# draft RISC-V TEE Internal API

National Institute of Advanced Industrial Science and Technology (AIST)

Akira Tsukamoto, 2019/10/02

# Contents

# 1 CommonPage Common API

Crypto, common.

/∗∗

- Random Data Generation Function.
- The quality of the random is implementation dependent.
- I am not sure this should be in Keystone or not, but it is very handy.
- Good to have adding a way to check the quality of the random implementation.
  ∗/ TEE_GenerateRandom();

## 2 Asymmetric Key Verification Functions

Crypto, Sing and Verify with Asymmetric Key Verification Functions.
Pseudo code.
Not sure these crypto features should be prepared as APIs for Enclave or just using openssl variants.

The library used in keystone.
https://github.com/orlp/ed25519/
(zlib License)
https://github.com/mjosaarinen/tiny_sha3/
(MIT license)
In keystone.
https://github.com/keystone-enclave/keystone-sdk/tree/master/lib/verifier/ed25519
https://github.com/keystone-enclave/keystone-sdk/tree/master/lib/verifier/sha3

```
--- Asymmetric Key sign start ---
    TEE_OperationHandle ho;
    TEE_OperationHandle ao;
    uint8_t hash[HASH_LENGTH];
    char data[DSIZE];
    char sig[64];
    uint8_t sig_len = 64;

    /* Calculate hash */
    /* sha3_init() in sha3.c */
    TEE_AllocateOperation(&ho, ALG_SHA256, SHA_LENGTH);

    /* sha3_update() in sha3.c */
    TEE_DigestUpdate(ho, data, CSIZE);

    /* sha3_final() in sha3.c */
    TEE_DigestDoFinal(ho, hash, data + CSIZE, DSIZE - CSIZE);

    /* set ed25519 key */
    TEE_AllocateOperation(&ao, TEE_ALG_ED25519, TEE_MODE_SIGN, BITS));
    TEE_SetOperationKey(&ao, rsa_keypair);

    /* Keystone has ed25519_sign()
     * Equivalent in openssl is EVP_DigestSign() */
    TEE_AsymmetricSignDigest(ho, hash, HASH_LENGTH, sig, &sig_len);

    /* free up */
    TEE_FreeOperation(ho);
    TEE_FreeOperation(ao);

    /* Get the signature */

--- Asymmetric Key sign end ---


--- Asymmetric Key verify start ---
    TEE_OperationHandle ho;
    TEE_OperationHandle ao;
    uint8_t hash[HASH_LENGTH];
    char data[DSIZE];
    char sig[64];
    uint8_t sig_len = 64;

    /* set ed25519 key */
    TEE_AllocateOperation(&ao, TEE_ALG_ED25519, TEE_MODE_VERIFY, BITS));
    TEE_SetOperationKey(&ao, rsa_keypair);

    /* Keystone has ed25519_verify()
     * Equivalent in openssl is EVP_DigestVerify() */
    verify_ok = TEE_AsymmetricVerifyDigest(ao, data, HASH_LENGTH, sig, sig_len);

    /* Check verify_ok for success of verification */

--- Asymmetric Key verify end ---
```

## 3 Message Digest Functions

Pseudo code of how to use Message Digest Functions.
Keystone uses sha3.c which is almost identical.

Ultimate question is whether this should be done in Enclave (U-Mode) or Runtime (S-Mode).

The library used in keystone.
https://github.com/mjosaarinen/tiny_sha3/
(MIT license)
In keystone.
https://github.com/keystone-enclave/keystone-sdk/tree/master/lib/verifier/sha3

```
--- start digest ---
#define SHA_LENGTH 256 / 8
#define CSIZE 8
#define DSIZE 16
    TEE_OperationHandle ho;
    uint8_t hash[SHA_LENGTH];
    char data[DSIZE];
    char *pdata;

    /* sha3_init() in sha3.c */
    TEE_AllocateOperation(&ho, ALG_SHA256, SHA_LENGTH);

    /* sha3_update() in sha3.c */
    TEE_DigestUpdate(ho, data, CSIZE);

    /* sha3_final() in sha3.c */
    TEE_DigestDoFinal(ho, hash, data + CSIZE, DSIZE - CSIZE);

    /* hash value is ready */

    TEE_FreeOperation(ho);
--- end digest ---
```

# 4 About current implementation

### TEE_GetREETime

Implemented by ocall.

### TEE_GetSystemTime

Unimplemented yet. Although keystone has rdcycle instruction based simple time function, it looks not a trusted time. sgx has the trusted time function but it doesn't work on linux ATM.

### TEE_GetRelTimeStart

Unimplemented yet.

### TEE_GetRelTimeEnd

Unimplemented yet.

#### TEE_CreatePersistentObject / TEE_OpenPersistentObject

Persistent objects are implemented with REE(Linux) files. The contents are ciphered with CBC mode AES. It means that there are restrictions with read/write objects.

- It can't be opened with the append mode. If you want to append something to the object, you have to read all content and write the appended one.
- Read/write is permitted only when the data size is a multiple of 16.
- Open with RW mode isn't supported. Storage(persistent object) should be opened with write-only mode or read-only mode.

The key and initial vector (iv) cause other implementation issue. The ideal key and initial vector are hard to get in the usual keystone environment. We use attestation report as the last resort. SGX has sgx_get_key function which is essentially EGETKEY/EREPORT wrapper and use it for file encryption. Keystone/SGX report is enclave/system invariant which depends on some given data. With using objectID (file name) as the given data, it returns an enclave/system/objectID invariant. We deduce the key and the initial vector from this invariant. We use the signature part of the report as key and the iv is got as a digest of the report. It means that the iv correlates with the key. This will reduce the endurance against the brute force, though the iv changes with the enclave and objectID. Those keys add another constraints on Persistent objects.

- An object can be accessed with only one enclave.
- Changes of system could make all persistent objects obsolete.

Changes of BIOS (sgx) or SM (keystone) will give the different signature even for same enclave.

#### TEE_GetObjectInfo1

Unimplemented yet.

#### TEE_WriteObjectData

Essentially ocall which is linux write but the data is encrypted.

#### TEE_ReadObjectData

Essentially ocall which is linux read but the data is decrypted.

#### TEE_CloseObject

Essentially ocall which is linux close. The AES context is lost with it.

#### TEE_GenerateRandom

Implemented by ocall on keystone. sgx has sgx_read_rand which is almost same function with TEE_GenerateRandom.

#### TEE_AllocateOperation

Only TEE_MODE_DIGEST, TEE_MODE_ENCRYPT, TEE_MODE_DECRYPT, TEE_MODE_SIGN and TEE_M↩ ODE_VERIFY mode are support.

#### TEE_FreeOperation

Trivial implementation.

#### TEE_DigestUpdate

SHA3 update op.

#### TEE_DigestDoFinal

SHA3 update and finalize op.

#### TEE_GenerateKey

Generate 256-bit AES key or ED25519 key pair.

#### TEE_SetOperationKey

Only set key and flags.

#### TEE_AEInit

Only in experimental branch. AES GCM context initialization.

#### TEE_AEUpdateAAD

Only in experimental branch. AES GCM update AADdata.

#### TEE_AEUpdate

Only in experimental branch. AES GCM encryption/decryption.

#### TEE_AEEncryptFinal

Only in experimental branch. AES GCM encryption and finalization. Return 16-bytes tag.

#### TEE_AEDecryptFinal

Only in experimental branch. AES GCM decryption and finalization. Verify 16-bytes tag.

#### TEE_CipherInit

AES CBC context initialization.

#### TEE_CipherUpdate

AES CBC encryption/decryption.

#### TEE_CipherDoFinal

AES CBC encryption/decryption. Only for compatibility because CBC doesn't need any finalization.

#### TEE_GenerateKey

Generate AES or ed25519 keypair.

#### TEE_AllocateTransientObject

Allocate and set up transient object.

#### TEE_InitRefAttribute

Trivial implementation.

#### TEE_FreeTransientObject

Clear keys and deallocate transient object.

#### TEE_AsymmetricSignDigest

Digest with SHA3 and sign it with ed25519 key.

#### TEE_AsymmetricVerifyDigest

Essentially ed25519 verify function.

### Remark

The unimplemented functions aren't used in sample application ATM. TEE_AE∗ GP API functions support GCM and CCM only. CBC mode can be handled with TEE_Cipher∗ GP API.

# 5 How to run ref-ta tests on each platforms (temporary version)

Assume that you are in tee-ta-reference directory.

### sgx

Prerequisites: linux-sgx installed environment. See http://192.168.100.100/vc707/docs/blob/master/intel-sgx-remote-attestation-sample.md "intel sgx RA sample" to set it up. If you don't need HW mode, only `sgx SDK` is needed to build/install.

Run ref-ta application with

```
$ cd ref-ta/sgx
$ make # or make SGX_MODE=HW if you are on SGX capable machine
$ ./app
```

### keystone

Prerequisites: keystone directory *path_to_keystone* which is already built with qemu. See http://192.168.100.↩ 100/vc707/keystone-docs/blob/master/qemu-keystone-build.md "keystone build for qemu".

Run ref-ta application with

```
$ export KEYSTONE_DIR=path_to_keystone
$ export EDGER_DIR=$(pwd)/keyedge
$ cd ref-ta/keystone
$ make
$ make copyto
$ make run
```

### OPTEE

Prerequisites: optee directory *path_to_optee* which is already built with qemu. See `prerequisites` and `build instruction` of optee and follows the build steps in the latter. Perhaps you need to install some missing python libraries:

```
$ pip3 install cryptodomex
```

For qemu, you can use QEMUv8 as TARGET. So the build steps will be

"' $ mkdir -p path_to_optee $ cd path_to_optee $ repo init -u `https://github.com/OP-TEE/manifest.↩ git` -m qemu_v8.xml $ repo sync -j4 –no-clone-bundle $ cd build $ make -j2 toolchains $ make -j `nproc`

```
Test that build with
```

make run

```
This will open 2 another tabs for linux and tee console. In main console prompts you with (qemu), you can
       type 'c' to run qemu. Then in linux tab, typing "root" will make you enter the system. Type
```

# optee_example_hello_world

```
will responds with
```

Invoking TA to increment 42 TA incremented value to 43

```
Now return to tee-ta-reference directory.
```

$ export OPTEE_DIR=path_to_optee $ cd ref-ta/op-tee $ make $ make copyto $ make run "'

"make run" opens an another xterm window to receive TA log. The final result is printed on the original console.

# 6 Documentation file

Draft TEE Internal API doc

## Prerequisites

```
sudo apt update
sudo apt upgrade
```

## For doc

```
sudo apt install texlive-full doxygen graphviz
```

## For keystone

```
sudo apt install autoconf automake autotools-dev bc bison build-essential curl expat libexpat1-dev flex
       gawk gcc git gperf libgmp-dev libmpc-dev libmpfr-dev libtool texinfo tmux patchutils zlib1g-dev wget bzip2
       patch vim-common lbzip2 python pkg-config libglib2.0-dev libpixman-1-dev device-tree-compiler expect
```

## For keyedge

```
sudo apt install clang-tools-6.0 libclang-6.0-dev cmake
```

### For SGX

http://150.82.217.189/vc707/docs/blob/master/intel-sgx-remote-attestation-sample.md "Install SGX SDK and Linux driver"

Don't for get adding PATH to ~/.profile

```
export PATH=/opt/intel/sgxsdk/bin/:${PATH}
```

## For build tee-reference

```
sudo apt install makeself screen
```

## Generate PDF doc

```
$ make doc
```

## Build

```
$ make
```

# 7 Secure Storage Functions

Core Functions, Secure Storage Functions.
Pseudo code of how to use Secure Storage.
These could be implemented using ocall on Keystone.
I prefer this feature is implemented in runtime in S-Mode for less overhead rather than switching to host os every time.

Almost identical to open(), clone(), read(), write() in POSIX API.

```
--- write file start ---
     TEE_ObjectHandle o;
     char buf[bufsize];

     TEE_CreatePersistantObject(&o, filename, namelen, WO);

     /* fill the date in buffer */

     TEE_WriteObjectData(o, buf, bufsize);

     TEE_CloseObject(o);
--- write file end ---
```

```
--- read file start ---
     TEE_ObjectHandle o;
     char buf[bufsize];

     TEE_OpenPersistantObject(&o, filename, namelen, RO);

     TEE_ReadObjectData(o, buf, bufsize);

     /* use the date in buffer */

     TEE_CloseObject(o);
--- read file end ---
```

# 8 Symmetric Key Verification Functions

Crypto, Authenticated Encryption with Symmetric Key Verification Functions.
Pseudo code.
Not sure these crypto features should be prepared as APIs for Enclave or just using openssl variants.

The library used in keystone.
https://github.com/kokke/tiny-AES-c
(The Unlicense, public domain)

In keystone.
https://github.com/keystone-enclave/keystone-sdk/tree/ef484d36db1c40a0e0a4367f31c95b90d6
AES-c/app

```
--- AE encryption start ---
    TEE_OperationHandle ho;

    /* set the AES key, skipping in this pseudo code */

    /* Equivalent in openssl is EVP_EncryptInit_ex() */
    TEE_AEInit(ho, nonce, nonce_len, AES_256_GCM_BITS);

    /* Equivalent in openssl is EVP_EncryptUpdate() */
    TEE_AEUpdate(ho, plain, plain_len, cipher, &cipher_len);

    /* Equivalent in openssl is EVP_EncryptFinal() */
    TEE_AEEncryptFinal(ho, plain, plain_len, cipher, &cipher_len, tag, &tag_len);

    /* Get the auth_tag */

--- AE encryption end ---
--- AE decrypt and verify start ---
    TEE_OperationHandle ho;

    /* set the AES key, skipping in this pseudo code */

    /* Equivalent in openssl is EVP_DecryptInit_ex() */
    TEE_AEInit(ho, nonce, nonce_len, AES_256_GCM_BITS);

    /* Equivalent in openssl is EVP_DecryptUpdate() */
    TEE_AEUpdate(ho, plain, plain_len, cipher, cipher_len);

    /* Equivalent in openssl require two functions
    EVP_CIPHER_CTX_ctrl(tag) and EVP_DecryptFinal(others) */
    verify_ok = TEE_AEDecryptFinal(ho, plain, plain_len, cipher, &cipher_len, tag, tag_len);

    /* Check verify_ok for success of decrypting and authentication */

--- AE decrypt and verify end ---
```

# 9 Class Index

## 9.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

# 10 File Index

## 10.1 File List

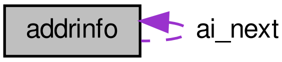
Here is a list of all files with brief descriptions:

# 11 Class Documentation

## 11.1 addrinfo Struct Reference

```
#include <tee_api_types.h>
```

Collaboration diagram for addrinfo:



### Public Attributes

- int ai_flags
- int ai_family
- int ai_socktype
- int ai_protocol
- socklen_t ai_addrlen
- struct sockaddr ∗ ai_addr
- char ∗ ai_canonname
- struct addrinfo ∗ ai_next

### 11.1.1 Member Data Documentation

#### 11.1.1.1 ai_addr

```
struct sockaddr* addrinfo::ai_addr
```

#### 11.1.1.2 ai_addrlen

```
socklen_t addrinfo::ai_addrlen
```

#### 11.1.1.3 ai_canonname

```
char* addrinfo::ai_canonname
```

#### 11.1.1.4 ai_family

```
int addrinfo::ai_family
```

#### 11.1.1.5 ai_flags

```
int addrinfo::ai_flags
```

#### 11.1.1.6 ai_next

```
struct addrinfo* addrinfo::ai_next
```

#### 11.1.1.7 ai_protocol

```
int addrinfo::ai_protocol
```

#### 11.1.1.8 ai_socktype

```
int addrinfo::ai_socktype
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.2 pollfd Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- int fd
- short int events
- short int revents

### 11.2.1 Member Data Documentation

#### 11.2.1.1 events

```
short int pollfd::events
```

#### 11.2.1.2 fd

```
int pollfd::fd
```

#### 11.2.1.3 revents

```
short int pollfd::revents
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.3 TEE_Attribute Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- uint32_t attributeID
- union {
    struct {
        void ∗ buffer
        uint32_t length
    } ref
    struct {
        uint32_t a
        uint32_t b
    } value
  } content

### 11.3.1 Member Data Documentation

#### 11.3.1.1 a

```
uint32_t TEE_Attribute::a
```

#### 11.3.1.2 attributeID

```
uint32_t TEE_Attribute::attributeID
```

#### 11.3.1.3 b

```
uint32_t TEE_Attribute::b
```

#### 11.3.1.4 buffer

```
void* TEE_Attribute::buffer
```

#### 11.3.1.5 content

```
union { ... } TEE_Attribute::content
```

#### 11.3.1.6 length

```
uint32_t TEE_Attribute::length
```

#### 11.3.1.7 ref

```
struct { ... } TEE_Attribute::ref
```

#### 11.3.1.8 value

```
struct { ... } TEE_Attribute::value
```

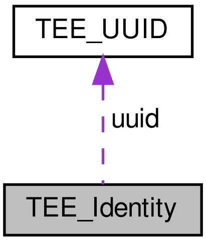
The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.4 TEE_Identity Struct Reference

```
#include <tee_api_types.h>
```

Collaboration diagram for TEE_Identity:



### Public Attributes

- uint32_t login
- TEE_UUID uuid

### 11.4.1 Member Data Documentation

#### 11.4.1.1 login

```
uint32_t TEE_Identity::login
```

#### 11.4.1.2 uuid

```
TEE_UUID TEE_Identity::uuid
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.5 TEE_ObjectInfo Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- uint32_t objectType
- union {
    uint32_t keySize
    uint32_t objectSize
  };
- union {
    uint32_t maxKeySize
    uint32_t maxObjectSize
  };
- uint32_t objectUsage
- uint32_t dataSize
- uint32_t dataPosition
- uint32_t handleFlags

### 11.5.1 Member Data Documentation

#### 11.5.1.1 "@3

```
__extension__ { ... }
```

#### 11.5.1.2 "@5

```
__extension__ { ... }
```

#### 11.5.1.3 dataPosition

```
uint32_t TEE_ObjectInfo::dataPosition
```

#### 11.5.1.4 dataSize

```
uint32_t TEE_ObjectInfo::dataSize
```

#### 11.5.1.5 handleFlags

```
uint32_t TEE_ObjectInfo::handleFlags
```

#### 11.5.1.6 keySize

```
uint32_t TEE_ObjectInfo::keySize
```

#### 11.5.1.7 maxKeySize

```
uint32_t TEE_ObjectInfo::maxKeySize
```

#### 11.5.1.8 maxObjectSize

```
uint32_t TEE_ObjectInfo::maxObjectSize
```

#### 11.5.1.9 objectSize

```
uint32_t TEE_ObjectInfo::objectSize
```

#### 11.5.1.10 objectType

```
uint32_t TEE_ObjectInfo::objectType
```

#### 11.5.1.11 objectUsage

```
uint32_t TEE_ObjectInfo::objectUsage
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.6 TEE_OperationInfo Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- uint32_t algorithm
- uint32_t operationClass
- uint32_t mode
- uint32_t digestLength
- uint32_t maxKeySize
- uint32_t keySize
- uint32_t requiredKeyUsage
- uint32_t handleState

### 11.6.1 Member Data Documentation

#### 11.6.1.1 algorithm

```
uint32_t TEE_OperationInfo::algorithm
```

#### 11.6.1.2 digestLength

```
uint32_t TEE_OperationInfo::digestLength
```

#### 11.6.1.3 handleState

```
uint32_t TEE_OperationInfo::handleState
```

#### 11.6.1.4 keySize

```
uint32_t TEE_OperationInfo::keySize
```

#### 11.6.1.5 maxKeySize

```
uint32_t TEE_OperationInfo::maxKeySize
```

#### 11.6.1.6 mode

```
uint32_t TEE_OperationInfo::mode
```

#### 11.6.1.7 operationClass

```
uint32_t TEE_OperationInfo::operationClass
```

#### 11.6.1.8 requiredKeyUsage

```
uint32_t TEE_OperationInfo::requiredKeyUsage
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.7 TEE_OperationInfoKey Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- uint32_t keySize
- uint32_t requiredKeyUsage

### 11.7.1 Member Data Documentation

#### 11.7.1.1 keySize

```
uint32_t TEE_OperationInfoKey::keySize
```

#### 11.7.1.2 requiredKeyUsage

```
uint32_t TEE_OperationInfoKey::requiredKeyUsage
```

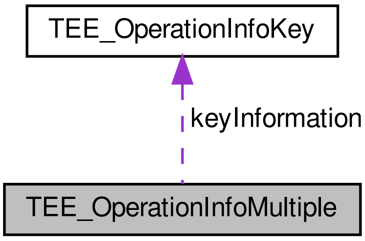
The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.8 TEE_OperationInfoMultiple Struct Reference

```
#include <tee_api_types.h>
```

Collaboration diagram for TEE_OperationInfoMultiple:



### Public Attributes

- uint32_t algorithm
- uint32_t operationClass
- uint32_t mode
- uint32_t digestLength
- uint32_t maxKeySize
- uint32_t handleState
- uint32_t operationState
- uint32_t numberOfKeys
- TEE_OperationInfoKey keyInformation [ ]

### 11.8.1 Member Data Documentation

#### 11.8.1.1 algorithm

```
uint32_t TEE_OperationInfoMultiple::algorithm
```

#### 11.8.1.2 digestLength

```
uint32_t TEE_OperationInfoMultiple::digestLength
```

#### 11.8.1.3 handleState

```
uint32_t TEE_OperationInfoMultiple::handleState
```

#### 11.8.1.4 keyInformation

