To: INCITS T10 Committee

From: Paul Entzel, Quantum

Date: 12 December 2005

Document: T10/05-446r1

Subject: SSC-3: Add commands to control data encryption

1 Revision History

Revision 0:
Posted to the T10 web site on 16 November 2005.

Revision 1:
Posted to the T10 web site on 12 December 2005. Includes:

1. In 3.1.X, change “ciphering process was performed by the device server” to “ciphering process was performed by a device server”.
2. In 7.X.4, change AK_C to MAC_C and the paragraph that describes it.
3. Add CKOD bit to Set Data Encryption page.
4. Add microcode update and other vendor specific events to the list of events that clear a key.
5. Add a paragraph to 4.2.19.5 describing the loss of a key due to a vendor specific event.
6. Change GLOBAL scope to ALL I_T NEXUS.
7. Added Key Generation concept (4.2.18.6 and 4.2.19.7) and removed the ENCRYPTE bit from the WRITE(6) and WRITE(16) commands.
8. Added descriptors for key-associated data (subclause 8.5). Add these descriptors to the Data Encryption Status page and the Set Data Encryption page.
9. Added Next Block Status page to report encryption status and KAD data for the next block to read (still looking for a better name for this page).
10. Added ALGORITHM INDEX field to the Data Encryption Status page.
11. Expand the ENCRYPT and DECRYPT fields (renamed ENCRYPTION MODE and DECRYPTION MODE) in the Set Data Encryption page and the Data Encryption Status page to 4 bits and add multiple modes from HP’s strawman proposal (with some name changes).
12. Add LOCK bit to Set Data Encryption and Data Encryption Status pages and a model clause to describe it.
13. Add maximum key-associated data lengths to the algorithm descriptors.
14. Added acronyms for A-KAD and U-KAD.
15. Added a key descriptor for the application client to set the nonce and for the device server to report this capability/requirement.

2 General

A great deal of interest has been expressed to add access security to data recorded on removable medium in the wake of several high profile cases where medium containing sensitive data has been lost or stolen. Many back-up application support adding passwords to data sets and/or encrypting the data. The password scheme does not protect from the data being recovered using a different application that understands the format but does not honor the password protection. The encryption approach is safer, but introduces significant performance degradation when used.

The major disadvantage to encrypting the data concerns the performance impact. Software algorithms to encrypt the data are available, but they take significant time to perform and this reduces the available performance. Also, since the encryption process removes redundancy in the data, attempts to compress the data after encryption are usually unsuccessful. In order to regain the performance boost from compression, the compression must be done before encryption, so it must be done in software. This adds even more time overhead, reducing the performance even more.

This proposal provides a method where encryption could be moved into the device, solving the problems described above. This proposal provides a method to control these features by introducing two new commands to the SSC-3 command set.

The DATA SECURITY IN command is a Service Action derivative of the SERVICE ACTION IN (12) command. This command can request several reports containing supported features and enabled modes.

The DATA SECURITY OUT command is a derivative of the SERVICE ACTION OUT (12) command. This command is used to set encryption keys and modes of protection.

Perhaps the most difficult problem to solve with the addition of this interface is how to deal with multiple initiators. One would think that the parameters controlling encryption and decryption would be I_T nexus specific, and that they would not be used without reservations in place. However, that won't work in many environments:

1. If the target port is behind a protocol bridge, the device server can not tell one initiator from another. In this environment, parameters that are normally per I_T nexus are shared at the target device level and the protocol bridge is required to sort out the individual initiator ports.
2. To support copy managers, we either need to add the ability to pass encryption modes and keys via the EXTENDED COPY command, have the ability to pass the set-up from one I_T Nexus to another, or have the ability to share set-up.

Future enhancements to this feature may include adding a method to encrypt the data encryption key before passing it to the device server. The addition of this single feature was proving to be significantly more complex than what was already contained in the proposal. We decided to postpone discussion of this feature until a later proposal. This proposal establishes plenty of reserved fields and page codes so that it can be added without breaking everything else.

3 Changes to SSC-3

3.1 Additions to the definition clause (3)

3.1.X Encrypted Block – A Logical Block in which the data has been subjected to a ciphering processes by the device server. Within this standard, a logical block is only considered encrypted if the ciphering process was performed by a device server.

3.1.Y Unencrypted Block – A logical Block in which the data has not been subjected to a ciphering process by the device server.

In subclause 3.2 add the following acronyms:

A-KAD Authenticated key-associated data

U-KAD Unauthenticated key-associated data

3.2 Addition to the model clause (4)

3.2.1 Additions to subclause 4.2.17 Reservations

Add to table 12 the following commands:

Command	Addressed LU 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 I_T nexus not registered	
	Write Exclusive	Exclusive Access		Write Exclusive - RR	Exclusive Access - RR
DATA SECURITY IN	Conflict	Conflict	Allowed	Allowed	Conflict
DATA SECURITY OUT	Conflict	Conflict	Allowed	Conflict	Conflict

3.2.2 Additional subclause

Add the following subclause to clause 4:

4.2.19 Data Encryption

4.2.19.1 Data Encryption overview

An SSC-3 compliant device may contain hardware or software that is capable of encrypting the data within logical blocks to provide security against unauthorized access to that data. The DATA SECURITY IN and DATA SECURITY OUT commands provide a means for the application client to monitor and control the encryption process within the device. A device server that supports the DATA SECURITY OUT command shall also support the DATA SECURITY IN command.

