Kanguru Solutions Kanguru Lock 1.0.4.25
Security Policy

Document Version 1.10

FIPS 140-2 Level 1 Validation

1 Introduction

This document is the Security Policy for Kanguru Solutions KanguruLock cryptographic module. This Security Policy specifies the security rules under which the module shall operate to meet the requirements of FIPS 140-2 Level 1. It describes how the module functions to meet the FIPS requirements, and the actions that operators must take to maintain the security of the module.

This Security Policy describes the features and design of the module using the terminology contained in the FIPS 140-2 specification. *FIPS 140-2, Security Requirements for Cryptographic Modules* specifies the security requirements that must be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information. The NIST Cryptographic Module Validation Program (CMVP) validates cryptographic modules to FIPS 140-2 and other cryptography-based standards. Validated products are accepted by the Federal agencies of both the USA and Canada for the protection of sensitive or protected information.

The FIPS 140-2 standard, and information on the CMVP program, can be found at http://csrc.nist.gov/cryptval. More information describing the KanguruLock application can be found at http://www.kanguru.com.

In this document, the KanguruLock application is also referred to as “the module”.

This Security Policy contains only non-proprietary information. All other documentation submitted for FIPS 140-2 conformance testing and validation is “Kanguru Solutions’ – Proprietary” and is releasable only under appropriate non-disclosure agreements.

The cryptographic module meets the overall requirements applicable to Level 1 security for FIPS140-2. The operating environment used in testing was Windows XP Service Pack 2.

Table 1. Cryptographic Module Security Requirements

Security Requirements Section	*Level*
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles and Services and Authentication	1
Finite State Machine Model	1
Physical Security	N/A
Operational Environment	1
Cryptographic Key Management	1
EMI/EMC	1
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A

1.1 Acronyms and Abbreviations

AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
CFB	Cipher Feedback
CMVP	Cryptographic Module Validation Program
CSE	Communications Security Establishment
CSP	Critical Security Parameter
DES	Data Encryption Standard
EDC	Error Detection Code
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FCC	Federal Communication Commission
FIPS	Federal Information Processing Standard
HMAC	Keyed-Hash Message Authentication Code
KAT	Known Answer Test
LAN	Local Area Network
NIST	National Institute of Standards and Technology
PUB	Publication
RAM	Random Access Memory
RFC	Request for Comment
ROM	Read Only Memory
RNG	Random Number Generator
RSA	Rivest Shamir and Adleman Public Key Algorithm
SHA	Secure Hash Algorithm

2 Cryptographic Module

2.1 Functional Overview

The module enables users to store data in encrypted format on a Kanguru Solutions' USB 2.0 flash drive device. Users can copy or drag and drop plaintext data files into the KanguruLock application that appears as a virtual drive in Windows Explorer. The KanguruLock application stores AES encrypted data on the virtual drive. It also decrypts data retrieved from the virtual drive. Users authenticate to the application by entering a password.