```
TEE_OperationInfoKey TEE_OperationInfoMultiple::keyInformation[ ]
```

#### 11.8.1.5 maxKeySize

```
uint32_t TEE_OperationInfoMultiple::maxKeySize
```

#### 11.8.1.6 mode

```
uint32_t TEE_OperationInfoMultiple::mode
```

#### 11.8.1.7 numberOfKeys

```
uint32_t TEE_OperationInfoMultiple::numberOfKeys
```

#### 11.8.1.8 operationClass

```
uint32_t TEE_OperationInfoMultiple::operationClass
```

#### 11.8.1.9 operationState

```
uint32_t TEE_OperationInfoMultiple::operationState
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.9 TEE_Param Union Reference

```
#include <tee_api_types.h>
```

#### Public Attributes

- struct {
  void ∗ buffer
  uint32_t size
  } memref
- struct {
  uint32_t a
  uint32_t b
  } value

### 11.9.1 Member Data Documentation

#### 11.9.1.1 a

```
uint32_t TEE_Param::a
```

#### 11.9.1.2 b

```
uint32_t TEE_Param::b
```

#### 11.9.1.3 buffer

```
void* TEE_Param::buffer
```

#### 11.9.1.4 memref

```
struct { ... } TEE_Param::memref
```

#### 11.9.1.5 size

```
uint32_t TEE_Param::size
```

#### 11.9.1.6 value

```
struct { ... } TEE_Param::value
```

The documentation for this union was generated from the following file:

- include/tee_api_types.h

## 11.10 TEE_SEAID Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- uint8_t ∗ buffer
- size_t bufferLen

### 11.10.1 Member Data Documentation

#### 11.10.1.1 buffer

```
uint8_t* TEE_SEAID::buffer
```

#### 11.10.1.2 bufferLen

```
size_t TEE_SEAID::bufferLen
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.11 TEE_SEReaderProperties Struct Reference

```
#include <tee_api_types.h>
```

### Public Attributes

- bool sePresent
- bool teeOnly
- bool selectResponseEnable

### 11.11.1 Member Data Documentation

#### 11.11.1.1 selectResponseEnable

```
bool TEE_SEReaderProperties::selectResponseEnable
```

#### 11.11.1.2 sePresent

```
bool TEE_SEReaderProperties::sePresent
```

#### 11.11.1.3 teeOnly

```
bool TEE_SEReaderProperties::teeOnly
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.12 TEE_Time Struct Reference

```
#include <tee_api_types.h>
```

#### Public Attributes

- uint32_t seconds
- uint32_t millis

### 11.12.1 Member Data Documentation

#### 11.12.1.1 millis

```
uint32_t TEE_Time::millis
```

#### 11.12.1.2 seconds

```
uint32_t TEE_Time::seconds
```

The documentation for this struct was generated from the following file:

- include/tee_api_types.h

## 11.13 TEE_UUID Struct Reference

```
#include <tee_api_types.h>
```

#### Public Attributes

- uint32_t timeLow
- uint16_t timeMid
- uint16_t timeHiAndVersion
- uint8_t clockSeqAndNode [8]

### 11.13.1 Member Data Documentation

#### 11.13.1.1 clockSeqAndNode

```
uint8_t TEE_UUID::clockSeqAndNode[8]
```

#### 11.13.1.2 timeHiAndVersion

```
uint16_t TEE_UUID::timeHiAndVersion
```

#### 11.13.1.3 timeLow

```
uint32_t TEE_UUID::timeLow
```

#### 11.13.1.4 timeMid

```
uint16_t TEE_UUID::timeMid
```

The documentation for this struct was generated from the following file:

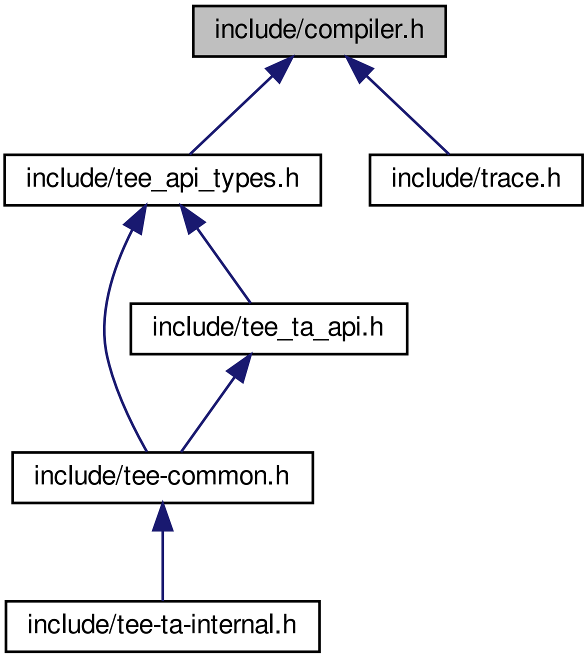
- include/tee_api_types.h

# 12 File Documentation

## 12.1 asymmetric-key-varification.md File Reference

## 12.2 include/compiler.h File Reference

This graph shows which files directly or indirectly include this file:



### Macros

- #define __deprecated __attribute__((deprecated))
- #define __packed __attribute__((packed))
- #define __weak __attribute__((weak))
- #define __noreturn __attribute__((noreturn))
- #define __pure __attribute__((pure))
- #define __aligned(x) __attribute__((aligned(x)))
- #define __printf(a, b) __attribute__((format(printf, a, b)))
- #define __noinline __attribute__((noinline))
- #define __attr_const __attribute__((__const__))
- #define __unused __attribute__((unused))
- #define __maybe_unused __attribute__((unused))
- #define __used __attribute__((__used__))
- #define __must_check __attribute__((warn_unused_result))
- #define __cold __attribute__((__cold__))
- #define __section(x) __attribute__((section(x)))
- #define __data __section(".data")
- #define __bss __section(".bss")
- #define __rodata __section(".rodata")
- #define __rodata_unpaged __section(".rodata.__unpaged")
- #define __early_ta __section(".rodata.early_ta")
- #define __noprof __attribute__((no_instrument_function))
- #define __compiler_bswap64(x) __builtin_bswap64((x))
- #define __compiler_bswap32(x) __builtin_bswap32((x))
- #define __compiler_bswap16(x) __builtin_bswap16((x))
- #define __GCC_VERSION
- #define __INTOF_HALF_MAX_SIGNED(type) ((type)1 << (sizeof(type)*8-2))
- #define __INTOF_MAX_SIGNED(type)
- #define __INTOF_MIN_SIGNED(type) (-1 - __INTOF_MAX_SIGNED(type))
- #define __INTOF_MIN(type) ((type)-1 < 1?__INTOF_MIN_SIGNED(type):(type)0)
- #define __INTOF_MAX(type) ((type)~__INTOF_MIN(type))
- #define __INTOF_ASSIGN(dest, src)
- #define __INTOF_ADD(c, a, b)
- #define __INTOF_SUB(c, a, b)
- #define __intof_mul_negate ((__intof_oa < 1) != (__intof_ob < 1))
- #define __intof_mul_hshift (sizeof(uintmax_t) * 8 / 2)
- #define __intof_mul_hmask (UINTMAX_MAX >> __intof_mul_hshift)
- #define __intof_mul_a0 ((uintmax_t)(__intof_a) >> __intof_mul_hshift)
- #define __intof_mul_b0 ((uintmax_t)(__intof_b) >> __intof_mul_hshift)
- #define __intof_mul_a1 ((uintmax_t)(__intof_a) & __intof_mul_hmask)
- #define __intof_mul_b1 ((uintmax_t)(__intof_b) & __intof_mul_hmask)
- #define __intof_mul_t
- #define __INTOF_MUL(c, a, b)
- #define __compiler_add_overflow(a, b, res) __INTOF_ADD(*(res), (a), (b))
- #define __compiler_sub_overflow(a, b, res) __INTOF_SUB(*(res), (a), (b))
- #define __compiler_mul_overflow(a, b, res) __INTOF_MUL(*(res), (a), (b))
- #define __compiler_compare_and_swap(p, oval, nval)
- #define __compiler_atomic_load(p) __atomic_load_n((p), __ATOMIC_RELAXED)
- #define __compiler_atomic_store(p, val) __atomic_store_n((p), (val), __ATOMIC_RELAXED)

### 12.2.1 Macro Definition Documentation

#### 12.2.1.1 __aligned

```
#define __aligned(
              x ) __attribute__((aligned(x)))
```

#### 12.2.1.2 __attr_const

```
#define __attr_const __attribute__((__const__))
```

#### 12.2.1.3 __bss

```
#define __bss __section(".bss")
```

#### 12.2.1.4 __cold

```
#define __cold __attribute__((__cold__))
```

#### 12.2.1.5 __compiler_add_overflow

```
#define __compiler_add_overflow(
              a,
              b,
              res ) __INTOF_ADD(*(res), (a), (b))
```

#### 12.2.1.6 __compiler_atomic_load

```
#define __compiler_atomic_load(
              p ) __atomic_load_n((p), __ATOMIC_RELAXED)
```

#### 12.2.1.7 __compiler_atomic_store

```
#define __compiler_atomic_store(
              p,
              val ) __atomic_store_n((p), (val), __ATOMIC_RELAXED)
```

#### 12.2.1.8 __compiler_bswap16

```
#define __compiler_bswap16(
              x ) __builtin_bswap16((x))
```

#### 12.2.1.9 __compiler_bswap32

```
#define __compiler_bswap32(
              x ) __builtin_bswap32((x))
```

#### 12.2.1.10 __compiler_bswap64

```
#define __compiler_bswap64(
              x ) __builtin_bswap64((x))
```

#### 12.2.1.11 __compiler_compare_and_swap

```
#define __compiler_compare_and_swap(
              p,
              oval,
              nval )
```

**Value:**

```
__atomic_compare_exchange_n((p), (oval), (nval), true, \
                    __ATOMIC_ACQUIRE, __ATOMIC_RELAXED) \
```

__HAVE_BUILTIN_OVERFLOW

#### 12.2.1.12 __compiler_mul_overflow

```
#define __compiler_mul_overflow(
              a,
              b,
              res ) __INTOF_MUL(*(res), (a), (b))
```

#### 12.2.1.13 __compiler_sub_overflow

```
#define __compiler_sub_overflow(
              a,
              b,
              res ) __INTOF_SUB(*(res), (a), (b))
```

#### 12.2.1.14 __data

```
#define __data __section(".data")
```

#### 12.2.1.15 __deprecated

```
#define __deprecated __attribute__((deprecated))
```

#### 12.2.1.16 __early_ta

```
#define __early_ta __section(".rodata.early_ta")
```

#### 12.2.1.17 __GCC_VERSION

```
#define __GCC_VERSION
```

**Value:**

```
(__GNUC__ * 10000 + __GNUC_MINOR__ * 100 + \
               __GNUC_PATCHLEVEL__)
```

#### 12.2.1.18 __INTOF_ADD

```
#define __INTOF_ADD(
              c,
              a,
              b )
```

**Value:**

```
(__extension__({ \
    typeof(a) __intofa_a = (a); \
    typeof(b) __intofa_b = (b); \
    \
    __intofa_b < 1 ? \
        ((__INTOF_MIN(typeof(c)) - __intofa_b <= __intofa_a) ? \
            __INTOF_ASSIGN((c), __intofa_a + __intofa_b) : 1) : \
        ((__INTOF_MAX(typeof(c)) - __intofa_b >= __intofa_a) ? \
            __INTOF_ASSIGN((c), __intofa_a + __intofa_b) : 1); \
}))
```

#### 12.2.1.19 __INTOF_ASSIGN

```
#define __INTOF_ASSIGN(
              dest,
              src )
```

**Value:**

```
(__extension__({ \
    typeof(src) __intof_x = (src); \
    typeof(dest) __intof_y = __intof_x; \
    (((uintmax_t)__intof_x == (uintmax_t)__intof_y) && \
     ((__intof_x < 1) == (__intof_y < 1)) ? \
        (void)((dest) = __intof_y) , 0 : 1); \
}))
```

#### 12.2.1.20 __INTOF_HALF_MAX_SIGNED

```
#define __INTOF_HALF_MAX_SIGNED(
                type ) ((type)1 << (sizeof(type)*8-2))
```

__HAVE_BUILTIN_OVERFLOW

#### 12.2.1.21 __INTOF_MAX

```
#define __INTOF_MAX(
                type ) ((type)~__INTOF_MIN(type))
```

#### 12.2.1.22 __INTOF_MAX_SIGNED

```
#define __INTOF_MAX_SIGNED(
                type )
```

**Value:**

```
(__INTOF_HALF_MAX_SIGNED(type) - 1 + \
                __INTOF_HALF_MAX_SIGNED(type))
```

#### 12.2.1.23 __INTOF_MIN

```
#define __INTOF_MIN(
                type ) ((type)-1 < 1?__INTOF_MIN_SIGNED(type):(type)0)
```

#### 12.2.1.24 __INTOF_MIN_SIGNED

```
#define __INTOF_MIN_SIGNED(
                type ) (-1 - __INTOF_MAX_SIGNED(type))
```

#### 12.2.1.25 __INTOF_MUL

```
#define __INTOF_MUL(
                c,
                a,
                b )
```

**Value:**

```
(__extension__({ \
    typeof(a) __intof_oa = (a); \
    typeof(a) __intof_a = __intof_oa < 1 ? -__intof_oa : __intof_oa; \
    typeof(b) __intof_ob = (b); \
    typeof(b) __intof_b = __intof_ob < 1 ? -__intof_ob : __intof_ob; \
    typeof(c) __intof_c; \
    \
    __intof_oa == 0 || __intof_ob == 0 || \
    __intof_oa == 1 || __intof_ob == 1 ? \
        __INTOF_ASSIGN((c), __intof_oa * __intof_ob) : \
    (__intof_mul_a0 && __intof_mul_b0) || \
     __intof_mul_t > __intof_mul_hmask ?  1 : \
    __INTOF_ADD((__intof_c), __intof_mul_t <<
      __intof_mul_hshift, \
                 __intof_mul_a1 * __intof_mul_b1) ? 1 : \
    __intof_mul_negate ? __INTOF_ASSIGN((c), -__intof_c) : \
                 __INTOF_ASSIGN((c), __intof_c); \
}))
```

#### 12.2.1.26 __intof_mul_a0

```
#define __intof_mul_a0 ((uintmax_t)(__intof_a) >> __intof_mul_hshift)
```

#### 12.2.1.27 __intof_mul_a1

```
#define __intof_mul_a1 ((uintmax_t)(__intof_a) & __intof_mul_hmask)
```

#### 12.2.1.28 __intof_mul_b0

```
#define __intof_mul_b0 ((uintmax_t)(__intof_b) >> __intof_mul_hshift)
```

#### 12.2.1.29 __intof_mul_b1

```
#define __intof_mul_b1 ((uintmax_t)(__intof_b) & __intof_mul_hmask)
```

#### 12.2.1.30 __intof_mul_hmask

```
#define __intof_mul_hmask (UINTMAX_MAX >> __intof_mul_hshift)
```

#### 12.2.1.31 __intof_mul_hshift

```
#define __intof_mul_hshift (sizeof(uintmax_t) * 8 / 2)
```

#### 12.2.1.32 __intof_mul_negate

```
#define __intof_mul_negate ((__intof_oa < 1) != (__intof_ob < 1))
```

#### 12.2.1.33 __intof_mul_t

```
#define __intof_mul_t
```

**Value:**

```
(__intof_mul_a1 * __intof_mul_b0 + \
                 __intof_mul_a0 * __intof_mul_b1)
```

#### 12.2.1.34 __INTOF_SUB

```
#define __INTOF_SUB(
              c,
              a,
              b )
```

**Value:**

```
(__extension__({ \
    typeof(a) __intofs_a = a; \
    typeof(b) __intofs_b = b; \
    \
    __intofs_b < 1 ? \
        ((__INTOF_MAX(typeof(c)) + __intofs_b >= __intofs_a) ? \
            __INTOF_ASSIGN((c), __intofs_a - __intofs_b) : 1) : \
        ((__INTOF_MIN(typeof(c)) + __intofs_b <= __intofs_a) ? \
            __INTOF_ASSIGN((c), __intofs_a - __intofs_b) : 1); \
}))
```

#### 12.2.1.35 __maybe_unused

```
#define __maybe_unused __attribute__((unused))
```

#### 12.2.1.36 __must_check

```
#define __must_check __attribute__((warn_unused_result))
```

#### 12.2.1.37 __noinline

```
#define __noinline __attribute__((noinline))
```

#### 12.2.1.38 __noprof

```
#define __noprof __attribute__((no_instrument_function))
```

#### 12.2.1.39 __noreturn

```
#define __noreturn __attribute__((noreturn))
```

#### 12.2.1.40 __packed

```
#define __packed __attribute__((packed))
```

#### 12.2.1.41 __printf

```
#define __printf(
            a,
            b ) __attribute__((format(printf, a, b)))
```

#### 12.2.1.42 __pure

```
#define __pure __attribute__((pure))
```

#### 12.2.1.43 __rodata

```
#define __rodata __section(".rodata")
```

#### 12.2.1.44 __rodata_unpaged

```
#define __rodata_unpaged __section(".rodata.__unpaged")
```

#### 12.2.1.45 __section

```
#define __section(
            x ) __attribute__((section(x)))
```

#### 12.2.1.46 __unused

```
#define __unused __attribute__((unused))
```

#### 12.2.1.47 __used

```
#define __used __attribute__((__used__))
```

#### 12.2.1.48 __weak

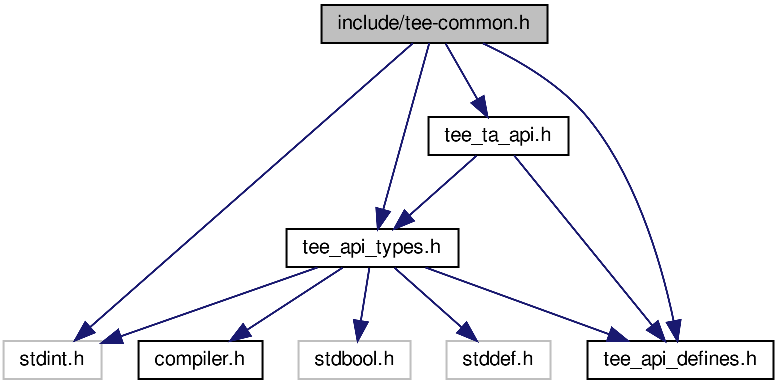
```
#define __weak __attribute__((weak))
```

## 12.3 include/tee-common.h File Reference

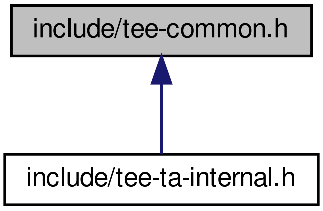
Common type and definitions of RISC-V TEE.

```
#include <stdint.h>
#include <tee_api_defines.h>
#include <tee_api_types.h>
#include <tee_ta_api.h>
```

Include dependency graph for tee-common.h:



This graph shows which files directly or indirectly include this file:



### Macros

- #define pr_deb(...) do { } while (0)

### 12.3.1 Detailed Description

Common type and definitions of RISC-V TEE.

draft RISC-V Internal TEE API

**Author**

Akira Tsukamoto, AIST

**Date**

2019/09/25

### 12.3.2 Macro Definition Documentation

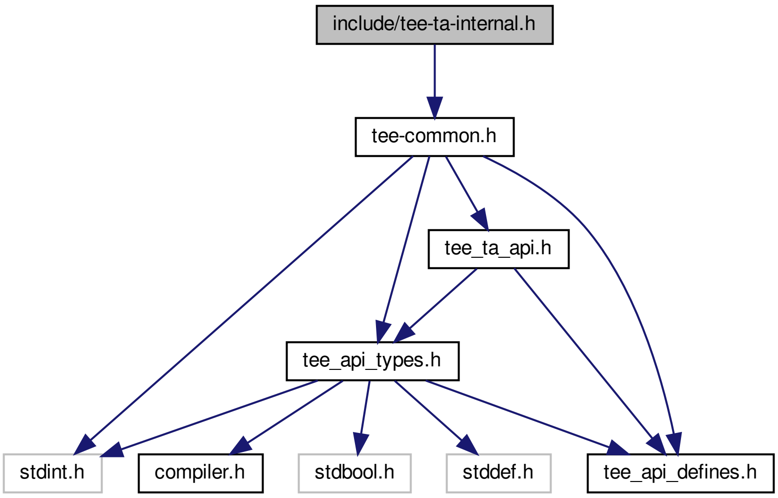
#### 12.3.2.1 pr_deb

```
#define pr_deb(
              ...  ) do { } while (0)
```

## 12.4 include/tee-ta-internal.h File Reference

Candidate API list for Global Platform like RISC-V TEE.

