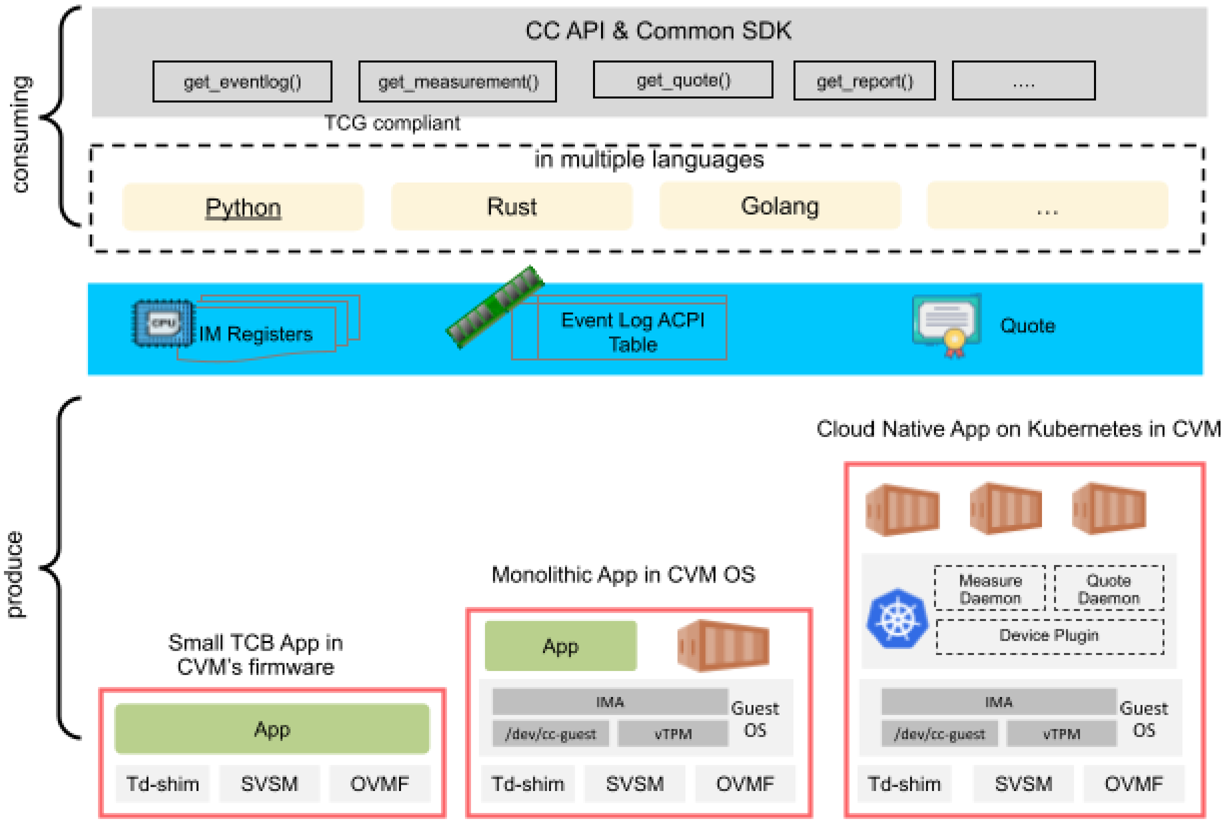
General Information

1.1. **Name of Project**: CC(Confidential Computing) API
1.2. **Project Description**:

CC API provides the vendor (Intel/AMD/ARM) agnostic confidential computing's primitives APIs for the handling of integrity measurement (*compliant with TCG FIM spec*), event log (*compliant with TCG ACPI spec and TCG PC PFP spec*) and quote (*binary blob which format is various across Intel TDX/Intel SGX/AMD SEV/ARM CCA and standard vTPM*) to support zero-trust design in diverse TEE environments from firmware, VM to container within CVM (Confidential Virtual Machine) guest.

CC API and SDK Overview



Beyond API definitions, it also provides back-end SDK for the common functions and utilities to process the measurement/eventlog according to *TCG FIM spec* and *TCG PC PFP spec*.

