August 23, 2016

SPYRUS MDTU-P384 Encryption Module
Non-Proprietary Security Policy

Revision Document No. 0.7

Document number 599-430007-06

Contents

I. Introduction

This Security Policy specifies the security rules under which the SPYRUS MDTU-P384 (Media Data Transport Unit) Cryptographic Module operates. Included in these rules are those derived from the security requirements of FIPS 140-2 and additionally, those imposed by SPYRUS, Inc. These rules, in total, define the interrelationship between:

1. Operators,
2. Services, and
3. Critical Security Parameters (CSPs).

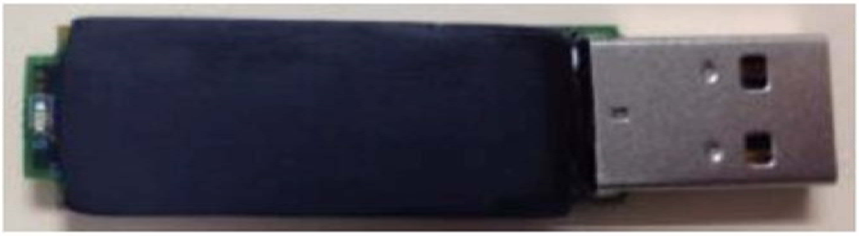


Figure 1 SPYRUS MDTU-P384 Encryption Module (Topside)

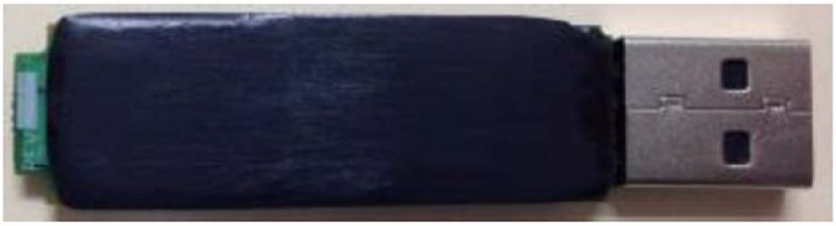


Figure 2 SPYRUS MDTU-P384 Encryption Module (Underside)

1.1 SPYRUS MDTU-P384 Encryption Module Overview

The SPYRUS MDTU-P384 Encryption Module enables security critical capabilities such as operator authentication and secure storage in rugged, tamper-evident hardware. The SPYRUS MDTU-P384 Encryption Module communicates with a host computer via the USB interface. The SPYRUS MDTU-P384 Encryption Module protects data for government, large enterprises, small organizations, and home users. Key features:

- Encryption technology uses Suite B algorithms approved by the U.S. government for protecting both Unclassified and Classified data
- Encrypted file storage on non-removable flash storage
- Strong protection against intruder attacks

Access protection is as important as encryption strength. Data encrypted with the SPYRUS MDTU-P384 Encryption Module cannot be decrypted until the authorized user gains access to the device.

The SPYRUS MDTU P-384 module is composed of three major Hardware modules:

- Arm Processor
- High Speed Symmetric crypto engine
- SPYRUS SPYCOS 3.0 security processor

1.2 SPYRUS MDTU-P384 Encryption Module Environmental Range

The SPYRUS MDTU-P384 Encryption Module operates in the following temperature range: -20 degrees C. to 65 degrees C.

1.3 SPYRUS MDTU-P384 Encryption Module Implementation

The SPYRUS MDTU-P384 Encryption Module is implemented as a multi-chip standalone module as defined by FIPS 140-2. The FIPS 140-2 module identification data for the SPYRUS MDTU-P384 Encryption Module is shown in the table below:

Part Number	FW Version	HW Version
880074014F	03.00.0D	2.00.02

The SPYRUS MDTU-P384 Encryption Module is available with a USB interface compliant to the *Universal Serial Bus Specification*, Revision 2.0, dated 23 September 1998. All Interfaces have been tested for compliance with FIPS 140-2. Version information shown above can be verified per instructions in the “MDTU P-384 User Guide” in the section entitled “About P-384”. See also Figure 4.

1.4 SPYRUS MDTU-P384 Encryption Module Cryptographic Boundary and Tamper Inspection

The Cryptographic Boundary is defined to be the outer perimeter of the hard, opaque epoxy potting. Please see Figure 3.

