Trusted RV: セキュアコプロセッサを有する64bit RISC-V TEEとその上のソフトウェア
Trusted RV: 64bit RISC-V TEE with Secure Coprocessor and software on them

須崎有康[1,2]

Kuniyasu Suzaki[1,2]

1)セキュアオープンアーキテクチャ・エッジ基盤技術研究組合

1) Technology Research Association of Secure IoT Edge application based on RISC-V Open architecture (TRASIO)

2) 産業技術総合研究所

2) National Institute of Advanced Industrial Science and Technology (AIST)

Self Introduction (Kuniyasu Suzaki,須崎有康)

- 2009年よりTrusted ComputingやTEE関連の研究を行う
  - TCG(Trusted Computing Group) Invited Expert from 2019
- TRASIO受託NEDOプロジェクト 「セキュアオープンアーキテクチャ基盤技術とそのＡ Ｉエッジ応用研究開発 FY2018-2020」でRISC-VベースのTEEの研究
  - 本講演はこちらの成果を中心に話します

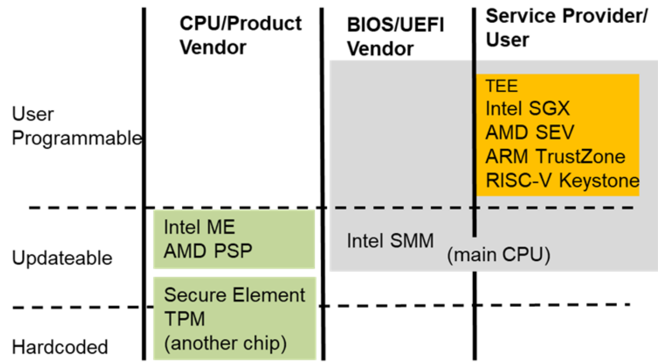
- **Reference 参考資料**
  - **Trusted Execution Environmentによるシステムの堅牢化, 情報処理20/06**
    - https://ci.nii.ac.jp/naid/40022255769
  - **Trusted Execution Environmentの実装とそれを支える技術,電子情報通信学会 基礎・境界ソサイエティ Fundamentals Review, 2020/10 (無償公開)**
    - https://www.jstage.jst.go.jp/article/essfr/14/2/14_107/_article/-char/ja/

Contents

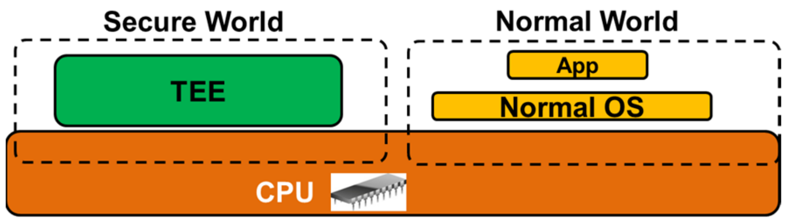
- What is TEE?
- 4 Issues of TEE (Trusted Execution Environment) on RISC-V
  1. Root of Trust
  2. Programming Environment
  3. TA(Trusted Application) Management
  4. Remote Attestation
- 4 Security Technologies offered by TRASIO
  - Hardware
    1. Trusted-RV Platform (64-bit RISC-V + 32-bit RISC-V Secure CoProcessor)
  - Software
    2. TEE Programming Environment (GlobalPlatform TEE Internal API)
    3. TA Management Framework: TEEP(Trusted Execution Environment Provisioning)
    4. Remote Attestation
- Future works and Conclusions

What is TEE (Trusted Execution Environment)?

- TEE is one of CPU’s execution environments isolated form OS
  - **Caution: TEE is not only-one isolated execution environment**
  - SMM, Intel ME, and TPM are also isolated from OS.
  - TEE is different from others because **it is programmable and opened for normal user.**



- TEE separates the execution environment into 2 worlds
  - Normal World (i.e., REE: Rich Execution Environment) for normal OS and apps
  - Secure World (i.e., TEE: Trusted Execution Environment) for critical apps.



- Available Hardware: ARM TrustZone, Intel SGX, AMD SEV, and RISC-V

RISC-V TEE

- RISC-V TEE implementations

Academia
- Sanctum [MIT,USENIX Sec'16]
- TIMBER-V [Graz University of Technology, NDSS'19]
- MI6 [MIT,MICRO'19]
- Keystone [UC Berkeley, EuroSys'20]
- HECTOR-V [Graz University of Technology, arXiv'21]
- CURE [Technische Universität Darmstadt, USENIX Sec'21]

Industry
- MultiZone [HexFive]

Keystone is an active open-source project.
This talk is based on Keystone.

Problem of TEE (Root of Trust) 1/4

- **TEE** is just an isolated execution environment and cannot be a *Root of Trust*.
  - **Root of Trust keeps keys and certificates and muse be Secure CoProcessor.**
  - ***Remote Attestation*** must be based on Root of Trust.
- Example of Root of Trust
  - Intel SGX has Intel ME(Management Engine). Intel Quark x86-based (32bit)
  - AMD SEV has PSP(Platform Security Processor). Arm Cortex-A5 (32bit)
  - Arm TrustZone needs an extra IP
    - CryptCell(Discretix -> Arm)
    - CryptoManager (Rambus)
    - Secure Element
    - Apple M2
  - RISC-V needs an extra IP
    - Rambus RISC-V CryptoManager
    - Silex Insight Secure Root of Trust
    - OpenTitan (Open Source)

Most of them, the detail is not open.
We cannot verify them and need to trust them.

Problem of TEE (Programing) 2/4

- Each TEE has each SDK for programing
  - Intel SGX
    - Intel SGX SDK
    - Open Enclave (Microsoft)
    - Asylo (Google)
  - AMD SEV
    - Asylo (Google)
    - Enarx (Redhat)
  - Arm Cortex-A TrustZone
    - Open Enclave (Microsoft)
    - GlobalPlatform (GP) TEE Internal API
  - RISC-V Keystone
    - Keystone SKD

No compatibility and
No portability
for different CPU architecture.

Problem of TEE (TA Management) 3/4

- Is a TA installed, updated, and deleted safely?
  - A TA is developed by a third party (e.g., video supplier, bank), but the supplier and client want to confirm the safety each other.
    - From the view of platform (TEE Edge device)
      - Is the TA trustable? Is the download server trustable?
    - From the view of TA
      - Is the platform genuine (no tempered)?

Management of TA must be safe.
In addition, the management must follow each CPU security procedure.