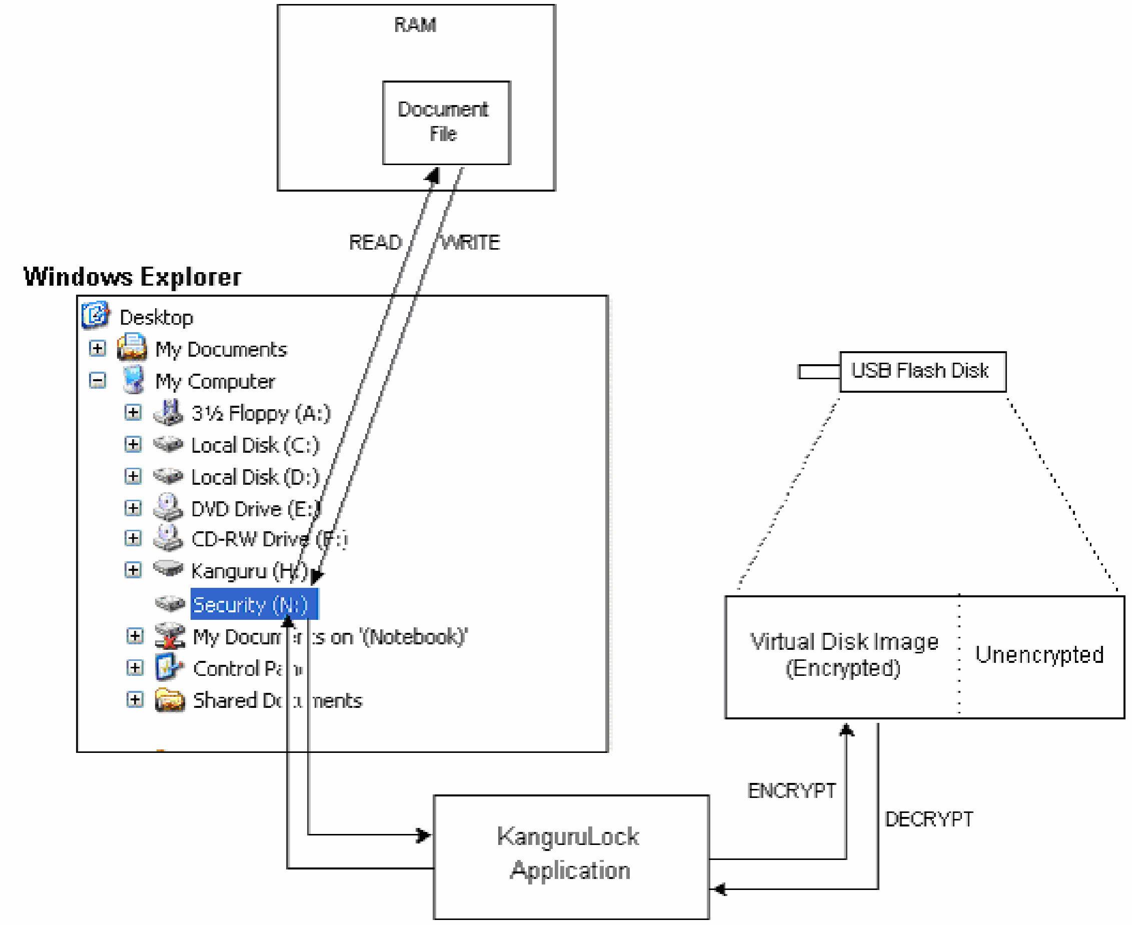
Features of the software include:

- Operator authentication
- AES (256-bit key) data encryption
- Invisibility of the ciphertext virtual drive to unauthenticated users

Figure 1 illustrates module operation. Once the KanguruLock application is installed, a user inserts the flash drive into a USB connector. The plaintext virtual drive (drive H: in this example) opens and is available. The ciphertext virtual drive (drive N: in this example) is unavailable until the user authenticates by entering the correct password in the module's password dialog box. On successful authentication, the module mounts the ciphertext virtual drive and provides data encryption and decryption services to the user. Users can also change the authentication password and regenerate the data encryption key as needed.

Figure 1. High Level Functional View of the Cryptographic Module.

.



2.2 Module Description

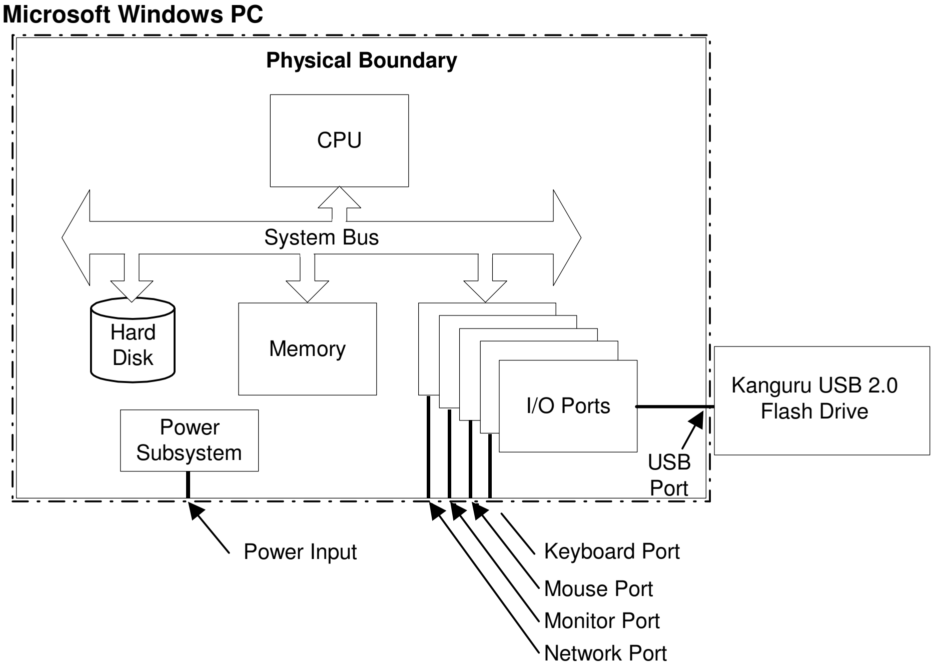
The module is a multi-chip standalone cryptographic module consisting of application software that executes on a general-purpose computing platform that is a Microsoft® Windows®-based PC, configured in single-user mode. The KanguruLock module stores AES-encrypted files off the module in a USB 2.0 flash drive system. The module uses a general purpose operating environment: Microsoft Windows XP. The module meets FIPS 140-2 level 1 security requirements.

