A Secure Filesystem in Userspace via SE*cube*™

Giuseppe AIRÒ FARULLA[1], Paolo PRINETTO[1], Antonio VARRIALE[2]

[1] CINI Cybersecurity National Lab & Politecnico di Torino, Italy, {name.surname}@polito.it
[2]Blu5 Labs Ltd., Malta, av@blu5labs.eu

Abstract

The **SE***cube*™ Open Source platform is a 3D SiP (System in Package) including three main cores: a low-power ARM Cortex-M4 processor, a flexible and fast Field-Programmable-Gate-Array (FPGA), and an EAL5+ certified Security Controller (Smart-Card). This makes it a unique Open Source security environment where each function can be optimized, executed, and verified on its proper hardware device.

Leveraging the **SE***cube*™, it is possible to virtualize and maintain secure filesystems, protecting sensible data and application, that can only be accessed through by means of a storage-access firewall: stored data are strongly encrypted and digitally signed, and are accessible from trusted third-parties' applications, only. Without the **SE***cube*™ device (which can be as small as a microSD card) and its password, not even the structure of the filesystem itself is disclosed.

Introduction

Any OS provides an abstraction layer in its kernel space, used to separate file system generic operations from their implementation. This is performed by defining a clean Virtual File System (VFS) interface. Data protection is provided at this level of abstraction, by means of a dedicated security engine, typically referred to as *secure layer*.

Several implementations of the VFS interface may coexist on the same machine, allowing transparent access to different types of file systems mounted locally. In any case, whichever is the chosen implementation, a malicious user, or software, still may exploit flaws in the application accessing the secure layer or even in the secure layer itself. A countermeasure to protect effectively data, thus, resorts to hardware key management techniques applied to powerful embedded systems that can perform complex cryptographic operations while, at the same time, increasing the confidence of data security. A secure device can guarantee data protection also when the host machine is compromised.

SE*file*™ is a file system abstraction layer which exploits the hardware key management and other functionalities from the **SE***cube*™ device. It has been developed having in mind the needs to ensure both simplicity of usage and security for *data-at-rest*: it allows secure creation, storage, retrieval and usage of information that could not be trusted if stored elsewhere, e.g., any personal computer, or cloud service provider.

Conceptually, **SE***file*™ targets any user that, by moving inside a secure environment, wants to perform basic operation on regular files. All the encryption functionalities it offers are demanded to the secure device in their entirety, guaranteeing the highest level of security. In addition, **SE***file*™ does not expose to the host device information and details about what data it is reading (writing), nor from (to) where: thus, the host OS, which might be untrusted, is totally unaware of the operations the user is performing on data.

The demo

The filesystem in user-space (FUSE) is an open source project seeking to create a module for the OS kernel that allows non-privileged users on the system to create their own file systems without being forced to write kernel-level code. This goal is achieved by running the file system code in user space, while the FUSE module operates as a "bridge" to the kernel interface, only. FUSE is a very powerful system: virtually every available storage resource on the machine can become a virtual filesystem.

We developed a Windows-based wrapper for FUSE that mounts a virtual filesystem and manages its files and directories. Every read and write operation to, and from, the virtual filesystem is encrypted and authenticated through the security functionalities exposed from the **SE***cube*™, to the extent that the file storing the filesystem itself is encrypted and cannot be accessed without a valid, and unlocked, **SE***cube*™ device.

The virtual filesystem is protected by a custom storage-access firewall that prevents unauthorized applications to mount and decrypt it, thus preventing undesired lacking of the stored information.

Participants to this demonstration will interact with a machine where the **SE***cube*™ and the FUSE environments are running. They will see how the virtual filesystem is accessible if the **SE***cube*™ device is plugged in and the proper login password is provided. When properly mounted, access attempts from unauthorized applications are blocked and logged.

Acknowledgments

The present work has been partially supported by the Project "*FilieraSicura: Securing the Supply Chain of Domestic Critical Infrastructures from Cyber Attacks*".