FIPS140-2 Crypto Module

Security Policy

Revision:	1.03 RELEASED
Date:	11 November 2009
Doc ref:	NX0408-ME03_v103

Document Control

Copyright and Confidentiality

Control and Ownership

This is a controlled document. Unless stated otherwise, the sole controlled copy of this document resides in the Nexus Wireless document control repository, and other copies in electronic and printed form are uncontrolled.

Ownership and authorship of this document are as follows.

Role	Responsibilities	Assigned to
Document Owner	Overall responsibility for the content and accuracy of the document, and nomination of authors, reviewers and approvers.	P.R.
Document Author(s)	Preparation of the document under the direction of the Document Owner.	J.D. P.R.

Review and Approval

This document must be reviewed and approved for release.

Role	Responsibilities	Assigned to
Document Reviewer(s)	Assessment of the accuracy and relevance of the document contents.	Identified in applicable review record.
Document Approver(s)	Approval of the document for release. The latest released revision of this document has been approved by the personnel identified.	P.R.

Revision History

Revisions ending in a letter (e.g. 1.00a) are draft versions of the indicated revision. The letter is omitted when the document is released (e.g. 1.00b → 1.00).

Revision	Date	Details
1.00	8 January 2009	Initial release.
1.01	8 May 2009	Added RNG Seed Key to CSP descriptions. Clarified data output description during Power-up Self Test.
1.02	14 May 2009	Added HMAC and DSA Key descriptions to CSP descriptions.
1.03	11 November 2009	Addressed comments from NIST Review

Contents

DOCUMENT CONTROL II

CONTENTS III

1. INTRODUCTION 1

1.1 DOCUMENT PURPOSE 1
1.2 AUDIENCE 1
1.3 NEXUS FIPS140-2 CRYPTO MODULE OVERVIEW 1
1.4 DESIGN ASSURANCE 1
1.4.1 Configuration Management *1*
1.5 REFERENCES 2
1.5.1 Standards *2*
1.6 DEFINITIONS AND ACRONYMS 3

2. NEXUS FIPS140-2 CRYPTO MODULE SECURITY POLICY 4

2.1 MODULE OVERVIEW 4
2.2 PORTS AND INTERFACES 5
2.2.1 Physical Ports *5*
2.2.2 Logical Interfaces *6*
2.3 SECURITY LEVELS 7
2.4 APPROVED MODE OF OPERATION 7
2.4.1 Approved Algorithms *7*
2.4.2 Non-Approved Algorithms *8*
2.4.3 Clear Bypass *8*
2.5 OPERATORS AND ROLES 8
2.6 SELF TESTS 8
2.6.1 Power-up Self-Tests *8*
2.6.2 Conditional Self-Tests *9*
2.7 SERVICES 9
2.7.1 Host Services *9*
2.7.2 KFD Services *13*
2.8 KEYS AND CRITICAL SECURITY PARAMETERS 14
2.8.1 Defined Keys and CSPs *14*
2.8.2 Key and CSP Access *17*
2.9 WARM START KEY 18
2.10 REVERSE WARM START KEY 18
2.11 ZEROIZATION 19
2.12 PHYSICAL SECURITY AND MITIGATION OF OTHER ATTACKS 19

3. USER GUIDANCE 20

3.1 PORTS, INTERFACES AND SERVICES 20
3.2 USER RESPONSIBILITIES 20

4. CRYPTO OFFICER (CO) GUIDANCE 21

4.1 PORTS, INTERFACES AND SERVICES 21
4.2 MODULE ADMINISTRATION 21
4.3 MODULE INSTALLATION AND STARTUP 21

1. Introduction

1.1 Document Purpose

This document contains the Security Policy, User Guidance and Crypto Officer Guidance for the Nexus FIPS140-2 Crypto Module.

1.2 Audience

The intended audience for this document consists of FIPS certification authorities and users wishing to evaluate the Security Policy of the module.

This document assumes an audience familiar with terms and definitions contained in the FIPS PUB 140-2 standard.

1.3 Nexus FIPS140-2 Crypto Module Overview

The Nexus FIPS140-2 Crypto Module is a single board, multi-chip module designed to conform to FIPS140-2 standards intended for use in P25 radio equipment.

The Nexus FIPS140-2 Crypto Module provides secure key management, Over-The-Air-Rekeying (OTAR), and voice encryption for the family of Nexus XR25 P25 Transceiver Boards which are a small form-factor, off-the-shelf platform for demanding public safety two-way radio communication products.

The module supports a dedicated 3-wire Key Fill Device (KFD) interface. It includes a complete key storage and critical security material management function for Traffic Encryption Key (TEK), Key Encryption Key (KEK) and Key Storage Key (KSK) keys in non-volatile memory within the module, with protection from unauthorized disclosure or modification.

The FIPS Module executes encryption and decryption of P25 Phase 1 voice and data traffic, Trunking Control Keystream and OTAR Message Authentication Code (MAC) operations.

