Implementation of Trusted Execution Environment Provisioning (TEEP) Protocol

TEEP-Device

User's Manual

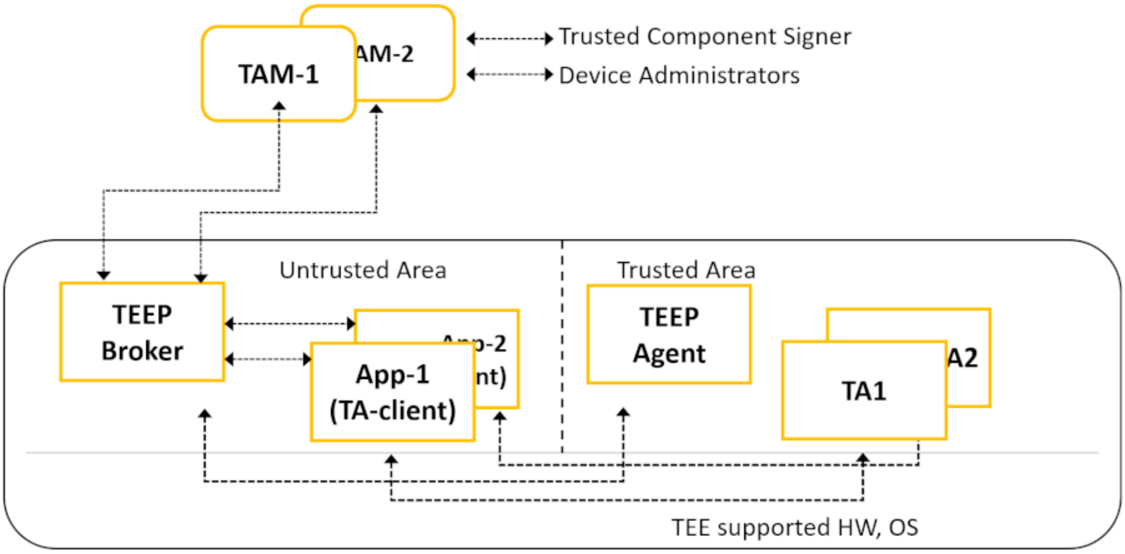
2024-12-03

1 Overview of TEEP-Device

There is a wiki page describing Introduction, objective and use cases of TEEP Protocol.

- https://github.com/ietf-teep/teep-protocol/wiki

The TEEP Protocol provides the protocol on a wide range of devices for install, update, and delete Trusted Applications and Personalization Data by Trusted Component Signer or Device Administrators who host Trusted Application Managers (TAMs).



The TEEP-Device is an implementation for defining the draft of Trusted Execution Environment Provisioning (TEEP) Protocol at the Internet Engineering Task Force (IETF). The chart above is a simplified diagram of components described in the TEEP Protocol and TEEP Architecture drafts. The TEEP Protocol on the TEEP-Device uses HTTP packets defined by HTTP Transport for Trusted Execution Environment Provisioning.

Following are the explanations of each component on the above diagram.

Trusted Application (TA): An application that runs in a TEE.

Trusted Application Manager (TAM): An entity that manages Trusted Applications and other Trusted Components running in TEEs of various devices.

TEEP Broker: A TEEP Broker is an application component running in a Rich Execution Environment (REE) that enables the message protocol exchange between a TAM and a TEE in a device. A TEEP Broker does not process messages on behalf of a TEE, but merely is responsible for relaying messages from the TAM to the TEE, and for returning the TEE's responses to the TAM.

TEEP Agent: The TEEP Agent is a processing module running inside a TEE that receives TAM requests (typically relayed via a TEEP Broker that runs in an REE). A TEEP Agent in the TEE may parse requests or forward requests to other processing modules in a TEE, which is up to a TEE provider's implementation.

The terminology of Trusted Application (TA) in the old draft was revised to Trusted Component (TC) to express the files installed from TAM to devices that could have both binaries of trusted applications and data files of personalization data. The TA and TC are interchangeable in this documentation.

The TEEP Protocol relies on the Software Updates for Internet of Things (SUIT) Manifest defined at IETF which expresses metadata of the TC. When the TAM sends one of the TEEP Messages called Update Message, it will include SUIT Manifest for installing TC to the Device.

The TEEP Protocol and SUIT Manifest are both in binary formats of Concise Binary Object Representation (CBOR). The Objective of CBOR is designed to reduce the message size as much as possible in the way of light way parsing which is appropriate for Constraint Hardware such as IoT, Edge and Embedded devices with limited CPU processing power and memory size. The CBOR has a similar representation of JSON that is widely used on the Internet and makes constructing CBOR easier to use.

The authentication feature of CBOR binaries in TEEP Messages and SUIT Manifests is proved by CBOR Object Signing and Encryption (COSE) required by TAM, Device, Developer and Owner of TC and etc. The COSE defined the method of Signing and Encryption of CBOR formats.

More details can be found in the URLs at iETF.

- TEEP Protocol
  - https://datatracker.ietf.org/doc/html/draft-ietf-teep-protocol
- HTTP Transport for Trusted Execution Environment Provisioning
  - https://datatracker.ietf.↩
    org/doc/html/draft-ietf-teep-otrp-over-http
- TEEP Architecture:
  - https://datatracker.ietf.org/doc/draft-ietf-teep-architecture/
- SUIT Manifest:
  - https://datatracker.ietf.org/doc/draft-ietf-suit-manifest/
- CBOR
  - https://datatracker.ietf.org/doc/rfc8949/
- COSE
  - https://datatracker.ietf.org/doc/rfc8152/

1.1 Use Cases of TEEP

Typical use cases for TEEP Protocol is a firmware update Over The Air (OTA) which TC containing a firmware binary, installing security sensitive applications used for payment, playing video with DRM, insurance software, enabling hardware feature with license keys, telemetry software, and softwares handles personal identification data, such as Social Security Number, and vaccination status.

1.2 Features of TEEP-Device

- The TEEP Protocol defines the protocol format and interaction between the server called Trusted Application Managers (TAM) and the IoT/Edge devices. The TEEP-Device is an implementation of Trusted Execution Environment Provisioning (TEEP) Protocol on the IoT/Edge devices.
- The TEEP-Device provides the requirements of the TEEP-Broker and TEEP-Agent in IETF drafts.
- Uses the tamproto as an implementation of the TAM server.
  - https://github.com/ko-isobe/tamproto
- Provides initiating the protocol from TEEP-Device, downloading a Trusted Components (TC) called Hello-↩ TEEP-TA as a sample of TC from the TAM, installing it inside TEE, and executing the Hello-TEEP-TA.
- Implemented on top of TA-Ref which provides a portable TEE programming environment among different TEEs on Intel CPU, ARM Cortex-A and RISC-V 64G to provide uniform source codes over OP-TEE on ARM-TrustZone for Cortex-A series and Keystone on RISC-V.
- The required features of TEEP-Agent in the draft is implemented as a application in user application privilege level inside TEE in this TEEP-Device to simplify the implementation which ideally should be combined with implementation in higher privilege levels, such as, the runtime in S-Mode and Secure Monitor in M-mode on RISC-V. Therefore, some of the assumed requirements on the draft are not fulfilled with the TEEP-Device. In the product, the features of TEEP-Agent must be enabled through root-of-trust from the boot up of the CPUs, the TCs must be saved in a secure manner and have protection of installed TCs.
- Supports Concise Binary Object Representation (CBOR) for current four TEEP Messages.
  - https://datatracker.ietf.org/doc/html/rfc7049
- Supports SUIT Manifest inside the Update message of TEEP Protocol.
  - https://datatracker.ietf.org/doc/draft-ietf-suit-manifest/

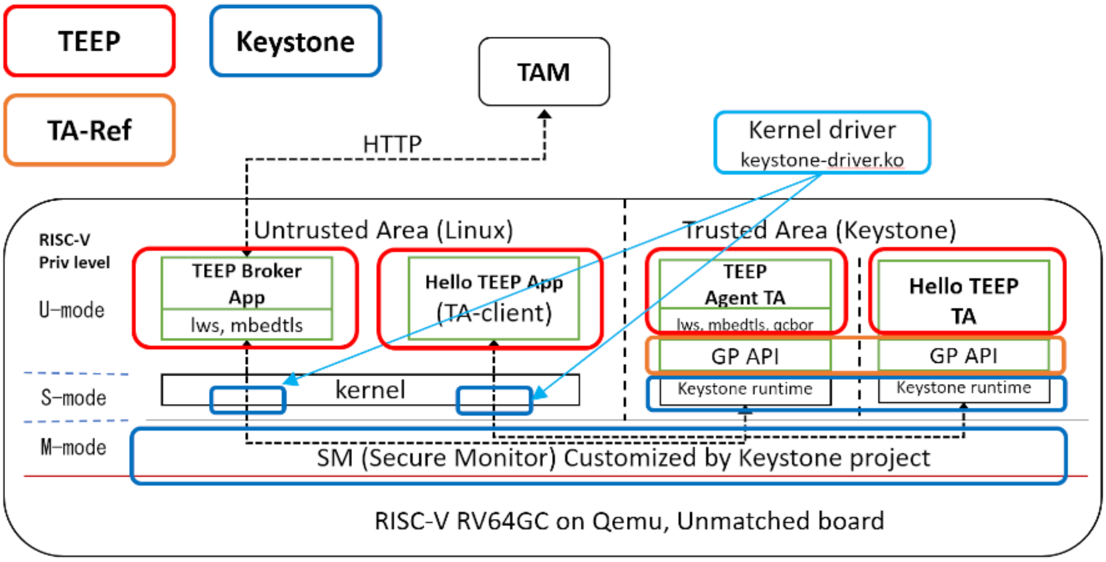
1.3 Components of TEEP-Device and TA-Ref

The Trusted Application Reference (TA-Ref) is a different software stack from this TEEP-Devie. The TA-Ref provides a portable API and SDK among Intel SGX, ARM TrustZone-A and RISC-V Keystone and enables portability for source codes of Trusted Applications among different CPUs.

The API of TA-Ref is a subset of TEE Internal Core API Specification defined by Global Platform.

- https://globalplatform.↩ org/specs-library/tee-internal-core-api-specification/

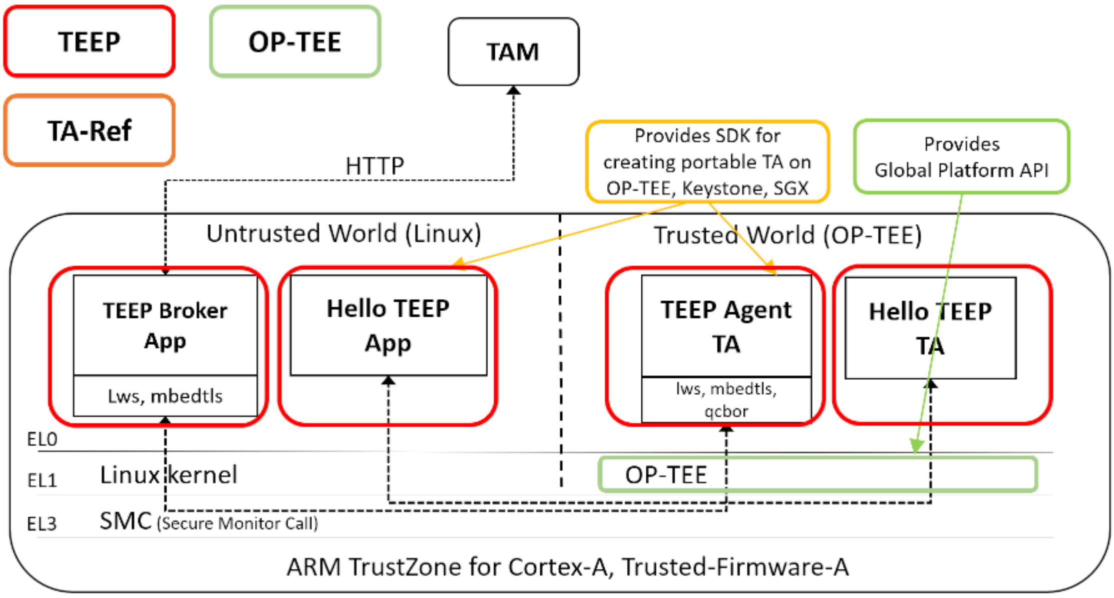
1.3.1 TEEP-Device and TA-Ref Components on Keystone



The TEEP-Device is implemented on top of the TA-Ref with TEE provided by the Keystone project on RISC-V RV64GC CPU. Each TA in the Trusted Area is protected with Physical memory protection (PMP) which is enabled by RISC-V hardware.

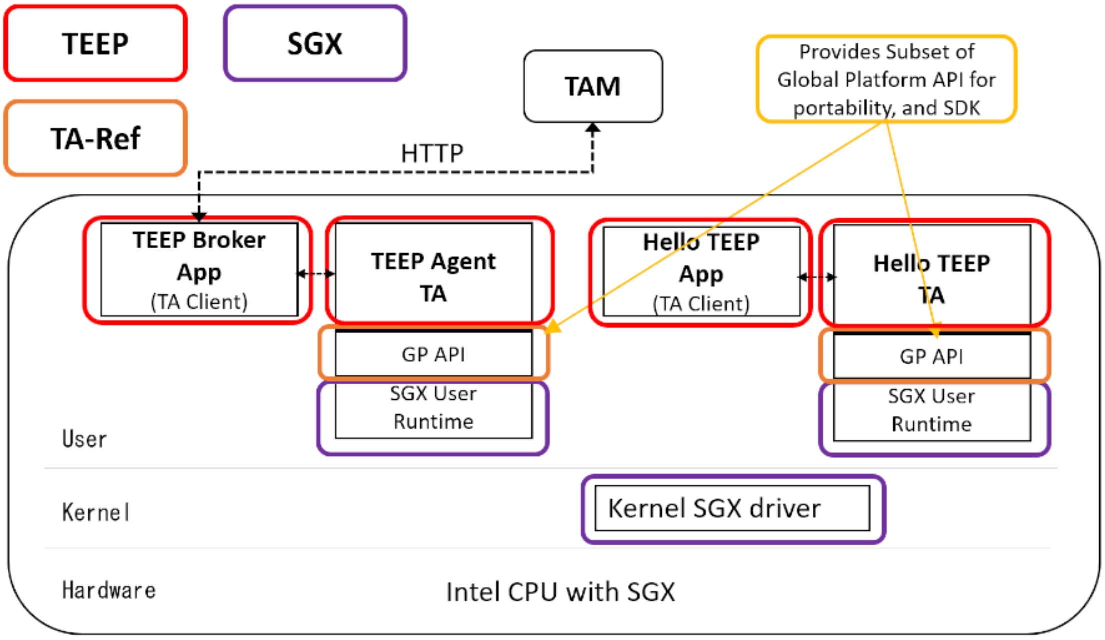
- Keystone project
  - https://keystone-enclave.org/

1.3.2 TEEP-Device and TA-Ref Components on OP-TEE



It is on OP-TEE but highly utilizes the programming environment provided by TA-Ref to simplify the TEEP-Device to be able to build and function on other CPUs with the single source code of TEEP-Agent and Hello-TEEP-TA. They both are using the subset of Global Platform API.

1.3.3 TEEP-Device and TA-Ref Components on SGX



The diagram is the ideal implementation of TEEP-Device on SGX. The current TEEP-Device is not utilizing SGX libraries and the SGX enabled CPU which provides SGX capability with SGX SDK. The TEEP-Device is built and executed as a regular user space application at the moment, and enabling the SGX capability is a future activity.

1.4 The design of TEEP Agent and sample TA of HELLO-TEEP-TA

All three TEEs (Kestone/OP-TEE/SGX) share the same design which requires

- Binaries of TA
- Client Application to execute the TA.

The features of TEEP-Agent described in the TEEP Architecture draft are realized as a TA (TEEP-Agent-TA, filename: teep-agent-ta) in this TEEP-Device implementation. The Client App which starts executing the TEEP-Agent-TA is the TEEP-Broker-App (filename: teep-broker-app).

The features of TEEP-Agent do not have to be implemented as a TA in TEEP Architecture draft, ideally it is recommended the features of TEEP-Agent to be implemented inside TEE-os or underlying privileged mode (SMC in OP-TEE and SM in Keystone) to improve preventing from malicious softwares to stop the feature of the TEEP-Agent. The reason for realizing the TEEP-Agent as a TA in the TEEP-Device design is to simplify the implementation only.

Purpose	Client App	TA Binary
TEEP	teep-broker-app	teep-agent-ta (teep-agent-ta.so for only sgx)

The HELLO-TEEP-TA is a sample TA application which prints "Hello TEEP from TEE!". This sample TA is executed after the successful build of target client applications (App-keystone / App-optee / App-sgx). The HELLO-TEEP-TA itself is not part of the TEEP Architecture design. It is for purely demonstration and debugging purposes of the TEE-Broker-App, TEEP-Agent-TA and the TAM.

In the real scenario, the HELLO-TEEP-TA will have features of Payment Application, DRM for video playback, OTA capability, serial code of enhancing features of the hardware or stopping the entire activity of the device when the drones or power plants fall into the unintended parties.

Following are the pairs needed for execution on all three TEEs.

Purpose	Client App	TA Binary
Keystone	App-keystone	8d82573a-926d-4754-9353-32dc29997f74.ta
OP-TEE	App-optee	8d82573a-926d-4754-9353-32dc29997f74.ta
Keystone	App-sgx	8d82573a-926d-4754-9353-32dc29997f74.ta

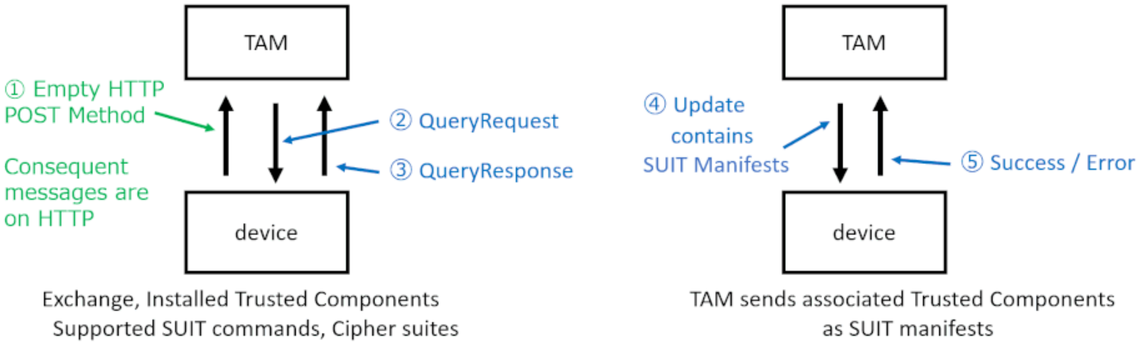
Executing the ./teep-broker-app with tamproto URL will allow the teep-agent-ta to talk with the tamproto server and download the TA Binary. Once the TA Binary is downloaded, the Client App (App-keystone / App-optee / App-sgx) will run the downloaded TA Binary.

The filename of downloaded TA Binary is: 8d82573a-926d-4754-9353-32dc29997f74.ta for all the targets.

2 Operation of TAM and device

Four TEEP Messages

- QueryRequest Message
- QueryResponse Message
- Update Message <- contains SUIT manifest
- Success Message / Error Message



Exchange, Installed Trusted Components
Supported SUIT commands, Cipher suites

TAM sends associated Trusted Components
as SUIT manifests

TEEP Protocol defines four messages, QueryRequest, QueryResponse, Update, and Success-Error.

The device initiates the first message by sending an empty HTTP POST Method macket to the TAM server, and then the TAM will be sending the QueryRequest to ask the capability of the device. All the consequent TEEP messages after the first empty HTTP POST are carried over HTTP Request/Response. The return message of the QueryResponse contains the information of supported cryptographic algorithms, installed TCs, etc on the device.

If the TAM decides the TC must be installed to the device or update the previously installed TC, then the TAM will send the Update message and the device will process it. The Update message includes software Updates for Internet of Things (SUIT) Manifest which could contain the TC in the body of the Update message or have URI pointing to the location of the TC hosted elsewhere. The result of processing the Update message in the device is reported to the TAM with a Success-Error message.

The SUIT Manifest provides metadata of TC including dependency of the TC, procedure of installing, method of verifying signature and information of invoking it. The SUIT Manifest is designed to meet the requirements of constrained devices with limited capacity of CPU and memory size of IoT and/or embedded devices.

All the messages are transmitted over HTTP packets in the current implementation. The type of transport layer in the drafts is not limited to HTTP, may use HTTPS or any other method.