The DATA SECURITY OUT command is used to enable and disable encryption mode and set the encryption key. The DATA SECURITY IN command is used to discover the type of data security features supported by the device server and the current configuration of data security features.

4.2.19.2 Encrypting data on the medium

The application client controls the data encryption process by use of the DATA SECURITY OUT command. Data encryption is enabled after the device server successfully processes a DATA SECURITY OUT command that sends a Set Encryption Key page (see 7.Y.2) with the ENCRYPTION MODE field set to ENCRYPT and a valid key. Data encryption is disabled for an I_T nexus after:

a) a DATA SECURITY OUT command from the I_T nexus with a Set Encryption Key page is processed successfully with the ENCRYPTION MODE field set to a value other than ENCRYPT;

b) the CKOD bit was set to one in the Set Encryption Key page that established the key and the volume is dismounted;

c) the CKORL bit was set to one in the Set Encryption Key page that enabled encryption and the I_T nexus that had enabled encryption losses its reservation;

d) the key with a scope of ALL I_T NEXUS or RESERVATION GROUP being used by the I_T nexus is cleared;

e) a microcode update is performed on the device;

f) a hard reset; or

g) other vendor specific events.

While the data encryption process is enabled, all data received by the device server as part of a WRITE(6) or WRITE(16) command shall be encrypted before being recorded on the medium. Filemarks written due to a WRITE FILEMARKS(6) or WRITE FILEMARKS(16) command shall not be affected by the encryption process.

4.2.19.3 Reading encrypted data on the medium

A volume may contain no encrypted blocks, all encrypted blocks, or a mixture of encrypted blocks and unencrypted blocks. Encrypted blocks shall have no impact on space or locate commands.

A device server that supports encryption should be able to distinguish encrypted blocks from unencrypted blocks. If the device server is able to distinguish encrypted blocks from unencrypted blocks, an attempt to read or verify an encrypted block when decryption has not been enabled shall cause the device server to terminate the command with CHECK CONDITION status, with the Sense Key set to DATA PROTECT, and the addition sense code set to UNABLE TO DECRYPT DATA. The device server shall establish the logical position at the BOP side of the encrypted block.

Editor's note: UNABLE TO DECRYPT DATA is a new ASC.

If the device server is able to distinguish encrypted blocks from unencrypted blocks, an attempt to read or verify an unencrypted block when the DECRYPTION MODE field is set to DECRYPT shall cause the device server to terminate the command with CHECK CONDITION status, with the Sense Key set to DATA PROTECT, and the addition sense code set to UNENCRYPTED DATA ENCOUNTERED WHILE DECRYPTING. The device server shall establish the logical position at the BOP side of the unencrypted block.

Editor's note: UNENCRYPTED DATA ENCOUNTERED WHILE DECRYPTING is a new ASC.

A device server that supports encryption and has been configured to decrypt the data may be able to determine if the encryption key is correct for an encrypted block. If the device server is able to determine if the encryption key is correct, an attempt to read or verify an encrypted block when decryption is enabled but the encryption key is incorrect for the block shall cause the device server to terminate the command with CHECK CONDITION status, with the Sense Key set to DATA PROTECT, and the additional sense code set to INCORRECT DATA ENCRYPTION KEY. The device server shall establish the logical position at the BOP side the encrypted block.

Editor's note: INCORRECT DATA ENCRYPTION KEY is a new ASC.

4.2.19.4 Exhaustive-search attack prevention

To prevent an exhaustive-search attack from discovering the encryption key, the device server shall provide a mechanism to prevent unlimited attempts at setting a data encryption key and then attempting to read the data.

If the device server is not capable of distinguishing encrypted data from unencrypted data or is not capable of authenticating the encryption key, it should limit the key changes by introducing a delay in the processing of the DATA SECURITY OUT command that will slow the key change process. The delay should be sufficient so that trying all possible encryption keys will take so many years that it becomes impractical.

If the device server is capable of distinguishing encrypted data from unencrypted data and is capable of authenticating the data encryption key, it shall use one of the following techniques:

a) introducing a delay in the processing of the DATA SECURITY OUT command that will slow the key change process;

b) limiting the number if attempts at changing the key and reading encrypted data per mount of the volume; or

c) some other vendor unique process that limits the ability to discover the encryption key using and exhaustive-search approach.

If the device server has reached its limit on failed attempts to set the data encryption key and decrypt data, it shall disable decryption for all I_T nexuses. All subsequence DATA SECURITY OUT commands with the PAGE CODE field set to the data encryption page and the DECRYPT bit set to one shall be terminated with CHECK CONDITION status, the Sense Key set to DATA PROTECT, and the additional sense code set to DATA DECRYPTION KEY FAIL LIMIT REACHED. This condition shall persist until the medium is dismounted from the device or a hard reset event.

Editor's note: DATA DECRYPTION KEY FAIL LIMIT REACHED is a new ASC.

4.2.19.5 Managing keys within the device server

The security provided by data encryption is only as good as the security used when managing the keys. For this reasons, the data encryption key and mode are volatile in the device server and never reported to an initiator. The device server also may have limited resources for storage of keys.

If a device server processes a Set Data Encryption page with the ENCRYPTION MODE field set to DISABLE and DECRYPTION MODE field set to DISABLE or RAW, the device server shall release any resources that it had allocated to store a key value for the I_T nexus associated with the DATA SECURITY OUT command and shall clear any memory containing the key value. A unit attention condition with the additional sense of DATA ENCRYPTION MODE CHANGED BY ANOTHER INITIATOR shall be establish for any other I_T nexus that is affected by the loss of the key, (i.e., any I_T nexus that is using a scope of PUBLIC and sharing the key.)