The module has been configured to store up to the following maximum number of cryptographic entities:

- 1024 keys
- 128 keysets
- 512 key assignments
- 512 Storage Location Number (SLN) mappings
- 32 Radio Set Identifiers (RSIs)

1.4 Design Assurance

1.4.1 Configuration Management

The configuration items comprising the Nexus Wireless FIPS140-2 Cryptographic Module as documented in this security policy document are described with their current versions in the following table:

Configuration Item	Version	Description
FIPS140-2 Crypto Module	1.00	The single board, multi-chip hardware module.
ES0408_RL01_R1_00_000.bin	1.00.000	The main image firmware of the module.
ES0408_RL02_R1_00_000.bin	1.00.000	The boot image firmware of the module.

1.5 References

In the references that follow, the latest edition of a document is implied unless a specific revision or date is given.

1.5.1 Standards

[1] National Institute of Standards and Technology,
Data Encryption Standard (DES),
Doc Ref: FIPS PUB 46-3, October 25, 1999.

[2] National Institute of Standards and Technology,
Security Requirements for Cryptographic Modules,
Doc Ref: FIPS PUB 140-2, May 2001 (amended March 2002).

[3] National Institute of Standards and Technology,
Key Management using ANSI X9.17,
Doc Ref: FIPS PUB 171, April 27, 1992.

[4] National Institute of Standards and Technology,
Secure Hash Standard,
Doc Ref: FIPS PUB 180-3, October, 2008.

[5] National Institute of Standards and Technology,
Digital Signature Standard,
Doc Ref: FIPS PUB 186-2, January 27, 2000.

[6] National Institute of Standards and Technology,
Advanced Encryption Standard (AES),
Doc Ref: FIPS PUB 197, November 26, 2001.

[7] National Institute of Standards and Technology,
The Keyed-Hash Message Authentication Code (HMAC)
Doc Ref: FIPS PUB 198-1, July 2008

[8] National Institute of Standards and Technology,
Recommendations for Applications Using Approved Hash Algorithms (DRAFT)
Doc Ref: draft-SP800-107-July2008, July 9, 2008

[9] National Institute of Standards and Technology,
Recommendation for Key Derivation Using Pseudorandom Functions
Doc Ref: SP 800-108, Nov 2008

[10] Telecommunications Industry Association,
Digital Land Mobile Radio, Security Services Overview,
Doc Ref: ANSI/TIA-102.AAAB-2002, July 2002.

[11] Telecommunications Industry Association,
TIA/EIA STANDARD, Project 25 Digital Radio Over-the-Air Rekeying (OTAR) Protocol,
Doc Ref: TIA/EIA-102.AACA, April 2001.

[12] Telecommunications Industry Association,
TIA STANDARD, Project 25 Digital Radio Over-the-Air-Rekeying (OTAR) Protocol, Addendum 2 – Data Link Independent OTAR
Doc Ref: TIA-102.AACA-2, March 2003.

[13] Telecommunications Industry Association,
TIA STANDARD, Project 25 – Over-the-Air-Rekeying (OTAR) Operational Description,
Doc Ref: TIA-102.AACB, November 2002.

[14] Telecommunications Industry Association,
TIA STANDARD, Project 25 Key Fill Device (KFD) Interface Protocol,
Doc Ref: TIA-102.AACD, February 2005.

1.6 Definitions and Acronyms

Acronym / Term	Definition
AES	Advanced Encryption Standard defined by FIPS197, see [6]
APCO	Association of Public-Safety Communications Officials. Note that in this document, “APCO P25” as used invariably refers to APCO P25 Project 25
APCO P25	APCO P25 Project 25 digital conventional and trunked radio standard
CBC	Cipher Block Chaining
CFB	Cipher Feedback
CSP	Critical Security Parameter
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DSS	Digital Signature Standard defined by FIPS186-2, see [7]
ECB	Electronic Code Book
FIPS	Federal Information Processing Standard
FIPS140-2	The FIPS publication that specifies the security requirements for cryptographic modules conformed to by module described in this policy
HMAC	keyed-Hash Message Authentication Code defined by FIPS198-1, see [7]
IV	Initialization Vector
KAT	Known Answer Test
KEK	Key Encryption Key. A key used solely for encryption/decryption of other keys, and never used for traffic (as opposed to a TEK).
KFD	Keyfill Device
KMF	Key Management Facility
KSK	Key Storage Key
MAC	Message Authentication Code
MI	Message Indicator
MN	Message Number, see [11]
MNP	Message Number Period, see [11]
NIST	National Institute of Standards and Technology
OFB	Output Feedback
OTAR	Over the Air Rekeying, see [11], [12] and [13]
P25	See APCO P25
RNG	Random Number Generator
RSI	Radio Set Identifier, see [11]
RSK	RNG Seed Key
SHA-1	Secure Hash Algorithm-1
SHS	Secure Hash Standard defined by FIPS180-3, see [4]
TEK	Traffic Encryption Key. A key used solely for encrypting / decrypting voice or data traffic, as opposed to a KEK.