3 Concise Binary Object Representation (CBOR) in TEEP-Device

3.1 Three format representations in TEEP and SUIT

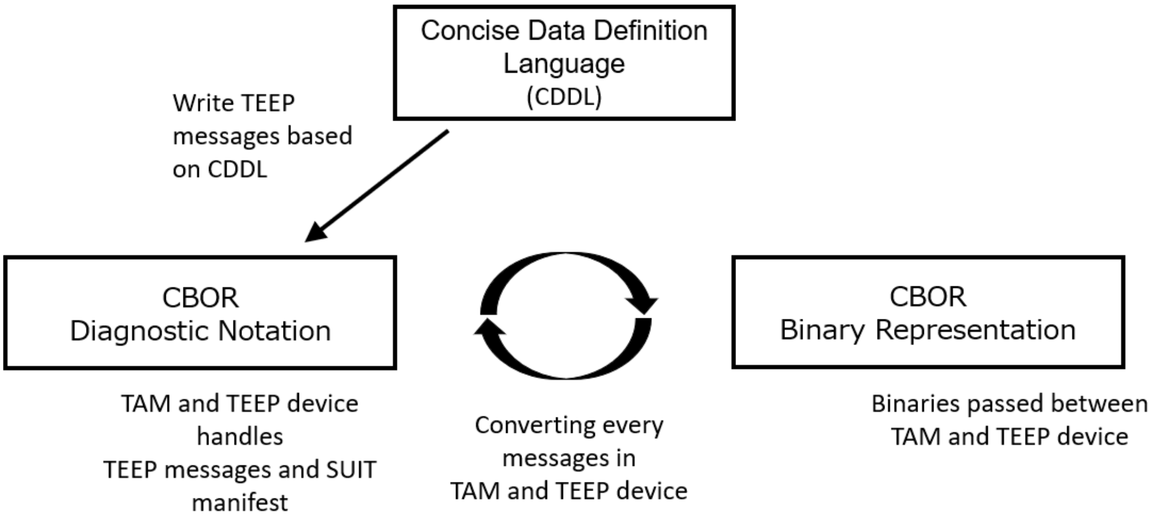
TEEP Messages use CBOR binary format contrary to text based JavaScript Object Notation (JSON). The JSON format is widely used on the modern Internet as an easy of use message exchanging format between servers and clients.

The CBOR provides binary format which enables smaller packet size to carry over the Internet as well as light weight parsing for the IoT and embedded devices.

The CBOR has three representations. The Concise Data Definition Language (CDDL), Diagnostic Notation and Binary Representation.

The CDDL is used for defining CBOR syntax of the TEEP Messages. The CBOR Diagnostic Notation is a way of expression of actuarial packet format of TEEP Messages in CBOR syntax. The CBOR Diagnostic Notation is similar text description to JSON format which are almost interchangeable.

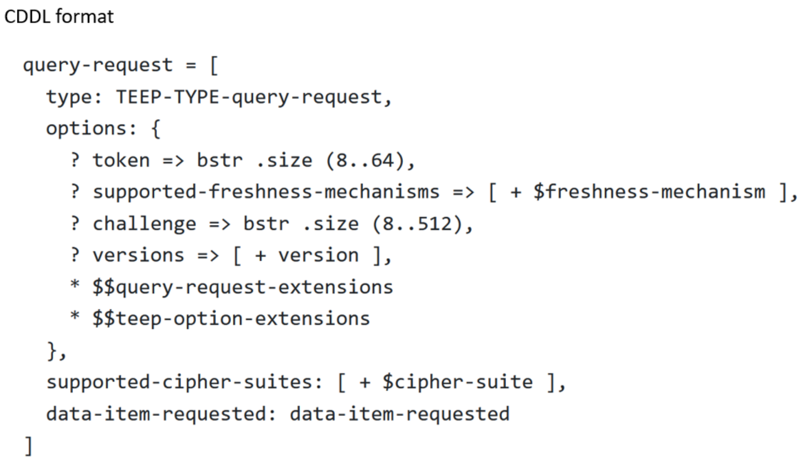
The CBOR Binary Representation is the result of converting the CBOR Diagnostic Notation to the binary. The TAM server and TEEP-Device parse the Binary Representation and handles the TEEP protocol operation.


Concise Data Definition Language (CDDL)
Write TEEP messages based on CDDL
CBOR Diagnostic Notation
CBOR Binary Representation
TAM and TEEP device handles TEEP messages and SUIT manifest
Converting every messages in TAM and TEEP device
Binaries passed between TAM and TEEP device

3.2 TEEP Message format

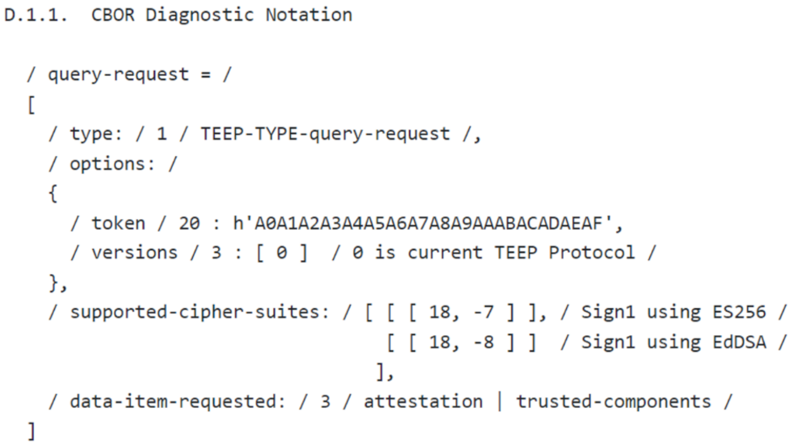
Example of the TEEP Message of QueryRequest in both CBOR Diagnostic Notation and Binary Representation. The Diagnostic Notation is expressed in similar text style with JSON when implementing the Query Request by reading the CDDL format.

The TEEP-Device and TAM will exchange the Binary Representation only. When the TEEP-Device receives the QueryRequest message from the TAM, the TEEP-Device will parse the Binary Representation to the Diagnostic Notation for understanding the contents of the message.

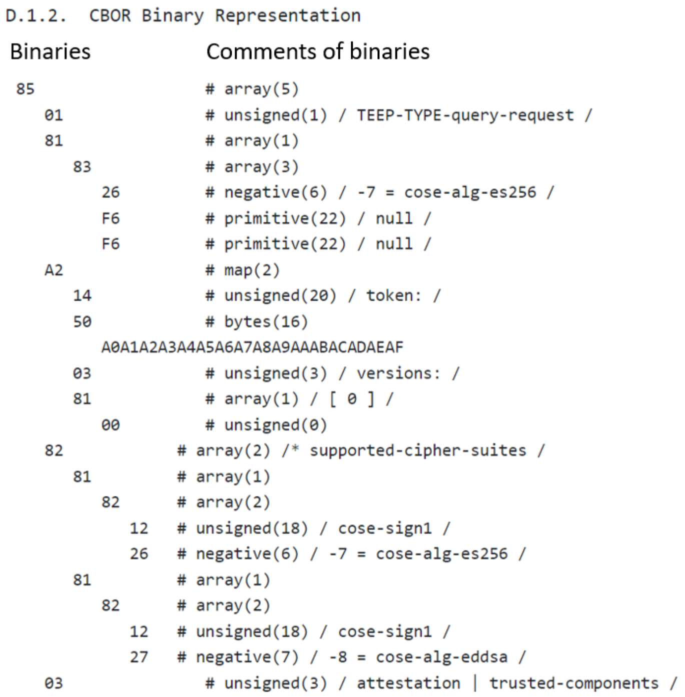


CDDL format

```
query-request = [
  type: TEEP-TYPE-query-request,
  options: {
    ? token => bstr .size (8..64),
    ? supported-freshness-mechanisms => [ + $freshness-mechanism ],
    ? challenge => bstr .size (8..512),
    ? versions => [ + version ],
    * $$query-request-extensions
    * $$teep-option-extensions
  },
  supported-cipher-suites: [ + $cipher-suite ],
  data-item-requested: data-item-requested
]
```



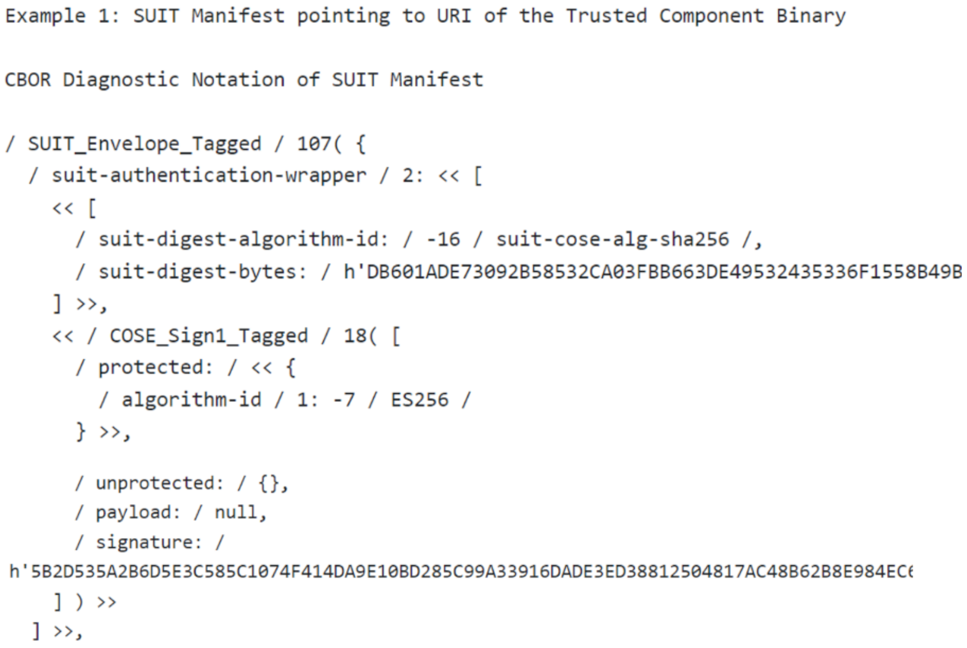
```
D.1.1.  CBOR Diagnostic Notation

  / query-request = /
  [
    / type: / 1 / TEEP-TYPE-query-request /,
    / options: /
    {
      / token / 20 : h'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF',
      / versions / 3 : [ 0 ]  / 0 is current TEEP Protocol /
    },
    / supported-cipher-suites: / [ [ [ 18, -7 ] ], / Sign1 using ES256 /
                                 [ [ 18, -8 ] ]  / Sign1 using EdDSA /
                               ],
    / data-item-requested: / 3 / attestation | trusted-components /
  ]
```



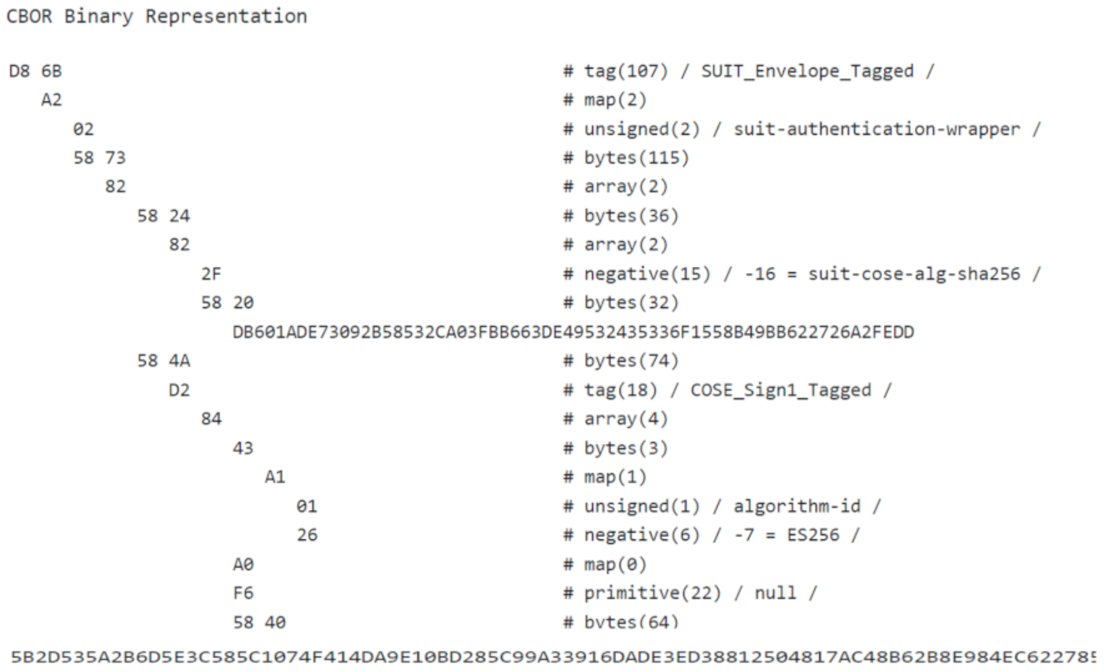
```
D.1.2.  CBOR Binary Representation
Binaries              Comments of binaries
85                    # array(5)
   01                 # unsigned(1) / TEEP-TYPE-query-request /
   81                 # array(1)
      83              # array(3)
         26           # negative(6) / -7 = cose-alg-es256 /
         F6           # primitive(22) / null /
         F6           # primitive(22) / null /
   A2                 # map(2)
      14              # unsigned(20) / token: /
      50              # bytes(16)
         A0A1A2A3A4A5A6A7A8A9AAABACADAEAF
      03              # unsigned(3) / versions: /
      81              # array(1) / [ 0 ] /
         00           # unsigned(0)
   82              # array(2) /* supported-cipher-suites /
      81           # array(1)
         82        # array(2)
            12     # unsigned(18) / cose-sign1 /
            26     # negative(6) / -7 = cose-alg-es256 /
      81           # array(1)
         82        # array(2)
            12     # unsigned(18) / cose-sign1 /
            27     # negative(7) / -8 = cose-alg-eddsa /
   03                 # unsigned(3) / attestation | trusted-components /
```

3.3 SUIT Manifest format

The Update message of TEEP protocol includes the SUIT Manifests for the information of TC and/or carrying the TC itself in the SUIT Manifest. These are the examples of SUIT Manifest in the Update message.



```
Example 1: SUIT Manifest pointing to URI of the Trusted Component Binary

CBOR Diagnostic Notation of SUIT Manifest

/ SUIT_Envelope_Tagged / 107( {
  / suit-authentication-wrapper / 2: << [
    << [
      / suit-digest-algorithm-id: / -16 / suit-cose-alg-sha256 /,
      / suit-digest-bytes: / h'DB601ADE73092B58532CA03FBB663DE49532435336F1558B49B
    ] >>,
    << / COSE_Sign1_Tagged / 18( [
      / protected: / << {
        / algorithm-id / 1: -7 / ES256 /
      } >>,

      / unprotected: / {},
      / payload: / null,
      / signature: /
h'5B2D535A2B6D5E3C585C1074F414DA9E10BD285C99A33916DADE3ED38812504817AC48B62B8E984EC
    ] ) >>
  ] >>,
```



```
CBOR Binary Representation

D8 6B                                           # tag(107) / SUIT_Envelope_Tagged /
   A2                                           # map(2)
      02                                        # unsigned(2) / suit-authentication-wrapper /
      58 73                                     # bytes(115)
         82                                     # array(2)
            58 24                               # bytes(36)
               82                               # array(2)
                  2F                            # negative(15) / -16 = suit-cose-alg-sha256 /
                  58 20                         # bytes(32)
                     DB601ADE73092B58532CA03FBB663DE49532435336F1558B49BB622726A2FEDD
            58 4A                               # bytes(74)
               D2                               # tag(18) / COSE_Sign1_Tagged /
                  84                            # array(4)
                     43                         # bytes(3)
                        A1                      # map(1)
                           01                   # unsigned(1) / algorithm-id /
                           26                   # negative(6) / -7 = ES256 /
                     A0                         # map(0)
                     F6                         # primitive(22) / null /
                     58 40                      # bytes(64)
5B2D535A2B6D5E3C585C1074F414DA9E10BD285C99A33916DADE3ED38812504817AC48B62B8E984EC622785
```

4 Directory structure of source files

Following are the important directories in the source code along with its description

Directory	Description
docs	Files for generating documentaions
hello-tc	Sample Trusted Application for TEEP Protocol
include	Header files to build hello-app/ta and teep-broker-app/teep-agent-ta
key	cryptograpic keys for TEEP protocol
lib	contains libraries used on TEEP-Device
submodules	Contains the submodules used for TEEP-Device
submodules/libwebsockets	HTTP/HTTPS library https://github.com/warmcat/libwebsockets
submodules/mbedtls	Cryptographic library https://github.com/ARMmbed/mbed-crypto
submodules/QCBOR	CBOR library https://github.com/laurencelundblade/↩ QCBOR.git
submodules/t_cose	COSE library https://github.com/laurencelundblade/t_↩ cose.git
teep-agent-ta	Main body of handling TEEP Protocol on TEE side
teep-broker-app	Main body of handling TEEP Protocol on Linux side

TEEP-Device Source Code

The below is the current TEEP-Device source code listing only the directories to one level.

```
.
 |-platform
 |   |-keystone
 |   |-pc
 |   |-sgx
 |   |-op-tee
```

```
|-docs
|  |-images
|  |-doxygen
|-key
|  |-CAs
|-lib
|  |-cose
|  |-suit
|  |-cbor
|  |-teep
|  |-log
|  |-include
|-hello-tc
|  |-manifest
|  |-build-optee
|  |-build-pc
|  |-build-keystone
|  |-build-sgx
|-sample
|  |-session
|-teep-broker-app
|  |-scripts
|-submodule
|  |-libwebsockets
|  |-t_cose
|  |-QCBOR
|  |-mbedtls
|  |-googletest
|-teep-agent-ta
|-include
```

5 Consideration of build machine and development environment

5.1 How to select the build machine

The development cycle of any software project has a loop of (∗) build the source and running the software to find the bug, (∗) write or revise the source to fix the bug.

Having as short as possible of the time for conducting the above loop will increase development productivity dramatically. Shorter loop time will be able to achieve more iteration in an hour. High number of iterations per time is the key to productivity.

In projects such as the TEEP-Device development, it is essential to have a fast machine as much as possible for building the sources and using multiple terminals for efficient development. Selection of a fast computer is mandatory because the speed of the build machine affects the efficiency of the development.

Oftenly developers using a laptop, are logging in to a fast remote computer and building the sources on the remote machine rather than on a local laptop.

The three key components of the fast-build machine are speed of CPU, speed of storage, and memory size. The laptop will be slower than the server or desktop machines because of the limited capability of the three key components.

The frequency of the CPU having above 3.8Ghz is ideal. The write speed of the storage has a significant impact on the build time. It is almost a must to use SSD than HDD and the SSD should have above 3000MB/s write speed which is only available with M.2 form factor with NVMe interface. The 32GB or higher memory size is recommended, since it will reduce disk swapping occurring when running out of memory, which significantly increases the build time. Please request reasonable development machines if you are working at a corporate or an organization.

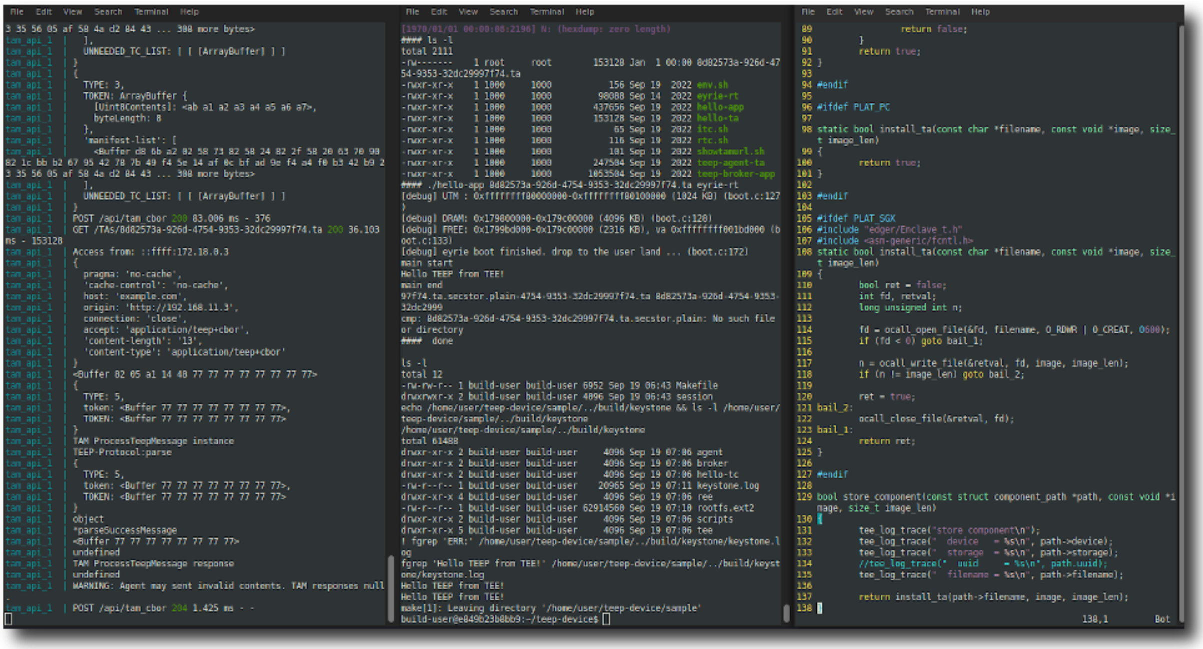
Some Examples of high-end machines: FUJITSU Server PRIMERGY RX2450 M1 with 2nd Gen AMD EPYC processor, FUJITSU Server PRIMERGY GX2460 M1 with 2nd Gen AMD EPYC processors etc.

5.2 How to setup an efficient development environment

To efficiently develop TEEP-Device source code, it is good to have three terminals.

- Terminal 1: To Run tamproto
- Terminal 2: To build and run the TEEP-Device.
- Terminal 3: To modify the TEEP-Device source code.

All three terminals opened simultaneously for efficiency. Logging in to the fast build machines mentioned in the previous chapter with the command `ssh -X user@Ip_address_build_machine` will provide forwarding X-Windows on your local machine.


