



AP-TEE (aka Confidential VM Extension - CoVE)

Chair: Ravi Sahita, Vice-chair: Guernsey Hunt
AP-TEE TG, Assignee: Security HC

Feb 23rd, @ CCC TAC

Background: Confidential Computing

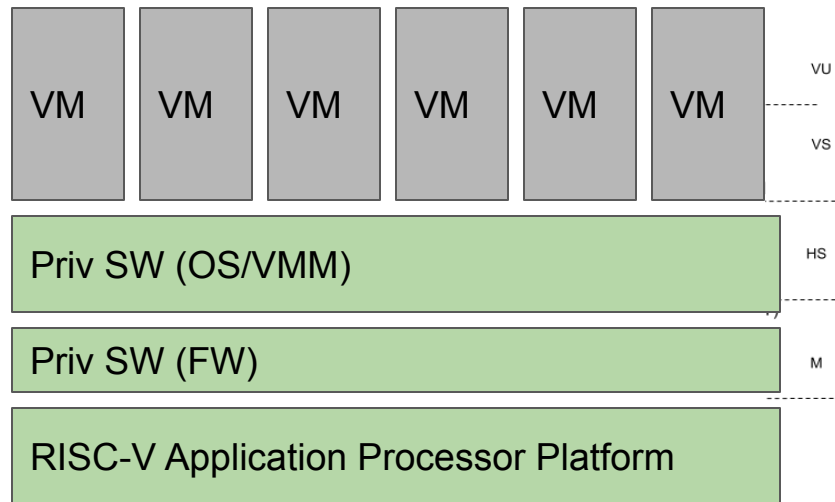
*Confidential Computing is the protection of data in use by performing computation in a hardware-based, attested Trusted Execution Environment**

This definition is independent of topological location, which processor does it, and whether encryption or some other isolation technique is used.

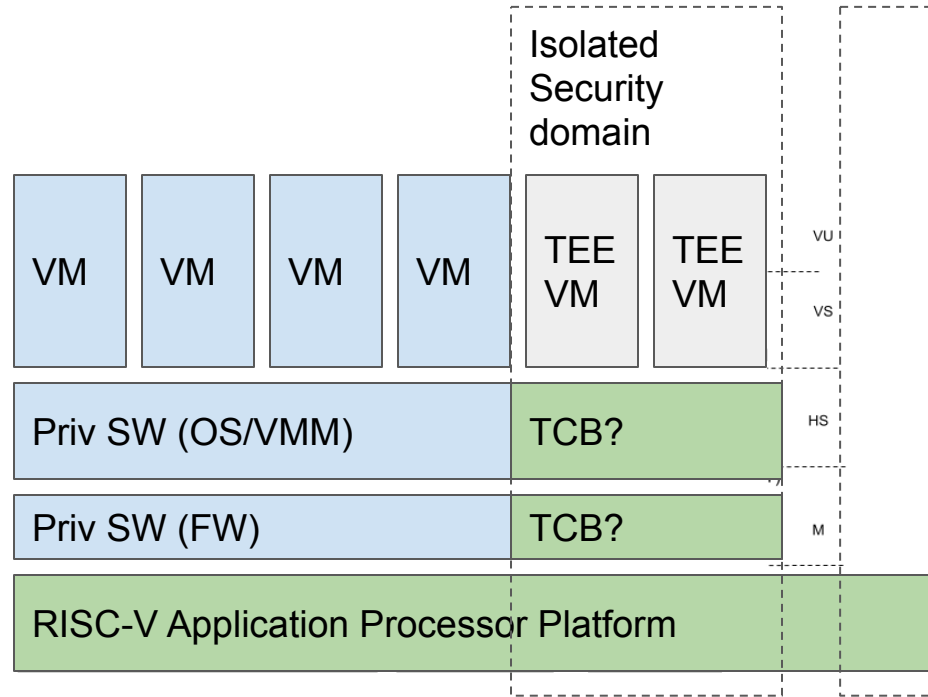
The protection of data in use is against a well-defined adversary.

