TCG Storage Interface Interactions Specification (SIIS)

Specification Version 1.05
Revision 1.00
March 16, 2016
Final

Contact: admin@trustedcomputinggroup.org

TCG

TCG PUBLISHED

Disclaimers, Notices, and License Terms

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

Without limitation, TCG disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification and to the implementation of this specification, and TCG disclaims all liability for cost of procurement of substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, direct, indirect, or special damages, whether under contract, tort, warranty or otherwise, arising in any way out of use or reliance upon this specification or any information herein.

Contact the Trusted Computing Group at www.trustedcomputinggroup.org for information on specification licensing through membership agreements.

Any marks and brands contained herein are the property of their respective owners.

Table of Contents

1 Introduction 1

1.1 Document Purpose 1

1.2 Scope 1

1.3 Intended Audience 1

1.4 References to Other Documents 1
1.4.1 Approved References 1
1.4.2 References under development 2

1.5 Definition of Terms 2

2 Overview 3

3 SCSI Interface 4

3.1 Mapping of Resets 4

3.2 Mapping of IF-SEND and IF-RECV 9
3.2.1 IF_SEND 9
3.2.2 IF_RECV 9

3.3 Handling Common TPer Errors 10

3.4 Discovery of Security Capabilities 11
3.4.1 Security Protocol 0x00 11

3.5 Miscellaneous 11
3.5.1 Queued Commands 11
3.5.2 MBR Interactions 12
3.5.3 LUN usage 12
3.5.4 Interaction of Opal SSC with the SANITIZE command 12
3.5.5 Interaction of Enterprise SSC with the SANITIZE command 13
3.5.6 Special Locking SP command interactions 13

4 ATA Interface 14

4.1 Mapping of Resets 14

4.2 Mapping of IF-SEND and IF-RECV 15
4.2.1 IF_SEND 15

4.2.2 IF_RECV 15

4.3 Handling Common TPer Errors 16

4.4 Discovery of Security Capabilities 17
4.4.1 IDENTIFY DEVICE 17
4.4.2 Security Protocol 0x00 17

4.5 Miscellaneous 17
4.5.1 Feature set interactions 17
4.5.1.1 Trusted Computing feature set 17
4.5.1.2 Sense Data Reporting feature set 18
4.5.1.3 Locking Template interactions with the ATA Security feature set 18
4.5.1.4 Interaction of Opal SSC with the ATA Sanitize Device feature set 18
4.5.1.5 Interaction of Enterprise SSC with the ATA Sanitize Device feature set .. 19
4.5.1.6 Interaction of the Opal SSC Activate method with the ATA Security feature set 19
4.5.2 Special Locking SP command interactions 19

5 NVM Express Interface 20

5.1 Mapping of Resets 20

5.2 Mapping of IF-SEND and IF-RECV 21
5.2.1 IF_SEND 21
5.2.2 IF_RECV 21

5.3 Handling Common TPer Errors 22

5.4 Discovery of Security Capabilities 22
5.4.1 Identify Controller Data Structure 22
5.4.2 Security Protocol 0x00 22

5.5 Miscellaneous 23
5.5.1 Namespaces 23
5.5.1.1 Overview 23
5.5.1.2 Single Namespace 23
5.5.1.3 Multiple Namespaces 23
5.5.2 Locking Template interactions with the Format NVM Command 24

6 *e*•MMC Interface 25

6.1 Mapping of Resets 25

6.2 Mapping of IF-SEND and IF-RECV 25
6.2.1 IF_SEND 25
6.2.2 IF_RECV 26

6.2.3 *e*•MMC Command Structure for TCG IF_SEND and IF_RECV 26
6.2.3.1 *e*•MMC Block Allocation Overview 26
6.2.3.2 *e*•MMC CMD23 SET_BLOCK_COUNT command 26
6.2.3.3 *e*•MMC CMD54 PROTOCOL_WR and CMD53 PROTOCOL_RD commands 27

6.3 Handling Common TPer Errors 28

6.4 Discovery of Security Capabilities 28
6.4.1 Discovery of Security Capabilities 28
6.4.1.1 Security Protocol Information 28

6.5 Miscellaneous 29
6.5.1 Partition Management 29

1 Introduction

1.1 Document Purpose

The TCG Storage specifications are intended to provide a comprehensive command architecture for putting Storage Devices under policy control as determined by the trusted platform host, the capabilities of the storage device to conform with the policies of the trusted platform, and the lifecycle state of the Storage Device as a trusted peripheral (TPer). This document also serves as a specification for TPers if that is deemed appropriate.

This document provides the essential mapping between concepts and features of the TCG Storage Architecture Core Specification, and several host/device interfaces.

1.2 Scope

The scope of this document is the interaction between the TPer and interface commands and transports. The command interfaces described are ATA and SCSI. SCSI transports described are SAS, FC, and ATAPI. This document is written from the perspective of the Storage Device, not the host.

1.3 Intended Audience

The intended audience for this document is Storage Device and peripheral device manufacturers and developers that wish to tie Storage Devices and peripherals into trusted platforms.

1.4 References to Other Documents

1.4.1 Approved References

[1]. IETF RFC 2119, 1997, “Key words for use in RFCs to Indicate Requirement Levels”

[2]. INCITS 447-2008, "Information technology - SCSI Architecture Model - 4 (SAM-4)". Available from http://webstore.ansi.org/

[3]. INCITS 482-2012, "Information technology - ATA/ATAPI Command Set - 2 (ACS-2)". Available from http://webstore.ansi.org/

[4]. INCITS 451-2008, "Information technology - AT Attachment – 8 ATA/ATAPI Architecture Model (ATA8-AAM)". Available from http://webstore.ansi.org/

[5]. INCITS 481-2011, "Information technology - Fibre Channel Protocol for SCSI, Fourth Version (FCP-4)". Available from http://webstore.ansi.org/

[6]. INCITS 417-2006, "Information technology - Serial Attached SCSI - 1.1 (SAS-1.1). Available from http://webstore.ansi.org/

[7]. INCITS 471-2010, Information technology - USB Attached SCSI (UAS), March 9, 2010. Available from http://webstore.ansi.org/

[8]. Universal Serial Bus Mass Storage Class USB Attached SCSI Protocol (UASP), Revision 1.0, June 24, 2009. Available from http://www.usb.org.