Problem of TEE (Remote Attestation) 4/4

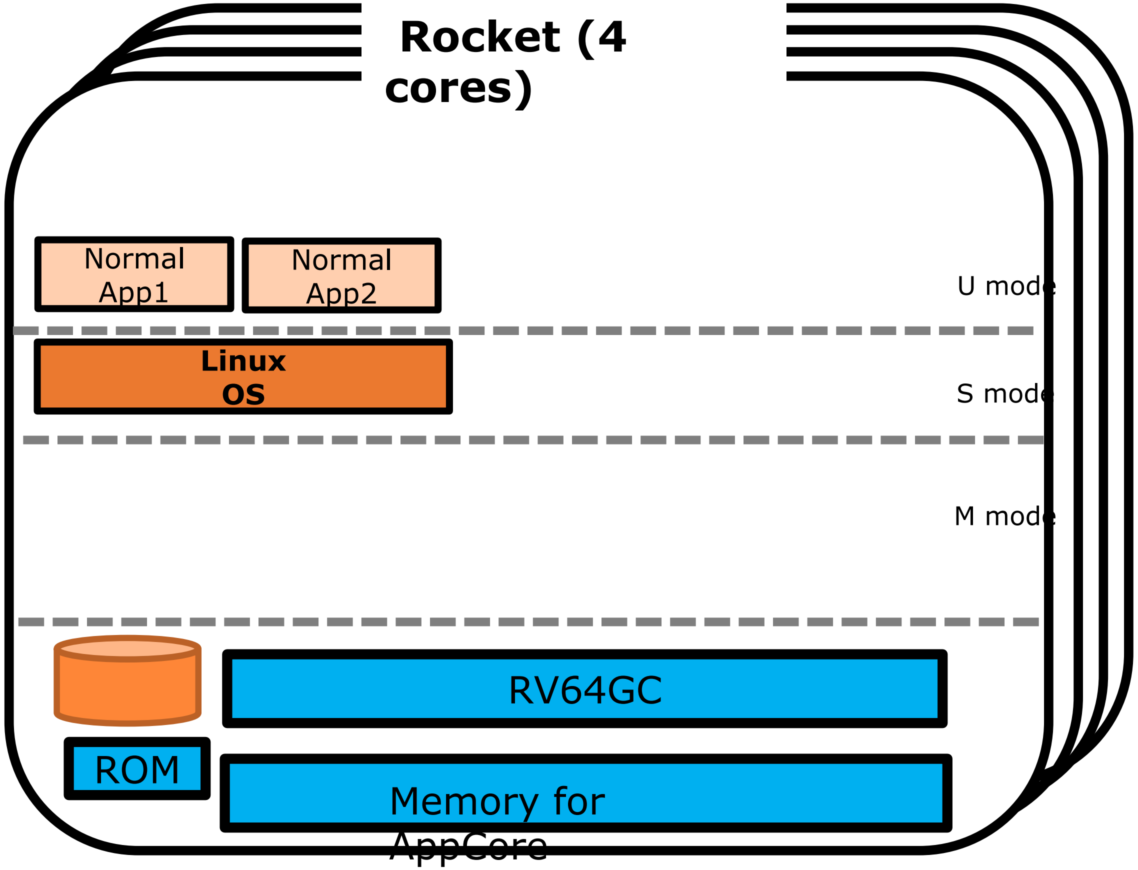
- Does a genuine TA run on a genuine platform (no tempered)?
  - Remote Attestation is a mechanism to certificate platform and TA.
    - Basement of install/update/delete (problem 3)

Device keys and certificates must be managed by ***Root of Trust***.

Security Technologies offered by TRASIO

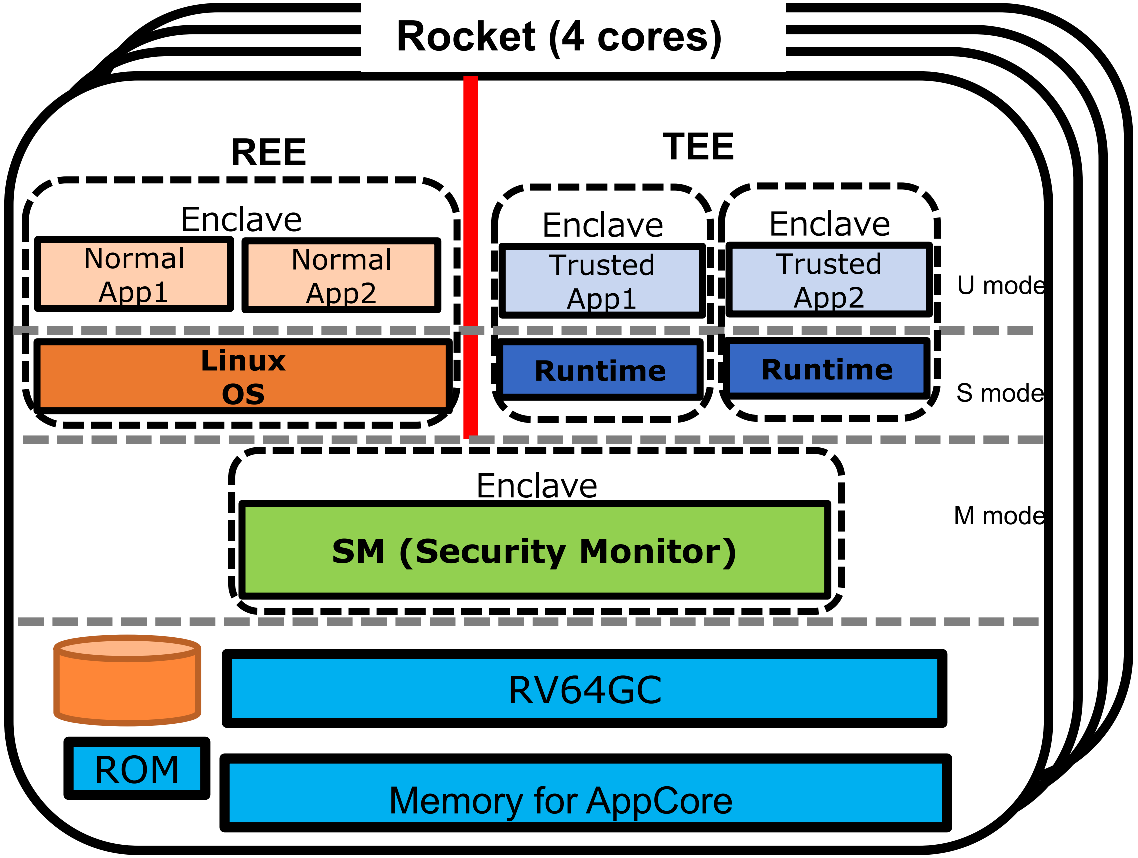
- Hardware
  1. Trusted-RV Platform (64-bit RISC-V + 32-bit RISC-V Secure CoProcessor)
- Software
  2. TEE's Programing Environment: GlobalPlatform TEE Internal API
  3. TA Management Framework: TEEP(Trusted Execution Environment Provisioning)
  4. Remote Attestation

Normal RISC-V



- Rocket cores are assumed.