2. Nexus FIPS140-2 Crypto Module Security Policy

2.1 Module Overview

The module is a multi-chip embedded board 28 mm x 25 mm in size and based around a 32-bit microcontroller.

The microcontroller contains a secure internal flash memory and in addition the board contains a high capacity flash memory device. All sensitive data stored in this memory device is encrypted with the Key Storage Key (KSK).

The crypto boundary is the entire circuit board with all interfaces provided through the 30-pin board-to-board connector shown at the top of Figure 1 below.

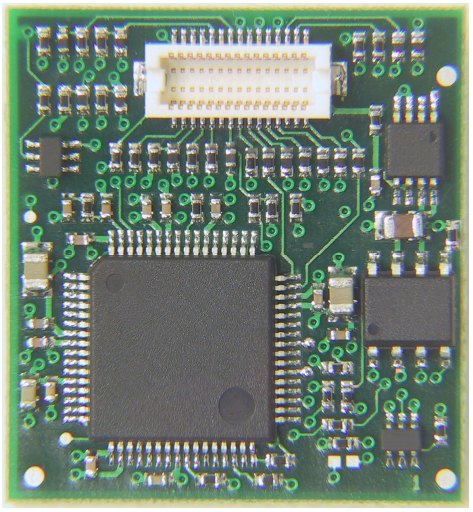

Figure 1 : Nexus FIP140-2 Crypto Module (Top)

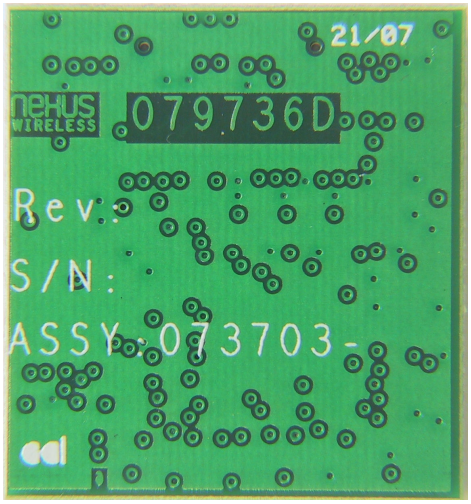

Figure 2 : Nexus Wireless FIPS140-2 Crypto Module (Bottom)

2.2 Ports and Interfaces

2.2.1 Physical Ports

The Nexus FIPS140-2 Crypto Module implements the following physical ports:

Physical Port	Description
Power Port	Power input.
Host Port	Host control, status and data (input / output).
KFD Port	Keyfill device (KFD) control, status and data (input / output).

The Power Port consists of a single power rail and ground pins.

The Host Port consists of clock and reset inputs, a bidirectional data interface that can operate in synchronous and asynchronous modes, zeroize input, error and attention outputs, and host LED indicator inputs and outputs.

The KFD Port consists of the three standard APCO Project 25 Keyfill Device Interface signals (Keyload, K/F, and GND) as defined in [14].

All physical ports are accessed via the module's 30-pin connector.

2.2.2 Logical Interfaces

The Nexus FIPS140-2 Crypto Module implements the following logical interfaces, which map to the physical ports as indicated:

Logical Interface	Description	Physical Port
Control Input Interface (Host)	Control input from host equipment. Control Input is in the form of hardware input signals and Host Commands received via the bidirectional data interface.	Host port
Status Output Interface (Host)	Status output to host equipment. Status Output is in the form of hardware output signals and Host Responses emitted via the bidirectional data interface.	Host port
Data Input Interface (Host)	Data input from host equipment. Data Input is contained in Host Commands received via the bidirectional data interface.	Host port
Data Output Interface (Host)	Data output to host equipment. Data Output is contained in Host Responses emitted via the bidirectional data interface.	Host port
Control Input Interface (KFD)	Control input from keyfill device. Control Input is in the form of the Keyload input signal, and in the form of Key Management Messages (KMMs) received via the K/F signal.	KFD port
Status Output Interface (KFD)	Status output to keyfill device. Status Output is in the form of Key Management Messages (KMMs) emitted via the K/Fsignal.	KFD port
Data Input Interface (KFD)	Data input from keyfill device. Data Input is contained in Key Management Messages (KMMs) received via the KFD_K_F signal.	KFD port
Data Output Interface (KFD)	Data output to keyfill device. Data Output is in the form of Key Management Messages (KMMs) emitted via the KFD_K_F signal.	KFD port

The KFD interfaces are activated by assertion of the Keyload control input. Activation of the KFD interfaces deactivates the Host interfaces.

2.3 Security Levels

The Nexus FIPS140-2 Crypto Module meets the following security levels, as defined in FIPS140-2:

Area	FIPS PUB 140-2 Security Level
Cryptographic Module Specification	Level 1
Cryptographic Module Ports and Interfaces	Level 1
Roles, Services, and Authentication	Level 1
Finite State Model	Level 1
Physical Security	Level 1
Operational Environment	N/A
Cryptographic Key Management	Level 1
EMI/EMC	Level 1
Power-up Self Tests	Level 1
Design Assurance Level	Level 1
Mitigation of Other Attacks	N/A

2.4 Approved Mode of Operation

The Nexus FIPS140-2 Crypto Module operates in both a FIPS approved mode and a non FIPS approved mode.

