# Arm TrustZone 과 CCA 의 비교 연구

황윤성 [1], 백윤흥 [2]
[1] 서울대학교 전기· 정보공학부 석박통합과정, 반도체공동연구소
[2] 서울대학교 전기· 정보공학부 교수, 반도체공동연구소
yshwang@sor.snu.ac.kr, ypaek@snu.ac.kr

# A Comparative Study of Arm TrustZone and CCA

Yun-Seong Hwang [1], Yun-Heung Paek [1]
[1]Dept. of Electrical and Computer Engineering and Inter-University
Semiconductor Research Center (ISRC), Seoul National University

**요 약**

클라우드 컴퓨팅 환경에서 사용자의 민감한 데이터를 보호하기 위해 하드웨어 기반 Trusted Execution Environment (TEE)의 중요성이 커지고 있다. 본 논문은 Arm 의 대표적인 TEE 기술인 TrustZone 과 Confidential Compute Architecture (CCA)를 비교 분석한다. 두 기술의 보안 모델과 확장성을 중심으로, Trusted Computing Base(TCB), memory encryption, 그리고 remote attestation 의 관점에서 두 기술의 차이를 확인한다. 하지만, 상용 CCA 하드웨어의 부재는 현재 CCA 관련 연구에 큰 제약이 되고 있다. 이러한 어려움을 확인하기 위해, 우리는 Arm Fixed Virtual Platform(FVP) 환경에서 새로운 Realm Service Interface(RSI)를 구현하는 실험을 진행했다. 실험 결과, 기능적 확장은 가능했으나 Realm VM 과 RMM 간 통신에서 상당한 instruction-level 오버헤드가 발생함을 확인했다. 이는 CCA 상에서 Realm VM 과 RMM 간 통신이 수반하는 비용과 하드웨어 부재 속에서 정확한 성능 예측의 어려움을 명확히 보여준다.

## 1. 서론

최근 클라우드에서 민감한 작업을 수행하고자 하는 수요가 늘어나고 있다. 클라우드에서 수행되는 사용자의 작업에는 비밀번호, 고유 생체 정보 등의 사용자의 개인적인 정보가 많이 포함되어 있으며, 만약 클라우드가 악의적이거나 침해당한다면 이러한 정보는 쉽게 노출될 수 있다. 그러한 위협으로부터 사용자 작업에 대한 기밀성 보장하는 것이 중요해지고 있으며, 하드웨어 기반 Trusted Execution Environment(TEE)가 그 해결책으로 주목받고 있다.

TEE 는 신뢰할 수 없는 서버 환경 속에서 사용자의 작업을 보호한다. Memory Management Units(MMU)나 Memory Encryption Engine(MEE)를 바탕으로 한 system-on-chips(SoCs)와 같은 하드웨어 Trusted Computing Base(TCB)를 바탕으로 TEE 는 사용자에게 안전하고 격리된 실행 환경을 제공한다.

Intel, AMD, Arm 등의 프로세서 제조사들은 각자 하드웨어에 맞는 신뢰 실행 환경 기술들(Intel TDX[1], AMD SEV[2], Arm TrustZone[3], Arm CCA[4])을 제공하고 있다. 특히, 모바일 기기에 주로 사용되는 Arm 아키텍처에서는 Arm TrustZone 이 가장 대표적인 하드웨어 기반 TEE 기술로써 도입되었다 [5]. TrustZone 은 전체 시스템을 Normal World 와 Secure World 로 구분되어 신뢰할 수 없는 호스트 환경으로부터 사용자의 코드와 데이터에 대한 접근을 보호하는 것을 목적으로 한다. 하지만, TrustzZone 은 Secure World 에서 사용되는 메모리 영역을 정적으로 나누고 격리하며 대부분 적은 수의 메모리 영역만을 지원할 수 있기 때문에 확장성이 부족하다. 뿐만 아니라, 원격의 사용자가 TEE 인스턴스에 대한 초기 상태의 무결성을 확인할 수 있는 remote attestation 을 지원하지 않기 때문에, 클라우드 환경에서 사용하기에는 적합하지 않다.

2021 년에 Arm 은 Armv9-A 부터 도입되는 Confidential Compute Architucture(CCA) [4]를 제시했다. CCA 는 Realm World 이라는 새로운 격리 공간을 도입했으며, 최소한의 신뢰 기반(minimal TCB) 위에서 증명 가능한 신뢰(attestable trust)를 제공하는 것을 목표로 한다 [6].

