Security and Trusted Hardware Applications
Week #8 Tutorial

Pedro Antunes - up201507254

May 30, 2022

Instructor: Bernardo Portela

1

The two trusted hardware technologies described above have specific functionality and are intended for specific applications. Trusted Platform Modules enhance specific validation functionalities, such as certifying boot processing and the integrity of an operating system. Hardware Security Modules are specifically intended for cryptographic operations, such as high-speed cryptographic services.

That said, none of these technologies can support secure computation over a set of information because the functionalities of performing cryptographic operations do not allow us to use the data to compute other types of information that derive from encrypted data.

Having said this, we have to use a technology that provides us with an environment that guarantees Secure Outsourced Computation. There are technologies with this type of guarantees, and these guarantees are usually made through a memory isolation/busbar.

2

Intel SGX security mechanism for intra-platform attestation focuses on asking the processor to produce MACs using the target enclave key. In an inter-platform scenario, we have a processor that controls the enclaves on the client side and a processor that controls the enclaves on the server side. However, to perform inter-platform attestation, the client-side processor cannot produce MACs for enclaves that reside on the server-side processor because it does not have access to those keys. Only the server-side processor knows the keys to its enclaves.

That said, we need better mechanisms for one machine to communicate with another machine and perform Secure Outsource Computation. Intel SGX inter-platform attestation focuses on assigning an Intel signed key to each processor with SGX abilities (the same key for all processors). Associated with this key is a special enclave called the quoting enclave, which is specifically initialized to manage this Intel signed key.

Thus, we just have to use the intra-platform mechanisms but using this specialized quoting enclave. So by generating a MAC for the quoting enclave, this enclave can verify the MAC and produce an Intel digital signature produced by the key only accessible in the quoting enclave. This signature that can then be verified on another independent Intel machine.

3

ARM TrustZone must only allow trusted applications because the memory bus in this technology consists of subdividing the physical memory in two. On one side is the normal world and on the other side is the trusted world. This means that an application will stay in one of these worlds, and depending on which world it stays, it will share its memory with other applications.

So we want only trusted applications to have access to the safe world. Otherwise, if we have a malicious application in this world, the memory bus has no effect, and so this application can read and write information from other trusted applications that also live in the safe world.

Intel SGX can have both trusted and malicious applications running because each enclave has its own bus. Therefore, each application running in one enclave is protected with this bus and cannot be read or written by another enclave.

4

ARM TrustZone is advantageous when compared to the x86 architecture associated with Intel SGX in situations like:

- when an system leans towards IoT devices, because usually this devices used ARM architecture.
- when we need to load trusted firmware, because ARM TrustZone work under the assumption that a root-of-trust has verified the system integrity, and using this, the secure boot verification is no more like chaining verifications of hashes values based on first trusted hash value from root-of-trust. In Intel SGX hardware, we can do this too but on a complex way, creating enclaves and using intra-platform security guarantees.

5

The advantages of having an initial development stage, where the trusted hardware component is left abstract are the ability to identify needed assumptions of trusted hardware system (confidentiality, integrity, isolated environment for code execution) and then build application functionalities based on security assumptions. After this being set, the system manager can choose the trusted hardware that would be more fit to perform predefined tasks and ensure that trusted anchor chosen can produce such security role.

6

a) If inadvertently was revealed the internal values stored in normal enclaves, the application aimed to work based on this enclaves are compromised because the memory bus is no longer there, the memories used by these enclaves are now accessible to any internal process, since the enclave key can be leaked. Therefore, an attacker living on a local machine will have access to read, write, and compute performed by the local enclaves.

b) If inadvertently was revealed the internal values in the quoting enclave, an attacker can sign messages with the leaked Intel signing key. That said, an attacker has sign Intel privileges which can be used to compromised any system that uses the quoting enclave to verify Intel signatures like inter-platform attestation.

c) With the absolute guarantee that only the input/output values are received, when interacting with Enclaves, does not imply absolute security of data stored in the container because it uses shared memory that have no bus memory.

   Therefore, one possible attack that can be done is side-channel attack to try to leak data stored on enclave. These range from physical attacks to logical attackes. One pratical example is try to leak cache information when the enclave produce some action with data stored (which is the time the data was cached), using known attacks like Meltdown and Spectre attack.