To operate in the FIPS-approved mode, the DES algorithm must not be used. Using only the FIPS approved AES algorithm ensures that the module operates in the FIPS approved mode.

2.4.1 Approved Algorithms

The following Approved algorithms are used in Approved mode:

Algorithm	Purpose	Validation Certificate	Firmware Image
AES-256 ECB	Key wrapping for storage and external transport	#914	Main
AES-256 OFB	P25 voice and data traffic encryption	#914	Main
AES-256 CBC	P25 OTAR / KFD MAC calculation	#914	Main
PRNG[1]	Key generation	#524	Main
SHA-1[2]	PRNG	#901	Main
SHA-1[2]	Firmware integrity test and Firmware load test	#928	Boot
DSA	Firmware load test	#337	Boot
HMAC	Firmware integrity test	#533	Boot

Note 1: The PRNG algorithm is FIPS186-2 Appendix 3.1.

Note 2: Separate implementations of SHA-1 are used by the PRNG, and for firmware integrity and firmware load tests.

2.4.2 Non-Approved Algorithms

The module employs the following non-FIPS approved cryptographic algorithms.

Algorithm	Purpose
DES ECB	Key wrapping for external transport
DES OFB	P25 voice and data traffic encryption
DES CBC	P25 OTAR/KFD MAC calculation
AES MAC	P25 OTAR/KFD MAC calculation

If any operation using the DES algorithm is performed, the module will not operate in a FIPS approved mode of operation.

2.4.3 Clear Bypass

The module does not support clear bypass.

2.5 Operators and Roles

The Nexus FIPS140-2 Crypto Module supports a single operator, who may assume the following roles as defined in FIPS PUB 140-2 section 4.3.1:

Role	Type of Authentication	Authentication Data
User	None	N/A
Crypto Officer (CO)	None	N/A

Each service is implicitly associated with one of the above roles and is automatically assumed when the service is invoked. No authentication is performed.

The module does not support a Maintenance role or any other role.

2.6 Self Tests

The Nexus FIPS140-2 Crypto Module performs both power-up self-tests and conditional self-tests.

In the case of any self-test failure the module enters a critical error state and issues a critical error indication on the Host Interface. In the critical error state no cryptographic functions can be invoked and the module must be reset or powered down to exit this state.

2.6.1 Power-up Self-Tests

Power-up self-tests are performed automatically when the module comes out of reset. The Power-up self tests consist of the following tests:

- Firmware integrity tests of the BOOT and MAIN firmware images using the approved HMAC algorithm.
- Cryptographic algorithm tests using known answer tests of all approved AES, PRNG, SHA-1, and HMAC algorithms listed in 2.4.1, and all non-approved DES algorithms listed in 2.4.2.
- A critical function test for the approved PRNG using an initial iteration of the continuous random number generator test.
- Integrity tests on the contents of the high capacity flash memory device.

During the execution of the power-up self tests the Crypto Module does not output any data via the data output interface. On successful completion of the power-up self tests the module issues a successful status indication on the Host Interface.

2.6.2 Conditional Self-Tests

Conditional self-tests are performed when appropriate security services are invoked. The conditional self-tests performed are:

- The firmware load test using the approved DSA algorithm when the firmware upgrade service is invoked.
- The continuous random number generator test using the approved PRNG algorithm when the random number generation function is implicitly invoked.

2.7 Services

The Nexus FIPS140-2 Crypto Module supports two types of services. **Host services** are provided to attached host equipment, such as a P25 radio via the Host Port. **KFD services** are provided to an attached keyfill device (KFD) via the KFD Port. The module permits only one service at a time to be invoked, regardless of whether it is a host service or a KFD service.

The 'Perform Self Test' service required by FIPS PUB 140-2 section 4.3.2 is invoked by powering on or resetting the module.

The 'Show Status' service required by FIPS PUB 140-2 is implicit in the 'Perform Self Test' host service, which reports the module status on success. Failure is indicated by a Critical Error status indication, a state that can only be cleared by resetting the module.

The 'Perform Approved Security Function' service required by FIPS PUB 140-2 is collectively provided by the remainder of the Host and KFD services.

2.7.1 Host Services

The module supports the following host services, each implicitly associated with the role indicated.

