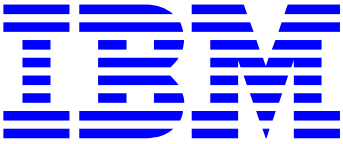
To: INCITS Technical Committee T10
From: Kevin Butt
Date: Monday, April 20, 2009 6:38 pm
Document: T10/08-390r0 — SSC-3: Resolution to LB SYM-019 item a



Revisions

08-390r0 (20 April 2009) Initial revision

Introduction

This proposal intends to resolve LB SYM-019 item a:

> [4.2.21.5 Keyless copy] should identify: a) How an application client determines that a Logical Unit has the capability to act as a KCSLU or a KCDLU

After several correspondance with Roger Cummings - and several times that we dropped the conversation and picked it up again forgetting everything, I am taking the final email I see between Roger and myself and will propose the following:

> In SSC-3 the only capability indicator I find related to this is in the algorithm descriptor - the RDMC_C. In the purist sense, this is for the algorithm support and not necessarily the drive. Practically, however, I think it is actually the capabilities supported for the algorithm in this drive. If this is true it does report the capabilities of the RAW read (or KCSLU). If you take that along with the new proposal that will mandate if RAW read is supported EXTERNAL write shall be supported and vice-versa, then you can determine everything you need.
>
> However, perhaps the cleaner solution is to add an "external encryption mode control capabilities (EEMC_C)" field to mirror the RDMC_C field and not mandate EXTERNAL if RAW is supported. The EEMC_C field would have to be defined as:
>
> 0h - No capabilities are specified.
> 1h - The encryption algorithm does not allow write operations in EXTERNAL encryption mode.
> 2h - The encryption algorithm allows write operations in EXTERNAL encryption mode.
> 3h - Reserved
>
> It does not make sense to have all the other modes from the RDMC_C field.

Proposal

4.2.21.5 Keyless copy of encrypted data

In some scenarios it is desirable to copy data from one volume to another without needing knowledge of the encryption parameters used to encrypt the data on the volume.

A keyless copy logical unit (KCLU) controls configuration and data flows related to a volume that is either a source or destination for encrypted data being transferred without requiring application client knowledge of an encryption key.

A keyless copy source logical unit (KCSLU) controls configuration and data flows related to the volume from which the encrypted data is copied without requiring device server knowledge of an encryption key when the decryption mode is set to RAW. The device servers capability to act as a KCSLU for a specific encryption algorithm is indicated in the RDMC_C field of the Data Encryption Capabilities page (see 8.5.2.4).

A keyless copy destination logical unit (KCDLU) controls configuration and data flows related to the volume to which the encrypted data is being copied without requiring device server knowledge of an encryption key when

the encryption mode is set to EXTERNAL. The device servers capability to act as a KCDLU for a specific encryption algorithm is indicated in the EEMC_C field of the Data Encryption Capabilities page (see 8.5.2.4).

To accomplish a keyless copy operation an application client sets the KCSLU decryption mode to RAW and the KCDLU encryption mode to EXTERNAL. The application client then reads one or more logical objects from the KCSLU and writes those logical objects to the KCDLU. During this process, if the KCSLU detects a mismatch between the key-associated data in the data encryption parameters and the key-associated data on the medium during a read operation, then the KCSLU returns a CHECK CONDITION status to the application client to notify it that some action is required. An example of this is shown in the informative flowchart in Annex C.

.

.

.

8.5.2.4 Data Encryption Capabilities page

The Data Encryption Capabilities page *(see table 121)* requests that information regarding the set of data encryption algorithms reported by this device server be sent to the application client. If external data encryption control is supported, then the set of data encryption algorithms reported by the device server may not include all of the algorithms in the set of data encryption algorithms supported by the physical device.

Table 121 — Data Encryption Capabilities page

Bit / Byte	7	6	5	4	3	2	1	0
0	(MSB)							
1	PAGE CODE (0010h)							(LSB)
2	(MSB)							
3	PAGE LENGTH (n-3)							(LSB)
4	Reserved				EXTDECC		CFG_P	
5	Reserved							
19								
	Data Encryption Algorithm descriptor list							
20	Data Encryption Algorithm descriptor (first)							
	Data Encryption Algorithm descriptor (last)							
n								

See SPC-4 for a description of the PAGE CODE field PAGE LENGTH field. The page code field shall be set to the value specified in *table 121*.

The external data encryption control capable (EXTDECC) field specifies the external data encryption control capability of the physical device. The EXTDECC field values are specified in *table 122*.

Table 122 — EXTDECC field values

Code	Description
00b	The external data encryption control capability is not supported.
01b	The physical device is not external data encryption control capable.
10b	The physical device is external data encryption control capable.
11b	Reserved

The configuration prevented (CFG_P) field specifies the data encryption parameters configuration capabilities for the algorithms reported in the Data Encryption Algorithm descriptors. The CFG_P field values are specified in *table 123*.