Ultimately, for the convenience of developers, it accommodates various programming languages such as Rust, Python, and Golang.

1.3. **How does this project align with the Consortium's Mission Statement**: "*The Confidential Computing Consortium brings together hardware vendors, cloud providers, and software developers to accelerate the adoption of Trusted Execution Environment (TEE) technologies and standards.*"

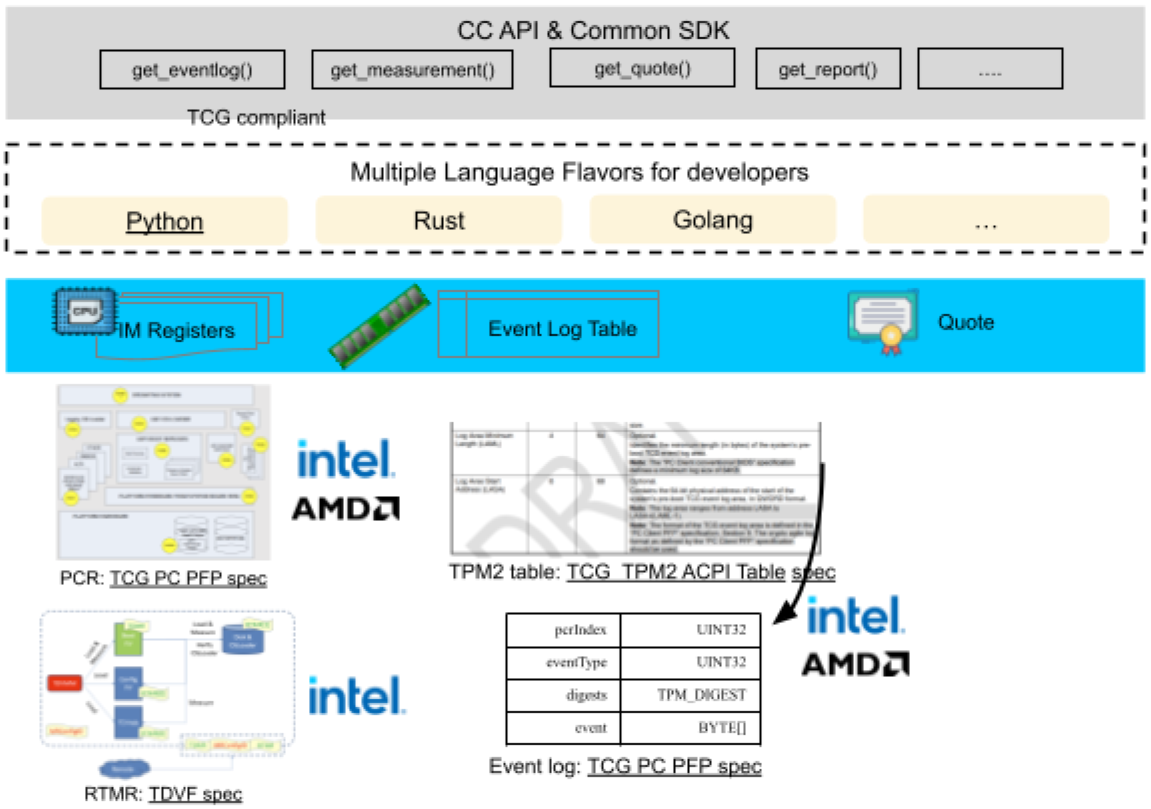
There are two missions of CC API & SDK project:

1. The first one is to simplify and unify API to consume TCG compliant measurement register, event log and binary quote blob
    a. across different vendor's platforms like Intel's RTMR (Runtime Measurement Register) + CCEL (Confidential Computing Event Logs) table (Intel) or vTPM (AMD/Intel)'s PCR (Platform Configuration Register) + TPM2 table.
    b. across diverse deployment environments from firmware, VM to cloud native.
    c. across diverse programming languages, for instance, Python, Rust, and Golang.