`#include "tee-common.h"`
Include dependency graph for tee-ta-internal.h:



### Functions

- void TEE_GetREETime (TEE_Time ∗time)

  *Core Functions, Time Functions.*
- void TEE_GetSystemTime (TEE_Time ∗time)

  *Core Functions, Time Functions.*
- TEE_Result GetRelTimeStart (uint64_t start)

  *Core Functions, Time Functions.*
- TEE_Result GetRelTimeEnd (uint64_t end)

  *Core Functions, Time Functions.*
- TEE_Result TEE_CreatePersistentObject (uint32_t storageID, const void ∗objectID, uint32_t objectIDLen, uint32_t flags, TEE_ObjectHandle attributes, const void ∗initialData, uint32_t initialDataLen, TEE_Object←Handle ∗object)

  *Core Functions, Secure Storage Functions (data is isolated for each TA)*
- TEE_Result TEE_OpenPersistentObject (uint32_t storageID, const void ∗objectID, uint32_t objectIDLen, uint32_t flags, TEE_ObjectHandle ∗object)

  *Core Functions, Secure Storage Functions (data is isolated for each TA)*
- TEE_Result TEE_GetObjectInfo1 (TEE_ObjectHandle object, TEE_ObjectInfo ∗objectInfo)

  *Core Functions, Secure Storage Functions (data is isolated for each TA)*
- TEE_Result TEE_WriteObjectData (TEE_ObjectHandle object, const void ∗buffer, uint32_t size)

  *Core Functions, Secure Storage Functions (data is isolated for each TA)*
- TEE_Result TEE_ReadObjectData (TEE_ObjectHandle object, void ∗buffer, uint32_t size, uint32_t ∗count)

  *Core Functions, Secure Storage Functions (data is isolated for each TA)*
- void TEE_CloseObject (TEE_ObjectHandle object)

  *Core Functions, Secure Storage Functions (data is isolated for each TA)*
- void TEE_GenerateRandom (void ∗randomBuffer, uint32_t randomBufferLen)

  *Crypto, common.*
- TEE_Result TEE_AllocateOperation (TEE_OperationHandle ∗operation, uint32_t algorithm, uint32_t mode, uint32_t maxKeySize)

  *Crypto, for all Crypto Functions.*
- void TEE_FreeOperation (TEE_OperationHandle operation)

  *Crypto, for all Crypto Functions.*
- void TEE_DigestUpdate (TEE_OperationHandle operation, const void ∗chunk, uint32_t chunkSize)

  *Crypto, Message Digest Functions.*
- TEE_Result TEE_DigestDoFinal (TEE_OperationHandle operation, const void ∗chunk, uint32_t chunkLen, void ∗hash, uint32_t ∗hashLen)
- TEE_Result TEE_SetOperationKey (TEE_OperationHandle operation, TEE_ObjectHandle key)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*
- TEE_Result TEE_AEInit (TEE_OperationHandle operation, const void ∗nonce, uint32_t nonceLen, uint32_t tagLen, uint32_t AADLen, uint32_t payloadLen)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*
- TEE_Result TEE_AEUpdate (TEE_OperationHandle operation, const void ∗srcData, uint32_t srcLen, void ∗destData, uint32_t ∗destLen)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*
- TEE_Result TEE_AEEncryptFinal (TEE_OperationHandle operation, const void ∗srcData, uint32_t srcLen, void ∗destData, uint32_t ∗destLen, void ∗tag, uint32_t ∗tagLen)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*
- TEE_Result TEE_AEDecryptFinal (TEE_OperationHandle operation, const void ∗srcData, uint32_t srcLen, void ∗destData, uint32_t ∗destLen, void ∗tag, uint32_t tagLen)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*
- void TEE_CipherInit (TEE_OperationHandle operation, const void ∗nonce, uint32_t nonceLen)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*

- TEE_Result TEE_CipherUpdate (TEE_OperationHandle operation, const void ∗srcData, uint32_t srcLen, void ∗destData, uint32_t ∗destLen)

  *Crypto, Authenticated Encryption with Symmetric key Verification Functions.*
- TEE_Result TEE_GenerateKey (TEE_ObjectHandle object, uint32_t keySize, TEE_Attribute ∗params, uint32_t paramCount)

  *Crypto, Asymmetric key Verification Functions.*
- TEE_Result TEE_AllocateTransientObject (TEE_ObjectType objectType, uint32_t maxKeySize, TEE_↩ ObjectHandle ∗object)

  *Crypto, Asymmetric key Verification Functions.*
- void TEE_InitRefAttribute (TEE_Attribute ∗attr, uint32_t attributeID, const void ∗buffer, uint32_t length)

  *Crypto, Asymmetric key Verification Functions.*
- void TEE_FreeTransientObject (TEE_ObjectHandle object)

  *Crypto, Asymmetric key Verification Functions.*
- TEE_Result TEE_AsymmetricSignDigest (TEE_OperationHandle operation, const TEE_Attribute ∗params, uint32_t paramCount, const void ∗digest, uint32_t digestLen, void ∗signature, uint32_t ∗signatureLen)

  *Crypto, Asymmetric key Verification Functions.*
- TEE_Result TEE_AsymmetricVerifyDigest (TEE_OperationHandle operation, const TEE_Attribute ∗params, uint32_t paramCount, const void ∗digest, uint32_t digestLen, const void ∗signature, uint32_t signatureLen)

  *Crypto, Asymmetric key Verification Functions.*

### 12.4.1 Detailed Description

Candidate API list for Global Platform like RISC-V TEE.

draft RISC-V Internal TEE API

**Author**

Akira Tsukamoto, AIST

**Date**

2019/09/25

### 12.4.2 Function Documentation

#### 12.4.2.1 GetRelTimeEnd()

```
TEE_Result GetRelTimeEnd (
            uint64_t end )
```

Core Functions, Time Functions.

Return the elapsed.

#### 12.4.2.2 GetRelTimeStart()

```
TEE_Result GetRelTimeStart (
            uint64_t start )
```

Core Functions, Time Functions.

Fast relative Time function which guarantees no hart switch or context switch between Trusted and Untrusted sides.

Most of the time ending up writing similar functions when only measuring the relative time in usec resolution which do not require the quality of the time itself but the distance of the two points.
For the usage above, the function does not have to return wall clock time.
Not prepared in both Keystone and GP.

#### 12.4.2.3 TEE_AEDecryptFinal()

```
TEE_Result TEE_AEDecryptFinal (
            TEE_OperationHandle operation,
            const void * srcData,
            uint32_t srcLen,
            void * destData,
            uint32_t * destLen,
            void * tag,
            uint32_t tagLen )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Supports TEE_ALG_AES_CCM, TEE_ALG_AES_GCM.

#### 12.4.2.4 TEE_AEEncryptFinal()

```
TEE_Result TEE_AEEncryptFinal (
            TEE_OperationHandle operation,
            const void * srcData,
            uint32_t srcLen,
            void * destData,
            uint32_t * destLen,
            void * tag,
            uint32_t * tagLen )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Supports TEE_ALG_AES_CCM, TEE_ALG_AES_GCM.

#### 12.4.2.5 TEE_AEInit()

```
TEE_Result TEE_AEInit (
            TEE_OperationHandle operation,
            const void * nonce,
            uint32_t nonceLen,
            uint32_t tagLen,
            uint32_t AADLen,
            uint32_t payloadLen )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Supports TEE_ALG_AES_CCM, TEE_ALG_AES_GCM.

#### 12.4.2.6 TEE_AEUpdate()

```
TEE_Result TEE_AEUpdate (
            TEE_OperationHandle operation,
            const void * srcData,
            uint32_t srcLen,
            void * destData,
            uint32_t * destLen )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Supports TEE_ALG_AES_CCM, TEE_ALG_AES_GCM.

#### 12.4.2.7 TEE_AllocateOperation()

```
TEE_Result TEE_AllocateOperation (
            TEE_OperationHandle * operation,
            uint32_t algorithm,
            uint32_t mode,
            uint32_t maxKeySize )
```

Crypto, for all Crypto Functions.

All Crypto Functions use TEE_OperationHandle* operation instances.
Create Crypto instance.

#### 12.4.2.8 TEE_AllocateTransientObject()

```
TEE_Result TEE_AllocateTransientObject (
            TEE_ObjectType objectType,
            uint32_t maxKeySize,
            TEE_ObjectHandle * object )
```

Crypto, Asymmetric key Verification Functions.

Create object storing asymmetric key.

#### 12.4.2.9 TEE_AsymmetricSignDigest()

```
TEE_Result TEE_AsymmetricSignDigest (
            TEE_OperationHandle operation,
            const TEE_Attribute * params,
            uint32_t paramCount,
            const void * digest,
            uint32_t digestLen,
            void * signature,
            uint32_t * signatureLen )
```

Crypto, Asymmetric key Verification Functions.

Sign a message digest within an asymmetric key operation.
Keystone has ed25519_sign().
Equivalent in openssl is EVP_DigestSign().

#### 12.4.2.10 TEE_AsymmetricVerifyDigest()

```
TEE_Result TEE_AsymmetricVerifyDigest (
             TEE_OperationHandle operation,
             const TEE_Attribute * params,
             uint32_t paramCount,
             const void * digest,
             uint32_t digestLen,
             const void * signature,
             uint32_t signatureLen )
```

Crypto, Asymmetric key Verification Functions.

Verifies a message digest signature within an asymmetric key operation.
Keystone has ed25519_verify().
Equivalent in openssl is EVP_DigestVerify().

#### 12.4.2.11 TEE_CipherInit()

```
void TEE_CipherInit (
             TEE_OperationHandle operation,
             const void * nonce,
             uint32_t nonceLen )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Supports TEE_ALG_AES_CBC.

#### 12.4.2.12 TEE_CipherUpdate()

```
TEE_Result TEE_CipherUpdate (
             TEE_OperationHandle operation,
             const void * srcData,
             uint32_t srcLen,
             void * destData,
             uint32_t * destLen )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Supports TEE_ALG_AES_CBC.

#### 12.4.2.13 TEE_CloseObject()

```
void TEE_CloseObject (
             TEE_ObjectHandle object )
```

Core Functions, Secure Storage Functions (data is isolated for each TA)

Destroy object (key, key-pair or Data).

#### 12.4.2.14 TEE_CreatePersistentObject()

```
TEE_Result TEE_CreatePersistentObject (
             uint32_t storageID,
             const void * objectID,
             uint32_t objectIDLen,
             uint32_t flags,
             TEE_ObjectHandle attributes,
             const void * initialData,
             uint32_t initialDataLen,
             TEE_ObjectHandle * object )
```

Core Functions, Secure Storage Functions (data is isolated for each TA)

Create persistent object (key, key-pair or Data).
For the people who have not written code on GP then probably do not need to care the meaning of what is Persistent Object is, since the following are enough to use secure storage feature.

#### 12.4.2.15 TEE_DigestDoFinal()

```
TEE_Result TEE_DigestDoFinal (
             TEE_OperationHandle operation,
             const void * chunk,
             uint32_t chunkLen,
             void * hash,
             uint32_t * hashLen )
```

Function accumulates message data for hashing.

#### 12.4.2.16 TEE_DigestUpdate()

```
void TEE_DigestUpdate (
             TEE_OperationHandle operation,
             const void * chunk,
             uint32_t chunkSize )
```

Crypto, Message Digest Functions.

Function accumulates message data for hashing.

#### 12.4.2.17 TEE_FreeOperation()

```
void TEE_FreeOperation (
             TEE_OperationHandle operation )
```

Crypto, for all Crypto Functions.

All Crypto Functions use TEE_OperationHandle∗ operation instances.
Destroy Crypto instance.

#### 12.4.2.18 TEE_FreeTransientObject()

```
void TEE_FreeTransientObject (
             TEE_ObjectHandle object )
```

Crypto, Asymmetric key Verification Functions.

Destroy object storing asymmetric key.

#### 12.4.2.19 TEE_GenerateKey()

```
TEE_Result TEE_GenerateKey (
             TEE_ObjectHandle object,
             uint32_t keySize,
             TEE_Attribute * params,
             uint32_t paramCount )
```

Crypto, Asymmetric key Verification Functions.

Generate asymmetric keypair.

#### 12.4.2.20 TEE_GenerateRandom()

```
void TEE_GenerateRandom (
             void * randomBuffer,
             uint32_t randomBufferLen )
```

Crypto, common.

Random Data Generation Function. The quality of the random is implementation dependent.
I am not sure this should be in Keystone or not, but it is very handy.
Good to have adding a way to check the quality of the random implementation.

#### 12.4.2.21 TEE_GetObjectInfo1()

```
TEE_Result TEE_GetObjectInfo1 (
             TEE_ObjectHandle object,
             TEE_ObjectInfo * objectInfo )
```

Core Functions, Secure Storage Functions (data is isolated for each TA)

Get length of object required before reading the object.

#### 12.4.2.22 TEE_GetREETime()

```
void TEE_GetREETime (
             TEE_Time * time )
```

Core Functions, Time Functions.

Wall clock time of host OS, expressed in the number of seconds since 1970-01-01 UTC.
This could be implemented on Keystone using ocall.

#### 12.4.2.23 TEE_GetSystemTime()

```
void TEE_GetSystemTime (
             TEE_Time * time )
```

Core Functions, Time Functions.

Time of TEE-controlled secure timer or Host OS time, implementation dependent.

#### 12.4.2.24 TEE_InitRefAttribute()

```
void TEE_InitRefAttribute (
            TEE_Attribute * attr,
            uint32_t attributeID,
            const void * buffer,
            uint32_t length )
```

Crypto, Asymmetric key Verification Functions.

Storing asymmetric key.

#### 12.4.2.25 TEE_OpenPersistentObject()

```
TEE_Result TEE_OpenPersistentObject (
            uint32_t storageID,
            const void * objectID,
            uint32_t objectIDLen,
            uint32_t flags,
            TEE_ObjectHandle * object )
```

Core Functions, Secure Storage Functions (data is isolated for each TA)

Open persistent object.

#### 12.4.2.26 TEE_ReadObjectData()

```
TEE_Result TEE_ReadObjectData (
            TEE_ObjectHandle object,
            void * buffer,
            uint32_t size,
            uint32_t * count )
```

Core Functions, Secure Storage Functions (data is isolated for each TA)

Read object.

#### 12.4.2.27 TEE_SetOperationKey()

```
TEE_Result TEE_SetOperationKey (
            TEE_OperationHandle operation,
            TEE_ObjectHandle key )
```

Crypto, Authenticated Encryption with Symmetric key Verification Functions.

Set symmetric key used in operation.

#### 12.4.2.28 TEE_WriteObjectData()

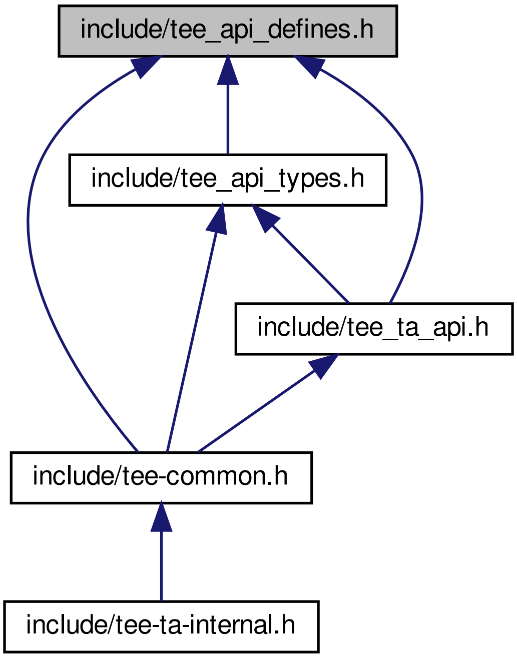
```
TEE_Result TEE_WriteObjectData (
            TEE_ObjectHandle object,
            const void * buffer,
            uint32_t size )
```

Core Functions, Secure Storage Functions (data is isolated for each TA)

Write object.

## 12.5 include/tee_api_defines.h File Reference

This graph shows which files directly or indirectly include this file:



### Macros

- #define TEE_INT_CORE_API_SPEC_VERSION 0x0000000A
- #define TEE_HANDLE_NULL 0
- #define TEE_TIMEOUT_INFINITE 0xFFFFFFFF
- #define TEE_SUCCESS 0x00000000
- #define TEE_ERROR_CORRUPT_OBJECT 0xF0100001
- #define TEE_ERROR_CORRUPT_OBJECT_2 0xF0100002
- #define TEE_ERROR_STORAGE_NOT_AVAILABLE 0xF0100003
- #define TEE_ERROR_STORAGE_NOT_AVAILABLE_2 0xF0100004
- #define TEE_ERROR_GENERIC 0xFFFF0000
- #define TEE_ERROR_ACCESS_DENIED 0xFFFF0001
- #define TEE_ERROR_CANCEL 0xFFFF0002
- #define TEE_ERROR_ACCESS_CONFLICT 0xFFFF0003
- #define TEE_ERROR_EXCESS_DATA 0xFFFF0004
- #define TEE_ERROR_BAD_FORMAT 0xFFFF0005
- #define TEE_ERROR_BAD_PARAMETERS 0xFFFF0006
- #define TEE_ERROR_BAD_STATE 0xFFFF0007
- #define TEE_ERROR_ITEM_NOT_FOUND 0xFFFF0008
- #define TEE_ERROR_NOT_IMPLEMENTED 0xFFFF0009
- #define TEE_ERROR_NOT_SUPPORTED 0xFFFF000A
- #define TEE_ERROR_NO_DATA 0xFFFF000B

- #define TEE_ERROR_OUT_OF_MEMORY 0xFFFF000C
- #define TEE_ERROR_BUSY 0xFFFF000D
- #define TEE_ERROR_COMMUNICATION 0xFFFF000E
- #define TEE_ERROR_SECURITY 0xFFFF000F
- #define TEE_ERROR_SHORT_BUFFER 0xFFFF0010
- #define TEE_ERROR_EXTERNAL_CANCEL 0xFFFF0011
- #define TEE_ERROR_OVERFLOW 0xFFFF300F
- #define TEE_ERROR_TARGET_DEAD 0xFFFF3024
- #define TEE_ERROR_STORAGE_NO_SPACE 0xFFFF3041
- #define TEE_ERROR_MAC_INVALID 0xFFFF3071
- #define TEE_ERROR_SIGNATURE_INVALID 0xFFFF3072
- #define TEE_ERROR_TIME_NOT_SET 0xFFFF5000
- #define TEE_ERROR_TIME_NEEDS_RESET 0xFFFF5001
- #define TEE_PARAM_TYPE_NONE 0
- #define TEE_PARAM_TYPE_VALUE_INPUT 1
- #define TEE_PARAM_TYPE_VALUE_OUTPUT 2
- #define TEE_PARAM_TYPE_VALUE_INOUT 3
- #define TEE_PARAM_TYPE_MEMREF_INPUT 5
- #define TEE_PARAM_TYPE_MEMREF_OUTPUT 6
- #define TEE_PARAM_TYPE_MEMREF_INOUT 7
- #define TEE_LOGIN_PUBLIC 0x00000000
- #define TEE_LOGIN_USER 0x00000001
- #define TEE_LOGIN_GROUP 0x00000002
- #define TEE_LOGIN_APPLICATION 0x00000004
- #define TEE_LOGIN_APPLICATION_USER 0x00000005
- #define TEE_LOGIN_APPLICATION_GROUP 0x00000006
- #define TEE_LOGIN_TRUSTED_APP 0xF0000000
- #define TEE_ORIGIN_API 0x00000001
- #define TEE_ORIGIN_COMMS 0x00000002
- #define TEE_ORIGIN_TEE 0x00000003
- #define TEE_ORIGIN_TRUSTED_APP 0x00000004
- #define TEE_PROPSET_TEE_IMPLEMENTATION (TEE_PropSetHandle)0xFFFFFFFD
- #define TEE_PROPSET_CURRENT_CLIENT (TEE_PropSetHandle)0xFFFFFFFE
- #define TEE_PROPSET_CURRENT_TA (TEE_PropSetHandle)0xFFFFFFFF
- #define TEE_MEMORY_ACCESS_READ 0x00000001
- #define TEE_MEMORY_ACCESS_WRITE 0x00000002
- #define TEE_MEMORY_ACCESS_ANY_OWNER 0x00000004
- #define TEE_MALLOC_FILL_ZERO 0x00000000
- #define TEE_STORAGE_PRIVATE 0x00000001
- #define TEE_DATA_FLAG_ACCESS_READ 0x00000001
- #define TEE_DATA_FLAG_ACCESS_WRITE 0x00000002
- #define TEE_DATA_FLAG_ACCESS_WRITE_META 0x00000004
- #define TEE_DATA_FLAG_SHARE_READ 0x00000010
- #define TEE_DATA_FLAG_SHARE_WRITE 0x00000020
- #define TEE_DATA_FLAG_OVERWRITE 0x00000400
- #define TEE_DATA_MAX_POSITION 0xFFFFFFFF
- #define TEE_OBJECT_ID_MAX_LEN 64
- #define TEE_USAGE_EXTRACTABLE 0x00000001
- #define TEE_USAGE_ENCRYPT 0x00000002
- #define TEE_USAGE_DECRYPT 0x00000004
- #define TEE_USAGE_MAC 0x00000008
- #define TEE_USAGE_SIGN 0x00000010
- #define TEE_USAGE_VERIFY 0x00000020
- #define TEE_USAGE_DERIVE 0x00000040
- #define TEE_HANDLE_FLAG_PERSISTENT 0x00010000

