GLOBALPLATFORM®

THE STANDARD FOR MANAGING APPLICATIONS ON SECURE CHIP TECHNOLOGY

GlobalPlatform Device Technology

TEE Client API Specification v1.0 Errata and Precisions

Version 2.0

Public Release

April 2014

Document Reference: GPD_EPR_028

THIS SPECIFICATION OR OTHER WORK PRODUCT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY IMPLEMENTATION OF THIS SPECIFICATION OR OTHER WORK PRODUCT SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE COMPANY, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER DIRECTLY OR INDIRECTLY ARISING FROM THE IMPLEMENTATION OF THIS SPECIFICATION OR OTHER WORK PRODUCT.

Contents

A Summary

This document includes all the Errata and Precisions that have been released for TEE Client API Specification v1.0.

A.1 Summary of Errata and Precisions

Errata and precisions are numbered according to the sequence in which they were developed, However, Table A-1 lists the errata and precisions in the order that the affected text occurs in TEE Client API Specification v1.0.

The latest Errata and Precisions are shown in blue characters in Table A-1.

Table A-1: Summary of Errata and Precisions

Errata / Precision	TEE Client API Specification v1.0 reference	Description
P.2	Section 1.2.1, Table 1-1	Add Normative Reference
P.3	Section 3.2.3	Clarify Return Code When a Fatal Error Occurs in a Trusted Application
E.1	Section 3.2.5, Section 4.4.4, Table 4-5	Correct reference to `TEEC_MemoryReference`
E.2	Section 4.3.11	Correct reference to `TEEC_PARAMS_TYPE`
P.4	Section 4.4.2, Table 4-2	Define Additional Return Codes
P.1	Section 4.5.7	Clarify State of Operation Parameters on Return from `TEEC_OpenSession`

A.2 Revision History

Table A-2: Revision History

Date	Version	Description
October 2013	1.0	Initial release
April 2014	2.0	Additional precisions shown in blue in Table A-1

E Errata

E.1 Correct Reference to TEEC_MemoryReference

In section 3.2.5, paragraph 1 which read:

> A *Memory Reference* is a range of bytes which is actually shared for a particular operation. A Memory Reference is described by either a **TEEC_MemoryReference** structure (see section **4.3.7**) or a TEEC_TempMemoryReference. It can specify:

shall be replaced with:

> A *Memory Reference* is a range of bytes which is actually shared for a particular operation. A Memory Reference is described by either a **TEEC_RegisteredMemoryReference** structure (see section **4.3.8)** or a TEEC_TempMemoryReference. It can specify:

In section 4.4.4, Table 4-5: API Parameter Types, the entry for TEEC_MEMREF_WHOLE which reads:

TEEC_MEMREF_WHOLE	0x0000000C	The Parameter is a Registered Memory Reference that refers to the entirety of its parent Shared Memory block. The parameter structure is a **TEEC_MemoryReference**. In this structure, the Implementation MUST read only the parent field and MAY update the size field when the operation completes.

shall be replaced with:

TEEC_MEMREF_WHOLE	0x0000000C	The Parameter is a Registered Memory Reference that refers to the entirety of its parent Shared Memory block. The parameter structure is a **TEEC_RegisteredMemoryReference**. In this structure, the Implementation MUST read only the parent field and MAY update the size field when the operation completes.

E.2 Correct Reference to TEEC_PARAMS_TYPE

In section 4.3.11, the definition of the paramTypes member which reads:

- paramTypes encodes the type of each of the Parameters in the operation. The layout of these types within a 32-bit integer is *implementation-defined* and the Client Application MUST use the macro **TEEC_PARAMS_TYPE** to construct a constant value for this field. As a special case, if the Client Application sets paramTypes to 0, then the Implementation MUST interpret it as meaning that the type for each Parameter is set to TEEC_NONE.

shall be replaced with:

- paramTypes encodes the type of each of the Parameters in the operation. The layout of these types within a 32-bit integer is *implementation-defined* and the Client Application MUST use the macro **TEEC_PARAM_TYPES** to construct a constant value for this field. As a special case, if the Client Application sets paramTypes to 0, then the Implementation MUST interpret it as meaning that the type for each Parameter is set to TEEC_NONE.

P Precisions

P.1 Clarify State of Operation Parameters on Return from TEEC_OpenSession

In section 4.5.7, TEEC_OpenSession, it is not clear under what circumstances the parameters within the Operation structure pointed to by the operation argument are valid.