본 논문에서는 TrustZone 과 CCA 가 제공하는 보안성에는 어떤 차이가 존재하는지 비교 및 분석한다. 더 나아가 CCA 의 확장성을 살펴보고 상용 하드웨어의 부재가 CCA 기반 연구에 미치는 어려움을 Arm Fixed Virtual Platform(FVP)[11] 기반의 실험을 통해 살펴본다.

## 2. 배경이론

### 2.1 Arm TrustZone

그림 1 은 TrustZone 의 전체적인 architecture 를 보여주고 있다. TrustZone 은 4 개의 Exception Levels(EL0, EL1, EL2, and EL3)를 가지며 2 개의 world(Normal world, Secure world)를 가진다. Processor 는 두 개의 world 중 하나에 속해 동작하며, processor 의 world 는 processor 의 33 번째 bit 인 Non-Secure(NS) bit 에 의해 결정된다. Normal world 에서는 Rich OS 와 그 위에서 동작하는 normal application 들이 실행되는 환경이다. 반면, Secure world 는 Trusted OS 와 그 위에서 동작하는 trusted application 들이 실행되는 환경이다.

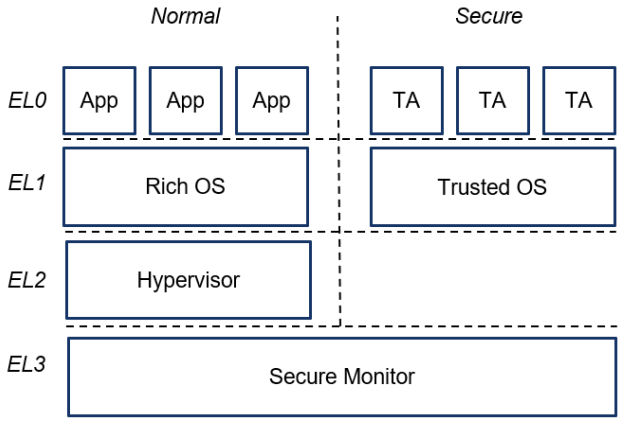


**그림 1. Arm TrustZone**

TrustZone 에서는 두 world 에 대한 isolation 을 위해, TrustZone Address Space Controller(TZASC)[7]와 TrustZone Protection Controller(TZPC)[8]를 도입했다. TZASC 는 특정 memory region 이 secure 인지 non-secure 인지를 설정할 때 사용된다. 예를 들어, TZASC 는 Secure world 에서 동작하는 application 이 non-secure memory region 을 접근하는 것은 허용하지만, 그 반대는 허용하지 않는다. TZPC 는 주변 기기들의 security state 를 설정할 때 사용되며, 접근의 허용 여부를 결정한다.

TrustZone 에서는 Secure Monitor Call(SMC) instruction 을 제공하며, 이를 통해 두 개의 world 간의 이동을 수행한다. Normal world 에서 Secure world 에 진입하고 싶을 경우 SMC instruction 을 수행하여 processor 가 monitor mode 에 진입한 뒤 Secure world 에 진입하게 된다. Normal world 로 돌아오는 것 역시 SMC instruction 을 통해 수행된다.