The module provides authentication, cryptographic key management, and software integrity services assuring operators of a valid software state within the module and privacy services for the secure storage of data, cryptographic keys, and CSPs. The module does not have a bypass or maintenance mode.

2.3 High Level Block Diagram

Figure 2 shows a block diagram of the cryptographic module that illustrates the physical boundary of the module and shows the module physical interfaces. The physical cryptographic boundary is the physical boundary of the PC case.

Figure 2. High Level Block Diagram Showing Physical Boundaries.



Figures 3 shows a logical block diagram of the cryptographic module illustrating KanguruLock application components related to, and included within, the cryptographic boundary of the module.

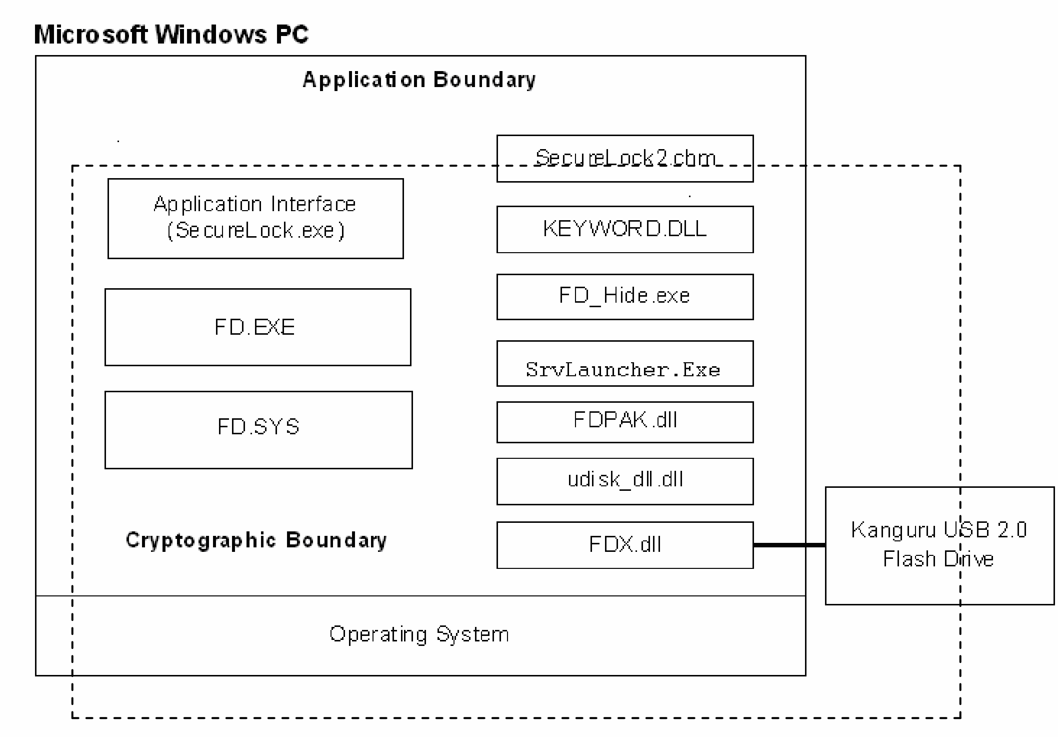
The cryptographic boundary includes all application files, as listed below:

- **Kanguru_Lock.exe** - KanguruLock user interface. Performs cryptographic and security functions.
- **FD.EXE** - Interface between application and drivers. Performs cryptographic and security functions.
- **FD.SYS** - Windows 2000/XP driver. Performs cryptographic and security functions.
- **KanguruLock2.chm** - KanguruLock online help file.
- **FD_Hide.exe** - Detects UFD insert/removal. On UFD insert, this runs KanguruLock to show the password dialog.
- **SrvLauncher.Exe** - Windows service program for Windows 2000/XP non-Administrator privileges.
- **FDPAK.dll** - Verifies whether hardware is functional.
- **udisk_dll.dll** - Accesses flash drive functions.