The operator detects physical attacks against the module by direct physical inspection. If the module is packaged in a plastic case or similar outer coating that is not inside the cryptographic boundary, any sign of entry, cracking, breakage or damage to the case due to prying or forcing using a sharp tool may require further inspection to confirm whether a penetration attack has taken place on the module's epoxy coating. The epoxy coating will either show tamper evidence or not. If it shows tamper evidence, the module has been compromised and the operator must treat the device in accordance with organizational security policy. This would include issuance of a new device. If it does not show tamper evidence, the operator may continue to use the device in accordance with organizational security policy.

All hardware, firmware, or software components that comprise the SPYRUS MDTU-P384 Encryption Module are included within the scope of the validation and satisfy the requirements of FIPS 140-2.

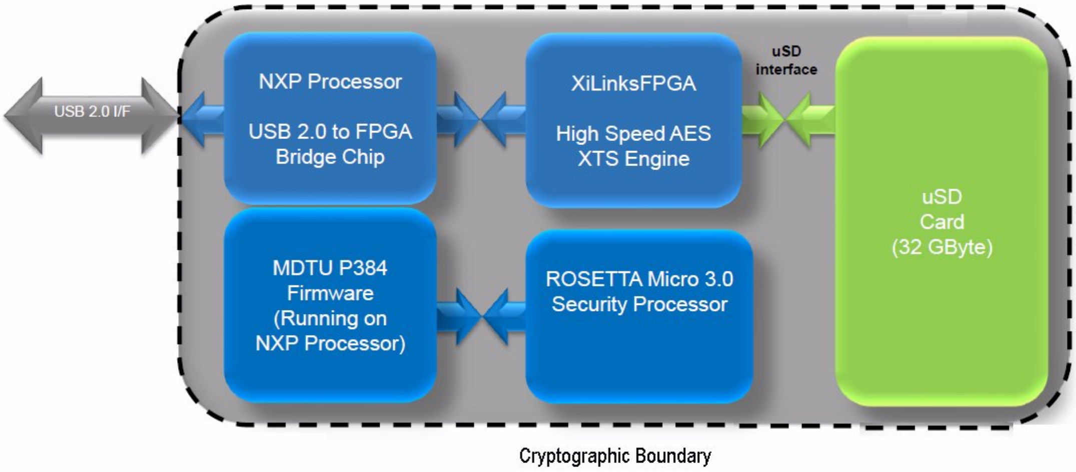
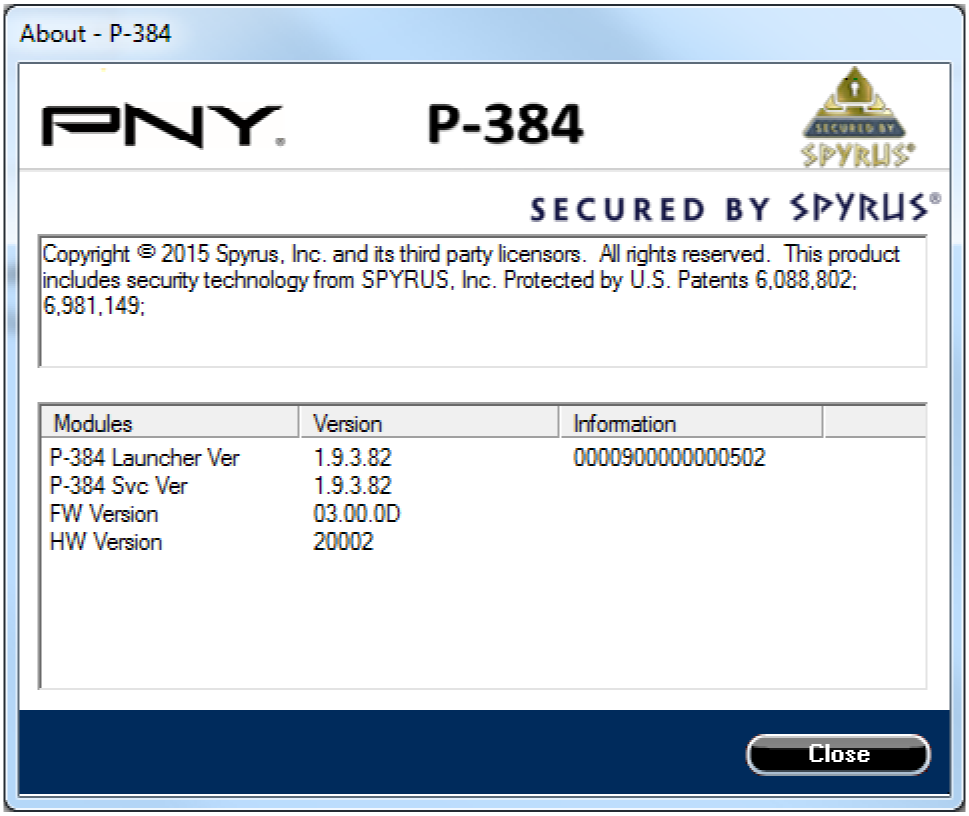

