THALES
Building a future we can all trust

SafeNet PCIe Cryptographic Module for SafeNet IS

NON-PROPRIETARY SECURITY POLICY

FIPS 140-2, Level 3

Document Information

Document Part Number	002-010934-001
Release Date	November 12, 2021

Revision History

Revision	Date	Reason
G	July 21, 2021	Initial version.
H	November 12, 2021	Updates per CMVP comments to address typos.

Trademarks, Copyrights, and Third-Party Software

Disclaimer

All information herein is either public information or is the property of and owned solely by Thales, and/or its subsidiaries who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any of Thales's information.

This document can be used for informational, non-commercial, internal and personal use only provided that:

- The copyright notice below, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any network computer or broadcast in any media other than on the NIST CMVP validation list and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided "AS IS" without any warranty of any kind. Unless otherwise expressly agreed in writing, Thales makes no warranty as to the value or accuracy of information contained herein.

Thales hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Thales be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Thales does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Thales be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Thales products. Thales disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service or loss of privacy.

CONTENTS

ACRONYMS AND ABBREVIATIONS

Term	Definition
ANSI	American National Standards Institute
CA	Certification Authority
CKE	Key Export with RA
CKG	Cryptographic Key Generation
CL	Cloning (a capability configuration used to allow the secure transfer of key objects from one module to another for backup and restore and object replication purposes).
CLI	Command Line Interface
CO	Crypto Officer
CRC	Cyclic Redundancy Check
CRT	Chinese Remainder Theorem
CSP	Critical Security Parameter
CU	Crypto User
DAK	Device Authentication Key
DH	Diffie Hellman
DRBG	Deterministic Random Bit Generator
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie Hellman
FIPS	Federal Information Processing Standard
FSC	Firmware Signing Certificate
GSK	Global Storage Key
HA	High Assurance
HMAC	Hash-based Message Authentication Code
HOC	Hardware Origin Certificate
HOK	Hardware Origin Key

Term	Definition
HSM	Hardware Security Module
KAT	Known Answer Test
KDF	Key Derivation Function
KEK	Key Encryption Key
MAC	Message Authentication Code
Masking	A SafeNet term to describe the encryption of a key for use only within a SafeNet cryptographic module.
MIC	Manufacturer’s Integrity Certificate
MIK	Manufacturer’s Integrity Key
MSK	Manufacturer’s Signature Key
MTK	Master Tamper Key
MVK	Manufacturers Verification Key
NDRNG	Non-Deterministic Random Number Generator
PCI	Peripheral Component Interconnect
PED	PIN Entry Device
PIN	Personal Identification Number
PKCS	Public-Key Cryptography Standards
PRNG	Pseudo-Random Number Generator
PSK	Partition Storage Key
PSS	Probabilistic Signature Scheme
RA	Registration Authority
RNG	Random Number Generator
RPED	Remote PED
RPK	Remote PED Key
RPV	Remote PED Vector
SA	Server-Attached
SADK	Security Audit Domain Key

Term	Definition
SALK	Security Audit Logging Key
SCU	Secure Capability Update
SFF	Small Form Factor
SHS	Secure Hash Standard
SO	Security Officer
SRK	Secure Recovery Key
TUK	Token or Module Unwrapping Key
TVK	Token or Module Variable Key
TWC	Token or Module Wrapping Certificate
TWK	Token or Module Wrapping Key
USK	User’s Storage Key

REFERENCES

[FIPS 140-2] Federal Information Processing Standards Publication (FIPS PUB) 140-2, ‘Security Requirements for Cryptographic Modules’, May 25, 2001 (including change notices 12-02-2002).

[FIPS 140-2 IG] NIST, Implementation Guidance for FIPS 140-2 and the Cryptographic Module Validation Program, May 1, 2021.

[FIPS 180-4] Federal Information Processing Standards Publication 180-4, Secure Hash Standard (SHS), NIST, August 2015.

[FIPS 186-4] Federal Information Processing Standards Publication 186-4, Digital Signature Standards (DSS), NIST, July 2013.

[FIPS 197] Federal Information Processing Standards Publication 197, Specification for the Advanced Encryption Standard (AES), November 26, 2001.

[FIPS 202] Federal Information Processing Standards Publication 202, SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, August 2015.

[FIPS 198-1] Federal Information Processing Standards Publication 198-1, The Keyed-Hash Message Authentication Code (HMAC), July 2008.

[SP800-38A] NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation – Methods and Techniques, Morris Dworkin, December 2001.