[9]. Universal Serial Bus Mass Storage Class Bulk-Only Transport (USBBOT), Revision 1.0, September 31, 1999. Available from http://www.usb.org.

[10]. NVM Express Specification version 1.2, November 3, 2014. Available from http://www.nvmexpress.org/

[11]. JESD84-B50 *e*•MMC Specification version 5.0. Available from http://www.jedec.org/

[12]. JESD220B UFS Specification version 2.0. Available from http://www.jedec.org/

[13]. PCI Express® Base Specification Revision 3.0. Available from http://www.pcisig.com/

[14]. NVM Express Technical Errata 005, June 3, 2015. Available from http://www.nvmexpress.org/

1.4.2 References under development

[15]. [INCITS T10/1731-D], "Information technology - SCSI Primary Commands - 4 (SPC-4)". Available from http://t10.org/

[16]. [INCITS T10/1799-D], "Information technology - SCSI Block Commands - 3 (SBC-3)". Available from http://t10.org/

[17]. *e*•MMC Security Extension version 1.0 Available from http://www.jedec.org/

[18]. UFS Security Extension version 1.0 Available from http://www.jedec.org/

1.5 Definition of Terms

Term	Definition
IF-RECV	An interface command used to retrieve security protocol data from the TPer.
IF-SEND	An interface command used to transmit security protocol data to the TPer.
Locking SP	A security provider that incorporates the Locking Template as described in the Core Spec.
SSC	Security Subsystem Class. SSC specifications describe profiled sets of TCG functionality
TCG Reset	A high-level reset type defined in the Core Spec.
TPer	The TCG security subsystem within a Storage Device.
Trusted Peripheral	A TPer.

2 Overview

This document defines for each interface:

- Mapping of interface events to TCG resets
- Mapping of IF-SEND, IF-RECV
- Handling of common TPer errors
- Discovery of security capabilities
- Miscellaneous Items

3 SCSI Interface

See [2], [15], [16], [5] and [6] for details on SCSI architecture, commands and transports.

See [3] for details on ATAPI commands.

See [7], [8] and [9] for details on UAS and USB.

See [12] and [18] for details on UFS.

3.1 Mapping of Resets

Table 1 - SAS Resets Mapped to TCG reset_type

SAS Event	**Maps to TCG reset_type**
Power on reset	Power cycle
I-T Nexus Loss	(none)
Task Management-Abort Task	(none)
Task Management-Abort Task Set	(none)
Task Management-Clear Task Set	(none)
Task Management-Clear ACA	(none)
Task Management-I-T Nexus reset	(none)
Task Management-LUN Reset	Hardware Reset
Link Reset Sequence	(none)
Link reset sequence with hard reset	Hardware Reset

Table 2 - Fibre Channel Resets Mapped to TCG reset_type

FC Event	**Maps to TCG** reset_type	**Other Comments**
Power on reset	Power cycle	
I-T Nexus Loss	(none)	
Task Management-Abort Task	(none)	
Task Management-Abort Task Set	(none)	
Task Management-Clear Task Set	(none)	
Task Management-Clear ACA	(none)	
Task Management-I-T Nexus reset	(none)	
Task Management-LUN Reset	Hardware Reset	
Task Management-Target reset	Hardware Reset	
LIP(AL_PD,AL_PS)	Hardware Reset	LIP directed reset
LIP(FF,AL_PS)	Hardware Reset	LIP Global reset
Port Login	(none)	
Process Login	(none)	

Table 3 - ATAPI Resets Mapped to TCG reset_type

ATAPI Event	**Maps to TCG** reset**_**type
Power on reset	Power cycle
Hardware reset	PATA: Hardware Reset SATA: If Software Settings Preservation is enabled, then COMRESET is not a TCG Hardware Reset. If Software Settings Preservation is disabled, then COMRESET is a TCG Hardware Reset.
Software reset	(none)
DEVICE RESET command	(none)

Table 4 - UAS Events Mapped to TCG reset_type

Event	Maps to TCG reset_type	Reference
Device Power Cycle	Power cycle	[9]
Task Management-Abort Task	(none)	[15]
Task Management-Abort Task Set	(none)	[15]
Task Management-Clear Task Set	(none)	[15]
Task Management-Clear ACA	(none)	[15]
Task Management-I-T Nexus reset	Hardware Reset	[15]
Task Management-LUN Reset	(none)	[15]
USB VBus Power Cycle	Power cycle	[9]
USB Port Reset	(none)	[9]
USB Set Configuration with wValue set to zero	(none)	[9]
USB Set Configuration with wValue set to non-zero value that is not equal to the current value of bConfiguration.	(none)	[9]
USB Set Configuration with wValue set to non-zero value that is equal to the current value of bConfiguration.	(none)	[9]
USB Bulk-Out Endpoint Reset (Also known as Clear Feature, Endpoint Halt of the first Bulk-Out pipe of the Mass Storage Interface)	(none)	[9]
USB Bulk-In Endpoint Reset (Also known as Clear Feature, Endpoint Halt of the first Bulk-In pipe of the Mass Storage Interface)	(none)	[9]
USB Suspend	Hardware Reset	[9]
USB Resume	Hardware Reset	[9]

Table 5 - USB Events Mapped to TCG reset_type

Event	Maps to TCG reset_type	Reference
Device Power Cycle	Power cycle	[9]
USB VBus Power Cycle	Power cycle	[9]
USB Port Reset	(none)	[9]
USB Set Configuration with wValue set to zero	(none)	[9]
USB Set Configuration with wValue set to non-zero value that is not equal to the current value of bConfiguration.	(none)	[9]
USB Set Configuration with wValue set to non-zero value that is equal to the current value of bConfiguration.	(none)	[9]
USB Bulk-Out Endpoint Reset (Also known as Clear Feature, Endpoint Halt of the first Bulk-Out pipe of the Mass Storage Interface)	(none)	[9]
USB Bulk-In Endpoint Reset (Also known as Clear Feature, Endpoint Halt of the first Bulk-In pipe of the Mass Storage Interface)	(none)	[9]
USB Interface Reset (Also known as the BBB Bulk Only Mass Storage Reset Request x 21 FF with wIndex addressing the bInterfaceNumber of the Mass Storage Interface)	(none)	[9]
USB Suspend	Hardware Reset	[9]
USB Resume	Hardware Reset	[9]