- **FDKey.dll** - Accesses flash drive functions and verifies that the hardware is functional.
- **FDX.dll** - Virtual disk core functions. It provides virtual disk operating functions consisting of create, mount, unmount, resize, and get virtual disk status.

The following files are excluded from requirements of FIPS 140-2 as not security relevant: KanguruLock2.chm, FD_Hide.exe, • SrvLauncher.Exe, FDPAK.dll, udisk_dll.dll, FDKey.dll, and FDX.dll.

Figure 3. High Level Block Diagram Showing Logical and Cryptographic Boundaries.



2.4 Module Ports and Interfaces

The cryptographic module has 10 physical interfaces and four logical interfaces. The physical ports have the functions described in Table 3. Where distinct logical interfaces share the same physical port, the system timing, software and hardware protocols, software APIs, and other controls logically separate and isolate these distinct categories of data from one another. The internal system bus acts as the physical path for clocking data into and out of the module. System synchronization and timing controls ensure that logically distinct categories of data do not occupy the data path at the same time.

Table 3. Physical Interfaces and Logical 140-02 Interfaces.

Physical Interface	*FIPS 140-2 Logical Interface*
PC USB port, PC network port, keyboard interface, mouse port, hard drive, floppy drive, CDROM drive	Data input interface
PC USB port, PC network port, keyboard interface, mouse port, hard drive, floppy drive, CDROM drive	Data output interface
PC keyboard port, mouse port, PC power button	Control input interface
PC monitor	Status output interface
PC power interface	Power interface

The physical interfaces map to logical interfaces as described in Table 4.

Table 4. FIPS 140-2 Logical Interfaces.

Logical Interface	*Description*
Data input	The data input is: • All plaintext data entering the KanguruLock application for the purpose of being encrypted and stored on an external USB flash drive. Components providing this interface are FD.EXE and FD.SYS. • All ciphertext data entering the KanguruLock application from the external USB flash drive for the purpose of being decrypted. Components providing this interface are FD.EXE and FD.SYS.
Data output	The data output is: • All plaintext data exiting the KanguruLock application. Components providing this interface are FD.EXE and FD.SYS. • All ciphertext data exiting the KanguruLock application for storage on an external USB flash drive. Components providing this interface are FD.EXE and FD.SYS.
Control input	The KanguruLock application accepts control input from the operator. Control input consists of all commands and command parameters. Control input commands include resizing the partition, changing the password, and locking and unlocking the module.
Status output	The status output consists of all messages either logged by the module or returned by the module, and all status data obtained as a result of user commands. The error messages from the USB flash device are also status output. To view the logged error messages, use the log viewer program. Other error messages may be output to the display.

3 Security Functions

The KanguruLock cryptographic module implements the security functions described in Table 5:

Table 5. Module Security Functions.

Approved Security Function	*Certificate*
Symmetric Key Encryption	
AES (FIPS PUB 197) ECB (e/d; 256)	243
Hashing	
SHS byte-oriented hashing. FIPS PUB 180-2	321
Random Number Generation	
Used for key generation. FIPS 186-2 Appendix 3.1(SHA-1)	78
Keyed-Hash Message Authentication Code (HMAC)	
Used for integrity checking. FIPS Pub 198	51
Non-Approved Security Function	
Password Based Key Derivation (for AES key encryption/decryption)	N/A

The module provides AES symmetric key encryption / decryption of data and of the AES storage key, random number generation, and SHA-1 hashing (for deriving the storage key from the user password). Password-based key derivation is a non-Approved algorithm. The AES key that is encrypted using the password-derived key is considered plaintext for the purposes of FIPS 140-2.

4 FIPS Approved Mode of Operation

The module's Approved mode of operation is restricted to performing only FIPS-approved cryptographic algorithms and security functions. The module has a non-Approved mode and an Approved mode. The non-Approved mode, active once the module powers up, lets users read and write data only to the plaintext partition of the virtual drive. The module enters the Approved mode when the operator enters the correct password. In the Approved mode, the user may read and write data to either the plaintext or the ciphertext partition of the virtual drive.

The module must run on a Windows XP operating system configured in single-user mode.