### 2.2 Arm Confidential Compute Architecture (CCA)

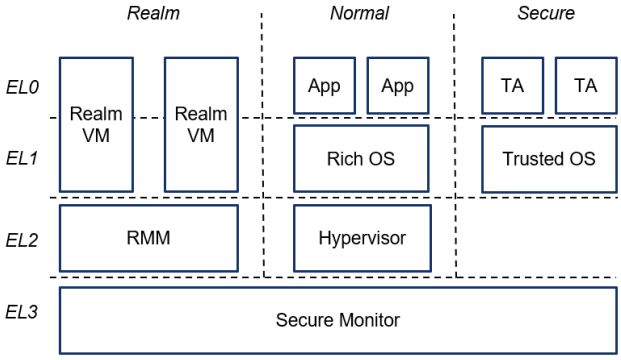


**그림 2. Arm CCA Architecture**

그림 2 는 CCA 의 전체적인 architecture 를 보여주고 있다. CCA 역시 TrustZone 과 마찬가지로 4 개의 Exception Level 을 가진다. CCA 는 기존의 Normal world, Secure world 에 더해 두 개의 추가적인 world(Root world, Realm world)를 도입했다. 그에 따라 CCA 에는 4 개의 security state(NS, Secure, Realm, Root)가 존재하며, 각 world 는 고유의 Physical Address Space(PAS)를 갖는다. Memory granule 이라 불리는 각 4KB 의 physical memory frame 은 특정 하나의 PAS 에 속하게 된다. PAS 를 바탕으로 한 CCA 의 access control policy 는 표 1 과 같다. Memory access 들이 control policy 를 만족하는지를 확인하기 위해, MMU 는 각 memory granule 의 PAS 를 저장하는 memory Granule Protection Table(GPT)를 바탕으로 Granule Protection Check(GPC)를 수행한다.

| Security State | PAS | | | |
|---|---|---|---|---|
| | NS | Secure | Realm | Root |
| NS | O | X | X | X |
| Secure | O | O | X | X |
| Realm | O | X | O | X |
| Root | O | O | O | O |

**표 1. Arm CCA access control**

CCA 에서는 Memory Protection Engine(MPE)라는 하드웨어를 기반으로 memory encryption 을 지원한다.

Realm world 에는 Realm Management Monitor(RMM)이 존재하며, 이는 Normal world 의 host 가 생성한 Realm world Realm virtual machines(RVMs)을 관리한다. RMM 은 Normal world hypervisor 에게 Realm Management Interface(RMI)를 제공하여, Realm 에 대한 생성, 제거, memory 할당 등의 동작을 지원한다. 추가로, RVM 에게는 Realm Services Interface(RSI)를 제공하여 RMM 에게 service 를 요청할 수 있게 한다. RMI 와 RSI 는 모

두 SMC 를 통해 구현된다.

### 3. Arm TrustZone 과 CCA 보안 비교

TrustZone 과 CCA 는 서로 다른 보안 정도를 가지며, TrustZone 은 CCA 의 보안 모델 상에서는 충분한 보안성를 제공하지 못한다.
기본적으로 TrustZone 은 CCA 에 비해 매우 큰 Trust Computing Base(TCB)를 갖는다. TrustZone 의 목적은 Normal world 로부터 Secure world 를 격리하는 것이기 때문에, TrustZone 의 TCB 는 Secure Monitor, Trusted OS, 그리고 모든 Trusted application 들을 포함한다. 따라서, 이들 중 한 곳이라도 취약점이 발견되면 전체 Secure world 가 compromise 될 수 있다. 이를 악용하여, 공격자는 Trusted Applcation 에 대한 full control 을 얻거나, Trusted OS 에 대한 full control 을 얻어 전체 Secure world 를 위협할 수 있다 [9]. 이와 달리 CCA 에서는 새로운 Realm world 를 제공하여 TCB 에서 Secure world 를 제외시키고 Realm world 내의 RMM 은 RVMs 에 대한 생성, 소멸, 스케줄링과 같은 최소한의 management 기능만을 수행하도록 하여 TCB 를 최소화하였다. 또한, VM level 의 격리를 제공함으로써 하나의 RVM 이 compromise 되더라도 전체 Realm world 에 문제가 생기지 않도록 했다.
또한, TrustZone 에는 하드웨어 기반의 memory encryption 기능이 존재하지 않는다. Secure world 의 memory 역시 DRAM 에 그대로 존재하기 때문에 cold boot attack[10]과 같은 physical attack 에 취약하다. 반면, CCA 에서는 MPE 를 기반으로 한 memory encryption 을 지원하며 Secure world, Realm world, Root world 에 대한 memory 는 각 world 의 encryption key 를 바탕으로 encrypt 된 상태로 존재한다.
마지막으로 TrustZone 은 local attestation(i.e., secure boot) 만을 지원하며 remote attestation 이 지원하지 않는다. 반면, CCA 에서는 RMM 이 crypto library 의 함수를 사용해 각 RVMs 에 대한 remote attestation 을 지원한다.
결론적으로, 표 2 와 같이 CCA 는 TrustZone 에 비해 간소화된 TCB 를 가지며, memory encryption 과 remote attestation 을 지원하여 보다 강한 보안을 제공한다.

