C -files

- host/demolib.c
  - create_enclave_bytes
  - terminate_enclave
- enclave_a/ecalls.c
  - get_remote_report_with_pubkey
  - verify_report_and_set_pubkey

- common/attestation.cpp
  - seal_bytes
  - unseal_bytes
  - Plus implementation of get_remote..., verify_report...
- Common/crypto.cpp
  - Encrypt, decrypt
  - test_decrypt (to communicate with python-crypto)
  - are mostly mbedtls - wrappers

Python files

- Example1.py
  - Shows how to use lib.py to create and talk to enclaves

- Simple.py
- Simple coap server and client to simulate the TEEP agent
- Simple.test is an example that creates and attests a remote enclave and sends and receives data via"seal_bytes" and "unseal_bytes"

Usage

- make
- python -c 'import simple; simple.start_server()'&
- python -c 'import simple; simple.sealingtest()'

TODO

- The code uses CBOR but not the exact tags and structure of the TEEP messages
- The TEEP agent (simple.server) doesn’t do any authentication or ocsp checks, but there is ocsp code functions that are half-way through

- I focused mostly on getting something that was easy to tailor to a use-case for the project, I.e. processing medical journals in an enclave, and not on getting the exact format of the CBOR records, which is trivial in TEEP.