To set the module into the Approved mode, the crypto officer must properly install and configure the software and the user must adhere to the requirements for operating in the Approved mode.

The crypto officer must perform the following steps to install the cryptographic module and confirm the correct software version:

1. Insert the drive into an available USB 2.0 slot. The Kanguru Drive (usually H:) opens in Windows Explorer.
2. Open the Kanguru Drive (usually H:) in Windows Explorer.
3. Click **KanguruLock_setup.exe** in the Kanguru Drive. The KanguruLock Setup dialog appears.
4. Select the Kanguru Drive if it is not already selected.
5. Select or clear the checkbox **Use NTFS File System** as appropriate for your system.
   Click **OK**.

   A dialog may appear saying "This computer already includes a Security Drive. Would you like to continue using the existing drive? Click **No** to create a new drive and start formatting it." Click **No**. The program creates a new formatted drive and displays a dialog "Kanguru Setup completed." Click **OK** to continue.

   The program installs some files and prompts "Would you like to have a password hint? Click no to disable password hint."
6. Click **No** to disable the password hint as this weakens the module's strength of authentication. The program configures installed files and prompts "Setup completed."
7. Click **OK**. The KanguruLock program screen opens. Only the Setup Menu tab is visible.
8. Click **Mount**. A password dialog appears.
9. Enter the default password "KanguruAES" and click **OK**. A prompt says "Security Drive mounted."
10. Click **OK**. The KanguruLock program screen reappears with Change Password and Setup tabs. Click **Change Password**. The Password Change dialog appears.
11. Enter the default password into the Old Password field and a new password into the New Password field. The new password must be a minimum of 6 characters in length and include at least one upper case, one lower case, one symbol (~!@#$%^&*()_+) and one number.
12. Reenter the new password into the Confirm Password field and click **OK**. A dialog appears asking "Do you want to change the AES key? This process may take a long time. Please wait…"
13. Click **Yes** to change the AES key. A prompt appears saying "Change AES key completed."
14. Click **OK**. The KanguruLock program screen reappears.

15. Confirm the installed components are the FIPS validated versions by right-clicking the following files, choosing **Properties**, and viewing the version number in the Version tab of the Properties dialog.

    On Windows XP systems, check these files:

Filename
KanguruDefenderDrive:\KanguruLock.exe
C:\WINDOWS\system32\drivers\FD.sys
C:\WINDOWS\system32\drivers\Kanguru_SAP\FD.exe

At this point secure installation is completed, the password hint is disabled, a new password has been set and the AES key has been changed. The module is in the FIPS Approved mode.

Module users must protect the password from discovery by others. Users must not write the password and leave it where others can find it. They should ensure no one is watching when entering the password.

5 Identification and Authentication

The module supports a crypto officer role and a user role. The crypto officer and user may be different people or they may be the same person performing role-specific module operations.

The crypto officer role is implicitly assumed by the operator configuring the module for use at installation time. Crypto officer operations consist of configuring the PC in single user mode and running the setup program.

The user role becomes available once the user takes possession of the module by changing the default password. Approved mode operations available to the user include changing the password, using the module data encryption and decryption services, and storage key generation.

Multiple concurrent operators are not allowed as the module is restricted to single user mode. Operators cannot change roles while authenticated to the module. The module does not display the password entered into the module. Users must disable the hint feature for FIPS Approved mode operation. Access to the authorized roles is restricted as explained in Table 6.

Table 6. Roles and Required Identification and Authentication.

Role	*Type of Authentication*	*Authentication Data*
Crypto Officer	Role-based	The crypto officer authenticates to the Microsoft Windows XP operating system before installing and configuring the module for use. Once a user has taken possession of the module by changing the default password, the crypto officer role becomes unavailable.
User	Role-based	An operator must enter the correct password to assume the user role.

The module does not require any physical maintenance. The strength of the operator authentication, per the above roles, is as follows in Table 7:

Table 7. Strength of Authentication.

Authentication Mechanism	*Strength of Mechanism*
Password	Users must authenticate using a password that is at least 6 characters and at most 31 characters. The characters used in the password must be from the ASCII character set of 0x20~7E This yields a minimum of 95^6 (over 735 billion) possible combinations; thus, the possibility of correctly guessing a password is less than 1 in 1,000,000. The possibility of randomly guessing a password in 60 seconds is less than 1 in 100,000. Key derivation from the password takes about .1 seconds to complete, limiting the total number of attempts to guess the password within 60 seconds to about 600.