[SP800-38B] NIST Special Publication 800-38B, Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication, May 2005 (with October 2016 updates).

[SP800-38D] NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007.

[SP800-38F] NIST Special Publication 800-38F, Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping, December 2012.

[SP800-52r2] NIST Special Publication 800-52 Rev 2, Guidelines for the Selection, Configuration, and Use of Transport Lauer Security (TLS) Implementations, August 2019.

[SP800-56Ar3] NIST Special Publication 800-56A, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, Revision 3, April 2018.

[SP800-56Br2] NIST Special Publication 800-56B, Recommendation for Pair-Wise Key-Establishment Schemes Using Integer Factorization Cryptography, Revision 2, March 2019.

[SP800-56Cr2] NIST Special Publication 800-56C, Recommendation for Key-Derivation Methods in Key-Establishment Schemes, Revision 1, April 2018.

[SP800-67r2] NIST Special Publication 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Rev 1, January 2012.

[SP800-90Ar1] NIST Special Publication SP800-90A, Recommendation for Random Number Generation Using Deterministic Bit Generators, Rev1, June 2015.

[SP800-90B] NIST, SP800-90B, “Recommendation for the Entropy Sources Used for Random Bit Generation”, Version 1.0, January 2018.

[SP800-131Ar2] NIST Special Publication 800-131A revision 2, Transitioning the Use of Cryptographic Algorithms and Key Lengths, March 2019.

[SP800-132] NIST Special Publication 800-132, Recommendation for Password-Based Key Derivation: Part 1: Storage Applications, December 2010.

[SP800-135r1] NIST Special Publication 800-135, Recommendation for Existing Application-Specific Key Derivation Functions, December 2011.

[PKCS #1] PKCS #1: RSA Cryptographic Standard, RSA Laboratories, v2.1.

[RFC5246] RFC 5246, The Transport Layer Security (TLS) Protocol, Version 1.2, August 2008.

[RFC5288] RFC 5288, AES Galois Counter Mode (GCM) Cipher Suites for TLS, August 2008.

[RFC7516] RFC 7516, JSON Web Encryption (JWE), May 2015.

[RFC8446] RFC 8446, The Transport Layer Security (TLS) Protocol Version 1.3, ISSN: 2070-1721, August 2018.

[KMIP 2.1] Key Management Interoperability Protocol Specification Version 2.1, Committee Specification 01, 07 May 2020.

PREFACE

This document deals only with operations and capabilities of the SafeNet PCIe Cryptographic Module in the technical terms of [FIPS 140-2].

General information on Thales products is available from the following sources:

- the Thales internet site contains information on the full line of available products at https://cpl.thalesgroup.com
- product updates and technical support literature is available from the Thales Customer Support Portal at https://supportportal.thalesgroup.com/csm
- technical or sales representatives of Thales can be contacted through one of the channels listed on https://cpl.thalesgroup.com/contact-us

> NOTE: You require an account to access the Customer Support Portal. To create a new account, go to the portal and click on the **REGISTER** link.

1 General

1.1 Purpose

This document describes the security policies enforced by the Thales SafeNet PCIe Cryptographic Module for SafeNet IS.

This document applies to Hardware Version VBD-05, Version Code 0101; VBD-05, Version Code 0102; and VBD-05, Version Code 0103 with Firmware Versions 6.3.4 and 6.3.5.

1.2 Scope

The security policies described in this document apply to the Trusted Path Authentication (Level 3) configuration of the SafeNet PCIe Cryptographic Module only and do not include any security policy that may be enforced by the host appliance or server.

1.3 Security Level

The cryptographic module meets all Level 3 security requirements for [FIPS 140-2] as summarized in the table below:

Table 1-1: FIPS 140-2 Security Levels

Security Requirements Section	Level
Cryptographic Module Specification	3
Cryptographic Module Ports and Interfaces	3
Roles and Services and Authentication	3
Finite State Machine Model	3
Physical Security	3
Operational Environment	N/A
Cryptographic Key Management	3
EMI/EMC	3
Self-Tests	3
Design Assurance	3
Mitigation of Other Attacks	3
Cryptographic Module Security Policy	3

2 Security Policy Model Introduction

2.1 Functional Overview

The SafeNet PCIe cryptographic module is a multi-chip embedded hardware cryptographic module in the form of a PCI-Express card that typically resides within a custom computing or secure communications appliance. The cryptographic module is contained in its own secure enclosure that provides physical resistance to tampering. The cryptographic boundary of the module is defined to encompass all components inside the secure enclosure on the PCIe card. Figure 2-1 depicts the SafeNet PCIe cryptographic module and Figure 2-2 depicts the SafeNet IS appliance, with the SafeNet PCIe module installed.