| 기능 | TrustZone | CCA |
|---|---|---|
| TCB | TA, Trusted OS, Secure Monitor | RVM, RMM, Secure Monitor |
| Memory Encryption | X | O |
| Remote Attestation | X | O |

**표 2. Arm TrustZone 과 CCA 보안 비교**

### 4. CCA 의 확장성 및 하드웨어 부재로 인한 어려움

CCA 는 VM 단위의 confidential computing 을 제공한다. 기존의 정적으로 memory region 을 할당하며 그 개수에도 제한이 있었던 TrustZone 과 달리 동적으로 RVM 을 생성하고 소멸시킬 수 있으며, RVM 에 할당된 memory resource 역시 유연하게 변경하게 할 수 있다. 따라서, CCA 는 많은 workload 가 만들어지고 사라지는 클라우드 환경에 사용하기 적합하다. 또한, 다양한 하드웨어 component 를 기반으로 RVM 에 대한 confidentiality, integrity, attestation 을 보장하여 원격의 클라우드 내에서 안전한 실행 환경을 제공한다.
현재 CCA 관련 연구는 활발히 진행되고 있다. CCA 의 개념을 GPU 하드웨어까지 확장하여 confidential GPU computing 을 지원하는 연구가 진행되었다 [14]. 또한, CCA 의 VM level 을 넘어서 하나의 process 안에서 무수히 많은 수의 isolation domain 을 제공하는 intra-process isolation 연구도 진행되었다 [15].
하지만, 현재까지 상용화된 CCA 하드웨어가 없기 때문에, 새로운 design 의 기능성 및 성능을 측정하기 위해서 Arm 에서 제공하는 software 기반 에뮬레이션 환경인 Arm Fixed Virtual Platform(FVP)[11]에 CCA 의 기능들을 software 로 구현하고 그 위에 새로운 design 을 구현하는 시도들이 이루어졌다. 하지만 FVP 는 instruction-accurate 하지만 cycle-accurate 하지는 않으며 모든 instruction 의 실행시간을 1 cycle 로 측정하기 때문에 정확한 성능을 측정하기에는 무리가 있다 [12, 13]. 따라서, 기능성의 확인은 FVP 를 활용하고, Arm Juno board 에 software 를 바탕으로 CCA 의 기능들과 새로운 design 을 구현해 성능을 측정한 연구도 존재한다 [14].

### 5. 구현 및 실험 결과

FVP 는 cycle-accurate 한 성능 측정이 어렵다는 한계가 있지만, instruction-accurate 하다는 장점을 가진다. 본 연구에서는 이러한 FVP 의 특성을 활용하여, CCA 에 새로운 RSI 를 추가했을 때의 기능성을 확인하고, 실행에 필요한 비용을 instruction count 를 통해 분석하는 초기 실험을 진행하였다

FVP_Base_RevC-2xAEMvA 를 바탕으로 한 CCA stack[16] 상에서 실험을 진행했으며, 고정된 intermediate physical address 를 translate 해 그에 대한 physical address 를 print 하는 RSI 와 해당 RSI 를 호출하는 system call 을 추가했다. userspace 에서 system call 을 바탕으로 새로운 RSI 를 trigger 했다. 그 결과, 제공한 intermediate physical address 를 기반으로 RMM 에서 translation 을 수행하는 것을 확인하였으며, 해당 RSI 를 실행하는데 걸리는 instruction count 는 10256379 instruction 이 측정되었다.

매우 많은 instruction count 가 측정되었는데 이는 RVM 과 RMM 간의 communication 과정에서 해당 RSI 에 대한 handling 과 그 이외에 다른 동작들이 추가적으로 실행되어 나타난 결과로 판단된다. 불필요한 과정을 제외할 수 있도록 환경을 제한한 후, 추가된 RSI 를 실행한다면 보다 정확한 instruction count 를 측정할 수 있을 것이다.