When the cryptographic module is powered off and subsequently powered on, the results of previous authentications (the decrypted Storage Key) is cleared from memory. When the module is powered up again, the operator must re-authenticate, entering the correct password to derive the storage key encryption key.

6 Cryptographic Keys and CSPs

The following table identifies the Cryptographic Keys and Critical Security Parameters (CSPs) employed within the module. All keys inside the logical boundary are stored in RAM and can be zeroized by powering off the PC.

Table 8. Cryptographic Keys and CSPs.

Data Item	*Description*
Storage Key (SK)	The SK is a 256-bit AES key that is generated anew according to FIPS PUB 186-2 Appendix 3.1 whenever the module partition is set or changed. The key is stored encrypted using password-based encryption (which is considered plaintext for the purposes of FIPS 140-2) within the header portion of the virtual disk image, located on the USB drive. Each time the storage key is generated, it is exported from the physical boundary of the module encrypted using password-based encryption and then stored on the USB drive within the header portion of the virtual disk image. Therefore, the key is exported in password-based encrypted form, which is considered plaintext for the purposes of FIPS 140-2. Whenever the module transitions from the locked state (ciphertext partition is not accessible) to the unlocked state (ciphertext partition is accessible) the SK is imported into memory for use in data encryption and decryption. Whenever the module transitions from the unlocked state to the locked state, the SK in memory is destroyed by zeroization (overwriting the memory location with ‘0’ characters).
HMAC Secret Key	An HMAC secret key (128 bits) is maintained in the software image. The key is created off the module during the manufacturing process and is stored in the software `image when the module is installed on a personal computer. This key is used for the software integrity test.. After installation, the key is stored in plaintext form within the

Data Item	Description
	header portion of the virtual disk image.
Key Encryption Key (KEK)	The KEK is an ephemeral 256-bit AES key that is derived by a SHA-1 hash of the password that is entered by the module operator for the module unlock operation. As password-based key derivation is a non-approved key generation method, anything encrypted using this key is considered plaintext within FIPS 140-2. This operation is used for authentication purposes only. The KEK is not stored but is used once to decrypt the SK within the module unlock operation. When the module unlock operation exits, the KEK is destroyed by zeroization (overwriting the memory location with '0' characters).
Password	The password is a 6 to 31 character password that is used to authenticate the user. The password derives an AES key that encrypts the SK. If the password-derived key successfully decrypts the SK, authentication is successful. The password is not stored within the system.
FIPS 186-2 RNG seed	The RNG seed is generated by the Microsoft Crypto API. It is not required to use a FIPS validated version off the Microsoft Crypto API.
FIPS 186-2 RNG seed key	The RNG seed key is generated by the Microsoft Crypto API. It is not required to use a FIPS validated version off the Microsoft Crypto API.

7 Roles and Services

The module supports services that are available to crypto officers and users. All of the services are described in detail in the module's user documentation.

Anyone having access to the module can read or write plaintext data to the plaintext partition of the virtual drive without entering authorization data.

The crypto officer role (unauthenticated) accesses the module initialization service, available only when the KanguruLock application is installed on a PC platform but not yet initialized. The crypto officer role becomes unavailable after the module is initialized.

The user role becomes available after the operator assumes ownership by changing the password from the default password supplied with the system. The operator must enter the correct password to assume the User role. An authenticated user accesses all module services (except initialization).

Table 9 shows the services available to the various roles.

Table 9. Roles and Services

Service	Crypto Officer / Owner	User
Install the module	X	
Generate Storage Key		X
Change Password		X
Read/Write encrypted files and directories		X
View version information		X

Service	Crypto Officer / Owner	User
Run Self-Tests		X
Show Status		X
Zeroize Keys		X

The services are described below:

Install the module - perform initial installation steps for the module.

Generate SK – generate (or re-generate) a new Storage Key. The key is then exported and stored on the USB drive outside of the physical boundary of the module. The key is encrypted using password-based encryption, which is not a FIPS 140-2 approved encryption method. Therefore, the key is considered exported in plaintext for the purposes of FIPS 140-2.

Read/Write encrypted files and directories – reads or writes information from or to the USB drive encrypting file contents on write and decrypting file contents on read.

View version information – outputs application version information

Run Self-Tests – runs self-tests on demand by restarting the module.