- #define TEE_HANDLE_FLAG_INITIALIZED 0x00020000
- #define TEE_HANDLE_FLAG_KEY_SET 0x00040000
- #define TEE_HANDLE_FLAG_EXPECT_TWO_KEYS 0x00080000
- #define TEE_OPERATION_CIPHER 1
- #define TEE_OPERATION_MAC 3
- #define TEE_OPERATION_AE 4
- #define TEE_OPERATION_DIGEST 5
- #define TEE_OPERATION_ASYMMETRIC_CIPHER 6
- #define TEE_OPERATION_ASYMMETRIC_SIGNATURE 7
- #define TEE_OPERATION_KEY_DERIVATION 8
- #define TEE_OPERATION_STATE_INITIAL 0x00000000
- #define TEE_OPERATION_STATE_ACTIVE 0x00000001
- #define TEE_ALG_AES_ECB_NOPAD 0x10000010
- #define TEE_ALG_AES_CBC_NOPAD 0x10000110
- #define TEE_ALG_AES_CTR 0x10000210
- #define TEE_ALG_AES_CTS 0x10000310
- #define TEE_ALG_AES_XTS 0x10000410
- #define TEE_ALG_AES_CBC_MAC_NOPAD 0x30000110
- #define TEE_ALG_AES_CBC_MAC_PKCS5 0x30000510
- #define TEE_ALG_AES_CMAC 0x30000610
- #define TEE_ALG_AES_CCM 0x40000710
- #define TEE_ALG_AES_GCM 0x40000810
- #define TEE_ALG_DES_ECB_NOPAD 0x10000011
- #define TEE_ALG_DES_CBC_NOPAD 0x10000111
- #define TEE_ALG_DES_CBC_MAC_NOPAD 0x30000111
- #define TEE_ALG_DES_CBC_MAC_PKCS5 0x30000511
- #define TEE_ALG_DES3_ECB_NOPAD 0x10000013
- #define TEE_ALG_DES3_CBC_NOPAD 0x10000113
- #define TEE_ALG_DES3_CBC_MAC_NOPAD 0x30000113
- #define TEE_ALG_DES3_CBC_MAC_PKCS5 0x30000513
- #define TEE_ALG_RSASSA_PKCS1_V1_5_MD5 0x70001830
- #define TEE_ALG_RSASSA_PKCS1_V1_5_SHA1 0x70002830
- #define TEE_ALG_RSASSA_PKCS1_V1_5_SHA224 0x70003830
- #define TEE_ALG_RSASSA_PKCS1_V1_5_SHA256 0x70004830
- #define TEE_ALG_RSASSA_PKCS1_V1_5_SHA384 0x70005830
- #define TEE_ALG_RSASSA_PKCS1_V1_5_SHA512 0x70006830
- #define TEE_ALG_RSASSA_PKCS1_V1_5_MD5SHA1 0x7000F830
- #define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA1 0x70212930
- #define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA224 0x70313930
- #define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA256 0x70414930
- #define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA384 0x70515930
- #define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA512 0x70616930
- #define TEE_ALG_RSAES_PKCS1_V1_5 0x60000130
- #define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA1 0x60210230
- #define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA224 0x60310230
- #define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA256 0x60410230
- #define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA384 0x60510230
- #define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA512 0x60610230
- #define TEE_ALG_RSA_NOPAD 0x60000030
- #define TEE_ALG_DSA_SHA1 0x70002131
- #define TEE_ALG_DSA_SHA224 0x70003131
- #define TEE_ALG_DSA_SHA256 0x70004131
- #define TEE_ALG_DH_DERIVE_SHARED_SECRET 0x80000032
- #define TEE_ALG_MD5 0x50000001
- #define TEE_ALG_SHA1 0x50000002

- #define TEE_ALG_SHA224 0x50000003
- #define TEE_ALG_SHA256 0x50000004
- #define TEE_ALG_SHA384 0x50000005
- #define TEE_ALG_SHA512 0x50000006
- #define TEE_ALG_MD5SHA1 0x5000000F
- #define TEE_ALG_HMAC_MD5 0x30000001
- #define TEE_ALG_HMAC_SHA1 0x30000002
- #define TEE_ALG_HMAC_SHA224 0x30000003
- #define TEE_ALG_HMAC_SHA256 0x30000004
- #define TEE_ALG_HMAC_SHA384 0x30000005
- #define TEE_ALG_HMAC_SHA512 0x30000006
- #define TEE_ALG_ECDSA_P192 0x70001041
- #define TEE_ALG_ECDSA_P224 0x70002041
- #define TEE_ALG_ECDSA_P256 0x70003041
- #define TEE_ALG_ECDSA_P384 0x70004041
- #define TEE_ALG_ECDSA_P521 0x70005041
- #define TEE_ALG_ECDH_P192 0x80001042
- #define TEE_ALG_ECDH_P224 0x80002042
- #define TEE_ALG_ECDH_P256 0x80003042
- #define TEE_ALG_ECDH_P384 0x80004042
- #define TEE_ALG_ECDH_P521 0x80005042
- #define TEE_TYPE_AES 0xA0000010
- #define TEE_TYPE_DES 0xA0000011
- #define TEE_TYPE_DES3 0xA0000013
- #define TEE_TYPE_HMAC_MD5 0xA0000001
- #define TEE_TYPE_HMAC_SHA1 0xA0000002
- #define TEE_TYPE_HMAC_SHA224 0xA0000003
- #define TEE_TYPE_HMAC_SHA256 0xA0000004
- #define TEE_TYPE_HMAC_SHA384 0xA0000005
- #define TEE_TYPE_HMAC_SHA512 0xA0000006
- #define TEE_TYPE_RSA_PUBLIC_KEY 0xA0000030
- #define TEE_TYPE_RSA_KEYPAIR 0xA1000030
- #define TEE_TYPE_DSA_PUBLIC_KEY 0xA0000031
- #define TEE_TYPE_DSA_KEYPAIR 0xA1000031
- #define TEE_TYPE_DH_KEYPAIR 0xA1000032
- #define TEE_TYPE_ECDSA_PUBLIC_KEY 0xA0000041
- #define TEE_TYPE_ECDSA_KEYPAIR 0xA1000041
- #define TEE_TYPE_ECDH_PUBLIC_KEY 0xA0000042
- #define TEE_TYPE_ECDH_KEYPAIR 0xA1000042
- #define TEE_TYPE_GENERIC_SECRET 0xA0000000
- #define TEE_TYPE_CORRUPTED_OBJECT 0xA00000BE
- #define TEE_TYPE_DATA 0xA00000BF
- #define TEE_ATTR_SECRET_VALUE 0xC0000000
- #define TEE_ATTR_RSA_MODULUS 0xD0000130
- #define TEE_ATTR_RSA_PUBLIC_EXPONENT 0xD0000230
- #define TEE_ATTR_RSA_PRIVATE_EXPONENT 0xC0000330
- #define TEE_ATTR_RSA_PRIME1 0xC0000430
- #define TEE_ATTR_RSA_PRIME2 0xC0000530
- #define TEE_ATTR_RSA_EXPONENT1 0xC0000630
- #define TEE_ATTR_RSA_EXPONENT2 0xC0000730
- #define TEE_ATTR_RSA_COEFFICIENT 0xC0000830
- #define TEE_ATTR_DSA_PRIME 0xD0001031
- #define TEE_ATTR_DSA_SUBPRIME 0xD0001131
- #define TEE_ATTR_DSA_BASE 0xD0001231
- #define TEE_ATTR_DSA_PUBLIC_VALUE 0xD0000131

- #define TEE_ATTR_DSA_PRIVATE_VALUE 0xC0000231
- #define TEE_ATTR_DH_PRIME 0xD0001032
- #define TEE_ATTR_DH_SUBPRIME 0xD0001132
- #define TEE_ATTR_DH_BASE 0xD0001232
- #define TEE_ATTR_DH_X_BITS 0xF0001332
- #define TEE_ATTR_DH_PUBLIC_VALUE 0xD0000132
- #define TEE_ATTR_DH_PRIVATE_VALUE 0xC0000232
- #define TEE_ATTR_RSA_OAEP_LABEL 0xD0000930
- #define TEE_ATTR_RSA_PSS_SALT_LENGTH 0xF0000A30
- #define TEE_ATTR_ECC_PUBLIC_VALUE_X 0xD0000141
- #define TEE_ATTR_ECC_PUBLIC_VALUE_Y 0xD0000241
- #define TEE_ATTR_ECC_PRIVATE_VALUE 0xC0000341
- #define TEE_ATTR_ECC_CURVE 0xF0000441
- #define TEE_ATTR_BIT_PROTECTED (1 $<<$ 28)
- #define TEE_ATTR_BIT_VALUE (1 $<<$ 29)
- #define TEE_ECC_CURVE_NIST_P192 0x00000001
- #define TEE_ECC_CURVE_NIST_P224 0x00000002
- #define TEE_ECC_CURVE_NIST_P256 0x00000003
- #define TEE_ECC_CURVE_NIST_P384 0x00000004
- #define TEE_ECC_CURVE_NIST_P521 0x00000005
- #define TEE_PANIC_ID_TA_CLOSESESSIONENTRYPOINT 0x00000101
- #define TEE_PANIC_ID_TA_CREATEENTRYPOINT 0x00000102
- #define TEE_PANIC_ID_TA_DESTROYENTRYPOINT 0x00000103
- #define TEE_PANIC_ID_TA_INVOKECOMMANDENTRYPOINT 0x00000104
- #define TEE_PANIC_ID_TA_OPENSESSIONENTRYPOINT 0x00000105
- #define TEE_PANIC_ID_TEE_ALLOCATEPROPERTYENUMERATOR 0x00000201
- #define TEE_PANIC_ID_TEE_FREEPROPERTYENUMERATOR 0x00000202
- #define TEE_PANIC_ID_TEE_GETNEXTPROPERTY 0x00000203
- #define TEE_PANIC_ID_TEE_GETPROPERTYASBINARYBLOCK 0x00000204
- #define TEE_PANIC_ID_TEE_GETPROPERTYASBOOL 0x00000205
- #define TEE_PANIC_ID_TEE_GETPROPERTYASIDENTITY 0x00000206
- #define TEE_PANIC_ID_TEE_GETPROPERTYASSTRING 0x00000207
- #define TEE_PANIC_ID_TEE_GETPROPERTYASU32 0x00000208
- #define TEE_PANIC_ID_TEE_GETPROPERTYASUUID 0x00000209
- #define TEE_PANIC_ID_TEE_GETPROPERTYNAME 0x0000020A
- #define TEE_PANIC_ID_TEE_RESETPROPERTYENUMERATOR 0x0000020B
- #define TEE_PANIC_ID_TEE_STARTPROPERTYENUMERATOR 0x0000020C
- #define TEE_PANIC_ID_TEE_PANIC 0x00000301
- #define TEE_PANIC_ID_TEE_CLOSETASESSION 0x00000401
- #define TEE_PANIC_ID_TEE_INVOKETACOMMAND 0x00000402
- #define TEE_PANIC_ID_TEE_OPENTASESSION 0x00000403
- #define TEE_PANIC_ID_TEE_GETCANCELLATIONFLAG 0x00000501
- #define TEE_PANIC_ID_TEE_MASKCANCELLATION 0x00000502
- #define TEE_PANIC_ID_TEE_UNMASKCANCELLATION 0x00000503
- #define TEE_PANIC_ID_TEE_CHECKMEMORYACCESSRIGHTS 0x00000601
- #define TEE_PANIC_ID_TEE_FREE 0x00000602
- #define TEE_PANIC_ID_TEE_GETINSTANCEDATA 0x00000603
- #define TEE_PANIC_ID_TEE_MALLOC 0x00000604
- #define TEE_PANIC_ID_TEE_MEMCOMPARE 0x00000605
- #define TEE_PANIC_ID_TEE_MEMFILL 0x00000606
- #define TEE_PANIC_ID_TEE_MEMMOVE 0x00000607
- #define TEE_PANIC_ID_TEE_REALLOC 0x00000608
- #define TEE_PANIC_ID_TEE_SETINSTANCEDATA 0x00000609
- #define TEE_PANIC_ID_TEE_CLOSEOBJECT 0x00000701
- #define TEE_PANIC_ID_TEE_GETOBJECTBUFFERATTRIBUTE 0x00000702

- #define TEE_PANIC_ID_TEE_GETOBJECTINFO 0x00000703
- #define TEE_PANIC_ID_TEE_GETOBJECTVALUEATTRIBUTE 0x00000704
- #define TEE_PANIC_ID_TEE_RESTRICTOBJECTUSAGE 0x00000705
- #define TEE_PANIC_ID_TEE_GETOBJECTINFO1 0x00000706
- #define TEE_PANIC_ID_TEE_RESTRICTOBJECTUSAGE1 0x00000707
- #define TEE_PANIC_ID_TEE_ALLOCATETRANSIENTOBJECT 0x00000801
- #define TEE_PANIC_ID_TEE_COPYOBJECTATTRIBUTES 0x00000802
- #define TEE_PANIC_ID_TEE_FREETRANSIENTOBJECT 0x00000803
- #define TEE_PANIC_ID_TEE_GENERATEKEY 0x00000804
- #define TEE_PANIC_ID_TEE_INITREFATTRIBUTE 0x00000805
- #define TEE_PANIC_ID_TEE_INITVALUEATTRIBUTE 0x00000806
- #define TEE_PANIC_ID_TEE_POPULATETRANSIENTOBJECT 0x00000807
- #define TEE_PANIC_ID_TEE_RESETTRANSIENTOBJECT 0x00000808
- #define TEE_PANIC_ID_TEE_COPYOBJECTATTRIBUTES1 0x00000809
- #define TEE_PANIC_ID_TEE_CLOSEANDDELETEPERSISTENTOBJECT 0x00000901
- #define TEE_PANIC_ID_TEE_CREATEPERSISTENTOBJECT 0x00000902
- #define TEE_PANIC_ID_TEE_OPENPERSISTENTOBJECT 0x00000903
- #define TEE_PANIC_ID_TEE_RENAMEPERSISTENTOBJECT 0x00000904
- #define TEE_PANIC_ID_TEE_CLOSEANDDELETEPERSISTENTOBJECT1 0x00000905
- #define TEE_PANIC_ID_TEE_ALLOCATEPERSISTENTOBJECTENUMERATOR 0x00000A01
- #define TEE_PANIC_ID_TEE_FREEPERSISTENTOBJECTENUMERATOR 0x00000A02
- #define TEE_PANIC_ID_TEE_GETNEXTPERSISTENTOBJECT 0x00000A03
- #define TEE_PANIC_ID_TEE_RESETPERSISTENTOBJECTENUMERATOR 0x00000A04
- #define TEE_PANIC_ID_TEE_STARTPERSISTENTOBJECTENUMERATOR 0x00000A05
- #define TEE_PANIC_ID_TEE_READOBJECTDATA 0x00000B01
- #define TEE_PANIC_ID_TEE_SEEKOBJECTDATA 0x00000B02
- #define TEE_PANIC_ID_TEE_TRUNCATEOBJECTDATA 0x00000B03
- #define TEE_PANIC_ID_TEE_WRITEOBJECTDATA 0x00000B04
- #define TEE_PANIC_ID_TEE_ALLOCATEOPERATION 0x00000C01
- #define TEE_PANIC_ID_TEE_COPYOPERATION 0x00000C02
- #define TEE_PANIC_ID_TEE_FREEOPERATION 0x00000C03
- #define TEE_PANIC_ID_TEE_GETOPERATIONINFO 0x00000C04
- #define TEE_PANIC_ID_TEE_RESETOPERATION 0x00000C05
- #define TEE_PANIC_ID_TEE_SETOPERATIONKEY 0x00000C06
- #define TEE_PANIC_ID_TEE_SETOPERATIONKEY2 0x00000C07
- #define TEE_PANIC_ID_TEE_GETOPERATIONINFOMULTIPLE 0x00000C08
- #define TEE_PANIC_ID_TEE_DIGESTDOFINAL 0x00000D01
- #define TEE_PANIC_ID_TEE_DIGESTUPDATE 0x00000D02
- #define TEE_PANIC_ID_TEE_CIPHERDOFINAL 0x00000E01
- #define TEE_PANIC_ID_TEE_CIPHERINIT 0x00000E02
- #define TEE_PANIC_ID_TEE_CIPHERUPDATE 0x00000E03
- #define TEE_PANIC_ID_TEE_MACCOMPAREFINAL 0x00000F01
- #define TEE_PANIC_ID_TEE_MACCOMPUTEFINAL 0x00000F02
- #define TEE_PANIC_ID_TEE_MACINIT 0x00000F03
- #define TEE_PANIC_ID_TEE_MACUPDATE 0x00000F04
- #define TEE_PANIC_ID_TEE_AEDECRYPTFINAL 0x00001001
- #define TEE_PANIC_ID_TEE_AEENCRYPTFINAL 0x00001002
- #define TEE_PANIC_ID_TEE_AEINIT 0x00001003
- #define TEE_PANIC_ID_TEE_AEUPDATE 0x00001004
- #define TEE_PANIC_ID_TEE_AEUPDATEAAD 0x00001005
- #define TEE_PANIC_ID_TEE_ASYMMETRICDECRYPT 0x00001101
- #define TEE_PANIC_ID_TEE_ASYMMETRICENCRYPT 0x00001102
- #define TEE_PANIC_ID_TEE_ASYMMETRICSIGNDIGEST 0x00001103
- #define TEE_PANIC_ID_TEE_ASYMMETRICVERIFYDIGEST 0x00001104
- #define TEE_PANIC_ID_TEE_DERIVEKEY 0x00001201

- #define TEE_PANIC_ID_TEE_GENERATERANDOM 0x00001301
- #define TEE_PANIC_ID_TEE_GETREETIME 0x00001401
- #define TEE_PANIC_ID_TEE_GETSYSTEMTIME 0x00001402
- #define TEE_PANIC_ID_TEE_GETTAPERSISTENTTIME 0x00001403
- #define TEE_PANIC_ID_TEE_SETTAPERSISTENTTIME 0x00001404
- #define TEE_PANIC_ID_TEE_WAIT 0x00001405
- #define TEE_PANIC_ID_TEE_BIGINTFMMCONTEXTSIZEINU32 0x00001501
- #define TEE_PANIC_ID_TEE_BIGINTFMMSIZEINU32 0x00001502
- #define TEE_PANIC_ID_TEE_BIGINTINIT 0x00001601
- #define TEE_PANIC_ID_TEE_BIGINTINITFMM 0x00001602
- #define TEE_PANIC_ID_TEE_BIGINTINITFMMCONTEXT 0x00001603
- #define TEE_PANIC_ID_TEE_BIGINTCONVERTFROMOCTETSTRING 0x00001701
- #define TEE_PANIC_ID_TEE_BIGINTCONVERTFROMS32 0x00001702
- #define TEE_PANIC_ID_TEE_BIGINTCONVERTTOOCTETSTRING 0x00001703
- #define TEE_PANIC_ID_TEE_BIGINTCONVERTTOS32 0x00001704
- #define TEE_PANIC_ID_TEE_BIGINTCMP 0x00001801
- #define TEE_PANIC_ID_TEE_BIGINTCMPS32 0x00001802
- #define TEE_PANIC_ID_TEE_BIGINTGETBIT 0x00001803
- #define TEE_PANIC_ID_TEE_BIGINTGETBITCOUNT 0x00001804
- #define TEE_PANIC_ID_TEE_BIGINTSHIFTRIGHT 0x00001805
- #define TEE_PANIC_ID_TEE_BIGINTADD 0x00001901
- #define TEE_PANIC_ID_TEE_BIGINTDIV 0x00001902
- #define TEE_PANIC_ID_TEE_BIGINTMUL 0x00001903
- #define TEE_PANIC_ID_TEE_BIGINTNEG 0x00001904
- #define TEE_PANIC_ID_TEE_BIGINTSQUARE 0x00001905
- #define TEE_PANIC_ID_TEE_BIGINTSUB 0x00001906
- #define TEE_PANIC_ID_TEE_BIGINTADDMOD 0x00001A01
- #define TEE_PANIC_ID_TEE_BIGINTINVMOD 0x00001A02
- #define TEE_PANIC_ID_TEE_BIGINTMOD 0x00001A03
- #define TEE_PANIC_ID_TEE_BIGINTMULMOD 0x00001A04
- #define TEE_PANIC_ID_TEE_BIGINTSQUAREMOD 0x00001A05
- #define TEE_PANIC_ID_TEE_BIGINTSUBMOD 0x00001A06
- #define TEE_PANIC_ID_TEE_BIGINTCOMPUTEEXTENDEDGCD 0x00001B01
- #define TEE_PANIC_ID_TEE_BIGINTISPROBABLEPRIME 0x00001B02
- #define TEE_PANIC_ID_TEE_BIGINTRELATIVEPRIME 0x00001B03
- #define TEE_PANIC_ID_TEE_BIGINTCOMPUTEFMM 0x00001C01
- #define TEE_PANIC_ID_TEE_BIGINTCONVERTFROMFMM 0x00001C02
- #define TEE_PANIC_ID_TEE_BIGINTCONVERTTOFMM 0x00001C03
- #define TEE_PARAM_TYPES(t0, t1, t2, t3) ((t0) | ((t1) << 4) | ((t2) << 8) | ((t3) << 12))
- #define TEE_PARAM_TYPE_GET(t, i) ((((uint32_t)t) >> ((i)*4)) & 0xF)
- #define TEE_PARAM_TYPE_SET(t, i) (((uint32_t)(t) & 0xF) << ((i)*4))
- #define TEE_NUM_PARAMS 4
- #define TEE_BigIntSizeInU32(n) ((((n)+31)/32)+2)