Replace the final paragraph and contained bullet points of the Description section with:

> The result of this function is returned **in the function TEEC_Result return code, the return origin stored in the variable pointed to by returnOrigin, and the contents of the Operation structure pointed to by operation**:
>
> - If the return origin is different from TEEC_ORIGIN_TRUSTED_APP, then the return code MUST be one of the error codes defined in Table 4-2. If the return code is TEEC_ERROR_CANCEL then it means that the operation was cancelled before it reached the Trusted Application. **The contents of the Operation are unchanged in this case.**
> - If the return origin is TEEC_ORIGIN_TRUSTED_APP, the meaning of the return code depends on the protocol between the Client Application and the Trusted Application. However, if TEEC_SUCCESS is returned, it always means that the session was successfully opened and if the function returns a code different from TEEC_SUCCESS, it means that the session opening failed. **The contents of the Operation are updated to match those set by the Trusted Application in this case even if a failure code is returned.**

P.2 Add Normative Reference

Add the following line to Table 1-1, Normative References, in section 1.2.1 (to support Precision P.4):

GPD_SPE_020	Trusted User Interface API	[6]

P.3 Clarify Return Code When a Fatal Error Occurs in a Trusted Application

In the section sub-titled “Return Codes and Return Origins” of section 3.2.3, append the following paragraph:

> If the Trusted Application that a session is communicating with encounters a fatal error (sometimes known as a panic), then the current and all subsequent commands will return the error TEE_ERROR_TARGET_DEAD with the origin TEE_ORIGIN_TEE until the session is closed.

P.4 Define Additional Return Codes

Replace table 4-2 in section 4.4.2 with the following:

Name	Value	Description / Cause
TEEC_SUCCESS	0x00000000	The operation was successful.
Implementation-Defined	0x00000001 - 0xFFFEFFFF	
TEEC_ERROR_GENERIC	0xFFFF0000	Non-specific cause.
TEEC_ERROR_ACCESS_DENIED	0xFFFF0001	Access privileges are not sufficient.
TEEC_ERROR_CANCEL	0xFFFF0002	The operation was cancelled.
TEEC_ERROR_ACCESS_CONFLICT	0xFFFF0003	Concurrent accesses caused conflict.
TEEC_ERROR_EXCESS_DATA	0xFFFF0004	Too much data for the requested operation was passed.
TEEC_ERROR_BAD_FORMAT	0xFFFF0005	Input data was of invalid format.
TEEC_ERROR_BAD_PARAMETERS	0xFFFF0006	Input parameters were invalid.
TEEC_ERROR_BAD_STATE	0xFFFF0007	Operation is not valid in the current state.
TEEC_ERROR_ITEM_NOT_FOUND	0xFFFF0008	The requested data item is not found.
TEEC_ERROR_NOT_IMPLEMENTED	0xFFFF0009	The requested operation should exist but is not yet implemented.
TEEC_ERROR_NOT_SUPPORTED	0xFFFF000A	The requested operation is valid but is not supported in this Implementation.
TEEC_ERROR_NO_DATA	0xFFFF000B	Expected data was missing.
TEEC_ERROR_OUT_OF_MEMORY	0xFFFF000C	System ran out of resources.
TEEC_ERROR_BUSY	0xFFFF000D	The system is busy working on something else.
TEEC_ERROR_COMMUNICATION	0xFFFF000E	Communication with a remote party failed.
TEEC_ERROR_SECURITY	0xFFFF000F	A security fault was detected.
TEEC_ERROR_SHORT_BUFFER	0xFFFF0010	The supplied buffer is too short for the generated output.
TEE_ERROR_EXTERNAL_CANCEL	**0xFFFF0011**	**Defined by the Trusted User Interface specification [6]: An external event has caused a User Interface operation to be aborted.**
TEE_ERROR_OVERFLOW	**0xFFFF300F**	**Internal TEE error – documented for completeness**
TEE_ERROR_TARGET_DEAD, TEEC_ERROR_TARGET_DEAD	**0xFFFF3024**	**The Trusted Application has terminated.**
TEE_ERROR_STORAGE_NO_SPACE	**0xFFFF3041**	**Internal TEE error – documented for completeness**

Name	Value	Description / Cause
TEE_ERROR_MAC_INVALID	**0xFFFF3071**	**Internal TEE error – documented for completeness**
TEE_ERROR_SIGNATURE_INVALID	**0xFFFF3072**	**Internal TEE error – documented for completeness**
TEE_ERROR_TIME_NOT_SET	**0xFFFF5000**	**Internal TEE error – documented for completeness**
TEE_ERROR_TIME_NEEDS_RESET	**0xFFFF5001**	**Internal TEE error – documented for completeness**
Reserved for Future Use	**All values in the range 0xFFFF0000 – 0xFFFFFFFF which are not defined in this table**	