# Arm Hypervisor and Trustzone Alternatives

**Nezer Jacob Zaidenberg**
*College of Management, Israel*

**Raz Ben Yehuda**
*University of Jyväskylä, Finland*

**Roee Leon**
*University of Jyväskylä, Finland*

## INTRODUCTION

ARM holdings have proposed TrustZone™ as means to create a Trusted Execution Environment (TEE) on the ARM platform. On many scenarios, such Trust Execution Environment is required to provide DRM support, secure wallets, Trusted endpoints, point of sale and other embedded systems.

Other than the mobile platform, TrustZone™ can be found in other ARM socs, such as AMD with their "Hiero falcon", AppleMicros X-Gene3, Cavium Thunder X etcetera.

Virtualization as a security solution is also widespread. Safe execution through sandboxing is a standard method for security. Several applications offer methods for trapping sensitive instructions into a hypervisor. Cloud computing technology, initially designed for dynamic provisioning of computing resources, is by its nature exposed to the public. Therefore, the virtual machine is exposed to many threats. Also, as the ARM architecture-based servers technology spreads, ARM virtualization technology can ease filtering out threats and monitor activities.

Multiple vendors offer their own TEE implementations. Some TEE implementations such as Trustonic and Qualcomm QSEE are closed source while others are open source or provide source code for a fee. This chapter surveys Trusted computing alternatives for implementations. The chapter mainly considers alternatives with available source code that offers a complete solution for the TrustZone™ environment, and also offers some ARM virtualization alternatives.

## BACKGROUND

### Trusted Execution Environment

The ARM architecture allows for co-existence of a Trusted Execution Environment (TEE) and Rich Execution Environment (REE). Trusted Execution Environment is a secure area inside the central processor unit (hereby CPU). The Trusted Execution Environment runs its own operating system. The TEE operating system is a separate operating system that is running in parallel with the REE(main) operating system, in an isolated environment. The Trusted Execution Environment guarantees that the code and data loaded in the TEE are protected concerning confidentiality and integrity.

DOI: 10.4018/978-1-5225-9715-5.ch079

Rich Execution Environment is another area inside the CPU. The Rich Execution Environment runs a separate operating system. Usually, Google's Android or Apple's iOS. The Rich Execution Environment refers to the standard operating system that the device is running. The Rich Execution Environment offers significantly more features and applications and as a result, is vulnerable to attacks. In most cases, the Rich Execution Environment is the environment where most applications are running. The Rich Execution Environment receive services such as decryptions keys from the Trusted Execution Environment.

The Trusted Execution Environment usually act as a monitor for the Rich Execution Environment.

The Trusted Execution Environment has higher permissions and usually have access to read the Rich Execution Environment memory and data structures. The Rich Execution Environment should not have access to the trusted Execution environment memory and data structures. The two worlds, the secured and the normal (not trusted, non-secured) worlds, can switch through the strict supervision of a Secure Monitor running in monitor mode. Switching between the secure and normal world can be done through a special instruction called "secure monitor call" or SMC. Software use SMC to communicate between the secure and normal worlds shared memory is used. TrustZone™ splits the SOC devices to the secure and normal worlds. TrustZone™ control the device hardware interrupts. TrustZone™ can route an interrupt to the secure world or the normal world. Like in the memory case, I/O and interrupts routing may change dynamically. TrustZone™ uses its own MMU. Operating systems and processes that execute in TrustZone™ do not share the same address space with their normal world counterparts. Thus, there is no need to have distinct TrustZone™ for each processor. A single TrustZone™ OS across multiple ARM processors/cores can manage all the device Trusted computing needs. The cryptographic keys are accessible only in TrustZone™. The manufacturer can burn platform-specific keys using fuses. These platform-specific keys are device specific, thus enabling protection in the end unit level.

Booting a Trusted Execution Environment must form a chain of trust in which a trust nexus verifies the next component on the boot chain. Each component verifies the next component until the system. Many vendors proposed.

## ARM Permission Model

ARMv8 architecture has a unique approach to privilege levels.

The ARM platform has 4 exception (permission) levels.

ARM also has the secure world (TrustZone™) and the normal world (non-TrustZone™)

ARM Exception levels are described in Table 1 Each of the exception levels provides its own set of registers and can access the registers of the lower exception levels but not registers of higher exception levels. The general-purpose registers are shared.

Thus, moving to a different exception level on the ARM architecture does not require the expensive context switch that is associated with the x86 architecture.

ARMv7 architecture is similar to ARMv8. ARMv7 offers virtualization as an extension that is only available to some late ARMv7 models. ARMv7 does offer TrustZone™. Furthermore, ARMv7 is 32bit architecture while ARMv8 is 64bit (and 32bit) architecture.

## Virtualization vs TrustZone™ Mode

The first question we must address is how the operating system should be verified. The REE operating system can be verified using HYP mode or TrustZone™. ARM has designed the TrustZone™ mode specifically for attesting and monitoring the Rich operating system. Only the vendor can install soft-