Editor's note: DATA ENCRYPTION MODE CHANGED BY ANOTHER INITIATOR is a new ASC.

If a device server processes a Set Data Encryption page that includes a key, the device server shall release any resources that it had allocated to store a key value set by a previous DATA SECURITY OUT command from that I_T nexus and shall clear any memory containing the key value. A unit attention condition with the additional sense of DATA ENCRYPTION MODE CHANGED BY ANOTHER INITIATOR shall be establish for any other I_T nexus that is affected by the change of the key (i.e., any I_T nexus that is using a scope of PUBLIC and sharing the key).

A device server shall save at most one key with a scope of ALL I_T NEXUS. If a device server processes a Set Data Encryption page with the SCOPE field set to ALL I_T NEXUS, the device server shall release any resources that it had allocated to store a key value set by a previous DATA SECURITY OUT command with a scope value of ALL I_T NEXUS and shall clear any memory containing the key value. A unit attention condition with the additional sense of DATA ENCRYPTION MODE CHANGED BY ANOTHER INITIATOR shall be establish for any other I_T nexus that is affected by the change of the key, that is, any I_T nexus that is using a scope of public and sharing the key.

A device server shall save at most one key with a scope of RESERVATION GROUP. If a device server processes a Set Data Encryption page with the SCOPE field set to RESERVATION GROUP, the device server shall release any resources that it had allocated to store a key value set by a previous DATA SECURITY OUT command with a scope value of RESERVATION GROUP and shall clear any memory containing the key value. A unit attention condition with the additional sense of DATA ENCRYPTION MODE CHANGED BY ANOTHER INITIATOR shall be establish for any other I_T nexus that is affected by the change of the key, that is, any I_T nexus that is using a scope of public and sharing the key.

If a vendor specific event occurs that changes or clears a data encryption key, the device server shall establish a unit attention condition with the additional sense of DATA ENCRYPTION MODE CHANGED BY OUT-OF-BAND EVENT for any I_T nexus that is affected by the change of the key.

Editor's note: DATA ENCRYPTION MODE CHANGED BY OUT-OF-BAND EVENT is a new ASC.

4.2.19.6 Key Generation

The device server shall keep a counter for each key that it is managing called the Key Generation. All Key Generation counters shall be set to zero on a hard reset event. Any other event that sets, clears, or changes the key shall cause the Key Generation counter for that key to be incremented. The value of the Key Generation counter associated with the currently selected key for an I_T Nexus is reported in the Data Encryption Status page of DATA SECURITY IN command.

4.2.19.7 Encryption mode locking

There are conditions outside of the control of an application client which cause the device server to stop encrypting once configured to encrypt (see 4.2.19.2) or could change the key being used to encrypt data. Each of these conditions cause the device server to establish a unit attention condition to report the change of operating mode, but the unit attention condition may not always be reported to the application client through protocol bridges and driver stacks.

The LOCK bit in the Set Data Encryption page is set to one to lock the I_T Nexus that issued the DATA SECURITY IN command to the encryption mode, key, and key generation values established at the completion of the processing of the command. The I_T nexus remains locked to that key and key generation value until a hard reset or another DATA SECURITY IN command including a Set Data Encryption page from the same I_T Nexus is processed.

If the device server processes a WRITE(6) or WRITE(16) command for an I_T Nexus that is locked to an encryption mode, key, and key generation, and the key has been cleared or the encryption mode or key generation has changed since the time it was locked, the device server shall terminate the command with CHECK CONDITION status, the Sense Key set to DATA PROTECT, and the additional sense code set to DATA ENCRYPTION KEY GENERATION HAS CHANGED. All subsequent WRITE(6) and WRITE(16) commands shall also be terminated this way until such time as a hard reset event occurs or a DATA SECURITY IN command including a Set Data Encryption page from the same I_T Nexus is processed.

Editor's note: DATA ENCRYPTION KEY GENERATION HAS CHANGED is a new ASC.

4.2.19.8 Nonce generation

Some encryption algorithms require the use of a none value or number used once value. Encryption algorithms that require a nonce value typically use it so that each block that is encrypted with a given key has a unique initialization vector or tweaking constant. The uniqueness of the nonce value prevents encryption breaking methods that determine the encryption key based on how multiple blocks are encrypted.

For a given encryption algorithm, the device server may:

a) not require a nonce value;

b) generate its own nonce value;

c) require a nonce value or part of the nonce value be provided by the application client; or

d) be configurable with respect to the source of the nonce value.

The device server reports its capability with respect to nonce values in the Encryption Algorithm Descriptions (see 7.X.4). If the device server reports that it requires a nonce value from the application client and a Set Data Encryption page is processed that does not include a nonce value descriptor, the device server shall terminate the command with CHECK CONDITION, the Sense Key shall be set to ILLEGAL REQUEST, and the additional sense code set to INCOMPLETE KEY-ASSOCIATED DATA SET.

Editor's note: INCOMPLETE KEY-ASSOCIATED DATA SET is a new ASC.