USB 2.0 I/F
NXP Processor
USB 2.0 to FPGA Bridge Chip
XiLinksFPGA
High Speed AES XTS Engine
uSD interface
uSD Card (32 GByte)
MDTU P384 Firmware (Running on NXP Processor)
ROSETTA Micro 3.0 Security Processor
Cryptographic Boundary

Figure 3 SPYRUS MDTU-P384 Block Diagram and Cryptographic Boundary

1.5 Approved Mode of Operations

The SPYRUS MDTU-P384 Encryption Module operates only in a FIPS Approved mode. The on-demand indicator that shows the operator that the module is in the approved mode is the ”About P-384” Window, which shows the module’s firmware and hardware versions as well as the product indicator (Figure 4). Instructions for viewing this information are found in the MDTU P-384 User Guide.



Copyright © 2015 Spyrus, Inc. and its third party licensors. All rights reserved. This product includes security technology from SPYRUS, Inc. Protected by U.S. Patents 6,088,802; 6,981,149;

Modules	Version	Information
P-384 Launcher Ver	1.9.3.82	0000900000000502
P-384 Svc Ver	1.9.3.82	
FW Version	03.00.0D	
HW Version	20002	

Figure 4 "About P-384" Window indicating Approved Mode of Module

Table 1-1 Approved Algorithms Supported by the SPYRUS MDTU-P384 Encryption Module

Encryption & Decryption
AES- ECB and CBC modes 128/192/256 (Cert. #3877); and FPGA (Cert. #3878)
AES - XTS 128/256 (Cert. 3877); and FPGA (Cert. #3878)
Digital Signatures
ECDSA – PKG, PKV, Sig Gen, Sig Ver with curves: 256, 384, 521 (Cert. #837)
Hash
SHA-224, SHA-256, SHA-384, SHA-512 (Cert. #3198); FPGA (Cert. #3199)
Key Agreement
KAS (Cert. #75) P-256, P-384, P-521; SHA-256, SHA-384, SHA-512
DRBG
HASH_DRBG (SP 800-90A) (Cert. #1106)

The module supports the following non-Approved but allowed algorithms for use within the FIPS Approved mode of operation:
- NDRNG (for seeding the Approved DRBG).

2 FIPS 140-2 Security Levels

The SPYRUS MDTU-P384 Encryption Module cryptographic module complies with the requirements for FIPS 140-2 validation to the levels defined in Table 2.1. The FIPS 140-2 overall rating of the SPYRUS MDTU-P384 Encryption Module is Level 3.

Table 2-1 FIPS 140-2 Validation Levels

FIPS 140-2 Category	Level
1. Cryptographic Module Specification	3
2. Cryptographic Module Ports and Interfaces	3
3. Roles, Services, and Authentication	3
4. Finite State Model	3
5. Physical Security	3
6. Operational Environment	N/A
7. Cryptographic Key Management	3
8. EMI/EMC	3
9. Self-tests	3
10. Design Assurance	3
11. Mitigation of Other Attacks	N/A

3 Ports and Interfaces

The pin configuration of the SPYRUS MDTU-P384 Module's USB physical receptacle interface is shown in Figure 5. The standard USB 2.0 pins form a set of 4 active contact points that comprise the physical ports of the cryptographic module. Table 3-1 shows the mapping of the pins to their functional description and logical interface description. The module also includes an LED for status output.

