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: Aug. 04, 2005

───

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: TRUSTED OUT and TRUSTED 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 Security Protocol identified in the CDB. The intent is to standardize this data content so it is identical across both ATA and SCSI. This proposal refers to the data payload format as "restricted" to indicate that the format shall conform to the definition established by the outside group that "owns" the particular SPID (Security Protocol ID) value.

0.1 Reference documents

— 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.
— ITU-T RECOMMENDATION X.509 | ISO/IEC 9594-8, *Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks*, ITU, 2000.

0.2 Definitions

0.2.a Trusted Action: A group of bytes that describe a security procedure that a device server is requested to perform.

0.2.b Trusted Result: A group of bytes that contain the results of a requested trusted action or the status from processing a requested trusted action.

0.3 Abbreviations

0.3.a TCG: Trusted Computing Group.

6.x Trusted Out command

The TRUSTED OUT command (see table 1) is used to send data to the device server. The data sent contains one or more trusted actions to be performed by the device server. The application client shall use TRUSTED IN command to retrieve any trusted results derived from the trusted actions.

Table 1 – Trusted Out command

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

The SPID (Security Protocol Identification) field identifies which security protocol is being used. This determines the format of the data that is transferred. The meaning of the SPID value is described in table 2.

Table 2 – Security Protocol ID field description

Value	Description
0h	Request for an X.509 certificate (see 6.y.2)
01h – 06h	Restricted to TCG.
07h – 0Bh	Reserved.
0Ch – 0Fh	Vendor Specific.
10h - FFh	Reserved

The TRANSFER LENGTH field specifies the number of bytes to be transferred divided by 512. The transfer size shall be a multiple of 512 bytes. Pad bytes as added as needed to meet this requirement. Pad bytes shall have a value of 00h.

The device server shall return GOOD status as soon as it determines the data has been correctly received. This does not indicate that the data has been parsed or that any trusted actions have been processed. These indications are only obtained by sending a TRUSTED IN command and receiving the results in the associated data transfer.

The data is organized as one or more trusted actions. For SPID set to zero, The TRANSFER LENGTH shall be zero and no data is transferred. In this case, if the transfer length is non-zero the command is terminated with CHECK CONDITION status with sense key set to ILLEGAL REQUEST and ASC set to Illegal Bit or Byte in CDB. The format of the trusted actions for other SPID values is documented by the group that owns the associated SPID value (e.g., the data format for a value of 01h is documented by TCG).

6.y Trusted In command

6.y.1 Trusted In command description

The TRUSTED IN command (see table 3) is used to retrieve trusted results from trusted actions that were sent in a previous TRUSTED OUT command.

Table 3 – Trusted In command

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

The SPID (Security Protocol Identification) field identifies which security protocol is being used. This determines the format of the data that is transferred. The meaning of the SPID values is defined in table 2.

The TRANSFER LENGTH field specifies the number of bytes to be transferred divided by 512. The transfer size shall be a multiple of 512 bytes. Pad bytes are added as needed to meet this requirement. Pad bytes shall have a value of 00h. If the transfer length is not sufficient to return all of the data bytes the device server has available to send, the device server shall send as many complete trusted results as possible without exceeding the transfer length.

The device server shall only return trusted results that were requested by a previous TRUSTED OUT command that specified the same I_T_L nexus and the same SPID value.

A single TRUSTED IN command may return trusted results requested in one or more previous TRUSTED OUT commands. A TRUSTED IN command may also return partial trusted results requested in a previous TRUSTED OUT command. If the device server has no trusted results to send (e.g., trusted results for a previously requested trusted action are not ready yet), the device server shall return a trusted result indicating it has no data to return and the command shall end with GOOD status.

The returned data is organized as one or more trusted results. For SPID set to 00h, the returned data is an X3.509 certificate. The format for this data is described in 6.y.2. The format of the trusted results for other SPID values is restricted. The format is documented by the group that owns the associated SPID value (e.g., the data format for a value of 01h is documented by TCG).

It is the application client's responsibility to have an outstanding TRUSTED IN command whenever it believes there are trusted results pending in the device server.

The device server shall retain trusted result data resulting from a TRUSTED OUT command awaiting retrieval by a TRUSTED IN command until one of the following events occurs:

   a) a matching TRUSTED IN command;
   b) logical unit reset;
   c) I_T nexus loss associated with the I_T nexus that sent the TRUSTED OUT command;
   d) processing any of the following task management functions (see SAM-3):
          A) CLEAR TASK SET;
          B) ABORT TASK SET.

If the trusted result data is lost due to one of these events and the application client still wants to perform the trusted action, the application client is required to resend the trusted action in a new TRUSTED OUT command.

6.y.2 Security Protocol ID 00h description

6.y.2.1 CDB description

The purpose of security protocol ID of 00h is to return the X3.509 certificate for the device. Typically this security identification credential is retrieved as part of a discovery process before initiating a specific security protocol with the device.

Note: X.509 certificates are designed to be transferred and/or stored as plaintext.

