1 Trusted Execution Environment: What it is, and What it is not

1.1 Introduction

Trusted Computing was defined to help systems to achieve secure computation, privacy and data protection. Originally, trusted computing relies on a separate hardware module that offers a functional interface for platform security. A TEE is a secure, integrity-protected processing environment, consisting of memory and storage capabilities.

1.2 Trusted Execution Environment

The separation kernel is a foundation component of the TEE. It is the element that assures the property of isolated execution. The separation kernel, is a security kernel used to simulate a distributed system. Basically, it divides the system into several partitions, and guarantees a strong isolation between them, except for the interface for inter-partition communication. The SKPP defines separation kernel as "hardware and/or firmware and/or software mechanisms whose primary function is to establish, isolate and control information flow between those partitions.".

Policies The security requirements are composed of four main security policies:

- Data (spatial) separation. Data within one partition cannot be read or modified by other partitions
- Sanitization (temporal separation). Shared resources cannot be used to leak information into other partitions
- Control of information flow. Communication between partitions cannot occur unless explicitly permitted
- Fault isolation. Security breach in one partition cannot spread to other partitions

Definition Trusted Execution Environment (TEE) is a tamper-resistant processing environment that runs on a separation kernel. It guarantees the authenticity of the executed code, the integrity of the runtime states and the confidentiality of its code, data and runtime states stored on a persistent memory. In addition, it shall be able to provide remote attestation that proves its trustworthiness for third-parties. The content of TEE is not static, it can be securely updated. The TEE resists against all software attacks as well as the physical attacks performed on the main memory of the system. Attacks performed by exploiting backdoor security flaws are not possible.

Trust In static trust, the trustworthiness of a system is measured only once and before its deployment. Dynamic trust is based on the state of the running system, and thus it varies accordingly. A system continuously changes its "trust status". In dynamic trust, the trustworthiness of a system is constantly measured throughout its lifecycle.

Root of Trust This definition requires a trusted entity called Root of Trust (RoT) to provide trustworthy evidence regarding the state of a system. The role of RoT is divided into two parts. First is the trusted measurement and second is the function that computes the trust score. The trustworthiness of the system, namely the generated score, depends on the reliability of the trust measurement. If a malicious entity can influence the trust measurement, then the generated score of trustworthiness is of no value. Therefore, RoT is necessarily a tamper-resistant hardware module.

Secure Execution Environment Secure Execution Environment (SEE) is a prerequisite for TEE, but it does not consider trust aspects. SEE is a processing environment that guarantees the following properties:

- Authenticity
- Integrity
- Confidentiality

In contrast to TEE, the design of SEE does not involve RoT to assert the integrity and authenticity of the loaded code. Moreover, it does not define secure mechanisms to update its applications and confidential data. In fact, in our definition, TEE is an open SEE that guarantees trust.

1.3 TEE Building Blocks

Secure Boot Secure Boot assures that only code of a certain property can be loaded. If a modification is detected, the bootstrap process is interrupted. An example implementation of secure boot is to verify the integrity of a succeeding component according to a given reference value.

Inter-Environment Communication Inter-Environment Communication defines an interface allowing TEE to communicate with the rest of the system. Each communication mechanism should satisfy three key attributes:

- Reliability (memory/time isolation)
- Minimum overhead
- Protection of communication structures

Secure Storage Secure Storage is storage where confidentiality, integrity and freshness of stored data are guaranteed, and where only authorized entities can access the data. It is based on three components:

- Integrity-protected secret key only accessible by the TEE
- Cryptographic mechanisms
- Data rollback protection mechanisms

Trusted I/O Trusted I/O Path protects authenticity, and optionally confidentiality, of communication between TEE and peripherals. Thus, input and output data are protected from being sniffed or tampered with by malicious applications. To be more precise, trusted I/O path protects against four classes of attacks: screen-capture attack, key logging attack, overlaying attack, and phishing attack. Trusted path to user-interface devices enables broader functionality within TEE. It allows a human user to directly interact with applications running inside TEE.

1.4 ARM TrustZone based TEE

ARM TrustZone technology can be seen as a special kind of virtualization with hardware support for memory, I/O and interrupt virtualization. This virtualization enables ARM core to provide an abstraction of two virtual cores (VCPUs): secure VCPU and non-secure VCPU. The monitor is seen as a minimal hypervisor whose main role is the control of information flow between the two virtual cores.