To: **T10 Committee**
From: **Gerry Houlder, Seagate Technology,** gerry_houlder@seagate.com
Developed for **Trusted Computing Group**, www.trustedcomputinggroup.org
Subj: **SPC-4 Security Commands proposal**
Date: **Jan. 11, 2006**

This document presents a proposal for defining an industry standard set of interface commands for a trusted device, which is a component of an overall trusted system.

A trusted device provides a horizontal security product embedded in devices whose behavior may be authorized via interaction with a trusted host system.

This proposal uses two commands: SECURITY PROTOCOL OUT and SECURITY PROTOCOL IN. These commands provide for variable length data transfers. These commands are 12 byte CDBs to provide portability between SCSI and ATAPI implementations.

The CDB parameters shall be defined by T10. The data payload and subsequent actions resulting from these commands are defined by the Security Protocol identified in the CDB. The intent is to standardize this data content so it is identical across both ATA and SCSI.

2.2 Approved References

- ITU-T RECOMMENDATION X.509 | ISO/IEC 9594-8, *Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks*, ITU, 2000.
- Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization. International Standard 8824, (December, 1987).

2.4 IETF References

- RFC 3280, *Internet X.509 Public Key Infrastructure: Certificate and Certificate Revocation List (CRL) Profile*, IETF, 2002.
- RFC 3281, *An Internet Attribute Certificate: Profile for Authorization*, IETF, 2002.

3.1 Definitions

3.1.o Object Identifier (OID): An ASN.1 format (see 2.2) identifier for an object in or related to a certificate. See ISO/IEC 9834.

3.1.t Trusted Computing Group (TCG): An organization that develops and promotes open standards for hardware-enabled trusted computing and security technologies. See https://www.trustedcomputinggroup.org.

3.2 Abbreviations

3.2.a ASN.1: Abstract Syntax Notation One (see 2.2).

3.2.o OID: Object Identifier (see 3.1.o).

3.2.t TCG: Trusted Computing Group (see 3.1.t).

4.3.4.4 Transfer length

[Note: No changes are proposed – included for reference only.]

The TRANSFER LENGTH field specifies the amount of data to be transferred, usually the number of blocks. Some commands use transfer length to specify the requested number of bytes to be sent as defined in the command description.

Commands that use one byte for the TRANSFER LENGTH field may allow up to 256 blocks or 256 bytes of data to be transferred by one command.

In commands that use multiple bytes for the TRANSFER LENGTH field, a transfer length of zero specifies that no data transfer shall take place. A value of one or greater specifies the number of blocks or bytes that shall be transferred.

Refer to the specific command description for further information.

4.3.4.6 Allocation length

[Note: For this clause, changes from current wording are underlined.]

The ALLOCATION LENGTH field specifies the maximum number of bytes or blocks that an application client has allocated in the Data-In Buffer. The field specifies bytes unless defined differently by the command.

An allocation length of zero specifies that no data shall be transferred. This condition shall not be considered as an error.

The device server shall terminate transfers to the Data-In Buffer when the number of bytes or blocks specified by the ALLOCATION LENGTH field have been transferred or when all available data have been transferred, whichever is less. The allocation length is used to limit the maximum amount of variable length data (e.g., mode data, log data, diagnostic data) returned to an application client. If the information being transferred to the Data-In Buffer includes fields containing counts of the number of bytes in some or all of the data, then the contents of these fields shall not be altered to reflect the truncation, if any, that results from an insufficient ALLOCATION LENGTH value, unless the standard that describes the Data-In Buffer format states otherwise.

If the amount of information to be transferred exceeds the maximum value that the ALLOCATION LENGTH field is capable of specifying, the device server shall transfer no data and terminate the command with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN CDB.

5.6.1 Persistent Reservations overview

[In table 31, add the SECURITY PROTOCOL IN and SECURITY PROTOCOL OUT commands as indicated below.]

Command	Addressed logical unit has this type of persistent Reservation held by another I_T nexus				
	From any I_T nexus		From registered I_T nexus (RR all types)	From not registered I_T nexus	
	Write excl	Excl access		Write excl RR	Excl access - RR
SECURITY PROTOCOL IN	allowed	conflict	allowed	allowed	conflict
SECURITY PROTOCOL OUT	conflict	conflict	allowed	conflict	conflict

6.x SECURITY PROTOCOL OUT command

The SECURITY PROTOCOL OUT command (see table 1) is used to send data to the device server. The data sent contains one or more instructions to be performed by the device server. The format and function of the instructions depends on the contents of the SECURITY PROTOCOL field (see table 2). The application client uses SECURITY PROTOCOL IN command to retrieve data derived from these instructions.