Service[1]	Description	Role
Show Status	Output the current status of the cryptographic module. **Inputs:** None. **Outputs:** Status, Main firmware version, Boot firmware version.	CO
Perform Self-Tests	Perform cryptographic validation test. **Inputs:** None. **Outputs:** Status, Main firmware version, Boot firmware version.	CO
Channel Open	Open a crypto channel for data processing. **Inputs:** Channel Type, Channel Number, Mode, Message Length, Derived Key Flag, Algorithm ID, Key ID, Key Address, Message Indicator (Optional). **Outputs:** Result, Current Message Indicator, Clocked Message Indicator.	User
Channel Data	Encryption/Decryption of ciphertext and plaintext through an open Data or Voice channel, or MAC generation from a message through an open MAC channel.. **Inputs:** Channel Type, Channel Number, Channel Data. **Outputs:** Result, Channel Data.	User
Channel Close	Close a crypto channel. **Inputs:** Channel Type, Channel Number. **Outputs:** Result, Channel Data.	User

Channel Short Data	Encryption/Decryption of ciphertext and plaintext or MAC generation through an appropriate channel type as a single operation, without maintaining channel state for subsequent processing. **Inputs:** Channel Type, Channel Number, Mode, Message Length, Derived Key Flag, Algorithm ID, Key ID, Key Address, Message Indicator (Optional), Channel Data. **Outputs:** Result, Current Message Indicator, Channel Data.	User
Set Key	Store a key. **Inputs:** Keyset ID, Key Algorithm ID, Key ID, Key Type, Key Name, SLN, KEK Algorithm ID, KEK Key ID, Temporary Key Flag, Key Name, Key Material. **Outputs:** Result.	CO
Delete Key	Delete a stored key. **Inputs:** Key Type (Optional), Algorithm ID, Key ID, Key Address (Optional). **Outputs:** Result.	CO
Delete Keyset	Delete a stored keyset. **Inputs:** Keyset ID. **Outputs:** Result.	CO
Changeover Keyset	Change to a new active keyset. **Inputs:** Superseded Keyset ID, Activated Keyset ID. **Outputs:** Result, Superseded Keyset ID, Activated Keyset ID.	CO
Get Key Info	Report information about stored keys. **Inputs:** Iteration Marker, Filter Type, Keyset ID (Optional), Storage Location Number (Optional). **Outputs:** Result, Iteration Marker, Key IDs, Key Keyset IDs, Key Storage Location Numbers, Key Statuses.	CO
Validate Key	Lookup and validate the existence of a key. **Inputs:** Key Type, Algorithm ID, Key ID. **Outputs:** Result.	CO
Get Keyset IDs	Report IDs of stored keysets. **Inputs:** None. **Outputs:** Result, Keyset IDs, Keyset Statuses.	CO
Set Keyset Info	Store keyset information. **Inputs:** Keyset ID, Algorithm ID, Key Type, Keyset Name. **Outputs:** Result.	CO
Get Keyset Info	Retrieve stored keyset information. **Inputs:** Keyset ID. **Outputs:** Result, Keyset ID, Algorithm ID, Key Type, Keyset Name.	CO
Set Key Assignment	Store key assignment mapping information.	CO

	Inputs: Key Assignment Type, Key Assignment ID, Storage Location Number. **Outputs:** Result.	
Get Key Assignment	Retrieve stored key assignment mapping information. **Inputs:** Key Assignment Type, Iteration Marker. **Outputs:** Result, Iteration Marker, Key Assignments.	CO
Set RSI	Store OTAR RSI information. **Inputs:** Affected RSI, New RSI, Message Number. **Outputs:** Result.	CO
Get RSI	Retrieve stored OTAR RSI information. **Inputs:** RSI or RSI Type. **Outputs:** Result, KMF RSI, Incoming Message Number, Outgoing Message Number, Message Number Period, RSIs.	CO
Generate Warm Start Key	Generate a key for use in an OTAR Reverse Warm State Segment. **Inputs:** TEK Algorithm ID (Optional), TEK Key ID (Optional), TEK Address (Optional), KEK Algorithm ID, KEK Key ID, KEK Address (Optional). **Outputs:** Result, Algorithm ID, Key ID, Key Address, KEK Algorithm ID, KEK Key ID, KEK Address, Wrapped Key Material.	CO
Set Reverse Warm Start Policy	Store the policy for the OTAR reverse warm start procedure. **Inputs:** TEK Algorithm ID. **Outputs:** TEK Key ID.	CO
Partial Zeroize[2]	Perform a partial zeroization. **Inputs:** Level. **Outputs:** Result.	CO
Zeroize[3]	Zeroize all critical security parameters (CSPs) and information. **Inputs:** None. **Outputs:** Result.	CO
Update Firmware[4, 5]	Update the main firmware image. **Inputs:** Firmware data, Firmware signature. **Outputs:** Result.	CO

Note 1: Other than Zeroize and Update Firmware, all host services are invoked via Host Commands.

Note 2: The level of zeroization is specified by the service inputs and described in 2.9.

Note 3: The Zeroize service is invoked by asserting the ZEROISE input signal.

Note 4: The Update Firmware service is invoked by asserting the ZEROISE input signal while bringing the module out of reset.

Note 5: Only FIPS validated firmware may be loaded in order to maintain the cryptographic module's FIPS validation and run in approved FIPS mode. Updating with any non FIPS validated cryptographic module

firmware will result in the module running as a non-validated module in a non-approved FIPS mode of operation.

2.7.2 KFD Services

The Nexus FIPS140-2 Crypto Module supports the following KFD services, each implicitly associated with the role indicated.

