TRUSTED® COMPUTING GROUP

ERRATA

Errata for TCG PC Client Platform TPM Profile for TPM 2.0
Version 1.07

Version 1.1

August 27, 2026

Contact: admin@trustedcomputinggroup.org

Published

DISCLAIMERS, NOTICES, AND LICENSE TERMS

CONTENTS

1 Introduction

This document describes errata and clarifications for the TCG PC Client Platform TPM Profile for TPM 2.0 Version 1.07 as published. The information in this document is likely – but not certain – to be incorporated into a future version of the specification. Suggested fixes proposed in this document may be modified before being published in a later TCG Specification. Therefore, the contents of this document are not normative and only become normative when included in an updated version of the published specification. Note that since the errata in this document are non-normative, the patent licensing rights granted by Section 16.4 of the Bylaws do not apply.

2 Errata

2.1 Errata 1 Table 3 Algorithm TPM_ALG_MLDSA Comments

The second sentence in the comment for TPM_ALG_MLDSA contains a typographical error. The current text reads as follows:

> This is a PQC signing algorithm
> The TPM SHALL support either ML-DSA-65 or ML-DSA-87. The TPM SHOULD support ML-DSA-44, ML-DSA-65, ***or*** ML-DSA-87. ML-DSA support is mandatory as of PTP version 1.07.

The second sentence is corrected as follows:

> This is a PQC signing algorithm
> The TPM SHALL support either ML-DSA-65 or ML-DSA-87. The TPM SHOULD support ML-DSA-44, ML-DSA-65, ***and*** ML-DSA-87. ML-DSA support is mandatory as of PTP version 1.07.

This correction resolves an inconsistency with the first sentence that indicates the TPM is required to support either ML-DSA-65 or ML-DSA-87. The second sentence as stated implies that a TPM could only support ML-DSA-44 and meet the requirements. That interpretation conflicts with the first sentence.

2.2 Errata 2 Table 3 PC Client Algorithms

For TPM_ALG_ID values not listed in Table 3, the algorithm ID can be considered Optional as defined in Section 4.3 PC Client Algorithms.

2.3 Errata 3 Table 16 Mandatory/Optional Commands

The commands TPM2_Encapsulate and TPM2_Decapsulate were added in PTP 1.07 and made mandatory. The TPM Library Specification allows Platform TPM Profiles to designate what algorithms are required for these commands. Table 16 does not indicate if support for TPM_ALG_ECC is required or not. Support for TPM_ALG_ECC is not required for these commands in PTP 1.07.

2.4 Errata 4 TPM_LOC_CTRL_x.Seize

The specification section 6.5.3.5.2.1 TPM_LOC_CTRL bitfield Seize describes the required actions a CRB TPM takes when the Seize field is set by a higher locality than the currently active locality. The current text reads:

Field: Seize

1. If the write to TPM_LOC_CTRL_x.Seize occurs from a locality of higher priority than the current locality:
    a. The TPM SHALL clear the TPM_LOC_STATE_x.activeLocality and TPM_LOC_STATE_x.locAssigned fields for any active locality of lower priority than the locality seizing the TPM.
    b. The TPM SHALL NOT change the state of the TPM_LOC_STATE_x.locAssigned field for any locality except the one writing this field.
    c. The TPM SHALL abort any command that is currently in process, as defined in Section **Error! Reference source not found. Error! Reference source not found.**.
2. If the write occurs from a locality that is equal to or lower than the current locality the TPM SHALL ignore the write.
3. A read from TPM_LOC_STATE_x.Seize SHALL return 0.

Normative 1.a and 1.b are incorrect and normative 3 contains a typographical error. The normative text should be interpreted to read:

Field: Seize

1. If the write to TPM_LOC_CTRL_x.Seize occurs from a locality of higher priority than the current locality:

a. The TPM SHALL set the TPM_LOC_STATE_x.activeLocality field to the locality value of the locality seizing the TPM.
b. The TPM SHALL abort any command that is currently in process, as defined in Section **Error! Reference source not found. Error! Reference source not found.**.

2. If the write occurs from a locality that is equal to or lower than the current locality the TPM SHALL ignore the write.
3. A read from TPM_LOC_CTRL_x.Seize SHALL return 0.

2.5 Errata 5 Table 20 – Allocation of Register Space for FIFO and CRB Access

The table contains an undefined register at offset 001Ch titled TPM_INTF_CAPABILITYX_0 in both the FIFO Register Name and CRB Register Name columns. This is an error. This address space should be marked as Reserved.