Keystone enabled RISC-V

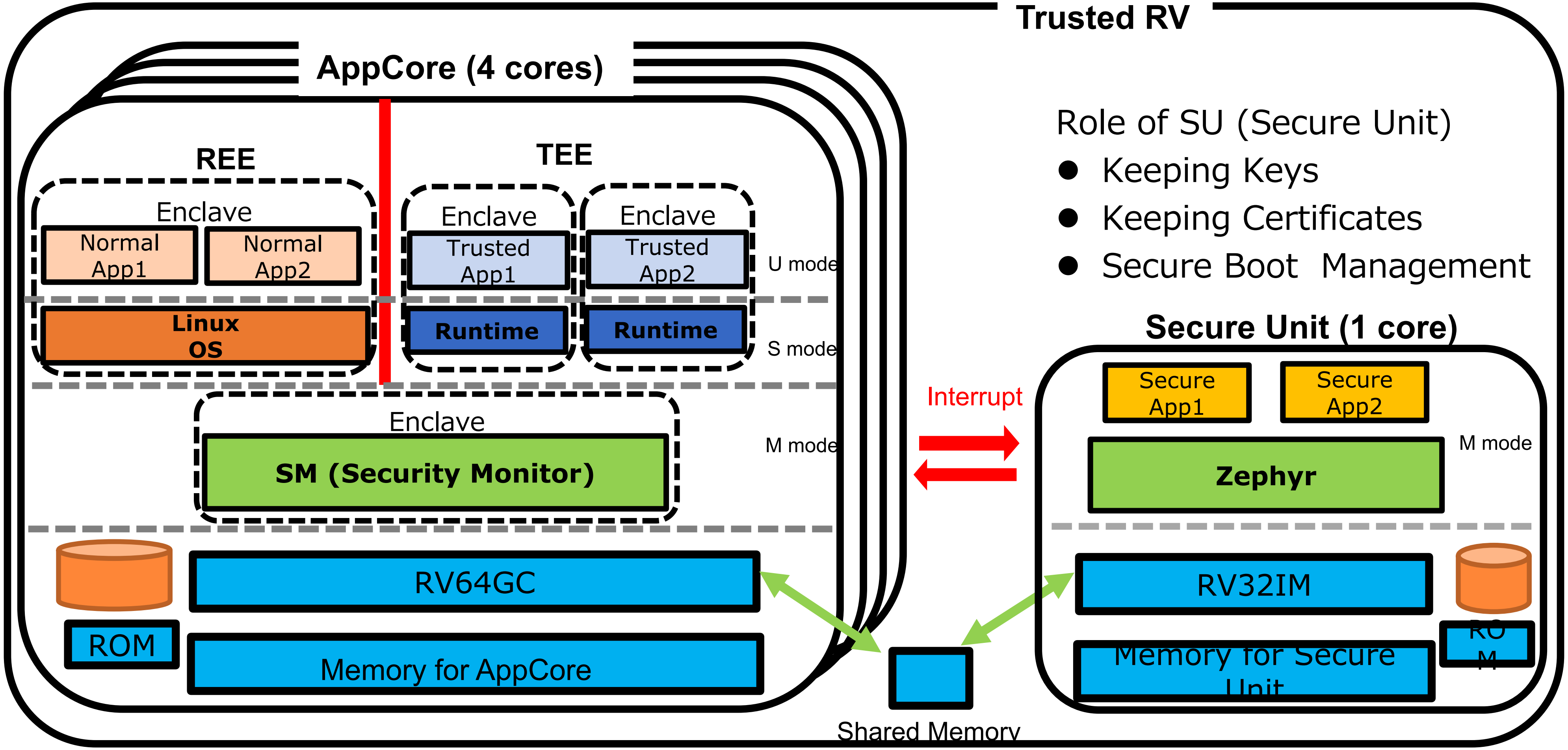


- PMP (Physical Memory Protection) isolates memory. Dotted line indicates an Enclave.
  - One Secure Monitor in M mode
  - One OS (Linux) in REE
  - Two Enclaves in TEE