Service	Description	Role
KFD Inventory - List Active Keyset IDs	Retrieve the active Keyset IDs. **Inputs:** None. **Outputs:** Active Keyset IDs.	CO
KFD Inventory - List Active Keys	Retrieve the stored Keyset IDs, SLNs, Algorithm IDs and Key IDs of the active keys. **Inputs:** Inventory Marker, Maximum Key Count. **Outputs:** Inventory Marker, Key Keyset IDs, Key Storage Location Numbers, Key Algorithm IDs, Key IDs.	CO
KFD Inventory - List RSI Items	Retrieve the stored individual RSI and group RSIs. **Inputs:** None. **Outputs:** RSIs, Message Numbers.	CO
KFD Inventory - List KMF RSI	Retrieve the stored KMF RSI. **Inputs:** None. **Outputs:** KMF RSI.	CO
KFD Inventory - List MNP	Retrieve the stored MNP parameter. **Inputs:** None. **Outputs:** Message Number Period.	CO
KFD Inventory - List Keyset Tagging Info	Retrieve stored Keyset information. **Inputs:** None. **Outputs:** Keyset IDs, Keyset Algorithm IDs, Update Instruction Blocks (Optional), Date Times (Optional), Keyset Names (Optional).	CO
KFD Modify Key	Modify, set or erase a stored key. **Inputs:** KEK Algorithm ID, KEK Key ID, Update Count (Optional), Message Indicator (Optional), Keyset ID, Keyset Algorithm ID, Storage Location Numbers, Key IDs, Key Materials, Key Checksums (Optional), Key Names (Optional). **Outputs:** Algorithm IDs, Key IDs, Modify Key Statuses.	CO
KFD Change RSI	Change or set the individual RSI or a group RSI. **Inputs:** Changed RSIs, Added RSIs, Message Numbers. **Outputs:** Changed RSIs, Added RSIs, Change RSI Statuses.	CO
KFD Load Config	Load KMF configuration parameters (KMF RSI, MNP) into the module. **Inputs:** KMF RSI, Message Number Period. **Outputs:** KMF RSI, Message Number Period, Result.	CO
KFD Zeroize	Zeroize all critical security parameters (CSPs) and	CO

	information. **Inputs:** None. **Outputs:** None.	
KFD Changeover	Perform a keyset changeover from one stored keyset to another stored keyset. **Inputs:** Superseded Keyset IDs, Activated Keyset IDs. **Outputs:** Superseded Keyset IDs, Activated Keyset IDs.	CO

2.8 Keys and Critical Security Parameters

2.8.1 Defined Keys and CSPs

The Nexus FIPS140-2 Crypto Module supports the following Keys and Critical Security Parameters (CSPs).

CSP	Key Type	Key Name	Stored	Generated	Zeroized	Description
KSK	AES 256	Key Storage Key	Yes	Yes	Yes	Used to encrypt stored keys. The KSK is auto-generated as the last step of zeroization using the approved RNG. The KSK is stored in the internal flash memory of the microcontroller in plaintext form. The internal flash memory of the microcontroller cannot be accessed from the external pins of the device.
TEK	AES 256	Traffic Encryption Key	Yes	No	Yes	Used to encrypt / decrypt voice or data traffic, or to perform OTAR MAC calculation. TEKs are established via Host or KFD services. TEKs are stored in the high capacity flash memory device in ciphertext form, encrypted using the KSK.
KEK	AES 256	Key Encryption Key	Yes	No	Yes	Used to encrypt / decrypt (‘wrap’ / ’unwrap’) TEK or other KEK keys for external transport. KEKs are established via Host or KFD services. KEKs are stored in the high capacity flash memory device in ciphertext form, encrypted (‘wrapped’) using the KSK.

CSP	Key Type	Key Name	Stored	Generated	Zeroized	Description
WK	AES 256	Working Key	No	Reverse Warm Start Key only	Yes	A working copy of any key type. WKs are 'working copies' of KSKs, TEKs or KEKs, initialized during operations that require the cleartext form of these keys, and stored in internal microcontroller RAM in plaintext form. The internal microcontroller RAM cannot be accessed from the external pins of the device. The Reverse Warm Start Key is generated using the approved RNG. See 2.10 for details. All WKs are erased by the Zeroize host service or module power down or reset.
RSK	256-bit seed key	RNG Seed Key	No	Yes	Yes	The RSK is used by the RNG. The RNG Seed Key is initialized at startup using entropy data generated from variations between independent clock sources in the microcontroller. The RNG Seed Key is stored in internal microcontroller RAM in plaintext form. The RNG Seed Key cannot be accessed from the external pins of the device. The RNG Seed Key is updated on each occasion that a random number is generated. The RNG Seed Key is erased by the Zeroise service or module power down or reset.
HK	256-bit secret key	HMAC Key	Yes	No	No	The HMAC Key is used by the firmware integrity test for the Boot and Main firmware images as described in 2.6.1. The HMAC Key is secret and is programmed at module production. The HMAC Key is stored in internal microcontroller Flash memory in plaintext form. The HMAC Key cannot be accessed from the external pins of the device.

