SGX学习一周总结

> 2021/2/7

本周已完成任务

- 安装sgx运行环境
- 运行样例代码（模拟&硬件模式）
- 学习sgx101的课程
- 学习vim

将来的任务

- 创建enclave
- ecall&ocall
- local&remote
- load&read

SGX学习一周总结

> 2021/2/14

本周已完成任务

- 创建enclave
- 通过调用ecall 和 ocall来打印“hello world” (模拟模式)

这周我发现上周的硬件模式并没有跑通，只是因为我在编译硬件模式前没有执行[make clean] 命令，导致程序一直运行在模拟模式下。

排查之后发现是sgx驱动没装好，但在多次尝试之后，即使驱动程序显示成功安装，系统中也依然找不到相应的硬件，仍处于未安装驱动的状态。

所以本周任务是在模拟模式下完成的

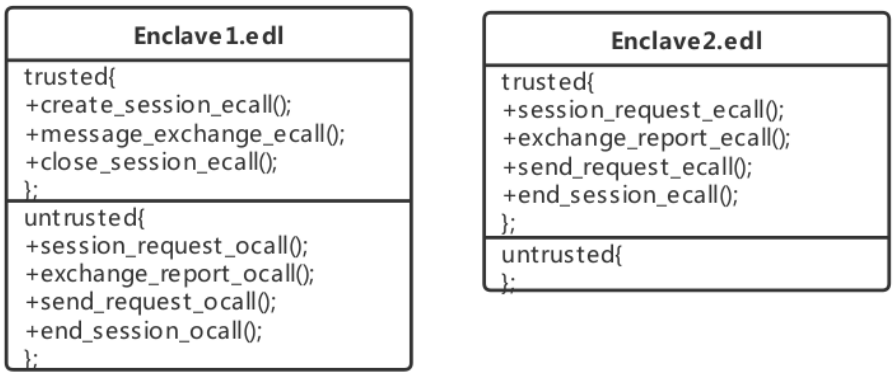
下周计划任务

- 实现local&remote attestation
- 实现sealing

SGX学习一周总结

> 2021/2/21

本周已完成任务

- Local attestation
  - 

- message exchange

Local Attestation总结

process

1. create Encalve1 and Enclave2
2. create session (Diffie-Hellman Key Exchange)
    1. Enclave1 send a session request to Enclave2 (Session Request)
        1. sgx_dh_init_session();
        2. session_request_ocall();
    2. Enclave2 process request and send msg1 to Enclave1
        1. sgx_dh_init_session();
        2. sgx_dh_responder_gen_msg1();
        3. return msg3;
    3. Enclave1 process msg1 and send msg2 to Enclave2 (Exchange Report)
        1. sgx_dh_initiator_proc_msg1();
        2. exchange_report_ocall();
    4. Enclave2 process msg2 and send msg3 to Enclave2
        1. sgx_dh_responder_gen_msg2();
        2. return msg3;
    5. Enclave1 process msg3 and the session created
        1. sgx_dh_initiator_proc_msg3();
3. message exchange
    1. Enclave1 send message exchange request to Enclave2
        1. send_request_ecall();
        2. sgx_rijndael128GCM_decrypt();
    2. Enclave2 send response to Enclave1
        1. generate_response_ecall();
        2. sgx_rijndael128GCM_decrypt();
4. close session
    1. Enclave1 send close session request
        1. end_session_ocall();

    2. Enclave2 close session and return
        1. end_session_ecall();
    3. Enclave1 close session
        1. close_session_ecall();
5. destroy Enclave1 and Enclave2

下周计划任务

- 申请EPID?
- 实现remote attestation
- 实现enclave在硬盘上的load和read