3.3 Changes to the explicit command set clause (5)

3.3.1 Addition to subclause 5.1 Summary of commands for explicit address mode

In table 13, add the following commands:

Command Name	Type	Opcode	Synchronization Operation Required	Reference
DATA SECURITY IN	O	A3h/XXh	No	7.X
DATA SECURITY OUT	O	A4h/XXh	No	7.Y

3.4 Changes to the implicit command set clause (6)

3.4.1 Addition to subclause 6.1 Summary of commands for implicit address mode

In table 20, add the following commands:

Command Name	Type	Opcode	Synchronization Operation Required	Reference
DATA SECURITY IN	O	A3h/XXh	No	7.X
DATA SECURITY OUT	O	A4h/XXh	No	7.Y

3.5 Additions to common commands clause (7)

7.X. DATA SECURITY IN command

7.X.1 DATA SECURITY IN command description

The DATA SECURITY IN (see table A1) requests the device server to return information about the data security methods in the device server and on the medium. The command supports a series of pages that are requested individually.

Table A1 –DATA SECURITY IN command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	Reserved			SERVICE ACTION				
2	PAGE CODE							
3	(MSB)			Reserved				
5								(LSB)
6	(MSB)			ALLOCATION LENGTH				
9								(LSB)
10	Reserved							
11	CONTROL							

The PAGE CODE field (see Table B1) indicates the type of report that the application client is requesting.

Table B1 – PAGE CODE field values

Value	Description	Reference
00h	Supported DATA SECURITY IN pages	7.X.2
01h	Supported DATA SECURITY OUT pages	7.X.3
02h – 0Fh	Reserved	
10h	Data encryption capabilities page	7.X.4
11h	Data encryption status page	7.X.5
12h	Next block status page	7.X.6
13h - FFh	Reserved	

The ALLOCATION LENGTH field specifies the maximum number of bytes that the device server may return (see SPC-4).

7.X.2 Supported DATA SECURITY IN pages

Table C1 shows the format of the Supported DATA SECURITY IN pages report.

Table C1 – Supported DATA SECURITY IN pages report

Bit / Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (00h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	(MSB)	SUPPORTED PAGE LIST						
n								(LSB)

The SUPPORTED PAGE LIST field shall contain a list of all of the pages that the device server supports for the DATA SECURITY IN command in numerical order starting with 00h.

7.X.3 Supported DATA SECURITY OUT page

Table D1 shows the format of the Supported DATA SECURITY OUT pages report.

Table D1 – Supported DATA SECURITY OUT pages report

Bit / Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (01h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	(MSB)	SUPPORTED PAGE LIST						
n								(LSB)

The SUPPORTED PAGE LIST field shall contain a list of all of the pages that the device server supports for the DATA SECURITY OUT command in numerical order.

7.X.4 Data encryption capabilities page

Table E1 shows the format of the Data encryption capabilities page.

Table E1 – Data Encryption capabilities page

Bit / Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (10h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
	Encryption algorithm descriptors							
4	(MSB)	Data encryption algorithm descriptor						
								(LSB)
	:							
	(MSB)	Data encryption algorithm descriptor						
s								(LSB)

See SPC-3 for a description of the PAGE LENGTH field.

Each data encryption algorithm descriptor (see table E2) 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 E2 – Data encryption algorithm descriptor

Bit / Byte	7	6	5	4	3	2	1	0
0	ALGORITHM INDEX							
1	Reserved							
2	(MSB)							
3	DESCRIPTOR LENGTH (n-3)							(LSB)
4	Reserved	KBR_C	MAC_C	DED_C	DECRYPT_C		ENCRYPT_C	
5	Reserved		NONCE_C		IV_RN	IV_EOU	IV_WPU	IV_MU
6	(MSB)							
7	MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES							(LSB)
8	(MSB)							
9	MAXIMUM AUTHENTICATED KEY-ASSOCIATED DATA BYTES							(LSB)
10	Reserved							
17								
18	ALGORITHM NAME LENGTH (n-19)							
19								
20	ALGORITHM NAME							
n								

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 DATA SECURITY OUT command Set Data Encryption page to select this algorithm.

The ENCRYPT_C field (see table E3) indicates the encryption capabilities of the device.

Table E3 - ENCRYPT_C field values

Value	Description
0	The device server has no data encryption capability using this algorithm.
1	The device server has the ability to encrypt data using this algorithm in software.
2	The device server has the ability to encrypt data using this algorithm in hardware.
3	Reserved

The DECRYPT_C field (see table E4) indicates the decryption capabilities of the device.

Table E4 - DECRYPT_C field values

Value	Description
0	The device server has no data decryption capability using this algorithm.
1	The device server has the ability to decrypt data using this algorithm in software.
2	The device server has the ability to decrypt data using this algorithm in hardware.
3	Reserved

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 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 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 key by reference capable (KBR_C) bit shall be set to one of the device server supports a value in the key FORMAT FIELD of the Set Data Encryption page that indicates key by reference format. The KBR_C bit shall be set to zero of the device server does not support a value in the key FORMAT FIELD of the Set Data Encryption page that indicates key by reference format.

The initialization vector medium unique (IV_MU) field shall be set to one if the initialization vector used by the encryption algorithm is unique for each medium. The initialization vector medium unique (IV_MU) field shall be set to zero if the initialization vector used by the encryption algorithm is not unique for each medium.

The initialization vector write pass unique (IV_WPU) field shall be set to one if the initialization vector used by the encryption algorithm is unique for each write operation that over writes the same portion of the medium. The initialization vector medium unique (IV_MU) field shall be set to zero if the initialization vector used by the encryption algorithm is not unique for each write operation that over writes the same portion of the medium.