File Edit View Search Terminal Help
File Edit View Search Terminal Help
File Edit View Search Terminal Help
138,1 Bot

```
3 35 56 05 af 58 4a d2 84 43 ... 308 more bytes>
tam_api_1  |    ],
tam_api_1  |    UNNEEDED_TC_LIST: [ [ [ArrayBuffer] ] ]
tam_api_1  |  }
tam_api_1  |  {
tam_api_1  |    TYPE: 3,
tam_api_1  |    TOKEN: ArrayBuffer {
tam_api_1  |      [Uint8Contents]: <ab a1 a2 a3 a4 a5 a6 a7>,
tam_api_1  |      byteLength: 8
tam_api_1  |    },
tam_api_1  |    'manifest-list': [
tam_api_1  |      <Buffer d8 6b a2 02 58 73 82 58 24 82 2f 58 20 63 70 98
82 1c bb b2 67 95 42 78 7b 49 f4 5e 14 af 0c bf ad 9e f4 a4 f0 b3 42 b9 2
3 35 56 05 af 58 4a d2 84 43 ... 308 more bytes>
tam_api_1  |    ],
tam_api_1  |    UNNEEDED_TC_LIST: [ [ [ArrayBuffer] ] ]
tam_api_1  |  }
tam_api_1  | POST /api/tam_cbor 200 83.006 ms - 376
tam_api_1  | GET /TAs/8d82573a-926d-4754-9353-32dc29997f74.ta 200 36.103
ms - 153128
tam_api_1  | Access from: ::ffff:172.18.0.3
tam_api_1  | {
tam_api_1  |   pragma: 'no-cache',
tam_api_1  |   'cache-control': 'no-cache',
tam_api_1  |   host: 'example.com',
tam_api_1  |   origin: 'http://192.168.11.3',
tam_api_1  |   connection: 'close',
tam_api_1  |   accept: 'application/teep+cbor',
tam_api_1  |   'content-length': '13',
tam_api_1  |   'content-type': 'application/teep+cbor'
tam_api_1  | }
tam_api_1  | <Buffer 82 05 a1 14 48 77 77 77 77 77 77 77 77>
tam_api_1  | {
tam_api_1  |   TYPE: 5,
tam_api_1  |   token: <Buffer 77 77 77 77 77 77 77 77>,
tam_api_1  |   TOKEN: <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | }
tam_api_1  | TAM ProcessTeepMessage instance
tam_api_1  | TEEP-Protocol:parse
tam_api_1  | {
tam_api_1  |   TYPE: 5,
tam_api_1  |   token: <Buffer 77 77 77 77 77 77 77 77>,
tam_api_1  |   TOKEN: <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | }
tam_api_1  | object
tam_api_1  | *parseSuccessMessage
tam_api_1  | <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | undefined
tam_api_1  | TAM ProcessTeepMessage response
tam_api_1  | undefined
tam_api_1  | WARNING: Agent may sent invalid contents. TAM responses null
.
tam_api_1  | POST /api/tam_cbor 204 1.425 ms - -
```

```
[1970/01/01 00:00:08:2196] N: (hexdump: zero length)
#### ls -l
total 2111
-rw-------    1 root     root        153128 Jan  1 00:00 8d82573a-926d-47
54-9353-32dc29997f74.ta
-rwxr-xr-x    1 1000     1000           156 Sep 19  2022 env.sh
-rwxr-xr-x    1 1000     1000         98088 Sep 14  2022 eyrie-rt
-rwxr-xr-x    1 1000     1000        437656 Sep 19  2022 hello-app
-rwxr-xr-x    1 1000     1000        153128 Sep 19  2022 hello-ta
-rwxr-xr-x    1 1000     1000            65 Sep 19  2022 itc.sh
-rwxr-xr-x    1 1000     1000           116 Sep 19  2022 rtc.sh
-rwxr-xr-x    1 1000     1000           181 Sep 19  2022 showtamurl.sh
-rwxr-xr-x    1 1000     1000        247504 Sep 19  2022 teep-agent-ta
-rwxr-xr-x    1 1000     1000       1053504 Sep 19  2022 teep-broker-app
#### ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127
)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (b
oot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
main start
Hello TEEP from TEE!
main end
97f74.ta.secstor.plain-4754-9353-32dc29997f74.ta 8d82573a-926d-4754-9353-
32dc2999
cmp: 8d82573a-926d-4754-9353-32dc29997f74.ta.secstor.plain: No such file
or directory
####  done

ls -l
total 12
-rw-rw-r-- 1 build-user build-user 6952 Sep 19 06:43 Makefile
drwxrwxr-x 2 build-user build-user 4096 Sep 19 06:43 session
echo /home/user/teep-device/sample/../build/keystone && ls -l /home/user/
teep-device/sample/../build/keystone
/home/user/teep-device/sample/../build/keystone
total 61488
drwxr-xr-x 2 build-user build-user     4096 Sep 19 07:06 agent
drwxr-xr-x 2 build-user build-user     4096 Sep 19 07:06 broker
drwxr-xr-x 2 build-user build-user     4096 Sep 19 07:06 hello-tc
-rw-r--r-- 1 build-user build-user    20965 Sep 19 07:11 keystone.log
drwxr-xr-x 4 build-user build-user     4096 Sep 19 07:06 ree
-rw-r--r-- 1 build-user build-user 62914560 Sep 19 07:10 rootfs.ext2
drwxr-xr-x 2 build-user build-user     4096 Sep 19 07:06 scripts
drwxr-xr-x 5 build-user build-user     4096 Sep 19 07:06 tee
! fgrep 'ERR:' /home/user/teep-device/sample/../build/keystone/keystone.l
og
fgrep 'Hello TEEP from TEE!' /home/user/teep-device/sample/../build/keyst
one/keystone.log
Hello TEEP from TEE!
Hello TEEP from TEE!
make[1]: Leaving directory '/home/user/teep-device/sample'
build-user@e849b23b8bb9:~/teep-device$
```

```
                return false;
        }
        return true;
}

#endif

#ifdef PLAT_PC

static bool install_ta(const char *filename, const void *image, size_t image_len)
{
        return true;
}

#endif

#ifdef PLAT_SGX
#include "edger/Enclave_t.h"
#include <asm-generic/fcntl.h>
static bool install_ta(const char *filename, const void *image, size_t image_len)
{
        bool ret = false;
        int fd, retval;
        long unsigned int n;

        fd = ocall_open_file(&fd, filename, O_RDWR | O_CREAT, 0600);
        if (fd < 0) goto bail_1;

        n = ocall_write_file(&retval, fd, image, image_len);
        if (n != image_len) goto bail_2;

        ret = true;
bail_2:
        ocall_close_file(&retval, fd);
bail_1:
        return ret;
}

#endif

bool store_component(const struct component_path *path, const void *image, size_t image_len)
{
        tee_log_trace("store component\n");
        tee_log_trace("  device   = %s\n", path->device);
        tee_log_trace("  storage  = %s\n", path->storage);
        //tee_log_trace("  uuid     = %s\n", path.uuid);
        tee_log_trace("  filename = %s\n", path->filename);

        return install_ta(path->filename, image, image_len);
}
```

In the above image, you can see that terminal1 is running tamproto, terminal2 is used for building and running TEEP-Device to catch the build errors and runtime errors. The terminal3 is used for editing the source code for debugging the errors found on terminal 2 and terminal 1. At the terminal 3, use your favorite editor or IDE such as Visual Studio Code.

Every time you update the source code in terminal 3, you can rebuild the source code on terminal2 and run it and see the debug messages of TEEP-Device at terminal 2 and logs message of tamproto at terminal 1 to find the eros whiteout distracted from other terminals.

The 27 inch or larger external monitor would suit best to operate multiple terminals during developments in this usage. The selection of the resolution is also important, not too dense and not too sparse. The 2560 x 1440 is often ideal on a 27 inch monitor, and 4K 3840 x 2160 on a 32 inch. Having more text on one display increases readability of the source code, as long as the size of the character is not too small.

6 Build TEEP-Device with Docker

The benefit of Docker images is to provide the environment of building and developing TEEP-Device to reduce the overhead of preparing them individually.

The TEEP-Device requires TA-Ref which provides a unified SDK among different TEEs for three CPU architectures, Keystone for RISC-V, OP-TEE for Arm64 and SGX for Intel.

Without the prepared Docker images, the developer will be required to build a massing software stack of Keystone, OP-TEE and SGX and install them on his/her development machine which needs downloading large sizes of source codes, a long time for building them. Also it may result in every individual having a slightly different environment which makes it difficult to reproduce when encountering errors.

The Docker images provide an easy to prepare development environment for TEEP-Device.

6.1 Preparation for Docker

To build the TEEP-Device with Docker, it is required to install Docker on Ubuntu.

For the first time users of Docker, please have a look on `https://docs.docker.com/engine/`.

The following installation steps is for Ubuntu 20.04

6.1.1 Install Docker

```
$ sudo apt update
# Next, install a few prerequisite packages which let apt use packages over HTTPS:
$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
# Then add the GPG key for the official Docker repository to your system:
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
# Add the Docker repository to APT sources:
$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
# This will also update our package database with the Docker packages from the newly added repo.
# Make sure you are about to install from the Docker repo instead of the default Ubuntu repo:
$ apt-cache policy docker-ce
#Finally, install Docker
$ sudo apt install docker-ce docker-compose
```

6.1.2 Executing Docker without sudo

By default, the Docker command can only be run by the root user or by a user in the Docker group, which is automatically created during Docker's installation process. If you attempt to run the Docker command without prefixing it with sudo or without being in the Docker group, you will get an output like this:

```
docker: Cannot connect to the Docker daemon. Is the docker daemon running on this host?.
```

To avoid typing sudo whenever we run the docker command, add your username to the docker group.

```
$ sudo groupadd docker
$ sudo gpasswd -a $USER docker
# Logout and then log-in again to apply the changes to the group
```

After you logout and login, you can probably run the Docker command without `sudo`.

```
$ docker run hello-world
```

Login to the docker to be able to access docker images. Make sure you have an account on docker-hub. If not please create one on `dockerhub.com`.

```
$ docker login -u ${YOUR_USERNAME} -p ${YOUR_PASSWD}
```

6.1.3 Create a Docker network tamproto

A Docker network named tamproto is required when we run TEEP-Device. The local network is required to connect with tamproto service running locally.

```
$ docker network create tamproto_default
```

6.2 Docker Images with pre-built TEEP-Device

The following are the docker images that have pre-built binaries of TEEP-Device with TA-Ref. Since these images are already prepared and built already, you can start using it directly without building the TEEP-Device oneself.

Purpose	Docker image	Description
Keystone	aistcpsec/teep-dev:keystone	Has pre-built binaries of TEEP-Device with TA-Ref for RISC-V Keystone
OP-TEE	aistcpsec/teep-dev:optee	Has pre-built binaries of TEEP-Device with TA-Ref for ARM OP-TEE
Intel SGX	aistcpsec/teep-dev:sgx	Has pre-built binaries of TEEP-Device with TA-Ref for Intel SSX
tamproto	aistcpsec/teep-dev:tamproto	Used for running tamproto
Doxygen	aistcpsec/teep-dev:doxygen	Used for generating TEEP-Device documentation

6.3 Preparation to build TEEP-Device on Docker

6.3.1 List of Docker images to build TEEP-Device

It requires Docker images of TA-Ref for building the TEEP-Device since TEEP-Device is developed on top of TA-Ref SDK. The TEEP-Device is one of the applications of TA-Ref.

Docker images have all necessary development packages for building TEEP-Device for all three TEEs. The instructions of usage are described from the next chapters.

Purpose	Docker image
Keystone	aistcpsec/taref-dev:keystone
OP-TEE	aistcpsec/taref-dev:optee
Intel SGX	aistcpsec/taref-dev:sgx

6.4 Run tamproto (TAM Server) - Required by all Keystone/OP-TEE/SGX

To run TEEP-Device, first we need to run tamproto inside the same host. Let's clone the tamproto and start it.

Running tamproto

Open the first terminal for the tamproto.

```
# Clone the tamproto repo and checkout master branch
$ git clone https://github.com/ko-isobe/tamproto.git
$ cd tamproto
$ git checkout fb1961bc964857384c9ed8696c0d5fc0a76a319d
$ docker-compose build
$ docker-compose up
```

Trimmed output of starting tamproto

```
tam_api_1  |   TEE_pub: ’teep.jwk’ }
tam_api_1  | Load key TAM_priv
tam_api_1  | Load key TAM_pub
tam_api_1  | Load key TEE_priv
tam_api_1  | Load key TEE_pub
tam_api_1  | Key binary loaded
tam_api_1  | 192.168.11.4
tam_api_1  | Express HTTP  server listening on port 8888
tam_api_1  | Express HTTPS server listening on port 8443
```

6.4.1 Build TEEP-Device for Keystone with Docker

Clone TEEP-Device

Open the second terminal for editing the sources of TEEP-Device. The directory of cloning sources is mounted when running Docker in the next step.

```
# Clone the teep-device repo and checkout master branch
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
```

Match the user privilege with the one used in the container to prevent permission errors when editing sources. Container uses a build-user account with 1000:1000.

```
# Return back to parent directory of teep-device
$ sudo chown -R 1000:1000 teep-device/
$ sudo chmod -R a+w teep-device/
$ git config --global --add safe.directory $(pwd)/teep-device
```

Start the Docker

Open the third terminal. Here we build the TEEP-Device and run it to communicate with tamproto opened on the first terminal. If any error occurs, edit the sources on the second terminal to debug.

```
# Change the directory into teep-device before starting the docker
# Start the docker
$ docker run --network tamproto_default -w /home/user/teep-device -it --rm -v $(pwd):/home/user/teep-device
      aistcpsec/taref-dev:keystone
```

After you start the Docker command, you will be logged-in inside the Docker container. Following are the commands to be executed inside the Docker.

Build

```
# [Inside docker image]
# Change to teep-device
```

```
$ cd ~/teep-device/
# Build the teep-device
$ make
# Trimmed output of make command
....
make -C sample rootfs TAM_URL=http://tamproto_tam_api_1:8888
make[1]: Entering directory ’/home/user/teep-device/sample’
cp /home/user/keystone/build/buildroot.build/images/rootfs.ext2
     /home/user/teep-device/sample/../build/keystone/rootfs.ext2
e2mkdir -O root -G root /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/hello-tc/App-keystone
         /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker/hello-app
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/agent/teep-agent-ta
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/broker/teep-broker-app
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/scripts/env.sh
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/scripts/itc.sh
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/scripts/rtc.sh
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/scripts/showtamurl.sh
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/scripts/get-ip.sh
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/scripts/cp_ta_to_tamproto.sh

        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/keystone/sdk/build64/runtime/eyrie-rt
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/root/teep-broker
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/ree/mbedtls/library/lib*
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/usr/lib
e2cp -O root -G root -p /home/user/teep-device/sample/../build/keystone/ree/libwebsockets/lib/lib*
        /home/user/teep-device/sample/../build/keystone/rootfs.ext2:/usr/lib
make[1]: Leaving directory ’/home/user/teep-device/sample’
```

Run manually

After the successful build, run the sample TEEP session with tamproto.

Launch qemu of RISC-V with Keystone. The password for root is 'sifive'

```
$ make run-qemu
make -C sample run-qemu TAM_URL=http://tamproto_tam_api_1:8888
make[1]: Entering directory ’/home/user/teep-device/sample’
qemu-system-riscv64 \
    -m 4G \
    -bios /home/user/keystone/build/bootrom.build/bootrom.bin \
    -nographic \
    -machine virt \
    -kernel /home/user/keystone/build/sm.build/platform/generic/firmware/fw_payload.elf \
    -append "console=ttyS0 ro root=/dev/vda cma=256M@0x00000000C0000000" \
    -device virtio-blk-device,drive=hd0
    -drive file=/home/user/teep-device/sample/../build/keystone/rootfs.ext2,format=raw,id=hd0 \
    -netdev user,id=net0,net=192.168.100.1/24,dhcpstart=192.168.100.128,hostfwd=tcp::10032-:22 \
    -device virtio-net-device,netdev=net0 \
    -device virtio-rng-pci
overriding secure boot ROM (file: /home/user/keystone/build/bootrom.build/bootrom.bin)
boot ROM size: 53869
fdt dumped at 57968
OpenSBI v0.8
.....
Starting dropbear sshd: OK
Welcome to Buildroot
buildroot login: root
Password:
#
```

Please enter login username and password as root and sifive.

After login, start from installing the driver for Keystone.

```
# ls
keystone-driver.ko  teep-broker         tests.ke
# cd teep-broker/
# ls
cp_ta_to_tamproto.sh  hello-app             showtamurl.sh
```

```
env.sh                hello-ta              teep-agent-ta
eyrie-rt              itc.sh                teep-broker-app
get-ip.sh             rtc.sh
# source env.sh
[  388.139452] keystone_driver: loading out-of-tree module taints kernel.
[  388.146850] keystone_enclave: keystone enclave v1.0.0
```

There are helper scripts to handle the teep-broker. Following are the few of them and its usage.

- showtamurl.sh
- itc.sh
- rtc.sh

showtamurl.sh

This script prints out the tamproto values which has to be suffixed when we execute the built teep-broker-app. This script gets the url of the Tamproto either from the TAM_URL env variable or by internally executing get-ip.sh (get-ip.sh returns the IP of tamproto running in the same machine)

example:

```
$ ./showtamurl.sh
--tamurl 192.168.100.114/api/tam_cbor
```

You can simply copy the output of the showtamurl.sh and paste it to the end of the generated teep-broker-app binary.For ex:

```
$ ./teep-broker-app --tamurl http://192.168.100.114:8888/api/tam_cbor
```

itc.sh

Initiate teep-agent with tamproto. This script is for debugging the confirmative and handling of formats of TEEP Messages and SUIT Manifest in teep-agent and tamproto. Make sure you have copied the below files into tamproto server before running the ./teep-broker-app.

1) build/keystone/hello-tc/8d82573a-926d-4754-9353-32dc29997f74.ta
2) build/keystone/hello-tc/signed-download-tc.suit as integrated-payload-manifest.cbor

```
# cat ita.sh
#!/bin/bash -x
./teep-broker-app --tamurl ${TAM_URL}/api/tam_cbor
# ./itc.sh
```

rtc.sh

Execute the downloaded TC from the tamproto. This script is for debugging the implementation of the TC.

```
# ./rtc.sh
+ echo Running downloaded TC from the TAM
Running downloaded TC from the TAM
+ ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
main start
Hello TEEP from TEE!
main end
```

To exit from qemu.

```
# poweroff
```

The log message of tamproto will be shown on the terminal of running tamproto.

```
tam_api_1  | POST /api/tam_cbor 200 2.816 ms - 399
tam_api_1  | Access from: ::ffff:172.18.0.3
tam_api_1  | {
tam_api_1  |   pragma: ’no-cache’,
tam_api_1  |   ’cache-control’: ’no-cache’,
tam_api_1  |   host: ’example.com’,
tam_api_1  |   origin: ’http://192.168.11.3’,
tam_api_1  |   connection: ’close’,
tam_api_1  |   accept: ’application/teep+cbor’,
tam_api_1  |   ’content-length’: ’13’,
tam_api_1  |   ’content-type’: ’application/teep+cbor’
tam_api_1  | }
tam_api_1  | <Buffer 82 05 a1 14 48 77 77 77 77 77 77 77 77>
tam_api_1  | {
tam_api_1  |   TYPE: 5,
tam_api_1  |   token: <Buffer 77 77 77 77 77 77 77 77>,
tam_api_1  |   TOKEN: <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | }
tam_api_1  | TAM ProcessTeepMessage instance
tam_api_1  | TEEP-Protocol:parse
tam_api_1  | {
tam_api_1  |   TYPE: 5,
tam_api_1  |   token: <Buffer 77 77 77 77 77 77 77 77>,
tam_api_1  |   TOKEN: <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | }
tam_api_1  | object
tam_api_1  | *parseSuccessMessage
tam_api_1  | <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | undefined
tam_api_1  | TAM ProcessTeepMessage response
tam_api_1  | undefined
tam_api_1  | WARNING: Agent may sent invalid contents. TAM responses null.
tam_api_1  | POST /api/tam_cbor 204 1.357 ms - -
```