### 12.5.1 Macro Definition Documentation

#### 12.5.1.1 TEE_ALG_AES_CBC_MAC_NOPAD

```
#define TEE_ALG_AES_CBC_MAC_NOPAD 0x30000110
```

#### 12.5.1.2 TEE_ALG_AES_CBC_MAC_PKCS5

```
#define TEE_ALG_AES_CBC_MAC_PKCS5 0x30000510
```

#### 12.5.1.3 TEE_ALG_AES_CBC_NOPAD

```
#define TEE_ALG_AES_CBC_NOPAD 0x10000110
```

#### 12.5.1.4 TEE_ALG_AES_CCM

```
#define TEE_ALG_AES_CCM 0x40000710
```

#### 12.5.1.5 TEE_ALG_AES_CMAC

```
#define TEE_ALG_AES_CMAC 0x30000610
```

#### 12.5.1.6 TEE_ALG_AES_CTR

```
#define TEE_ALG_AES_CTR 0x10000210
```

#### 12.5.1.7 TEE_ALG_AES_CTS

```
#define TEE_ALG_AES_CTS 0x10000310
```

#### 12.5.1.8 TEE_ALG_AES_ECB_NOPAD

```
#define TEE_ALG_AES_ECB_NOPAD 0x10000010
```

#### 12.5.1.9 TEE_ALG_AES_GCM

```
#define TEE_ALG_AES_GCM 0x40000810
```

#### 12.5.1.10 TEE_ALG_AES_XTS

```
#define TEE_ALG_AES_XTS 0x10000410
```

#### 12.5.1.11 TEE_ALG_DES3_CBC_MAC_NOPAD

```
#define TEE_ALG_DES3_CBC_MAC_NOPAD 0x30000113
```

#### 12.5.1.12 TEE_ALG_DES3_CBC_MAC_PKCS5

```
#define TEE_ALG_DES3_CBC_MAC_PKCS5 0x30000513
```

#### 12.5.1.13 TEE_ALG_DES3_CBC_NOPAD

```
#define TEE_ALG_DES3_CBC_NOPAD 0x10000113
```

#### 12.5.1.14 TEE_ALG_DES3_ECB_NOPAD

```
#define TEE_ALG_DES3_ECB_NOPAD 0x10000013
```

#### 12.5.1.15 TEE_ALG_DES_CBC_MAC_NOPAD

```
#define TEE_ALG_DES_CBC_MAC_NOPAD 0x30000111
```

#### 12.5.1.16 TEE_ALG_DES_CBC_MAC_PKCS5

```
#define TEE_ALG_DES_CBC_MAC_PKCS5 0x30000511
```

#### 12.5.1.17 TEE_ALG_DES_CBC_NOPAD

```
#define TEE_ALG_DES_CBC_NOPAD 0x10000111
```

#### 12.5.1.18 TEE_ALG_DES_ECB_NOPAD

```
#define TEE_ALG_DES_ECB_NOPAD 0x10000011
```

#### 12.5.1.19 TEE_ALG_DH_DERIVE_SHARED_SECRET

```
#define TEE_ALG_DH_DERIVE_SHARED_SECRET 0x80000032
```

#### 12.5.1.20 TEE_ALG_DSA_SHA1

```
#define TEE_ALG_DSA_SHA1 0x70002131
```

#### 12.5.1.21 TEE_ALG_DSA_SHA224

```
#define TEE_ALG_DSA_SHA224 0x70003131
```

#### 12.5.1.22 TEE_ALG_DSA_SHA256

```
#define TEE_ALG_DSA_SHA256 0x70004131
```

#### 12.5.1.23 TEE_ALG_ECDH_P192

```
#define TEE_ALG_ECDH_P192 0x80001042
```

#### 12.5.1.24 TEE_ALG_ECDH_P224

```
#define TEE_ALG_ECDH_P224 0x80002042
```

#### 12.5.1.25 TEE_ALG_ECDH_P256

```
#define TEE_ALG_ECDH_P256 0x80003042
```

#### 12.5.1.26 TEE_ALG_ECDH_P384

```
#define TEE_ALG_ECDH_P384 0x80004042
```

#### 12.5.1.27 TEE_ALG_ECDH_P521

```
#define TEE_ALG_ECDH_P521 0x80005042
```

#### 12.5.1.28 TEE_ALG_ECDSA_P192

```
#define TEE_ALG_ECDSA_P192 0x70001041
```

#### 12.5.1.29 TEE_ALG_ECDSA_P224

```
#define TEE_ALG_ECDSA_P224 0x70002041
```

#### 12.5.1.30 TEE_ALG_ECDSA_P256

```
#define TEE_ALG_ECDSA_P256 0x70003041
```

#### 12.5.1.31 TEE_ALG_ECDSA_P384

```
#define TEE_ALG_ECDSA_P384 0x70004041
```

#### 12.5.1.32 TEE_ALG_ECDSA_P521

```
#define TEE_ALG_ECDSA_P521 0x70005041
```

#### 12.5.1.33 TEE_ALG_HMAC_MD5

```
#define TEE_ALG_HMAC_MD5 0x30000001
```

#### 12.5.1.34 TEE_ALG_HMAC_SHA1

```
#define TEE_ALG_HMAC_SHA1 0x30000002
```

#### 12.5.1.35 TEE_ALG_HMAC_SHA224

```
#define TEE_ALG_HMAC_SHA224 0x30000003
```

#### 12.5.1.36 TEE_ALG_HMAC_SHA256

```
#define TEE_ALG_HMAC_SHA256 0x30000004
```

#### 12.5.1.37 TEE_ALG_HMAC_SHA384

```
#define TEE_ALG_HMAC_SHA384 0x30000005
```

#### 12.5.1.38 TEE_ALG_HMAC_SHA512

```
#define TEE_ALG_HMAC_SHA512 0x30000006
```

#### 12.5.1.39 TEE_ALG_MD5

```
#define TEE_ALG_MD5 0x50000001
```

#### 12.5.1.40 TEE_ALG_MD5SHA1

```
#define TEE_ALG_MD5SHA1 0x5000000F
```

#### 12.5.1.41 TEE_ALG_RSA_NOPAD

```
#define TEE_ALG_RSA_NOPAD 0x60000030
```

#### 12.5.1.42 TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA1

```
#define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA1 0x60210230
```

#### 12.5.1.43 TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA224

```
#define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA224 0x60310230
```

#### 12.5.1.44 TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA256

```
#define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA256 0x60410230
```

#### 12.5.1.45 TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA384

```
#define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA384 0x60510230
```

#### 12.5.1.46 TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA512

```
#define TEE_ALG_RSAES_PKCS1_OAEP_MGF1_SHA512 0x60610230
```

#### 12.5.1.47 TEE_ALG_RSAES_PKCS1_V1_5

```
#define TEE_ALG_RSAES_PKCS1_V1_5 0x60000130
```

#### 12.5.1.48 TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA1

```
#define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA1 0x70212930
```

#### 12.5.1.49 TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA224

```
#define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA224 0x70313930
```

#### 12.5.1.50 TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA256

```
#define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA256 0x70414930
```

#### 12.5.1.51 TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA384

```
#define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA384 0x70515930
```

#### 12.5.1.52 TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA512

```
#define TEE_ALG_RSASSA_PKCS1_PSS_MGF1_SHA512 0x70616930
```

#### 12.5.1.53 TEE_ALG_RSASSA_PKCS1_V1_5_MD5

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_MD5 0x70001830
```

#### 12.5.1.54 TEE_ALG_RSASSA_PKCS1_V1_5_MD5SHA1

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_MD5SHA1 0x7000F830
```

#### 12.5.1.55 TEE_ALG_RSASSA_PKCS1_V1_5_SHA1

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_SHA1 0x70002830
```

#### 12.5.1.56 TEE_ALG_RSASSA_PKCS1_V1_5_SHA224

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_SHA224 0x70003830
```

#### 12.5.1.57 TEE_ALG_RSASSA_PKCS1_V1_5_SHA256

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_SHA256 0x70004830
```

#### 12.5.1.58 TEE_ALG_RSASSA_PKCS1_V1_5_SHA384

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_SHA384 0x70005830
```

#### 12.5.1.59 TEE_ALG_RSASSA_PKCS1_V1_5_SHA512

```
#define TEE_ALG_RSASSA_PKCS1_V1_5_SHA512 0x70006830
```

#### 12.5.1.60 TEE_ALG_SHA1

```
#define TEE_ALG_SHA1 0x50000002
```

#### 12.5.1.61 TEE_ALG_SHA224

```
#define TEE_ALG_SHA224 0x50000003
```

#### 12.5.1.62 TEE_ALG_SHA256

```
#define TEE_ALG_SHA256 0x50000004
```

#### 12.5.1.63 TEE_ALG_SHA384

```
#define TEE_ALG_SHA384 0x50000005
```

#### 12.5.1.64 TEE_ALG_SHA512

```
#define TEE_ALG_SHA512 0x50000006
```

#### 12.5.1.65 TEE_ATTR_BIT_PROTECTED

```
#define TEE_ATTR_BIT_PROTECTED (1 << 28)
```

#### 12.5.1.66 TEE_ATTR_BIT_VALUE

```
#define TEE_ATTR_BIT_VALUE (1 << 29)
```

#### 12.5.1.67 TEE_ATTR_DH_BASE

```
#define TEE_ATTR_DH_BASE 0xD0001232
```

#### 12.5.1.68 TEE_ATTR_DH_PRIME

```
#define TEE_ATTR_DH_PRIME 0xD0001032
```

#### 12.5.1.69 TEE_ATTR_DH_PRIVATE_VALUE

```
#define TEE_ATTR_DH_PRIVATE_VALUE 0xC0000232
```

#### 12.5.1.70 TEE_ATTR_DH_PUBLIC_VALUE

```
#define TEE_ATTR_DH_PUBLIC_VALUE 0xD0000132
```

#### 12.5.1.71 TEE_ATTR_DH_SUBPRIME

```
#define TEE_ATTR_DH_SUBPRIME 0xD0001132
```

#### 12.5.1.72 TEE_ATTR_DH_X_BITS

```
#define TEE_ATTR_DH_X_BITS 0xF0001332
```

#### 12.5.1.73 TEE_ATTR_DSA_BASE

```
#define TEE_ATTR_DSA_BASE 0xD0001231
```

#### 12.5.1.74 TEE_ATTR_DSA_PRIME

```
#define TEE_ATTR_DSA_PRIME 0xD0001031
```

#### 12.5.1.75 TEE_ATTR_DSA_PRIVATE_VALUE

```
#define TEE_ATTR_DSA_PRIVATE_VALUE 0xC0000231
```

#### 12.5.1.76 TEE_ATTR_DSA_PUBLIC_VALUE

```
#define TEE_ATTR_DSA_PUBLIC_VALUE 0xD0000131
```

#### 12.5.1.77 TEE_ATTR_DSA_SUBPRIME

```
#define TEE_ATTR_DSA_SUBPRIME 0xD0001131
```

#### 12.5.1.78 TEE_ATTR_ECC_CURVE

```
#define TEE_ATTR_ECC_CURVE 0xF0000441
```

#### 12.5.1.79 TEE_ATTR_ECC_PRIVATE_VALUE

```
#define TEE_ATTR_ECC_PRIVATE_VALUE 0xC0000341
```

#### 12.5.1.80 TEE_ATTR_ECC_PUBLIC_VALUE_X

```
#define TEE_ATTR_ECC_PUBLIC_VALUE_X 0xD0000141
```

#### 12.5.1.81 TEE_ATTR_ECC_PUBLIC_VALUE_Y

```
#define TEE_ATTR_ECC_PUBLIC_VALUE_Y 0xD0000241
```

#### 12.5.1.82 TEE_ATTR_RSA_COEFFICIENT

```
#define TEE_ATTR_RSA_COEFFICIENT 0xC0000830
```

#### 12.5.1.83 TEE_ATTR_RSA_EXPONENT1

```
#define TEE_ATTR_RSA_EXPONENT1 0xC0000630
```

#### 12.5.1.84 TEE_ATTR_RSA_EXPONENT2

```
#define TEE_ATTR_RSA_EXPONENT2 0xC0000730
```

#### 12.5.1.85 TEE_ATTR_RSA_MODULUS

```
#define TEE_ATTR_RSA_MODULUS 0xD0000130
```

#### 12.5.1.86 TEE_ATTR_RSA_OAEP_LABEL

```
#define TEE_ATTR_RSA_OAEP_LABEL 0xD0000930
```

#### 12.5.1.87 TEE_ATTR_RSA_PRIME1

```
#define TEE_ATTR_RSA_PRIME1 0xC0000430
```

#### 12.5.1.88 TEE_ATTR_RSA_PRIME2

```
#define TEE_ATTR_RSA_PRIME2 0xC0000530
```

#### 12.5.1.89 TEE_ATTR_RSA_PRIVATE_EXPONENT

```
#define TEE_ATTR_RSA_PRIVATE_EXPONENT 0xC0000330
```

#### 12.5.1.90 TEE_ATTR_RSA_PSS_SALT_LENGTH

```
#define TEE_ATTR_RSA_PSS_SALT_LENGTH 0xF0000A30
```

#### 12.5.1.91 TEE_ATTR_RSA_PUBLIC_EXPONENT

```
#define TEE_ATTR_RSA_PUBLIC_EXPONENT 0xD0000230
```

#### 12.5.1.92 TEE_ATTR_SECRET_VALUE

```
#define TEE_ATTR_SECRET_VALUE 0xC0000000
```

#### 12.5.1.93 TEE_BigIntSizeInU32

```
#define TEE_BigIntSizeInU32(
              n ) ((((n)+31)/32)+2)
```

#### 12.5.1.94 TEE_DATA_FLAG_ACCESS_READ

```
#define TEE_DATA_FLAG_ACCESS_READ 0x00000001
```

#### 12.5.1.95 TEE_DATA_FLAG_ACCESS_WRITE

```
#define TEE_DATA_FLAG_ACCESS_WRITE 0x00000002
```

#### 12.5.1.96 TEE_DATA_FLAG_ACCESS_WRITE_META

```
#define TEE_DATA_FLAG_ACCESS_WRITE_META 0x00000004
```

#### 12.5.1.97 TEE_DATA_FLAG_OVERWRITE

```
#define TEE_DATA_FLAG_OVERWRITE 0x00000400
```

#### 12.5.1.98 TEE_DATA_FLAG_SHARE_READ

```
#define TEE_DATA_FLAG_SHARE_READ 0x00000010
```

#### 12.5.1.99 TEE_DATA_FLAG_SHARE_WRITE

```
#define TEE_DATA_FLAG_SHARE_WRITE 0x00000020
```

#### 12.5.1.100 TEE_DATA_MAX_POSITION

```
#define TEE_DATA_MAX_POSITION 0xFFFFFFFF
```

#### 12.5.1.101 TEE_ECC_CURVE_NIST_P192

```
#define TEE_ECC_CURVE_NIST_P192 0x00000001
```

#### 12.5.1.102 TEE_ECC_CURVE_NIST_P224

```
#define TEE_ECC_CURVE_NIST_P224 0x00000002
```

#### 12.5.1.103 TEE_ECC_CURVE_NIST_P256

```
#define TEE_ECC_CURVE_NIST_P256 0x00000003
```

#### 12.5.1.104 TEE_ECC_CURVE_NIST_P384

```
#define TEE_ECC_CURVE_NIST_P384 0x00000004
```

#### 12.5.1.105 TEE_ECC_CURVE_NIST_P521

```
#define TEE_ECC_CURVE_NIST_P521 0x00000005
```

#### 12.5.1.106 TEE_ERROR_ACCESS_CONFLICT

```
#define TEE_ERROR_ACCESS_CONFLICT 0xFFFF0003
```

#### 12.5.1.107 TEE_ERROR_ACCESS_DENIED

```
#define TEE_ERROR_ACCESS_DENIED 0xFFFF0001
```

#### 12.5.1.108 TEE_ERROR_BAD_FORMAT

```
#define TEE_ERROR_BAD_FORMAT 0xFFFF0005
```

#### 12.5.1.109 TEE_ERROR_BAD_PARAMETERS

```
#define TEE_ERROR_BAD_PARAMETERS 0xFFFF0006
```

#### 12.5.1.110 TEE_ERROR_BAD_STATE

```
#define TEE_ERROR_BAD_STATE 0xFFFF0007
```

#### 12.5.1.111 TEE_ERROR_BUSY

```
#define TEE_ERROR_BUSY 0xFFFF000D
```

#### 12.5.1.112 TEE_ERROR_CANCEL

```
#define TEE_ERROR_CANCEL 0xFFFF0002
```

#### 12.5.1.113 TEE_ERROR_COMMUNICATION

```
#define TEE_ERROR_COMMUNICATION 0xFFFF000E
```

#### 12.5.1.114 TEE_ERROR_CORRUPT_OBJECT

```
#define TEE_ERROR_CORRUPT_OBJECT 0xF0100001
```

#### 12.5.1.115 TEE_ERROR_CORRUPT_OBJECT_2

```
#define TEE_ERROR_CORRUPT_OBJECT_2 0xF0100002
```

#### 12.5.1.116 TEE_ERROR_EXCESS_DATA

```
#define TEE_ERROR_EXCESS_DATA 0xFFFF0004
```

#### 12.5.1.117 TEE_ERROR_EXTERNAL_CANCEL

```
#define TEE_ERROR_EXTERNAL_CANCEL 0xFFFF0011
```

#### 12.5.1.118 TEE_ERROR_GENERIC

```
#define TEE_ERROR_GENERIC 0xFFFF0000
```

#### 12.5.1.119 TEE_ERROR_ITEM_NOT_FOUND

```
#define TEE_ERROR_ITEM_NOT_FOUND 0xFFFF0008
```

#### 12.5.1.120 TEE_ERROR_MAC_INVALID

```
#define TEE_ERROR_MAC_INVALID 0xFFFF3071
```

#### 12.5.1.121 TEE_ERROR_NO_DATA

```
#define TEE_ERROR_NO_DATA 0xFFFF000B
```

#### 12.5.1.122 TEE_ERROR_NOT_IMPLEMENTED

```
#define TEE_ERROR_NOT_IMPLEMENTED 0xFFFF0009
```

#### 12.5.1.123 TEE_ERROR_NOT_SUPPORTED

```
#define TEE_ERROR_NOT_SUPPORTED 0xFFFF000A
```

#### 12.5.1.124 TEE_ERROR_OUT_OF_MEMORY

```
#define TEE_ERROR_OUT_OF_MEMORY 0xFFFF000C
```

#### 12.5.1.125 TEE_ERROR_OVERFLOW

```
#define TEE_ERROR_OVERFLOW 0xFFFF300F
```

#### 12.5.1.126 TEE_ERROR_SECURITY

```
#define TEE_ERROR_SECURITY 0xFFFF000F
```

#### 12.5.1.127 TEE_ERROR_SHORT_BUFFER

```
#define TEE_ERROR_SHORT_BUFFER 0xFFFF0010
```

#### 12.5.1.128 TEE_ERROR_SIGNATURE_INVALID

```
#define TEE_ERROR_SIGNATURE_INVALID 0xFFFF3072
```

#### 12.5.1.129 TEE_ERROR_STORAGE_NO_SPACE

```
#define TEE_ERROR_STORAGE_NO_SPACE 0xFFFF3041
```

#### 12.5.1.130 TEE_ERROR_STORAGE_NOT_AVAILABLE

```
#define TEE_ERROR_STORAGE_NOT_AVAILABLE 0xF0100003
```

#### 12.5.1.131 TEE_ERROR_STORAGE_NOT_AVAILABLE_2

```
#define TEE_ERROR_STORAGE_NOT_AVAILABLE_2 0xF0100004
```

#### 12.5.1.132 TEE_ERROR_TARGET_DEAD

```
#define TEE_ERROR_TARGET_DEAD 0xFFFF3024
```

#### 12.5.1.133 TEE_ERROR_TIME_NEEDS_RESET

```
#define TEE_ERROR_TIME_NEEDS_RESET 0xFFFF5001
```

#### 12.5.1.134 TEE_ERROR_TIME_NOT_SET

```
#define TEE_ERROR_TIME_NOT_SET 0xFFFF5000
```

#### 12.5.1.135 TEE_HANDLE_FLAG_EXPECT_TWO_KEYS

```
#define TEE_HANDLE_FLAG_EXPECT_TWO_KEYS 0x00080000
```

#### 12.5.1.136 TEE_HANDLE_FLAG_INITIALIZED

```
#define TEE_HANDLE_FLAG_INITIALIZED 0x00020000
```

#### 12.5.1.137 TEE_HANDLE_FLAG_KEY_SET

```
#define TEE_HANDLE_FLAG_KEY_SET 0x00040000
```

#### 12.5.1.138 TEE_HANDLE_FLAG_PERSISTENT

```
#define TEE_HANDLE_FLAG_PERSISTENT 0x00010000
```

#### 12.5.1.139 TEE_HANDLE_NULL

```
#define TEE_HANDLE_NULL 0
```

#### 12.5.1.140 TEE_INT_CORE_API_SPEC_VERSION

```
#define TEE_INT_CORE_API_SPEC_VERSION 0x0000000A
```

#### 12.5.1.141 TEE_LOGIN_APPLICATION

```
#define TEE_LOGIN_APPLICATION 0x00000004
```

#### 12.5.1.142 TEE_LOGIN_APPLICATION_GROUP

```
#define TEE_LOGIN_APPLICATION_GROUP 0x00000006
```

#### 12.5.1.143 TEE_LOGIN_APPLICATION_USER

```
#define TEE_LOGIN_APPLICATION_USER 0x00000005
```

#### 12.5.1.144 TEE_LOGIN_GROUP

```
#define TEE_LOGIN_GROUP 0x00000002
```

#### 12.5.1.145 TEE_LOGIN_PUBLIC

```
#define TEE_LOGIN_PUBLIC 0x00000000
```

#### 12.5.1.146 TEE_LOGIN_TRUSTED_APP

```
#define TEE_LOGIN_TRUSTED_APP 0xF0000000
```

#### 12.5.1.147 TEE_LOGIN_USER

```
#define TEE_LOGIN_USER 0x00000001
```

#### 12.5.1.148 TEE_MALLOC_FILL_ZERO

```
#define TEE_MALLOC_FILL_ZERO 0x00000000
```

#### 12.5.1.149 TEE_MEMORY_ACCESS_ANY_OWNER

```
#define TEE_MEMORY_ACCESS_ANY_OWNER 0x00000004
```

#### 12.5.1.150 TEE_MEMORY_ACCESS_READ

```
#define TEE_MEMORY_ACCESS_READ 0x00000001
```

#### 12.5.1.151 TEE_MEMORY_ACCESS_WRITE

```
#define TEE_MEMORY_ACCESS_WRITE 0x00000002
```

#### 12.5.1.152 TEE_NUM_PARAMS

```
#define TEE_NUM_PARAMS 4
```

#### 12.5.1.153 TEE_OBJECT_ID_MAX_LEN

```
#define TEE_OBJECT_ID_MAX_LEN 64
```

#### 12.5.1.154 TEE_OPERATION_AE

```
#define TEE_OPERATION_AE 4
```

#### 12.5.1.155 TEE_OPERATION_ASYMMETRIC_CIPHER

```
#define TEE_OPERATION_ASYMMETRIC_CIPHER 6
```

#### 12.5.1.156 TEE_OPERATION_ASYMMETRIC_SIGNATURE

```
#define TEE_OPERATION_ASYMMETRIC_SIGNATURE 7
```

#### 12.5.1.157 TEE_OPERATION_CIPHER

```
#define TEE_OPERATION_CIPHER 1
```

#### 12.5.1.158 TEE_OPERATION_DIGEST

```
#define TEE_OPERATION_DIGEST 5
```

#### 12.5.1.159 TEE_OPERATION_KEY_DERIVATION

```
#define TEE_OPERATION_KEY_DERIVATION 8
```

#### 12.5.1.160 TEE_OPERATION_MAC

```
#define TEE_OPERATION_MAC 3
```

#### 12.5.1.161 TEE_OPERATION_STATE_ACTIVE

```
#define TEE_OPERATION_STATE_ACTIVE 0x00000001
```

#### 12.5.1.162 TEE_OPERATION_STATE_INITIAL

```
#define TEE_OPERATION_STATE_INITIAL 0x00000000
```

#### 12.5.1.163 TEE_ORIGIN_API

```
#define TEE_ORIGIN_API 0x00000001
```

#### 12.5.1.164 TEE_ORIGIN_COMMS

```
#define TEE_ORIGIN_COMMS 0x00000002
```

#### 12.5.1.165 TEE_ORIGIN_TEE

```
#define TEE_ORIGIN_TEE 0x00000003
```

#### 12.5.1.166 TEE_ORIGIN_TRUSTED_APP

```
#define TEE_ORIGIN_TRUSTED_APP 0x00000004
```

#### 12.5.1.167 TEE_PANIC_ID_TA_CLOSESESSIONENTRYPOINT

```
#define TEE_PANIC_ID_TA_CLOSESESSIONENTRYPOINT 0x00000101
```

