Intel SGX

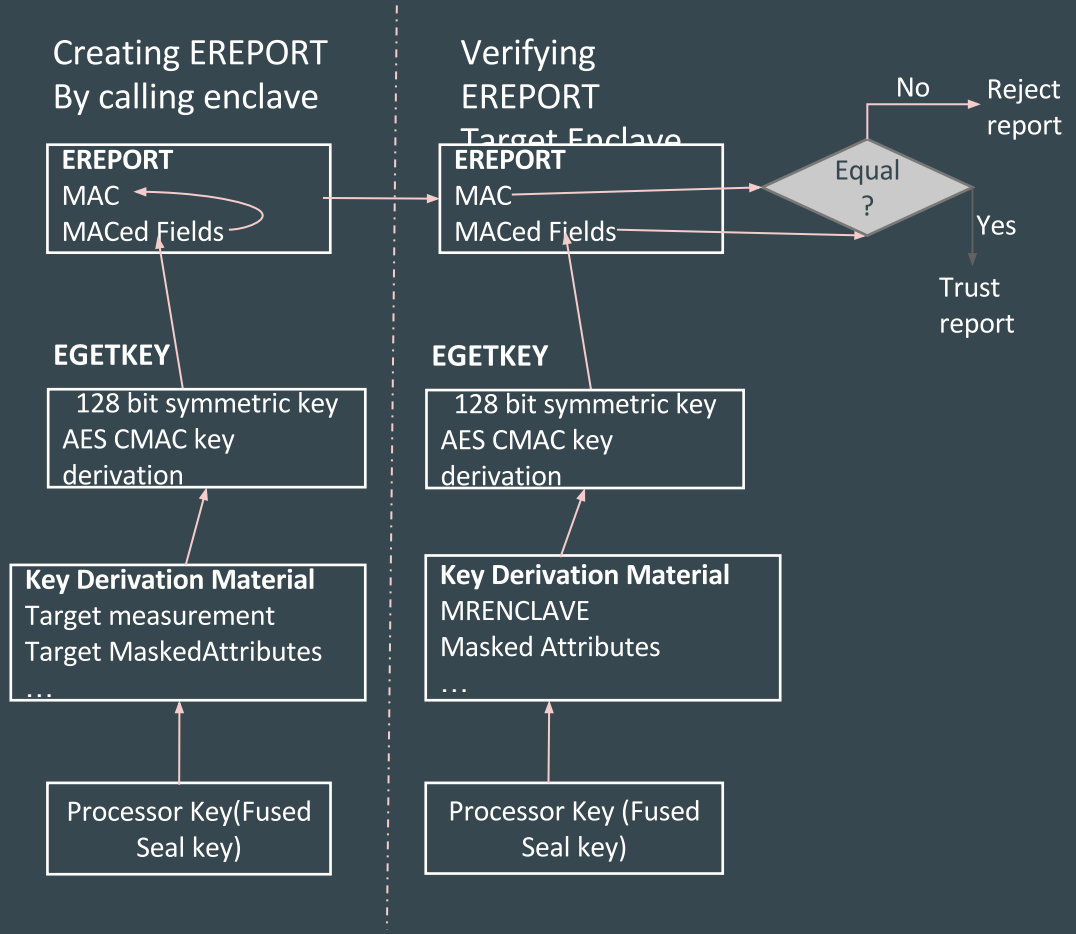
Praveen Keshavamurthy
Aboobacker Rizwan

Introduction

- Provides Trusted Execution Environment by reducing the level of trust involved.
- Enclaves in Processor Reserved Memory
- Enclave Page Cache (EPC)
- Executes in ring 3
- Enclave measurements (MRENCLAVE)
  - Secure hash over inputs to ECREATE EADD EXTEND
- Remote attestation for authenticating an enclave based on its measurement.