CSP	Key Type	Key Name	Stored	Generated	Zeroized	Description
DK	1024-bit public key	DSA Key	Yes	No	No	The DSA Key is used by the firmware load test for signature verification of new Main firmware images as described in 2.6.2. The DSA Key is stored in internal microcontroller Flash memory in plaintext form. The DSA Key cannot be accessed from the external pins of the device.

2.8.2 Key and CSP Access

The following tables define how Host and KFD services access Keys and CSPs. The following terminology is used:

- R: Read, the module uses the Key / CSP without modifying it.
- W: Write, the module modifies or deletes the Key / CSP.

The Host Services access the Keys and CSPs as indicated in the following table. Services not listed do not access a Key or CSP.

Host Service	Key and CSP Access						
	KSK[1]	KEK	TEK[2]	WK	RSK	HK	DK
Channel Open	-	-	R	R[1] / W	R / W[5]	-	-
Channel Data	-	-	-	R[2] / W		-	-
Channel Close	-	-	-	R[2] / W		-	-
Channel Short Data	-	-	R	R[3] / W		-	-
Set Key	-	R / W	R / W	R[1]		-	-
Delete Key	-	W	W	-		-	-
Delete Keyset	-	W	W	-		-	-
Generate Warm Start Key	-	R	-	R[1] / W	R / W	-	-
Partial Zeroize[4]	-	W	W	W		-	-
Zeroize[4]	W	W	W	W	W	-	-

Note 1: The KSK is read from internal microcontroller flash memory during module initialization and stored in internal microcontroller RAM as a WK until the module is reset, powered down, or zeroized.

Note 2: The TEK is read from the high capacity flash memory device in ciphertext form when a channel is opened, decrypted using the AES 256 KSK WK, and stored as a WK in internal microcontroller RAM until the module is reset, powered down, zeroized or the channel closed.

Note 3: Both the KSK WK and the TEK WK are read. The Channel Short Data is a short form of the sequence Channel Open, Channel Data and Channel Close.

Note 4: See description of zeroization in section 2.9.

Note 5: When opening a channel for encryption, the client may request the module to generate the initialization vector using the approved RNG.

The KFD Services access the CSPs as indicated in the following table:

KFD Service	Key and CSP Access						
	KSK[1]	KEK	TEK	WK	RSK	HK	DK
KFD Inventory - List Active Keyset IDs	-	-	R[3]	-	-	-	-
KFD Invetory - List Active Keys	-	-	R[3]	-	-	-	-
KFD Inventory - List RSI Items	-	-	R[3]	-	-	-	-
KFD Inventory - List KMF RSI	-	-	R[3]	-	-	-	-
KFD Inventory - List MNP	-	-	R[3]	-	-	-	-
KFD Invetory - List Keyset Tagging Info	-	-	R[3]	-	-	-	-
KFD Modify Key	-	W	R[3] / W	R[1]	-	-	-
KFD Change RSI	-	-	R[3]	-	-	-	-
KFD Load Config	-	-	R[3]	-	-	-	-
KFD Zeroize[2]	W	W	R[3] / W	W[1]	-	-	-
KFD Changeover	-	-	R[3]	-	-	-	-

Note 1: The KSK is read from internal microcontroller flash memory during module initialization and stored in internal microcontroller RAM as a WK until the module is reset, powered down, or zeroized.

Note 2: See description of zeroization in section 2.9.

Note 3: If the KFD service request is encrypted or a MAC is applied, a TEK must be read.

2.9 Warm Start Key

The Warm Start Key is a Working Key and is a temporary TEK used for OTAR infrastructure initiated message authentication and encryption when the radio has no TEKs available. The Warm Start Key is used in the OTAR Warm Start Procedure described in [11].

The Warm Start Key is a type of Working Key that it is provided in wrapped form by the host using the Set Key host service rather than being read from the high capacity flash memory device. The key is stored in the internal microcontroller RAM and cannot be accessed directly or indirectly, from the external pins of the device. Like all Working Keys, the Warm Start Key is erased by the Zeroize host service or module power down or reset.

2.10 Reverse Warm Start Key

The Reverse Warm Start Key is a Working Key and is a temporary TEK used for OTAR radio initiated message authentication when the radio has no TEKs available. The Reverse Warm Start Key is used in the OTAR Reverse Warm Start Segment described in [12].

The Reverse Warm Start Key is generated using the approved RNG in response to the Generate Warm Start Key host service request.

The Reverse Warm Start Key is a type of Working Key that it is generated rather than being read from the high capacity flash memory device. The key is stored in the internal microcontroller RAM and cannot be accessed directly or indirectly, from the external pins of the device. Like all Working Keys, the Reverse Warm Start Key is erased by the Zeroize host service or module power down or reset.