Table 6 - UFS Events Mapped to TCG reset_type

UFS Event	Maps to TCG reset_type	Reference
Power-on	Power cycle	[12]
HW Pin Reset	Hardware Reset	[12]
EndPoint Reset	Hardware Reset	[12]
Task Management-Abort Task	(none)	[15]
Task Management-AbortTask Set	(none)	[15]
Task Management-Clear Task Set	(none)	[15]
Task Management-LUN Reset	(none)	[15]
Host System UniPro Reset	Hardware Reset	[12]

3.2 Mapping of IF-SEND and IF-RECV

3.2.1 IF_SEND

IF_SEND SHALL be implemented with the SECURITY PROTOCOL OUT [15] command, with additional requirements on the CDB as specified in Table 7.

Table 7 - IF-SEND CDB field contents (SCSI)

SECURITY PROTOCOL	SECURITY PROTOCOL SPECIFIC	INC_512	TRANSFER LENGTH
`0x00`	Security Protocol `0x00` is not defined for IF-SEND		
`0x01`	a ComID	1[a]	Non-zero[b] number of 512-byte data units.
`0x02`	a ComID	1[a]	Non-zero[b] number of 512-byte data units.
`0x06`	a ComID	0	Number of bytes of data.

[a] If the INC_512 parameter in the CDB is zero, then the TPer SHALL report Other Invalid Command Parameter (see 3.3).

[b] If the TRANSFER LENGTH parameter in the CDB is zero, then the TPer SHALL report Other Invalid Command Parameter (see 3.3).

3.2.2 IF_RECV

IF_RECV SHALL be implemented with the SECURITY PROTOCOL IN [15] command, with additional requirements on the CDB as described in Table 8.

Table 8 - IF-RECV CDB field contents (SCSI)

SECURITY PROTOCOL	SECURITY PROTOCOL SPECIFIC	INC_512	ALLOCATION LENGTH
`0x00`	(See [15] for details)	0 or 1	INC_512=0: Number of bytes of data. INC_512=1: Number of 512-byte data units.
`0x01`	a ComID	1[a]	Non-zero[b] number of 512-byte data units.
`0x02`	a ComID	1[a]	Non-zero[b] number of 512-byte data units.
`0x06`	a ComID	0	Number of bytes of data.

[a] If the INC_512 parameter in the CDB is zero, then the TPer SHALL report Other Invalid Command Parameter (see 3.3).

[b] If the ALLOCATION LENGTH parameter in the CDB is zero, then the TPer SHALL report Other Invalid Command Parameter (see 3.3), even though SPC-4 allows ALLOCATION LENGTH to be zero.

3.3 Handling Common TPer Errors

There are some common errors detected by the TPer. This section describes how they are reported via the SCSI interface.

Table 9 - TPer Errors (SCSI)

TPer Error ID	Status	Sense Key	ASC/ASCQ	Comments
Good	GOOD	NO SENSE	NO ADDITIONAL SENSE INFORMATION	Normal command completion
Invalid Security Protocol ID parameter	CHECK CONDITION	ILLEGAL REQUEST	INVALID FIELD IN CDB	No data SHALL be transferred
Invalid Transfer Length parameter on IF-SEND	CHECK CONDITION	ILLEGAL REQUEST	INVALID FIELD IN CDB	No data SHALL be transferred.
Other Invalid Command Parameter	CHECK CONDITION	ILLEGAL REQUEST	INVALID FIELD IN CDB	No data SHALL be transferred.
Synchronous Protocol Violation	CHECK CONDITION	ILLEGAL REQUEST	COMMAND SEQUENCE ERROR	No data SHALL be transferred.
Data Protection Error	CHECK CONDITION	DATA PROTECT	ACCESS DENIED– NO ACCESS RIGHTS	No data SHALL be transferred.

3.4 Discovery of Security Capabilities

3.4.1 Security Protocol 0x00

See the description of SECURITY PROTOCOL IN [15] for information on Security Protocol `0x00`.

3.5 Miscellaneous

3.5.1 Queued Commands

The TPer requires that for a given ComID the order of the IF-SEND and IF-RECV command completion be the same as the order that the host application sent the commands.

Some transport protocols MAY NOT guarantee ordering of delivery or ordering of IF-SEND and IF-RECV command completion. Therefore, the host application communicating with the TPer SHOULD ensure that a prior IF-SEND or IF-RECV has completed prior to issuing another, or use mechanisms in the interface protocol to ensure ordering (e.g. ORDERED Task Attribute for SCSI Transport protocols).

Begin Informative Content

The following definition of synchronous behavior does not affect the queuing behavior (if any) of the device interface. On queuing devices, synchronicity is enforced at the time IF-SEND/RECV commands are dequeued for processing by the drive. For non-queuing devices, synchronicity is enforced at the time the IF-SEND/RECV is initially received by the device. If queuing behavior is supported, the host should use Ordered Queuing for IF-SEND/RECV commands or indeterminate behavior may result.

It is assumed that the drive can only process one IF-SEND/RECV interface command at a time.