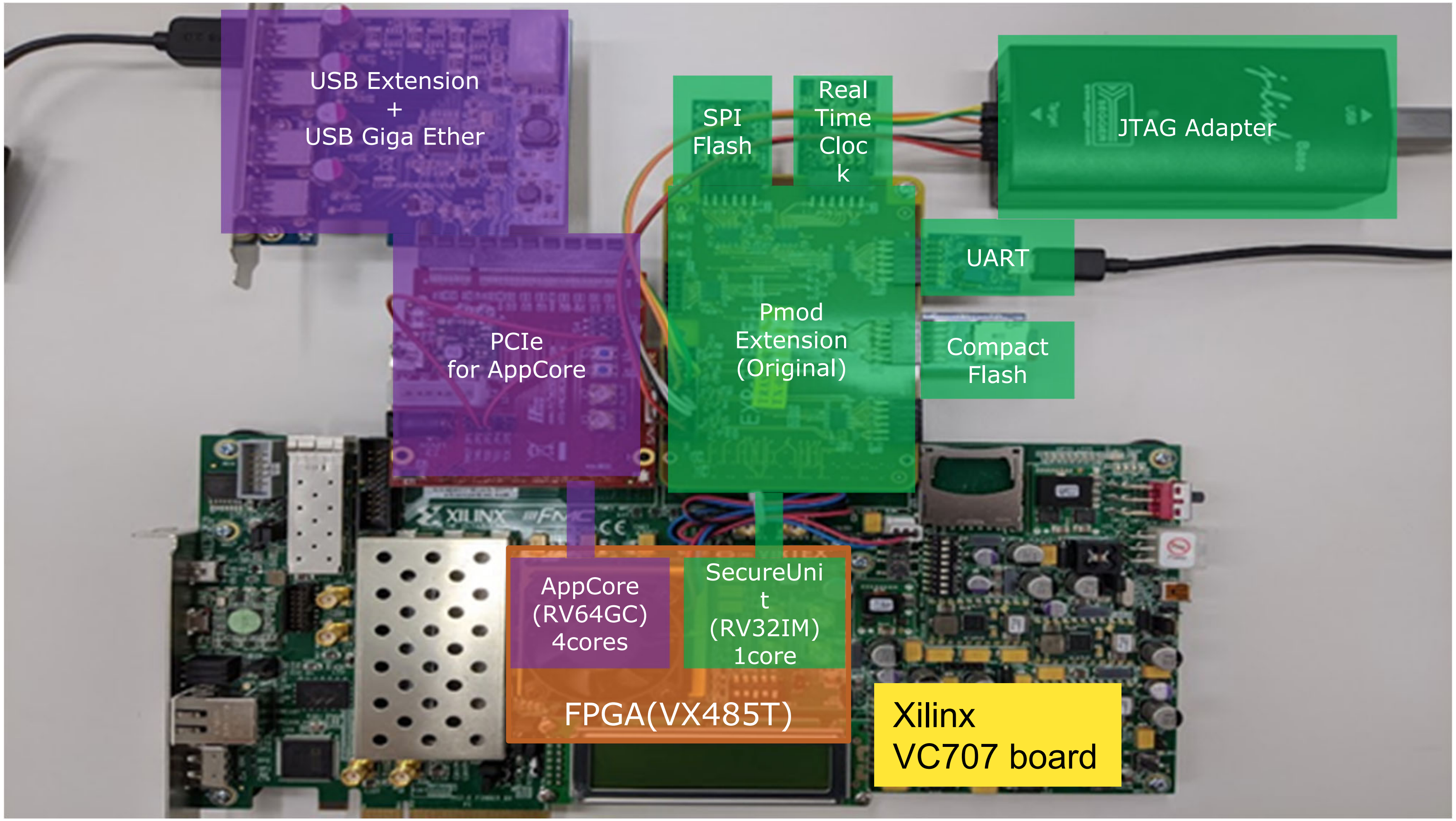
Keystone with Secure CoProcessor

Role of SU (Secure Unit)