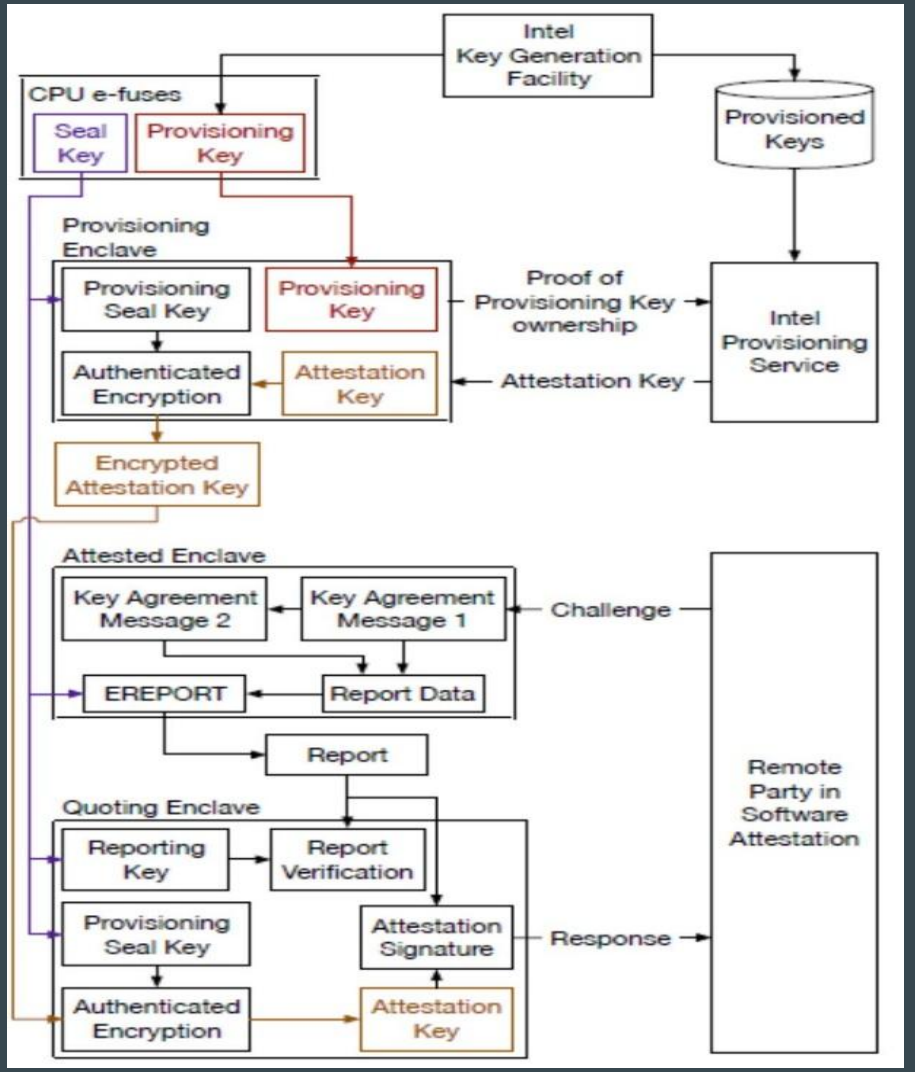
Attestation

- **Local Attestation**
- Secure communication between enclaves
- EREPORT require virtual address of Report Target Info structure



Remote Attestation

- Fused seal key only used by EGETKEY
- Provisioning Enclave identified by PROVISIONKEY aatribute



SGX Application Development Guidelines...

- Main mode of shipment of SGX binary is signed dll format.
- Author should use 3072-bit RSA key with public exponent set to 3 for signing.
- The SGX DLL should not depend on any runtime dlls. It can be linked to static SGX library, if required.
- Max run time memory reserved for SGX enclaves is 128 MB. This brings in a restriction on number of active encalves at any point of time in the system.
- Max heap size per enclave is limited to 96MB. This brings in a limitation on the dynamic memory allocation to the enclaves.
- SGX is not an end to end secure solution. The design should be such that only the secure data processing code should be part of enclave and application design should have a secure architecture as part of non-enclave code.