End Informative Content

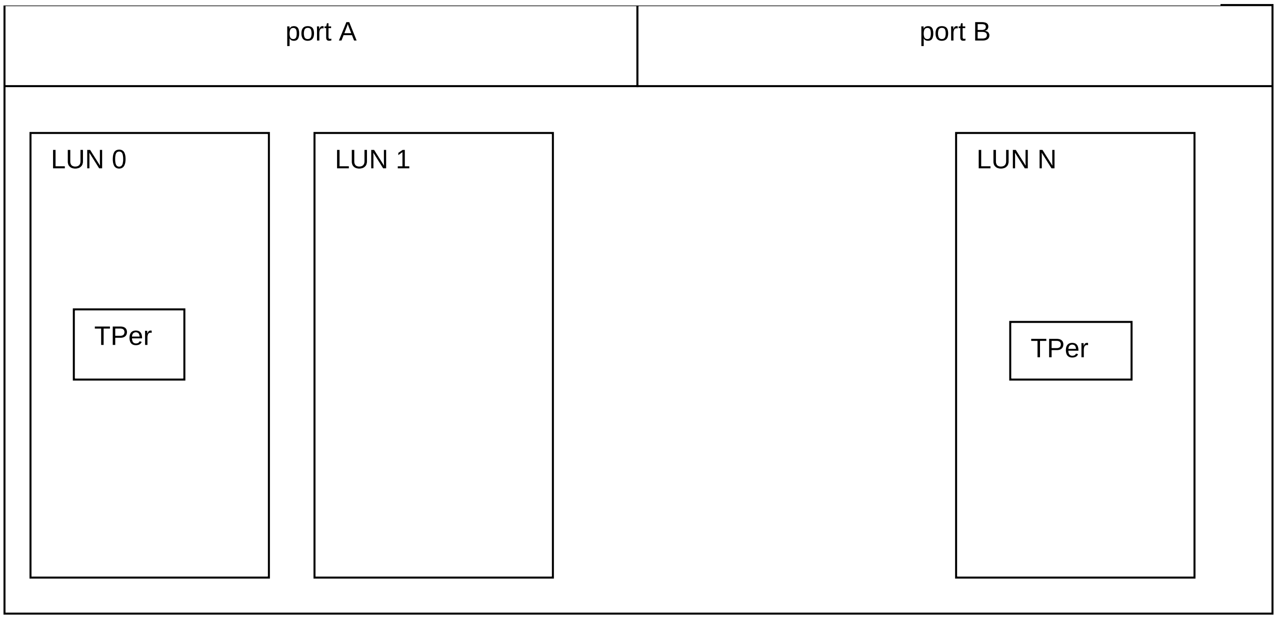
3.5.2 MBR Interactions

The LUN associated with the MBR is the boot LUN.

3.5.3 LUN usage

A target that has multiple LUNs MAY have multiple TPers. Each TPer SHALL be associated with a different LUN. Every LUN on a device is not required to have a TPer, but LUNs that support the TCG Core specification commands and functionality SHALL have a TPer. A TPer SHALL only be associated with exactly one LUN. A LUN MAY have no TPer.

Figure 1 - SCSI target: port, LUN and TPer relationships



3.5.4 Interaction of Opal SSC with the SANITIZE command

The Storage Device MAY support (i.e., REPORT SUPPORTED OPERATION CODES command) SANITIZE commands when no SP exists that incorporates the Locking Template or when an SP that incorporates the Locking Template is in the Manufactured-Inactive state.

In all other cases, the Storage Device SHALL:

a) report that SANITIZE commands are not supported (e.g., response to REPORT SUPPORTED OPERATION CODES command, and terminate SANITIZE commands); or

b) perform the following:
   a. report that SANITIZE commands are supported; and
   b. terminate SANITIZE commands with a Data Protection Error (see 3.3).

3.5.5 Interaction of Enterprise SSC with the SANITIZE command

If:

a) the EraseMaster C_PIN credential is not equal to MSID;

b) any Bandmaster C_PIN credential is not equal to MSID; or

c) for any Locking object:

   a. the value of the WriteLockEnabled column is TRUE;

   b. the value of the ReadLockedEnabled column is TRUE;

   c. the value of the RangeStart column is not equal to zero; or

   d. the value of the RangeLength column is not equal to zero,

then the Storage Device SHALL terminate a SANITIZE command with a Data Protection Error (see 3.3).

A successful SANITIZE command SHALL eradicate all Locking SP media encryption keys and generate new media encryption keys.

3.5.6 Special Locking SP command interactions

For an SD implementing the Opal SSC or the Enterprise SSC, the SD SHALL terminate the following commands with a Status of CHECK CONDITION, sense key set to ILLEGAL REQUEST and additional sense code set to INVALID COMMAND OPERATION CODE:

a) READ LONG(10);
b) READ LONG(16);
c) WRITE LONG(10), (WR_UNCOR = 0); and
d) WRITE LONG(16), (WR_UNCOR = 0).

4 ATA Interface

See [3] and [4] for details on ATA architecture, commands and transports.

4.1 Mapping of Resets

Table 10 - ATA Resets Mapped to TCG reset_type

ATA Event	**Maps to TCG** reset_type
Power on reset	Power Cycle
Software reset	(none)
Hardware reset	PATA: Hardware Reset SATA: If Software Settings Preservation is enabled, then COMRESET is not a TCG Hardware Reset. If Software Settings Preservation is disabled, then COMRESET is a TCG Hardware Reset.

4.2 Mapping of IF-SEND and IF-RECV

4.2.1 IF_SEND

IF_SEND SHALL be implemented with either the TRUSTED SEND or TRUSTED SEND DMA commands, with additional requirements on the inputs as described in Table 11:

Table 11 - IF-SEND command parameters (ATA)

Security Protocol	SP_Specific	Transfer Length
0x00	Security Protocol 0x00 is not defined for IF-SEND	
0x01	a ComID	Non-zero [a] number of 512-byte data units.
0x02	a ComID	Non-zero [a] number of 512-byte data units.
0x06	Protocol 0x06 is defined for SCSI only.	
[a] If the Transfer Length parameter is zero, then the TPer SHALL report Other Invalid Command Parameter (see 4.3).		

4.2.2 IF_RECV

IF_RECV SHALL be implemented with either the TRUSTED RECEIVE or TRUSTED RECEIVE DMA commands, with additional requirements on the inputs as described in Table 12:

Table 12 - IF-RECV command parameters (ATA)

Security Protocol	SP_Specific	Transfer Length
0x00	(See [3])	Non-zero number of 512-byte data units.
0x01	a ComID	Non-zero [a] number of 512-byte data units.
0x02	a ComID	Non-zero [a] number of 512-byte data units.
0x06	Protocol 0x06 is defined for SCSI only.	
[a] If the Transfer Length parameter is zero, then the TPer SHALL report Other Invalid Command Parameter (see 4.3).		