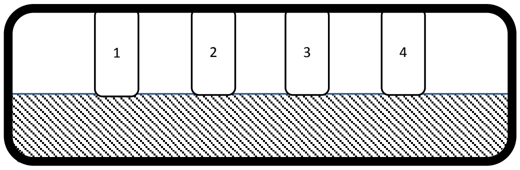


Figure 5 SPYRUS MDTU-P384 Module USB 2.0 Interface, End View, Pins 1 - 4

Table 3-1
SPYRUS MDTU-384 Module Pins and Logical Interfaces

#	Pin	Function	FIPS 140-2 Logical Interface
1	V_{BUS}	Operating voltage	Power Interface
2	D-	USB 2.0 Data Input/ Output (half-duplex)	Data Input / Data Output; Control Input; Status Output
3	D+	USB 2.0 Data Input / Output (half-duplex)	Data Input / Data Output; Control Input; Status Output
4	GND	Ground for power return	Power Interface

4 SPYRUS MDTU-P384 Encryption Module Roles and Services

4.1 Roles

The SPYRUS MDTU-P384 Encryption Module supports three roles, Administrator (Crypto Officer), User, and DDS and enforces the separation of these roles by restricting the services available to each one. Each role is associated with a single user identity, namely the service that has been requested and is associated with the role.

Table 4-1 Roles and Responsibilities

Role	Responsibilities
Administrator	The Administrator is responsible for performing Firmware Updates and setting configuration of the SPYRUS MDTU-P384 Encryption Module (HPC140-F). The SPYRUS MDTU-P384 Encryption Module validates the Administrator identity by way of a signature before accepting any FirmwareUpdate or SetConfiguration commands.
User	The User role is available after the SPYRUS MDTU-P384 Encryption Module has been initialized. The user can load, generate and use secret keys for encryption services.
DDS	Digital Data Set (DDS) Role is responsible for challenge response logon to the SPYRUS MDTU-P384 Encryption Module, mounting of drive and acquisition of flight plan and related data assets

The SPYRUS MDTU-P384 Encryption Module validates the User identity by password before access is granted. DDS access is further described in Section 5.

4.2 Services

The following table describes the services provided by the SPYRUS MDTU-P384 Encryption Module.

Table 4-2 SPYRUS MDTU-P384 Encryption Module Services

Service	CO	User	DDS	Description
ChangePassword		X		Changes User Password.
Erase_User_Area		X		Erases a specified user area. Writes zeros at the specified offset for the specified length.
Format		X		Formats the mounted CDROM.
GetCapabilities	X	X		Returns the current capabilities of the system including: global Information, Sector storage size and the product name. This service provides a response that indicates the approved mode of operation (see Section 1.5).
GetConfigurations	X	X		Returns the card configuration structure.
GetUserState	X	X		Returns the state and the Logon attempts remaining.
Initialize		X		Generates a new encryption key and changes the PIN. Secure channel is required. Formats the media.
LogOff		X		Log Off; Return to unauthenticated state.
LogOn		X		Log on with the user PIN if system is initialized.
MountCDROM		X		Allows the CDROM drive to be mounted as the read/write drive. This permits the CDROM software to be updated by a user application.
ReadMedia		X		Read user media from SCSI drive.
ReadUserArea	X	X		Get a block of data from a specified user area.
SetupBasicSecureChannel	X	X		Used prior to initialization of the module and its purpose is to set up the secure channel.
SetConfig	X			Writes the card configuration structure if the signature on the structure is valid.
UpdateFirmware	X			Writes signed blocks to the firmware area of the module.
WriteMedia		X		Writes user media to SCSI drive.
WriteUserArea		X		Write a block of data to a specified user area. All areas will require the token to

Service	CO	User	DDS	Description
				be logged on for writes and updates.
Zeroize	X	X		Clears the encryption keys. Requires the Initialize command to be run again.
Inquiry			X	SCSI pass-through command to read the Inquiry Data.
Authentication and Mount			X	SCSI pass-through command to send the calculated DDS Authentication Value to the Module.
Test Unit Ready			X	SCSI pass-through command to detect if the media state has changed.
Set or Change DDS Special Value			X	SCSI pass-through command used to change the 64 byte Security Value used in the challenge response exchange.

Note: the Inquiry, Authentication and Mount, Test Unit Ready and Set or Change DDS Special Value commands are used by the DDS to complete authentication to the module by a challenge response mechanism. This is described in more detail in Section 5.2

The following services are unauthenticated, i.e., they can be invoked without a prior operator authentication:

- GetCapabilites
- SetupBasicSecureChannel
- GetConfigurations
- GetUserState
- ReadUserArea
- SelfTest

5 Identification and Authentication

5.1 Initialization Overview

The SPYRUS MDTU-P384 Encryption Module modules are initialized at the factory to be in the zeroized state. Before an operator can access or operate a SPYRUS MDTU-P384 Encryption Module, the User must first initialize the module with a User ID and PIN.

