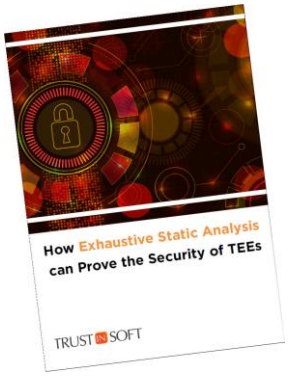
CLIENT	**TrustInSoft**
PROJECT	**White Paper on securing trusted execution environments**
OBJECTIVE	**Create a lead-generation white paper on securing trusted execution environments with exhaustive static analysis.**

COPY EXCERPT

How Exhaustive Static Analysis Can Prove the Security of TEEs



Call or write CopyEngineer to receive a PDF of the complete case study.

Or view/download it online at: *https://bit.ly/TrustInSoft-WP-TEE*

The need for trust… and a trusted execution environment

Twenty-first-century technology is increasingly complex, software-driven, and connected. Plus, consumer technology is becoming increasingly personalized, carrying more and more sensitive data that must be protected.

We need to be able to **trust our devices** to protect both themselves and our personal data from malevolent hackers. In essence, we need our devices to guarantee a high level of trust.

A trusted execution environment (TEE) is a secure area within a processor designed to provide the level of trust we require. It is an environment in which the code executed and the data accessed are both isolated and protected.

It ensures both confidentiality (no unauthorized parties can access the data) and integrity (nothing can change the code and its behavior).1 Figure 1 illustrates the partitioning of a trusted execution environment (“Secure World”) within a processor.

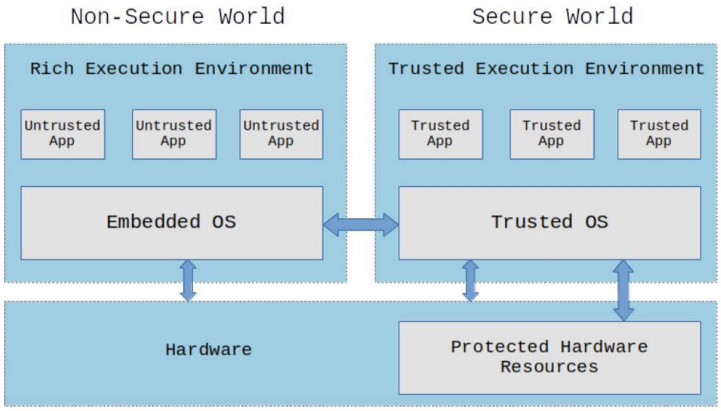


Figure 1: Partitioning a system using a trusted execution environment (Source: #embeddedbits)2

TEEs are essential components of many of the devices we own and use, including smartphones, tablets, game consoles, set-top boxes and smart TVs. They are well suited to providing security for applications like storage and management of device encryption keys, biometric authentication, mobile e-commerce applications, and digital copyright protection.

For a TEE to fulfill its function, however, the behavior of its code must be perfectly deterministic, reliable, and impervious to attack. Therefore, it must be free of software errors that could be sources of anomalous behavior and vulnerabilities for hackers to exploit.