2.6 Errata 6 Section 10.2 NV Index Provisioning for Platform Firmware Supported Features

This section was inadvertently removed from PTP 1.07. The requirements contained in PTP 1.06 still apply to PTP 1.07. For the convenience of the reader, the section from PTP 1.06 is copied below.

Start of informative comment

The TCG PC Client Platform Firmware Profile version 1.06 specifies NV Extend operations to support capturing measurement of instance specific (such as a Certificate serial number or SMBIOS information containing serial numbers) information required to verify measurements extended to TPM PCRs. This section specifies the required attributes for these indices if they are prepopulated by the TPM Vendor or a PC OEM.

The TCG PC Client Platform Firmware Profile version 1.06 also defines an optional set of structures and a method using NV Indexes for event log integrity protection. There are two NV Indexes defined for this purpose: EVENT_LOG_INTEGRITY.EXIT_PM_AUTH and EVENT_LOG_INTEGRITY.READY_TO_BOOT. If the platform firmware implements this feature, both indexes need to be created in a secure manufacturing environment with the attributes specified below.

Note: The NV_Extend indexes should be provisioned in platform manufacturing. If the platform firmware is updated post manufacturing to support additional Hash algorithms or the Platform Firmware Setup menu allows alteration of the allocated PCR banks, the NV_Extend indexes might need to be reprovisioned by the platform manufacturer to reflect the revised Hash algorithms.

End of informative comment

The NV Index handles defined in Table 1 — PFP Defined NV Extend Indexes are for provisioning Platform Firmware Supported Features.

Table 1 — PFP Defined NV Extend Indexes

NV Index	Name	Purpose
0x01C40200	NV_EXTEND_INDEX_FOR_INSTANCE	NV Extend Record for instance data
0x01C40201	NV_EXTEND_INDEX_FOR_DYNAMIC	NV Extend Record for dynamic data
0x01C40202	EVENT_LOG_INTEGRITY.EXIT_PM_AUTH	Event Log Integrity for ExitPmAuth
0x01C40203	EVENT_LOG_INTEGRITY.READY_TO_BOOT	Event Log Integrity for ReadyToBoot

1. The NV_EXTEND Indexes (0x01C40200 and 0x01C40201) and the NV Indexes for Event Log Integrity SHALL have the following attributes SET: TPMA_NV_PPWRITE, TPMA_NV_PPREAD, TPMA_NV_OWNERREAD, TPMA_NV_AUTHREAD, TPMA_NV_POLICYREAD, TPMA_NV_NO_DA, and TPMA_NV_PLATFORMCREATE.
2. The NV_EXTEND Indexes (0x01C40200 and 0x01C40201) SHALL have the following additional attributes SET: TPMA_NV_ORDERLY and TPMA_NV_CLEAR_STCLEAR and SHALL have TPM_NT_EXTEND
3. The NV Indexes for Event Log Integrity (0x01C40202 and 0x01C40203) SHALL be an Ordinary Index.
4. The NV_EXTEND Indexes SHALL be created with the Index nameAlg corresponding to the strongest algorithm (bit strength) among the allocated PCR banks.
5. The NV Indexes for Event Log Integrity SHALL be created with a Public DataSize = a 2-Byte Version + 2(Size of NV Index nameAlg). Note: This size is intended to accommodate the EVENT_LOG_INTEGRITY_DATA structure defined in the TCG PC Client Platform Firmware Profile version 1.06 and later.

2.7 Errata 1 Table 3 Algorithm RSAEP / RSADP

The TPM Library Specification permits RSAEP / RSADP, i.e., RSA encryption/decryption without any padding scheme, by allowing the caller to select the TPM_ALG_RSA where the scheme parameter is set to TPM_ALG_NULL. RSAEP and RSADP have been deprecated by NIST, but the PTP is silent on the use of RSAEP / RSADP; RSAEP and RSADP are optional.

3 Clarifications

3.1 Clarification 1 Table 3 PC Client Algorithms

The description for the truncated SHAKE algorithms is inaccurate.

The current description states:

TPM_ALG_SHAKE_256_192	O	This is an extendible output function
TPM_ALG_SHAKE_256_256	O	This is an extendible output function
TPM_ALG_SHAKE_256_512	O	This is an extendible output function

The correct description is:

TPM_ALG_SHAKE_256_192	O	This is a hash function
TPM_ALG_SHAKE_256_256	O	This is a hash function
TPM_ALG_SHAKE_256_512	O	This is a hash function

3.2 Clarification 2 Table 5 Mandatory/Optional Commands

The description for TPM2_SignDigest includes a statement that a new capability allows a caller to discover which parameters and options the TPM supports for this command, such as support for external mu. This statement could be interpreted to mean that support for external mu is mandatory. Support for external mu is optional.