The module may be explicitly configured to operate in FIPS mode, or in a non-FIPS mode of operation. Configuration in FIPS mode enforces the use of FIPS-approved algorithms only. Configuration in FIPS Level 3 mode enforces the use of trusted path authentication. Note that selection of FIPS mode occurs at initialization of the cryptographic module, and cannot be changed during normal operation. SafeNet IS is configured to operate in FIPS mode only.

A cryptographic module is accessed directly (i.e., electrically) via either the Trusted Path PIN Entry Device (PED) serial interface or via the PCI-Express communications interface. A module provides secure key generation and storage for symmetric keys and asymmetric key pairs along with symmetric and asymmetric cryptographic services. Access to key material and cryptographic services for users and user application software is provided through the PKCS #11 programming interface. A module may host multiple user definitions or “partitions” that are cryptographically separated and are presented as “virtual tokens” to user applications. Each partition must be separately authenticated in order to make it available for use.

This Security Policy is specifically written for the SafeNet PCIe cryptographic module in a Trusted Path Authentication (FIPS Level 3) configuration.

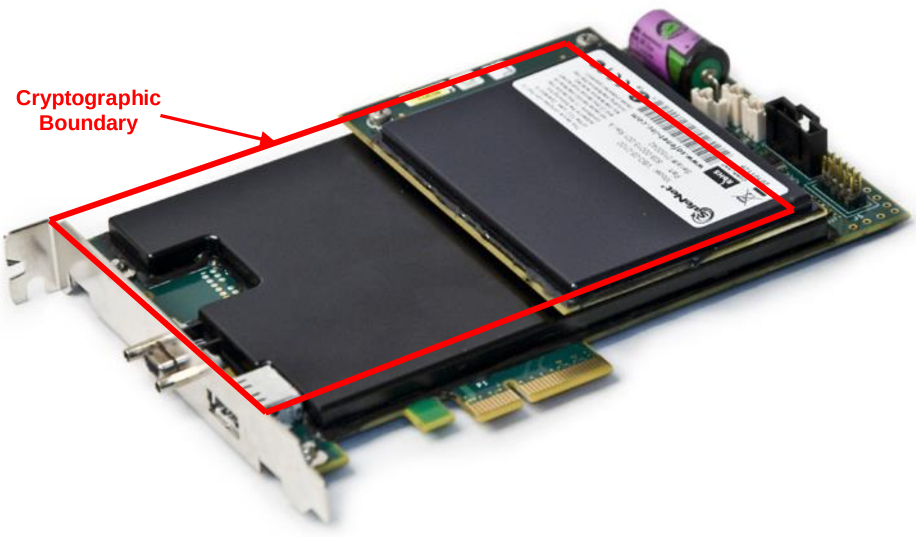

Figure 2-1. SafeNet PCIe Cryptographic Module

Figure 2-2. SafeNet IS with SafeNet PCIe Installed

2.1 Ports and Interfaces

The module supports the following ports and interfaces in the table below.Table 2-1: Mapping of FIPS 140-2 Interfaces to Physical and Logical Interfaces

FIPS 140-2 Interface	Physical Interface	Logical Interface
Data Input	PCIe interface	PKCS #11 API
	RS-232 serial port	Trusted path to Luna PED
Data Output	PCIe interface	PKCS #11 API
	RS-232 serial port	Trusted path to Luna PED
Control Input	PCIe interface	PKCS #11 API
Status Output	PCIe interface	PKCS #11 API
Power	PCIe interface	N/A

2.2 Assets to be Protected

The module is designed to protect the following assets:

- User-generated private keys;
- User-generated secret keys;
- Cryptographic services; and
- Module security critical parameters.

2.3 Operating Environment

The module is assumed to operate as a key management and cryptographic processing card within a security appliance that may operate in a TCP/IP network environment. The host appliance may be used in an internal network environment when key management security is a primary requirement. It may also be deployed in environments where it is used primarily as a cryptographic accelerator, in which case it will often be connected to external networks. It is assumed that the appliance includes an internal host computer that runs a suitably secured operating system, with an interface for use by locally connected or remote administrators and an interface to provide access to the module's cryptographic functions by application services running on the host computer. It is also assumed that only known versions of the application services are permitted to run on the internal host computer of the appliance.

It is assumed that trained and trustworthy administrators are responsible for the initial configuration and ongoing maintenance of the appliance and the cryptographic module.