The Trusted Execution Environment (TEE) provides - Data confidentiality, Code and Data integrity, Attestation and TCB Recovery

RISC-V Privilege Levels



Introduce Multiple Isolation Domains

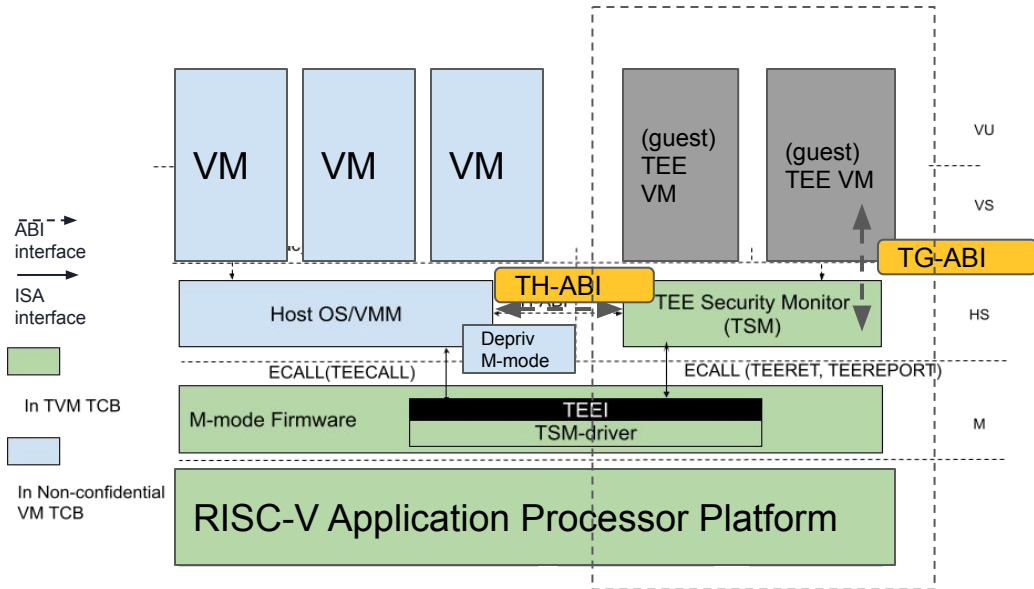


AP-TEE CoVE - Charter /1

AP-TEE CoVE interfaces enable Confidential VMs for application processor platforms using the ratified RISC-V ISA with H-extension. (ABI definition to be forward looking to incorporate future ISA security, performance improvements)

Goal - Create a reference architecture for confidential computing on RISC-V Application Processor platforms, specifying:

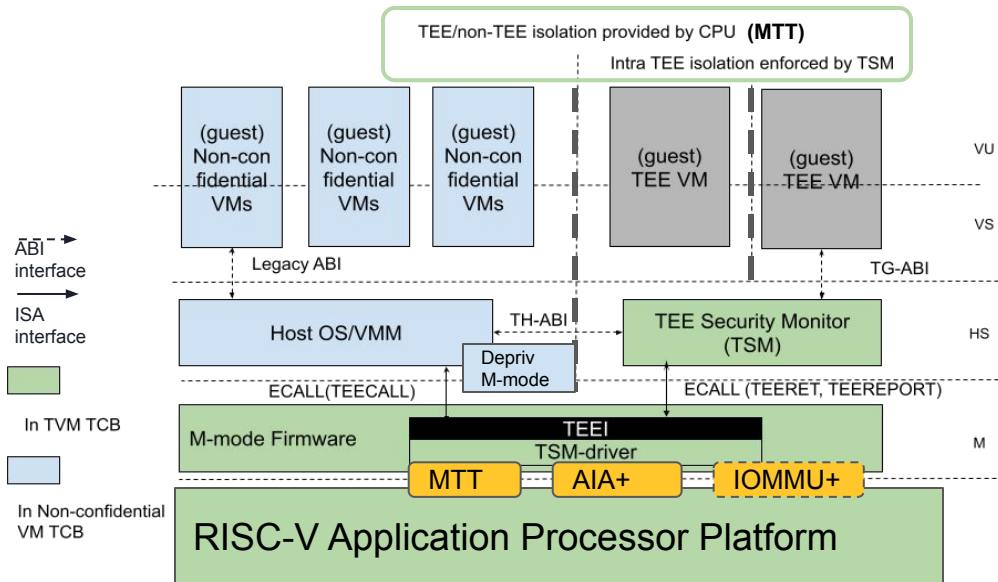
- **Non-ISA interfaces** between TCB and non-TCB components -- normative
<https://github.com/riscv-non-isa/riscv-ap-tee/blob/main/specification/riscv-apteree-spec.pdf>



AP-TEE CoVE - Charter /2

Goal - Create a reference architecture for confidential computing on RISC-V Application Processor platforms, specifying:

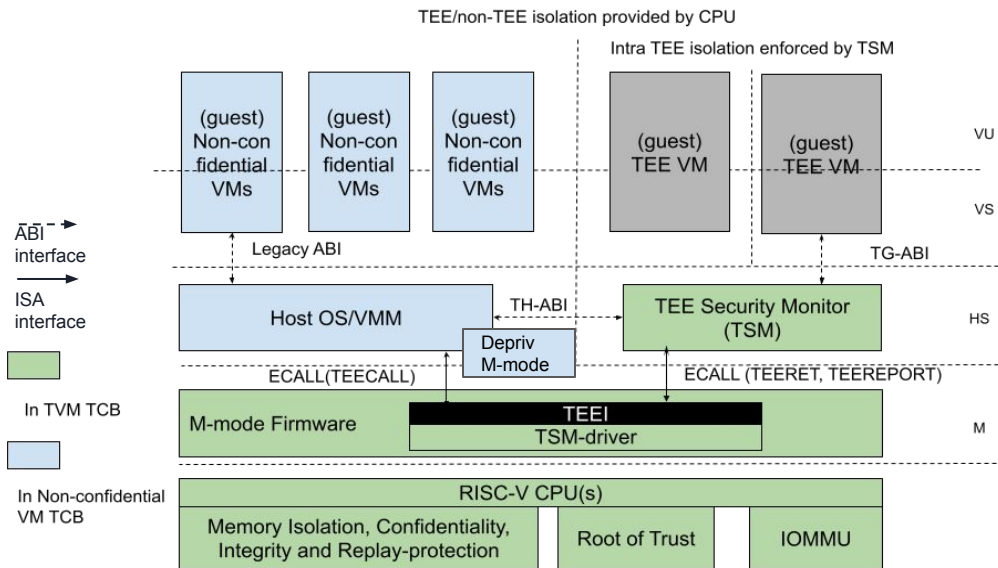
- **Non-ISA interfaces** between TCB and non-TCB components -- normative
- **ISA extension(s)** - identified ISA gaps
-WIP in TG (*to be worked on in separate TG/FT with priv IC*)
 - Confidential memory PMA (MTT)
 - Secure Interrupts using AIA
 - M-mode isolation for TSM-driver



