TAC Meeting

February 6, 2020

CONFIDENTIAL COMPUTING CONSORTIUM

Antitrust Policy Notice

- Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.
- Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at http://www.linuxfoundation.org/antitrust-policy. If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrove of the firm of Gesmer Updegrove LLP, which provides legal counsel to the Linux Foundation.

CONFIDENTIAL COMPUTING CONSORTIUM

Agenda

1. Roll call
2. Approval of minutes
3. Action item review
4. Scoping discussion:
   I. Outreach committee report from Seth
   II. CCC presentation/document for governing board
   III. CCC presentation to Homomorphic Encryption Workshop
5. Review github issues and pull requests
6. Any other business

Roll Call of TAC Voting Representatives

Quorum requires 5 or more voting reps:

Member	**Representative**	**Email**
Alibaba	Xiaoning Li	xiaoning.li@alibaba-inc.com
ARM	Grant Likely	grant.likely@arm.com
Facebook	Jinsong Yu	jinsongyu@fb.com
Google	Brandon Baker	bsb@google.com
Huawei	Zhipeng (Howard) Huang	huangzhipeng@huawei.com
Intel	Simon Johnson	simon.p.johnson@intel.com
Microsoft	Dave Thaler(*)	dthaler@microsoft.com
Oracle	John Haxby	john.haxby@oracle.com
Red Hat	Mike Bursell	mbursell@redhat.com

**TAC chair*

CONFIDENTIAL COMPUTING CONSORTIUM

Approval of Minutes

https://lists.confidentialcomputing.io/g/main/files/TAC/Meetings/2020/CCC%20TAC%20Minutes%202020-01-23.docx

RESOLVED: That the minutes of the January 23, 2019 meeting of the Technical Advisory Committee meeting of the Confidential Computing Consortium as distributed to the members of the TAC in advance of this meeting are hereby adopted and approved.

Action Item Review

- [Stephano] Eventually provide a better way (wiki list, perhaps GitHub issues) to track website content requests and their progress.
- [Seth/Dave] Discuss how TAC/Outreach can coordinate.
- [Stephano] Determine if the last meeting was recorded and if so, place a link to that recording in Groups.io
- [Dave] Summarize the discussion around CCC definition and scope for the Governing Board [ON AGENDA LATER]
- [Simon Johnson] Provide a proposal on project template changes for a project to describe its TCB. [DONE, ON AGENDA LATER]
- [Pushkar Chitnis] Provide details for OE SDK ask on CI/CD pipeline costs.
- [All] Review and comment on Code of Conduct on GitHub.

Scoping discussion

- Outreach committee report [Seth]

Upcoming scoping discussion meetings

1. CCC presentation/document for governing board (Feb. 27)
   - See separate document in progress
2. CCC presentation to Homomorphic Encryption Workshop (today)
   - See separate deck in progress

Review github issues and pull requests

- https://github.com/confidential-computing/governance/issues
  - [DONE] #4: Project Progression Policy: requirement for 2 mentors
  - [DONE] #7: Progression Template: Add requirement for description of TCB
  - #9: Template needs link to Consortium's Mission Statement
- https://github.com/confidential-computing/governance/pulls
  - [MERGED] #8: Update to progression template to Address issue 7
  - [MERGED] #10: Add links and expand acronyms in template

Any other business

- Next face to face meeting: **February 27, 10am-noon**
  - 8-10am is Outreach meeting, but also a BOF at RSA
  - 2-4pm is Governing Board meeting
- Next meeting: Feb. 13 or 20 or 27?
  - “There is agreement that 1 hour every 2 weeks is not enough time to be productive.”
  - Dave will be on a plane on the 20th returning from IETF hackathon in Berlin