The initialization vector encrypted object unique (IV_EOU) field shall be set to one if the initialization vector used by the encryption algorithm is unique for each encrypted object on the medium. The IV_EOU field shall be set to zero if the initialization vector used by the encryption algorithm is not unique for each encrypted object on the medium.

The initialization vector random number (IV_RN) field shall be set to one if the initialization vector used by the encryption algorithm is either in part or wholly a random number. The IV_RN field shall be set to zero if the initialization vector used by the encryption algorithm is not in part or wholly a random number.

The MAXIMUM UNAUTHENTICATED KEY-ASSOCIATED DATA BYTES field indicates the maximum size of the unauthenticated key-associated data 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 that the device server can support for this algorithm.

Table E5 describes the values in the NONCE_C field.

Table E5 - NONCE_C field values

Value	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.

The ALGORITHM NAME LENGTH contains the length in bytes of the ALGORITHM NAME field.

The ALGORITHM NAME field contains a null terminated, null padded UTF-8 format string that describes the encryption algorithm. The string shall contain the standardization authority that has registered the algorithm if there is a standard that defines it.

7.X.5 Data encryption status page

Table S1 shows the format of the Data Encryption status page.

Table S1 – Data encryption status page

Bit / Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (11h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	DECRYPTION MODE				ENCRYPTION MODE			
5	SCOPE			Reserved		SOURCE		
6	KEY GENERATION							
7	ALGORITHM INDEX							
8	(MSB)	KEY-ASSOCIATED DATA DESCRIPTORS						
n								(LSB)

Table S2 defined the values for the ENCRYPTION MODE field.

Table S2 – ENCRYPTION MODE field values

Value	Name	Description
0h	DISABLE	Data encryption is disabled.
1h	EXTERNAL	The device server has been configured to treat the data associated with WRITE(6) and WRITE(16) commands as if it has been encrypted by a system that is compatible with the algorithm specified by the ALGORITHM INDEX field.
2h	ENCRYPT	The device server has been configured to encrypt all data that it receives for a WRITE(6) or WRITE(16) using the algorithm specified in the ALGORITHM INDEX field.
3h – Fh		Reserved

Table S3 defined the values for the DECRYPTION MODE field.

Table S3 – DECRYPTION MODE field values

Value	Name	Description
0h	DISABLE	Data decryption is disabled. If the device server encounters an encrypted block while reading, it shall not allow access to the block (see 4.2.19.3)
1h	RAW	Data decryption is disabled. If the device server encounters an encrypted block while reading, it shall pass the block and any additional metadata affixed to the block to the host without decrypting it (see 4.2.19.3)
2h	DECRYPT	The device server has been configured to decrypt all data that is read from the medium in response to a READ(6) or READ(16) command or verifying when processing a VERIFY(6) or VERIFY(16) command. The data shall be decrypted using the algorithm specified in the ALGORITHM INDEX field and the key provided in the KEY field. If the device server encounters unencrypted data when processing a READ(6), READ(16), VERIFY(6), or VERIFY(16) command, the device server shall terminate the command with CHECK CONDITION status, with the sense key set to DATA PROTECT, and the additional sense code set to UNENCRYPTED DATA ENCOUNTERED WHILE DECRYPTING. The device server shall leave the medium positioned in front of the unencrypted block.
3h	MIXED	The device server has been configured to decrypt all data that is read from the medium that it determines was encrypted in response to a READ(6) or READ(16) command or verifying when processing a VERIFY(6) or VERIFY(16) command. The data shall be decrypted using the algorithm specified in the ALGORITHM INDEX field and the key provided in the KEY field. If the device server encounters unencrypted data when processing a READ(6), READ(16), VERIFY(6), or VERIFY(16) command, the data shall be processed without decrypting.
4h – Fh		Reserved

The scope field (see table S4) indicates the scope of the data encryption key and mode set by this I_T nexus.

Table S4 - SCOPE field values

Value	Description
0	The data encryption key and mode are default values.
1	The data encryption key and mode are unique to this I_T nexus.
2	The data encryption key and mode set by this I_T nexus are shared with all I_T Nexus that share in a reservation for the logical unit.
3	The data encryption key and mode set by this I_T nexus are shared with all other I_T nexuses.
4 – 7	Reserved

The SOURCE field (see table S5) indicates the source of the encryption key and data encryption mode for this I_T nexus.

Table S5 - SOURCE field values

Value	Description
0	The data encryption key and mode are default values.
1	The data encryption key and mode are unique to this I_T nexus.
2	The data encryption key and mode were established by another I_T nexus and are being shared with other reservation holds.
3	The data encryption key and mode were established by another I_T nexus and are being shared globally.
4- 7	Reserved

The KEY GENERATION field contains the value of the Key Generation counter (see 4.2.19.6) assigned to the key indicated by the SOURCE field value.

The ALGORITHM INDEX field indicates which of the encryption algorithms reported by the DATA SECURITY IN command Data Encryption Capabilities page is selected. If the ENCRYPT and DECRYPT bits are both set to zero, the value in the ALGORITHM INDEX field is undefined.

If encryption and decryption are both disabled, the KEY-ASSOCIATED DATA DESCRIPTORS field shall be not be included in the page.