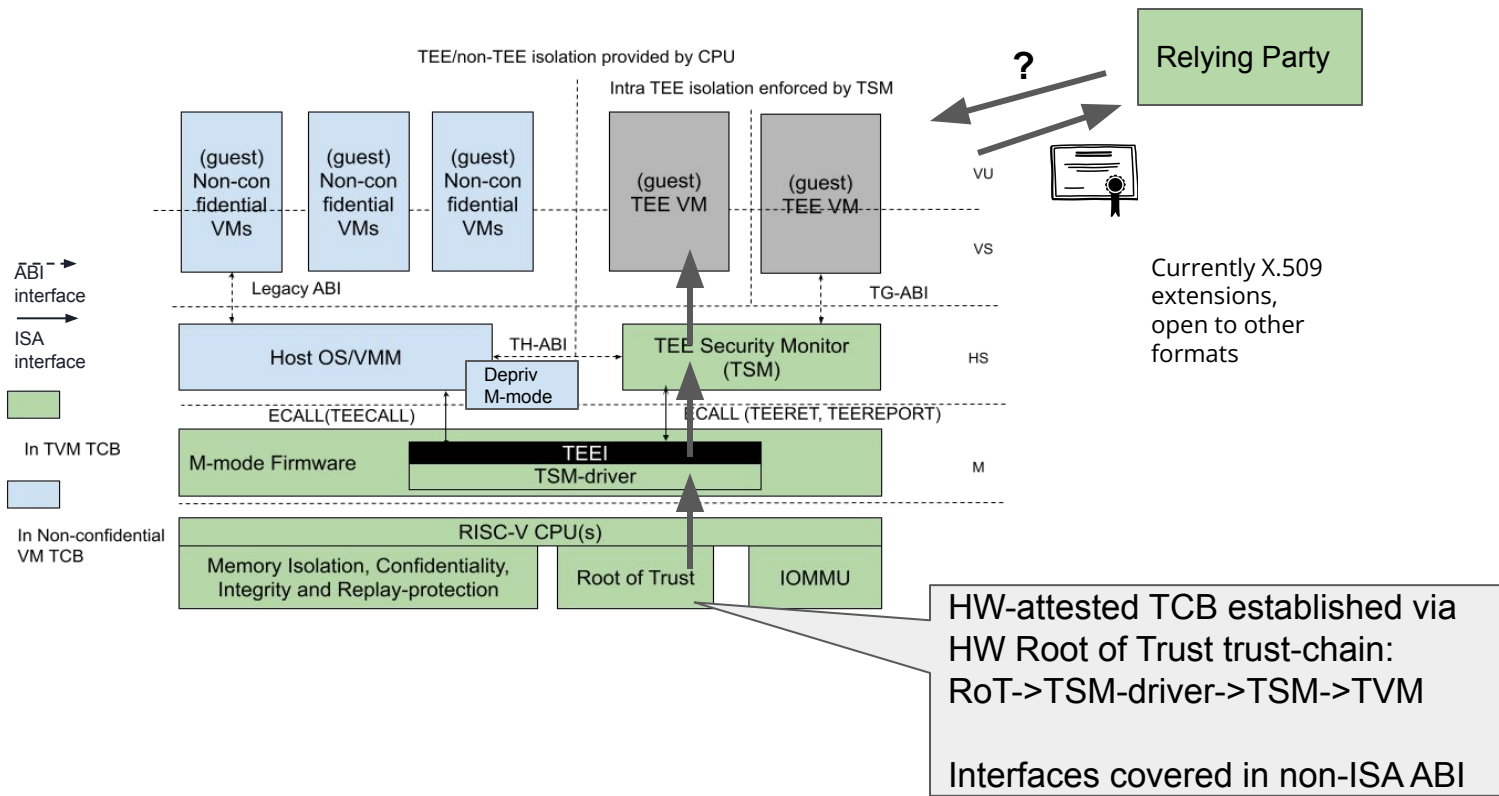
AP-TEE CoVE - Charter /3

Goal - Create a reference architecture for confidential computing on RISC-V Application Processor platforms, specifying:

- **Non-ISA interfaces** between TCB and non-TCB components -- normative
- **ISA extension(s)** - identified ISA gaps
-WIP in TG (*to be worked on in separate TG/FT with priv IC*)
 - Confidential memory PMA (MTT)
 - Secure Interrupts using AIA
 - M-mode isolation for TSM-driver
- **Recommendations** for Platform and SoC requirements -- informative
 - RVI Security Model spec.
 - Security Arch Analysis



AP-TEE CoVE Attestation



Non-ISA Interface Design Considerations

- **ABI intrinsics must meet security goals to isolate Confidential VMs from host adversaries**
 - Memory Confidentiality & Isolation
 - TVM HW state isolation & execution
 - Msmt. and Attestation
 - Secure Interrupt Mgmt using AIA
 - Debug & Performance monitoring
- **Document SOC infra. requirements for confidential computing**
- **Out of scope for version 1 of ABI**
 - Direct-IO
 - Live Migration
 - Sealing

Functional goals

- Enable VM, app, container, & other SW deployment models while avoiding application software refactoring
- Leverage standards for attestation e.g. RATS, SPDM, DICE
- Provide line of sight to future operational and performance features

Proof-of-concept and RISC-V Tests

Proof-of Concept

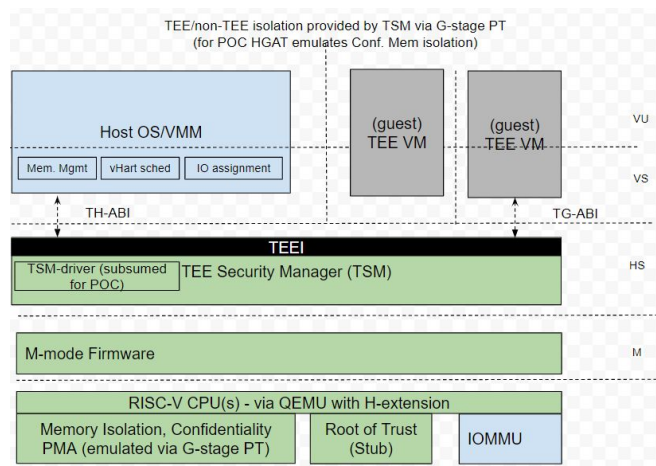
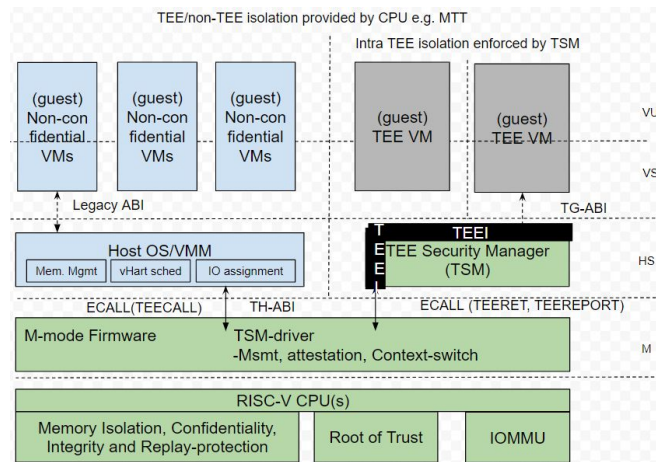
- Exercise AP-TEE CoVE SBI extension from the untrusted OS/VMM host to create Confidential TVMs. POC will focus on SMP TVM guests with para-virtualized IO (via shared memory). Demonstrate end-to-end TVM attestation flows.