These are trimmed outputs of all procedures above on the terminal of the running container.

```
build-user@86417bb9c512:~/teep-device$ make run-qemu
make -C sample run-qemu TAM_URL=http://tamproto_tam_api_1:8888
make[1]: Entering directory ’/home/user/teep-device/sample’
qemu-system-riscv64 \
    -m 4G \
    -bios /home/user/keystone/build/bootrom.build/bootrom.bin \
    -nographic \
    -machine virt \
    -kernel /home/user/keystone/build/sm.build/platform/generic/firmware/fw_payload.elf \
    -append "console=ttyS0 ro root=/dev/vda cma=256M@0x00000000C0000000" \
    -device virtio-blk-device,drive=hd0 -drive
    file=/home/user/teep-device/sample/../build/keystone/rootfs.ext2,format=raw,id=hd0 \
    -netdev user,id=net0,net=192.168.100.1/24,dhcpstart=192.168.100.128,hostfwd=tcp::10032-:22 \
    -device virtio-net-device,netdev=net0 \
    -device virtio-rng-pci
overriding secure boot ROM (file: /home/user/keystone/build/bootrom.build/bootrom.bin)
boot ROM size: 53869
fdt dumped at 57968
OpenSBI v0.8
   ____                    _____ ____ _____
  / __ \                  / ____|  _ \_   _|
 | |  | |_ __   ___ _ __ | (___ | |_) || |
 | |  | | ’_ \ / _ \ ’_ \ \___ \|  _ < | |
 | |__| | |_) |  __/ | | |____) | |_) || |_
  \____/| .__/ \___|_| |_|_____/|____/_____|
        | |
        |_|
Platform Name             : riscv-virtio,qemu
Platform Features         : timer,mfdeleg
Platform HART Count       : 1
Firmware Base             : 0x80000000
Firmware Size             : 204 KB
Runtime SBI Version       : 0.2
Domain0 Name              : root
Domain0 Boot HART         : 0
Domain0 HARTs             : 0*
Domain0 Region00          : 0x0000000080000000-0x000000008003ffff ()
Domain0 Region01          : 0x0000000000000000-0xffffffffffffffff (R,W,X)
Domain0 Next Address      : 0x0000000080200000
Domain0 Next Arg1         : 0x0000000082200000
Domain0 Next Mode         : S-mode
```

```
Domain0 SysReset          : yes
...
...
...
Starting syslogd: OK
Starting klogd: OK
Running sysctl: OK
Saving random seed: OK
Starting network: udhcpc: started, v1.32.0
udhcpc: sending discover
udhcpc: sending select for 192.168.100.128
udhcpc: lease of 192.168.100.128 obtained, lease time 86400
deleting routers
adding dns 192.168.100.3
OK
Starting dropbear sshd: OK
Welcome to Buildroot
buildroot login: root
Password:
#
#
# ls
keystone-driver.ko  teep-broker         tests.ke
# cd teep-broker/
# source env.sh
[   18.953988] keystone_driver: loading out-of-tree module taints kernel.
[   18.960803] keystone_enclave: keystone enclave v1.0.0
#
# ./itc.sh
+ ./teep-broker-app --tamurl http://tamproto_tam_api_1:8888/api/tam_cbor
teep-broker.c compiled at Nov 24 2022 05:19:23
uri = http://tamproto_tam_api_1:8888/api/tam_cbor, cose=0, talist=
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799f2000-0x179c00000 (2104 KB), va 0xffffffff001f2000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
[1970/01/01 00:00:37:6062] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:37:6073] N: (hexdump: zero length)
[1970/01/01 00:00:37:6117] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:37:6716] N:
[1970/01/01 00:00:37:6725] N: 0000: 83 01 A5 01 81 01 03 81 00 04 43 01 02 05 14 48
       ..........C....H
[1970/01/01 00:00:37:6731] N: 0010: 77 77 77 77 77 77 77 77 15 81 00 02                wwwwwwww....

[1970/01/01 00:00:37:6738] N:
[1970/01/01 00:00:37:6829] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:37:6843] N:
[1970/01/01 00:00:37:6849] N: 0000: 82 02 A4 14 48 77 77 77 77 77 77 77 77 08 80 0E
       ....Hwwwwwwww...
[1970/01/01 00:00:37:6856] N: 0010: 80 0F 80                                           ...

[1970/01/01 00:00:37:6862] N:
[1970/01/01 00:00:37:6907] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:37:7111] N:
[1970/01/01 00:00:37:7115] N: 0000: 82 03 A2 0A 81 59 01 66 D8 6B A2 02 58 73 82 58
       .....Y.f.k..Xs.X
[1970/01/01 00:00:37:7123] N: 0010: 24 82 2F 58 20 63 70 90 82 1C BB B2 67 95 42 78    $./X
       cp.....g.Bx
[1970/01/01 00:00:37:7129] N: 0020: 7B 49 F4 5E 14 AF 0C BF AD 9E F4 A4 F0 B3 42 B9
       {I.^.........B.
[1970/01/01 00:00:37:7135] N: 0030: 23 35 56 05 AF 58 4A D2 84 43 A1 01 26 A0 F6 58
       #5V..XJ..C..&..X
[1970/01/01 00:00:37:7142] N: 0040: 40 BF B8 79 C8 E0 9B DE 52 2E 84 38 10 D7 7A 2E
       @..y....R..8..z.
[1970/01/01 00:00:37:7149] N: 0050: AF 88 62 53 C1 C1 83 D6 A2 FD 9E A1 DA 6F 4A D1
       ..bS.........oJ.
[1970/01/01 00:00:37:7156] N: 0060: DD 16 84 72 BC D6 10 81 F0 AE 30 A8 05 36 91 0E
       ...r......0..6..
[1970/01/01 00:00:37:7163] N: 0070: C9 D7 63 90 B4 E9 C9 64 A1 C3 6C F0 FE 29 71 91
       ..c....d..l..)q.
[1970/01/01 00:00:37:7170] N: 0080: 84 03 58 EA A5 01 01 02 01 03 58 86 A2 02 81 84
       ..X.......X.....
[1970/01/01 00:00:37:7176] N: 0090: 4B 54 45 45 50 2D 44 65 76 69 63 65 48 53 65 63
       KTEEP-DeviceHSec
[1970/01/01 00:00:37:7182] N: 00A0: 75 72 65 46 53 50 8D 82 57 3A 92 6D 47 54 93 53
       ureFSP..W:.mGT.S
[1970/01/01 00:00:37:7188] N: 00B0: 32 DC 29 99 7F 74 42 74 61 04 58 56 86 14 A4 01
       2.)..tBta.XV....
[1970/01/01 00:00:37:7194] N: 00C0: 50 FA 6B 4A 53 D5 AD 5F DF BE 9D E6 63 E4 D4 1F
       P.kJS.._....c...
[1970/01/01 00:00:37:7201] N: 00D0: FE 02 50 14 92 AF 14 25 69 5E 48 BF 42 9B 2D 51
       ..P....%i^H.B.-Q
[1970/01/01 00:00:37:7207] N: 00E0: F2 AB 45 03 58 24 82 2F 58 20 00 11 22 33 44 55    ..E.X$./X
       ..3DU
[1970/01/01 00:00:37:7213] N: 00F0: 66 77 88 99 AA BB CC DD EE FF 01 23 45 67 89 AB
```

```
       fw.........#Eg..
[1970/01/01 00:00:37:7219] N: 0100: CD EF FE DC BA 98 76 54 32 10 0E 19 87 D0 01 0F
       ......vT2.......
[1970/01/01 00:00:37:7226] N: 0110: 02 0F 09 58 54 86 13 A1 15 78 4A 68 74 74 70 3A
       ...XT....xJhttp:
[1970/01/01 00:00:37:7233] N: 0120: 2F 2F 74 61 6D 70 72 6F 74 6F 5F 74 61 6D 5F 61
       //tamproto_tam_a
[1970/01/01 00:00:37:7239] N: 0130: 70 69 5F 31 3A 38 38 38 38 2F 54 41 73 2F 38 64
       pi_1:8888/TAs/8d
[1970/01/01 00:00:37:7246] N: 0140: 38 32 35 37 33 61 2D 39 32 36 64 2D 34 37 35 34
       82573a-926d-4754
[1970/01/01 00:00:37:7253] N: 0150: 2D 39 33 35 33 2D 33 32 64 63 32 39 39 39 37 66
       -9353-32dc29997f
[1970/01/01 00:00:37:7259] N: 0160: 37 34 2E 74 61 15 02 03 0F 0A 43 82 03 0F 14 48
       74.ta.....C....H
[1970/01/01 00:00:37:7265] N: 0170: AB A1 A2 A3 A4 A5 A6 A7                             ........

[1970/01/01 00:00:37:7272] N:
TTRC:verifying signature of suit manifest
TTRC:verify OK
TTRC:command: 20
TTRC:execute suit-set-parameters
TTRC:command: 1
TTRC:execute suit-condition-vendor-identifier
TTRC:command: 2
TTRC:execute suit-condition-class-identifier
TTRC:command: 19
TTRC:execute suit-set-parameters
TTRC:command: 21
TTRC:execute suit-directive-fetch
TTRC:fetch_and_store component
[1970/01/01 00:00:38:3479] N: GET: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[1970/01/01 00:00:38:3497] N: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
TTRC:component download 153128
TTRC:ta-store.c: store_component() store component
TTRC:  device   = TEEP-Device
TTRC:  storage  = SecureFS
TTRC:  filename = 8d82573a-926d-4754-9353-32dc29997f74.ta
TTRC:  image_len = 153128
TTRC:finish fetch
TTRC:command: 3
TTRC:execute suit-condition-image-match
TTRC:end of command seq
[1970/01/01 00:00:38:8690] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:38:8695] N:
[1970/01/01 00:00:38:8699] N: 0000: 82 05 A1 14 48 77 77 77 77 77 77 77 77             ....Hwwwwwwww

[1970/01/01 00:00:38:8704] N:
[1970/01/01 00:00:38:8714] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:38:8873] N: (hexdump: zero length)
#
# ./rtc.sh
+ echo Running downloaded TC from the TAM
Running downloaded TC from the TAM
+ ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
main start
Hello TEEP from TEE!
main end
#
# poweroff
# Stopping dropbear sshd: OK
Stopping network: OK
Saving random seed: OK
Stopping klogd: OK
Stopping syslogd: OK
umount: devtmpfs busy - remounted read-only
[  177.941806] EXT4-fs (vda): re-mounted. Opts: (null)
The system is going down NOW!
logout
Sent SIGTERM to all processes
Sent SIGKILL to all processes
Requesting system poweroff
[  179.965743] reboot: Power down
make[1]: Leaving directory '/home/user/teep-device/sample'
```

Run automatically

This command will run all the previous manual procedures. It is mainly prepared for running TEEP-Device in CI.

```
$ make run-sample-session
```

Trimmed output printing 'Hello TEEP from TEE!'

```
Welcome to Buildroot
buildroot login: root
Password: sifive
# PS1='##"## '
#### insmod keystone-driver.ko || echo 'err"or'
[    5.247409] keystone_driver: loading out-of-tree module taints kernel.
[    5.254006] keystone_enclave: keystone enclave v1.0.0
#### cd /root/teep-broker
#### ls -l
total 3422
-rwxr-xr-x    1 1000     1000           567 Nov 24  2022 cp_ta_to_tamproto.sh
-rwxr-xr-x    1 1000     1000           156 Nov 24  2022 env.sh
-rwxr-xr-x    1 1000     1000         98088 Nov 21  2022 eyrie-rt
-rwxr-xr-x    1 1000     1000           290 Nov 24  2022 get-ip.sh
-rwxr-xr-x    1 1000     1000        437656 Nov 24  2022 hello-app
-rwxr-xr-x    1 1000     1000        153128 Nov 24  2022 hello-ta
-rwxr-xr-x    1 1000     1000            65 Nov 24  2022 itc.sh
-rwxr-xr-x    1 1000     1000           116 Nov 24  2022 rtc.sh
-rwxr-xr-x    1 1000     1000           134 Nov 24  2022 showtamurl.sh
-rwxr-xr-x    1 1000     1000        415280 Nov 24  2022 teep-agent-ta
-rwxr-xr-x    1 1000     1000       2372112 Nov 24  2022 teep-broker-app
#### ./hello-app hello-ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
main start
Hello TEEP from TEE!
main end
#### ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[Keystone SDK] /home/user/keystone/sdk/src/host/ElfFile.cpp:26 : file does not exist -
       8d82573a-926d-4754-9353-32dc29997f74.ta
[Keystone SDK] /home/user/keystone/sdk/src/host/Enclave.cpp:209 : Invalid enclave ELF
./hello-app: Unable to start enclave
#### ./teep-broker-app --tamurl http://tamproto_tam_api_1:8888/api/tam_cbor
teep-broker.c compiled at Nov 24 2022 05:09:43
uri = http://tamproto_tam_api_1:8888/api/tam_cbor, cose=0, talist=
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799f2000-0x179c00000 (2104 KB), va 0xffffffff001f2000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
[1970/01/01 00:00:08:4238] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:08:4285] N: (hexdump: zero length)
[1970/01/01 00:00:08:4349] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:08:5193] N:
[1970/01/01 00:00:08:5203] N: 0000: 83 01 A5 01 81 01 03 81 00 04 43 01 02 05 14 48
       ..........C....H
[1970/01/01 00:00:08:5211] N: 0010: 77 77 77 77 77 77 77 77 15 81 00 02                wwwwwwww....

[1970/01/01 00:00:08:5217] N:
[1970/01/01 00:00:08:5347] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:08:5354] N:
[1970/01/01 00:00:08:5364] N: 0000: 82 02 A4 14 48 77 77 77 77 77 77 77 77 08 80 0E
       ....Hwwwwwwww...
[1970/01/01 00:00:08:5373] N: 0010: 80 0F 80                                           ...

[1970/01/01 00:00:08:5380] N:
[1970/01/01 00:00:08:5427] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:08:5735] N:
[1970/01/01 00:00:08:5741] N: 0000: 82 03 A2 0A 81 59 01 66 D8 6B A2 02 58 73 82 58
       .....Y.f.k..Xs.X
[1970/01/01 00:00:08:5748] N: 0010: 24 82 2F 58 20 63 70 90 82 1C BB B2 67 95 42 78    $./X
       cp.....g.Bx
[1970/01/01 00:00:08:5756] N: 0020: 7B 49 F4 5E 14 AF 0C BF AD 9E F4 A4 F0 B3 42 B9
       {I.^..........B.
[1970/01/01 00:00:08:5763] N: 0030: 23 35 56 05 AF 58 4A D2 84 43 A1 01 26 A0 F6 58
       #5V..XJ..C..&..X
[1970/01/01 00:00:08:5770] N: 0040: 40 4C 09 82 3E 54 8D 4B 51 23 A7 68 34 2F 65 3F
       @L..>T.KQ#.h4/e?
[1970/01/01 00:00:08:5778] N: 0050: CE 7E F1 8D 0A C0 24 19 2E AD D7 0C 67 6C 81 25
       .~....$.....gl.%
[1970/01/01 00:00:08:5785] N: 0060: FF A3 37 23 17 2C FB B7 67 73 45 88 70 13 DF A1
       ..7#.,..gsE.p...
[1970/01/01 00:00:08:5792] N: 0070: 1D 74 8C D3 14 03 B7 7C 84 40 46 D4 66 9E 37 44
       .t.....|.@F.f.7D
[1970/01/01 00:00:08:5801] N: 0080: FE 03 58 EA A5 01 01 02 01 03 58 86 A2 02 81 84
       ..X.......X.....
[1970/01/01 00:00:08:5808] N: 0090: 4B 54 45 45 50 2D 44 65 76 69 63 65 48 53 65 63
       KTEEP-DeviceHSec
[1970/01/01 00:00:08:5815] N: 00A0: 75 72 65 46 53 50 8D 82 57 3A 92 6D 47 54 93 53
       ureFSP..W:.mGT.S
[1970/01/01 00:00:08:5822] N: 00B0: 32 DC 29 99 7F 74 42 74 61 04 58 56 86 14 A4 01
       2.)..tBta.XV....
[1970/01/01 00:00:08:5829] N: 00C0: 50 FA 6B 4A 53 D5 AD 5F DF BE 9D E6 63 E4 D4 1F
       P.kJS.._....c...
[1970/01/01 00:00:08:5836] N: 00D0: FE 02 50 14 92 AF 14 25 69 5E 48 BF 42 9B 2D 51
       ..P....%i^H.B.-Q
[1970/01/01 00:00:08:5845] N: 00E0: F2 AB 45 03 58 24 82 2F 58 20 00 11 22 33 44 55    ..E.X$./X
       .."3DU
[1970/01/01 00:00:08:5852] N: 00F0: 66 77 88 99 AA BB CC DD EE FF 01 23 45 67 89 AB
       fw.........#Eg..
[1970/01/01 00:00:08:5860] N: 0100: CD EF FE DC BA 98 76 54 32 10 0E 19 87 D0 01 0F
       ......vT2.......
[1970/01/01 00:00:08:5868] N: 0110: 02 0F 09 58 54 86 13 A1 15 78 4A 68 74 74 70 3A
       ...XT....xJhttp:
[1970/01/01 00:00:08:5876] N: 0120: 2F 2F 74 61 6D 70 72 6F 74 6F 5F 74 61 6D 5F 61
       //tamproto_tam_a
[1970/01/01 00:00:08:5885] N: 0130: 70 69 5F 31 3A 38 38 38 38 2F 54 41 73 2F 38 64
       pi_1:8888/TAs/8d
[1970/01/01 00:00:08:5893] N: 0140: 38 32 35 37 33 61 2D 39 32 36 64 2D 34 37 35 34
       82573a-926d-4754
[1970/01/01 00:00:08:5900] N: 0150: 2D 39 33 35 33 2D 33 32 64 63 32 39 39 39 37 66
       -9353-32dc29997f
[1970/01/01 00:00:08:5907] N: 0160: 37 34 2E 74 61 15 02 03 0F 0A 43 82 03 0F 14 48
       74.ta.....C....H
[1970/01/01 00:00:08:5914] N: 0170: AB A1 A2 A3 A4 A5 A6 A7                            ........

[1970/01/01 00:00:08:5923] N:
TTRC:verifying signature of suit manifest
TTRC:verify OK
TTRC:command: 20
TTRC:execute suit-set-parameters
TTRC:command: 1
TTRC:execute suit-condition-vendor-identifier
TTRC:command: 2
TTRC:execute suit-condition-class-identifier
TTRC:command: 19
TTRC:execute suit-set-parameters
TTRC:command: 21
TTRC:execute suit-directive-fetch
TTRC:fetch_and_store component
[1970/01/01 00:00:09:3454] N: GET: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[1970/01/01 00:00:09:3475] N: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
TTRC:component download 153128
TTRC:ta-store.c: store_component() store component
TTRC:  device   = TEEP-Device
TTRC:  storage  = SecureFS
TTRC:  filename = 8d82573a-926d-4754-9353-32dc29997f74.ta
TTRC:  image_len = 153128
TTRC:finish fetch
TTRC:command: 3
TTRC:execute suit-condition-image-match
TTRC:end of command seq
[1970/01/01 00:00:09:9560] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:09:9565] N:
[1970/01/01 00:00:09:9569] N: 0000: 82 05 A1 14 48 77 77 77 77 77 77 77 77             ....Hwwwwwwww

[1970/01/01 00:00:09:9574] N:
[1970/01/01 00:00:09:9583] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[1970/01/01 00:00:09:9754] N: (hexdump: zero length)
#### ls -l
total 3573
-rw-------    1 root     root        153128 Jan  1 00:00 8d82573a-926d-4754-9353-32dc29997f74.ta
-rwxr-xr-x    1 1000     1000           567 Nov 24  2022 cp_ta_to_tamproto.sh
-rwxr-xr-x    1 1000     1000           156 Nov 24  2022 env.sh
-rwxr-xr-x    1 1000     1000         98088 Nov 21  2022 eyrie-rt
-rwxr-xr-x    1 1000     1000           290 Nov 24  2022 get-ip.sh
-rwxr-xr-x    1 1000     1000        437656 Nov 24  2022 hello-app
-rwxr-xr-x    1 1000     1000        153128 Nov 24  2022 hello-ta
-rwxr-xr-x    1 1000     1000            65 Nov 24  2022 itc.sh
-rwxr-xr-x    1 1000     1000           116 Nov 24  2022 rtc.sh
-rwxr-xr-x    1 1000     1000           134 Nov 24  2022 showtamurl.sh
-rwxr-xr-x    1 1000     1000        415280 Nov 24  2022 teep-agent-ta
-rwxr-xr-x    1 1000     1000       2372112 Nov 24  2022 teep-broker-app
#### ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
main start
Hello TEEP from TEE!
main end
####  done
```

Cleaning built binaries. Deleting the built binaries are required when starting to build TEEP-Device on other CPU architectures otherwise will generate errors.

```
$ make clean
```

6.4.2 Build TEEP-Device for OP-TEE with Docker

Clone TEEP-Device

```
# Clone the teep-device repo and checkout master branch
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
```

Start the Docker

```
# Start the Docker
$ docker run --network tamproto_default -w /home/user/teep-device -it --rm -v $(pwd):/home/user/teep-device
       aistcpsec/taref-dev:optee
```

After you start the Docker command, you will be logged-in inside the Docker container. Following are the commands to be executed inside the Docker.

```
# [Inside docker image]
# Change to teep-device
$ cd ~/teep-device/
# Build the teep device
$ make
```

After the successful build, run the sample TEEP session with tamproto.

```
# After the successful build
# Run the TEEP-Device
$ make run-sample-session
```