- Keeping Keys
- Keeping Certificates
- Secure Boot Management

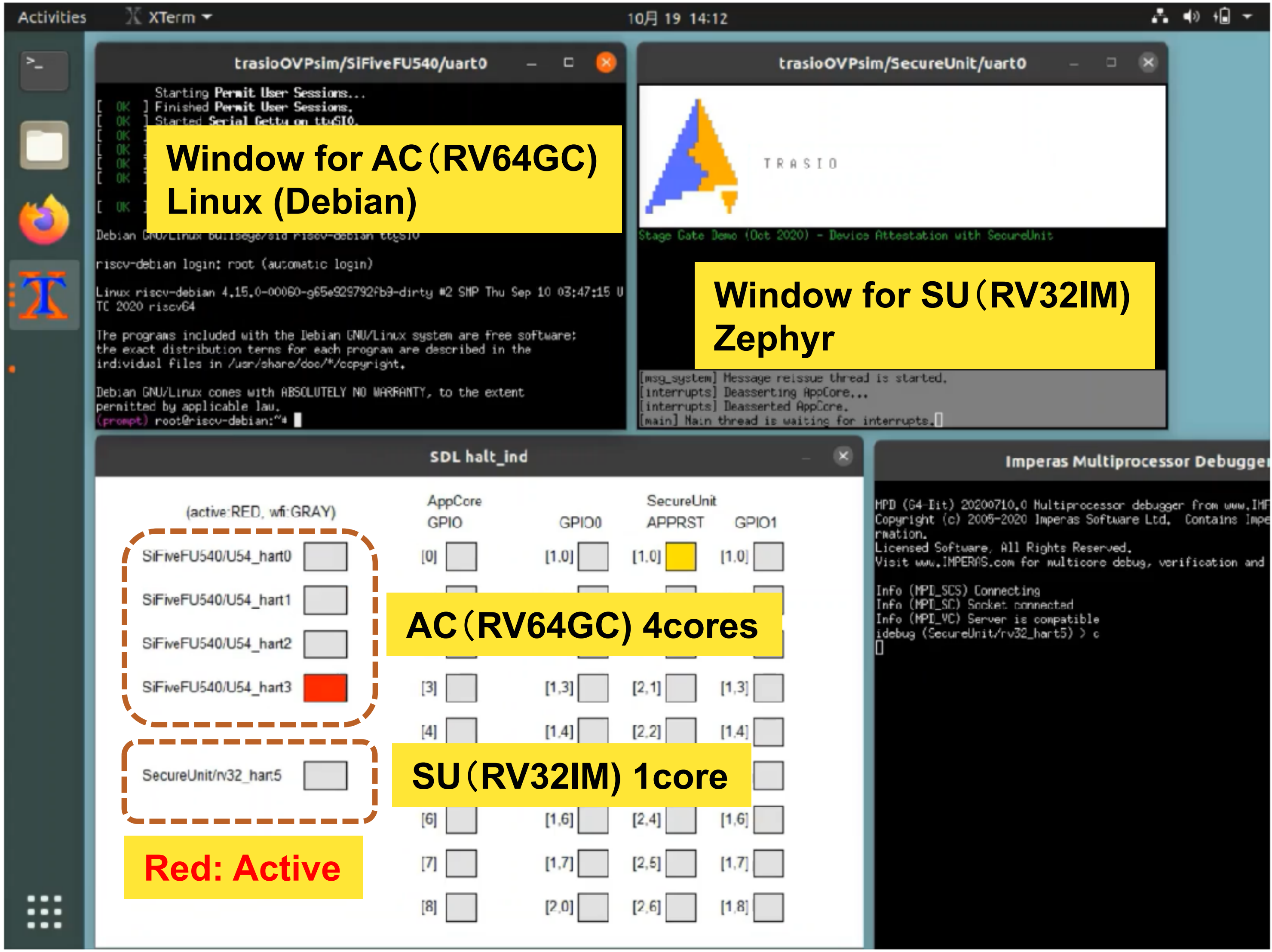


FPGA Implementation



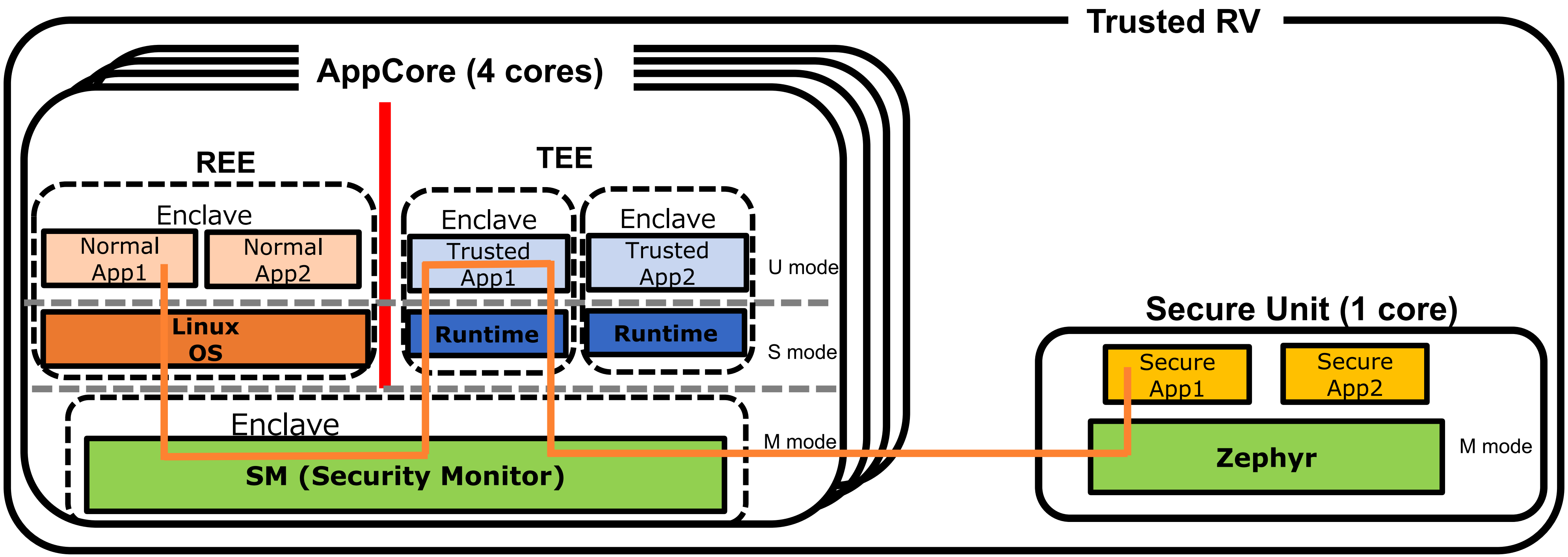
Simulator

- Based on Imperas RISC-V simulator
- Used for system software development



Software structure for Secure CoProcessor

- The communication is limited
  - (Normal App) -> (Secure Monitor) -> (Trusted App) -> (Secure Monitor) -> (Zephyr) -> (Secure App)
  - No direct access from Normal App / Linux is not allowed.
  - ***Trusted App and Secure App must not leak critical information***.
    - design depends on use case because TEE is powerful 64RV and SU is low power RV32.



TRASIO

AIST CPSEC Cyber Physical Security Research Center

Layer of TEE and Secure CoProcessor

2 layers

REE + TEE + Secure CoProcessor
- **The order is important**
  - **REE -> TEE -> Secure CoProcessor**

1 layer

REE + Secure CoProcessor
- No isolated environment
- Critical processing is not protected

REE + TEE
- No hardware tamper-proof
- Critical data is not protected

0 layer

REE only
- No isolated environment
- No hardware tamper-proof
- Critical data and processing are not protected

Comparison of RISC-V Secure CoProcessor

*this table is not complete.

	Google OpenTitan	Rambus RISC-V CryptoManager (RT-6*0, RT-7*0)	Silex Insight eSecure (BA470)	Trusted RV Secure Unit
Core	Ibex (RV32IMC/EMC) M/U Mode	Custom (RV32IMC) M/S/U Mode	Andes N22 (RV32IMAC/EMAC) M or M/U mode	Custom (RV32IM) M mode
OS	Tock OS	Zephyr	---	Zephyr
Comm to Main	SPI	GPIO/SPI	---	GPIO Shared Memory
Accelerator	AES,SHA	AES, SHA	AES, SHA	Not yet ***
Peripherals	Timer, RNG	Timer, RNG	RNG	Timer, RNG, Flash
Anti-tampering	Yes?	Yes	Yes	Not yet
Target	Key Management, Secure Boot, OTA	Key Management, Secure Boot, OTA, User App	Key Management, Secure Boot, OTA	Key Management, Secure Boot, OTA
Misc.	QEMU support	FIPS 140-2 Level 2	FIPS 140 2 level 3 PUF for Unique Key	**Design with TEE (Different part from OpenTitan)**

*** We have developed the accelerator for SHA-3 and Ed25519 for quick boot.
"Quick Boot of Trusted Execution Environment With Hardware Accelerator", IEEE Access 2020 https://ieeexplore.ieee.org/document/9064723

Security Technologies offered by TRASIO

- Hardware
  1. Trusted-RV Platform (64-bit RISC-V + 32-bit RISC-V Secure CoProcessor)
- Software
  2. TEE's Programing Environment: GlobalPlatform TEE Internal API
  3. TA Management Framework: TEEP(Trusted Execution Environment Provisioning)
  4. Remote Attestation

TEE's Programing: GlobalPlatform TEE Internal API

- GP TEE Internal API does not depend on CPU architecture and is used by many Smartphones
  - kinibi (Trustonic)
    - Kinibi runs on 1.7 billion devices [USENIX Sec20,PARTEMU]
  - QSEE (Qualcomm)
    - 60% Android phones uses SQEE as of 2019 [USENIX Sec20,PARTEMU]
  - OP-TEE(Linaro)
    - Open-source implementation.
- We have developed some applications with GP API on OP-TEE and want to port them to Intel SGX and RISC-V Keystone.

What we did for Keystone and SGX

- We designed the GP internal API library to be portable.
  - We utilize SDK to implement a library which offers new abstraction.
  - The library is ported to Intel SGX as well as RISC-V Keystone.

- **Implementation Challenge**
  - Some APIs depend on hardware.
    - We separate APIs into hardware dependent / independent.
  - Integrate GP TEE Internal APIs to Keystone SDK
    - Keystone SDK includes EDL (Enclave Definition Language) named “keedger”.
    - EDL creates the code for communication (request from TEE to REE) to check the pointer and boundary.