It is assumed that physical access to the cryptographic module will be controlled, and that connections will be controlled either by accessing the module via a direct local connection or by accessing it via remote connections controlled by the host operating system and application service.

3 Security Policy Model Description

This section provides a narrative description of the security policy enforced by the module in its most general form. It is intended both to state the security policy enforced by the module and to give the reader an overall understanding of the security behaviour of the module. The detailed functional specification for the module is provided elsewhere.

The security behaviour of the cryptographic module is governed by the following security policies:

- Operational Policy
- Identification and Authentication Policy
- Access Control Policy
- Cryptographic Material Management Policy
- Firmware Security Policy
- Physical Security Policy

These policies complement each other to provide assurance that cryptographic material is securely managed throughout its life cycle and that access to other data and functions provided by the product is properly controlled. Configurable parameters that determine many of the variable aspects of the module's behaviour are specified by the higher level Operational Policy implemented at two levels: the cryptographic module as a whole and the individual partition. This is described in section 3.1.

The Identification and Authentication policy is crucial for security enforcement and it is described in section 3.4. The access control policy is the main security functional policy enforced by the module and is described in section 3.5, which also describes the supporting object re-use policy. Cryptographic Material Management is described in section 3.6. Firmware security, physical security and fault tolerance are described in sections 3.9 through 3.12.

3.1 Operational Policy

The module employs the concept of the Operational Policy to control the overall behaviour of the module and each of the partitions within. At each level, either the module or the partition is assigned a fixed set of "capabilities" that govern the allowed behaviour of the module or individual partition. The Security Officer (SO) establishes the Operational Policy by enabling/disabling or refining the corresponding policy elements to equate to or to be more restrictive than the pre-assigned capabilities.

The set of configurable policy elements is a proper subset of the corresponding capability set. That is, not all elements of the capability set can be refined. Which of the capability set elements have corresponding policy set elements is pre-determined based on the "personality" of the partition or manufacturing restrictions placed on the module. For example, the module capability setting for "domestic algorithms and key sizes available" does not have a corresponding configurable policy element.

There are also several fixed settings that do not have corresponding capability set elements. These are elements of the cryptographic module's behaviour that are truly fixed and, therefore, are not subject to configuration by the SO. The specific settings[1] are the following:

- Allow/disallow non-sensitive secret keys – fixed as disallow.
- Allow/disallow non-sensitive private keys – fixed as disallow.
- Allow/disallow non-private secret keys – fixed as disallow.
- Allow/disallow non-private private keys – fixed as disallow.
- Allow/disallow secret key creation through the create objects interface – fixed as disallow.
- Allow/disallow private key creation through the create objects interface – fixed as disallow.

Further, policy set elements can only refine capability set elements to more restrictive values. Even if an element of the policy set exists to refine an element of the capability set, it may not be possible to assign the policy set element to a value other than that held by the capability set element. Specifically, if a capability set element is set to allow, the corresponding policy element may be set to either enable or disable. However, if a capability set element is set to disallow, the corresponding policy element can only be set to disable. Thus, an SO cannot use policy refinement to lift a restriction set in a capability definition.

3.1.1 Module Capabilities

- The following is the set of capabilities supported at the module level:
- Allow/disallow non-FIPS algorithms available
- Allow/disallow password authentication (disallowed in Trusted Path configuration)
- Allow/disallow trusted path authentication (allowed and must be enabled in Level 3 configuration)
- Allow/disallow partition groups
- Allow/disallow cloning
- Allow/disallow masking[2]
- Allow/disallow unmasking
- Allow/disallow Korean algorithms[3]
- Allow/disallow SO reset of partition PIN[4]
- Allow/disallow network replication (set to disallow)
- Allow/disallow forcing change of User authentication data
- Allow/disallow Remote PED (RPED) operations
- Allow/disallow external Master Tamper Key (MTK) split storage
- Allow/disallow Acceleration

[1] The nomenclature used for these setting is based on PKCS#11.

[2] A Thales term used to describe the encryption of a key for use only within a SafeNet cryptographic module.

[3] Korean algorithms include SEED, ARIA, and KCDSA.

[4] In this instance PIN is used generically to represent a Personal Identification Number or a password.