Trimmed output of the TEEP-Device

```
...
...
...
FI stub: Booting Linux Kernel...
EFI stub: EFI_RNG_PROTOCOL unavailable, no randomness supplied
EFI stub: Using DTB from configuration table
EFI stub: Exiting boot services and installing virtual address map...
Starting syslogd: OK
Starting klogd: OK
Initializing random number generator... [    3.248115] random: dd: uninitialized urandom read (512
       bytes read)
done.
Set permissions on /dev/tee*: OK
Set permissions on /dev/ion: OK
Create/set permissions on /data/tee: OK
Starting tee-supplicant: OK
Starting network: OK
Starting network (udhcpc): OK
Welcome to Buildroot, type root or test to login
buildroot login: root
#  done, guest is booted.
export LD_LIBRARY_PATH=/lib:/lib/arm-linux-gnueabihf:/lib/optee_armtz:/usr/lib
# cd teep-broker
# ls -l
total 4224
-rwxr-xr-x    1 root     root           567 Nov 24 06:51 cp_ta_to_tamproto.sh
-rwxr-xr-x    1 root     root           153 Nov 24 06:51 env.sh
-rwxr-xr-x    1 root     root           290 Nov 24 06:51 get-ip.sh
-rwxr-xr-x    1 root     root         14112 Nov 24 06:51 hello-app
-rwxr-xr-x    1 root     root            65 Nov 24 06:51 itc.sh
```

```
-rwxr-xr-x    1 root     root           116 Nov 24 06:51 rtc.sh
-rwxr-xr-x    1 root     root           134 Nov 24 06:51 showtamurl.sh
-rwxr-xr-x    1 root     root       4280472 Nov 24 06:51 teep-broker-app
# ./hello-app
hello-app: TEEC_Opensession failed with code 0xffff0008 origin 0x3
# ./teep-broker-app --tamurl http://tamproto_tam_api_1:8888/api/tam_cbor
teep-broker.c compiled at Nov 24 2022 06:51:18
uri = http://tamproto_tam_api_1:8888/api/tam_cbor, cose=0, talist=
[2022/11/24 06:52:30:1216] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 06:52:30:1226] N: (hexdump: zero length)
[2022/11/24 06:52:30:1270] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 06:52:30:1915] N:
[2022/11/24 06:52:30:1923] N: 0000: 83 01 A5 01 81 01 03 81 00 04 43 01 02 05 14 48
       ..........C....H
[2022/11/24 06:52:30:1929] N: 0010: 77 77 77 77 77 77 77 77 15 81 00 02                wwwwwwww....

[2022/11/24 06:52:30:1932] N:
[2022/11/24 06:52:30:2009] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 06:52:30:2013] N:
[2022/11/24 06:52:30:2016] N: 0000: 82 02 A4 14 48 77 77 77 77 77 77 77 77 08 80 0E
       ....Hwwwwwwww...
[2022/11/24 06:52:30:2020] N: 0010: 80 0F 80                                           ...

[2022/11/24 06:52:30:2023] N:
[2022/11/24 06:52:30:2033] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 06:52:30:2215] N:
[2022/11/24 06:52:30:2217] N: 0000: 82 03 A2 0A 81 59 01 66 D8 6B A2 02 58 73 82 58
       .....Y.f.k..Xs.X
[2022/11/24 06:52:30:2222] N: 0010: 24 82 2F 58 20 63 70 90 82 1C BB B2 67 95 42 78    $./X
       cp.....g.Bx
[2022/11/24 06:52:30:2226] N: 0020: 7B 49 F4 5E 14 AF 0C BF AD 9E F4 A4 F0 B3 42 B9
       {I.^..........B.
[2022/11/24 06:52:30:2229] N: 0030: 23 35 56 05 AF 58 4A D2 84 43 A1 01 26 A0 F6 58
       #5V..XJ..C..&..X
[2022/11/24 06:52:30:2233] N: 0040: 40 91 2A 3A BF 8A 24 6E 5A A1 A7 69 D6 8F 12 DB
       @.*:..$nZ..i....
[2022/11/24 06:52:30:2236] N: 0050: 8F D1 FF F3 11 9F 02 58 C0 A4 B2 8F FF D0 6C A9
       .......X......l.
[2022/11/24 06:52:30:2242] N: 0060: 96 75 B1 43 37 B1 8C B9 73 58 15 05 5E F4 39 3A
       .u.C7...sX..^.9:
[2022/11/24 06:52:30:2246] N: 0070: 88 D7 DC B2 06 5D 58 F4 8C 70 78 D1 70 C3 1B 7B
       .....]X..px.p..{
[2022/11/24 06:52:30:2250] N: 0080: 4F 03 58 EA A5 01 01 02 01 03 58 86 A2 02 81 84
       O.X.......X.....
[2022/11/24 06:52:30:2253] N: 0090: 4B 54 45 45 50 2D 44 65 76 69 63 65 48 53 65 63
       KTEEP-DeviceHSec
[2022/11/24 06:52:30:2257] N: 00A0: 75 72 65 46 53 50 8D 82 57 3A 92 6D 47 54 93 53
       ureFSP..W:.mGT.S
[2022/11/24 06:52:30:2261] N: 00B0: 32 DC 29 99 7F 74 42 74 61 04 58 56 86 14 A4 01
       2.)..tBta.XV....
[2022/11/24 06:52:30:2264] N: 00C0: 50 FA 6B 4A 53 D5 AD 5F DF BE 9D E6 63 E4 D4 1F
       P.kJS.._....c...
[2022/11/24 06:52:30:2268] N: 00D0: FE 02 50 14 92 AF 14 25 69 5E 48 BF 42 9B 2D 51
       ..P....%i^H.B.-Q
[2022/11/24 06:52:30:2271] N: 00E0: F2 AB 45 03 58 24 82 2F 58 20 00 11 22 33 44 55    ..E.X$./X
       ..3DU
[2022/11/24 06:52:30:2274] N: 00F0: 66 77 88 99 AA BB CC DD EE FF 01 23 45 67 89 AB
       fw.........#Eg..
[2022/11/24 06:52:30:2278] N: 0100: CD EF FE DC BA 98 76 54 32 10 0E 19 87 D0 01 0F
       ......vT2.......
[2022/11/24 06:52:30:2285] N: 0110: 02 0F 09 58 54 86 13 A1 15 78 4A 68 74 74 70 3A
       ...XT....xJhttp:
[2022/11/24 06:52:30:2289] N: 0120: 2F 2F 74 61 6D 70 72 6F 74 6F 5F 74 61 6D 5F 61
       //tamproto_tam_a
[2022/11/24 06:52:30:2292] N: 0130: 70 69 5F 31 3A 38 38 38 38 2F 54 41 73 2F 38 64
       pi_1:8888/TAs/8d
[2022/11/24 06:52:30:2297] N: 0140: 38 32 35 37 33 61 2D 39 32 36 64 2D 34 37 35 34
       82573a-926d-4754
[2022/11/24 06:52:30:2301] N: 0150: 2D 39 33 35 33 2D 33 32 64 63 32 39 39 39 37 66
       -9353-32dc29997f
[2022/11/24 06:52:30:2305] N: 0160: 37 34 2E 74 61 15 02 03 0F 0A 43 82 03 0F 14 48
       74.ta.....C....H
[2022/11/24 06:52:30:2309] N: 0170: AB A1 A2 A3 A4 A5 A6 A7                            ........

[2022/11/24 06:52:30:2312] N:
[2022/11/24 06:52:30:5524] N: GET: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 06:52:30:5539] N: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 06:52:30:6340] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 06:52:30:6344] N:
[2022/11/24 06:52:30:6348] N: 0000: 82 05 A1 14 48 77 77 77 77 77 77 77 77             ....Hwwwwwwww

[2022/11/24 06:52:30:6352] N:
[2022/11/24 06:52:30:6359] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 06:52:30:6513] N: (hexdump: zero length)
# ls -l
```

```
total 4224
-rwxr-xr-x    1 root     root           567 Nov 24 06:51 cp_ta_to_tamproto.sh
-rwxr-xr-x    1 root     root           153 Nov 24 06:51 env.sh
-rwxr-xr-x    1 root     root           290 Nov 24 06:51 get-ip.sh
-rwxr-xr-x    1 root     root         14112 Nov 24 06:51 hello-app
-rwxr-xr-x    1 root     root            65 Nov 24 06:51 itc.sh
-rwxr-xr-x    1 root     root           116 Nov 24 06:51 rtc.sh
-rwxr-xr-x    1 root     root           134 Nov 24 06:51 showtamurl.sh
-rwxr-xr-x    1 root     root       4280472 Nov 24 06:51 teep-broker-app
# ./hello-app
#  done
cat /home/user/optee/out/bin/serial1.log
...
...
I/TC: Switching console to device: /pl011@9040000
I/TC: OP-TEE version: 3.10.0-dev (gcc version 8.3.0
 (GNU Toolchain for the A-profile Architecture 8.3-2019.03 (arm-rel-8.36))) #1 Mon 21 Nov 2022
       11:59:41 AM UTC aarch64
I/TC: Primary CPU initializing
D/TC:0 0 paged_init_primary:1188 Executing at offset 0xc6842000 with virtual load address 0xd4942000
D/TC:0 0 call_initcalls:21 level 1 register_time_source()
D/TC:0 0 call_initcalls:21 level 1 teecore_init_pub_ram()
D/TC:0 0 call_initcalls:21 level 3 check_ta_store()
D/TC:0 0 check_ta_store:636 TA store: "Secure Storage TA"
D/TC:0 0 check_ta_store:636 TA store: "REE"
D/TC:0 0 call_initcalls:21 level 3 init_user_ta()
D/TC:0 0 call_initcalls:21 level 3 verify_pseudo_tas_conformance()
D/TC:0 0 call_initcalls:21 level 3 mobj_mapped_shm_init()
D/TC:0 0 mobj_mapped_shm_init:434 Shared memory address range: d6400000, d8400000
D/TC:0 0 call_initcalls:21 level 3 tee_cryp_init()
D/TC:0 0 call_initcalls:21 level 4 tee_fs_init_key_manager()
D/TC:0 0 call_initcalls:21 level 6 mobj_init()
D/TC:0 0 call_initcalls:21 level 6 default_mobj_init()
D/TC:0 0 call_finalcalls:40 level 1 release_external_dt()
I/TC: Primary CPU switching to normal world boot
I/TC: Secondary CPU 1 initializing
D/TC:1   select_vector:1118 SMCCC_ARCH_WORKAROUND_1 (0x80008000) available
D/TC:1   select_vector:1119 SMC Workaround for CVE-2017-5715 used
I/TC: Secondary CPU 1 switching to normal world boot
D/TC:1   tee_entry_exchange_capabilities:102 Dynamic shared memory is enabled
D/TC:1 0 core_mmu_entry_to_finer_grained:762 xlat tables used 7 / 7
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 7011a688-ddde-4053-a5a9-7b3c4ddf13b8
D/TC:? 0 tee_ta_init_pseudo_ta_session:296 Open device.pta
D/TC:? 0 tee_ta_init_pseudo_ta_session:310 device.pta : 7011a688-ddde-4053-a5a9-7b3c4ddf13b8
D/TC:? 0 tee_ta_close_session:499 csess 0xd49bea00 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 load_ldelf:704 ldelf load address 0x40006000
D/LD:  ldelf:134 Loading TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 tee_ta_init_pseudo_ta_session:296 Open system.pta
D/TC:? 0 tee_ta_init_pseudo_ta_session:310 system.pta : 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 8d82573a-926d-4754-9353-32dc29997f74 (Secure
       Storage TA)
D/TC:? 0 system_open_ta_binary:260 res=0xffff0008
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 8d82573a-926d-4754-9353-32dc29997f74 (REE)
D/TC:? 0 system_open_ta_binary:260 res=0xffff0008
D/TC:? 0 tee_ta_invoke_command:773 Error: ffff0008 of 4
E/LD:  init_elf:438 sys_open_ta_bin(8d82573a-926d-4754-9353-32dc29997f74)
E/TC:? 0 init_with_ldelf:232 ldelf failed with res: 0xffff0008
D/TC:? 0 tee_ta_close_session:499 csess 0xd49be860 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0xd49be800)
D/TC:? 0 tee_ta_close_session:499 csess 0xd49be060 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
E/TC:? 0 tee_ta_open_session:728 Failed. Return error 0xffff0008
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 68373894-5bb3-403c-9eec-3114a1f5d3fc
D/TC:? 0 load_ldelf:704 ldelf load address 0x40006000
D/LD:  ldelf:134 Loading TA 68373894-5bb3-403c-9eec-3114a1f5d3fc
D/TC:? 0 tee_ta_init_session_with_context:573 Re-open TA 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 68373894-5bb3-403c-9eec-3114a1f5d3fc (Secure
       Storage TA)
D/TC:? 0 system_open_ta_binary:260 res=0xffff0008
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 68373894-5bb3-403c-9eec-3114a1f5d3fc (REE)
D/TC:? 0 system_open_ta_binary:260 res=0x0
D/LD:  ldelf:169 ELF (68373894-5bb3-403c-9eec-3114a1f5d3fc) at 0x4007c000
D/TC:? 0 tee_ta_close_session:499 csess 0xd49bd340 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
M/TA: TTRC:verifying signature of suit manifest
M/TA: TTRC:verify OK
M/TA: TTRC:command: 20
M/TA: TTRC:execute suit-set-parameters
M/TA: TTRC:command: 1
M/TA: TTRC:execute suit-condition-vendor-identifier
M/TA: TTRC:command: 2
```

```
M/TA: TTRC:execute suit-condition-class-identifier
M/TA: TTRC:command: 19
M/TA: TTRC:execute suit-set-parameters
M/TA: TTRC:command: 21
M/TA: TTRC:execute suit-directive-fetch
M/TA: TTRC:fetch_and_store component
M/TA: TTRC:component download 55976
M/TA: TTRC:ta-store.c: store_component() store component
M/TA: TTRC:  device   = TEEP-Device
M/TA: TTRC:  storage  = SecureFS
M/TA: TTRC:  filename = 8d82573a-926d-4754-9353-32dc29997f74.ta
M/TA: TTRC:  image_len = 55976
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 6e256cba-fc4d-4941-ad09-2ca1860342dd
D/TC:? 0 tee_ta_init_pseudo_ta_session:296 Open secstor_ta_mgmt
D/TC:? 0 tee_ta_init_pseudo_ta_session:310 secstor_ta_mgmt : 6e256cba-fc4d-4941-ad09-2ca1860342dd
D/TC:? 0 install_ta:99 Installing 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_close_session:499 csess 0xd49bbfa0 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
M/TA: TTRC:finish fetch
M/TA: TTRC:command: 3
M/TA: TTRC:execute suit-condition-image-match
M/TA: TTRC:end of command seq
D/TC:? 0 tee_ta_close_session:499 csess 0xd49bdb40 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0xd49bdae0)
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 load_ldelf:704 ldelf load address 0x40006000
D/LD:  ldelf:134 Loading TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_init_session_with_context:573 Re-open TA 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 8d82573a-926d-4754-9353-32dc29997f74 (Secure
       Storage TA)
D/TC:? 0 system_open_ta_binary:260 res=0x0
D/LD:  ldelf:169 ELF (8d82573a-926d-4754-9353-32dc29997f74) at 0x4003a000
D/TC:? 0 tee_ta_close_session:499 csess 0xd49bb600 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
Hello TEEP from TEE!
D/TC:? 0 tee_ta_close_session:499 csess 0xd49bbe00 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0xd49bbda0)
! fgrep 'ERR:' /home/user/optee/out/bin/serial1.log
fgrep 'Hello TEEP from TEE!' /home/user/optee/out/bin/serial1.log
Hello TEEP from TEE!
make[1]: Leaving directory '/home/user/teep-device/sample'
build-user@1364029c42f3:~/teep-device$
```

Cleaning built binaries. Deleting the built binaries are required when starting to build TEEP-Device on other CPU architectures otherwise will generate errors.

```
$ make clean
```

6.4.3 Build TEEP-Device for SGX with Docker

Clone TEEP-Device

```
# Clone the teep-device repo and checkout master branch
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
```

Start the Docker

```
# Start the Docker
$ docker run --network tamproto_default -w /home/user/teep-device -it --rm -v $(pwd):/home/user/teep-device
       aistcpsec/taref-dev:sgx
```

After you start the Docker command, you will be logged-in inside the Docker container. Following are the commands to be executed inside the Docker

```
# [Inside docker image]
# Change to teep-device
$ cd ~/teep-device/
# set the TEE environments for SGX
# The MACHINE=SIM specifies running SGX in simulation mode which
# will allow running SGX on all Intel and AMD cpu regardless of SGX support.
$ export MACHINE=SIM
# Build the teep device
$ make
```

After the successful build, run the sample TEEP session with tamproto.

```
# After the successful build
# Run the TEEP-Device
$ make run-sample-session
```