Table 1 – SECURITY PROTOCOL OUT command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (B5h)							
1	SECURITY PROTOCOL							
2	SP_SPECIFIC							
3								
4	INC_512	RESERVED						
5	RESERVED							
6	(MSB)	TRANSFER LENGTH						
7								
8								
9								(LSB)
10	RESERVED							
11	CONTROL							

The SECURITY PROTOCOL field specifies which security protocol is being used (see table 2).

Table 2 – SECURITY PROTOCOL OUT SECURITY PROTOCOL field

Code	Description	Reference
00h	Reserved	
01h – 06h	Defined by the TCG	3.1.t
07h – EFh	Reserved	
F0h – FFh	Vendor specific	

The contents of the SP_SPECIFIC field depend on the protocol specified by the SECURITY PROTOCOL field (see table 2).

A 512 increment (INC_512) bit set to one specifies that the TRANSFER LENGTH field (see 4.3.4.4) expresses the number of bytes to be transferred in increments of 512 bytes (e.g., a value of one means 512 bytes, two means 1 024 bytes, etc.). Pad bytes shall be appended as needed to meet this requirement. Pad bytes shall have a value of 00h. A 512_INC bit set to zero specifies that the TRANSFER LENGTH field indicates the number of bytes to be transferred.

Any association between a SECURITY PROTOCOL OUT command and a subsequent SECURITY PROTOCOL IN command depends on the protocol specified by the SECURITY PROTOCOL field (see table 4). Each protocol shall specify whether:

a) the device server shall complete the command with GOOD status as soon as it determines the data has been correctly received. An indication that the data has been processed is obtained by sending a SECURITY PROTOCOL IN command and receiving the results in the associated data transfer; or
b) the device server shall complete the command with GOOD status only after the data has been successfully processed and an associated SECURITY PROTOCOL IN command is not required.

The format of the data depends on the protocol specified by the SECURITY PROTOCOL field (see table 2).

6.y SECURITY PROTOCOL IN command

6.y.1 SECURITY PROTOCOL IN command description

The SECURITY PROTOCOL IN command (see table 3) is used to retrieve security protocol information (see 6.y.2) or the results of one or more SECURITY PROTOCOL OUT commands (see 6.x).

Table 3 – SECURITY PROTOCOL IN command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A2h)							
1	SECURITY PROTOCOL							
2	SP_SPECIFIC							
3								
4	INC_512	RESERVED						
5	RESERVED							
6	(MSB)	ALLOCATION LENGTH						
7								
8								
9								(LSB)
10	RESERVED							
11	CONTROL							

The SECURITY PROTOCOL field specifies which security protocol is being used (see table 4).

Table 4 – SECURITY PROTOCOL IN SECURITY PROTOCOL field

Code	Description	Reference
00h	Security protocol information	6.y.2
01h – 06h	Defined by the TCG	3.1.t
07h – EFh	Reserved	
F0h - FFh	Vendor specific	

The contents of the SP_SPECIFIC field depend on the protocol specified by the SECURITY PROTOCOL field (see table 4).

A 512 increment (INC_512) bit set to one specifies that the ALLOCATION LENGTH field (see 4.3.4.6) expresses the maximum number of bytes available to receive data in increments of 512 bytes (e.g., a value of one means 512 bytes, two means 1 024 bytes, etc.). Pad bytes may or may not be appended to meet this length. Pad bytes shall have a value of 00h. An INC_512 bit set to zero specifies that the ALLOCATION LENGTH field expresses the number of bytes to be transferred.

Indications of data overrun or underrun and the mechanism, if any, for processing retries depend on the protocol specified by the SECURITY PROTOCOL field (see table 4).

Any association between a previous SECURITY PROTOCOL OUT command and the data transferred by a SECURITY PROTOCOL IN command depends on the protocol specified by the TRUSTED PROTOCOL field (see table 4). If the device server has no data to transfer (e.g., the results for any previous SECURITY PROTOCOL OUT commands are not yet available), the device server may transfer data indicating it has no other data to transfer.

The format of the data depends on the protocol specified by the SECURITY PROTOCOL field (see table 4).