TRASIO

AIST CPSEC | Cyber Physical Security Research Center

The specification of GP TEE internal API

GLOBALPLATFORM®

THE STANDARD FOR SECURE DIGITAL SERVICES AND DEVICES

GlobalPlatform Technology

TEE Internal Core API Specification

Version 1.1.2.50 (Target v1.2)

Public Review

June 2018

Document Reference: GPD_SPE_010

Functions by Category

29 categories

131 functions

Cipher Suite

Table 6-1: Supported Cryptographic Algorithms[4]

Algorithm Type	Supported Algorithm
Digests	MD5 SHA-1 SHA-256 SHA-224 SHA-384 SHA-512 SM3-256
Symmetric ciphers	DES Triple-DES with double-length and triple-length keys AES SM4
Message Authentication Codes (MACs)	DES-MAC AES-MAC AES-CMAC HMAC with one of the supported digests
Authenticated Encryption (AE)	AES-CCM with support for Additional Authenticated Data (AAD) AES-GCM with support for Additional Authenticated Data (AAD)
Asymmetric Encryption Schemes	RSA PKCS1-V1.5 RSA OAEP
Asymmetric Signature Schemes	DSA RSA PKCS1-V1.5 RSA PSS
Key Exchange Algorithms	Diffie-Hellman

Table 6-2: Optional Cryptographic Algorithms

Algorithm Type	Algorithm Name	When Supported
Asymmetric Signature Schemes on generic curve types	ECDSA	Any of the curves in Table 6-14 for which "generic" is Y
Key Exchange Algorithms on generic curve types	ECDH	Any of the curves in Table 6-14 for which "generic" is Y
Asymmetric Signature on Edwards Curves	ED25519	Any Edwards curve is supported
Key Exchange Algorithms on Edwards Curves	X25519	Any Edwards curve is supported
Various asymmetric Elliptic Curve-based cryptographic schemes using the SM2 curve.	SM2	SM2 is supported
Various signature and HMAC schemes based on the SM3 hash function.	SM3	SM2 is supported (SM2 support implies support for SM3. See Table 4-14).
Various symmetric encryption-based schemes based on SM4 symmetric encryption	SM4	SM2 is supported (SM2 support implies support for SM4. See Table 4-14).

Separate GP TEE Internal API

- Hardware dependent
  - Random Generator, Time, Secure Storage, Transient Object(TEE_GenerateKey)
- Hardware independent　(Crypto)
  - Transient Object(exclude TEE_GenerateKey), Crypto Common, Authenticated Encryption, Symmetric/Asymmetric Cipher, Message Digest

Category	CPU (In)Dependent	Functions
Random Number	Dependent	TEE_GenerateRandom
Time	Dependent	TEE_GetREETime, TEE_GetSystemTime
Secure Storage	Dependent	TEE_CreatePersistentObject, TEE_OpenPersistentObject, TEE_ReadObjectData, TEE_WriteObjectData, TEE_CloseObject
Transient Object	Dependent Independent	TEE_GenerateKey, TEE_AllocateTransientObject, TEE_FreeTransientObject, TEE_InitRefAttribute, TEE_InitValueAttribute, TEE_SetOperationKey
Crypto Common	Independent	TEE_AllocateOperation, TEE_FreeOperation
Authenticated Encryption	Independent	TEE_AEInit, TEE_AEUpdateAAD, TEE_AEUpdate, TEE_AEEncryptFinal, TEE_AEDecryptFinal
Symmetric Cipher	Independent	TEE_CipherInit, TEE_CipherUpdate, TEE_CipherDoFinal
Asymmetric Cipher	Independent	TEE_AsymmetricSignDigest, TEE_AsymmetricVerifyDigest
Message Digest	Independent	TEE_DigestUpdate, TEE_DigestDoFinal

Reference

1. **Library Implementation and Performance Analysis of GlobalPlatform TEE Internal API for Intel SGX and RISC-V Keystone[TrustCom2020]** https://conferences.computer.org/trustcompub/pdfs/TrustCom2020-4sgfK5r538MStgrShyle8b/438000b200/438000b200.pdf
2. **Portable Implementation of GlobalPlatform API for TEE[RISC-V Global Forum 2020]** https://riscvglobalforum2020.sched.com/event/dO37

Security Technologies offered by TRASIO

- Hardware
  1. Trusted-RV Platform (64-bit RISC-V + 32-bit RISC-V Secure CoProcessor)
- Software
  2. TEE's Programing Environment: GlobalPlatform TEE Internal API
  3. TA Management Framework: TEEP(Trusted Execution Environment Provisioning)
  4. Remote Attestation

TEEP(Trusted Execution Environment Provisioning)

- TEEP is a protocol to manage TA(Trusted Application) to Install/Update/Delete.
  - Caution: Execution is out of scope because it depends CPU Architecture.