Trimmed output of the run.

```
build-user@4fcbd11fb97c:~/teep-device$ make run-sample-session
make -C sample run-session TAM_URL=http://tamproto_tam_api_1:8888
make[1]: Entering directory '/home/user/teep-device/sample'
make -C /home/user/teep-device/sample/../hello-tc/build-sgx
SOURCE=/home/user/teep-device/sample/../hello-tc upload-download-manifest
make[2]: Entering directory '/home/user/teep-device/hello-tc/build-sgx'
curl http://tamproto_tam_api_1:8888/panel/upload \
  -F "file=@/home/user/teep-device/hello-tc/build-sgx/signed-download-tc.suit;
    filename=integrated-payload-manifest.cbor"
<!-- /*
...
...
...
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7f62b7fe9000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7f62b841d000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7f62b8851000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7f62b8c85000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7f62b90b9000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7f62b94ed000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:342] Debug enclave. Checking if VTune is
       profiling or SGX_DBG_OPTIN is set
[read_cpusvn_file ../cpusvn_util.cpp:96] Couldn't find/open the configuration file
       /home/user/.cpusvn.conf.
[2022/11/24 07:06:03:9250] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 07:06:03:9250] N: (hexdump: zero length)
[2022/11/24 07:06:03:9250] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 07:06:03:9329] N:
[2022/11/24 07:06:03:9329] N: 0000: 83 01 A5 01 81 01 03 81 00 04 43 01 02 05 14 48
       ..........C....H
[2022/11/24 07:06:03:9329] N: 0010: 77 77 77 77 77 77 77 77 15 81 00 02                wwwwwwww....

[2022/11/24 07:06:03:9330] N:
[2022/11/24 07:06:03:9330] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 07:06:03:9330] N:
[2022/11/24 07:06:03:9330] N: 0000: 82 02 A4 14 48 77 77 77 77 77 77 77 77 08 80 0E
       ....Hwwwwwwww...
[2022/11/24 07:06:03:9330] N: 0010: 80 0F 80                                           ...

[2022/11/24 07:06:03:9330] N:
[2022/11/24 07:06:03:9331] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 07:06:03:9393] N:
[2022/11/24 07:06:03:9393] N: 0000: 82 03 A2 0A 81 59 01 66 D8 6B A2 02 58 73 82 58
       .....Y.f.k..Xs.X
[2022/11/24 07:06:03:9393] N: 0010: 24 82 2F 58 20 63 70 90 82 1C BB B2 67 95 42 78    $./X
       cp.....g.Bx
[2022/11/24 07:06:03:9393] N: 0020: 7B 49 F4 5E 14 AF 0C BF AD 9E F4 A4 F0 B3 42 B9
       {I.^.........B.
[2022/11/24 07:06:03:9393] N: 0030: 23 35 56 05 AF 58 4A D2 84 43 A1 01 26 A0 F6 58
       #5V..XJ..C..&..X
[2022/11/24 07:06:03:9393] N: 0040: 40 12 4A E1 1D DB DA 8A F7 FE 39 D2 57 D3 27 FF
       @.J.......9.W.'.
[2022/11/24 07:06:03:9393] N: 0050: 28 E6 F3 EF D9 E2 7A AD A9 70 B8 50 A3 EA 43 3C
       (.....z..p.P..C<
[2022/11/24 07:06:03:9393] N: 0060: 94 DC B3 58 6D E9 B0 21 EF 50 4B F7 02 81 A4 AF
       ...Xm..!.PK.....
[2022/11/24 07:06:03:9393] N: 0070: 7D DF 7E E6 57 2A F0 07 0A 89 3D E4 B7 BA 99 5F
       }.~.W*....=...._
[2022/11/24 07:06:03:9393] N: 0080: E3 03 58 EA A5 01 01 02 01 03 58 86 A2 02 81 84
       ..X.......X.....
[2022/11/24 07:06:03:9393] N: 0090: 4B 54 45 45 50 2D 44 65 76 69 63 65 48 53 65 63
       KTEEP-DeviceHSec
[2022/11/24 07:06:03:9393] N: 00A0: 75 72 65 46 53 50 8D 82 57 3A 92 6D 47 54 93 53
       ureFSP..W:.mGT.S
[2022/11/24 07:06:03:9393] N: 00B0: 32 DC 29 99 7F 74 42 74 61 04 58 56 86 14 A4 01
       2.)..tBta.XV....
[2022/11/24 07:06:03:9394] N: 00C0: 50 FA 6B 4A 53 D5 AD 5F DF BE 9D E6 63 E4 D4 1F
       P.kJS.._....c...
[2022/11/24 07:06:03:9394] N: 00D0: FE 02 50 14 92 AF 14 25 69 5E 48 BF 42 9B 2D 51
       ..P....%i^H.B.-Q
[2022/11/24 07:06:03:9394] N: 00E0: F2 AB 45 03 58 24 82 2F 58 20 00 11 22 33 44 55    ..E.X$./X
       ..3DU
[2022/11/24 07:06:03:9394] N: 00F0: 66 77 88 99 AA BB CC DD EE FF 01 23 45 67 89 AB
       fw.........#Eg..
[2022/11/24 07:06:03:9394] N: 0100: CD EF FE DC BA 98 76 54 32 10 0E 19 87 D0 01 0F
       ......vT2.......
[2022/11/24 07:06:03:9394] N: 0110: 02 0F 09 58 54 86 13 A1 15 78 4A 68 74 74 70 3A
       ...XT....xJhttp:
[2022/11/24 07:06:03:9394] N: 0120: 2F 2F 74 61 6D 70 72 6F 74 6F 5F 74 61 6D 5F 61
       //tamproto_tam_a
[2022/11/24 07:06:03:9394] N: 0130: 70 69 5F 31 3A 38 38 38 38 2F 54 41 73 2F 38 64
       pi_1:8888/TAs/8d
[2022/11/24 07:06:03:9394] N: 0140: 38 32 35 37 33 61 2D 39 32 36 64 2D 34 37 35 34
       82573a-926d-4754
[2022/11/24 07:06:03:9394] N: 0150: 2D 39 33 35 33 2D 33 32 64 63 32 39 39 39 37 66
       -9353-32dc29997f
[2022/11/24 07:06:03:9394] N: 0160: 37 34 2E 74 61 15 02 03 0F 0A 43 82 03 0F 14 48
       74.ta.....C....H
[2022/11/24 07:06:03:9394] N: 0170: AB A1 A2 A3 A4 A5 A6 A7                            ........

[2022/11/24 07:06:03:9394] N:
[2022/11/24 07:06:03:0110] N: GET: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 07:06:03:0110] N: http://tamproto_tam_api_1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 07:06:03:0162] N: POST: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 07:06:03:0162] N:
[2022/11/24 07:06:03:0162] N: 0000: 82 05 A1 14 48 77 77 77 77 77 77 77 77             ....Hwwwwwwww

[2022/11/24 07:06:03:0162] N:
[2022/11/24 07:06:03:0163] N: http://tamproto_tam_api_1:8888/api/tam_cbor
[2022/11/24 07:06:03:0184] N: (hexdump: zero length)
[CEnclavePool /home/user/linux-sgx/psw/urts/enclave.cpp:627] enter CEnclavePool constructor
[build_secs /home/user/linux-sgx/psw/urts/loader.cpp:516] Enclave start addr. = 0x7f62b5a94000, Size
       = 0x8000000, 131072 KB
TTRC:verifying signature of suit manifest
TTRC:verify OK
TTRC:command: 20
TTRC:execute suit-set-parameters
TTRC:command: 1
TTRC:execute suit-condition-vendor-identifier
TTRC:command: 2
TTRC:execute suit-condition-class-identifier
TTRC:command: 19
TTRC:execute suit-set-parameters
TTRC:command: 21
TTRC:execute suit-directive-fetch
TTRC:fetch_and_store component
TTRC:component download 326768
TTRC:ta-store.c: store_component() store component
TTRC:  device   = TEEP-Device
TTRC:  storage  = SecureFS
TTRC:  filename = 8d82573a-926d-4754-9353-32dc29997f74.ta
TTRC:  image_len = 326768
TINF: ta-store.c: 138: install_ta(): Return value of ocall_open_file is : retval  = 0
TTRC:finish fetch
TTRC:command: 3
TTRC:execute suit-condition-image-match
TTRC:end of command seq
ls -l /home/user/teep-device/sample/../build/sgx/agent
total 1720
-rw------- 1 build-user build-user 326768 Nov 24 07:06 8d82573a-926d-4754-9353-32dc29997f74.ta
-rw-r--r-- 1 build-user build-user   8248 Nov 24 07:04 ta-store.o
-rw-r--r-- 1 build-user build-user  59664 Nov 24 07:04 teep-agent-ta.o
-rw-r--r-- 1 build-user build-user 675904 Nov 24 07:04 teep-agent-ta.signed.so
-rwxr-xr-x 1 build-user build-user 675904 Nov 24 07:04 teep-agent-ta.so
cd /home/user/teep-device/sample/../build/sgx/../../hello-tc/build-sgx/ && \
    cp /home/user/teep-device/sample/../build/sgx/agent/8d82573a-926d-4754-9353-32dc29997f74.ta
    /home/user/teep-device/sample/../build/sgx/../../hello-tc/build-sgx/ && \
    ./App-sgx | \
    tee -a /home/user/teep-device/sample/../build/sgx/sgx.log
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 19, ptr = 1faa8
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 1b, ptr = 0
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 1a, ptr = 1faa8
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 1c, ptr = 18
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 6ffffef5, ptr = 2e8
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 5, ptr = 3c8
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 6, ptr = 320
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = a, ptr = 5c
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = b, ptr = 18
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 15, ptr = 0
```

```
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 3, ptr = 1ff98
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 7, ptr = 470
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 8, ptr = 570
[parse_dyn /home/user/linux-sgx/psw/urts/parser/elfparser.cpp:176] dynamic tag = 9, ptr = 18
...
...
[setreate_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7fef9dcaa000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7fef9e0de000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7fef9e512000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7fef9e946000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:332] add tcs 0x7fef9ed7a000
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:342] Debug enclave. Checking if VTune is
       profiling or SGX_DBG_OPTIN is set
[__create_enclave /home/user/linux-sgx/psw/urts/urts_com.h:388] VTune is not profiling and
       SGX_DBG_OPTIN is not set.
TCS Debug OPTIN bit not set and API to do module mapping not invoked
[read_cpusvn_file ../cpusvn_util.cpp:96] Couldn't find/open the configuration file
       /home/user/.cpusvn.conf.
[CEnclavePool /home/user/linux-sgx/psw/urts/enclave.cpp:627] enter CEnclavePool constructor
[build_secs /home/user/linux-sgx/psw/urts/loader.cpp:516] Enclave start addr. = 0x7fef9b365000, Size
       = 0x8000000, 131072 KB
main start
Hello TEEP from TEE!
main end
Info: Enclave successfully returned.
ls -l /home/user/teep-device/sample/../build/sgx/../../hello-tc/build-sgx
total 1048
-rw-r--r-- 1 build-user build-user 326768 Nov 24 07:06 8d82573a-926d-4754-9353-32dc29997f74.ta
-rwxr-xr-x 1 build-user build-user  31128 Nov 24 07:04 App-sgx
-rw-r--r-- 1 build-user build-user   4712 Nov 24 07:04 App-sgx.o
-rw-rw-rw- 1 build-user build-user    149 Nov 24 06:44 Enclave.lds
-rw-r--r-- 1 build-user build-user   3064 Nov 24 07:04 Enclave.o
-rw-rw-rw- 1 build-user build-user   2455 Nov 24 06:44 Enclave_private.pem
-rw-rw-rw- 1 build-user build-user    455 Nov 24 06:44 Makefile
-rw-rw-rw- 1 build-user build-user   1080 Nov 24 06:44 app.mk
drwxrwxrwx 2 build-user build-user   4096 Nov 24 06:44 config
-rw-r--r-- 1 build-user build-user    282 Nov 24 07:04 download-tc.suit
-rw-r--r-- 1 build-user build-user    761 Nov 24 07:04 download.json
-rw-r--r-- 1 build-user build-user 326986 Nov 24 07:04 embed-tc.suit
-rw-r--r-- 1 build-user build-user    209 Nov 24 07:04 embed-tc.suit.tmp
-rw-r--r-- 1 build-user build-user    690 Nov 24 07:04 embed.json
-rw-rw-rw- 1 build-user build-user   1917 Nov 24 06:44 enclave.mk
-rw-r--r-- 1 build-user build-user    358 Nov 24 07:04 signed-download-tc.suit
-rw-r--r-- 1 build-user build-user 327062 Nov 24 07:04 signed-embed-tc.suit
if ! [ -f /home/user/teep-device/sample/../build/sgx/../../hello-tc/build-sgx/8d82573a-926d-4754-9353-32dc29997f74.ta
       ];
    then \
    echo ERR: No TC found | tee -a /home/user/teep-device/sample/../build/sgx/sgx.log; \
fi
! fgrep 'ERR:' /home/user/teep-device/sample/../build/sgx/sgx.log
fgrep 'Hello TEEP from TEE!' /home/user/teep-device/sample/../build/sgx/sgx.log
Hello TEEP from TEE!
make[1]: Leaving directory '/home/user/teep-device/sample'
```

Cleaning built binaries. Deleting the built binaries are required when starting to build TEEP-Device on other CPU architectures otherwise will generate errors.

```
$ make clean
```

6.5 Generate Documentation

These TEEP-Device documentations in pdf and html format are generated by using Doxygen.

6.5.1 Start the container

```
docker run -it --rm -v $(pwd):/home/user/teep-device aistcpsec/teep-dev:doxygen
```

6.5.2 Generate pdf and html documentation

```
$ make docs
```

Location of created documentation.

```
docs/teep-device.pdf
docs/teep-device_readme_html.tar.gz
```

7 Build TEEP-Device without Docker and for Development boards

Clone the TEEP-Device's source code and build it for Keystone, OP-TEE and SGX.

To build TEEP-Device for any TEE, the preparation of the TA-Ref sdk has to be done in advance. The detailed steps of building TA-Ref can be found in the TA-Ref documentation. Export the path of TA-Ref in the following environment variable.

```
$ export TAREF_DIR=<ta-ref dir>
```

7.1 Prerequisite

Have tested on Ubuntu 20.04.

Install packages for compiling.

```
sudo apt-get -y install build-essential git autoconf automake cmake git wget curl expect python3-pip
       libcap-dev debian-ports-archive-keyring locales
openssh-client openssh-server file libcurl4-gnutls-dev libjansson-dev rsync ntpdate usbutils
       pciutils net-tools iproute2 vim e2fsprogs
```

Install suit-tool

The TEEP Messages use SUIT Manifest format for acquiring TCs. The suit-tools is used in TEEP-Device for parsing and handling SUIT Manifests.

```
# Cloning suit-tool
git clone https://git.gitlab.arm.com/research/ietf-suit/suit-tool.git
# Checkout the version of suit-tools compatible with current TEEP-Device
cd suit-tool
git checkout ca66a97bac153864617e7868e44f4b409e3e6ed4 -b for-teep-device
python3 -m pip install --upgrade .
```

7.2 Run tamproto (TAM Server) - Required by all Kestone/OP-TEE/SGX

Running a tamproto on a separate terminal is required as when the TEEP_Device is executed which communicates with the tamproto server to install the TC's.

```
# Clone the tamproto repo and checkout master branch
$ git clone https://github.com/ko-isobe/tamproto.git
$ cd tamproto
$ git checkout master
$ docker-compose build
$ docker-compose up
```

Once the TAM server is up, it will wait for incoming packets from TEEP-Device.

```
naga@smartie:~/Aist_Dev/test/tamproto$ docker-compose up
Starting tamproto_tam_api_1 ... done
Attaching to tamproto_tam_api_1
tam_api_1  | { ’supported-cipher-suites’: 1,
tam_api_1  |   challenge: 2,
tam_api_1  |   versions: 3,
tam_api_1  |   ’ocsp-data’: 4,
tam_api_1  |   ’selected-cipher-suite’: 5,
tam_api_1  |   ’selected-version’: 6,
tam_api_1  |   evidence: 7,
tam_api_1  |   ’tc-list’: 8,
tam_api_1  |   ’ext-list’: 9,
tam_api_1  |   ’manifest-list’: 10,
tam_api_1  |   msg: 11,
tam_api_1  |   ’err-msg’: 12,
tam_api_1  |   ’evidence-format’: 13,
tam_api_1  |   ’requested-tc-list’: 14,
tam_api_1  |   ’unneeded-tc-list’: 15,
tam_api_1  |   ’component-id’: 16,
tam_api_1  |   ’tc-manifest-sequence-number’: 17,
tam_api_1  |   ’have-binary’: 18,
tam_api_1  |   ’suit-reports’: 19,
tam_api_1  |   token: 20,
tam_api_1  |   ’supported-freshness-mechanisms’: 21 }
tam_api_1  | Loading KeyConfig
tam_api_1  | { TAM_priv: ’test-jw_tsm_identity_private_tam-mytam-private.jwk’,
tam_api_1  |   TAM_pub: ’test-jw_tsm_identity_tam-mytam-public.jwk’,
tam_api_1  |   TEE_priv: ’teep.jwk’,
tam_api_1  |   TEE_pub: ’teep.jwk’ }
tam_api_1  | Load key TAM_priv
tam_api_1  | Load key TAM_pub
tam_api_1  | Load key TEE_priv
tam_api_1  | Load key TEE_pub
tam_api_1  | Key binary loaded
tam_api_1  | 192.168.11.4
tam_api_1  | Express HTTP  server listening on port 8888
tam_api_1  | Express HTTPS server listening on port 8443
tam_api_1  | GET /api/ 200 5.239 ms - 24
```

Please keep opening this terminal running tamproto. Cloning and building the TEEP-Device will be done on separate terminals.

7.3 Keystone

Instruction to build `TEEP-Device` with Keystone. The Keystone and its supporting sources must be built and installed on the build environment beforehand. Refer to the Keystone section of the "Preparation before building TA-Ref without Docker" in the TA-Ref document.

7.3.1 Clone and Build

Prepare the environment setup.

```
$ export TEE=keystone
$ export KEYSTONE_DIR=<path to keystone dir>
$ export PATH=$PATH:$KEYSTONE_DIR/riscv/bin
$ export KEYEDGE_DIR=<path to keyedge dir>
```

Clone and Build

```
# Clone the TEEP-Device
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
# Build the TEEP-Device
$ make
```

7.3.2 Run hello-app & teep-broker-app on QEMU Environment

To check TEEP-Device on Keystone, we need to run TAM server on PC.

```
# After the successful build
$ make run-sample-session
```

Trimmed output printing 'Hello TEEP from TEE!'

```
Welcome to Buildroot
buildroot login: root
Password: sifive
#### insmod keystone-driver.ko || echo ’err”or’
[    5.350967] keystone_driver: loading out-of-tree module taints kernel.
[    5.358283] keystone_enclave: keystone enclave v1.0.0
#### cd /root/teep-device
#### ls -l
total 1367
-rwxr-xr-x    1 root     root         98088 Feb 15  2022 eyrie-rt
-rwxr-xr-x    1 root     root        437480 Feb 15  2022 hello-app
-rwxr-xr-x    1 root     root        152016 Feb 15  2022 hello-ta
-rwxr-xr-x    1 root     root        247416 Feb 15  2022 teep-agent-ta
-rwxr-xr-x    1 root     root        470568 Feb 15  2022 teep-broker-app
#### ./hello-app hello-ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
Hello TEEP from TEE!
#### ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[Keystone SDK] /home/user/keystone/sdk/src/host/ElfFile.cpp:26 : file does not exist -
       8d82573a-926d-4754-9353-32dc29997f74.ta
[Keystone SDK] /home/user/keystone/sdk/src/host/Enclave.cpp:209 : Invalid enclave ELF
./hello-app: Unable to start enclave
#### ./teep-broker-app --tamurl http://tamproto_tam_api_1:8888/api/tam_cbor
teep-broker.c compiled at Feb 15 2022 10:07:55
uri = http://tamproto_tam_api_1:8888/api/tam_cbor, cose=0, talist=
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799c6000-0x179c00000 (2280 KB), va 0xffffffff001c6000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
<output trimmed>
#### ls -l
total 1517
-rw-------    1 root     root        152016 Jan  1 00:00 8d82573a-926d-4754-9353-32dc29997f74.ta
-rwxr-xr-x    1 root     root         98088 Feb 15  2022 eyrie-rt
-rwxr-xr-x    1 root     root        437480 Feb 15  2022 hello-app
-rwxr-xr-x    1 root     root        152016 Feb 15  2022 hello-ta
-rwxr-xr-x    1 root     root        247416 Feb 15  2022 teep-agent-ta
-rwxr-xr-x    1 root     root        470568 Feb 15  2022 teep-broker-app
#### ./hello-app 8d82573a-926d-4754-9353-32dc29997f74.ta eyrie-rt
[debug] UTM : 0xffffffff80000000-0xffffffff80100000 (1024 KB) (boot.c:127)
[debug] DRAM: 0x179800000-0x179c00000 (4096 KB) (boot.c:128)
[debug] FREE: 0x1799bd000-0x179c00000 (2316 KB), va 0xffffffff001bd000 (boot.c:133)
[debug] eyrie boot finished. drop to the user land ... (boot.c:172)
Hello TEEP from TEE!
97f74.ta.secstor.plain-4754-9353-32dc29997f74.ta 8d82573a-926d-4754-9353-32dc2999
cmp: 8d82573a-926d-4754-9353-32dc29997f74.ta.secstor.plain: No such file or directory
####  done
```

7.3.3 Run hello-app and teep-broker-app on RISC-V Unleashed

To check TEEP-Device on Unleashed, we need to run TAM server (refer above to run tamproto) and networking with Unleashed dev board

7.3.3.1 Copy the hello-app and teep-broker-app binaries to Unleashed Manual Copy

- Connect to Unleased over serial console then assign IP address `ifconfig eth0 192.168.0.6`
- Copy the binaries from build PC over SSH (user:root, password: sifive)

Here `192.168.0.6` is IP Address of Unleased board

```
$ scp platform/keystone/build/hello-ta/hello-ta root@192.168.0.6:/root/teep-device
$ scp platform/keystone/build/hello-app/hello-app root@192.168.0.6:/root/teep-device
$ scp platform/keystone/build/teep-agent-ta/teep-agent-ta root@192.168.0.6:/root/teep-device
$ scp platform/keystone/build/teep-broker-app/teep-broker-app root@192.168.0.6:/root/teep-device
$ scp $KEYSTONE_DIR/sdk/rts/eyrie/eyrie-rt root@192.168.0.6:/root/teep-device
$ scp platform/keystone/build/libteep/ree/mbedtls/library/lib* root@192.168.0.6:/usr/lib/
$ scp platform/keystone/build/libteep/ree/libwebsockets/lib/lib* root@192.168.0.6:/usr/lib/
```

Write to SD card

Please follow below steps to write the TEEP-Device binaries to SD-card

- Insert SD card to your PC for Unleashed
- Edit `platform/keystone/script/sktinst.sh`
- Check SD-card device name detected on yor PC and fix `prefix=?`
- `export prefix=/dev/mmcblk0`
- execute `script/sktinst.sh` as follows
- `cd platform/keystone; script/sktinst.sh`
- Move the sd to unleashed board and boot it

7.3.3.2 Run hello-app and teep-broker-app on Unleased

There are two methods to connect to Unleased.

- Serial Port using minicom (/dev/ttyUSB0)
- Over SSH: `ssh root@192.168.0.6`; password is `sifive`

Setup environment in Unleashed (create /root/env.sh file and add following lines)

```
$ export PATH=$PATH:/root/teep-device
$ export TAM_IP=tamproto_tam_api_1
$ export TAM_PORT=8888
$ insmod keystone-driver.ko
```

Run hello-app

```
$ source env.sh
[ 2380.618514] keystone_driver: loading out-of-tree module taints kernel.
[ 2380.625305] keystone_enclave: keystone enclave v0.2
$ cd teep-broker/
$ ./hello-app hello-ta eyrie-rt
Hello TEEP from TEE!
$
```

Run teep-broker-app

Use the TAM server IP address (i.e 192.168.11.4)

```
$ ./teep-broker-app --tamurl http://192.168.11.4:8888/api/tam_cbor
```