#### 12.5.1.168 TEE_PANIC_ID_TA_CREATEENTRYPOINT

```
#define TEE_PANIC_ID_TA_CREATEENTRYPOINT 0x00000102
```

#### 12.5.1.169 TEE_PANIC_ID_TA_DESTROYENTRYPOINT

```
#define TEE_PANIC_ID_TA_DESTROYENTRYPOINT 0x00000103
```

#### 12.5.1.170 TEE_PANIC_ID_TA_INVOKECOMMANDENTRYPOINT

```
#define TEE_PANIC_ID_TA_INVOKECOMMANDENTRYPOINT 0x00000104
```

#### 12.5.1.171 TEE_PANIC_ID_TA_OPENSESSIONENTRYPOINT

```
#define TEE_PANIC_ID_TA_OPENSESSIONENTRYPOINT 0x00000105
```

#### 12.5.1.172 TEE_PANIC_ID_TEE_AEDECRYPTFINAL

```
#define TEE_PANIC_ID_TEE_AEDECRYPTFINAL 0x00001001
```

#### 12.5.1.173 TEE_PANIC_ID_TEE_AEENCRYPTFINAL

```
#define TEE_PANIC_ID_TEE_AEENCRYPTFINAL 0x00001002
```

#### 12.5.1.174 TEE_PANIC_ID_TEE_AEINIT

```
#define TEE_PANIC_ID_TEE_AEINIT 0x00001003
```

#### 12.5.1.175 TEE_PANIC_ID_TEE_AEUPDATE

```
#define TEE_PANIC_ID_TEE_AEUPDATE 0x00001004
```

#### 12.5.1.176 TEE_PANIC_ID_TEE_AEUPDATEAAD

```
#define TEE_PANIC_ID_TEE_AEUPDATEAAD 0x00001005
```

#### 12.5.1.177 TEE_PANIC_ID_TEE_ALLOCATEOPERATION

```
#define TEE_PANIC_ID_TEE_ALLOCATEOPERATION 0x00000C01
```

#### 12.5.1.178 TEE_PANIC_ID_TEE_ALLOCATEPERSISTENTOBJECTENUMERATOR

```
#define TEE_PANIC_ID_TEE_ALLOCATEPERSISTENTOBJECTENUMERATOR 0x00000A01
```

#### 12.5.1.179 TEE_PANIC_ID_TEE_ALLOCATEPROPERTYENUMERATOR

```
#define TEE_PANIC_ID_TEE_ALLOCATEPROPERTYENUMERATOR 0x00000201
```

#### 12.5.1.180 TEE_PANIC_ID_TEE_ALLOCATETRANSIENTOBJECT

```
#define TEE_PANIC_ID_TEE_ALLOCATETRANSIENTOBJECT 0x00000801
```

#### 12.5.1.181 TEE_PANIC_ID_TEE_ASYMMETRICDECRYPT

```
#define TEE_PANIC_ID_TEE_ASYMMETRICDECRYPT 0x00001101
```

#### 12.5.1.182 TEE_PANIC_ID_TEE_ASYMMETRICENCRYPT

```
#define TEE_PANIC_ID_TEE_ASYMMETRICENCRYPT 0x00001102
```

#### 12.5.1.183 TEE_PANIC_ID_TEE_ASYMMETRICSIGNDIGEST

```
#define TEE_PANIC_ID_TEE_ASYMMETRICSIGNDIGEST 0x00001103
```

#### 12.5.1.184 TEE_PANIC_ID_TEE_ASYMMETRICVERIFYDIGEST

```
#define TEE_PANIC_ID_TEE_ASYMMETRICVERIFYDIGEST 0x00001104
```

#### 12.5.1.185 TEE_PANIC_ID_TEE_BIGINTADD

```
#define TEE_PANIC_ID_TEE_BIGINTADD 0x00001901
```

#### 12.5.1.186 TEE_PANIC_ID_TEE_BIGINTADDMOD

```
#define TEE_PANIC_ID_TEE_BIGINTADDMOD 0x00001A01
```

#### 12.5.1.187 TEE_PANIC_ID_TEE_BIGINTCMP

```
#define TEE_PANIC_ID_TEE_BIGINTCMP 0x00001801
```

#### 12.5.1.188 TEE_PANIC_ID_TEE_BIGINTCMPS32

```
#define TEE_PANIC_ID_TEE_BIGINTCMPS32 0x00001802
```

#### 12.5.1.189 TEE_PANIC_ID_TEE_BIGINTCOMPUTEEXTENDEDGCD

```
#define TEE_PANIC_ID_TEE_BIGINTCOMPUTEEXTENDEDGCD 0x00001B01
```

#### 12.5.1.190 TEE_PANIC_ID_TEE_BIGINTCOMPUTEFMM

```
#define TEE_PANIC_ID_TEE_BIGINTCOMPUTEFMM 0x00001C01
```

#### 12.5.1.191 TEE_PANIC_ID_TEE_BIGINTCONVERTFROMFMM

```
#define TEE_PANIC_ID_TEE_BIGINTCONVERTFROMFMM 0x00001C02
```

#### 12.5.1.192 TEE_PANIC_ID_TEE_BIGINTCONVERTFROMOCTETSTRING

```
#define TEE_PANIC_ID_TEE_BIGINTCONVERTFROMOCTETSTRING 0x00001701
```

#### 12.5.1.193 TEE_PANIC_ID_TEE_BIGINTCONVERTFROMS32

```
#define TEE_PANIC_ID_TEE_BIGINTCONVERTFROMS32 0x00001702
```

#### 12.5.1.194 TEE_PANIC_ID_TEE_BIGINTCONVERTTOFMM

```
#define TEE_PANIC_ID_TEE_BIGINTCONVERTTOFMM 0x00001C03
```

#### 12.5.1.195 TEE_PANIC_ID_TEE_BIGINTCONVERTTOOCTETSTRING

```
#define TEE_PANIC_ID_TEE_BIGINTCONVERTTOOCTETSTRING 0x00001703
```

#### 12.5.1.196 TEE_PANIC_ID_TEE_BIGINTCONVERTTOS32

```
#define TEE_PANIC_ID_TEE_BIGINTCONVERTTOS32 0x00001704
```

#### 12.5.1.197 TEE_PANIC_ID_TEE_BIGINTDIV

```
#define TEE_PANIC_ID_TEE_BIGINTDIV 0x00001902
```

#### 12.5.1.198 TEE_PANIC_ID_TEE_BIGINTFMMCONTEXTSIZEINU32

```
#define TEE_PANIC_ID_TEE_BIGINTFMMCONTEXTSIZEINU32 0x00001501
```

#### 12.5.1.199 TEE_PANIC_ID_TEE_BIGINTFMMSIZEINU32

```
#define TEE_PANIC_ID_TEE_BIGINTFMMSIZEINU32 0x00001502
```

#### 12.5.1.200 TEE_PANIC_ID_TEE_BIGINTGETBIT

```
#define TEE_PANIC_ID_TEE_BIGINTGETBIT 0x00001803
```

#### 12.5.1.201 TEE_PANIC_ID_TEE_BIGINTGETBITCOUNT

```
#define TEE_PANIC_ID_TEE_BIGINTGETBITCOUNT 0x00001804
```

#### 12.5.1.202 TEE_PANIC_ID_TEE_BIGINTINIT

```
#define TEE_PANIC_ID_TEE_BIGINTINIT 0x00001601
```

#### 12.5.1.203 TEE_PANIC_ID_TEE_BIGINTINITFMM

```
#define TEE_PANIC_ID_TEE_BIGINTINITFMM 0x00001602
```

#### 12.5.1.204 TEE_PANIC_ID_TEE_BIGINTINITFMMCONTEXT

```
#define TEE_PANIC_ID_TEE_BIGINTINITFMMCONTEXT 0x00001603
```

#### 12.5.1.205 TEE_PANIC_ID_TEE_BIGINTINVMOD

```
#define TEE_PANIC_ID_TEE_BIGINTINVMOD 0x00001A02
```

#### 12.5.1.206 TEE_PANIC_ID_TEE_BIGINTISPROBABLEPRIME

```
#define TEE_PANIC_ID_TEE_BIGINTISPROBABLEPRIME 0x00001B02
```

#### 12.5.1.207 TEE_PANIC_ID_TEE_BIGINTMOD

```
#define TEE_PANIC_ID_TEE_BIGINTMOD 0x00001A03
```

#### 12.5.1.208 TEE_PANIC_ID_TEE_BIGINTMUL

```
#define TEE_PANIC_ID_TEE_BIGINTMUL 0x00001903
```

#### 12.5.1.209 TEE_PANIC_ID_TEE_BIGINTMULMOD

```
#define TEE_PANIC_ID_TEE_BIGINTMULMOD 0x00001A04
```

#### 12.5.1.210 TEE_PANIC_ID_TEE_BIGINTNEG

```
#define TEE_PANIC_ID_TEE_BIGINTNEG 0x00001904
```

#### 12.5.1.211 TEE_PANIC_ID_TEE_BIGINTRELATIVEPRIME

```
#define TEE_PANIC_ID_TEE_BIGINTRELATIVEPRIME 0x00001B03
```

#### 12.5.1.212 TEE_PANIC_ID_TEE_BIGINTSHIFTRIGHT

```
#define TEE_PANIC_ID_TEE_BIGINTSHIFTRIGHT 0x00001805
```

#### 12.5.1.213 TEE_PANIC_ID_TEE_BIGINTSQUARE

```
#define TEE_PANIC_ID_TEE_BIGINTSQUARE 0x00001905
```

#### 12.5.1.214 TEE_PANIC_ID_TEE_BIGINTSQUAREMOD

```
#define TEE_PANIC_ID_TEE_BIGINTSQUAREMOD 0x00001A05
```

#### 12.5.1.215 TEE_PANIC_ID_TEE_BIGINTSUB

```
#define TEE_PANIC_ID_TEE_BIGINTSUB 0x00001906
```

#### 12.5.1.216 TEE_PANIC_ID_TEE_BIGINTSUBMOD

```
#define TEE_PANIC_ID_TEE_BIGINTSUBMOD 0x00001A06
```

#### 12.5.1.217 TEE_PANIC_ID_TEE_CHECKMEMORYACCESSRIGHTS

```
#define TEE_PANIC_ID_TEE_CHECKMEMORYACCESSRIGHTS 0x00000601
```

#### 12.5.1.218 TEE_PANIC_ID_TEE_CIPHERDOFINAL

```
#define TEE_PANIC_ID_TEE_CIPHERDOFINAL 0x00000E01
```

#### 12.5.1.219 TEE_PANIC_ID_TEE_CIPHERINIT

```
#define TEE_PANIC_ID_TEE_CIPHERINIT 0x00000E02
```

#### 12.5.1.220 TEE_PANIC_ID_TEE_CIPHERUPDATE

```
#define TEE_PANIC_ID_TEE_CIPHERUPDATE 0x00000E03
```

#### 12.5.1.221 TEE_PANIC_ID_TEE_CLOSEANDDELETEPERSISTENTOBJECT

```
#define TEE_PANIC_ID_TEE_CLOSEANDDELETEPERSISTENTOBJECT 0x00000901
```

#### 12.5.1.222 TEE_PANIC_ID_TEE_CLOSEANDDELETEPERSISTENTOBJECT1

```
#define TEE_PANIC_ID_TEE_CLOSEANDDELETEPERSISTENTOBJECT1 0x00000905
```

#### 12.5.1.223 TEE_PANIC_ID_TEE_CLOSEOBJECT

```
#define TEE_PANIC_ID_TEE_CLOSEOBJECT 0x00000701
```

#### 12.5.1.224 TEE_PANIC_ID_TEE_CLOSETASESSION

```
#define TEE_PANIC_ID_TEE_CLOSETASESSION 0x00000401
```

#### 12.5.1.225 TEE_PANIC_ID_TEE_COPYOBJECTATTRIBUTES

```
#define TEE_PANIC_ID_TEE_COPYOBJECTATTRIBUTES 0x00000802
```

#### 12.5.1.226 TEE_PANIC_ID_TEE_COPYOBJECTATTRIBUTES1

```
#define TEE_PANIC_ID_TEE_COPYOBJECTATTRIBUTES1 0x00000809
```

#### 12.5.1.227 TEE_PANIC_ID_TEE_COPYOPERATION

```
#define TEE_PANIC_ID_TEE_COPYOPERATION 0x00000C02
```

#### 12.5.1.228 TEE_PANIC_ID_TEE_CREATEPERSISTENTOBJECT

```
#define TEE_PANIC_ID_TEE_CREATEPERSISTENTOBJECT 0x00000902
```

#### 12.5.1.229 TEE_PANIC_ID_TEE_DERIVEKEY

```
#define TEE_PANIC_ID_TEE_DERIVEKEY 0x00001201
```

#### 12.5.1.230 TEE_PANIC_ID_TEE_DIGESTDOFINAL

```
#define TEE_PANIC_ID_TEE_DIGESTDOFINAL 0x00000D01
```

#### 12.5.1.231 TEE_PANIC_ID_TEE_DIGESTUPDATE

```
#define TEE_PANIC_ID_TEE_DIGESTUPDATE 0x00000D02
```

#### 12.5.1.232 TEE_PANIC_ID_TEE_FREE

```
#define TEE_PANIC_ID_TEE_FREE 0x00000602
```

#### 12.5.1.233 TEE_PANIC_ID_TEE_FREEOPERATION

```
#define TEE_PANIC_ID_TEE_FREEOPERATION 0x00000C03
```

#### 12.5.1.234 TEE_PANIC_ID_TEE_FREEPERSISTENTOBJECTENUMERATOR

```
#define TEE_PANIC_ID_TEE_FREEPERSISTENTOBJECTENUMERATOR 0x00000A02
```

#### 12.5.1.235 TEE_PANIC_ID_TEE_FREEPROPERTYENUMERATOR

```
#define TEE_PANIC_ID_TEE_FREEPROPERTYENUMERATOR 0x00000202
```

#### 12.5.1.236 TEE_PANIC_ID_TEE_FREETRANSIENTOBJECT

```
#define TEE_PANIC_ID_TEE_FREETRANSIENTOBJECT 0x00000803
```

#### 12.5.1.237 TEE_PANIC_ID_TEE_GENERATEKEY

```
#define TEE_PANIC_ID_TEE_GENERATEKEY 0x00000804
```

#### 12.5.1.238 TEE_PANIC_ID_TEE_GENERATERANDOM

```
#define TEE_PANIC_ID_TEE_GENERATERANDOM 0x00001301
```

#### 12.5.1.239 TEE_PANIC_ID_TEE_GETCANCELLATIONFLAG

```
#define TEE_PANIC_ID_TEE_GETCANCELLATIONFLAG 0x00000501
```

#### 12.5.1.240 TEE_PANIC_ID_TEE_GETINSTANCEDATA

```
#define TEE_PANIC_ID_TEE_GETINSTANCEDATA 0x00000603
```

#### 12.5.1.241 TEE_PANIC_ID_TEE_GETNEXTPERSISTENTOBJECT

```
#define TEE_PANIC_ID_TEE_GETNEXTPERSISTENTOBJECT 0x00000A03
```

#### 12.5.1.242 TEE_PANIC_ID_TEE_GETNEXTPROPERTY

```
#define TEE_PANIC_ID_TEE_GETNEXTPROPERTY 0x00000203
```

#### 12.5.1.243 TEE_PANIC_ID_TEE_GETOBJECTBUFFERATTRIBUTE

```
#define TEE_PANIC_ID_TEE_GETOBJECTBUFFERATTRIBUTE 0x00000702
```

#### 12.5.1.244 TEE_PANIC_ID_TEE_GETOBJECTINFO

```
#define TEE_PANIC_ID_TEE_GETOBJECTINFO 0x00000703
```

#### 12.5.1.245 TEE_PANIC_ID_TEE_GETOBJECTINFO1

```
#define TEE_PANIC_ID_TEE_GETOBJECTINFO1 0x00000706
```

#### 12.5.1.246 TEE_PANIC_ID_TEE_GETOBJECTVALUEATTRIBUTE

```
#define TEE_PANIC_ID_TEE_GETOBJECTVALUEATTRIBUTE 0x00000704
```

#### 12.5.1.247 TEE_PANIC_ID_TEE_GETOPERATIONINFO

```
#define TEE_PANIC_ID_TEE_GETOPERATIONINFO 0x00000C04
```

#### 12.5.1.248 TEE_PANIC_ID_TEE_GETOPERATIONINFOMULTIPLE

```
#define TEE_PANIC_ID_TEE_GETOPERATIONINFOMULTIPLE 0x00000C08
```

#### 12.5.1.249 TEE_PANIC_ID_TEE_GETPROPERTYASBINARYBLOCK

```
#define TEE_PANIC_ID_TEE_GETPROPERTYASBINARYBLOCK 0x00000204
```

#### 12.5.1.250 TEE_PANIC_ID_TEE_GETPROPERTYASBOOL

```
#define TEE_PANIC_ID_TEE_GETPROPERTYASBOOL 0x00000205
```

#### 12.5.1.251 TEE_PANIC_ID_TEE_GETPROPERTYASIDENTITY

```
#define TEE_PANIC_ID_TEE_GETPROPERTYASIDENTITY 0x00000206
```