The device server shall retain data resulting from a SECURITY PROTOCOL OUT command , if any, until one of the following events is processed:

a) transfer of the data via a SECURITY PROTOCOL IN command from the same I_T_L nexus as defined by the protocol specified by the SECURITY PROTOCOL field (see table 4);
b) logical unit reset; or
c) I_T nexus loss associated with the I_T nexus that sent the SECURITY PROTOCOL OUT command.

If the data is lost due to one of these events the application client may send a new SECURITY PROTOCOL OUT command to retry the instructions.

6.y.2 Security protocol information description

6.y.2.1 Overview

The purpose of SECURITY PROTOCOL of 00h is to transfer security protocol related information from the logical unit. A SECURITY PROTOCOL IN command using SECURITY PROTOCOL field set to 00h is not associated with an earlier SECURITY PROTOCOL OUT command.

If the SECURITY PROTOCOL IN command is supported, the SECURITY PROTOCOL value of 00h shall be supported as defined in this standard.

6.y.2.2 CDB description

When the SECURITY PROTOCOL field is set to 00h, the SP_SPECIFIC field contains a single numeric value as defined in 3.5 (see table 5).

Table 5 – SP_SPECIFIC field

Code	Description	Support	Reference
0000h	Supported security protocol list	Mandatory	6.y.2.3
0001h	Certificate data	Mandatory	6.y.2.4
0002h – FFFFh	Reserved		

All other CDB fields for SECURITY PROTOCOL IN command shall meet the requirements of 6.y.1.

Each time a SECURITY PROTOCOL IN command with SECURITY PROTOCOL field set to 00h is received, the device server shall transfer the data defined by this subclause starting with byte 0.

6.y.2.3 Supported security protocols list description

When the SECURITY PROTOCOL field is set to 00h and the SP_SPECIFIC field is set to 0000h in a SECURITY PROTOCOL IN command, the parameter data shall have the format shown in Table 6.

Table 6 – SECURITY PROTOCOL IN parameter data for SP_SPECIFIC 0000h

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVED							
1	RESERVED							
2	RESERVED							
3	RESERVED							
4	RESERVED							
5	RESERVED							
6	(MSB)	LIST LENGTH (m - 7)						
7								(LSB)
8	SUPPORTED SECURITY PROTOCOL LIST							
m								
m+1	PAD BYTES (optional)							
n								

The LIST LENGTH field indicates the total length, in bytes, of the supported security protocol list.

The SUPPORTED SECURITY PROTOCOL LIST field shall contain a list of all supported SECURITY PROTOCOL field values. Each byte indicates a supported SECURITY PROTOCOL field value. The values shall be in ascending order starting with 00h.

The total data length shall conform to the ALLOCATION LENGTH field requirements. Pad bytes may be appended to meet this length. Pad bytes shall have a value of 00h.

6.y.2.4 Certificate data description

6.y.2.4.1 Certificate overview

A certificate is either an X.509 Attribute Certificate (see 6.y.2.4.3) or an X.509 Public Key Certificate (see 6.y.2.4.2) depending on the capabilities of the logical unit.

When the SECURITY PROTOCOL field is set to 00h and the SP_SPECIFIC field is set to 0001h in a SECURITY PROTOCOL IN command, the parameter data shall have the format shown in Table 7.

Table 7 – SECURITY PROTOCOL IN parameter data for SP_SPECIFIC 0001h

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVED							
1	RESERVED							
2	(MSB)	CERTIFICATE LENGTH (m - 3)						
3								(LSB)
4	CERTIFICATE							
m								
m+1	PAD BYTES (optional)							
n								

The CERTIFICATE LENGTH field indicates the total length, in bytes, of the certificate. This length includes one or more certificates. If the device server doesn't have a certificate to transfer, the certificate length shall be set to 0000h.

The contents of the CERTIFICATE field are defined in 6.y.2.4.2 and 6.y.2.4.3.

The total data length shall conform to the ALLOCATION LENGTH field requirements. Pad bytes may be appended to meet this length. Pad bytes shall have a value of 00h.

6.y.2.4.2 Public Key certificate description

RFC 3280 defines the certificate syntax for certificates consistent with X.509v3 Public Key Certificate Specification. Any further restrictions beyond the requirements of RFC 3280 are TBD.

6.y.2.4.3 Attribute certificate description

RFC 3281 defines the certificate syntax for certificates consistent with X.509v2 Attribute Certificate Specification. Any further restrictions beyond the requirements of RFC 3281 are TBD.