4.3 Handling Common TPer Errors

There are some common errors detected by the TPer. This section describes how they are reported via the ATA interface.

See [3] for information about the Sense Data Reporting (SDR) feature set and the SENSE DATA AVAILABLE (SDA) (i.e., ATA STATUS field bit 1) bit.

Table 13 describes common TPer errors if:

a) SDR is not supported;
a) SDR is supported and SDR is disabled; or
b) SDR is supported and SDR is enabled and SENSE DATA AVAILABLE is cleared to zero.

Table 14 describes common TPer errors if:

a) SDR is supported and SDR is enabled and SENSE DATA AVAILABLE is set to one.

Table 13 - TPer Errors (ATA) – Without Sense Data Reporting (SDA=0)

TPer Error ID	ATA Status Field	ATA Error Field	Comments
Good	0x50	0x00	Normal command completion
Invalid Security Protocol ID parameter	0x51	0x04	No data SHALL be transferred
Invalid Transfer Length parameter on IF-SEND	0x51	0x04	No data SHALL be transferred.
Other Invalid Command Parameter	0x51	0x04	No data SHALL be transferred.
Synchronous Protocol Violation	0x51	0x04	No data SHALL be transferred.
Data Protection Error	0x51	0x04	No data SHALL be transferred.

Table 14 - TPer Errors (ATA) – With Sense Data Reporting (SDA=1)

TPer Error ID	ATA Status Field Bit 1	Sense Key	ASC/ASCQ	Comments
Good	1	NO SENSE	NO ADDITIONAL SENSE	Normal command completion
Invalid Security Protocol ID parameter	1	ILLEGAL REQUEST	INVALID FIELD IN CDB	No data SHALL be transferred
Invalid Transfer Length parameter on IF-SEND	1	ILLEGAL REQUEST	INVALID FIELD IN CDB	No data SHALL be transferred.
Other Invalid Command Parameter	1	ILLEGAL REQUEST	INVALID FIELD IN CDB	No data SHALL be transferred.
Synchronous Protocol Violation	1	ILLEGAL REQUEST	COMMAND SEQUENCE ERROR	No data SHALL be transferred.
Data Protection Error	1	DATA PROTECT	ACCESS DENIED– NO ACCESS RIGHTS	No data SHALL be transferred.

4.4 Discovery of Security Capabilities

4.4.1 IDENTIFY DEVICE

The IDENTIFY DEVICE command (see [3]) indicates whether the device has support for the ATA Security feature set or the Trusted Computing feature set. See IDENTIFY DEVICE data words 48, 82, and 128 for further information.

4.4.2 Security Protocol 0x00

The TRUSTED RECEIVE command (see [3]) describes Security Protocol `0x00`.

4.5 Miscellaneous

4.5.1 Feature set interactions

4.5.1.1 Trusted Computing feature set

The Trusted Computing feature set SHALL be supported by the device.

4.5.1.2 Sense Data Reporting feature set

If the Sense Data Reporting (SDR) feature set is supported and enabled, then common TPer errors are reported as Sense Codes instead of as regular ATA errors. (See [3] and 4.3).

4.5.1.3 Locking Template interactions with the ATA Security feature set

If the lifecycle state of the Locking SP changes from the Manufactured-Inactive state to the Manufactured state, then:

1) the TPer SHALL save the current value of:
    a. IDENTIFY DEVICE, word 82, bit 1;
    b. IDENTIFY DEVICE, word 85, bit 1; and
    c. IDENTIFY DEVICE, word 128;

   and
2) the TPer SHALL change the value of IDENTIFY DEVICE, word 82, bit 1 to zero.

If the lifecycle state of the Locking SP is in the Manufactured state, then IDENTIFY DEVICE commands processed by the device SHALL indicate that the ATA Security feature set is not supported.

If the lifecycle state of the Locking SP changes from the Manufactured state to the Manufactured-Inactive state, then the TPer SHALL restore the value of the IDENTIFY DEVICE data to the values that were saved when the TPer changed the state from Manufactured-Inactive to Manufactured:

a) IDENTIFY DEVICE, word 82, bit 1;
b) IDENTIFY DEVICE, word 85, bit 1; and
c) IDENTIFY DEVICE, word 128.

If there is no Locking SP or the lifecycle state of the Locking SP is in the Manufactured-Inactive state, IDENTIFY DEVICE commands processed by the device MAY indicate that the ATA Security feature set is supported.

When ATA Security is Enabled (a User Password is set), the TPer SHALL prohibit issuance of an SP that incorporates the Locking Template, and SHALL prohibit a SP that incorporates the Locking Template from transitioning out of the Manufactured-Inactive state.

4.5.1.4 Interaction of Opal SSC with the ATA Sanitize Device feature set

The Storage Device MAY support (i.e., IDENTIFY DEVICE, word 59, bit 12 = 1) the ATA Sanitize Device feature set when no SP exists that incorporates the Locking Template or when an SP that incorporates the Locking Template is in the Manufactured-Inactive state.

In all other cases, the Storage Device SHALL:

a) report that the ATA Sanitize Device feature set is not supported (i.e., IDENTIFY DEVICE, word 59, bit 12 = 0); or
b) perform the following:
    a. report that the ATA Sanitize Device feature set is supported (i.e., IDENTIFY DEVICE word 59, bit 12 = 1); and
    b. terminate the following commands with a Data Protection Error (see 4.3):
        i. CRYPTO SCRAMBLE EXT command;
        ii. OVERWRITE EXT command;
        iii. BLOCK ERASE EXT command;
        iv. SANITIZE ANTIFREEZE LOCK EXT command; and
        v. SANITIZE FREEZE LOCK EXT command.