#### 12.5.1.252 TEE_PANIC_ID_TEE_GETPROPERTYASSTRING

```
#define TEE_PANIC_ID_TEE_GETPROPERTYASSTRING 0x00000207
```

#### 12.5.1.253 TEE_PANIC_ID_TEE_GETPROPERTYASU32

```
#define TEE_PANIC_ID_TEE_GETPROPERTYASU32 0x00000208
```

#### 12.5.1.254 TEE_PANIC_ID_TEE_GETPROPERTYASUUID

```
#define TEE_PANIC_ID_TEE_GETPROPERTYASUUID 0x00000209
```

#### 12.5.1.255 TEE_PANIC_ID_TEE_GETPROPERTYNAME

```
#define TEE_PANIC_ID_TEE_GETPROPERTYNAME 0x0000020A
```

#### 12.5.1.256 TEE_PANIC_ID_TEE_GETREETIME

```
#define TEE_PANIC_ID_TEE_GETREETIME 0x00001401
```

#### 12.5.1.257 TEE_PANIC_ID_TEE_GETSYSTEMTIME

```
#define TEE_PANIC_ID_TEE_GETSYSTEMTIME 0x00001402
```

#### 12.5.1.258 TEE_PANIC_ID_TEE_GETTAPERSISTENTTIME

```
#define TEE_PANIC_ID_TEE_GETTAPERSISTENTTIME 0x00001403
```

#### 12.5.1.259 TEE_PANIC_ID_TEE_INITREFATTRIBUTE

```
#define TEE_PANIC_ID_TEE_INITREFATTRIBUTE 0x00000805
```

#### 12.5.1.260 TEE_PANIC_ID_TEE_INITVALUEATTRIBUTE

```
#define TEE_PANIC_ID_TEE_INITVALUEATTRIBUTE 0x00000806
```

#### 12.5.1.261 TEE_PANIC_ID_TEE_INVOKETACOMMAND

```
#define TEE_PANIC_ID_TEE_INVOKETACOMMAND 0x00000402
```

#### 12.5.1.262 TEE_PANIC_ID_TEE_MACCOMPAREFINAL

```
#define TEE_PANIC_ID_TEE_MACCOMPAREFINAL 0x00000F01
```

#### 12.5.1.263 TEE_PANIC_ID_TEE_MACCOMPUTEFINAL

```
#define TEE_PANIC_ID_TEE_MACCOMPUTEFINAL 0x00000F02
```

#### 12.5.1.264 TEE_PANIC_ID_TEE_MACINIT

```
#define TEE_PANIC_ID_TEE_MACINIT 0x00000F03
```

#### 12.5.1.265 TEE_PANIC_ID_TEE_MACUPDATE

```
#define TEE_PANIC_ID_TEE_MACUPDATE 0x00000F04
```

#### 12.5.1.266 TEE_PANIC_ID_TEE_MALLOC

```
#define TEE_PANIC_ID_TEE_MALLOC 0x00000604
```

#### 12.5.1.267 TEE_PANIC_ID_TEE_MASKCANCELLATION

```
#define TEE_PANIC_ID_TEE_MASKCANCELLATION 0x00000502
```

#### 12.5.1.268 TEE_PANIC_ID_TEE_MEMCOMPARE

```
#define TEE_PANIC_ID_TEE_MEMCOMPARE 0x00000605
```

#### 12.5.1.269 TEE_PANIC_ID_TEE_MEMFILL

```
#define TEE_PANIC_ID_TEE_MEMFILL 0x00000606
```

#### 12.5.1.270 TEE_PANIC_ID_TEE_MEMMOVE

```
#define TEE_PANIC_ID_TEE_MEMMOVE 0x00000607
```

#### 12.5.1.271 TEE_PANIC_ID_TEE_OPENPERSISTENTOBJECT

```
#define TEE_PANIC_ID_TEE_OPENPERSISTENTOBJECT 0x00000903
```

#### 12.5.1.272 TEE_PANIC_ID_TEE_OPENTASESSION

```
#define TEE_PANIC_ID_TEE_OPENTASESSION 0x00000403
```

#### 12.5.1.273 TEE_PANIC_ID_TEE_PANIC

```
#define TEE_PANIC_ID_TEE_PANIC 0x00000301
```

#### 12.5.1.274 TEE_PANIC_ID_TEE_POPULATETRANSIENTOBJECT

```
#define TEE_PANIC_ID_TEE_POPULATETRANSIENTOBJECT 0x00000807
```

#### 12.5.1.275 TEE_PANIC_ID_TEE_READOBJECTDATA

```
#define TEE_PANIC_ID_TEE_READOBJECTDATA 0x00000B01
```

#### 12.5.1.276 TEE_PANIC_ID_TEE_REALLOC

```
#define TEE_PANIC_ID_TEE_REALLOC 0x00000608
```

#### 12.5.1.277 TEE_PANIC_ID_TEE_RENAMEPERSISTENTOBJECT

```
#define TEE_PANIC_ID_TEE_RENAMEPERSISTENTOBJECT 0x00000904
```

#### 12.5.1.278 TEE_PANIC_ID_TEE_RESETOPERATION

```
#define TEE_PANIC_ID_TEE_RESETOPERATION 0x00000C05
```

#### 12.5.1.279 TEE_PANIC_ID_TEE_RESETPERSISTENTOBJECTENUMERATOR

```
#define TEE_PANIC_ID_TEE_RESETPERSISTENTOBJECTENUMERATOR 0x00000A04
```

#### 12.5.1.280 TEE_PANIC_ID_TEE_RESETPROPERTYENUMERATOR

```
#define TEE_PANIC_ID_TEE_RESETPROPERTYENUMERATOR 0x0000020B
```

#### 12.5.1.281 TEE_PANIC_ID_TEE_RESETTRANSIENTOBJECT

```
#define TEE_PANIC_ID_TEE_RESETTRANSIENTOBJECT 0x00000808
```

#### 12.5.1.282 TEE_PANIC_ID_TEE_RESTRICTOBJECTUSAGE

```
#define TEE_PANIC_ID_TEE_RESTRICTOBJECTUSAGE 0x00000705
```

#### 12.5.1.283 TEE_PANIC_ID_TEE_RESTRICTOBJECTUSAGE1

```
#define TEE_PANIC_ID_TEE_RESTRICTOBJECTUSAGE1 0x00000707
```

#### 12.5.1.284 TEE_PANIC_ID_TEE_SEEKOBJECTDATA

```
#define TEE_PANIC_ID_TEE_SEEKOBJECTDATA 0x00000B02
```

#### 12.5.1.285 TEE_PANIC_ID_TEE_SETINSTANCEDATA

```
#define TEE_PANIC_ID_TEE_SETINSTANCEDATA 0x00000609
```

#### 12.5.1.286 TEE_PANIC_ID_TEE_SETOPERATIONKEY

```
#define TEE_PANIC_ID_TEE_SETOPERATIONKEY 0x00000C06
```

#### 12.5.1.287 TEE_PANIC_ID_TEE_SETOPERATIONKEY2

```
#define TEE_PANIC_ID_TEE_SETOPERATIONKEY2 0x00000C07
```

#### 12.5.1.288 TEE_PANIC_ID_TEE_SETTAPERSISTENTTIME

```
#define TEE_PANIC_ID_TEE_SETTAPERSISTENTTIME 0x00001404
```

#### 12.5.1.289 TEE_PANIC_ID_TEE_STARTPERSISTENTOBJECTENUMERATOR

```
#define TEE_PANIC_ID_TEE_STARTPERSISTENTOBJECTENUMERATOR 0x00000A05
```

#### 12.5.1.290 TEE_PANIC_ID_TEE_STARTPROPERTYENUMERATOR

```
#define TEE_PANIC_ID_TEE_STARTPROPERTYENUMERATOR 0x0000020C
```

#### 12.5.1.291 TEE_PANIC_ID_TEE_TRUNCATEOBJECTDATA

```
#define TEE_PANIC_ID_TEE_TRUNCATEOBJECTDATA 0x00000B03
```

#### 12.5.1.292 TEE_PANIC_ID_TEE_UNMASKCANCELLATION

```
#define TEE_PANIC_ID_TEE_UNMASKCANCELLATION 0x00000503
```

#### 12.5.1.293 TEE_PANIC_ID_TEE_WAIT

```
#define TEE_PANIC_ID_TEE_WAIT 0x00001405
```

#### 12.5.1.294 TEE_PANIC_ID_TEE_WRITEOBJECTDATA

```
#define TEE_PANIC_ID_TEE_WRITEOBJECTDATA 0x00000B04
```

#### 12.5.1.295 TEE_PARAM_TYPE_GET

```
#define TEE_PARAM_TYPE_GET(
             t,
             i ) ((((uint32_t)t) >> ((i)*4)) & 0xF)
```

#### 12.5.1.296 TEE_PARAM_TYPE_MEMREF_INOUT

```
#define TEE_PARAM_TYPE_MEMREF_INOUT 7
```

#### 12.5.1.297 TEE_PARAM_TYPE_MEMREF_INPUT

```
#define TEE_PARAM_TYPE_MEMREF_INPUT 5
```

#### 12.5.1.298 TEE_PARAM_TYPE_MEMREF_OUTPUT

```
#define TEE_PARAM_TYPE_MEMREF_OUTPUT 6
```

#### 12.5.1.299 TEE_PARAM_TYPE_NONE

```
#define TEE_PARAM_TYPE_NONE 0
```

#### 12.5.1.300 TEE_PARAM_TYPE_SET

```
#define TEE_PARAM_TYPE_SET(
              t,
              i ) (((uint32_t)(t) & 0xF) << ((i)*4))
```

#### 12.5.1.301 TEE_PARAM_TYPE_VALUE_INOUT

```
#define TEE_PARAM_TYPE_VALUE_INOUT 3
```

#### 12.5.1.302 TEE_PARAM_TYPE_VALUE_INPUT

```
#define TEE_PARAM_TYPE_VALUE_INPUT 1
```

#### 12.5.1.303 TEE_PARAM_TYPE_VALUE_OUTPUT

```
#define TEE_PARAM_TYPE_VALUE_OUTPUT 2
```

#### 12.5.1.304 TEE_PARAM_TYPES

```
#define TEE_PARAM_TYPES(
              t0,
              t1,
              t2,
              t3 ) ((t0) | ((t1) << 4) | ((t2) << 8) | ((t3) << 12))
```

#### 12.5.1.305 TEE_PROPSET_CURRENT_CLIENT

```
#define TEE_PROPSET_CURRENT_CLIENT (TEE_PropSetHandle)0xFFFFFFFE
```

#### 12.5.1.306 TEE_PROPSET_CURRENT_TA

```
#define TEE_PROPSET_CURRENT_TA (TEE_PropSetHandle)0xFFFFFFFF
```

#### 12.5.1.307 TEE_PROPSET_TEE_IMPLEMENTATION

```
#define TEE_PROPSET_TEE_IMPLEMENTATION (TEE_PropSetHandle)0xFFFFFFFD
```

#### 12.5.1.308 TEE_STORAGE_PRIVATE

```
#define TEE_STORAGE_PRIVATE 0x00000001
```

#### 12.5.1.309 TEE_SUCCESS

```
#define TEE_SUCCESS 0x00000000
```

#### 12.5.1.310 TEE_TIMEOUT_INFINITE

```
#define TEE_TIMEOUT_INFINITE 0xFFFFFFFF
```

#### 12.5.1.311 TEE_TYPE_AES

```
#define TEE_TYPE_AES 0xA0000010
```

#### 12.5.1.312 TEE_TYPE_CORRUPTED_OBJECT

```
#define TEE_TYPE_CORRUPTED_OBJECT 0xA00000BE
```

#### 12.5.1.313 TEE_TYPE_DATA

```
#define TEE_TYPE_DATA 0xA00000BF
```

#### 12.5.1.314 TEE_TYPE_DES

```
#define TEE_TYPE_DES 0xA0000011
```

#### 12.5.1.315 TEE_TYPE_DES3

```
#define TEE_TYPE_DES3 0xA0000013
```

#### 12.5.1.316 TEE_TYPE_DH_KEYPAIR

```
#define TEE_TYPE_DH_KEYPAIR 0xA1000032
```

#### 12.5.1.317 TEE_TYPE_DSA_KEYPAIR

```
#define TEE_TYPE_DSA_KEYPAIR 0xA1000031
```

#### 12.5.1.318 TEE_TYPE_DSA_PUBLIC_KEY

```
#define TEE_TYPE_DSA_PUBLIC_KEY 0xA0000031
```

#### 12.5.1.319 TEE_TYPE_ECDH_KEYPAIR

```
#define TEE_TYPE_ECDH_KEYPAIR 0xA1000042
```

#### 12.5.1.320 TEE_TYPE_ECDH_PUBLIC_KEY

```
#define TEE_TYPE_ECDH_PUBLIC_KEY 0xA0000042
```

#### 12.5.1.321 TEE_TYPE_ECDSA_KEYPAIR

```
#define TEE_TYPE_ECDSA_KEYPAIR 0xA1000041
```

#### 12.5.1.322 TEE_TYPE_ECDSA_PUBLIC_KEY

```
#define TEE_TYPE_ECDSA_PUBLIC_KEY 0xA0000041
```

#### 12.5.1.323 TEE_TYPE_GENERIC_SECRET

```
#define TEE_TYPE_GENERIC_SECRET 0xA0000000
```

#### 12.5.1.324 TEE_TYPE_HMAC_MD5

```
#define TEE_TYPE_HMAC_MD5 0xA0000001
```

#### 12.5.1.325 TEE_TYPE_HMAC_SHA1

```
#define TEE_TYPE_HMAC_SHA1 0xA0000002
```

#### 12.5.1.326 TEE_TYPE_HMAC_SHA224

```
#define TEE_TYPE_HMAC_SHA224 0xA0000003
```

#### 12.5.1.327 TEE_TYPE_HMAC_SHA256

```
#define TEE_TYPE_HMAC_SHA256 0xA0000004
```

#### 12.5.1.328 TEE_TYPE_HMAC_SHA384

```
#define TEE_TYPE_HMAC_SHA384 0xA0000005
```

#### 12.5.1.329 TEE_TYPE_HMAC_SHA512

```
#define TEE_TYPE_HMAC_SHA512 0xA0000006
```

#### 12.5.1.330 TEE_TYPE_RSA_KEYPAIR

```
#define TEE_TYPE_RSA_KEYPAIR 0xA1000030
```

#### 12.5.1.331 TEE_TYPE_RSA_PUBLIC_KEY

```
#define TEE_TYPE_RSA_PUBLIC_KEY 0xA0000030
```

#### 12.5.1.332 TEE_USAGE_DECRYPT

```
#define TEE_USAGE_DECRYPT 0x00000004
```

#### 12.5.1.333 TEE_USAGE_DERIVE

```
#define TEE_USAGE_DERIVE 0x00000040
```

#### 12.5.1.334 TEE_USAGE_ENCRYPT

```
#define TEE_USAGE_ENCRYPT 0x00000002
```

#### 12.5.1.335 TEE_USAGE_EXTRACTABLE

```
#define TEE_USAGE_EXTRACTABLE 0x00000001
```

#### 12.5.1.336 TEE_USAGE_MAC

```
#define TEE_USAGE_MAC 0x00000008
```

#### 12.5.1.337 TEE_USAGE_SIGN

```
#define TEE_USAGE_SIGN 0x00000010
```

#### 12.5.1.338 TEE_USAGE_VERIFY

```
#define TEE_USAGE_VERIFY 0x00000020
```

## 12.6 include/tee_api_defines_extensions.h File Reference

### Macros

- #define TEE_ALG_HKDF_MD5_DERIVE_KEY 0x800010C0
- #define TEE_ALG_HKDF_SHA1_DERIVE_KEY 0x800020C0
- #define TEE_ALG_HKDF_SHA224_DERIVE_KEY 0x800030C0
- #define TEE_ALG_HKDF_SHA256_DERIVE_KEY 0x800040C0
- #define TEE_ALG_HKDF_SHA384_DERIVE_KEY 0x800050C0
- #define TEE_ALG_HKDF_SHA512_DERIVE_KEY 0x800060C0
- #define TEE_TYPE_HKDF_IKM 0xA10000C0
- #define TEE_ATTR_HKDF_IKM 0xC00001C0
- #define TEE_ATTR_HKDF_SALT 0xD00002C0
- #define TEE_ATTR_HKDF_INFO 0xD00003C0
- #define TEE_ATTR_HKDF_OKM_LENGTH 0xF00004C0
- #define TEE_ALG_CONCAT_KDF_SHA1_DERIVE_KEY 0x800020C1
- #define TEE_ALG_CONCAT_KDF_SHA224_DERIVE_KEY 0x800030C1
- #define TEE_ALG_CONCAT_KDF_SHA256_DERIVE_KEY 0x800040C1

- #define TEE_ALG_CONCAT_KDF_SHA384_DERIVE_KEY 0x800050C1
- #define TEE_ALG_CONCAT_KDF_SHA512_DERIVE_KEY 0x800060C1
- #define TEE_TYPE_CONCAT_KDF_Z 0xA10000C1
- #define TEE_ATTR_CONCAT_KDF_Z 0xC00001C1
- #define TEE_ATTR_CONCAT_KDF_OTHER_INFO 0xD00002C1
- #define TEE_ATTR_CONCAT_KDF_DKM_LENGTH 0xF00003C1
- #define TEE_ALG_PBKDF2_HMAC_SHA1_DERIVE_KEY 0x800020C2
- #define TEE_TYPE_PBKDF2_PASSWORD 0xA10000C2
- #define TEE_ATTR_PBKDF2_PASSWORD 0xC00001C2
- #define TEE_ATTR_PBKDF2_SALT 0xD00002C2
- #define TEE_ATTR_PBKDF2_ITERATION_COUNT 0xF00003C2
- #define TEE_ATTR_PBKDF2_DKM_LENGTH 0xF00004C2
- #define TEE_STORAGE_PRIVATE_REE 0x80000000
- #define TEE_STORAGE_PRIVATE_RPMB 0x80000100
- #define TEE_STORAGE_PRIVATE_SQL_RESERVED 0x80000200
- #define TEE_MEMORY_ACCESS_NONSECURE 0x10000000
- #define TEE_MEMORY_ACCESS_SECURE 0x20000000

### 12.6.1 Macro Definition Documentation

#### 12.6.1.1 TEE_ALG_CONCAT_KDF_SHA1_DERIVE_KEY

```
#define TEE_ALG_CONCAT_KDF_SHA1_DERIVE_KEY 0x800020C1
```

#### 12.6.1.2 TEE_ALG_CONCAT_KDF_SHA224_DERIVE_KEY

```
#define TEE_ALG_CONCAT_KDF_SHA224_DERIVE_KEY 0x800030C1
```

#### 12.6.1.3 TEE_ALG_CONCAT_KDF_SHA256_DERIVE_KEY

```
#define TEE_ALG_CONCAT_KDF_SHA256_DERIVE_KEY 0x800040C1
```

#### 12.6.1.4 TEE_ALG_CONCAT_KDF_SHA384_DERIVE_KEY

```
#define TEE_ALG_CONCAT_KDF_SHA384_DERIVE_KEY 0x800050C1
```

#### 12.6.1.5 TEE_ALG_CONCAT_KDF_SHA512_DERIVE_KEY

```
#define TEE_ALG_CONCAT_KDF_SHA512_DERIVE_KEY 0x800060C1
```

#### 12.6.1.6 TEE_ALG_HKDF_MD5_DERIVE_KEY

```
#define TEE_ALG_HKDF_MD5_DERIVE_KEY 0x800010C0
```

#### 12.6.1.7 TEE_ALG_HKDF_SHA1_DERIVE_KEY

```
#define TEE_ALG_HKDF_SHA1_DERIVE_KEY 0x800020C0
```

#### 12.6.1.8 TEE_ALG_HKDF_SHA224_DERIVE_KEY

```
#define TEE_ALG_HKDF_SHA224_DERIVE_KEY 0x800030C0
```

#### 12.6.1.9 TEE_ALG_HKDF_SHA256_DERIVE_KEY

```
#define TEE_ALG_HKDF_SHA256_DERIVE_KEY 0x800040C0
```

#### 12.6.1.10 TEE_ALG_HKDF_SHA384_DERIVE_KEY

```
#define TEE_ALG_HKDF_SHA384_DERIVE_KEY 0x800050C0
```

#### 12.6.1.11 TEE_ALG_HKDF_SHA512_DERIVE_KEY

```
#define TEE_ALG_HKDF_SHA512_DERIVE_KEY 0x800060C0
```

#### 12.6.1.12 TEE_ALG_PBKDF2_HMAC_SHA1_DERIVE_KEY

```
#define TEE_ALG_PBKDF2_HMAC_SHA1_DERIVE_KEY 0x800020C2
```

#### 12.6.1.13 TEE_ATTR_CONCAT_KDF_DKM_LENGTH

```
#define TEE_ATTR_CONCAT_KDF_DKM_LENGTH 0xF00003C1
```

#### 12.6.1.14 TEE_ATTR_CONCAT_KDF_OTHER_INFO

```
#define TEE_ATTR_CONCAT_KDF_OTHER_INFO 0xD00002C1
```

#### 12.6.1.15 TEE_ATTR_CONCAT_KDF_Z

```
#define TEE_ATTR_CONCAT_KDF_Z 0xC00001C1
```

#### 12.6.1.16 TEE_ATTR_HKDF_IKM

```
#define TEE_ATTR_HKDF_IKM 0xC00001C0
```

#### 12.6.1.17 TEE_ATTR_HKDF_INFO

```
#define TEE_ATTR_HKDF_INFO 0xD00003C0
```

#### 12.6.1.18 TEE_ATTR_HKDF_OKM_LENGTH

```
#define TEE_ATTR_HKDF_OKM_LENGTH 0xF00004C0
```

#### 12.6.1.19 TEE_ATTR_HKDF_SALT

```
#define TEE_ATTR_HKDF_SALT 0xD00002C0
```

#### 12.6.1.20 TEE_ATTR_PBKDF2_DKM_LENGTH

```
#define TEE_ATTR_PBKDF2_DKM_LENGTH 0xF00004C2
```

#### 12.6.1.21 TEE_ATTR_PBKDF2_ITERATION_COUNT

```
#define TEE_ATTR_PBKDF2_ITERATION_COUNT 0xF00003C2
```

#### 12.6.1.22 TEE_ATTR_PBKDF2_PASSWORD

```
#define TEE_ATTR_PBKDF2_PASSWORD 0xC00001C2
```

#### 12.6.1.23 TEE_ATTR_PBKDF2_SALT

```
#define TEE_ATTR_PBKDF2_SALT 0xD00002C2
```

#### 12.6.1.24 TEE_MEMORY_ACCESS_NONSECURE

```
#define TEE_MEMORY_ACCESS_NONSECURE 0x10000000
```

#### 12.6.1.25 TEE_MEMORY_ACCESS_SECURE

```
#define TEE_MEMORY_ACCESS_SECURE 0x20000000
```

#### 12.6.1.26 TEE_STORAGE_PRIVATE_REE

```
#define TEE_STORAGE_PRIVATE_REE 0x80000000
```

#### 12.6.1.27 TEE_STORAGE_PRIVATE_RPMB

```
#define TEE_STORAGE_PRIVATE_RPMB 0x80000100
```

#### 12.6.1.28 TEE_STORAGE_PRIVATE_SQL_RESERVED

```
#define TEE_STORAGE_PRIVATE_SQL_RESERVED 0x80000200
```

#### 12.6.1.29 TEE_TYPE_CONCAT_KDF_Z

```
#define TEE_TYPE_CONCAT_KDF_Z 0xA10000C1
```

#### 12.6.1.30 TEE_TYPE_HKDF_IKM

```
#define TEE_TYPE_HKDF_IKM 0xA10000C0
```

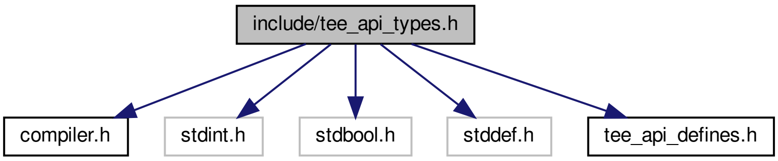
#### 12.6.1.31 TEE_TYPE_PBKDF2_PASSWORD

```
#define TEE_TYPE_PBKDF2_PASSWORD 0xA10000C2
```