If encryption or decryption is enabled, the KEY-ASSOCIATED DATA DESCRIPTORS field shall contain data security descriptors (see 8.5) describing attributes assigned to the key defined by the SCOPE and SOURCE fields at the time the key was established in the device server by processing a Set Data Encryption page. If more than one key associated descriptor is included, they shall be order of increasing value of the DESCRIPTOR TYPE field. Descriptors shall be included as defined by the following paragraphs.

An unauthenticated key-associated data descriptor (see 8.5.2) shall be included if an unauthenticated key-associated data descriptor was included in the Set Data Encryption page that established the key in the device server. The VALID bit shall be set to one and the AUTH bit shall be set to zero. The KEY DESCRIPTOR field shall contain the U-KAD value associated with the key.

An authenticated key-associated data descriptor (see 8.5.3) shall be included if an authenticated key-associated data descriptor was included in the Set Data Encryption page that established the key in the

device server. The VALID bit shall be set to one and the AUTH bit shall be set to one. The KEY DESCRIPTOR field shall contain the A-KAD value associated with the key.

A nonce value descriptor (see 8.5.4) shall be included if a nonce value descriptor was included in the Set Data Encryption page that established the key in the device server. The VALID bit shall be set to one and the AUTH bit shall be set to one. The KEY DESCRIPTOR field shall contain the nonce value associated with the key. A nonce value descriptor may be included if no nonce value descriptor was included in the Set Data Encryption page that established the key in the device server. In this case, the KEY DESCRIPTOR field shall be set to the nonce value established by the device server for use with the selected key.

7.X.6 Next block status page

Table N1 shows the format of the Next block status page.

Table N1 – Next block status page

Bit / Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (12h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3)						
3								(LSB)
4	(MSB)	OBJECT NUMBER						
5								(LSB)
6	COMPRESSION STATUS				ENCRYPTION STATUS			
7	ALGORITHM INDEX							
8	(MSB)	KEY-ASSOCIATED DATA DESCRIPTORS						
n								(LSB)

The COMPRESSION STATUS field is described in table N2.

Table N2 – COMPRESSION STATUS field values

Value	Description
0h	The device server is incapable of determining if the next block to read has been compressed.
1h	The device server is capable of determining if the next block to read has been compressed, but is not able to at this time. Possible reasons are: a) The next block has not yet been read into the buffer; b) There was an error reading the next block; or c) There are no more blocks (i.e: end of data).
2h	The next block is not compressed.
3h	The next block is compressed.
4h – Fh	Reserved

The ENCRYPTION STATUS field is described in table N3.

Table N3 – ENCRYPTION STATUS field values

Value	Description
0h	The device server is incapable of determining if the next block to read has been encrypted.
1h	The device server is capable of determining if the next block to read has been encrypted, but is not able to at this time. Possible reasons are: d) The next block has not yet been read into the buffer; e) There was an error reading the next block; or f) There are no more blocks (i.e: end of data).
2h	The next block is not encrypted.
3h	The next block is encrypted by an algorithm that is not supported by this device server. The values in the KEY-ASSOCIATED DATA DESCRIPTORS fields contain information pertaining to the block.
4h	The next block is encrypted by an algorithm supported by this device server. The values in the ALGORITHM INDEX and KEY-ASSOCIATED DATA DESCRIPTORS fields contain information pertaining to the block.
5h – Fh	Reserved

The ALGORITHM INDEX field indicates which of the encryption algorithms reported by the DATA SECURITY IN command Data Encryption Capabilities page was used to encrypt the block. For values in the ENCRYPTION STATUS field (see table N3) that do not indicate the ALGORITHM INDEX field is valid, the algorithm index is undefined.

If the encryption status indicates that the next block is encrypted by a supported algorithm, the device server shall include in the KEY-ASSOCIATED DATA DESCRIPTORS field all key-associated data that is associated with the next block that was recorded on the medium. If more than one key-associated data descriptor is include in the page, they shall be in increasing numeric order of the value in the DESCRIPTOR TYPE field.

An unauthenticated key-associated data descriptor (see 8.5.2) shall be included if any unauthenticated key-associated data is associated with the next block. The VALID bit shall be set to one and the AUTH bit shall be set to zero. The KEY DESCRIPTOR field shall contain the U-KAD value associated with the block.

An authenticated key-associated data descriptor (see 8.5.3) shall be included if any authenticated key-associated data is associated with the next block. The VALID bit shall be set to one. The AUTH bit shall be set to one if the device server has authenticated the data. The AUTH bit shall be set to zero if the device server has not authenticated the data or if the authentication has failed. The KEY DESCRIPTOR field shall contain the A-KAD value associated with the block.

A nonce value descriptor (see 8.5.4) shall be included if a nonce value was not generated by the device server (i.e., it was established by a nonce value descriptor included in the Set Encryption Key page use to set the key and algorithm used to encrypt the block.) or if the device server can not determine if the nonce was generated by the device server that encrypted the block. A nonce value descriptor may be included if the nonce value was generated by the device server that encrypted the block. The VALID bit shall be set to one. The AUTH bit shall be set to one if the device server has authenticated the nonce value. The AUTH bit shall be set to zero if the device server has not authenticated the nonce value or if the authentication has failed. The KEY DESCRIPTOR field shall contain the nonce value associated with the block

7.Y. DATA SECURITY OUT command

7.Y.1 DATA SECURITY OUT command description

The DATA SECURITY OUT (see table A2) is used to configure the data security methods in the device server and on the medium. The command supports a series of pages, only one of which may be sent with each command.