Table 123 — CFG_P field values

Code	Description
00b	The data encryption configuration capabilities are not reported.
01b	The physical device configured to allow this device server to establish or change data encryption parameters.
10b	The physical device is configured to not allow this device server to establish or change data encryption parameters.
11b	Reserved

Each Data Encryption Algorithm descriptor *(see table 124)* contains information about a data encryption algorithm supported by the device server. If more than one descriptor is included, they shall be sorted in ascending order of the value in the ALGORITHM INDEX field.

Table 124 — Data Encryption Algorithm descriptor

Bit / Byte	7	6	5	4	3	2	1	0
0	ALGORITHM INDEX							
1	Reserved							
2	(MSB)	DESCRIPTOR LENGTH (20)						
3								(LSB)
4	AVFMV	SDK_C	MAC_C	DED_C	DECRYPT_C		ENCRYPT_C	
5	AVFCLP		NONCE_C		Reserved	VCELB_C	UKADF	AKADF
6	(MSB)	MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES						
7								(LSB)
8	(MSB)	MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES						
9								(LSB)
10	(MSB)	KEY SIZE						
11								(LSB)
12	Reserved		EEMC_C		RDMC_C			EAREM
13	Reserved							
19								
20	(MSB)	SECURITY ALGORITHM CODE						
23								(LSB)

The ALGORITHM INDEX field is a device server assigned value associated with the algorithm that is being described. The value in the ALGORITHM INDEX field is used by the SECURITY PROTOCOL OUT command Set Data Encryption page to select this algorithm.

The algorithm valid for mounted volume (AVFMV) bit shall be set to one if there is a volume currently mounted and the encryption algorithm being described is valid for that volume. The AVFMV bit shall be set to zero if there is no volume mounted or the algorithm is not valid for the currently mounted volume.

The supplemental decryption key capable (SDK_C) bit shall be set to one if the device server is capable of supporting one or more supplemental decryption keys. The supplemental decryption keys shall be used for decryption only. The SDK_C bit shall be set to zero if the device server is not capable of supporting supplemental decryption keys.

The distinguish encrypted data capable (DED_C) bit shall be set to one if the device server is capable of distinguishing encrypted data from unencrypted data when reading it from the medium. The DED_C bit shall be set to zero if the device server is not capable of distinguishing encrypted data from unencrypted data when reading it from the medium. If the ability to distinguish encrypted data from unencrypted data is format specific and a volume is mounted, the DED_C bit shall be set based on the current format of the medium. If no volume is mounted, the DED_C bit shall be set to one if the device server is capable of distinguishing encrypted data from unencrypted data in any format that the device server supports.

The message authentication code capable (MAC_C) bit shall be set to one if the algorithm includes a message authentication code added to encrypted blocks. The MAC_C bit shall be set to zero if the algorithm does not include a message authentication code added to encrypted blocks. If the inclusion of a message authentication

code is format specific and a volume is mounted, the MAC_C bit shall be set based on the current format of the medium. If no volume is mounted, the MAC_C bit shall be set to one if the device server adds a message authentication code to data encrypted with this algorithm in any format that the device server supports.

The DECRYPT_C field *(see table 125)* specifies the decryption capabilities of the physical device.

Table 125 — DECRYPT_C field values

Code	Name	Description
00b	no capability	The physical device has no has data decryption capability using this algorithm. This value shall be returned if the specified algorithm is disabled.
01b	software capable	The physical device has the ability to decrypt data using this algorithm in software.
10b	hardware capable	The physical device has the ability to decrypt data using this algorithm in hardware.
11b	capable with external control	The physical device has the capability to decrypt data using this algorithm, but control of the data encryption parameters by this device server is prevented.

The ENCRYPT_C field *(see table 126)* specifies the encryption capabilities of the physical device.

Table 126 — ENCRYPT_C field values

Code	Name	Description
00b	no capability	The physical device has no has data encryption capability using this algorithm. This value shall be returned if the specified algorithm is disabled.
01b	software capable	The physical device has the ability to encrypt data using this algorithm in software.
10b	hardware capable	The physical device has the ability to encrypt data using this algorithm in hardware.
11b	capable with external control	The physical device has the capability to encrypt data using this algorithm, but control of the data encryption parameters by this device server is prevented.

The algorithm valid for current logical position (AVFCLP) field specifies if the encryption algorithm being specified is valid for writing to the mounted volume at the current logical position. Table 127 specifies the values for the AVFCLP field.

Table 127 — AVFCLP field values

Code	Description
00b	Current logical position is not applicable to the encryption algorithm validity or no volume is loaded.
01b	The ecryption algorithm being specified is not valid for writing to the mounted volume at the current logical position.
10b	The ecryption algorithm being specified is valid for writing to the mounted volume at the current logical position.
11b	Reserved

Table 128 specifies the values for the NONCE_C field.