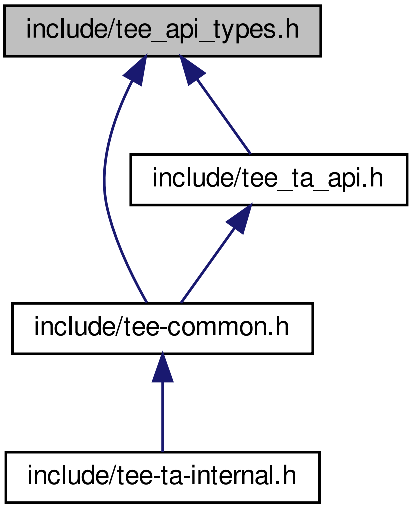
## 12.7 include/tee_api_types.h File Reference

```
#include <compiler.h>
#include <stdint.h>
#include <stdbool.h>
#include <stddef.h>
#include <tee_api_defines.h>
```

Include dependency graph for tee_api_types.h:



This graph shows which files directly or indirectly include this file:



### Classes

- struct TEE_UUID
- struct TEE_Identity
- union TEE_Param
- struct TEE_ObjectInfo
- struct TEE_Attribute
- struct TEE_OperationInfo

- struct TEE_OperationInfoKey
- struct TEE_OperationInfoMultiple
- struct TEE_Time
- struct TEE_SEReaderProperties
- struct TEE_SEAID
- struct pollfd
- struct addrinfo

### Macros

- #define DMREQ_FINISH 0
- #define DMREQ_WRITE 1
- #define TEE_MEM_INPUT 0x00000001
- #define TEE_MEM_OUTPUT 0x00000002
- #define TEE_MEMREF_0_USED 0x00000001
- #define TEE_MEMREF_1_USED 0x00000002
- #define TEE_MEMREF_2_USED 0x00000004
- #define TEE_MEMREF_3_USED 0x00000008
- #define TEE_SE_READER_NAME_MAX 20
- #define socklen_t unsigned int

### Typedefs

- typedef uint32_t TEE_Result
- typedef struct __TEE_TASessionHandle ∗ TEE_TASessionHandle
- typedef struct __TEE_PropSetHandle ∗ TEE_PropSetHandle
- typedef struct __TEE_ObjectHandle ∗ TEE_ObjectHandle
- typedef struct __TEE_ObjectEnumHandle ∗ TEE_ObjectEnumHandle
- typedef struct __TEE_OperationHandle ∗ TEE_OperationHandle
- typedef uint32_t TEE_ObjectType
- typedef uint32_t TEE_BigInt
- typedef uint32_t TEE_BigIntFMM
- typedef uint32_t TEE_BigIntFMMContext __aligned(__alignof__(void ∗))
- typedef struct __TEE_SEServiceHandle ∗ TEE_SEServiceHandle
- typedef struct __TEE_SEReaderHandle ∗ TEE_SEReaderHandle
- typedef struct __TEE_SESessionHandle ∗ TEE_SESessionHandle
- typedef struct __TEE_SEChannelHandle ∗ TEE_SEChannelHandle
- typedef uint32_t TEE_ErrorOrigin
- typedef void ∗ TEE_Session
- typedef unsigned long int nfds_t

### Enumerations

- enum TEE_Whence { TEE_DATA_SEEK_SET = 0, TEE_DATA_SEEK_CUR = 1, TEE_DATA_SEEK_END = 2 }
- enum TEE_OperationMode {
  TEE_MODE_ENCRYPT = 0, TEE_MODE_DECRYPT = 1, TEE_MODE_SIGN = 2, TEE_MODE_VERIFY = 3,
  TEE_MODE_MAC = 4, TEE_MODE_DIGEST = 5, TEE_MODE_DERIVE = 6 }

### 12.7.1 Macro Definition Documentation

#### 12.7.1.1 DMREQ_FINISH

```
#define DMREQ_FINISH 0
```

#### 12.7.1.2 DMREQ_WRITE

```
#define DMREQ_WRITE 1
```

#### 12.7.1.3 socklen_t

```
#define socklen_t unsigned int
```

#### 12.7.1.4 TEE_MEM_INPUT

```
#define TEE_MEM_INPUT 0x00000001
```

#### 12.7.1.5 TEE_MEM_OUTPUT

```
#define TEE_MEM_OUTPUT 0x00000002
```

#### 12.7.1.6 TEE_MEMREF_0_USED

```
#define TEE_MEMREF_0_USED 0x00000001
```

#### 12.7.1.7 TEE_MEMREF_1_USED

```
#define TEE_MEMREF_1_USED 0x00000002
```

#### 12.7.1.8 TEE_MEMREF_2_USED

```
#define TEE_MEMREF_2_USED 0x00000004
```

#### 12.7.1.9 TEE_MEMREF_3_USED

```
#define TEE_MEMREF_3_USED 0x00000008
```

#### 12.7.1.10 TEE_SE_READER_NAME_MAX

```
#define TEE_SE_READER_NAME_MAX 20
```

### 12.7.2 Typedef Documentation

#### 12.7.2.1 __aligned

```
typedef uint32_t TEE_BigIntFMMContext __aligned(__alignof__(void *))
```

#### 12.7.2.2 nfds_t

```
typedef unsigned long int nfds_t
```

#### 12.7.2.3 TEE_BigInt

```
typedef uint32_t TEE_BigInt
```

#### 12.7.2.4 TEE_BigIntFMM

```
typedef uint32_t TEE_BigIntFMM
```

#### 12.7.2.5 TEE_ErrorOrigin

```
typedef uint32_t TEE_ErrorOrigin
```

#### 12.7.2.6 TEE_ObjectEnumHandle

```
typedef struct __TEE_ObjectEnumHandle* TEE_ObjectEnumHandle
```

#### 12.7.2.7 TEE_ObjectHandle

```
typedef struct __TEE_ObjectHandle* TEE_ObjectHandle
```

#### 12.7.2.8 TEE_ObjectType

```
typedef uint32_t TEE_ObjectType
```

#### 12.7.2.9 TEE_OperationHandle

```
typedef struct __TEE_OperationHandle* TEE_OperationHandle
```

#### 12.7.2.10 TEE_PropSetHandle

```
typedef struct __TEE_PropSetHandle* TEE_PropSetHandle
```

#### 12.7.2.11 TEE_Result

```
typedef uint32_t TEE_Result
```

#### 12.7.2.12 TEE_SEChannelHandle

```
typedef struct __TEE_SEChannelHandle* TEE_SEChannelHandle
```

#### 12.7.2.13 TEE_SEReaderHandle

```
typedef struct __TEE_SEReaderHandle* TEE_SEReaderHandle
```

#### 12.7.2.14 TEE_SEServiceHandle

```
typedef struct __TEE_SEServiceHandle* TEE_SEServiceHandle
```

#### 12.7.2.15 TEE_SESessionHandle

```
typedef struct __TEE_SESessionHandle* TEE_SESessionHandle
```

#### 12.7.2.16 TEE_Session

```
typedef void* TEE_Session
```

#### 12.7.2.17 TEE_TASessionHandle

```
typedef struct __TEE_TASessionHandle* TEE_TASessionHandle
```

### 12.7.3 Enumeration Type Documentation

#### 12.7.3.1 TEE_OperationMode

```
enum TEE_OperationMode
```

**Enumerator**

| | |
|---|---|
| TEE_MODE_ENCRYPT | |
| TEE_MODE_DECRYPT | |
| TEE_MODE_SIGN | |
| TEE_MODE_VERIFY | |
| TEE_MODE_MAC | |
| TEE_MODE_DIGEST | |
| TEE_MODE_DERIVE | |

#### 12.7.3.2 TEE_Whence

```
enum TEE_Whence
```

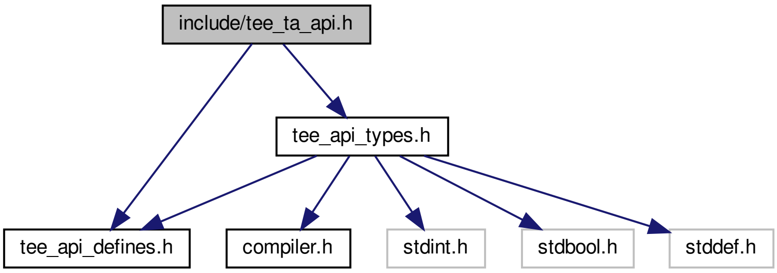
**Enumerator**

| | |
|---|---|
| TEE_DATA_SEEK_SET | |
| TEE_DATA_SEEK_CUR | |
| TEE_DATA_SEEK_END | |

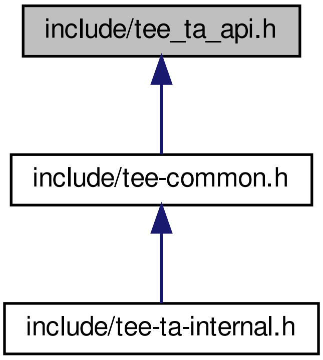
## 12.8 include/tee_ta_api.h File Reference

```
#include <tee_api_defines.h>
#include <tee_api_types.h>
```

Include dependency graph for tee_ta_api.h:



This graph shows which files directly or indirectly include this file:



### Macros

- #define TA_EXPORT

### Functions

- TEE_Result TA_EXPORT TA_CreateEntryPoint (void)
- void TA_EXPORT TA_DestroyEntryPoint (void)
- TEE_Result TA_EXPORT TA_OpenSessionEntryPoint (uint32_t paramTypes, TEE_Param params[TEE_NUM_PARAMS], void ∗∗sessionContext)
- void TA_EXPORT TA_CloseSessionEntryPoint (void ∗sessionContext)
- TEE_Result TA_EXPORT TA_InvokeCommandEntryPoint (void ∗sessionContext, uint32_t commandID, uint32_t paramTypes, TEE_Param params[TEE_NUM_PARAMS])

### 12.8.1 Macro Definition Documentation

#### 12.8.1.1 TA_EXPORT

```
#define TA_EXPORT
```

### 12.8.2 Function Documentation

#### 12.8.2.1 TA_CloseSessionEntryPoint()

```
void TA_EXPORT TA_CloseSessionEntryPoint (
             void * sessionContext )
```

#### 12.8.2.2 TA_CreateEntryPoint()

```
TEE_Result TA_EXPORT TA_CreateEntryPoint (
             void  )
```

#### 12.8.2.3 TA_DestroyEntryPoint()

```
void TA_EXPORT TA_DestroyEntryPoint (
             void  )
```

#### 12.8.2.4 TA_InvokeCommandEntryPoint()

```
TEE_Result TA_EXPORT TA_InvokeCommandEntryPoint (
             void * sessionContext,
             uint32_t commandID,
             uint32_t paramTypes,
             TEE_Param params[TEE_NUM_PARAMS] )
```

#### 12.8.2.5 TA_OpenSessionEntryPoint()

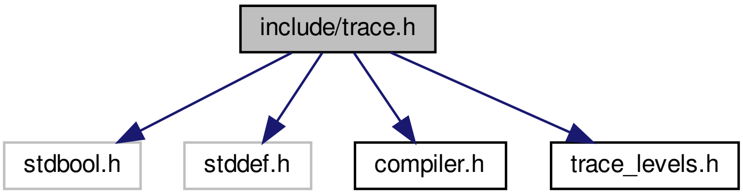
```
TEE_Result TA_EXPORT TA_OpenSessionEntryPoint (
             uint32_t paramTypes,
             TEE_Param params[TEE_NUM_PARAMS],
             void ** sessionContext )
```

## 12.9 include/test_dev_key.h File Reference

## 12.10 include/trace.h File Reference

```
#include <stdbool.h>
#include <stddef.h>
#include <compiler.h>
#include <trace_levels.h>
```

Include dependency graph for trace.h:



### Macros

- #define MAX_PRINT_SIZE 256
- #define MAX_FUNC_PRINT_SIZE 32
- #define TRACE_LEVEL TRACE_MAX
- #define trace_printf_helper(level, level_ok, ...)
- #define MSG(...) (void)0
- #define EMSG(...) trace_printf_helper(TRACE_ERROR, true, __VA_ARGS__)
- #define IMSG(...) trace_printf_helper(TRACE_INFO, true, __VA_ARGS__)
- #define DMSG(...) trace_printf_helper(TRACE_DEBUG, true, __VA_ARGS__)
- #define FMSG(...) trace_printf_helper(TRACE_FLOW, true, __VA_ARGS__)
- #define INMSG(...) FMSG("> " __VA_ARGS__)
- #define OUTMSG(...) FMSG("< " __VA_ARGS__)
- #define OUTRMSG(r)
- #define DHEXDUMP(buf, len)
- #define trace_printf_helper_raw(level, level_ok, ...) trace_printf(NULL, 0, (level), (level_ok), __VA_ARGS__)
- #define MSG_RAW(...) (void)0
- #define EMSG_RAW(...) trace_printf_helper_raw(TRACE_ERROR, true, __VA_ARGS__)
- #define IMSG_RAW(...) trace_printf_helper_raw(TRACE_INFO, true, __VA_ARGS__)
- #define DMSG_RAW(...) trace_printf_helper_raw(TRACE_DEBUG, true, __VA_ARGS__)
- #define FMSG_RAW(...) trace_printf_helper_raw(TRACE_FLOW, true, __VA_ARGS__)
- #define SMSG(...) (void)0
- #define EPRINT_STACK() (void)0
- #define IPRINT_STACK() (void)0
- #define DPRINT_STACK() (void)0
- #define FPRINT_STACK() (void)0

### Functions

- void trace_ext_puts (const char ∗str)
- int trace_ext_get_thread_id (void)
- void trace_set_level (int level)
- int trace_get_level (void)
- void trace_printf (const char ∗func, int line, int level, bool level_ok, const char ∗fmt,...) __printf(5
- void dhex_dump (const char ∗function, int line, int level, const void ∗buf, int len)

### Variables

- int trace_level
- const char trace_ext_prefix [ ]

## 12.10.1 Macro Definition Documentation

### 12.10.1.1 DHEXDUMP

```
#define DHEXDUMP(
              buf,
              len )
```

**Value:**

```
dhex_dump(__func__, __LINE__, TRACE_DEBUG, \
                  buf, len)
```

### 12.10.1.2 DMSG

```
#define DMSG(
              ... ) trace_printf_helper(TRACE_DEBUG, true, __VA_ARGS__)
```

### 12.10.1.3 DMSG_RAW

```
#define DMSG_RAW(
              ... ) trace_printf_helper_raw(TRACE_DEBUG, true, __VA_ARGS__)
```

### 12.10.1.4 DPRINT_STACK

```
#define DPRINT_STACK( ) (void)0
```

#### 12.10.1.5 EMSG

```
#define EMSG(
              ... ) trace_printf_helper(TRACE_ERROR, true, __VA_ARGS__)
```

#### 12.10.1.6 EMSG_RAW

```
#define EMSG_RAW(
              ... ) trace_printf_helper_raw(TRACE_ERROR, true, __VA_ARGS__)
```

#### 12.10.1.7 EPRINT_STACK

```
#define EPRINT_STACK( ) (void)0
```

#### 12.10.1.8 FMSG

```
#define FMSG(
              ... ) trace_printf_helper(TRACE_FLOW, true, __VA_ARGS__)
```

#### 12.10.1.9 FMSG_RAW

```
#define FMSG_RAW(
              ... ) trace_printf_helper_raw(TRACE_FLOW, true, __VA_ARGS__)
```

#### 12.10.1.10 FPRINT_STACK

```
#define FPRINT_STACK( ) (void)0
```

#### 12.10.1.11 IMSG

```
#define IMSG(
              ... ) trace_printf_helper(TRACE_INFO, true, __VA_ARGS__)
```

#### 12.10.1.12 IMSG_RAW

```
#define IMSG_RAW(
              ... ) trace_printf_helper_raw(TRACE_INFO, true, __VA_ARGS__)
```

#### 12.10.1.13 INMSG

```
#define INMSG(
              ... ) FMSG("> " __VA_ARGS__)
```

#### 12.10.1.14 IPRINT_STACK

```
#define IPRINT_STACK( ) (void)0
```

#### 12.10.1.15 MAX_FUNC_PRINT_SIZE

```
#define MAX_FUNC_PRINT_SIZE 32
```

#### 12.10.1.16 MAX_PRINT_SIZE

```
#define MAX_PRINT_SIZE 256
```

#### 12.10.1.17 MSG

```
#define MSG(
              ... ) (void)0
```

#### 12.10.1.18 MSG_RAW

```
#define MSG_RAW(
              ... ) (void)0
```

#### 12.10.1.19 OUTMSG

```
#define OUTMSG(
              ... ) FMSG("< " __VA_ARGS__)
```

#### 12.10.1.20 OUTRMSG

```
#define OUTRMSG(
              r )
```

**Value:**

```
do {                    \
        OUTMSG("r=[%x]", r);    \
        return r;        \
    } while (0)
```

#### 12.10.1.21 SMSG

```
#define SMSG(
              ... ) (void)0
```

#### 12.10.1.22 TRACE_LEVEL

```
#define TRACE_LEVEL TRACE_MAX
```

#### 12.10.1.23 trace_printf_helper

```
#define trace_printf_helper(
              level,
              level_ok,
              ... )
```

**Value:**

```
trace_printf(__func__, __LINE__, (level), (level_ok), \
             __VA_ARGS__)
```

#### 12.10.1.24 trace_printf_helper_raw

```
#define trace_printf_helper_raw(
              level,
              level_ok,
              ... ) trace_printf(NULL, 0, (level), (level_ok), __VA_ARGS__)
```

### 12.10.2 Function Documentation

#### 12.10.2.1 dhex_dump()

```
void dhex_dump (
             const char * function,
             int line,
             int level,
             const void * buf,
             int len )
```

#### 12.10.2.2 trace_ext_get_thread_id()

```
int trace_ext_get_thread_id (
             void  )
```

#### 12.10.2.3 trace_ext_puts()

```
void trace_ext_puts (
             const char * str )
```

#### 12.10.2.4 trace_get_level()

```
int trace_get_level (
             void  )
```

#### 12.10.2.5 trace_printf()

```
void trace_printf (
             const char * func,
             int line,
             int level,
             bool level_ok,
             const char * fmt,
              ...  )
```

#### 12.10.2.6 trace_set_level()

```
void trace_set_level (
             int level )
```

### 12.10.3 Variable Documentation

#### 12.10.3.1 trace_ext_prefix

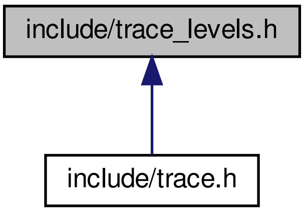
```
const char trace_ext_prefix[]
```

#### 12.10.3.2 trace_level

```
int trace_level
```

## 12.11 include/trace_levels.h File Reference

This graph shows which files directly or indirectly include this file:



### Macros

- #define TRACE_MIN 1
- #define TRACE_ERROR TRACE_MIN
- #define TRACE_INFO 2
- #define TRACE_DEBUG 3
- #define TRACE_FLOW 4
- #define TRACE_MAX TRACE_FLOW
- #define TRACE_PRINTF_LEVEL TRACE_ERROR

### 12.11.1 Macro Definition Documentation

#### 12.11.1.1 TRACE_DEBUG

```
#define TRACE_DEBUG 3
```

#### 12.11.1.2 TRACE_ERROR

```
#define TRACE_ERROR TRACE_MIN
```

#### 12.11.1.3 TRACE_FLOW

```
#define TRACE_FLOW 4
```

#### 12.11.1.4 TRACE_INFO

```
#define TRACE_INFO 2
```

#### 12.11.1.5 TRACE_MAX

```
#define TRACE_MAX TRACE_FLOW
```

#### 12.11.1.6 TRACE_MIN

```
#define TRACE_MIN 1
```

#### 12.11.1.7 TRACE_PRINTF_LEVEL

```
#define TRACE_PRINTF_LEVEL TRACE_ERROR
```

## 12.12 mainpage.md File Reference

## 12.13 message-digest.md File Reference

## 12.14 readme-implementation.md File Reference

## 12.15 readme-test.md File Reference

## 12.16 README.md File Reference

## 12.17 secure-storage.md File Reference

## 12.18 symmetric-key-varification.md File Reference

# Index