Upon execution, you see following log

```
teep-bro[ 2932.269897] ------------[ cut here ]------------
[ 2932.274191] WARNING: CPU: 4 PID: 164
[ 2932.287053] Modules linked in: keystone_driver(O)
[ 2932.291716] CPU: 4 PID: 164 Comm: teep-broker-app Tainted: G
[ 2932.301867] Call Trace:
[ 2932.304314] [<0000000036e46dc0>] walk_stackframe+0x0/0xa2
[ 2932.309686] [<00000000893dfe1c>] show_stack+0x26/0x34
[ 2932.314725] [<00000000c57ed7ce>] dump_stack+0x5e/0x7c
[ 2932.319759] [<00000000a68ce031>] __warn+0xca/0xe0
[ 2932.324445] [<00000000bec1f8a6>] warn_slowpath_null+0x2c/0x3e
[ 2932.330176] [<00000000e8c56bf2>] __alloc_pages_nodemask+0x14c/0x8da
[ 2932.336426] [<00000000ec1f9596>] __get_free_pages+0xc/0x52
[ 2932.341920] [<000000003e8cccc8>] epm_init+0x158/0x1a0 [keystone_driver]
[ 2932.348502] [<0000000032e4188b>] create_enclave+0x56/0xb0 [keystone_driver]
[ 2932.355447] [<000000008a656a96>] keystone_create_enclave+0x16/0x40 [keystone_driver]
[ 2932.363174] [<000000003bbf2147>] keystone_ioctl+0x132/0x164 [keystone_driver]
[ 2932.370288] [<00000000755f7993>] do_vfs_ioctl+0x76/0x4f4
[ 2932.375582] [<00000000b88b9c1d>] SyS_ioctl+0x36/0x60
[ 2932.380533] [<00000000aae667a5>] check_syscall_nr+0x1e/0x22
[ 2932.386132] ---[ end trace 66814e3a8c80ec12 ]---
ker.c compiled at Feb 16 2021 11:17:21
uri = http://192.168.11.4:8888/api/tam_cbor, cose=0, talist=
[1970/01/01 00:48:56:0796] NOTICE: POST: http://192.168.11.4:8888/api/tam_cbor
[1970/01/01 00:48:56:0798] NOTICE: (hexdump: zero length)
[1970/01/01 00:48:56:0801] NOTICE: created client ssl context for default
[1970/01/01 00:48:56:0802] NOTICE: http://192.168.11.4:8888/api/tam_cbor
[1970/01/01 00:48:56:0861] NOTICE:
[1970/01/01 00:48:56:0862] NOTICE: 0000: 83 01 A4 01 81 01 03 81 00 14 1A 77 77 77 77 04
       ...........wwww.
[1970/01/01 00:48:56:0862] NOTICE: 0010: 43 01 02 03 02                                     C....
[1970/01/01 00:48:56:0862] NOTICE:
[1970/01/01 00:48:56:0871] NOTICE: POST: http://192.168.11.4:8888/api/tam_cbor
[1970/01/01 00:48:56:0871] NOTICE:
[1970/01/01 00:48:56:0871] NOTICE: 0000: 82 02 A4 14 1A 77 77 77 77 08 80 0E 80 0F 80
       .....wwww......
[1970/01/01 00:48:56:0872] NOTICE:
[1970/01/01 00:48:56:0873] NOTICE: created client ssl context for default
[1970/01/01 00:48:56:0874] NOTICE: http://192.168.11.4:8888/api/tam_cbor
[1970/01/01 00:48:56:0962] NOTICE:
[1970/01/01 00:48:56:0962] NOTICE: 0000: 82 03 A2 0A 81 59 01 37 A2 02 58 72 81 58 6F D2
       .....Y.7..Xr.Xo.
[1970/01/01 00:48:56:0963] NOTICE: 0010: 84 43 A1 01 26 A0 58 24 82 02 58 20 75 80 7C 54
       .C..&.X$..X u.|T
[1970/01/01 00:48:56:0963] NOTICE: 0020: 62 40 D2 14 E5 7B D5 C4 6A 7C E5 2D ED B0 3D 0E
       b@...{..j|.-..=.
[1970/01/01 00:48:56:0964] NOTICE: 0030: CC 80 75 F3 F7 E0 65 B3 60 CE AD 85 58 40 54 81
       ..u...e.`...X@T.
[1970/01/01 00:48:56:0964] NOTICE: 0040: 49 CD CA D8 17 72 CC EA 61 4A 19 99 05 AB 97 33
       I....r..aJ.....3
[1970/01/01 00:48:56:0965] NOTICE: 0050: EA 48 D7 1F 13 AE 33 0D 47 FF F5 B8 6C 5C 9B 7A
       .H....3.G...l\.z
[1970/01/01 00:48:56:0965] NOTICE: 0060: BB 12 BC 2D FE 9C 20 6A C8 7F E2 28 58 74 E0 74    ...-..
       j...(Xt.t
[1970/01/01 00:48:56:0965] NOTICE: 0070: A3 BD C4 DA B9 20 C4 37 35 8F 67 46 90 76 03 58    .....
       .75.gF.v.X
[1970/01/01 00:48:56:0966] NOTICE: 0080: BE A5 01 01 02 01 03 58 60 A2 02 44 81 81 41 00
       .......X`..D..A.
[1970/01/01 00:48:56:0966] NOTICE: 0090: 04 58 56 86 14 A4 01 50 FA 6B 4A 53 D5 AD 5F DF
       .XV....P.kJS.._.
[1970/01/01 00:48:56:0967] NOTICE: 00A0: BE 9D E6 63 E4 D4 1F FE 02 50 14 92 AF 14 25 69
       ...c.....P....%i
[1970/01/01 00:48:56:0967] NOTICE: 00B0: 5E 48 BF 42 9B 2D 51 F2 AB 45 03 58 24 82 02 58
       ^H.B.-Q..E.X$..X
[1970/01/01 00:48:56:0968] NOTICE: 00C0: 20 00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE
       .."3DUfw.......
[1970/01/01 00:48:56:0968] NOTICE: 00D0: FF 01 23 45 67 89 AB CD EF FE DC BA 98 76 54 32
       ..#Eg........vT2
[1970/01/01 00:48:56:0969] NOTICE: 00E0: 10 0E 19 87 D0 01 F6 02 F6 09 58 4E 86 13 A1 15
       ..........XN....
[1970/01/01 00:48:56:0969] NOTICE: 00F0: 78 44 68 74 74 70 3A 2F 2F 31 39 32 2E 31 36 38
       xDhttp://192.168
[1970/01/01 00:48:56:0970] NOTICE: 0100: 2E 31 31 2E 33 3A 38 38 38 38 2F 54 41 73 2F 38
       .0.5:8888/TAs/8
[1970/01/01 00:48:56:0970] NOTICE: 0110: 64 38 32 35 37 33 61 2D 39 32 36 64 2D 34 37 35
       d82573a-926d-475
[1970/01/01 00:48:56:0971] NOTICE: 0120: 34 2D 39 33 35 33 2D 33 32 64 63 32 39 39 39 37
       4-9353-32dc29997
[1970/01/01 00:48:56:0971] NOTICE: 0130: 66 37 34 2E 74 61 15 F6 03 F6 0A 43 82 03 F6 14
       f74.ta.....C....
[1970/01/01 00:48:56:0972] NOTICE: 0140: 1A 77 77 77 78                                     .wwwx
[1970/01/01 00:48:56:0972] NOTICE:
[1970/01/01 00:48:56:0983] NOTICE:
GET: http://192.168.11.4:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[1970/01/01 00:48:56:0984] NOTICE: created client ssl context for default
[1970/01/01 00:48:56:0985] NOTICE:
http://192.168.11.4:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
teep_message_unwrap_ta_image: msg len 234110
Decrypt
```

```
Decrypt OK: length 174887
Verify
Signature OK 0 130552
ta_store_install: ta_image_len = 130552 ta_name=8d82573a-926d-4754-9353-32dc29997f74
[1970/01/01 00:49:01:9453] NOTICE: POST: http://192.168.11.4:8888/api/tam_cbor
[1970/01/01 00:49:01:9454] NOTICE:
[1970/01/01 00:49:01:9454] NOTICE: 0000: 82 05 A1 14 1A 77 77 77 77
       .....wwww
[1970/01/01 00:49:01:9454] NOTICE:
[1970/01/01 00:49:01:9456] NOTICE: created client ssl context for default
[1970/01/01 00:49:01:9457] NOTICE: http://192.168.11.4:8888/api/tam_cbor
[1970/01/01 00:49:01:9505] NOTICE: (hexdump: zero length)
```

7.4 OP-TEE

Instruction to build `TEEP-Device` with OP-TEE. The OP-TEE and its supporting sources must be built and installed on the build environment beforehand. Refer to the OP-TEE section of the "Preparation before building TA-Ref without Docker" in the TA-Ref document.

7.4.1 Clone and Build

Prepare the environment setup

```
$ export TEE=optee
$ export OPTEE_DIR=<optee_dir>
$ export PATH=$PATH:$OPTEE_DIR/toolchains/aarch64/bin:$OPTEE_DIR/toolchains/aarch32/bin
```

Clone and Build

```
# Clone the TEEP-Device
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
# Build the TEEP-Device
$ make
```

7.4.2 Run hello-app & teep-broker-app on QEMU Environment

To check TEEP-Device on OP-TEE, we need to run TAM server on PC.

```
# Install the TA on qemu
$ make optee_install_qemu
# After the successful build
# Test the TEEP-Device
$ make run-sample-session
```

Trimmed output of the test

```
make -C sample run-session TAM_URL=http://172.17.0.32:8888
make[1]: Entering directory ’/builds/rinkai/teep-device/sample’
make -C /builds/rinkai/teep-device/sample/../hello-tc/build-optee
 SOURCE=/builds/rinkai/teep-device/sample/../hello-tc upload-download-manifest
make[2]: Entering directory ’/builds/rinkai/teep-device/hello-tc/build-optee’
curl http://172.17.0.32:8888/panel/upload \
    -F "file=@/builds/rinkai/teep-device/hello-tc/build-optee/../../build/optee/hello-tc/signed-download-tc.suit;f
    ilename=integrated-payload-manifest.cbor"
...
...
...
```

```
cd /home/user/optee/out/bin && \
    QEMU=/home/user/optee/qemu/aarch64-softmmu/qemu-system-aarch64 \
    QEMU_SMP=2 \
    TAM_URL=http://172.17.0.32:8888 \
    ROOTFS=/builds/rinkai/teep-device/sample/../build/optee/rootfs.cpio.gz \
    expect /builds/rinkai/teep-device/sample/session/test-optee.expect
Starting QEMU...
..
..
FI stub: Booting Linux Kernel...
EFI stub: EFI_RNG_PROTOCOL unavailable, no randomness supplied
EFI stub: Using DTB from configuration table
EFI stub: Exiting boot services and installing virtual address map...
Starting syslogd: OK
Starting klogd: OK
Initializing random number generator... [    3.248115] random: dd: uninitialized urandom read (512
       bytes read)
done.
Set permissions on /dev/tee*: OK
Set permissions on /dev/ion: OK
Create/set permissions on /data/tee: OK
Starting tee-supplicant: OK
Starting network: OK
Starting network (udhcpc): OK
Welcome to Buildroot, type root or test to login
buildroot login: root
#  done, guest is booted.
export LD_LIBRARY_PATH=/lib:/lib/arm-linux-gnueabihf:/lib/optee_armtz:/usr/lib
# cd teep-broker
# ls -l
total 4228
-rwxr-xr-x    1 root     root           567 Nov 24 07:27 cp_ta_to_tamproto.sh
-rwxr-xr-x    1 root     root           146 Nov 24 07:27 env.sh
-rwxr-xr-x    1 root     root           290 Nov 24 07:27 get-ip.sh
-rwxr-xr-x    1 root     root         14112 Nov 24 07:27 hello-app
-rwxr-xr-x    1 root     root            65 Nov 24 07:27 itc.sh
-rwxr-xr-x    1 root     root           116 Nov 24 07:27 rtc.sh
-rwxr-xr-x    1 root     root           134 Nov 24 07:27 showtamurl.sh
-rwxr-xr-x    1 root     root       4285528 Nov 24 07:27 teep-broker-app
# ./hello-app
hello-app: TEEC_Opensession failed with code 0xffff0008 origin 0x3
# ./teep-broker-app --tamurl http://172.17.0.32:8888/api/tam_cbor
teep-broker.c compiled at Nov 24 2022 07:27:53
uri = http://172.17.0.32:8888/api/tam_cbor, cose=0, talist=
[2022/11/24 07:28:26:9408] N: POST: http://172.17.0.32:8888/api/tam_cbor
[2022/11/24 07:28:26:9425] N: (hexdump: zero length)
[2022/11/24 07:28:26:9506] N: http://172.17.0.32:8888/api/tam_cbor
[2022/11/24 07:28:26:0224] N:
[2022/11/24 07:28:26:0234] N: 0000: 83 01 A5 01 81 01 03 81 00 04 43 01 02 05 14 48
       .........C....H
[2022/11/24 07:28:26:0243] N: 0010: 77 77 77 77 77 77 77 77 15 81 00 02                wwwwwwww....

[2022/11/24 07:28:26:0250] N:
[2022/11/24 07:28:26:0374] N: POST: http://172.17.0.32:8888/api/tam_cbor
[2022/11/24 07:28:26:0379] N:
[2022/11/24 07:28:26:0382] N: 0000: 82 02 A4 14 48 77 77 77 77 77 77 77 77 08 80 0E
       ....Hwwwwwwww...
[2022/11/24 07:28:26:0389] N: 0010: 80 0F 80                                           ...

[2022/11/24 07:28:26:0395] N:
[2022/11/24 07:28:26:0417] N: http://172.17.0.32:8888/api/tam_cbor
[2022/11/24 07:28:27:0742] N:
[2022/11/24 07:28:27:0746] N: 0000: 82 03 A2 0A 81 59 01 5F D8 6B A2 02 58 73 82 58
       .....Y._.k..Xs.X
[2022/11/24 07:28:27:0751] N: 0010: 24 82 2F 58 20 D7 FC F7 75 1E CB 77 96 39 A4 0F    $./X
       ...u..w.9..
[2022/11/24 07:28:27:0758] N: 0020: 58 66 56 EF D3 08 7D 31 ED C3 C4 5B EE DD 9E 95
       XfV...}1...[....
[2022/11/24 07:28:27:0767] N: 0030: 38 CE 0D 3E 8A 58 4A D2 84 43 A1 01 26 A0 F6 58
       8..>.XJ..C..&..X
[2022/11/24 07:28:27:0774] N: 0040: 40 6F D3 76 AE AE CF F3 BC E7 7E 60 E1 22 0A 20
       @o.v......~`..
[2022/11/24 07:28:27:0780] N: 0050: 1C 3C 10 3F 85 BE 71 A7 10 E5 6D C1 C5 0A A6 C6
       .<.?..q...m.....
[2022/11/24 07:28:27:0786] N: 0060: 47 D4 D4 EE DD 20 2D 08 EA 4F 74 6F 48 FF 3D D3    G....
       -..OtoH.=.
[2022/11/24 07:28:27:0793] N: 0070: 32 B4 60 18 95 15 6A 5D 25 12 EF E8 8B 35 CE CD
       2.`...j]%....5..
[2022/11/24 07:28:27:0799] N: 0080: 1C 03 58 E3 A5 01 01 02 01 03 58 86 A2 02 81 84
       ..X.......X.....
[2022/11/24 07:28:27:0808] N: 0090: 4B 54 45 45 50 2D 44 65 76 69 63 65 48 53 65 63
       KTEEP-DeviceHSec
[2022/11/24 07:28:27:0815] N: 00A0: 75 72 65 46 53 50 8D 82 57 3A 92 6D 47 54 93 53
       ureFSP..W:.mGT.S
[2022/11/24 07:28:27:0821] N: 00B0: 32 DC 29 99 7F 74 42 74 61 04 58 56 86 14 A4 01
```

```
       2.)..tBta.XV....
[2022/11/24 07:28:27:0827] N: 00C0: 50 FA 6B 4A 53 D5 AD 5F DF BE 9D E6 63 E4 D4 1F
       P.kJS.._....c...
[2022/11/24 07:28:27:0833] N: 00D0: FE 02 50 14 92 AF 14 25 69 5E 48 BF 42 9B 2D 51
       ..P....%i^H.B.-Q
[2022/11/24 07:28:27:0840] N: 00E0: F2 AB 45 03 58 24 82 2F 58 20 00 11 22 33 44 55    ..E.X$./X
       ..3DU
[2022/11/24 07:28:27:0847] N: 00F0: 66 77 88 99 AA BB CC DD EE FF 01 23 45 67 89 AB
       fw.........#Eg..
[2022/11/24 07:28:27:0853] N: 0100: CD EF FE DC BA 98 76 54 32 10 0E 19 87 D0 01 0F
       ......vT2.......
[2022/11/24 07:28:27:0859] N: 0110: 02 0F 09 58 4D 86 13 A1 15 78 43 68 74 74 70 3A
       ...XM....xChttp:
[2022/11/24 07:28:27:0865] N: 0120: 2F 2F 31 37 32 2E 31 37 2E 30 2E 33 32 3A 38 38
       //172.17.0.32:88
[2022/11/24 07:28:27:0871] N: 0130: 38 38 2F 54 41 73 2F 38 64 38 32 35 37 33 61 2D
       88/TAs/8d82573a-
[2022/11/24 07:28:27:0877] N: 0140: 39 32 36 64 2D 34 37 35 34 2D 39 33 35 33 2D 33
       926d-4754-9353-3
[2022/11/24 07:28:27:0884] N: 0150: 32 64 63 32 39 39 39 37 66 37 34 2E 74 61 15 02
       2dc29997f74.ta..
[2022/11/24 07:28:27:0890] N: 0160: 03 0F 0A 43 82 03 0F 14 48 AB A1 A2 A3 A4 A5 A6
       ...C....H.......
[2022/11/24 07:28:27:0896] N: 0170: A7                                                 .

[2022/11/24 07:28:27:0901] N:
[2022/11/24 07:28:27:6609] N: GET: http://172.17.0.32:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 07:28:27:6632] N: http://172.17.0.32:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 07:28:27:7968] N: POST: http://172.17.0.32:8888/api/tam_cbor
[2022/11/24 07:28:27:7973] N:
[2022/11/24 07:28:27:7977] N: 0000: 82 05 A1 14 48 77 77 77 77 77 77 77 77             ....Hwwwwwwww

[2022/11/24 07:28:27:7982] N:
[2022/11/24 07:28:27:7993] N: http://172.17.0.32:8888/api/tam_cbor
[2022/11/24 07:28:27:8228] N: (hexdump: zero length)
# ls -l
total 4228
-rwxr-xr-x    1 root     root           567 Nov 24 07:27 cp_ta_to_tamproto.sh
-rwxr-xr-x    1 root     root           146 Nov 24 07:27 env.sh
-rwxr-xr-x    1 root     root           290 Nov 24 07:27 get-ip.sh
-rwxr-xr-x    1 root     root         14112 Nov 24 07:27 hello-app
-rwxr-xr-x    1 root     root            65 Nov 24 07:27 itc.sh
-rwxr-xr-x    1 root     root           116 Nov 24 07:27 rtc.sh
-rwxr-xr-x    1 root     root           134 Nov 24 07:27 showtamurl.sh
-rwxr-xr-x    1 root     root       4285528 Nov 24 07:27 teep-broker-app
# ./hello-app
#  done
..
M/TA: TTRC:verifying signature of suit manifest
M/TA: TTRC:verify OK
M/TA: TTRC:command: 20
M/TA: TTRC:execute suit-set-parameters
M/TA: TTRC:command: 1
M/TA: TTRC:execute suit-condition-vendor-identifier
M/TA: TTRC:command: 2
M/TA: TTRC:execute suit-condition-class-identifier
M/TA: TTRC:command: 19
M/TA: TTRC:execute suit-set-parameters
M/TA: TTRC:command: 21
M/TA: TTRC:execute suit-directive-fetch
M/TA: TTRC:fetch_and_store component
M/TA: TTRC:component download 55976
M/TA: TTRC:ta-store.c: store_component() store component
M/TA: TTRC:  device   = TEEP-Device
M/TA: TTRC:  storage  = SecureFS
M/TA: TTRC:  filename = 8d82573a-926d-4754-9353-32dc29997f74.ta
M/TA: TTRC:  image_len = 55976
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 6e256cba-fc4d-4941-ad09-2ca1860342dd
D/TC:? 0 tee_ta_init_pseudo_ta_session:296 Open secstor_ta_mgmt
D/TC:? 0 tee_ta_init_pseudo_ta_session:310 secstor_ta_mgmt : 6e256cba-fc4d-4941-ad09-2ca1860342dd
D/TC:? 0 install_ta:99 Installing 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_close_session:499 csess 0xc37ecfb0 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
M/TA: TTRC:finish fetch
M/TA: TTRC:command: 3
M/TA: TTRC:execute suit-condition-image-match
M/TA: TTRC:end of command seq
D/TC:? 0 tee_ta_close_session:499 csess 0xc37eeb40 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0xc37eeae0)
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 load_ldelf:704 ldelf load address 0x40006000
D/LD:  ldelf:134 Loading TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_init_session_with_context:573 Re-open TA 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 8d82573a-926d-4754-9353-32dc29997f74 (Secure
       Storage TA)
D/TC:? 0 system_open_ta_binary:260 res=0x0
D/LD:  ldelf:169 ELF (8d82573a-926d-4754-9353-32dc29997f74) at 0x4004b000
D/TC:? 0 tee_ta_close_session:499 csess 0xc37ec610 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
Hello TEEP from TEE!
D/TC:? 0 tee_ta_close_session:499 csess 0xc37ece10 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0xc37ecdb0)
! fgrep 'ERR:' /home/user/optee/out/bin/serial1.log
fgrep 'Hello TEEP from TEE!' /home/user/optee/out/bin/serial1.log
Hello TEEP from TEE!
make[1]: Leaving directory '/home/user/teep-device/sample'
```

7.4.3 Run hello-app and teep-broker-app on Raspberry PI 3

The RPI3 board needs the following to boot. 1) Kernal Image - kernel8.img 2) DTB file for RPI3 - bcm2710-rpi-3-b-plus.dtb 3) Rootfs file system - arm64-20.04-rootfs-teep-device.tar.xz

You can copy the Kernal Image and DTB file from the **aistcpsec/tee-dev:optee-3.10.0_rpi3** image. It will located inside /home/user/optee/out-br/target/boot folder.

The rootfs file system can be downloaded from the docker image **aistcpsec/teep-dev:rootfs_teep-device_rpi3↩ _netboot_nfs** It will be available in the / folder. Tar file name is arm64-20.04-rootfs-teep-device.tar.xz.

Please copy the above files from docker to your local using docker cp command.

Partition SD Card

Partition the SD card into two partitions Partition 1 - Boot Partition - Place the Kernal Image file and DTB File Partition 2 - Rootfs file system - Copy the extracted arm64-20.04-rootfs-teep-device.tar.xz contents.

Write to SD card
Please follow below steps to write the TEEP-Device binaries to SD-card

- Insert SD card to your PC for Unleashed
- Copy the binaries to SD card
- Move the SD card to Raspberry PI 3 board and boot it

7.4.3.1 Run hello-app and teep-broker-app on Raspberry PI 3 There are two methods to connect to Raspberry PI 3.

- Serial Port using minicom (/dev/ttyUSB0)
- Over SSH: `ssh root@<rpi3_ip_address>`

In the below steps, let us consider the IP address of RPI3 connected system is 192.168.100.118 and the IP address of RPI3 is 192.168.100.114

Also, Tamproto Server is required to test the TEEP-Device. We can start the tamproto in the PC or we can start inside RPI3 itself. In our case, we will start the tamproto server inside the RPI3.