BACKUP:
Terminology/Scoping Slides from Previous Meetings

“Confidential computing” definition & CCC scope

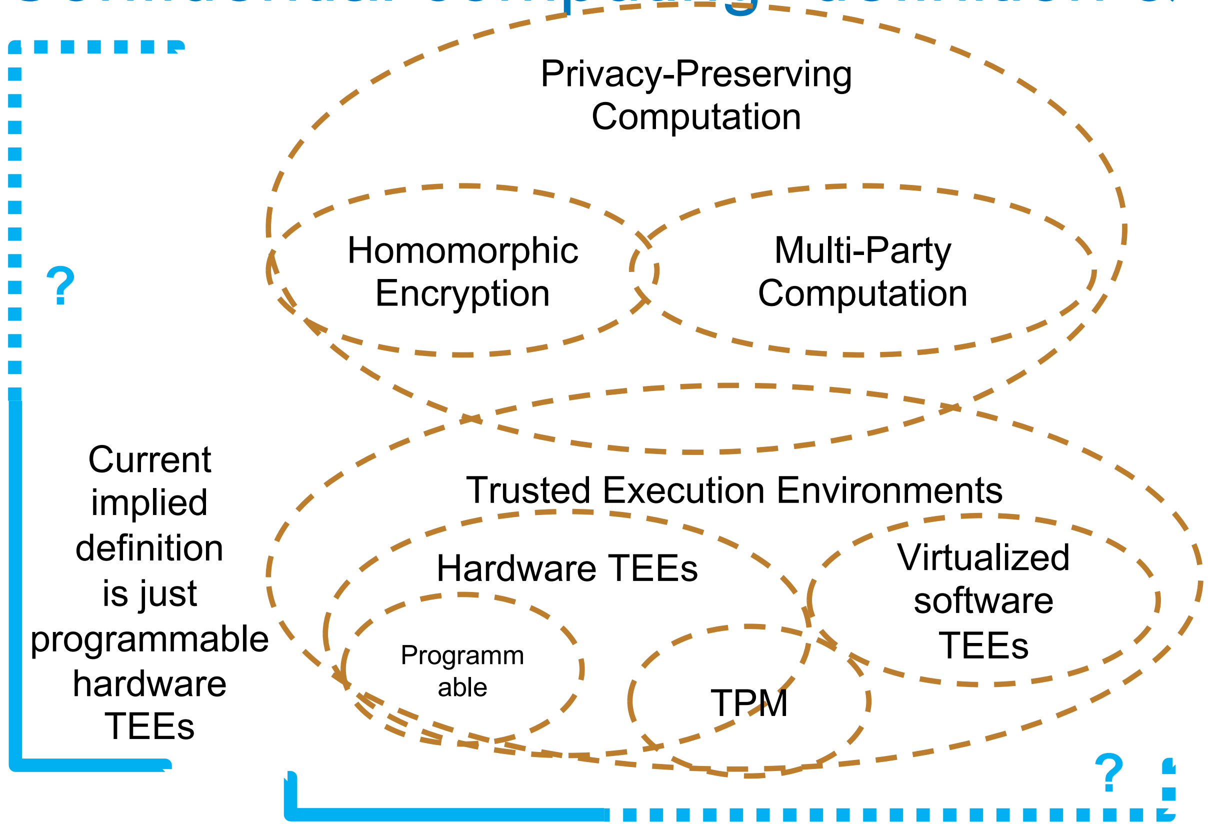
Two related, but different, questions:

1. Should the term “confidential computing” be broad like “privacy preserving computation”, or narrowly scoped to TEEs (or even certain classes of TEE)?
2. Should the consortium’s scope be more inclusive, or narrowly scoped to TEE-based projects

    - If narrow, focus stays on TEEs, and messaging on terminology might compete with other bodies in the industry
    - If broad, this discussion happens inside the CCC and the CCC has the opportunity to have unified messaging