Insight to Intel SGX SDK Libraries

sgx_t* - trusted libraries; sgx_u*- untrusted libraries

sgx_trts.lib - Intel® SGX internals - rand functions, dynamic enclave checks, instruction exception handlers

sgx_tstd.lib; sgx_tstdcxx.lib - standard C & C++ functionalities including STL support

sgx_tcrypto_opt.lib - Crypto functionality - AES-GCM, DH, ECC

sgx_tkey_exchange.lib

sgx_tservice.lib - Uses crypto and key exchange libraries. Supports APIs for data seal/unseal, trusted Architectural enclaves support, Elliptic Curve Diffie-Hellman (EC DH) library

Continued..

sgx_urts.lib - Provides functionality for applications to manage enclaves

sgx_uae_service.lib - Provides both enclaves and untrusted applications access to services provided by the AEs

sgx_ukey_exchange.lib - Untrusted key exchange library

sgx_status.dll - Provides functionality for applications to register Enclave Signing Key White List Certificate Chain.

sgx_capable.dll - Provides functionality for applications to check if the client platform is enabled for Intel SGX or to enable the Intel SGX device.

SGX Use cases explored..

1. Cloud use cases - Should be more to do with the frameworks
    a. Adapt Cloud Infrastructures to support Intel SGX
        i. Openstack (python), ebbRT library (Linux)
    b. Explored different DB's as these will be the building blocks for any application development using SGX in future
        i. NoSQL Database
            1. In-memory DB - Not applicable as the reserved memory limit is 128 MB for Intel SGX
            2. Graph DB - Neo4J (Java)
        ii. SQL Database
            1. SQLite - Looks feasible as this is a lightweight library. There is an extension to this called SQLCipher. we can build it in windows and use it for our implementation
    c. TOR Network - Not a cloud use case. But can be tested on real Intel SGX
    d. Data Center Applications:
        i. Distributed Large Data Graph Processing
            1. Apache Giraph (Java Framework/ Linux)
            2. GraphLab PowerLab (C++ Framework/ Linux)
    e. KVM Support for Intel SGX

SQLiteCipher

- AES-CBC 256 bit encryption on each page - Page size defaults to 1 KB
- IV is stored at end of page & HMAC-SHA1 of IV and Ciphertext is stored after IV at the end of the page
- Unique random salt is stored at the start of the db file which uniquely identifies the db key though the key is same
- Key data is generated using passphrase, PBKDF2 and random salt
- All the crypto logic are implemented using OpenSSL Library.
- The build environment uses MinGW.

Steps to Adapt SQLiteCipher to use Intel SGX

- Identify the equivalent SGX crypto Library APIs to implement the encryption logic of SQLiteCipher.
- How do you maintain the cipher context inside enclave ??
- Move the data processing logic , encryption and decryption logic to Enclave library. Generated signed dll of the same using sgx_edge8r tool and sgx_signer tool.
- Use this generated dll to link to the other part of the SQLiteCipher non enclave dll.

DB Benchmark Measure

- Total No. of Single-record read/write in a given interval (ops/sec)
- Total No. of Bulk read/write in a given interval (ops/sec)
- DB Startup Time - **Not Applicable**
- Load on CPU and Disk while executing Complex query and Simple query.
- Transaction Processing Council Benchmarks - Concurrency and Atomic Commits

Other Links..

SGX Use cases https://docs.google.com/a/husky.neu.edu/presentation/d/15wt5_duoowKsySTpt842FLJh3TJq07eJ7DvxqCjf7N4/edit?usp=sharing

SGX Architecture

https://docs.google.com/a/husky.neu.edu/presentation/d/11k4JCWPh2LFxV5Gp_iQeFFoKIhOSEaGBzk3mhqat6cI/edit?usp=sharing

Github Repo for reference papers

https://github.com/praveenkmurthy/Projects/tree/master/Intel%20SGX