### 6. 결론

본 논문에서는 Arm 의 대표적인 두 TEE 기술인 TrustZone 과 CCA 의 전반적인 architecture 와 제공하는

보안 정도의 차이를 살펴보았다. 분석 결과, TrustZone 의 거대한 TCB 와 달리 CCA 는 Realm world 를 Secure World 로부터도 격리하여 TCB 를 최소화했으며, memory encryption 과 remote attestation 기능을 통해 더 강력한 보안을 제공합니다. 또한, TrustZone 의 정적인 자원 할당 방식과 달리 CCA 는 동적으로 RVM 을 생성하고 자원을 유연하게 관리할 수 있어 클라우드 환경에 필수적인 확장성을 확보했습니다. 하지만 CCA 의 상용화된 하드웨어의 부재로 인해, 관련 연구들이 기능 구현 및 성능 측정에 어려움을 겪고 있음을 확인했다. 연구들은 기능 검증을 위해 FVP 와 같은 에뮬레이션 환경을 활용한다. 이러한 어려움을 확인하기 위해 FVP 기반의 RSI 추가 실험을 진행했다. 이를 통해 기능성은 확인했으나, instruction count 측정 결과 막대한 결과값을 보였다. 이는 FVP 환경에서의 실험이 cycle-accurate 하지는 않더라도 기능성의 확인과 instruction-level 의 비용을 가늠하는데 중요한 의미를 가짐을 보여준다.

## ACKNOWLEDGEMENT

이 논문은 2025 년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (RS-2023-00277326). 이 논문은 2025 년도 BK21 FOUR 정보기술 미래인재 교육연구단에 의하여 지원되었음. 본 연구는 반도체 공동연구소 지원의 결과물임을 밝힙니다. 이 논문은 2025 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2021-0-00528, 하드웨어 중심 신뢰 계산기반과 분산 데이터보호박스를 위한 표준 프로토콜 개발). 이 논문은 2025 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구 결과임(No.RS-2024-00438729, 익명화된 기밀실행을 이용한 전주기적 데이터 프라이버시 보호 기술 개발). 본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2025 년도 문화기술 연구개발사업으로 수행되었음 (과제명: On-Device AI 모델 저작권 보호 및 관리를 위한 글로벌 인재양성, 과제번호: RS-2025-02221620, 기여율: 20%)

## 참고문헌

[1] Intel, “Intel Trust Domain Extensions (IntelTDX) Module Base Architecture Specification,”January 2023.

[2] Kaplan, David, Jeremy Powell, and TomWoller. "AMD 메모리 encryption." White paper 13(2016).

[3] Alves, Tiago. "Trustzone: Integrated hardware and software security." Information Quarterly 3 (2004): 18-24.

[4] ARM, “Introducing Arm Confidential Compute Architecture,” https:// developer.arm.com/documentation/ den0125/, 2022.

[5] Pinto, Sandro, and Nuno Santos. "Demystifying arm trustzone: A comprehensive survey." ACM computing surveys (CSUR) 51.6 (2019): 1-36.

[6] ARM, “Arm CCA Security Model,” https://developer.Arm.com/ documentation/DEN0096, 2021.

[7] “ARM CoreLink TZC-400 TrustZone Address Space Controller Technical Reference Manual,” https://developer.arm. com/documentation/ddi0504/latest/, 2014.

[8] “PrimeCell Infrastructure AMBA 3 TrustZone Protection Controller (BP147),” https://developer.arm.com/documentation/ dto0015/latest/, 2023.

[9] Cerdeira, David, et al. "Sok: Understanding the prevailing security vulnerabilities in trustzone-assisted tee systems." 2020 IEEE Symposium on Security and Privacy (SP). IEEE, 2020.

[10] Halderman, J. Alex, et al. "Lest we remember: cold-boot attacks on encryption keys." *Communications of the ACM* 52.5 (2009): 91-98.

[11] “Arm fixed virtual platforms.” https://developer.arm.com/ tools-and-software/simulation-models/fixed-virtual-platforms, 2021.

[12] Zhang, Yiming, et al. "{SHELTER}: Extending arm {CCA} with isolation in user space." 32nd USENIX Security Symposium (USENIX Security 23). 2023.

[13] Li, Xupeng, et al. "Design and verification of the arm confidential compute architecture." 16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 22). 2022.

[14] Wang, Chenxu, et al. "Cage: Complementing arm cca with gpu extensions." Network and Distributed System Security (NDSS) Symposium. Vol. 2024. 2024.

[15] Liu, Shiqi, et al. "NanoZone: Scalable, Efficient, and Secure Memory Protection for Arm CCA." arXiv preprint arXiv:2506.07034 (2025).

[16] https://learn.Arm.com/learning-paths/servers-and-cloud-computing/rme-cca-basics/rme-cca-fvp/