Software Support

- TEE Security Manager
 - POC subsumes TSM-driver context switch flows
 - Conf. PMA emulated via guest stage PT
- Linux KVM (host)
 - Invokes TEEH aka COVEH SBI extension
- Linux TVM (guest)
 - Invokes TEEG aka COVEG SBI extension

Tests (Software) in Linux CoVE

- Compute-bound workload in TVM - kernel compile
- Memory intensive workload - stress-ng
- Paravirt IO intensive workload - iperf



Ref.
arch

POC

Key Milestones to freeze

Ratification Plan: [\[LINK\]](#)

1. Plan milestone: Acceptance Criteria checklist requirements

Acceptance Criteria status checklist link:	link
Planned Plan Approval Date:	1/31/2023
Actual Plan Approval Date:	[MM/DD/YYYY]
Status:	Needs sign-off
(Planned, Needs sign-off, Approved)	

2. Architecture Review

Planned Architecture Review Start Date:	1/31/2023
Actual Architecture Review Start Date:	[MM/DD/YYYY]
Planned Architecture Review Complete Date:	3/15/2023
Actual Architecture Review Complete Date:	[MM/DD/YYYY]
Status:	Planned
(Planned, Requested, Approved)	

3. Freeze: Pass non-ISA Acceptance Criteria requirements, including sign-offs

Planned Freeze Approval Date:	7/14/2023
Actual Freeze Approval Date:	[MM/DD/YYYY]
Status:	Planned
(Planned, Needs sign-off, Approved)	

Dev,
RFCs

4. Specification Public review

Reviewers: (Public review on isa-dev, Public review other, Technical Chairs, or ...)	Public review on isa-dev, Public review on OpenSBI mailing list, Public review on Linux RISC-V mailing list, and Public review on KVM RISC-V mailing list
Review rationale if not Public review on isa-dev: The Linux KVM RISC-V project, KVMtool and TSM (TEE Security Manager) implementers will have support for the interfaces defined by the RISC-V AP-TEE Interface specification so developer communities of these projects should be included in public review.	
Duration (Days)	45
Rational if duration is not 45 days: Not applicable	
Planned Review Start Date:	7/15/2023
Actual Review Start Date:	[MM/DD/YYYY]
Status:	Planned
(Planned, In progress, Complete)	

5. Ratification-ready: Pass non-ISA Acceptance Criteria requirements, including sign offs

Planned DoD Sign-off Date:	9/15/2023
Actual DoD Sign-off Date:	[MM/DD/YYYY]
Status:	Planned
(Planned, Sign-off requested, Approved)	

6. TSC Approval

Planned TSC Approval Date:	9/30/2023
Actual TSC Approval Date:	[MM/DD/YYYY]
Status:	Planned
(Planned, Approval requested, Approved)	

7. Board ratification

Planned BOD Ratification Date:	10/15/2023
Actual BOD Ratification Date:	[MM/DD/YYYY]
Status:	Planned
(N/A, Planned, Ratification requested, Ratified)	
Justification if no Board ratification: [EXPLANATION IF REQUIRED]	

Freeze Checklist

Status Checklist [[LINK](#)]