Change Password - changes the password. The Storage Key stored on the USB drive will be re-encrypted using password-based encryption with the new password, and then exported to the USB drive.

Show Status - shows status of the module. In an error state, the module displays a message box specifying the error that occurred, and then exits. If self-tests pass successfully, the module shows a secure drive image in Windows Explorer.

Zeroize Keys – all keys inside the logical boundary are stored in RAM and are zeroized by powering off the PC.

8 Access Control

Table 10 shows services that use or affect cryptographic keys or CSPs. For each service, the key or CSP is indicated along with the type of access.

R - The item is **read** or referenced by the service.
W - The item is **written** or updated by the service.
E - The item is **executed** by the service. (The item is used as part of a cryptographic service.)
D - The item is **deleted** by the service.

Table 10. Access Control.

Authentication Data (Key or CSP)	Service	Access Control
Key Encryption Key (KEK)	Unlock	R,E,D
	Change Password	R,E,D
	Resize Virtual Drive	R,E,D
Storage Key (SK)	Unlock	R,E
	Change Password	R,E,D,W
	Resize Virtual Drive	R,E,D,W
	Install New Virtual Drive	R,E,D,W
	Read/Write ciphertext data	R,E

Authentication Data (Key or CSP)	Service	Access Control
	Lock	D
Password	Unlock	R,E,D
	Change Password	R,E,D
	Resize Virtual Drive	R,E,D

9 Self Tests

The module performs both power-on self test (POST) and conditional self tests to verify the integrity and correct operational functioning of the cryptographic module. If the system fails a self test, it reports status indicating which failure occurred and transitions to an error state. The C++ functions that run the self-test do not include any calls to output data and this blocks all traffic on the data ports, preventing use of any cryptographic keys, CSPs, cryptographic algorithms, and security functions except as needed by the self tests. If self-tests fail, the module displays a message box specifying the error that occurred, and then exits. If self-tests pass successfully, the module shows a secure drive image in Windows Explorer.

To indicate failure of self-tests, the module issues a Windows API command to display an error message box containing a description of the error. To indicate success of self-tests, the module issues a Windows API command to display a secure drive image in Windows Explorer.

The user can run self-tests on demand by using the Run Self Tests service, which amounts to restarting the module. Table 11 summarizes the system self tests.

Table 11. Self Tests.

Self Test	*Description*
Mandatory power-up tests performed at power-up and on demand:	
Cryptographic Algorithm Known Answer Tests	Each cryptographic algorithm (AES, SHA-1, HMAC, and RNG) performed by the module, is tested using a "known answer" test to verify the operation of the function. The AES known answer performs both encryption and decryption.
Software Integrity Test	The module verifies the integrity of the software by generating an HMAC message authentication code for Secure_Lock.exe, FD.EXE, and FD.SYS and comparing the code against the expected values stored in the registry.
Conditional tests performed, as needed, during operation:	
Continuous RNG	This test checks the RNG output data for failure to a constant value.

Known answer tests for encryption/decryption or hashing, function by encrypting (or hashing) a string for which the calculated output is known and stored within the cryptographic module. An encryption or hashing test passes when the freshly calculated output matches the expected (stored) value. A test fails when the calculated outmatch does not match the expected value. The test then decrypts the ciphertext string. A decryption test passes when the freshly calculated output matches the plaintext value. A test fails when the calculated output does not match the plaintext value.

Known answer tests for Random Number Generators function by seeding the RNG with known values and checking that the output matches the pre-calculated value stored within the cryptographic module. The test passes when the freshly generated output matches the pre-calculated value. A test fails when the generated outmatch does not match the pre-calculated value.

10 References

National Institute of Standards and Technology, *FIPS PUB 140-2: Security Requirements for Cryptographic Modules*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex A: Approved Security Functions*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex B: Approved Protection Profiles*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex C: Approved Random Number Generators*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex D: Approved Key Establishment Techniques*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology and Communications Security Establishment, *Derived Test Requirements (DTR) for FIPS PUB 140-2, Security Requirements for Cryptographic Modules*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *Data Encryption Standard (DES)*, Federal Information Processing Standards Publication 46-3, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *DES Modes of Operation*, Federal Information Processing Standards Publication 81, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *Digital Signature Standard (DSS)*, Federal Information Processing Standards Publication 186-2, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *Secure Hash Standard (SHS)*, Federal Information Processing Standards Publication 180-1, available at URL: http://www.nist.gov/cmvp.