 For SPID field set to 00h, the SP_SPECIFIC bytes are defined as shown in table 4.

Table 4 – SP_SPECIFIC description for SPID = 00h

Bit / Byte	7	6	5	4	3	2	1	0
1	SPID = 00H							
2	(MSB)	SP_SPECIFIC						
3								(LSB)

The SP_SPECIFIC field shall be set to 0000h for both the TRUSTED OUT and TRUSTED IN commands. All other CDB fields for TRUSTED OUT command shall meet the requirements of 6.x. All other CDB fields for TRUSTED IN command shall meet the requirements of 6.y.1.

6.y.2.2 Certificate header

When the SPID field is set to 00h in a TRUSTED IN command, an X.509 certificate is returned in the data. A header, the certificate bytes, and pad bytes (if any) shall be returned as shown in Table 7.

Table 7 – X.509 header and certificate description

Bit / Byte	7	6	5	4	3	2	1	0
0	RESERVED							
1	RESERVED							
2	(MSB)	CERTIFICATE LENGTH (m - 3)						
3								(LSB)
4 ... m	X.509 CERTIFICATE BYTES							
m+1 ... n	PAD BYTES (IF ANY)							

The CERTIFICATE LENGTH 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 return, the certificate length is set to 0000h and only the 4 byte header and 508 pad bytes are returned.

The total data length shall conform to the TRANSFER LENGTH field requirements (e.g., the total data length shall be a multiple of 512). Pad bytes are added as needed to meet this requirement. Pad bytes shall have a value of 00h.

6.y.2.3 Certificate description

6.y.2.3.1 Certificate overview

The instantiation of a X.509 conformant credential is either through an X.509 Attribute Certificate or an X.509 Public Key Certificate depending on the capabilities of the device. A X.509 Attribute Certificate shall be issued for any device not capable of asymmetric key operations or any device for which the credential issuer does not want to include any public key information in the credential. A X.509 Public Key Certificate shall be issued for any device capable of asymmetric key operations and for which the certificate issuer wants to bind the public key to the device.

6.y.2.3.2 Public Key certificate description

RFC 3280 defines the certificate syntax for certificates consistent with X.509v3 Public Key Certificate Specification. Table 8 describes the trusted command usage of the X.509 public key certificate fields and the relationship of that usage to the definitions of RFC 3280.

Table 8 –Usage of X.509 certificate values in RFC 3280 context

Certificate Field [1]	Details
SignatureAlgorithm	As described in RFC 3280.
SignatureValue	As described in RFC 3280.
Version	Shall be version 3.
SerialNumber	As described in RFC 3280.
Signature	As described in RFC 3280.
Issuer	As described in RFC 3280 with the added constraint that UTF8String encoding of DirectoryString shall be used.
Validity	As described in RFC 3280. It is recommended to set Begin Date to the time of credential issuance and the Expiration Date to the Begin Date plus one hundred years if the intent is not to indicate an expiration date.
Subject	As described in RFC 3280. Information contained in this field shall either be populated with a non-empty distinguished name identifying the device or a null value.
SubjectPublicKeyInfo	As described in RFC 3280.
subject Alternate Name Extension	As described in RFC 3280, but may be ignored. This specification restricts the use to the following options only: • otherName; • directoryName. One and only one of the following values is allowed for subjectAltName: • The device serial number using directoryName; • The device serial number using otherName. If this field is used then subject field shall contain a null value.
basicConstraints Extension	As described in RFC 3280.
cRLDistributionPoints Extension	As described in RFC 3280.
subjectDirectoryAttributes Extension: protocols	SEQUENCE OF OID Defines supported Security and Integrity Protocols
[1] Certificate field names are as described in RFC 3280.	

6.y.2.3.3 Attribute certificate description

RFC 3281 defines the certificate syntax for certificates consistent with X.509v2 Attribute Certificate Specification. Table 9 describes the trusted command usage of the X.509 attribute key certificate fields and the relationship of that usage to the definitions of RFC 3281.

Table 9 –Usage of X.509 certificate values in RFC 3281 context

Certificate Field [1]	Details
SignatureAlgorithm	As described in RFC 3281.
SignatureValue	As described in RFC 3281.
Version	Shall be version 2.
Holder	As described in RFC 3281 with the added constraint that entityName option be used in the Holder field containing one and only one of the of the following values: • an URI using uniformResourceIdentifier; • the device serial number using directoryName or otherName; • a null value.
issuer	As described in RFC 3281.
signature	As described in RFC 3281.
serialNumber	As described in RFC 3281.
attrCertValidityPeriod	As described in RFC 3281. It is recommended to set Begin Date to the time of credential issuance and the Expiration Date to the Begin Date plus one hundred years if the intent is not to indicate an expiration date.
attributes: protocols	SEQUENCE OF OID Defines supported Security and Integrity Protocols.
basicAttConstraints Extension	As described in RFC 3281.
cRLDistributionPoints Extension	As described in RFC 3281.
[1] Certificate field names are as described in RFC 3281.	