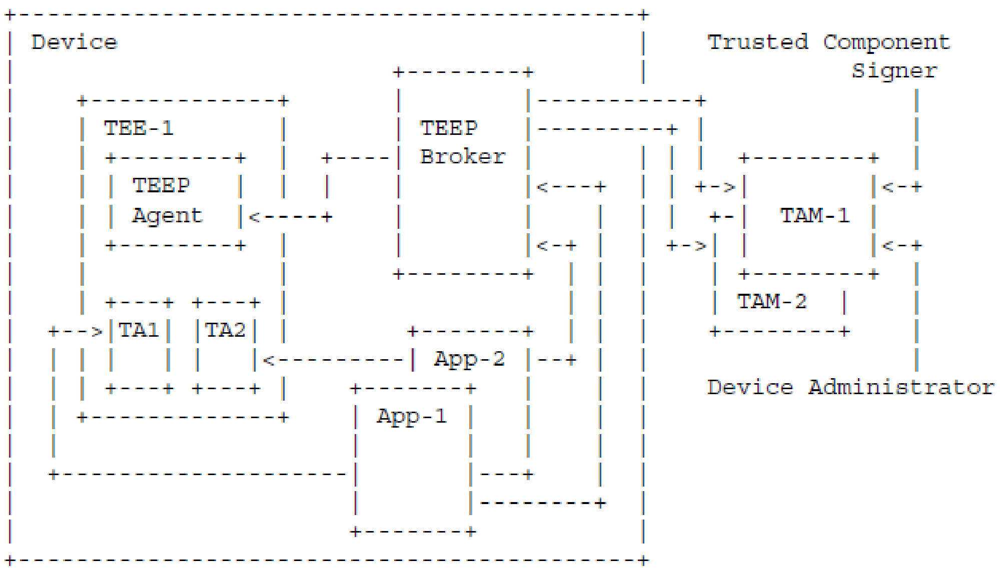


Figure 1: Notional Architecture of TEEP

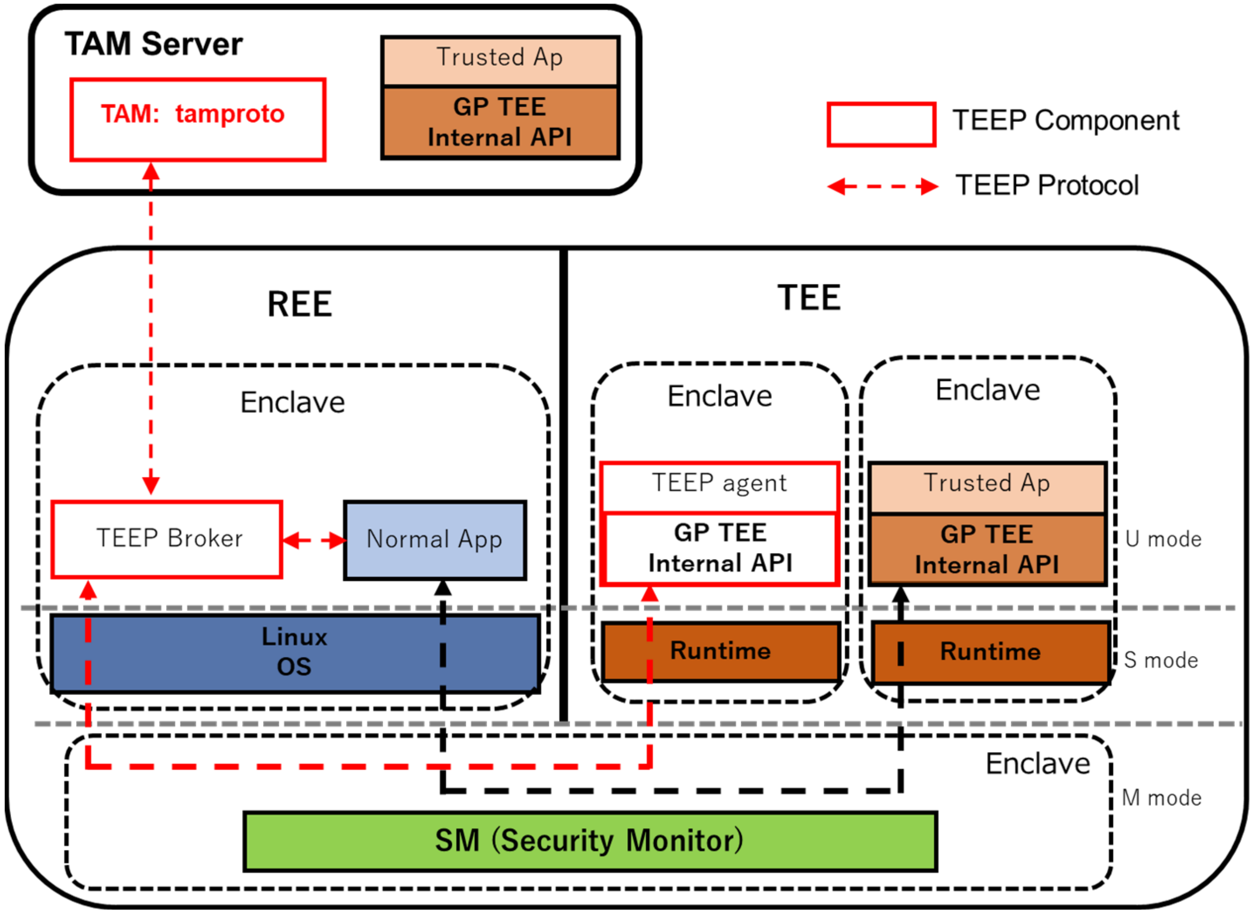
Purpose	Cardinality & Location of Private Key	Private Key Signs	Location of Trust Anchor Store
Authenticating TEE	1 per TEE	TEEP responses	TAM
Authenticating TAM	1 per TAM	TEEP requests	TEEP Agent
Code Signing	1 per Trusted Component Signer	TA binary	TEE

Figure 4: Signature Keys

Trusted Execution Environment Provisioning (TEEP) Architecture draft-ietf-teep-architecture-14 https://tools.ietf.org/pdf/draft-ietf-teep-architecture-14.pdf

TEE on RISC-V Keystone

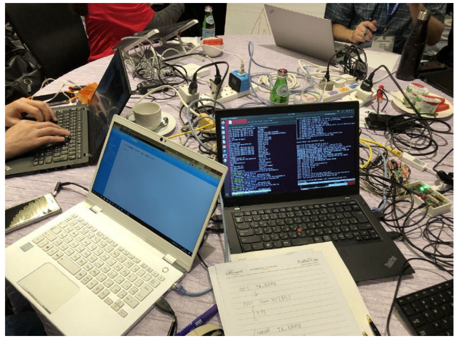
- The implementation uses the GP TEE internal API



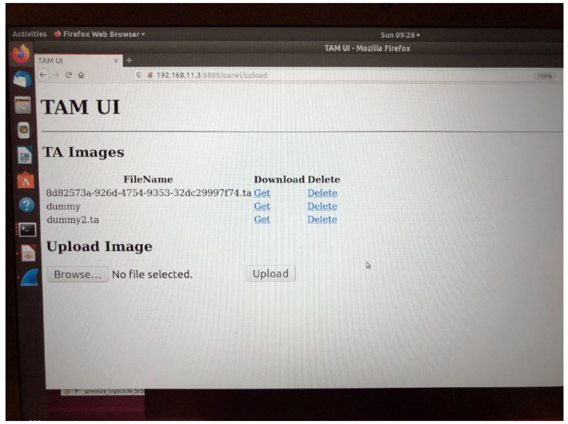
- *Reference*
  - *"TEEP (Trusted Execution Environment Provisioning) Implementation on RISC-V", FOSDEM2020*

IETF TEEP Hackathon

- IETF 104 (Prague, March/2019) TEEP Hackathon
  - Design of key management
- IETF 105 (Montreal, July,2019) TEEP Hackathon
  - TEEP on Arm Hikey
- IETF 106(Singapore, Nov/2019) TEEP Hackathon
  - Connect to Prototype TAM Sever
- IETF 108(Online, Nov/2020) TEEP Hackathon
  - Adapting revised TEEP and proposing Improving TEEP spec
- IETF 109(Online, Nov/2020) TEEP Hackathon
  - Adapting revised TEEP and proposing Improving TEEP spec
- IETF 110(Online, March2020) TEEP Hackathon
  - Applying SUIT manifest