- Allow/disallow FW5 compatibility mode
- Allow/disallow remote authentication
- Allow/disallow offboard storage
- Allow/disallow ECIES support
- Allow/disallow force single domain
- Allow/disallow unified PED key
- Allow/disallow M of N
- Allow/disallow small form factor backup/restore
- Allow/disallow Secure Trusted Channel
- Allow/disallow decommission on tamper
- Allow/disallow partition re-initialize

3.1.2 Partition Capabilities

The following is the set of capabilities supported at the partition level. All capability elements described as “allow/disallow some functionality” are Boolean values where false (or “0”) equates to disallow the functionality and true (or “1”) equates to allow the functionality. The remainder of the elements are integer values of the indicated number of bits.

Allow/disallow changing of certain key attributes once a key has been created.

Allow/disallow user key management capability. (This would be disabled by the SO at the policy level to prevent any key management activity in the partition, even by a user in the Crypto Officer role. This could be used, for example, at a CA once the root signing key pair has been generated and backed up, if appropriate, to lock down the partition for signing use only.)

- Allow/disallow incrementing of failed login attempt counter on failed challenge response validation (Ignore failed challenge responses).
- Allow/disallow activation.
- Allow/disallow automatic activation (auto-activation).
- Allow/disallow High Availability (HA) recovery.
- Allow/disallow multipurpose keys.
- Allow/disallow operation without RSA blinding.
- Allow/disallow signing operations with non-local keys.
- Allow/disallow raw RSA operations.
- Allow/disallow private key wrapping
- Allow/disallow private key unwrapping.
- Allow/disallow secret key wrapping
- Allow/disallow secret key unwrapping
- Allow/disallow RSA signing without confirmation

- Number of failed Partition User logins allowed before partition is locked out/cleared (default is 15; SO can configure it to be 3 ≤ N ≤ 15)
- Minimum/maximum PIN length
- Allow/disallow remote authentication[5]
- Allow/disallow RSA PKCS mechanism
- Allow/disallow CBC-PAD (un)wrap keys of any size
- Allow/disallow private key SFF backup/restore
- Allow/disallow secret key SFF backup/restore
- Allow/disallow Secure Trusted Channel

The following capabilities are configurable only if the corresponding capability/policy is allowed and enabled at the module level:

- Allow/disallow private key cloning.
- Allow/disallow secret key cloning.
- Allow/disallow private key masking[6].
- Allow/disallow secret key masking.
- Allow/disallow private key unmasking.
- Allow/disallow secret key unmasking.

The following tables summarize the module and partition capabilities, showing typical capability settings for SafeNet PCIe cryptographic modules configured for use in SafeNet IS. An X indicates the default capability setting for each configuration of the module.

Table 3-1. Module Capabilities and Policies

Description	Capability	IS	Policy	Comments
Non-FIPS algorithms available	Allow	X	Enable Disable	SO can configure the policy to enable or disable the availability of non-FIPS algorithms at the time the cryptographic module is initialized.
	Disallow		Disable	The cryptographic module must operate using FIPS-approved algorithms only. Must be disabled in FIPS mode
Password authentication	Allow		Enable Disable	SO can configure the policy to enable or disable the use of passwords without trusted path for authentication.
	Disallow	X	Disable	The cryptographic module must operate using the trusted path and module-generated secrets for authentication.
Trusted path authentication	Allow	X	Enable Disable	SO can configure the policy to enable or disable the use of the trusted path and module-generated secrets for authentication.
	Disallow		Disable	The cryptographic module must operate using passwords without trusted path for authentication.[7]

[5] Remote authentication was a legacy capability now replaced with remote PED

[6] Key masking is a SafeNet product feature that provides encrypted key output via key wrapping. This Key masking utilizes an Approved Key Transport Scheme.

[7] One and only one means of authentication ("user password" or "trusted path") must be enabled by the policy. Therefore, one of the authentication capabilities must be allowed and, if one of the capabilities is disallowed or the policy setting disabled, then the policy setting for the other must be enabled.