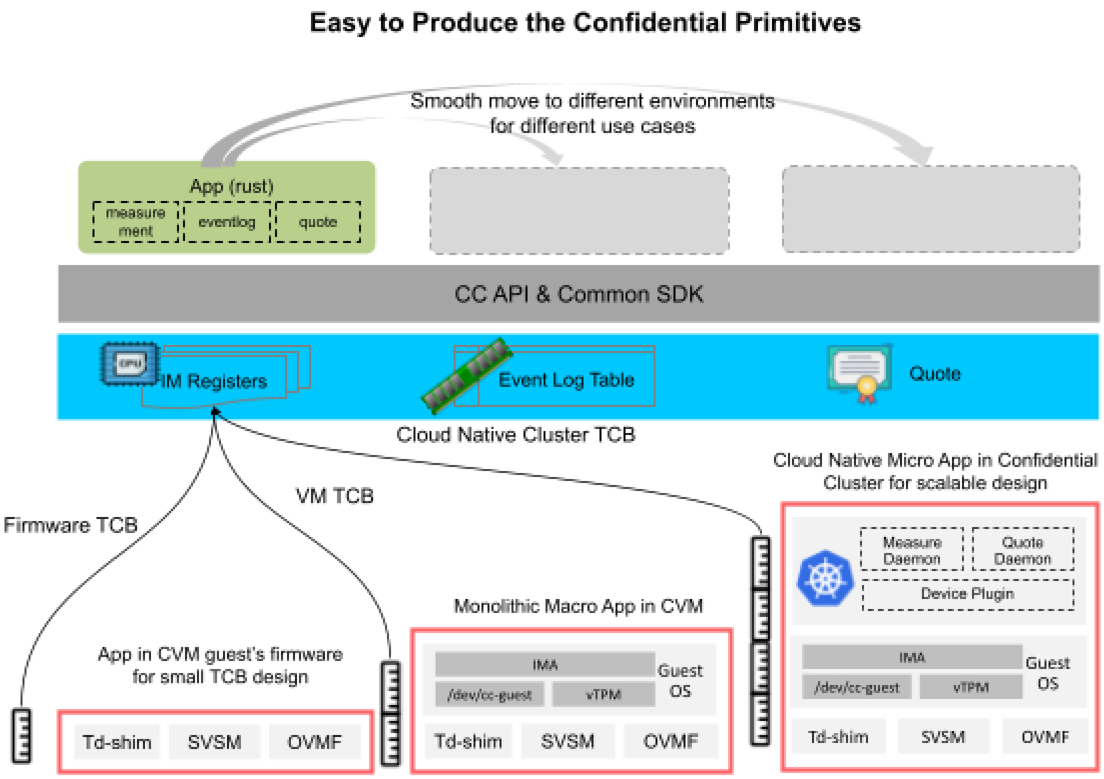
Note:

- *The data structure of the Intel RTMR register in the specification of TDVF is compatible with TCG PCR register in the specification of TCG PFP spec.*
- *The data structure of the event log in Intel CCEL ACPI table in the specification of TDVF is also compatible with the TPM2 ACPI table in the specification of TCG ACPI table.*

Easy for Tenant App to Access Confidential Primitives



2. The second one is to make it easy to process confidential primitives for diverse deployment environments via back-end SDK and services.
3. The back-end SDK and service supports various applications, for example, applications written in Python, bash script, C, Java, Rust, and Golang.


Easy to Produce the Confidential Primitives

Deployment Environments	Measurement	Quote
Firmware (td-shim/SVSM)	Platform call like TDVMCALL for TDX or TCG firmware protocol	Platform call like TDVMCALL for TDX
VM launch time	CC_MEASUREMEN_PROTOCOL or TCG protocol for vTPM	Platform call like TDVMCALL for TDX or TCG protocol for vTPM
VM OS runtime	IMA over RTMR or vTPM	Device node like /dev/tdx-guest, /dev/sev-guest, or vTPM.quote
Confidential Cluster (Kubernetes in CVM)	Device plugin or CCNP(Confidential Cloud Native Primitives) daemonset/sidecar. Calculate SRTM measurement per node, namespace and container level	Device plugin or CCNP(Confidential Cloud Native Primitives) daemonset/sidecar