4.5.1.5 Interaction of Enterprise SSC with the ATA Sanitize Device feature set

If:

a) the EraseMaster C_PIN credential is not equal to MSID;

b) any Bandmaster C_PIN credential is not equal to MSID; or

c) for any Locking object:

   a. the value of the WriteLockEnabled column is TRUE;

   b. the value of the ReadLockedEnabled column is TRUE;

   c. the value of the RangeStart column is not equal to zero; or

   d. the value of the RangeLength column is not equal to zero,

then the Storage Device SHALL terminate the following commands with a Data Protection Error (see 4.3):

a) CRYPTO SCRAMBLE EXT command;
b) OVERWRITE EXT command;
c) BLOCK ERASE EXT command;
d) SANITIZE ANTIFREEZE LOCK EXT command; and
e) SANITIZE FREEZE LOCK EXT command,

A successful SANITIZE command SHALL eradicate all Locking SP media encryption keys and generate new media encryption keys.

4.5.1.6 Interaction of the Opal SSC Activate method with the ATA Security feature set

If the Activate method is invoked on the Locking SP while ATA Security is Enabled (i.e., a User Password is set), the method invocation SHALL fail with a status of ACTIVATE FAILED.

4.5.2 Special Locking SP command interactions

If:

a) an SD implements the Opal SSC or the Enterprise SSC; and
b) the Sense Data Reporting feature is supported and is enabled,

then the SD SHALL terminate the following ATA commands with the Sense Key set to ILLEGAL REQUEST and the additional sense set to INVALID COMMAND OPERATION CODE:

a) READ LONG;
b) WRITE LONG;
c) SCT READ LONG; and
d) SCT WRITE LONG.

If:

a) an SD implements the Opal SSC or the Enterprise SSC; and
b) the Sense Data Reporting feature is not supported or is not enabled,

then the SD SHALL return command aborted for the following ATA commands:

a) READ LONG;
b) WRITE LONG;
c) SCT READ LONG; and
d) SCT WRITE LONG.

5 NVM Express Interface

See [10] for details on NVM Express architecture, commands and transports.

5.1 Mapping of Resets

Table 15 – NVM Express Resets Mapped to TCG reset_type

NVM Express Event	**Maps to TCG** reset_type	**Reference**
Power (power-up)	Power Cycle	[10]
PCIe hot reset	None	[13]
PCIe warm reset	None	[13]
PCIe cold reset	None	[13]
PCIe transaction layer Data Link Down status	None	[13]
NVMe subsystem reset	None	[10]
NVMe Controller reset (CC.EN transitions from 1 to 0)	None	[10]
NVMe Function level (PCI) reset	None	[10]
NVMe Queue level reset	None	[10]

5.2 Mapping of IF-SEND and IF-RECV

5.2.1 IF_SEND

IF_SEND SHALL be implemented with the Security Send command [10], with additional requirements on the inputs as described in Table 16:

Table 16 - IF-SEND command parameters (NVM Express)

Security Protocol	SP_Specific	Transfer Length	Namespace Identifier [10]
0x00	Security Protocol 0x00 is not defined for IF-SEND		Reserved
0x01	a ComID	Number of bytes to transfer.	Reserved
0x02	a ComID	Number of bytes to transfer.	Reserved
0x06	Protocol 0x06 is defined for SCSI only.		Reserved

5.2.2 IF_RECV

IF_RECV SHALL be implemented with the Security Receive command [10], with additional requirements on the inputs as described in Table 17:

Table 17 - IF-RECV command parameters (NVM Express)

Security Protocol	SP_Specific	Allocation Length	Namespace Identifier [10]
0x00	(See [10])	Number of bytes to transfer.	Reserved
0x01	a ComID	Number of bytes to transfer.	Reserved
0x02	a ComID	Number of bytes to transfer.	Reserved
0x06	Protocol 0x06 is defined for SCSI only.		Reserved

5.3 Handling Common TPer Errors

There are some common errors detected by the TPer. This section describes how they are reported via the NVM Express interface.

Common Tper errors are reported in the NVM Express Admin Completion Queue, Status Field (see [10]). The Status Code Type (SCT) field and the Status Code (SC) field SHALL indicate and map the TPer error as in Table 18.

Table 18 - TPer Errors (NVM Express)

TPer Error ID	Status Code Type	Status Code	Comments
Good	Generic Command Status	Successful Completion	Normal command completion
Invalid Security Protocol ID parameter	Generic Command Status	Invalid Field in Command	No data SHALL be transferred.
Invalid Transfer Length parameter on IF-SEND	Generic Command Status	Invalid Field in Command	No data SHALL be transferred.
Other Invalid Command Parameter	Generic Command Status	Invalid Field in Command	No data SHALL be transferred.
Synchronous Protocol Violation	Generic Command Status	Command Sequence Error	No data SHALL be transferred.
Data Protection Error	Media and Data Integrity Errors	Access Denied	No data SHALL be transferred.
Invalid Security State	Command Specific Status	Invalid Format	No data SHALL be transferred.
Access Denied	Generic Command Status	0x15	No data SHALL be transferred.

5.4 Discovery of Security Capabilities

5.4.1 Identify Controller Data Structure

The Optional Admin Command Support (OACS) of the Identify Controller Data Structure (see [10]) indicates whether the device has support for the Security Send and Security Receive commands.

5.4.2 Security Protocol 0x00

The Security Receive command (see [10]) describes Security Protocol `0x00`.

5.5 Miscellaneous

5.5.1 Namespaces

5.5.1.1 Overview

An NVM subsystem SHALL have no more than one TPer. The TPer is associated with the NVM subsystem rather than with any controller within the NVM subsystem.

The requirements for namespace interactions with TCG vary depending on the number of existing namespaces (see [10]) in the NVM subsystem (see Table 19).

Table 19 - Namespace Interaction overview

Number of Existing Namespaces	Reference
0	N/A
1	5.5.1.2
Greater than 1	5.5.1.3

5.5.1.2 Single Namespace

5.5.1.2.1 Global Range Locking object Interactions

If only one namespace exists (see [14]) in the NVM subsystem, then the column values of the Global Range Locking object (e.g., ReadLocked and WriteLocked) apply to all LBAs within that namespace that are not associated with any non-Global Range Locking objects.