Description	Capability	IS	Policy	Comments
Partition groups	Allow	X	Enable	SO can configure the policy to enable or disable groups of partitions, such that members of a group share the same PED authentication data and configuration options.
			Disable	
	Disallow		Disable	The module cannot have groups of partitions.
Remote PED Operations	Allow	X	Enable	The cryptographic module can use Remote PED for Trusted Path authentication.[8] Allowed in Trusted Path authentication only.
			Disable	
	Disallow		Disable	The cryptographic module cannot use remote PED for Trusted Path authentication.
Cloning	Allow	X	Enable	SO can configure the policy to enable or disable the availability of the cloning function for the cryptographic module as a whole.
			Disable	
	Disallow		Disable	The cryptographic module must operate without cloning.
Masking	Allow	X	Enable	SO can configure the policy to enable or disable the availability of the masking function for the cryptographic module as a whole.
			Disable	
	Disallow		Disable	The cryptographic module must operate without masking.
Unmasking	Allow	X	Enable	SO can configure the policy to enable or disable the availability of the unmasking function for the cryptographic module as a whole.
			Disable	
	Disallow		Disable	The cryptographic module must operate without unmasking.
Korean algorithms[9]	Allow		Enable	SO can configure the policy to enable or disable the availability of Korean algorithms for the cryptographic module as a whole.
			Disable	
	Disallow	X	Disable	The cryptographic module must operate without Korean algorithms.
SO reset of partition PIN	Allow	X	Enable	SO can configure the policy to enable a partition PIN to be reset if it is locked as a result of exceeding the maximum number of failed login attempts.
			Disable	
	Disallow		Disable	A partition cannot reset the partition PIN and must be re-created as a result of exceeding the maximum number of failed login attempts.
Network Replication	Allow		Enable	SO can configure the policy to enable the replication of the module's key material over the network to a second module.
			Disable	
	Disallow	X	Disable	The module cannot be replicated over the network.
Force user PIN change	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled, it forces the user to change the PIN upon first login.
			Disable	
	Disallow		Disable	The user is never forced to change PIN on first login.
External MTK split storage	Allow		Enable	This capability is set prior to shipment to the customer. It allows the use of external storage of the MTK split.
			Disable	
	Disallow	X	Disable	External MTK split storage cannot be enabled for the module.
Acceleration	Allow	X	Enable	This capability is set prior to shipment to the customer. It allows the use of the onboard crypto accelerator.
			Disable	
	Disallow		Disable	Remote authentication cannot be enabled for the module.
FW5	Allow	X	Enable	Allows the use of the FW5 compatibility mode. The compatibility mode allows the cryptographic module to use the legacy Token Wrapping Certificate (TWC) to communicate with the installed base of legacy units in the field. It is always set to Allow at the factory.
			Disable	
	Disallow		Disallow	FW5 compatibility mode cannot be enabled for the module.
Remote Authentication	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows remote authentication.
			Disable	
	Disallow		Disallow	Remote Authentication cannot be enabled for the module.
Offboard Storage	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows the use of a KTS Approved key masking method for encrypted offboard storage.
			Disable	
	Disallow		Disallow	Offboard storage cannot be enabled for the module.
ECIES Support	Allow		Enable	This capability is set prior to shipment to the customer. If enabled it allows support for ECIES.
			Disable	

[8] Enabled in the Trusted Path configuration. Operator can connect the cryptographic module to a Remote PED using Command Line Interface (CLI) commands.

[9] Korean algorithms are only available upon customer request.

Description	Capability	IS	Policy	Comments
	Disallow	X	Disallow	ECIES support cannot be enabled for the module.
Force Single Domain	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows the forcing of a single domain for a module.
			Disable	
	Disallow		Disallow	The module cannot force a single domain.
Unified PED Key	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows the creation and use of a unified PED key.
			Disable	
	Disallow		Disallow	Unified PED key cannot be enabled for the module.
M of N	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows the use of M of N keys. If disabled, M and N are set to 1.
			Disable	
	Disallow		Disallow	M of N cannot be enabled for the module.
Small Form Factor Backup / Restore	Allow		Enable	This capability is set prior to shipment to the customer. If enabled it allows the use of small form factor backup and restore.
			Disable	
	Disallow	X	Disallow	Small form factor backup/restore cannot be enabled for the module.
Secure Trusted Channel	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows the use of the secure trusted channel.
			Disable	
	Disallow		Disallow	Secure trusted channel cannot be enabled for the module.
Decommission on Tamper	Allow		Enable	This capability is set prior to shipment to the customer. If enabled it allows decommission on tamper.
			Disable	
	Disallow	X	Disallow	Decommission on tamper cannot be enabled for the module.
Partition Re-initialize	Allow	X	Enable	This capability is set prior to shipment to the customer. If enabled it allows a partition to be re-initialized.
			Disable	
	Disallow		Disallow	Partition re-initialize cannot be enabled for the module.