5.2 Operator Authentication

Operator Authentication is accomplished by PIN entry by the User or valid ECDSA signature by the CO. Once valid authentication information has been accepted, the SPYRUS MDTU-P384 Encryption Module is ready for operation. DDS authentication is described below.

The SPYRUS MDTU-P384 Encryption Module stores the number of operator logon attempts in non-volatile memory. The count is reset after every successful entry of a User PIN. If an incorrect PIN is entered during the authentication process, the count of unsuccessful logon attempts is incremented by one.

If the operator fails to log on to the SPYRUS MDTU-P384 Encryption Module in 10 consecutive attempts, the SPYRUS MDTU-P384 Encryption Module will block the user's access to the module, by transitioning to the blocked state. To restore operation to the SPYRUS MDTU-P384 Encryption Module, the User will have to zeroize the token and reload the User PIN and optional details. When the SPYRUS MDTU-P384 Encryption Module is inserted after zeroization, it will power up and transition to the Zeroized State, where it can be initialized.

The SPYRUS MDTU-P384 interface with the DDS is authenticated by a separate challenge response mechanism that utilizes SCSI command data block (CDB) commands. The SPYRUS MDTU-P384 supports the USB Mass Storage interface as a USB wrapper around the standard SCSI commands. A SCSI driver directly sends commands to a SCSI device regardless of the actual interface. The necessary USB packets are handled by a driver with a SCSI interface.

The SCSI command sequence for authentication of a DDS to the SPYRUS MDTU-P384 follows below.

Inquiry Command

A specially formatted Inquiry Command is sent by the OS automatically. The DDS operator implements a SCSI pass-through command to read the Inquiry Data. This is used to calculate the response to the challenge embedded in the Inquiry Data.

Authentication and Mount Command

The Authentication and Mount Command is used by the DDS to send a calculated response to the challenge that was received by way of the Inquiry Command. The DDS uses the Device Shared Secret in this calculation along with challenge information from the module's Inquiry Command. If the response is correct, Authentication and Mount Command succeeds and the DDS is granted access to the module. The Authentication and Mount Command will fail if the authentication response is not correct, and the module will be disconnected as a logical unit number, or LUN, i.e., a number used to identify a logical unit. A LUN is a device addressed by the Small Computer System Interface (SCSI) protocol.

Test Unit Ready Command

The DDS is required to send this command to both LUNS periodically to detect if the media state has changed. If this is not being done by the OS then the DDS will need to several of these to each LUN in order for the MDTU-P384 to successfully mount the drives.

Set or Change DDS Special Value Command

This command is a vendor specific SCSI command and is used by the DDS to change the 64 byte Special Value used in the challenge response exchange.

5.3 Generation of Random Numbers

The Random Number Generators are not invoked directly by the user. The Random Number output is generated by the HASH-DRBG algorithm specified in SP 800-90A in the case of static private keys and associated key wrapping keys, ephemeral keys and symmetric keys.

5.4 Strength of Authentication

The table below describes the type of authentication and the authentication data to be used by operators, by role.

Table 5 Identification and Authentication Roles and Data

Role	Type of Authentication	Authentication Data
Administrator (CO)	Identity-based	ECDSA Signature Verification
User	Identity-based	PIN (minimum 7 to 262 characters)
DDS	Identity-based	Device Shared Key (32 Bytes)

The strength of the authentication mechanism is stated in Table 5-1 below.

Table 5-1 Strength of Authentication

Authentication Mechanism	Strength of Mechanism
User Single PIN-entry attempt / False Acceptance Rate	Given a keyboard of 93 characters, the number of PINs using 7 characters is