Successful execution of any method that results in the cryptographic erase of the Global Range Locking object SHALL result in the cryptographic erase of all LBAs within that namespace that are not associated with any non-Global Range Locking objects.

5.5.1.2.2 Non-Global Range Locking Object Interactions

If only one namespace exists in the NVM subsystem, then the device MAY support configuration of non-Global Range Locking objects.

5.5.1.2.3 Namespace Management

If only one namespace exists in the NVM subsystem, and:

a) the value of the ReadLockEnabled column of the Global Range Locking object is TRUE and the value of the ReadLocked column of the Global Range Locking object is TRUE;

b) the value of the WriteLockEnabled column of the Global Range Locking object is TRUE and the value of the WriteLocked column of the Global Range Locking object is TRUE;

c) the value of the RangeStart column of any non-Global Range Locking object is not equal to zero; or

d) the value of the RangeLength column of any non-Global Range Locking object is not equal to zero,

then execution of the Namespace Management command SHALL fail and report Access Denied.

5.5.1.3 Multiple Namespaces

5.5.1.3.1 Global Range Locking object Interactions

If more than one namespace exists (see [14]) in the NVM subsystem, then the column values of the Global Range Locking object (e.g., ReadLocked and WriteLocked) apply to all existing namespaces in the NVM subsystem.
If:

a) the value of the ReadLockEnabled column of the Global Range Locking object is TRUE; and
b) the value of the ReadLocked column of the Global Range Locking object is TRUE,

then all namespaces are read locked, and any command that reads user data or metadata (e.g., Read commands) SHALL fail and report a Data Protection Error.

If:

a) the value of the WriteLockEnabled column of the Global Range Locking object is TRUE; and
b) the value of the WriteLocked column of the Global Range Locking object is TRUE,

then all namespaces are write locked and any command that modifies user data or metadata (e.g., Write, Write Zeros, Write Uncorrectable, or Data Management - Deallocate commands) SHALL fail and report a Data Protection Error.

An NVM subsystem with more than one namespace MAY support a separate media encryption key for each namespace. In this case, the K_AES_* object referenced by the ActiveKey column value of the Global Range Locking object SHALL represent all media encryption keys in use for individual namespace encryption. Successful execution of any method that results in the cryptographic erase of the Global Range Locking object SHALL result in the cryptographic erase of all existing namespaces in the NVM subsystem.

5.5.1.3.2 Non-Global Range Locking Object Interactions

If more than one namespace exists (see [14]) in the NVM subsystem, the Global Range Locking object is the only Locking object that is configurable. Attempts to modify other Locking objects SHALL fail with status of INVALID_PARAMETER. Other operations on non-Global Range Locking objects (e.g., Get, Next) SHALL operate as indicated in the applicable SSC specification.

5.5.1.3.3 Namespace Management

If more than one namespace exists (see [14]) in the NVM subsystem, and:

a) the value of the ReadLockEnabled column of the Global Range Locking object is TRUE and the value of the ReadLocked column of the Global Range Locking object is TRUE; or
b) the value of the WriteLockEnabled column of the Global Range Locking object is TRUE and the value of the WriteLocked column of the Global Range Locking object is TRUE,

then execution of the Namespace Management command SHALL fail and report Access Denied.

5.5.1.3.4 Geometry Feature Descriptor with Multiple Namespaces

The host SHOULD ignore the Geometry Feature Descriptor.

5.5.1.3.5 LockingInfoTable with Multiple Namespaces

The host SHOULD ignore the AlignmentRequired, LogicalBlockSize, Alignment Granularity, and LowestAlignedLBA columns in the LockingInfo Table. The MaxRanges column of the LockingInfo table SHALL operate as indicated in the applicable SSC specification.

5.5.2 Locking Template interactions with the Format NVM Command

The Format NVM command MAY be supported on an NVM subsystem that contains an SP that incorporates the Locking Template.

The Format NVM command SHALL fail and report Invalid Security State if for any Locking object:

a) the value of the WriteLockEnabled column of the Locking object is TRUE; and
b) the value of the WriteLocked column of the Locking object is TRUE.

6 *e*•MMC Interface

See [11] for details on e•MMC architecture, commands and transports. In addition further details relating to the mapping provided below are found in [17].

6.1 Mapping of Resets

Table 20 specifies the *e*•MMC events that are mapped to TCG resets.

Table 20 - e•MMC Events Mapped to TCG reset_type

***e*•MMC Event**	**Maps to TCG reset_type**	**Reference**
Power On	Power cycle	[11]
H/W Reset (Pin, Reset Signal)	Hardware Reset	[11]
GO_IDLE_STATE (CMD0)	Hardware Reset	[11]
GO_PRE_IDLE_STATE (CMD0)	Hardware Reset	[11]
GO_INACTIVE_ STATE (CMD15)	Power cycle	[11]
HPI (High Priority Interrupt)	None	[11]

6.2 Mapping of IF-SEND and IF-RECV

6.2.1 IF_SEND

IF_SEND is implemented with the combination of a CMD23 (i.e., SET_BLOCK_COUNT), followed by a CMD54 (PROTOCOL_WR), with additional requirements on the inputs as described in Table 21.
CMD23 command is used to set the transfer block count for the CMD54. See [11] for details about CMD23 and CMD54.

Table 21 - IF-SEND command parameters (*e*•MMC)

Security Protocol	**SP_Specific**	**Transfer Length**
`0x00`	Security Protocol `0x00` is not defined for IF-SEND	
`0x01`	a ComID	Non-zero[1] number of 512 byte data units as defined in CMD23
`0x02`	a ComID	Non-zero[1] number of 512 byte data units as defined in CMD23
`0x06`	Protocol `0x06` is defined for SCSI only.	
[1] If the Transfer Length parameter (“number of blocks”) in CMD23 is zero or if CMD23 was not successfully received, then the e•MMC device SHALL report SEC_INVALID_COMMAND_PARAMETER (see 6.4).		