Table 3-2. Partition Capabilities and Policies

Description	Prerequisite	Capability	IS	Policy	Comments
User key management capability[10]	Trusted path authentication enabled, Trusted Path operation without a challenge disabled	Allow	X	Enable	SO can configure the policy to enable the normal PKCS #11 user role to perform key management functions. If enabled, the Crypto Officer key management functions are available. If disabled, only the Crypto User role functions are accessible.
				Disable	
		Disallow		Disable	Only the Crypto User role functions are accessible.
Count failed challenge-response validations (Ignore failed challenge responses)	Trusted path authentication enabled	Allow	X	Enable	SO can configure the policy to count failures of the challenge-response validation against the maximum login failures or not. Must be enabled if either activation or auto-activation is enabled
				Disable	
		Disallow		Disable	Failures of the challenge-response validation are not counted against the maximum login failures.
Activation	Trusted path authentication enabled	Allow	X	Enable	SO can configure the policy to enable the authentication data provided via the PED trusted path to be cached in the module, allowing all subsequent access to the partition, after the first login, to be done on the basis of challenge-response validation alone.
				Disable	
		Disallow		Disable	PED trusted path authentication is required for every access to the partition.

[10] This capability/policy is intended to offer customers a greater level of control over key management functions. By disabling the policy, the Security Officer places the partition into a state in which the key material is locked down and can only be used by connected applications, i.e., only Crypto User access is possible.

Description	Prerequisite	Capability	IS	Policy	Comments
Auto-activation	Trusted path authentication enabled	Allow	X	Enable	SO can configure the policy to enable the activation data to be stored on the appliance server in encrypted form, allowing the partition to resume its authentication state after a re-start. This is intended primarily to allow partitions to automatically re-start operation when the appliance returns from a power outage.
				Disable	
		Disallow		Disable	Activation data cannot be externally cached.
High Availability Recovery	N/A	Allow	X	Enable	SO can configure the policy to enable the use of the High Availability recovery feature.
				Disable	
		Disallow		Disable	High Availability recovery cannot be enabled.
Multipurpose keys	N/A	Allow	X	Enable	SO can configure the policy to enable the use of keys for more than one purpose, e.g., an RSA private key could be used for digital signature and for decryption for key transport purposes.
				Disable	
		Disallow		Disable	Keys can only be used for a single purpose.
Change attributes	N/A	Allow	X	Enable	SO can configure the policy to enable changing key attributes.
				Disable	
		Disallow		Disable	Key attributes cannot be changed.
Operate without RSA blinding	N/A	Allow	X	Enable	SO can configure the use of blinding mode for RSA operations. Blinding mode is used to defeat timing analysis attacks on RSA digital signature operations, but it also imposes a significant performance penalty on the signature operations.
				Disable	
		Disallow		Disable	Blinding mode is not used for RSA operations.
Signing with non-local keys	N/A	Allow	X	Enable	SO can configure the ability to sign with externally-generated private keys that have been imported into the partition.
				Disable	
		Disallow		Disable	Externally-generated private keys cannot be used for signature operations.
Raw RSA operations	N/A	Allow	X	Enable	SO can configure the ability to use raw (no padding) format for RSA encrypt/decrypt operations for key transport purposes.
				Disable	
		Disallow		Disable	Raw RSA cannot be used.
Private key wrapping	N/A	Allow		Enable	SO can configure the ability to wrap private keys for export.
				Disable	
		Disallow	X	Disable	Private keys cannot be wrapped and exported from the partition.
Private key unwrapping	N/A	Allow	X	Enable	SO can configure the ability to unwrap private keys and import them into the partition.
				Disable	
		Disallow		Disable	Private keys cannot be unwrapped and imported into the partition.
Secret key wrapping	N/A	Allow	X	Enable	SO can configure the ability to wrap secret keys and export them from the partition.
				Disable	
		Disallow		Disable	Secret keys cannot be wrapped and exported from the partition.
Secret key unwrapping	N/A	Allow	X	Enable	SO can configure the ability to unwrap secret keys and import them into the partition.
				Disable	
		Disallow		Disable	Secret keys cannot be unwrapped and imported into the partition.
Private key cloning	Cloning enabled, Trusted path authentication enabled	Allow		Enable	SO can configure the ability to clone private keys from one module and partition to another.
				Disable	
		Disallow	X	Disable	Private keys cannot be cloned.
Secret key cloning	Cloning enabled, Trusted path	Allow		Enable	SO can configure the ability to clone secret keys from one module and partition to another.
				Disable	