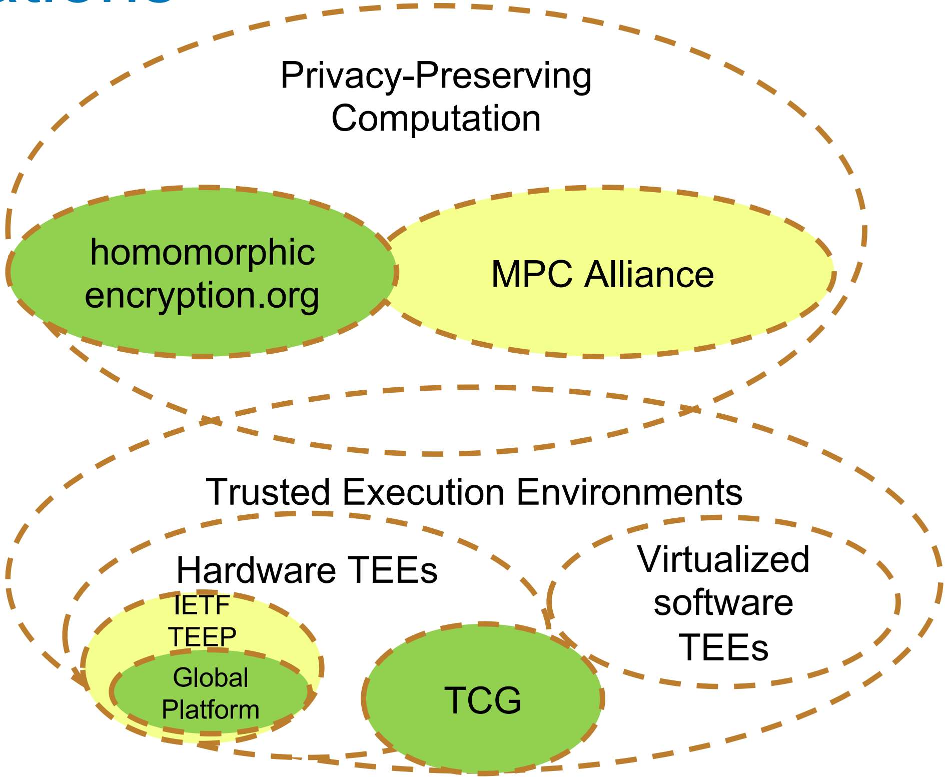
CONFIDENTIAL COMPUTING CONSORTIUM

“Confidential computing” definition & CCC scope


Privacy-Preserving Computation
Homomorphic Encryption
Multi-Party Computation
?
Current implied definition is just programmable hardware TEEs
Trusted Execution Environments
Hardware TEEs
Virtualized software TEEs
Programmable
TPM
?

Disclaimer:
Some terms have multiple competing definitions, so boundaries are often fuzzy.

Organizations


Privacy-Preserving Computation
homomorphic encryption.org
MPC Alliance
Trusted Execution Environments
Hardware TEEs
IETF TEEP
Global Platform
TCG
Virtualized software TEEs

Possible axes for categorising technologies

Rough consensus that the following 5 axes are important to answer:

Axis	**TAC Consensus**
algorithmic (mathematical) vs hardware/software	? (split opinions)
hardware (+firmware?) vs software	? (some argued for software too)
generalised vs specialised computation only	No one argued for non-programmable
on-main CPU vs off-main CPU	Broad
cloud vs on-prem (incl. IoT)	Broad (whole axis)

Other attributes (e.g., TCB size) are also important evaluation criteria but by themselves weren't seen as part of scoping question per se

Example technology	Hard-/software Implementation or Algorithmic	Hardware or Software)	Generalised compute or Specialised)	ON-main CPU (vs oFf-main CPU)	Cloud (vs on-Prem, incl. IoT)
Homomorphic encryption	A	---	S?	---	C/P
Multi-party Computation	A	---	G?	---	C/P
HSM	I	H	S (can be G?)	F	C/P
TPM	I	H	S	F	C/P
Hardware TEE on main CPU (e.g., SGX)	I	H	G	N	C/P
Virtualised software TEE	I	S	G	**N**	**C/P**
FPGA	I	H	S	F	C/P
TEE in NIC	I	H	S	F	C/P
Secure Element	I	H	S?	F	P
…?					

Different definitions of TEE

Problem

- **Wikipedia:** A secure area of a main processor. It guarantees code and data loaded inside to be protected with respect to confidentiality and integrity. A TEE as an isolated execution environment provides security features such as isolated execution, integrity of applications executing with the TEE, along with confidentiality of their assets.
- **ARM:** a secure area inside a main processor. It runs in parallel of the operating system, in an isolated environment. It guarantees that the code and data loaded in the TEE are protected with respect to confidentiality and integrity.
- **IETF TEEP WG:** An environment that enforces that only authorized code can execute with that environment, and that any data used by such code cannot be read or tampered with by any code outside that environment.
- **GlobalPlatform:** A device that conforms to specifications from GP's TEE Committee
- **Mike**: a hardware-based technique for securing sensitive data and algorithms in such a way that even the kernel, root user or hypervisor can't see what's going on

Other aspects that are important but may not be part of the definition itself: attestation, identity, hardware tamper-evident/resistant, …

CONFIDENTIAL COMPUTING CONSORTIUM

TEE variations

- A processor (e.g., an MCU) might *only* have a TEE and no REE
- Separate processors may have (or be) a “TEE”:
  - Secure Element, FPGA, HSM, TPM, NIC
- A “TEE” might not be programmable
  - E.g., TPM, secure cryptoprocessor
- A virtualized TEE might be indistinguishable in practice from a hardware TEE except in terms of which certificate(s) it chains up to

TEE variation definitions (1/2)

- **secure cryptoprocessor**: a dedicated computer-on-a-chip or microprocessor for carrying out cryptographic operations, embedded in a packaging with multiple physical security measures, which give it a degree of tamper resistance. Unlike cryptographic processors that output decrypted data onto a bus in a secure environment, a secure cryptoprocessor does not output decrypted data or decrypted program instructions in an environment where security cannot always be maintained. The purpose of a secure cryptoprocessor is to act as the keystone of a security subsystem, eliminating the need to protect the rest of the subsystem with physical security measures.
- **Trusted Platform Module** (**TPM**, also known as **ISO/IEC 11889**): an international standard for a secure cryptoprocessor, a dedicated microcontroller designed to secure hardware through integrated cryptographic keys.
- **hardware security module** (**HSM**): a physical computing device that safeguards and manages digital keys for strong authentication and provides cryptoprocessing.
- **Secure Element (SE)**: a microprocessor chip which can store sensitive data and run secure apps such as payment. It acts as a vault, protecting what's inside the SE (applications and data) from malware attacks that are typical in the host (i.e. the device operating system).