Isobe(TRASIO/Secom) TAM UI



Online hackathon
On gather.town

Security Technologies offered by TRASIO

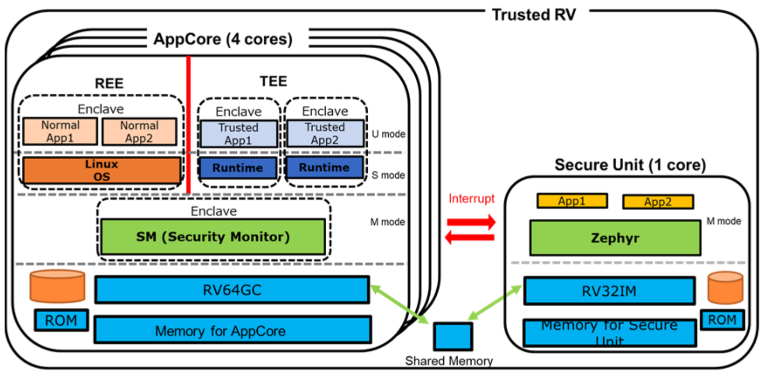
- Hardware
  1. Trusted-RV Platform (64-bit RISC-V + 32-bit RISC-V Secure CoProcessor)
- Software
  2. TEE's Programing Environment: GlobalPlatform TEE Internal API
  3. TA Management Framework: TEEP(Trusted Execution Environment Provisioning)
  4. Remote Attestation

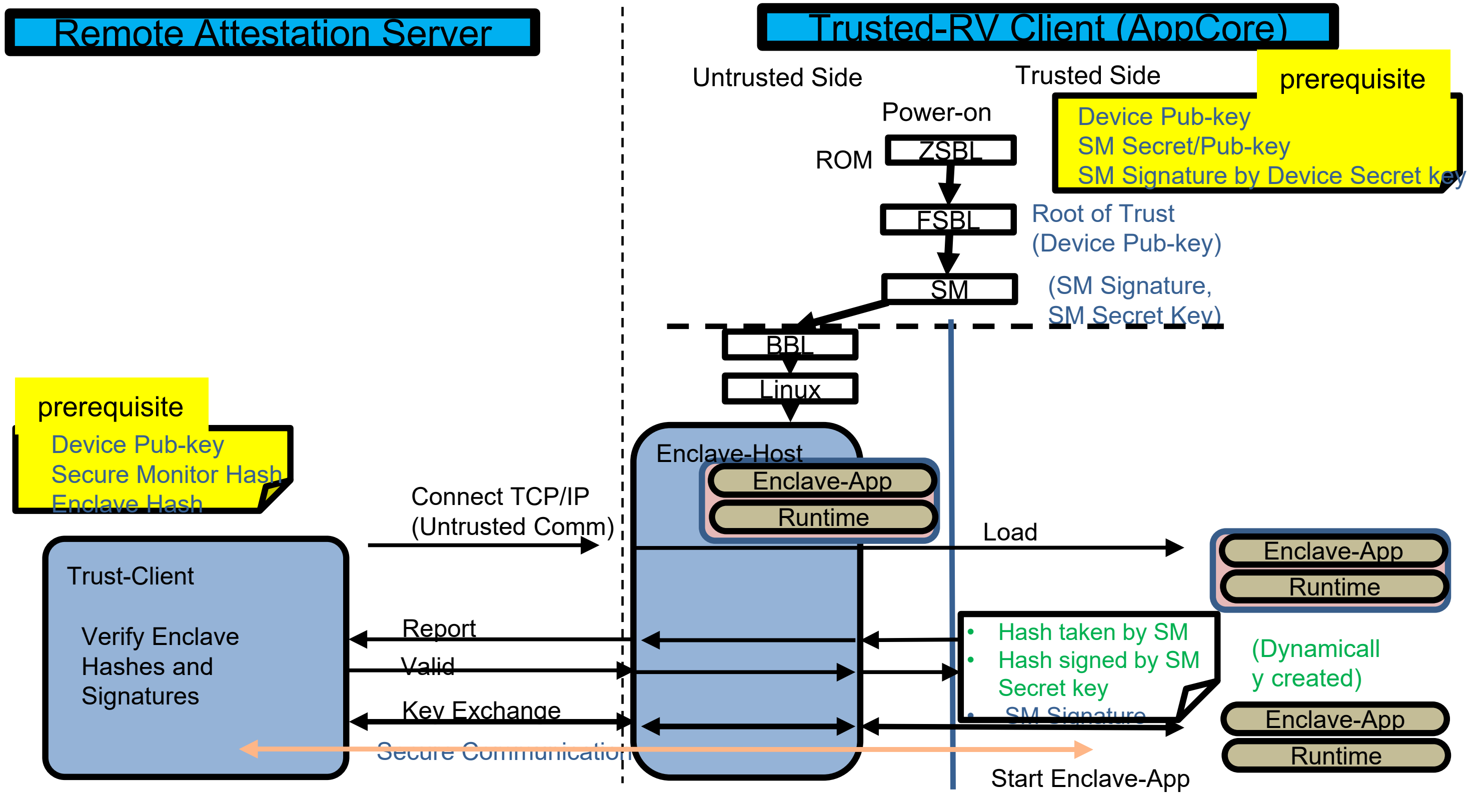
Remote Attestation

- Remote attestation offers
  - Platform authentication
  - Platform integrity
  - Binary integrity
- Remote attestation is achieved before the execution of TA and keeps the safe execution of TA on the TEE.
- Remote attestation assumes
  - On Edge (platform)
    - Keys or certificates protected by hardware, i.e., Root of Trust.
  - On Sever (verifier)
    - Data base for hash of TA, Device Pub-Key

Customized Remote Attestation

- We reuse the Keystone's Remote Attestation.
  - Trusted RV keeps a device key at secure unit.





Future Work

- We have developed the infrastructure of RISC-V TEE hardware/software.
- Nest step is creation of PoC(Proof of Concept) for real usage.
  - Server
    - Code and Data hiding for Machine Learning
  - Edge
    - Smart city

Conclusions

- Current TEE has some issues and mitigated by Security Technologies offered by TRASIO
  - Hardware
    1. Trusted-RV Platform (64-bit RISC-V + 32-bit RISC-V Secure CoProcessor)
  - Software
    2. TEE's Programing Environment: GlobalPlatform TEE Internal API
    3. TA Management Framework: TEEP(Trusted Execution Environment Provisioning)
    4. Remote Attestation

Acknowledgement: This presentation is based on results obtained from a project (JPNP16007) commissioned by the New Energy and Industrial Technology Development Organization (NEDO).