S
p
e
c

P
O
C

Freeze Checklist			
Name	Task Description	Status	Notes
	<p>Specification - The RISC-V AP-TEE specification defines the reference architecture, programming interfaces to support a scalable confidential VM architecture for RISC-V application processor platforms. ISA extensions required for supporting data-center deployment models of AP-TEE will be documented to be pursued separately with IC.</p> <p>The RISC-V AP-TEE programming interface (primary deliverable) defines an SBI extension expected to be implemented by the AP-TEE platform TCB (Trusted Computing Base), and invoked by the untrusted OS/VMM/hosting software, so this ratification plan covers the SBI extension (i.e. non-ISA parts) of the RISC-V AP-TEE specification. This interface will be designed to be extensible to be able to use any ISA extensions for AP-TEE.</p>		
Document Complete		In process	Repo: https://github.com/riscv/riscv-ap-tee PDF: https://github.com/riscv-non-isa/riscv-ap-tee/tree/main/spec
Proof of Concept (Code)	TSM (TEE Security Manager) - POC subsumes TSM-driver	In process	TSM is a new component. https://github.com/rivosinc/salus
Proof of Concept (Code)	Linux VMM (KVM) RISC-V host interface	In process	The KVM RISC-V APTEE patches are pending.
Proof of Concept (Code)	Linux RISC-V guest interface	In process	The Guest Linux RISC-V APTEE patches are pending.
Proof of Concept	KVMtool	In process	The KVMtool will be extended to support RISC-V AP-TEE Trusted Virtual Machine (TVM). These patches are pending.
Tests	End-to-end tests: TEE Security Manager (TSM) + Linux KVM RISC-V host + RISC-V Linux Guest as a compatibility test	In process	Test platforms to host confidential TVM workloads will be built using Qemu (RV64 with H-extension) with a Linux RISC-V KVM as host VMM and Linux RISC-V Guests as Confidential/TEE VM for ABI tests. TSM is a new RV64 HS component that exports the AP-TEE ABI to the KVM host, and the TG-ABI to the Linux TVM (confidential) guest.
Arch Review	Email tech-arch-review@lists.riscv.org for review. Policy in development.		
RISC-V Spec Policies	Abide by policies: encumbered information, friendly terminology, anonymous contributor (completed by RISC-V staff)		
Committee Chair Signoffs	OpaVote for all Committee Chairs (completed by RISC-V staff)		
CTO Signoff	Final check by CTO		

Summary

- TG Charter : [GitHub CHARTER [LINK](#)]
- Draft specification : [Github Repo [LINK](#)]
- Ratification Plan : [Ratification Plan [LINK](#)]
- Status checklist : [Status Checklist [LINK](#)]

Current status:

- **Development - 0.1 Spec draft published for TG review - [LINK](#)**
- **Open source TEE Security Manager (Salus) - [LINK](#)**
- **Linux/KVM change RFCs - are WIP**
 - **Under the name RISC-V COV* SBI Extensions**

**Goal is to freeze the non-ISA ABI extension by Q3'23 and ratify by Q4'23.
ISA FT proposals will run in parallel w/ Priv IC.**





Call to Action

- *Confidential computing is a key security capability for RISC-V platforms for scalable multi-tenant data-in-use protection.*
- **Review and provide feedback (via issues/PRs) on APTEE CoVE ABI specification**
 - See details of proposed interfaces in the AP-TEE draft asciidoc [specification](#)
 - A pdf version is [here](#) and an overview in these [slides](#)
- **Join POC efforts**
 - TEE Security Manager ([TSM](#)) for RISC-V implementing TH/TG-ABI
 - Extend RISC-V-KVM to interface with proposed TH/TG-ABI
 - joint task in AP-TEE TG, Hypervisor SIG and Linux/KVM projects
- **Develop common test cases to evaluate compatibility** for Linux/KVM TVM guests across different architectures and scenarios
 - Add RISC-V support to the [kvm-unit-tests](#)

BACKUP



AP-TEE TG: Interface spec status

		Area	Function	Resources
Specs 		AP-TEE TH-ABI	SBI Extension Interface implemented by the TSM via ECALL for use by OS/VMM to manage TVMs	APTEE TG WG members
		AP-TEE TG-ABI	SBI Extension Interface implemented by the TSM via ECALL for use by TVM guest workloads	
POCs 		TEE Security Manager (TSM)	TSM is a RISC-V 64 bit SW module that uses RISC-V H-extension and implements TH and TG-ABI. It is in the TCB for all TVM workloads (Expected to be HW-vendor signed and may be HW-operator signed)	Rivos contributes to start collab.
		TSM-driver	TCB component) to support TSM initialization and isolation, TEECALL, TEERET implementation. In TCB for all TVM workloads (Expected to be HW-vendor signed and may be HW-operator signed) - Collab with OpenSBI, Qemu as required.	Collaborating on these existing projects from Software HC (Priv sw)
		Linux, KVM (Host OS/VMM)	<i>Untrusted</i> (enlightened) host OS/VMM that manage resources for TVM-based confidential workloads [TSM enforces security properties] - Collab with Hypervisor SIG	
		Linux (TVM Guest OS), Guest Firmware	Enlightened guest OS/runtime (in TCB of TVM workload) - Collab with Priv SW HC	