Usage of the Reverse Warm Start Key may be enabled or disabled with the Set Reverse Warm Start Policy host service. The default policy is to not allow usage of the Reverse Warm Start Key. Full Zeroization or Partial Zeroization of Levels 0 and 1 restore the Reverse Warm Start policy to this default.

2.11 Zeroization

The module supports three types of zeroization.

- Full zeroization is performed by the Zeroize service, invoked via the ZEROISE control input. All CSPs are zeroized and a new KSK is generated.
- KFD zeroization is performed by the Zeroize-Command KFD service, invoked via the corresponding KFD command. It has the same effect as full zeroization except that the RSK is not zeroized.
- Partial zeroization is performed by the Partial Zeroize host service, invoked via the corresponding Host command. It deletes most of the CSPs and related information held by the module, but retains a minimal set of CSPs (specifically the RSK, the KSK and the unique KEK) that allow the module to perform further OTAR operations such as Warm Start.

The supported levels of partial zeroization are

- Level 0 – all CSP data is zeroized with the exception of the RSK, i.e. the KSK, all KEKs, TEKs and WKs are zeroized. A new KSK is generated.
- Level 1 – all CSP data is zeroized, i.e. the KSK, all KEKs, TEKs and WKs are zeroized. Identical to Level 0 except that some non-CSP OTAR configuration data is retained. A new KSK is generated.
- Level 2 – all CSP data except the minimum required to allow participation in further OTAR transactions. The KSK and the unique KEK are not zeroized, however all other KEKs, TEKs and WKs are zeroized. Some non-CSP OTAR configuration data is retained.

Only full zeroization is considered to be zeroization as defined by FIPS PUB 140-2.

2.12 Physical Security and Mitigation of Other Attacks

The Nexus FIPS140-2 Crypto Module is a multi-chip Embedded Cryptographic Module which utilises production-grade components with standard passivation techniques. The module is intended for deployment within production-grade host equipment and does not incorporate any enclosure of its own.

CSPs stored in the high capacity flash memory device are encrypted with the KSK to prevent unauthorised disclosure via physical probing of device pins. The KSK is itself stored in internal flash memory of the microcontroller. Access to the microcontroller's internal flash memory and debug facilities is disabled during module manufacture.

The module is not designed to mitigate any other specific attacks.

No actions are required by the operator(s) to ensure that physical security is maintained.

3. User Guidance

3.1 Ports, Interfaces and Services

The FIPS140-2 Crypto Module makes available to the User:

- The Host port as described in section 2.2.1.
- The Host logical interfaces as described in section 2.2.2.
- The Host services associated with the User role as described in section 2.7.1.

3.2 User Responsibilities

To ensure the module operates in the approved FIPS mode of operation the User must only invoke Host Services using the approved algorithms listed in 2.4.1. Invoking Host Services that use non-approved algorithms disables the approved FIPS mode of operation. Specifically, invoking any operation that uses the DES algorithm for encryption or decryption, MAC calculations, or key wrapping or unwrapping, causes the module to operate in a non-approved FIPS mode of operation.

When updating firmware only FIPS validated firmware may be loaded in order to maintain the cryptographic modules FIPS validation and run in approved FIPS mode. Updating with any non FIPS validated cryptographic module firmware will result in the module running as a non-validated module in a non-approved FIPS mode of operation.

4. Crypto Officer (CO) Guidance

4.1 Ports, Interfaces and Services

The FIPS140-2 Crypto Module makes available to the CO:

- The Host and KFD ports as described in section 2.2.1.
- The Host and KFD logical interfaces as described in section 2.2.2.
- The Host services associated with the CO role as described in section 2.7.1.
- The KFD services associated with the CO role as described in section 2.7.2.
- The security parameters described in section 2.8.1.

4.2 Module Administration

CO administration of the FIPS140-2 Crypto Module is performed using the CO Host and KFD services. These services are invoked by the host equipment and keyfill device respectively. Procedures for operating host equipment and keyfill devices are beyond the scope of this document.

After manufacture or zeroization, the module will contain no keys other than the KSK and must be provisioned via the Host or KFD ports. The KSK is automatically generated after zeroization is complete.

The following assumptions about user behaviour are relevant to the secure operation of the module. It is assumed that:

- Any Keyfill Device (KFD) that conforms to the APCO Project 25 standard KFD interface specification is authorised to invoke KFD services on the module. Any mechanisms implemented by that device to authenticate the operator of the device are beyond the scope of this document.
- Any Host equipment that conforms to the module's Host Interface specification is authorised to invoke host services on the module. Any mechanisms implemented by the host equipment to authenticate the operator of the equipment are beyond the scope of this document.

4.3 Module Installation and Startup

Installation of the module consists of plugging it into the FIPS140-2 Crypto Module socket on the host equipment. Care should be taken to ensure that the module is correctly oriented, and appropriate anti-static guidelines should be observed when handling the module.

Startup of the module consists of application of power and clock to the module followed by negation of the FIPS_RESET signal, and is performed by the host equipment. Any operator actions required by the host equipment to initiate module startup are beyond the scope of this document.

End of Document.