Table A2 –DATA SECURITY OUT command

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	Reserved			SERVICE ACTION				
2	PAGE CODE							
3	(MSB)	Reserved						
5								(LSB)
6	(MSB)	PARAMETER LIST LENGTH						
9								(LSB)
10	Reserved							
11	CONTROL							

The page code field (see Table B2) indicates the type of page that the application client is sending. If the page code field is set to a reserved or unsupported value, the device server shall 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.

Table B2 – PAGE CODE field values

Value	Description	Reference
00h – 0Fh	Reserved	
10h	Set data encryption page	7.Y.2
11h - FFh	Reserved	

The parameter list length field specifies the length in bytes of parameter data to be transferred from the application client to the device server. If the parameter length does not match the length of the specified page, the device server shall terminate the command with CHECK CONDITION status, with the sense key set to ILLEGAL REQUEST, and the additional sense code set to PARAMETER LIST LENGTH ERROR.

7.Y.2 Set data encryption page

Table Y1 shows the parameter list format of the set data encryption page.

Table Y1 – Set Data Encryption page

Bit / Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (10h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (m-3)						
3								(LSB)
4	SCOPE			Reserved		LOCK	CKOD	CKORL
5	DECRYPTION MODE				ENCRYPTION MODE			
6	ALGORITHM INDEX							
7	KEY FORMAT							
8	(MSB)	Reserved						
17								(LSB)
18	(MSB)	KEY LENGTH (n-19)						
19								(LSB)
20		KEY						
n								
n+1		KEY-ASSOCIATED DATA DESCRUPTORS						
m								

The page length field indicates the number of bytes of parameter data to follow. If the page length value results in the truncation of any field, the device server shall 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 PARAMETER LIST.

The SCOPE field (see table Y2) indicates the scope of the data encryption mode and key.

Table Y2 – SCOPE field values

Value	Name	Description
0	PUBLIC	The data encryption mode and key shall be ignored. The I_T nexus shall use values that are shared by other I_T nexuses. If no I_T nexuses are sharing values, the device server shall use default values.
1	LOCAL	The data encryption mode and key are unique to the I_T nexus associated with the DATA SECURITY OUT command and shall not be shared with other I_T nexuses.
2	RESERVATION GROUP	The data encryption mode and key shall be shared with all participants in a reservation.
3	ALL I_T NEXUS	The data encryption mode and key shall be shared with all I_T nexuses.
4 – 7		Reserved

The data encryption mode and key that shall be used for an I_T nexus shall be established by the following order of precedence:

1. If the scope for the I_T nexus is not PUBLIC, the values set by a DATA SECURITY OUT command associated with the I_T nexus; or
2. If the scope for the I_T nexus is PUBLIC:
    1) If the I_T nexus is participating in a reservation for the logical unit, the values set by another participant in the reservation with a scope of RESERVATION GROUP;
    2) the values set by anther I_T nexus with a scope of ALL I_T NEXUS; or
    3) the default values.

If the clear key on dismount (CKOD) bit is set the device server shall set the encryption key and encryption mode to default values after completing a dismount of a volume. If the CKD bit is set to zero, the dismounting of a volume shall not affect the encryption key or encryption mode. If the CKOD bit is set to one and there is no volume mounted in the device, the device server shall 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 PARAMETER DATA.

If the clear key on reservation loss (CKORL) bit is set the device server shall set the encryption key and encryption mode to default values on the loss or change in scope of the reservation. If the CKORL bit is set to zero, the loss of a reservation shall not affect the encryption key or encryption mode. If the CKORL bit is set to one and there is no reservation in affect for the I_T nexus associated with the DATA SECURITY OUT command, the device server shall 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 PARAMETER DATA.

Table Y3 defined the values for the ENCRYPTION MODE field.

Table Y3 – ENCRYPTION MODE field values

Value	Name	Description
0h	DISABLE	Data encryption is disabled.
1h	EXTERNAL	The data associated with WRITE(6) and WRITE(16) commands has been encrypted by a system that is compatible with the algorithm specified by the ALGORITHM INDEX field.
2h	ENCRYPT	The device server shall encrypt all data that it receives for a WRITE(6) or WRITE(16) using the algorithm specified in the ALGORITHM INDEX field and the key provided in the KEY field.
3h – Fh		Reserved

Table Y4 defined the values for the DECRYPTION MODE field.

Table Y4 – DECRYPTION MODE field values

Value	Name	Description
0h	DISABLE	Data decryption is disabled. If the device server encounters an encrypted block while reading, it shall not allow access to the block (see 4.2.19.3)
1h	RAW	Data decryption is disabled. If the device server encounters an encrypted block while reading, it shall pass the block and any additional metadata affixed to the block to the host without decrypting it (see 4.2.19.3)
2h	DECRYPT	The device server shall decrypt all data that is read from the medium in response to a READ(6) or READ(16) command or verifying when processing a VERIFY(6) or VERIFY(16) command. The data shall be decrypted using the algorithm specified in the ALGORITHM INDEX field and the key provided in the KEY field. If the device server encounters unencrypted data when processing a READ(6), READ(16), VERIFY(6), or VERIFY(16) command, the device server shall terminate the command with CHECK CONDITION status, with the sense key set to DATA PROTECT, and the additional sense code set to UNENCRYPTED DATA ENCOUNTERED WHILE DECRYPTING. The device server shall leave the medium positioned in front of the unencrypted block.
3h	MIXED	The device server shall decrypt all data that is read from the medium that it determines was encrypted in response to a READ(6) or READ(16) command or verifying when processing a VERIFY(6) or VERIFY(16) command. The data shall be decrypted using the algorithm specified in the ALGORITHM INDEX field and the key provided in the KEY field. If the device server encounters unencrypted data when processing a READ(6), READ(16), VERIFY(6), or VERIFY(16) command, the data shall be processed without decrypting.
4h – Fh		Reserved

If the ENCRYPTION MODE field is set to ENCRYPT and the key length field is set to zero, the device server shall 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 PARAMETER DATA.