Access the RPI3 Terminal - Using Minicom

When RPI3 is booting, we can access the RPI3 using the minicom. This is access from the PC to which the RPI3 is connected. Also /dev/ttyUSB0 is not fixed, it may be /dev/ttyUSB1 or 2 etc.

```
$ minicom -D /dev/ttyUSB0
root@arm64-ubuntu:~# NOTICE:  Booting Trusted Firmware
NOTICE:  BL1: v2.2(debug):v2.2
NOTICE:  BL1: Built : 03:31:10, Nov 15 2022
INFO:    BL1: RAM 0x100ee000 - 0x100f7000
INFO:    BL1: cortex_a53: CPU workaround for 843419 was applied
INFO:    BL1: cortex_a53: CPU workaround for 855873 was applied
NOTICE:  rpi3: Detected: Raspberry Pi 3 Model B+ (1GB, Sony, UK) [0x00a020d3]
INFO:    BL1: Loading BL2
INFO:    Loading image id=1 at address 0x100b4000
INFO:    Image id=1 loaded: 0x100b4000 - 0x100bc410
NOTICE:  BL1: Booting BL2
INFO:    Entry point address = 0x100b4000
INFO:    SPSR = 0x3c5
NOTICE:  BL2: v2.2(debug):v2.2
NOTICE:  BL2: Built : 03:31:10, Nov 15 2022
INFO:    BL2: Doing platform setup
INFO:    BL2: Loading image id 3
INFO:    Loading image id=3 at address 0x100e0000
INFO:    Image id=3 loaded: 0x100e0000 - 0x100ea078
INFO:    BL2: Loading image id 4
INFO:    Loading image id=4 at address 0x10100000
INFO:    Image id=4 loaded: 0x10100000 - 0x1010001c
INFO:    OPTEE ep=0x10100000
INFO:    OPTEE header info:
INFO:          magic=0x4554504f
INFO:          version=0x2
INFO:          arch=0x1
arm64-ubuntu login: root
Password:
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 4.14.56-v8 aarch64)
```

Access the RPI3 Terminal - Using SSH After RPI3 is booted, we can access the RPI3 terminal using SSH

```
$ ssh root@192.168.100.114
root@192.168.100.114's password:
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 4.14.56-v8 aarch64)
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Wed Aug 31 15:28:32 2022
root@arm64-ubuntu:~#
```

Starting the Tamproto Server

First, lets install npm and node server in RPI3. This is done by executing the ./install_node.sh file.

```
$ # cd /home/user/./install_node.sh
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 15916  100 15916    0     0  42329      0 --:--:-- --:--:-- --:--:-- 42217
=> Downloading nvm as script to '/root/.nvm'
=> Appending nvm source string to /root/.bashrc
=> Appending bash_completion source string to /root/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:
export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"  # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion"  # This loads nvm bash_completion
Installing latest LTS version.
Downloading and installing node v18.12.1...
Downloading https://nodejs.org/dist/v18.12.1/node-v18.12.1-linux-arm64.tar.xz...
########################################### 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v18.12.1 (npm v8.19.2)
Creating default alias: default -> lts/* (-> v18.12.1)
Now using node v18.12.1 (npm v8.19.2)
root@arm64-ubuntu:/home/user#
```

Then install the node modules and start the tamproto server.

```
root@arm64-ubuntu:/home/user# cd tamproto/
root@arm64-ubuntu:/home/user/tamproto# ls
Dockerfile  README.md  app.js       docker-compose.yml  keymanager.js      package.json
       query_response.cbor  suit.cbor        teep-p.js
LICENSE     TAs        config.json  key                 package-lock.json  query_request.cbor
       routes               ta_install.cbor  views
```

```
root@arm64-ubuntu:/home/user/tamproto# npm install
added 154 packages, and audited 155 packages in 23s
13 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
npm notice
npm notice New major version of npm available! 8.19.2 -> 9.1.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.1.2
npm notice Run npm install -g npm@9.1.2 to update!
npm notice
root@arm64-ubuntu:/home/user/tamproto#
root@arm64-ubuntu:/home/user/tamproto# node app.js
{
  'supported-cipher-suites': 1,
  challenge: 2,
  versions: 3,
  'ocsp-data': 4,
  'selected-cipher-suite': 5,
  'selected-version': 6,
  evidence: 7,
  'tc-list': 8,
  'ext-list': 9,
  'manifest-list': 10,
  msg: 11,
  'err-msg': 12,
  'evidence-format': 13,
  'requested-tc-list': 14,
  'unneeded-tc-list': 15,
  'component-id': 16,
  'tc-manifest-sequence-number': 17,
  'have-binary': 18,
  'suit-reports': 19,
  token: 20,
  'supported-freshness-mechanisms': 21
}
Loading KeyConfig
{
  TAM_priv: 'teep_ecP256.jwk',
  TAM_pub: 'teep_ecP256.jwk',
  TEE_priv: 'teep_ecP256.jwk',
  TEE_pub: 'teep_agent_prime256v1_pub.pem'
}
Load key TAM_priv
Load key TAM_pub
Load key TEE_priv
Load key TEE_pub
Key binary loaded
(node:1098) Warning: Accessing non-existent property 'request' of module exports inside circular
       dependency
(Use `node --trace-warnings ...` to show where the warning was created)
192.168.100.114
Express HTTP  server listening on port 8888
Express HTTPS server listening on port 8443
```

Copy the TA's into Tamproto server

Open another terminal using ssh

```
$ cd /home/user/teep-broker
$ ./cp_ta_to_tamproto.sh
```

Run the teep-broker-app

(Inside teep-broker folder)

```
$ ./showtamurl.sh
   --tamurl http://192.168.100.114:8888/api/tam_cbor
$ ./teep-broker-app --tamurl http://192.168.100.114:8888/api/tam_cbor
teep-broker.c compiled at Nov 24 2022 08:14:22
uri = http://192.168.100.114:8888/api/tam_cbor, cose=0, talist=
[2022/11/24 08:54:57:7358] N: POST: http://192.168.100.114:8888/api/tam_cbor
[2022/11/24 08:54:57:7358] N: (hexdump: zero length)
[2022/11/24 08:54:57:7369] N: http://192.168.100.114:8888/api/tam_cbor
[2022/11/24 08:54:57:8365] N:
[2022/11/24 08:54:57:8366] N: 0000: 83 01 A5 01 81 01 03 81 00 04 43 01 02 05 14 48
       ..........C....H
[2022/11/24 08:54:57:8366] N: 0010: 77 77 77 77 77 77 77 77 15 81 00 02                wwwwwwww....

[2022/11/24 08:54:57:8366] N:
```

```
[2022/11/24 08:54:57:8375] N: POST: http://192.168.100.114:8888/api/tam_cbor
[2022/11/24 08:54:57:8375] N:
[2022/11/24 08:54:57:8376] N: 0000: 82 02 A4 14 48 77 77 77 77 77 77 77 77 08 80 0E
       ....Hwwwwwwww...
[2022/11/24 08:54:57:8376] N: 0010: 80 0F 80                                           ...

[2022/11/24 08:54:57:8376] N:
[2022/11/24 08:54:57:8380] N: http://192.168.100.114:8888/api/tam_cbor
[2022/11/24 08:54:57:9134] N:
[2022/11/24 08:54:57:9135] N: 0000: 82 03 A2 0A 81 59 01 5D D8 6B A2 02 58 73 82 58
       .....Y.].k..Xs.X
[2022/11/24 08:54:57:9136] N: 0010: 24 82 2F 58 20 E5 2A E9 E8 AC 01 49 41 2E 3C EB    $./X
       .*....IA.<.
[2022/11/24 08:54:57:9136] N: 0020: E8 8D 6C B7 27 A9 DE D6 42 24 1A FD 39 D5 ED 0E
       ..l.'...B$..9...
[2022/11/24 08:54:57:9136] N: 0030: 51 E8 9A 95 BF 58 4A D2 84 43 A1 01 26 A0 F6 58
       Q....XJ..C..&..X
[2022/11/24 08:54:57:9136] N: 0040: 40 97 C2 E8 79 81 C5 23 6B 63 C5 AF 51 41 6C 43
       @...y..#kc..QAlC
[2022/11/24 08:54:57:9137] N: 0050: F6 9D E8 91 D7 EB AB 73 7A 30 52 A7 74 02 73 01
       .......sz0R.t.s.
[2022/11/24 08:54:57:9137] N: 0060: 0F B6 70 E2 0B 70 D8 3B CF C3 31 8A 26 39 D0 4D
       ..p..p.;..1.&9.M
[2022/11/24 08:54:57:9137] N: 0070: 1F CF 4B 77 83 F3 24 7F 43 2E 9D 77 00 37 6B CC
       ..Kw..$.C..w.7k.
[2022/11/24 08:54:57:9137] N: 0080: E0 03 58 E1 A5 01 01 02 01 03 58 86 A2 02 81 84
       ..X.......X.....
[2022/11/24 08:54:57:9138] N: 0090: 4B 54 45 45 50 2D 44 65 76 69 63 65 48 53 65 63
       KTEEP-DeviceHSec
[2022/11/24 08:54:57:9138] N: 00A0: 75 72 65 46 53 50 8D 82 57 3A 92 6D 47 54 93 53
       ureFSP..W:.mGT.S
[2022/11/24 08:54:57:9138] N: 00B0: 32 DC 29 99 7F 74 42 74 61 04 58 56 86 14 A4 01
       2.)..tBta.XV....
[2022/11/24 08:54:57:9138] N: 00C0: 50 FA 6B 4A 53 D5 AD 5F DF BE 9D E6 63 E4 D4 1F
       P.kJS.._....c...
[2022/11/24 08:54:57:9139] N: 00D0: FE 02 50 14 92 AF 14 25 69 5E 48 BF 42 9B 2D 51
       ..P....%i^H.B.-Q
[2022/11/24 08:54:57:9139] N: 00E0: F2 AB 45 03 58 24 82 2F 58 20 00 11 22 33 44 55    ..E.X$./X
       .."3DU
[2022/11/24 08:54:57:9139] N: 00F0: 66 77 88 99 AA BB CC DD EE FF 01 23 45 67 89 AB
       fw.........#Eg..
[2022/11/24 08:54:57:9140] N: 0100: CD EF FE DC BA 98 76 54 32 10 0E 19 87 D0 01 0F
       ......vT2.......
[2022/11/24 08:54:57:9140] N: 0110: 02 0F 09 58 4B 86 13 A1 15 78 41 68 74 74 70 3A
       ...XK....xAhttp:
[2022/11/24 08:54:57:9141] N: 0120: 2F 2F 31 32 37 2E 30 2E 30 2E 31 3A 38 38 38 38
       //127.0.0.1:8888
[2022/11/24 08:54:57:9141] N: 0130: 2F 54 41 73 2F 38 64 38 32 35 37 33 61 2D 39 32
       /TAs/8d82573a-92
[2022/11/24 08:54:57:9141] N: 0140: 36 64 2D 34 37 35 34 2D 39 33 35 33 2D 33 32 64
       6d-4754-9353-32d
[2022/11/24 08:54:57:9142] N: 0150: 63 32 39 39 39 37 66 37 34 2E 74 61 15 02 03 0F
       c29997f74.ta....
[2022/11/24 08:54:57:9142] N: 0160: 0A 43 82 03 0F 14 48 AB A1 A2 A3 A4 A5 A6 A7
       .C....H........
[2022/11/24 08:54:57:9142] N:
[2022/11/24 08:54:58:1626] N: GET: http://127.0.0.1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 08:54:58:1630] N: http://127.0.0.1:8888/TAs/8d82573a-926d-4754-9353-32dc29997f74.ta
[2022/11/24 08:54:58:3440] N: POST: http://192.168.100.114:8888/api/tam_cbor
[2022/11/24 08:54:58:3441] N:
[2022/11/24 08:54:58:3441] N: 0000: 82 05 A1 14 48 77 77 77 77 77 77 77 77             ....Hwwwwwwww

[2022/11/24 08:54:58:3441] N:
[2022/11/24 08:54:58:3444] N: http://192.168.100.114:8888/api/tam_cbor
[2022/11/24 08:54:58:3646] N: (hexdump: zero length)
root@arm64-ubuntu:/home/user/teep-broker#
```

Execute the ./hello-app

```
$ ./hello-app
# Check for the "Hello from TEE!" output on Minicom window
D/LD:  ldelf:134 Loading TA 68373894-5bb3-403c-9eec-3114a1f5d3fc
D/TC:? 0 tee_ta_init_session_with_context:573 Re-open TA 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 68373894-5bb3-403c-9eec-3114a1f5d3fc (Secure
       Storage TA)
D/TC:? 0 system_open_ta_binary:260 res=0xffff0008
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 68373894-5bb3-403c-9eec-3114a1f5d3fc (REE)
D/TC:? 0 system_open_ta_binary:260 res=0x0
D/LD:  ldelf:169 ELF (68373894-5bb3-403c-9eec-3114a1f5d3fc) at 0x4008c000
D/TC:? 0 tee_ta_close_session:499 csess 0x101776c0 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
M/TA: TTRC:verifying signature of suit manifest
M/TA: TTRC:verify OK
M/TA: TTRC:command: 20
```

```
M/TA: TTRC:execute suit-set-parameters
M/TA: TTRC:command: 1
M/TA: TTRC:execute suit-condition-vendor-identifier
M/TA: TTRC:command: 2
M/TA: TTRC:execute suit-condition-class-identifier
M/TA: TTRC:command: 19
M/TA: TTRC:execute suit-set-parameters
M/TA: TTRC:command: 21
M/TA: TTRC:execute suit-directive-fetch
M/TA: TTRC:fetch_and_store component
M/TA: TTRC:component download 55976
M/TA: TTRC:ta-store.c: store_component() store component
M/TA: TTRC:  device   = TEEP-Device
M/TA: TTRC:  storage  = SecureFS
M/TA: TTRC:  filename = 8d82573a-926d-4754-9353-32dc29997f74.ta
M/TA: TTRC:  image_len = 55976
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 6e256cba-fc4d-4941-ad09-2ca1860342dd
D/TC:? 0 tee_ta_init_pseudo_ta_session:296 Open secstor_ta_mgmt
D/TC:? 0 tee_ta_init_pseudo_ta_session:310 secstor_ta_mgmt : 6e256cba-fc4d-4941-ad09-2ca1860342dd
D/TC:? 0 install_ta:99 Installing 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_close_session:499 csess 0x101769c0 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
M/TA: TTRC:finish fetch
M/TA: TTRC:command: 3
M/TA: TTRC:execute suit-condition-image-match
M/TA: TTRC:end of command seq
D/TC:? 0 tee_ta_close_session:499 csess 0x10177ec0 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0x10177e60)
D/TC:? 0 tee_ta_init_pseudo_ta_session:283 Lookup pseudo TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 load_ldelf:704 ldelf load address 0x40006000
D/LD:  ldelf:134 Loading TA 8d82573a-926d-4754-9353-32dc29997f74
D/TC:? 0 tee_ta_init_session_with_context:573 Re-open TA 3a2f8978-5dc0-11e8-9c2d-fa7ae01bbebc
D/TC:? 0 system_open_ta_binary:257 Lookup user TA ELF 8d82573a-926d-4754-9353-32dc29997f74 (Secure
       Storage TA)
D/TC:? 0 system_open_ta_binary:260 res=0x0
D/LD:  ldelf:169 ELF (8d82573a-926d-4754-9353-32dc29997f74) at 0x40070000
D/TC:? 0 tee_ta_close_session:499 csess 0x10176070 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
Hello TEEP from TEE!
D/TC:? 0 tee_ta_close_session:499 csess 0x10176870 id 1
D/TC:? 0 tee_ta_close_session:518 Destroy session
D/TC:? 0 destroy_context:298 Destroy TA ctx (0x10176810)
```

7.5 SGX

Instruction to build `TEEP-Device` with SGX. The SGX and its supporting sources must be built and installed on the build environment beforehand. Refer to the SGX section of the "Preparation before building TA-Ref without Docker" in the TA-Ref document.

7.5.1 Clone and Build

As a preparation step, it is required to set up the Intel SGX SDK. Please refer to the preparation steps for building without Docker for SGX in TA-Ref documentation.

```
$ source /opt/intel/sgxsdk/environment
$ export TAREF_DIR=<ta-ref dir>
```

Clone and Build

```
# Clone the TEEP-Device
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
# Build the TEEP-Device
$ export TEE=sgx
$ make
```

7.5.2 Run hello-app & teep-broker-app on Intel SGX

To run TEEP-Device on SGX, confirm that the make is completed first and the tamproto is executed in another terminal.

Unlide Keystone and OP-TEE, it is directly executed on PC without using QEMU.

```
$ cd ~/teep-device
$ make run-sample-session
```

The output would contain the string `Hello TEEP from TEE!` The messages between TEEP-Device and tamproto are printed out as the log on the terminals.

```
main start
Hello TEEP from TEE!
main end
Info: Enclave successfully returned.
```

7.6 Generate Documentation

This documentation (teep-device.pdf) is generated by using Doxygen. To install Doxygen the following procedure is necessary.

7.6.1 Required Packages

Install the following packages on Ubuntu.

```
$ sudo apt -y install doxygen-latex graphviz texlive-full texlive-latex-base latex-cjk-all flex
      bison
```

Above packages required to generate PDF using doxygen.

7.6.2 Build and Install Doxygen

It is using a specific commit since we had a compatibility issue.

```
$ git clone https://github.com/doxygen/doxygen.git
$ cd doxygen
$ git checkout 227952da7562a6f13da2a9d19c3cdc93812bc2de -b for-teep-device
$ mkdir build
$ cd build
$ cmake -G "Unix Makefiles" ..
$ make
$ sudo make install
```

7.6.3 Generate pdf and html documentation

```
$ make docs
```

Location of created documentation.

```
docs/teep-device.pdf
docs/teep-device_readme_html.tar.gz
```

8 Build TEEP-Device without having TEE installed

The building without TEE is prepared for debugging purposes during developing TEEP-Device itself. This method does not require any TEE hardware (TrustZone, SGX and etc) and TEE SDK (Keystone, OP_TEE and SGX SDK) installed in the local machine and it is meant to build and run on TEEP-Device on any x64 PC.

To run TEEP-Device, first we need to run tamproto inside the same host. Let's clone the tamproto and start it.

Prerequisite

Installing required packages.

```
sudo apt-get install -y build-essential git autoconf automake cmake
sudo apt-get install -y libcap-dev python3-pip
```

Installing suit-tools with specific commit which is compatible with current TEEP-Device.

```
git clone https://git.gitlab.arm.com/research/ietf-suit/suit-tool.git
cd suit-tool
git checkout ca66a97bac153864617e7868e44f4b409e3e6ed4 -b for-teep-device
python3 -m pip install --upgrade .
```

Run tamproto

```
# Clone the tamproto repo and checkout master branch
$ git clone https://github.com/ko-isobe/tamproto.git
$ cd tamproto
$ git checkout master
$ docker-compose build
$ docker-compose up &
$ cd ..
```

Trimmed output of starting tamproto. The tamproto is running at 192.168.11.4.

```
tam_api_1  |   TEE_pub: ’teep.jwk’ }
tam_api_1  | Load key TAM_priv
tam_api_1  | Load key TAM_pub
tam_api_1  | Load key TEE_priv
tam_api_1  | Load key TEE_pub
tam_api_1  | Key binary loaded
tam_api_1  | 192.168.11.4
tam_api_1  | Express HTTP  server listening on port 8888
tam_api_1  | Express HTTPS server listening on port 8443
```

Clone TEEP-Device

```
# Clone the teep-device repo and checkout master branch
$ git clone https://github.com/mcd500/teep-device.git
$ cd teep-device
$ git checkout master
# Sync and update the submodules
$ git submodule sync --recursive
$ git submodule update --init --recursive
```

Build

```
# Change to teep-device
$ cd ~/teep-device/
```

```
# set the TEE environment to PC which do not use any of TEEs
$ export TEE=pc
# Build the teep device
$ make
```

After the successful build, run the sample TEEP session with tamproto.

```
$ export TAM_IP=localhost
$ export TAM_URL="http://$TAM_IP:8888"
$ make run-sample-session
```

Trimmed output of the run.

```
$ make run-sample-session
make -C sample run-session
make[1]: Entering directory ’/home/gitlab-runner/projects/teep-device/sample’
make -C /home/gitlab-runner/projects/teep-device/sample/../hello-tc/build-pc SOURCE=/home/gitlab-runner/projects/teep-devi
       upload-embed-manifest
make[2]: Entering directory ’/home/gitlab-runner/projects/teep-device/hello-tc/build-pc’
curl http://localhost:8888/panel/upload \
    -F "file=@/home/gitlab-runner/projects/teep-device/hello-tc/build-pc/../../build/pc/hello-tc/signed-embed-tc.suit;file
tam_api_1  | [
tam_api_1  |   Dirent { name: ’dummy’, [Symbol(type)]: 1 },
tam_api_1  |   Dirent { name: ’dummy2.ta’, [Symbol(type)]: 1 },
tam_api_1  |   Dirent {
tam_api_1  |     name: ’integrated-payload-manifest.cbor’,
tam_api_1  |     [Symbol(type)]: 1
tam_api_1  |   },
tam_api_1  |   Dirent {
tam_api_1  |     name: ’integrated-payload-manifest_hex.txt’,
tam_api_1  |     [Symbol(type)]: 1
tam_api_1  |   },
tam_api_1  |   Dirent { name: ’suit_manifest_exp1.cbor’, [Symbol(type)]: 1 },
tam_api_1  |   Dirent { name: ’suit_manifest_expX.cbor’, [Symbol(type)]: 1 },
tam_api_1  |   Dirent { name: ’tamproto.md’, [Symbol(type)]: 1 }
tam_api_1  | ]
......
tam_api_1  |   ’content-type’: ’application/teep+cbor’
tam_api_1  | }
tam_api_1  | <Buffer 82 05 a1 14 48 77 77 77 77 77 77 77 77>
tam_api_1  | {
tam_api_1  |   TYPE: 5,
tam_api_1  |   token: <Buffer 77 77 77 77 77 77 77 77>,
tam_api_1  |   TOKEN: <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | }
tam_api_1  | TAM ProcessTeepMessage instance
tam_api_1  | TEEP-Protocol:parse
tam_api_1  | {
tam_api_1  |   TYPE: 5,
tam_api_1  |   token: <Buffer 77 77 77 77 77 77 77 77>,
tam_api_1  |   TOKEN: <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | }
tam_api_1  | object
tam_api_1  | *parseSuccessMessage
tam_api_1  | <Buffer 77 77 77 77 77 77 77 77>
tam_api_1  | undefined
tam_api_1  | TAM ProcessTeepMessage response
tam_api_1  | undefined
tam_api_1  | WARNING: Agent may sent invalid contents. TAM responses null.
tam_api_1  | POST /api/tam_cbor 204 1.294 ms - -
fgrep ’store component’ /home/gitlab-runner/projects/teep-device/sample/../build/pc/pctest.log
store componen
```