1.4. **Project website URL**: https://github.com/intel/confidential-cloud-native-primitives (temporary, will move out later)

1.5. **Social media accounts**

Legal Information

2.1. **Project Logo**: (None)
2.2. **Project license**. Apache License 2.0 (See GitHub license.md).
2.3. **Existing financial sponsorship**. Intel SATG/SSE/Cloud Software Engineering
2.4. **Trademark status**: Currently there are no artifacts in the CC API repository and collateral which are trademarked.
2.5. **Proposed Technical Charter**: See attached "CCAPI Technical Charter" file)

Technical Information

3.1. **High level assessment of project synergy** with existing projects under the CCC, including how the project compliments/overlaps with existing projects, and potential ways to harmonize over time.
The CC API has not used existing projects under the CCC, but it can be used to simplify the other projects or tenant's application to access confidential computing primitives in CVM. The project used several specifications from TCG group and tools initiated by Intel like Confidential Cloud Native Primitives (CCNP) and tdx-tools.

3.2. **Describe the Trusted Computing Base (TCB) of the project**. If the project supports multiple environments please describe each TCB. Also identify if dependencies of other projects (both CCC or non-CCC) TCBs are taken.

CC-API provides unified APIs across different TCB. The back-end SDK/Services support diverse TCB from firmware, CVM OS (launch time and runtime) and Confidential Cluster, please see above table.

3.3. **Project Code of Conduct URL**:
https://github.com/intel/confidential-cloud-native-primitives/blob/main/CODE_OF_CONDUCT.md based on Contributor Covenant v2.0 (In future, the official project may be moved out of CCNP)
3.4. **Source control URL**: https://github.com/intel/confidential-cloud-native-primitives/
(In future, the official project will be migrated to new github)
3.5. **Issue tracker URL**:
https://github.com/intel/confidential-cloud-native-primitives/issues
3.6. External dependencies

Package/Library	Dependency at	
	Build-time	Run-time
Google Protocol Buffers	Yes	Yes
Go-lang Go, protoc-gen-go	Yes	Yes
Go-lang GRPC	yes	yes
Python3, pytest		Yes
Python3 grpc		Yes
Rust-lang, tokio	Yes	Yes
Rust-lang clap	Yes	Yes
Rust-lang tonic	Yes	Yes

3.7. **Standards implemented by the project**, if any. Include links to any such standards.

1. TCG PC Client Platform Firmware Profile Specification
2. TCG ACPI Specification
3. Intel TDX Virtual Firmware Design Guide
4. GHCI Specification for Intel TDX
5. Advanced Configuration and Power Interface (ACPI) Specification

3.8. **Release methodology and mechanics**: Currently, the CCAPI is at alpha stage in a publicly accessible Git repository. We expect a stable "v.9" release around Feb 2024.

3.9. **Names of initial committers**, if different from those submitting proposal (See OWNERS.md)

- Zhang, Wenhui (Bytedance/Tiktok)
- Jianjun Chen (Bytedance/Tiktok)
- Lu Ken (Intel)
- Dong, Xiaocheng (Intel)
- Cheng, Hairong (Intel)
- Ying, Ruoyu (Intel)

- Hao, Ruomeng (Intel)

3.10. **List of project's official communication channels** (slack, irc, mailing lists)
Github Discussions is our current channel.

3.11. Project Security Response Policy
On Git website, see SECURITY.md. The security policy process will be enforced in future releases.

3.12. Preferred maturity level (Sandbox, Incubation, Graduation, or Emeritus):
SandBox

3.13. Any additional information the TAC and Board should take into consideration when reviewing your proposal.

- Linux Plumber Conference 2022: Secure bootloader for Confidential Computing
- Runtime Integrity Measurement and Attestation in a Trust Domain
- Virtual TPM based attestation for Intel® TDX