Table 128 — NONCE_C field values

Code	Description
0	This algorithm does not require a nonce value.
1	The device server generates the nonce value.
2	The device server requires all or part of the nonce value to be provided by the application client.
3	The device server supports all or part of the nonce value provided by the application client. If the Set Data Encryption page that enables encryption does not include a nonce value descriptor, the device server generates the nonce value.

If the volume contains encrypted logical block capable (VCELB_C) bit is set to one, then the device server is capable of determining that a volume contains logical blocks encrypted using this algorithm when the volume is mounted. If the VCELB_C is set to zero, then the device server is not capable of determining that a volume contains logical blocks encrypted using this algorithm when the volume is mounted. If the capability of determining that a volume contains logical blocks encrypted using this algorithm is format specific and a volume is mounted, then the VCELB_C bit is set based on the current format of the medium. If no volume is mounted, the VCELB_C bit is set to one if for at least one algorithm that the device server supports the device server is capable of determining that a volume contains logical blocks encrypted using that algorithm.

The U-KAD Fixed (UKADF) bit shall be set to one if the device server requires the length of U-KAD in the parameter data for a SECURITY PROTOCOL OUT command to equal the value in the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED BYTES field. If the UKADF bit is set to one, then the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED BYTES field shall contain a non-zero value. If the UKADF bit is set to zero and the value in the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED BYTES field is non-zero, then the length of the U-KAD, if present in the parameter data for a SECURITY PROTOCOL OUT command, shall be a value between one and the value in the MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED BYTES field.

The A-KAD Fixed (AKADF) bit shall be set to one if the device server requires the length of A-KAD in the parameter data for a SECURITY PROTOCOL OUT command to equal the value in the MAXIMUM AUTHENTICATED KEY-ASSOCIATED BYTES field. If the AKADF bit is set to one, then the MAXIMUM AUTHENTICATED KEY-ASSOCIATED BYTES field shall contain a non-zero value. If the AKADF bit is set to zero and the value in the MAXIMUM AUTHENTICATED KEY-ASSOCIATED BYTES field is non-zero, then the length of the A-KAD, if present in the parameter data for a SECURITY PROTOCOL OUT command, shall be a value between one and the value in the MAXIMUM AUTHENTICATED KEY-ASSOCIATED BYTES field.

The MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES field indicates the maximum size of the unauthenticated key-associated data *(see 4.2.21.13 on page 60)* that the device server can support for this algorithm.

The MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES field indicates the maximum size of the authenticated key-associated data *(see 4.2.21.13 on page 60)* that the device server can support for this algorithm.

The KEY SIZE field indicates the size in bytes of the encryption key required by the algorithm.

The external encryption mode control capabilities (EEMC_C) field indicates the capabilities the encryption algorithm provides to the application client to control write operations that transfer encrypted blocks while the encryption mode is set to EXTERNAL. Table 129 defines the values for the EEMC_C field.

Table 129 — EEMC_C field values

Code	Description
0h	No capabilites are specified
1h	The encryption algorithm does not allow write operations in EXTERNAL encryption mode. The device server does not act as a KCDLU (see 4.2.21.5) for this encryption algorithm.
2h	The encryption algorithm allows write operations in EXTERNAL encryption mode. The device server does act as a KCDLU (see 4.2.21.5) for this encryption algorithm.
3h	Reserved

The raw decryption mode control capabilities (RDMC_C) field indicates the capabilities the encryption algorithm provides to the application client to control read operations that access encrypted blocks while the decryption mode is set to RAW. Table 130 defines the values for the RDMC_C field.

Table 130 — RDMC_C field values

Code	Description
0h	No capabilities are specified.
1h	The encryption algorithm does not allow read operations in RAW decryption mode. The device server does not act as a KCSLU (see 4.2.21.5) for this encryption algorithm.
2h-3h	Reserved
4h	The encryption algorithm disables read operations in RAW mode by default and allows the application client to control RAW reads via the RDMC field in the Set Data Encryption page (see 8.5.3.2). The device server acts as a KCSLU (see 4.2.21.5) for this encryption algorithm.
5h	The encryption algorithm enables read operations in RAW mode by default and allows the application client to control RAW reads via the RDMC field in the Set Data Encryption page (see 8.5.3.2). The device server acts as a KCSLU ((see 4.2.21.5) for this encryption algorithm.
6h	The encryption algorithm disables read operations in RAW mode by default and does not allow the application client to control RAW reads via the RDMC field in the Set Data Encryption page (see 8.5.3.2). The device server does not act as a KCSLU (see 4.2.21.5) for this encryption algorithm.
7h	The encryption algorithm enables read operations in RAW mode by default and does not allow the application client to control RAW reads via the RDMC field in the Set Data Encryption page (see 8.5.3.2). The device server acts as a KCSLU (see 4.2.21.5) for this encryption algorithm.

The encryption algorithm records encryption mode (EAREM) bit shall be set to one if the encryption mode is recorded with each encrypted block. The EAREM bit shall be set to zero if the encryption mode is not recorded with each encrypted block.

The SECURITY ALGORITHM CODE field contains an security algorithm code (see SPC-4).