Confidential Compute Consortium - Project Veraison Proposal

Updated: May 2023 for CCC project review

General Information

1.1. Name of Project:

- Project Veraison

1.2. Project Description (what it does, why it is valuable, origin and history)

- See https://github.com/veraison#welcome-to-veraison for background

1.3. How does this project align with the Consortium's Mission Statement

The CCC Mission Statement is: "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."

It is common to Trusted Execution Environments, from any vendor, that a means must be available for users to establish the trustworthiness of the environments. The recognised technique for achieving this is to use attestation to generate a set of evidence about the environment that can be appraised to establish trustworthiness. Verifying the evidence requires checks that the report is of valid format, from a known source and contains measurements that correspond to those approved by the relevant manufacturing supply chain for the deployment. Making verification checks consistently across multiple deployments and technologies and connecting to supply chain data can be complex, leading to custom solutions of varying quality and raising the complexity of establishing a Confidential Compute environment. Project Veraison will build software components that solve the complex problems in attestation verification. The intent is that these components can then be deployed into any ecosystem for any TEE technologies and bring some consistency to this aspect of a CC deployment.

1.4. Project website URL

- http://github.com/veraison

1.5. Social media accounts

- N/A

Legal Information

2.1. Project Logo URL or attachment (Vector Graphic: SVG, EPS).

- https://github.com/veraison/.github/blob/main/veraison-logo.png. SVG logo file also available.

2.2. Project license. We recommend an OSI-approved license, so if the license is not one on the list, explain why.

- The project uses the Apache license 2.0.

2.3. Existing financial sponsorship.

- The core team for the project has been staffed by members of the Architecture and Technology Group from Arm. The commitment of these resources to the project is currently unbounded.

2.4. Trademark status.

- Veraison is not trademark protected.

2.5. Proposed Technical Charter, based on the template.

- Draft of Technical Charter doc: https://github.com/veraison/community/CCC/verasion-ccc-technical-charter.pdf

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. Responses may be included both inline and/or in accompanying documentation.

A core part of any CCC deployment must be attestation and many of the CCC projects acknowledge this in some aspect. The potential for integration with Veraison components for the verification aspects of that integration can bring consistency and reduce custom solutions across projects.

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 project (both CCC or non-CCC) TCBs are taken.

TCB notes:

- Software components produced, e.g.: Verification pipeline, Verification Trusted Services component, Provisioning pipeline. All of the source code for these components is available within the project github repositories for analysis
- The correctness of the Go compiler and the Go runtime
- The correctness of all dependent libraries pulled into the project (see dependencies Section)
- In particular:
    - The HashiCorp go-plugin modules used to connect components of the Verification pipeline. Note there is an option to replace use of this component with a hard wired selection of scheme plugins.
- The correctness of any underlying deployment platform used to satisfy the platform abstraction. The project itself will only prepare reference deployments, not production ready code. The responsibility for this aspect of the TCB therefore lies with consumers of the project.
- The correctness of any storage DB technology deployed to hold provisioned data. The storage requirements are designed to be satisfied by multiple technologies such that a consumer of the project can select a technology which they trust to satisfy their TCB.

3.3. Project Code of Conduct URL. We recommend a Contributor Covenant v2.0 based Code of Conduct, so if the Code of Conduct is not based on that, explain why.

- Each repository under the Veraison org has a code of conduct statement based on CoCo v2. E.g., see https://github.com/veraison/.github/blob/main/CODE_OF_CONDUCT.md

3.4. Source control URL

- https://github.com/veraison

3.5. Issue tracker URL

- Global project board:
  - https://github.com/orgs/veraison/projects/1
- Per repo issue trackers: NB all repos have an issues section
  - https://github.com/veraison/services/issues
  - https://github.com/veraison/docs/issues
  - https://github.com/veraison/community/issues
  - https://github.com/veraison/apiclient/issues
  - https://github.com/veraison/corim/issues
  - https://github.com/veraison/dice/issues
  - https://github.com/veraison/eat/issues
  - https://github.com/veraison/go-cose/issues
  - https://github.com/veraison/psatoken/issues
  - https://github.com/veraison/swid/issues
  - https://github.com/veraison/eat/issues
  - https://github.com/veraison/ear/issues

3.6. External dependencies (including licenses, and indicate whether each is a build time or runtime dependency)

- See folder “license-scan” for the list of dependent packages and associated licenses
- All dependencies are build-time except for the Go runtime.

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

- TCG-DICE
  - https://trustedcomputinggroup.org/resource/dice-layering-architecture/ (in-progress)
- IETF EAT
  - https://datatracker.ietf.org/doc/draft-ietf-rats-eat/
- TCG CoRIM/CoMID
  - https://datatracker.ietf.org/doc/html/draft-birkholz-rats-corim
- ISO SWID
  - https://www.iso.org/standard/65666.html
  - https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8060.pdf
- IETF CoSWID
  - https://datatracker.ietf.org/doc/draft-ietf-sacm-coswid/
- PSA Attestation Token
  - https://datatracker.ietf.org/doc/draft-tschofenig-rats-psa-token/
- PSA CoRIM profile
  - https://datatracker.ietf.org/doc/draft-fdb-rats-psa-endorsements/
- IETF AR4SI
  - https://datatracker.ietf.org/doc/draft-ietf-rats-ar4si/

3.8. Release methodology and mechanics

- No public release yet
- In the future, we plan to use github's "releases" feature (https://github.blog/2013-07-02-release-your-software/)
    - Each release is tagged
    - Each release is uniquely identifiable using a SemVer string
    - Each release includes a "Release Notes" document, a human-readable summary of major changes in that release
        - Including any fixed CVE - or otherwise identifiable security vulnerability
- We will document the release procedure, along with a security response policy (see below) *before* any project deliverable is publicly released. This activity is tracked at: https://github.com/orgs/veraison/projects/2#card-76213010

3.9. Names of initial committers, if different from those submitting proposal

- See https://github.com/veraison/.github/blob/main/MAINTAINERS.toml

3.10. List of project's official communication channels (slack, irc, mailing lists)

- https://veraison.zulipchat.com
    - Project coordination
    - Weekly meetings agenda and minutes
    - GitHub CI pipeline integrations
- https://armltd.zoom.us/j/93024860563?pwd=dVpVcFRtSVFmV29HV3dHWENrZk5WQT09
    - Zoom room for the weekly project meeting
- veraison-project@confidentialcomputing.io (Point of contact for CoC violations)

3.11. Project Security Response Policy

- We haven't publicly released any deliverables yet, so strictly speaking there is no need to document the security response policy. This activity is tracked at: https://github.com/orgs/veraison/projects/2#card-76212867

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.

NB: this file resident at https://github.com/veraison/community/CCC/verasion-ccc-project-veraison-proposal.pdf