To: INCITS T10/SSC-3 Committee
From: Matt Ball, M.V. Ball Technical Consulting, Inc.
Date: Sept 6, 2008
Subject: SSC-3: Using NIST AES Key-Wrap for Key Establishment

Revision History

Revision 0 (T10/08-155r0):
Posted to the T10 web site on Sept 6, 2008.

Related Documents

T10/SSC-3 r4a "SCSI Stream Commands 3"
NIST AES Key-Wrap (Draft, November 2001)
NIST FIPS 140-2 "Security Requirements for Cryptographic Modules"
NIST FIPS 140-2, Annex D "Approved Key Establishment Techniques"
NIST FIPS 140-2 IG "Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program" (January 2008)
NIST FIPS 197 "Announcing the Advanced Encryption Standard (AES)"
IEEE P1619.3 “Draft Standard for Key Management Infrastructure for Cryptographic Protection of Stored Data”

General

This is a proposal against SSC-3 r4a. At the discretion of the SSC working group, this could instead be a proposal against SSC-4.

The purpose of this proposal is to provide a way for the application client to pass a wrapped encryption key to the device server (note: this is different than passing an *encryption* key; the goal is to encrypt the encryption key). This feature has some of the following benefits:

- To comply with NIST FIPS 140-2, it is necessary to encrypt the key before passing it to the device server. NIST recently released updated Implementation guidance in January 2008 that no longer allows ESP (or similar protection) when entering keys into a cryptographic module (i.e., device server in T10 parlance)
- In many systems, it is desirable for a different entity to perform the key wrapping than the application client. By using a lightweight method such as AES Key Wrap, it becomes possible to more easily pass the key through different protocols. For example, in an IEEE P1619.3 system that includes an automation library, the library might talk to the key management server using P1619.3 and talk to a tape drive using SSC-3 or ADC. It is advantageous to only certify the key manager and tape drive under FIPS 140-2, which means the library cannot do any wrapping or unwrapping. It would just translate the protocols.

Currently, there are two FIPS 140-2 approved methods for key establishment (see <http://csrc.nist.gov/publications/fips/fips140-2/fips1402annexd.pdf>):

- Key Management using ANSI X9.17 (NIST FIPS 171)
- NIST AES Key-Wrap (See <http://csrc.nist.gov/groups/ST/toolkit/documents/kms/AES_key_wrap.pdf>)

The first method, FIPS 171, was withdrawn by NIST on February 8, 2005. This pretty much leaves us with 'AES Key-Wrap'. Accordingly, this proposal provides a method for using AES Key-Wrap to establish a key within a device server.

There are also methods allowed by FIPS 140-2IG, section 7.1.

The AES Key-Wrap algorithm requires the use of a shared *Key Encrypting Key* (KEK) for wrapping the key. Establishing this KEK is out of scope for this proposal.

Proposed Changes

Editor’s Note: These proposed changes apply to SSC-3 r4a (2008-01-28).

1 Scope

2 Normative references

3 Definitions, acronyms, keywords, and conventions

4 General Concepts

8.5.3.2.1 Set Data Encryption page overview

Table 141 shows the parameter list format of the Set Data Encryption page. (new changes are in **blue**)

Table 141 – Set Data Encryption page

Byte \ Bit	7	6	5	4	3	2	1	0
0	(MSB)			PAGE CODE (0010h)				
1								(LSB)
2	(MSB)			PAGE LENGTH (m-3)				
3								(LSB)
4	SCOPE			Reserved				LOCK
5	CEEM		RDMC		SDK	CKOD	CKORP	CKORL
6	ENCRYPTION MODE							
7	DECRYPTION MODE							
8	ALGORITHM INDEX							
9	KEY FORMAT							
10				Reserved				
17								
18	(MSB)			KEY LENGTH (n-19)				
19								(LSB)
20				KEY				
n								
n+1				KEY-ASSOCIATED DATA DESCRIPTORS LIST				
m								

The KEY FORMAT field indicates the format of the value in the KEY field. Values for this field are described in Table 147.

Table 147 – KEY FORMAT field values

Code	Description	Reference
00h	The KEY field contains the key to be used to encrypt or decrypt data.	8.5.3.2.2
01h	The KEY field contains a vendor specific key reference	8.5.3.2.3
	...	
04h	The KEY field contains a key encrypted using AES Key Wrap	8.5.3.2.6
05h – BFh	Reserved	
C0h – FFh	Vendor specific	

8.5.3.2.6 Logical block encryption key wrapped using AES Key Wrap

If the KEY FORMAT field of Table 147 is 04h, the KEY field contains a key that is wrapped with the AES Key Wrap algorithm, as specified by the NIST document 'AES Key Wrap Specification (16 Nov 2001)' (AES Key Wrap). Table Y shows the format of the KEY field if the KEY FORMAT field is set to 04h.

Table Y – KEY field contents with KEY FORMAT field set to 02h

Bit / Byte	7	6	5	4	3	2	1	0
0	(MSB)	KEK IDENTIFIER TYPE						
1								(LSB)
2	(MSB)	KEK IDENTIFIER LENGTH (m-4)						
3								(LSB)
4	(MSB)	KEK IDENTIFIER						
m-1								(LSB)
m	(MSB)	KEY WRAPPED WITH AES KEY WRAP						
n-1								(LSB)

The KEK IDENTIFIER TYPE field defines the format of the KEK IDENTIFIER field and shall contain a value from Table Z.

Table Z – KEK IDENTIFIER TYPE values

KEK IDENTIFIER TYPE	Description
0000h	Reserved
0001h	The KEK IDENTIFIER is a DS_SAI (see SPC-4)
0002h	The KEK IDENTIFIER is a value assigned by the device server
0003h – 7FFFh	Reserved
8000h – FFFFh	Vendor Specific

The KEK IDENTIFIER LENGTH field contains the length of the KEK IDENTIFIER field.

The KEK IDENTIFIER field contains a value that allows the device server to identify the key encrypting key (KEK) needed to unwrap the logical block encryption key. If the device server does not have a KEK that corresponds to the value in the KEK IDENTIFIER field, then 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 UNKNOWN KEK IDENTIFIER.

[NOTE: UNKNOWN KEK IDENTIFIER is a new ASC]

The KEK WRAPPED WITH AES KEY WRAP field contains a key encrypted by the AES Key Wrap algorithm (see AES Key Wrap) using the KEK identified by the KEK identifier in the KEK IDENTIFIER field and with the default initial value (IV) of A6A6A6A6A6A6A6A6h. The device servicer uses this KEK to invoke the Key Unwrap algorithm (see AES Key Wrap) with the KEY WRAPPED WITH AES KEY WRAP field as input. If the Key Unwrap algorithm succeeds, then the device server uses the resulting value as the key. If the Key Unwrap algorithm fails, then 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 KEY UNWRAP FAILED.

[NOTE: KEY UNWRAP FAILED is a new ASC]

If the length of the KEY WRAPPED WITH AES KEY WRAP field (i.e., n-m) is not a multiple of 8 bytes, then 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 SIZE FOR AES KEY WRAP.

[NOTE: INVALID SIZE FOR AES KEY WRAP is a new ASC]