	937 = 6.01 x 1013. The probability that a random PIN-entry attempt will succeed or a false acceptance will occur is (6.01 x 1013)-1 = 1.66 x10-14. Alternatively, for 7-bit ascii encodings, the probability is (1287)-1 = 1.78 x 10-15.The requirement for a single–attempt / false acceptance rate of no more than 1 in 1,000,000 (i.e., less than a probability of 10-6) is therefore met.
User Multiple PIN-entry attempt in one minute	SPYRUS MDTU-P384 Encryption Module has a maximum bound of 10 successive failed authentication attempts before zeroization occurs. The probability of a successful attack of multiple attempts in a one minute period is no more than 10 * (1.66 x10-14) = 6.01 x 1013. The estimate for ascii input is even less (1.78 x 10-14).The probability of a successful attempt in one minute is therefore less than one in 100,000 (i.e., 1 x 10-5), as required.
Crypto-Officer Single attempt / False Acceptance Rate	Given that the elliptic curve is P-384, the order of the curve is no more than 2^384 and the number of choices of a private key is also bounded by the same value. The probability that a random ECDSA signature verification authentication attempt will succeed or a false acceptance will occur is the same as a successful guess of the signer's private key. This is less than or equal to 1/2^384 = 2.54 x10^{-116}. The requirement for a single–attempt / false acceptance rate of no more than 1 in 1,000,000 (i.e., less than a probability of 10^{-6}) is therefore met.
Crypto-Officer Multiple Signature verification attempt in one minute	The probability of a successful attack of multiple ECDSA signature authentication attempts in a one minute period is approximately bounded by 20 attacks times the probability of a single success 1/2^192. Given that zeroization occurs if the crypto-officer's signature fails verification, the

	computational power needed to process more than this number of attacks is outside of the ability of the module. The probability of a successful attack is less than or equal to 20/2^192 = 3.18 x10^{-57} This is less than one in 100,000 (i.e., 1 x 10^{-5}), as required.
DDS Single Challenge Response attempt / False Acceptance Rate	The Device Shared Key is 32 bytes in length. The probability that a random challenge-response attempt will succeed or a false acceptance will occur is 1/2^256. The requirement for a single–attempt / false acceptance rate of no more than 1 in 1,000,000 (i.e., less than a probability of 10^{-6}) is therefore met.
DDS Multiple Challenge Response attempt in one minute	Based on computation time, the module can process at most 20 challenge/responses in a one minute period. The probability of a successful attack of multiple attempts in a one minute period is 20/2^256. This is less than one in 100,000 (i.e.,1 x 10^-5), as required.

6 Access Control

6.1 Critical Security Parameters (CSPs) and Public Keys

Table 6-1 SPYRUS MDTU-P384 Encryption Module CSPs

CSP Designation	Algorithm(s) / Standards	Symbolic Form	Description
Disk Ephemeral Private	SP 800-56A	$d_{e,U}$	ECDH ephemeral private key used to generate shared secret.
Disk Key Encryption Key (DKEK)	AES 256	DKEK	AES key used to unwrap the Disk Encryption Key (DEK).
Drive Encryption Key (DEK)	AES 256	DEK	A pair of AES 256 keys. The concatenated value is used to encrypt and decrypt the User's encrypted drive.
Hash-DRBG Seed	SP 800-90A	S	NDRNG generated seed used to seed the Hash-DRBG RNG.
Hash-DRBG State	SP 800-90A	s_{HDRBG}	Hash_DRBG state value
Master Encryption Key (MEK)	AES 256	MEK	AES 256 wraps / unwraps user's static private keys in storage.
Secure Channel Private	SP 800-56A	$d_{e,SCHP}$	ECDH Ephemeral Transport Private
Secure Channel Session Key	SP 800-56A	k_{SCSK}	AES key used to encrypt and decrypt commands and responses to and from the card.
Device Shared Secret	N/A	K_{shared}	32 Byte secret value shared by Module and DDS device to support DDS challenge/response authentication.
User PIN	N/A	PIN	The user's 7 character PIN for authentication to the module.
User's Static Signature Private	FIPS 186-4	$d_{ECDSA,s,U}$	ECDSA Static Signature private key
User's Static Transport Private	SP 800-56A	$d_{s,U}$	ECDH Static Transport private key

Table 6-2 SPYRUS MDTU-P384 Encryption Module Public Keys

Key	Algorithm(s) Standards	Description/Usage
Configuration Update Key	FIPS 186-4	The ECDSA P-384 public Key is used to verify the signature of the CO before the settings are changed.
Card Firmware Update Key	FIPS 186-4	The ECDSA P-384 public Key is used to verify the signature of the CO before loading firmware.
Disk Ephemeral Public	SP 800-56A	ECDH Ephemeral Transport Public P384. The key is used to generate a shared secret using ECDH with the User's Static Transport Private key.
Secure Channel Host Public	SP 800-56A	ECDH Ephemeral Transport Public P256
Secure Channel HYDRA Public	SP 800-56A	ECDH Ephemeral Transport Public P256. The key is used to generate a shared secret between the host and the card.
User's Static Signature Public	SP 800-56A	ECDSA Static Signature Public P384.
User's Static Transport Public	SP 800-56A	ECDH Static Transport Public P384. The key for ECDH.

