칭 키 암호화 알고리즘 가속기의 사용과 TRNG의 사용을 제공한다. Cryptography는 [10]의 SER-002, SER-003, SER-068, SER-069를 만족한다.
- **Secure Storage**: 암호화 가속기를 통해 Secure Flash 메모리 영역에 Secure data를 읽고 쓰는 기능을 제공한다. Secure Storage는 [10]의 SER-002, SER-003, SER-068, SER-069를 만족한다.
- **Key Management**: 프로비전된 암호화 키와 DUK에

표 2 제안하는 Secure Platform Service APIs
Table 2 Secure Platform Service APIs

Service	Function	Parameter(s)	Return
Attestation	int getMeasure(measureInfo_t data, measureInfo_t *result)	Measured information of device data, Result of measurement	1(Success) or 0(Failure)
Cryptography	int getRand( )	-	Random integer value by TRNG
	int ecdsa(ecdsaContext *ctx, type sign_verify, char *in, size_t in_len, char *out, size_t out_len, int rng)	ECDSA context, Sign or Verify, Input data, Size of input data, Output data, Size of output data, Random integer value	1(Success) or 0(Failure)
	int aes128(int keyID, type enc_dec, char *in, size_t in_len, char *out, size_t *out_len)	ID of derived key, Input data, Size of input data, Output data, Size of output data	1(Success) or 0(Failure)
	int sha256(char *plain, size_t plain_len, char *digest, size_t *digest_len)	Plain data, Size of plain data, Digest data, Size of digest data	1(Success) or 0(Failure)
	int rsa1024(int keyID, type enc_dec, char *in, size_t in_len, char *out, size_t *out_len)	ID of derived key, Input data, Size of input data, Output data, Size of output data	1(Success) or 0(Failure)
Secure Storage	int storeData(char *cipher, int size)	Cipher data, Size of cipher data	Page ID where data are stored (pageID)
	void loadData(int pageID, int size)	Page ID where data are stored, Size of cipher data	-
Key Management	int genKey()	-	ID of the derived key (keyID)
	void delKey(int keyID)	ID of the derived key	-

서 생성된 Derived Key의 사용을 제공한다. Key Management는 [10]의 SER-002, SER-003, SER-068, SER-069를 만족한다.

한편, KHU-TEE는 표 2와 같이 IoT 서비스 어플리케이션 개발자에게 편의성과 빠른 개발을 제공하기 위해, Non-secure state의 어플리케이션에서 Secure Platform Service를 호출 할 수 있는 *Secure Platform Service API*를 제공한다. Secure Boot와 Secure Firmware Update 서비스는 KHU-TEE의 시스템 보안을 위해 존재하기 때문에 어플리케이션 개발자는 고려할 필요가 없다.

4. PoC 구현 및 검증

4.1 KHU-TEE

본 장은 3장에서 설계한 KHU-TEE의 PoC 구현을 통해 제안하는 KHU-TEE의 보안성과 유용함을 증명하고자 한다. 본 PoC 구현은 Nuvoton의 *NuMaker-PFM-M2351* 보드[12]를 사용했다. [12]는 TrustZone-based RoT를 지원하는 ARM Cortex-M23 프로세서 기반의 M2351KIAAE 칩이 포함된 상용 보드다. 따라서 PSA 기반의 보안 플랫폼인 KHU-TEE의 포팅이 가능하다. 또한, M2351KIAAE는 SPI 통신 및 UART 통신을 통해 W77F32W 칩[13]과 ESP8266 모듈[14]에 연결되며, 각각 Secure Flash 메모리와 Wi-Fi Module로 사용한다. 이를 통해 제안하는 KHU-TEE가 포팅된 [12]를 사용하여 Secure Platform Service와 Secure Platform Service API의 사용이 가능하다. 그리고 어플리케이션 개발자는 이를 통해 다양한 분야의 IoT 보안 서비스를 손쉽게 구현할 수 있다.

4.2 커넥티드 카 보안 서비스 구현 사례

본 장에서는 앞서 PoC 구현에 사용된 M2351KIAAE 칩이 CAN 통신 프로토콜을 지원하고 OTA (Over-The-Air) 펌웨어 업데이트가 가능하다는 특징을 활용하여, 커넥티드 카 (Connected Car) 개발 회사의 자동차 보안 서비스 구현 사례를 보인다. 그리고 이를 통해 자동차 ECU (Electronic Control Unit)에 적용된 KHU-TEE를 검증한다.