If the DECRYPTION MODE field is set to DECRPT or MIXED and the key length field is set to zero, the device server shall 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 PARAMETER DATA.

The ALGORITHM INDEX field indicates which of the encryption algorithms reported by the DATA SECURITY IN command Data Encryption Capabilities pages shall be used to encrypt and decrypt data.

The KEY FORMAT field indicates the format of the value in the KEY field. Values for this field are described in table Y5.

Table Y5 – KEY FORMAT field values

Value	Description
00h	Key is in plain text
01h – FFh	Reserved

The KEY LENGTH field indicates the length of the key field in bytes.

The KEY field contains the encryption key to use when encrypting and decrypting data.

If the ENCRYPTION MODE field is set to ENCRYPT the device server shall save the key-associated descriptors in the in the KEY-ASSOCIATED DATA DESCRIPTORS field and assign them to every block that is encrypted by the device server. If more than one key-associated data descriptor is include in the page, they shall be in increasing numeric order of the value in the DESCRIPTOR TYPE field. If the ENCRYPTION MODE field is not set to ENCRYPT and key-associated descriptors are included in the KEY-ASSOCIATED DATA DESCRIPTORS field, the device server shall terminate the command with CHECK CONDITION, with the Sense Key shall be set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST.

An unauthenticated key-associated data descriptor (see 8.5.2) may be included if any unauthenticated key-associated data is to be associated with blocks encrypted with the algorithm and key. The VALID bit shall be set to one and the AUTH bit shall be set to zero. The KEY DESCRIPTOR field shall contain the U-KAD value associated with the block.

An authenticated key-associated data descriptor (see 8.5.3) may be included if any authenticated key-associated data is to be associated with the next block. The VALID bit shall and the AUTH bit shall be set to one. The KEY DESCRIPTOR field shall contain the A-KAD value associated with the block.

If a nonce value descriptor (see 8.5.4) is included and the algorithm and the device server supports application client generate nonce values, the value in the KEY DESCRIPTOR field shall be used as the nonce value for the encryption process. If the encryption algorithm or the device server does not support application client generated nonce values and a nonce value descriptor is included, the device server shall terminate the command with CHECK CONDITION, with the Sense Key shall be set to ILLEGAL REQUEST, and the additional sense code set to INVALID FIELD IN PARAMETER LIST. If the encryption algorithm or the device server requires an application client generated nonce value and a nonce value descriptor is not included, the device server shall terminate the command with CHECK CONDITION, with the Sense Key shall be set to ILLEGAL REQUEST, and the additional sense code set to INCOMPLETE KEY-ASSOCIATE DATA SET.

If a nonce value descriptor (see 8.5.4) is included, the VALID bit and the AUTH bit shall be set to one. The KEY DESCRIPTOR field shall contain the nonce value associated with the block

3.6 Additions to subclause 8

8.5 DATA SECURITY IN and DATA SECURITY OUT descriptors

8.5.1 DATA SECURITY descriptors overview

Several of the parameter pages in used by the DATA SECURITY IN and DATA SECURITY OUT commands allow for the inclusion of descriptors to provide additional optional data. This subslause defines the descriptors that are common between multiple pages.

8.5.1 DATA SECURITY descriptors format

Each DATA SECURITY descriptor shall take the form as defined in table D1.

Table D1 –DATA SECURITY descriptor format

Bit / Byte	7	6	5	4	3	2	1	0
0	KEY DESCRIPTOR TYPE							
1	Reserved						AUTH	VALID
2	(MSB)	KEY DESCRIPTOR LENGTH (n-3)						
3								(LSB)
4	(MSB)	KEY DESCRIPTOR						
n								(LSB)

The DESCRIPTOR TYPE field contains a value from table D2 that defines the contents of the DESCRIPTOR field

Table D2 – DESCRITOR TYPE field values

Value	Description	Reference
00h	Unauthenticated key-associated data	8.5.2
01h	Authenticated key-associated data	8.5.3
02h	Nonce value	8.5.4
03h – FFh	Reserved	

Use of the VALID field is defined by the page using the descriptor.

Use of the authenticated (AUTH) field is defined by the descriptor.

8.5.2 Unauthenticated key-associated data key descriptor

The AUTH field in an unauthenticated key-associated data descriptor shall be set to zero.

The KEY DESCRIPTOR field of an unauthenticated key-associated data descriptor shall contain any unauthenticated key-associated data assigned to the key.

8.5.3 Authenticated key-associated data key descriptor

The AUTH field shall be set as defined by the page in which the descriptor is included.

The KEY DESCRIPTOR field of an authenticated key-associated data descriptor shall contain any authenticated key-associated data assigned to the key.

8.5.4 Nonce value descriptor

The AUTH field in a nonce value descriptor shall be set to zero.

The KEY DESCRIPTOR field of a nonce value descriptor shall contain the nonce value used with the data encryption key.