CONFIDENTIAL COMPUTING CONSORTIUM

TEE variation definitions (2/2)

- **Dedicated Security Component:** the combination of a **hardware** component and its controlling firmware dedicated to providing the encompassing platform with services for the provisioning, protection, and use of Security Data Objects (SDOs) consisting of keys, identities, attributes, and other types of Security Data Elements (SDEs).

From
https://www.commoncriteriaportal.org/communities/docs/cpp_dsc_v10d_DRAFT_20190501.docx

Privacy-preserving computation

- **multi-party computation (MPC)**, or **privacy-preserving computation**: a subfield of cryptography with the goal of creating methods for parties to jointly compute a function over their inputs while keeping those inputs private. Unlike traditional cryptographic tasks, where cryptography assures security and integrity of communication or storage and the adversary is outside the system of participants (an eavesdropper on the sender and receiver), the cryptography in this model protects participants' privacy from each other.
- **Homomorphic encryption**: a form of encryption that allows computation on ciphertexts, generating an encrypted result which, when decrypted, matches the result of the operations as if they had been performed on the plaintext. Homomorphic encryption can be used for **privacy-preserving** outsourced storage and **computation**. This allows data to be encrypted and out-sourced to commercial cloud environments for processing, all while encrypted.

Confidential Computing (1/2)

Problem

- **Gartner report:** Confidential computing is the combination of CPU-based hardware technology and infrastructure as a service (IaaS) cloud provider virtual machine (VM) images and software tools that enable cloud-using organizations to create completely isolated trusted execution environments (TEE), also called enclaves. Because they offer a form of encryption of data in use, these enclaves render sensitive information invisible to host OSs and cloud providers.
- **CCC press release:** Established in 2019, the Confidential Computing Consortium brings together hardware vendors, cloud providers, developers, open source experts and academics to accelerate the confidential computing market; influence technical and regulatory standards; build open source tools that provide the right environment for **TEE** development' and host industry outreach and education initiatives. Its aims to address **computational trust and security for data in use, enabling encrypted data to be processed in memory without exposing it to the rest of the system**, reducing exposure to sensitive data and providing greater control and transparency for users.

CONFIDENTIAL COMPUTING CONSORTIUM

Confidential Computing (2/2)

- **Mark Russinovich blog:**
  - Put simply, confidential computing offers a protection that to date has been missing from public clouds, **encryption of data while in use**. …
  - Confidential computing ensures that when data is “in the clear,” which is required for efficient processing, the data is protected inside a **Trusted Execution Environment** (TEE - also known as an enclave), an example of which is shown in the figure below. TEEs ensure there is no way to view data or the operations inside from the outside, even with a debugger. They even ensure that only authorized code is permitted to access data. If the code is altered or tampered, the operations are denied and the environment disabled. The TEE enforces these protections throughout the execution of code within it.

John Haxby wrote:

- After last week's meeting I think we almost had a definition of the scope as simple as
  - **"Software solutions to enable the widespread use of hardware trusted execution environments"**.
- "Software solutions" probably needs to be replaced by something else, perhaps even just "software" and perhaps "hardware" could be "hardware-assisted".
- So, clearly, all three projects adopted so far fall under that definition but some others might be useful:
  - A software TEE emulation for development
  - A virtual machine TEE using encrypted memory so no one, not even the hypervisor, can look inside it. (That would be "hardware-assisted perhaps".)
- A TEE in/on a NIC, GPU, discrete (socketed) chip, thumbdrive, etc all fall into the "hardware" category as something you physically hold.
- A TEE that relies on, for example, isolated or encrypted memory to keep its function away from prying eyes would be "hardware assisted". (SGX falls into that category doesn't it?)

Software TEE examples

- **Virtual Secure Mode (VSM):** a software-based TEE that's implemented by Hyper-V in Windows 10 and Windows Server 2016. Hyper-V prevents administrator code running on the computer or server, as well as local administrators and cloud service administrators from viewing the contents of the VSM enclave or modifying its execution.
    - https://azure.microsoft.com/en-us/blog/introducing-azure-confidential-computing/
- **QEMU ("quick emulator"):** very widely used open source machine emulator. … Developers can use the QEMU Arm Security Extensions to develop and work with Trusted Execution Environments (TEEs) that are likely to be the primary consumers of the added functionality. Secure applications can then be developed on the added TEEs without the need for dedicated hardware.
    - https://www.linaro.org/blog/arm-trustzone-qemu/