본 사례의 커넥티드 카 ECU는 연결된 센서에서 감지한 센서 데이터를 W77F32W에 암호화하여 안전하게 저장한다. 그리고 주기적으로 센서 데이터를 ESP8266을 사용하여 안전한 채널을 통해 클라우드 서버로 전송한다. 이를 위해 프로비저닝 과정에서 클라우드 서버에 ECU의 공개 키(*PubK.ECU*)와 ECU의 고유 ID (*ECU.ID*)를 저장하는 것을 가정한다. 또한, ECU 제조 과정에서 ECU의 개인 키(*PriK.ECU*)와 *ECU.ID*를 M2351KIAAE의 OTP 메모리에 저장하고, 센서 데이터의 무결성을 위해 W77F32W 칩에 DUK를 통해 생성한 Derived Key로 암호화하여 저장한다.

그림 2는 커넥티드 카의 어플리케이션 개발자가 개발한 Non-secure state 어플리케이션의 동작 과정이다.

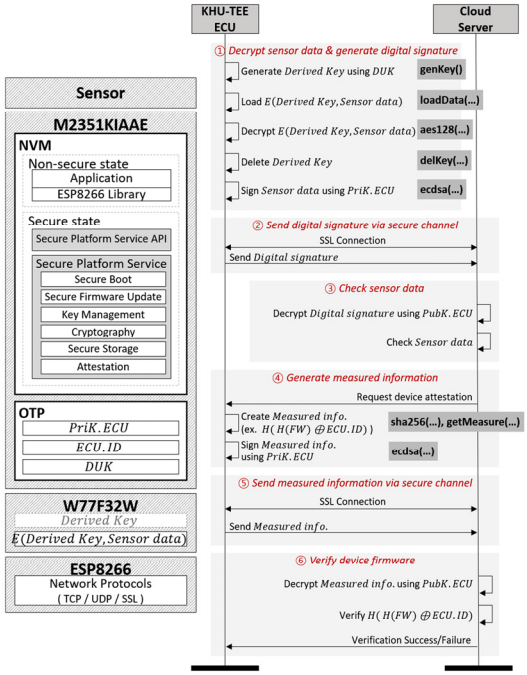

그림 2 KHU-TEE 기반 ECU의 시퀀스 다이어그램
Fig. 2 Sequence diagram for ECU based on KHU-TEE

어플리케이션 개발자는 Secure Platform Service API의 사용을 통해 Secure Platform Service를 호출 할 수 있다. 커넥티드 카 ECU의 안전한 센서 데이터 전송 과정은 다음과 같다.

① 저장된 센서 데이터 복호화 및 서명 생성

A. Key Management Service: Derived Key 생성/제거(*genKey()*, *delKey()* 사용)

B. Secure Storage service: 암호화 센서 데이터 로드(*loadData()* 사용)

C. Cryptography service: 센서 데이터 복호화 및 서명 생성(*aes128()*, *ecdsa()* 사용)

② 안전한 채널을 통한 서명 전송

③ 클라우드 서버의 센서 데이터 검사

그러나 ECU가 전송한 센서 데이터의 이상이 있을 시, ECU 해킹 여부를 판단하기 위해 디바이스 검증 과정을 거친다. 디바이스 검증 단계는 다음과 같다.

④ 검증 데이터 생성

A. Cryptography service: 데이터 해쉬 및 서명(*sha256()*, *ecdsa()* 사용)

B. Attestation service: 검증 데이터 생성(*getMeasure()* 사용)

⑤ 안전한 채널을 통한 검증 데이터 전송

⑥ 서버의 디바이스 검증

또한, ECU의 부팅 및 펌웨어 업데이트 시, 어플리케이션 개발자의 개입 없이 자동 동작하는 Secure Boot와 Secure Firmware Update의 동작 과정은 다음과 같다.