AP-TEE TG: Platform & ISA Reqs.

Area	Function	Resources
CPU	Evaluate AP-TEE mode qualifier, Sparse (page-based) confidential memory PMA, access-control	TG members
IOMMU	AP-TEE mode qualifier; Sparse (page-based) confidential memory PMA, access-control, fabric i/f	w/ IOMMU TG
TLB, Caches	AP-TEE mode qualifier and other micro-architectural structures	TG members to document requirements into the Security Model and SOC Infra HC
Interconnect, Fabric	Platform-specific cryptographic memory isolation and mode qualifier	
Memory	Platform-specific cryptographic memory isolation and mode qualifier	
HW Root-of-trust	Platform-specific subsystem to support HW Attestation, Sealing interfaces	
Devices	Device-specific subsystem to support Device attestation, link security	
QoS, RAS, DC	Platform-specific, Domain-specific	



AP-TEE Security Arch for CC and Implementers Guide covers recommendations on:

- Mapping of mitigations to threat model
- Recommendations for crypto modes
- Attestation protocols, formats



Threats — Terminology - TVM: TEE VM (a confidential workload example); TSM: TEE Security Monitor (a TCB element enforcing the confidentiality of TVMs)

T1: Loss of confidentiality of TVM and TSM memory via in-scope adversaries that may **read TSM/TVM memory via CPU accesses**

T2: Tamper/content-injection to TVM and TSM memory from in-scope adversaries that may **modify TSM/TVM memory via CPU side accesses**

T3: Tamper of TVM/TSM memory from in-scope adversaries via **software-induced row-hammer attacks on memory**

T4: Malicious injection of content into TSM/TVM execution context using **physical memory aliasing attacks via system firmware adversary**

T5: Information leakage of workload data **via read of CPU registers, CSRs** via in-scope adversaries

T6: Incorrect execution of workload via **runtime modification of CPU registers, CSRs, mode switches** via in-scope adversaries

T7: Invalid code execution or data injection/replacement via **second-level paging remap attacks** via system software adversary

T8: **Malicious asynchronous interrupt injection** or denied leading to information leakage or incorrect execution of the TEE

T9: **Malicious hardware mtime register manipulation** or manipulation of time read from the time CSR causing invalid execution of TVM to lead to information loss

T10: Loss of Confidentiality **via DMA access from devices under adversary control** e.g. via manipulation of IOMMU programming

T11: Loss of Confidentiality **via DMA access from devices assigned to a TVM**. Devices bound to a TVM must enforce similar properties as the TEE on the SOC.

T12: Content injection, exfiltration or replay (within and across TEE memory) **via hardware approaches, including via exposed interface/links** to other CPU sockets, memory and/or devices assigned to a TVM

T13: **Downgrading TEE TCB elements** (example M-mode firmware, TSM) to older versions or loading Invalid TEE TCB elements on the platform to enable confidentiality, integrity attacks

T14: **Leveraging transient execution side-channel attacks** to leak confidential data e.g. via shared caches, branch predictor poisoning, page-faults.

T15: **Leveraging architectural side-channel attacks** due to shared cache and other shared resources e.g. via prime/probe, flush/reload approaches

T16: **Malicious access to ciphertext with known plaintext** to launch a dictionary attack on a TVM to extract confidential data.

T17: **Tamper of TVM state during migration** of a TEE workload from one platform to another.

T18: **Forging attestation reports** from the RoT

T19: **Stale TLB translations** (for U/HS mode or for VU/VS) created during TSM or TVM operations are used to execute malicious code in the TVM (or consume stale/invalid data)

T20: **Unexpected enabling of performance monitoring and/or debug** on a TVM leading to information loss via performance monitoring events/counters and debug mode accessible information.

T21: A **TVM causes a denial of service** on the platform