6.2 CSP Access Modes

Table 6-3 SPYRUS MDTU-P384 Encryption Module Access Modes

Access Type	Description
Generate (G)	"Generate" is defined as the creation of a CSP
Delete (D)	"Delete" is defined as the zeroization of a CSP
Use (U)	"Use" is defined as the process in which a CSP is employed. This can be in the form of loading, encryption, decryption, signature verification, or key wrapping.

6.3 Access Matrix

The following table shows the services (see Section 4.2) of the SPYRUS MDTU-P384 Encryption Module (HPC140-F), the roles (see Section 4.1) capable of performing the service, the CSPs (see Section 6.1) that are accessed by the service and the mode of access (see Section 6.2) required for each CSP. The following convention is used: if the role column has an 'X', then that role may execute the command.

Table 6-4 SPYRUS MDTU-P384 Encryption Module Access Matrix

Service Name	Roles			Access to Critical Security Parameters	
	Admin	User	DDS	CSPs	Access Mode
ChangePassword		X		k_{SCSK} $d_{s,U}$ $d_{ECDSA,s,U}$ $d_{e,U,}$ DKEK DEK PIN	**U** **U** **U** **U** **G, U, D** **U** **D,G**
Erase_User_Area		X			
Format		X		$d_{e,U}$ DKEK, DEK	**G, U, D** **G,U,D** **G,U**
GetCapabilities	X	X			
GetConfiguration	X	X			
GetUserState	X	X			
Initialize		X		k_{SCSK} $d_{s,U}$ $d_{ECDSA,s,U}$ $d_{e,U,}$ DKEK DEK MEK	**U** **G** **G** **G, U, D** **G, U, D** **G** **U**
LogOff		X			
LogOn		X		k_{SCSK} $d_{s,U}$ DKEK DEK PIN	**U** **U** **G,U,D** **U** **U**
MountCDROM		X		DEK	**U**
ReadMedia		X		DEK	**U**
ReadUserArea	X	X			
SelfTest	X	X		S, S_{HDRBG},	**G**
SetConfiguration	X			$d_{s,U}$ $d_{ECDSA,s,U}$ DEK	**D** **D** **D**

Service Name	Roles			Access to Critical Security Parameters	
	Admin	User	DDS	CSPs	Access Mode
UpdateFirmware	X			$d_{s,U}$ $d_{ECDSA,s,U}$ DEK	D D D
WriteMedia		X		DEK	U
WriteUserArea		X			
Zeroize	X	X		$d_{s,U}$ $d_{ECDSA,s,U}$ DEK	D D D
Inquiry			X		
Authentication and Mount			X	K_{shared}	U
Test Unit Ready			X		
Set or Change DDS Special Value			X	K_{shared}	D,G

7 Self-Tests

The module performs both power-on and conditional self-tests. The module performs the following power-on self-tests:

- Cryptographic Algorithm Tests:
    - AES-CBC 256, AES-XTS encrypt and decrypt KATs
    - ECDSA-256 sign and verify KATs
    - EC-Diffie-Hellman-256 KAT
    - SHA-224 KAT
    - SHA-256 KAT
    - SHA-384 KAT
    - SHA-512 KAT
    - HASH-DRBG (SHA 512) KAT
- Firmware Test
    - SHA-384 Hash

The module performs the following Conditional Tests:

- Firmware Load Test
    - ECDSA P-384 signed SHA-384 hash verification
- Pairwise Consistency Test
    - ECDSA key pair generation
    - EC-Diffie-Hellman key pair generation
- Continuous Random Number Generator Test
    - HASH-DRBG SP800-90
    - SPYCOS 3.0 HW RNG

8 DRBG Health Tests

The SPYRUS MDTU-P384 Module conforms to the requirements of SP 800-90A regarding Health Tests for the DRBG, including the error handling requirements of SP 800-90A, Section 11.3.

9 Mitigation of Other Attacks

No claims of mitigation of other attacks listed in Section 4.11 of FIPS 140-2 by the SPYRUS MDTU-P384 Encryption Module are made or implied in this document.