① Secure Boot service: ECU 리셋 시, ECU 부트로더의 Secure state 및 Non-secure state 펌웨어 무결성 검증

A. ECU 펌웨어 교체 공격 방지

② Secure Firmware Update service: 클라우드 서버의 OTA 펌웨어 업데이트 시, ECU 부트로더의 펌웨어 버전 및 펌웨어 무결성 검사

A. 원격 서버를 통한 ECU 펌웨어 교체 공격 방지

B. MITM (Man-In-The-Middle) 공격 방지

C. 펌웨어 롤백 (Rollback) 공격 방지

이처럼 어플리케이션 개발자는 Secure Platform Service API를 사용하여 보안을 요구하는 IoT 서비스를 쉽게 구현할 수 있으며, 시스템 보안 서비스를 통해 IoT 디바이스 보안 위협을 안전하게 방지할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 ARMv8-M 프로세서를 사용한 PSA 기반 보안 플랫폼과 플랫폼의 보안 기능 및 API를 제안하였다. 그리고 커넥티드 카 보안 서비스 구현 사례의 PoC를 통해 제안하는 플랫폼을 증명하였다. 이를 통해 안전한 IoT 서비스의 개발을 원하는 어플리케이션 개발자들은 하드웨어의 특징 및 암호화 기술 연구에 노력할 필요 없이 쉽게 PSA 서비스를 이용할 수 있다. 향후 연구로는 안전한 데이터 저장소 [15]로서 SE (Secure Element)를 사용하는 PSA 기반 보안 플랫폼을 개발할 예정이다.

References

[1] S. Ray, E. Peeters, M. Tehranipoor and S. Bhunia, "System-on-Chip Platform Security Assurance: Architecture and Validation," *Proc. Of the IEEE*, Vol. 106, No. 1, pp. 21, 2018.

[2] Arm Limited, "Arm Security Technology Building a Secure System using TrustZone Technology," 2009.

[3] Arm Limited, "Arm Platform Security Architecture Overview Revision1.2," 2018.

[4] S. Pinto and N. Santos, "Demystifying Arm TrustZone: A Comprehensive Survey," *ACM Computing Surveys*, Vol. 51, No. 6, pp. 1-130, Feb. 2019.

[5] Arm Limited, "Water Meter Threat Model and Security Analysis (English language Protection Profile) Beta-1," 2018.

[6] Arm Limited, "Arm Platform Security Architecture Firmware Framework 1.0 Beta-2," 2019.

[7] Arm Limited, "Arm Platform Security Architecture Security Model 1.0 Alpha-2," 2019.

[8] Arm Limited, "Arm Platform Security Architecture Trusted Base System Architecture for Armv6-M, Armv7-M and Armv8-M 1.0 Beta-1," 2019.
[9] T. Alves and D. Felton, "TrustZone: Integrated hardware and software security," ARM white paper 3, 4, pp. 18-24, 2004.
[10] oneM2M Partners, "TS-0002-Requirements-V4.6.0," 2019.
[11] C. Kil, E. C. Sezer, A. M. Azab, P. Ning, and X. Zhang, "Remote attestation to dynamic system integrity evidence," *2009 IEEE/IFIP International Conference on Dependable System & Networks*, Lisbon, pp. 115-124, 2009.
[12] Nuvoton Inc., "NuMaker-PFM-M2351 User Manual Rev 1.00," 2018.
[13] Winbond Electronics Corporation, "W75F32W 32M-bit Secure Serial Flash Memory Security Target Revision B," 2017.
[14] Espressif Inc., "ESP8266 Technical Reference Version 1.4," 2019.
[15] Global Platform Inc., "Secure Elements Configuration V2.0," 2018.

정 준 영

2018년 경희대학교 전자공학과, 컴퓨터공학과 학사. 2020년 경희대학교 대학원 컴퓨터공학과 석사. 관심분야는 IoT 보안, 시스템 보안, 임베디드 시스템. 2019년~현재 현대오트론 전자플랫폼팀 연구원

조 진 성

1992년 서울대학교 컴퓨터공학과 학사 1994년 서울대학교 대학원 컴퓨터공학과 석사. 2000년 서울대학교 대학원 컴퓨터공학과 박사. 1998년 IBM T.J. Watson Research Center Visiting Researcher 1999년~2003년 삼성전자 책임연구원. 2003년~현재 경희대학교 컴퓨터공학과 교수. 관심분야는 IoT 보안, 시스템 보안, 임베디드 시스템