6.2.2 IF_RECV

IF_RECV is implemented with the combination of a CMD23 (SET_BLOCK_COUNT), followed by a CMD53 (PROTOCOL_RD), with additional requirements on the inputs as described in Table 22.
CMD23 command is used to set the transfer block count for the CMD53. See [11] for details about CMD23 and CMD53.

Table 22 - IF-RECV command parameters (*e*•MMC)

Security Protocol	SP_Specific	Allocation Length
`0x00`	See [11] [2]	Non-zero[1] number of 512 byte data units as defined in CMD23
`0x01`	a ComID	Non-zero[1] number of 512 byte data units as defined in CMD23
`0x02`	a ComID	Non-zero[1] number of 512 byte data units as defined in CMD23
`0x06`	Protocol `0x06` is defined for SCSI only.	
[1] If the Transfer Length parameter ("number of blocks") in CMD23 is zero or if CMD23 was not successfully received, then the e•MMC device SHALL report SEC_INVALID_COMMAND_PARAMETER (see 6.4). [2] When receiving CMD53 (PROTOCOL_RD) with Security Protocol value equal to 00h the device SHALL return the list of supported protocols.		

6.2.3 *e*•MMC Command Structure for TCG IF_SEND and IF_RECV

6.2.3.1 *e*•MMC Block Allocation Overview

The *e*•MMC protocol uses the CMD23 SET_BLOCK_COUNT command (see 6.2.3.2) to set the block count for the CMD54 command or the CMD53 command (see 6.2.3.3) that immediately follows it. The block count of the CMD54 command or the CMD53 command is specified in 512-byte blocks (i.e., Allocation Length maps to the number of blocks in the payload multiplied by 512). Payload padding to the specified number of 512 byte blocks SHALL consist of zeros.

For TCG on the *e*•MMC transport, the IF_SEND command consists of the combination of a CMD23, followed by a CMD54.

In TCG on the *e*•MMC transport, the IF_RECV command consists of the combination of a CMD23, followed by a CMD53.

6.2.3.2 *e*•MMC CMD23 SET_BLOCK_COUNT command

CMD23 SET_BLOCK_COUNT is sent before CMD54 or CMD53 to set a transfer length of one or more 512-byte block. See Table 23.

Table 23 - e•MMC CMD23 Command Block

Bit Byte	7	6	5	4	3	2	1	0
0	[47] Start Bit	[46] Transition Bit	[45:40] Command Index					

1	[39] Reliable Write Request	[38] '0' non-packed	[37] tag request	[36:33] context ID	[32]: forced programming
2	[31:24] set to 0				
3	[23:16] Number of Blocks (15:8)				
4	[15:8]: Number of Blocks (7:0)				
5	[7:1] CRC7				[0] Stop Bit

The value of Command Index is defined as 23 for this command. See [11] for more information.

The value in the Number of Blocks field specifies how many blocks are to be transferred in the next command. See [11] for more information.

All other fields are defined in [11].

6.2.3.3 *e*•MMC CMD54 PROTOCOL_WR and CMD53 PROTOCOL_RD commands

CMD54 PROTOCOL_WR and CMD53_PROTOCOL_RD commands are used to send the Security Protocol and the Security Protocol Specific parameters of the TCG IF_SEND and IF_RECV commands. See Table 24.

Table 24 - e•MMC CMD54 and CMD53 Structure

Bit Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	[47] Start Bit	[46] Transition Bit	[45:40] Command Index					
1	[39:32] Security Protocol Specific (15:8)							
2	[31:24] Security Protocol Specific (7:0)							
3	[23:16] Security Protocol							
4	[15:8] Reserved							
5	[7:1] CRC7							[0] Stop Bit

See Table 21 and Table 22 for usage of Bytes 1 and 2, the Security Protocol Specific fields in addition with the Security Protocol field.

All other fields are defined in [11].

6.3 Handling Common TPer Errors

Security related errors are detected by the *e*•MMC interface or by the TPer. This section describes how they are reported by the *e*•MMC interface.

See [11] for details.

Table 25 - TPer Errors (*e*•MMC)

TPer Error ID	***e*•MMC Device Status**	**EXCEPTION EVENTS STATUS [1]**	**EXT SECURITY ERR [2]**	**Comments**
Good	No error	No error	No error	Normal command completion
Invalid Security Protocol ID parameter	EXCEPTION EVENT=1	EXTENDED SECURITY FALURE =1	SEC INVALID COMMAND PARAMETERS =1	No data SHALL be transferred.
Invalid Transfer Length parameter on IF-SEND	EXCEPTION EVENT=1	EXTENDED SECURITY FALURE =1	SEC INVALID COMMAND PARAMETERS =1	No data SHALL be transferred.
Other Invalid Command Parameter	EXCEPTION EVENT=1	EXTENDED SECURITY FALURE =1	SEC INVALID COMMAND PARAMETERS =1	No data SHALL be transferred.
Synchronous Protocol Violation	EXCEPTION EVENT=1	EXTENDED SECURITY FALURE =1	SEC INVALID COMMAND PARAMETERS =1	No data SHALL be transferred.
Data Protection Error	EXCEPTION EVENT=1	EXTENDED SECURITY FALURE =1	ACCESS DENIED=1	No data SHALL be transferred.
[1] EXCEPTION_EVENTS_STATUS field of the EXT_CSD register [2] EXT_SECURITY_ERR field of the EXT_CSD register				

6.4 Discovery of Security Capabilities

6.4.1 Discovery of Security Capabilities

6.4.1.1 Security Protocol Information

In order to discover whether the extended protocol pass through commands are supported the host SHOULD verify that Command Class 10 is supported by the device (in CCC field in CSD Register).

In order to receive and send extended protocol information CMD53 and CMD54 SHALL be used.

Refer to Security Protocol Information (see [11]) for the discovery of which security feature set is supported.

When receiving PROTOCOL_RD (CMD53) with Security Protocol value equal to 00h the device SHALL return the list of supported protocols.

6.5 Miscellaneous

6.5.1 Partition Management

The Locking Template SHALL be associated with and manage only the User Data Area partition (see [11]).