Description	Prerequisite	Capability	IS	Policy	Comments
	authentication enabled	Disallow	X	Disable	Secret keys cannot be cloned.
Private key masking	Masking enabled	Allow	X	Enable	SO can configure the ability to mask private keys for storage outside the partition.
				Disable	
		Disallow		Disable	Private keys cannot be masked for storage outside the partition.
Secret key masking	Masking enabled	Allow	X	Enable	SO can configure the ability to mask secret keys for storage outside the partition.
				Disable	
		Disallow		Disable	Secret keys cannot be masked for storage outside the partition.
Private key unmasking	Unmasking enabled	Allow	X	Enable	This setting allows unmasking of private keys.
				Disable	
		Disallow		Disable	Private keys cannot be unmasked
Secret key unmasking	Unmasking enabled	Allow	X	Enable	This setting allows unmasking of secret keys.
				Disable	
		Disallow		Disable	Secret keys cannot be unmasked
Minimum / maximum password length		7-255 characters		Configurable	The SO can configure the minimum password length, but minimum length must always be ≥ 7.
Number of failed Partition User logins allowed	N/A	Minimum:3, Maximum:15		Configurable	The SO can configure; default maximum value is 15.
RSA signing without confirmation	N/A	Allow	X	Enable	SO can configure internal RSA signature verification in the module before returning the signature to the caller.
				Disable	
		Disallow		Disable	Internal RSA signature verification cannot be turned on.
Remote Authentication	Remote authentication enabled	Allow	X	Enable	SO can configure the ability to allow remote authentication.
				Disable	
		Disallow		Disable	Remote authentication cannot be enabled.
RSA PKCS Mechanism	N/A	Allow	X	Enable	SO can configure the ability to allow the use of RSA PKCS mechanisms.
				Disable	
		Disallow		Disable	RSA PKCS Mechanism cannot be enabled.
CBC-PAD (Un)Wrap Keys of Any Size	N/A	Allow	X	Enable	SO can configure the ability to allow the unwrapping using CBC-PAD for keys of any size.
				Disable	
		Disallow		Disable	CBC-PAD (un)wrap cannot be enabled.
Private Key SFF Backup/Restore	SFF backup/ restore enabled	Allow		Enable	SO can configure the ability to backup/restore private keys using SFF.
				Disable	
		Disallow	X	Disable	SFF backup/restore of private keys cannot be enabled.
Secret Key SFF Backup/Restore	SFF backup/ restore enabled	Allow		Enable	SO can configure the ability to backup/restore secret keys using SFF.
				Disable	
		Disallow	X	Disable	SFF backup/restore of secret keys cannot be enabled.
Secure Trusted Channel	STC enabled	Allow	X	Enable	SO can configure the ability to use a Secure Trusted Channel.
				Disable	
		Disallow		Disable	Secure Trusted Channel cannot be enabled.

3.2 FIPS-Approved Mode

The SO controls operation of a module in FIPS-approved mode, as defined by FIPS PUB 140-2, by enabling or disabling the appropriate Module Policy settings (assuming each is allowed at the Module Capability level). To operate in FIPS-approved mode, the following policy settings are required:

- "Non-FIPS Algorithms Available" must be disabled.

Additionally, for operation at **FIPS Level 3**:

- "Trusted path authentication" must be enabled (implies that password authentication is disallowed or disabled),
- "Count failed challenge – response validations" must be enabled if activation or auto-activation is enabled, and
- Raw RSA operations can only be used for key transport in FIPS mode
- Disable "remote authentication"

3.3 Description of Operator, Subject and Object

3.3.1 Operator

An operator is defined as an entity that acts to perform an operation on a module. An operator may be directly mapped to a responsible individual or organization, or it may be mapped to a composite of a responsible individual or organization plus an agent (application program) acting on behalf of the responsible individual or organization.

In the case of a Certification Authority (CA), for example, the organization may empower one individual or a small group of individuals acting together to operate a cryptographic module as part of the company's service. The operator might be that individual or group, particularly if they are interacting with a module locally. The operator might also be the composite of the individual or group, who might still be present locally to a module (particularly for activation purposes, see section 3.4.2), plus the CA application running on a network-attached host computer.

3.3.2 Roles

In the Trusted Path Authentication configuration, the SafeNet cryptographic module supports the following **authenticated** operator roles: The **Security Officer**[11] (SO) at the module level plus Partition **Users**[12] (also known by sub-roles – Crypto Officer and Crypto User) for each Partition. The cryptographic module also supports one **unauthenticated** operator role, the Public User, primarily to permit access to status information and diagnostics before authentication.

The SO is a privileged role, which exists only at the module level, whose primary purpose is to initially configure a module for operation and to perform security administration tasks such as partition creation.

The Crypto Officer is the key management role for each partition and the Crypto User is an optional read-only role that limits the operator to performing cryptographic operations only.

[11] Within the confines of the operational use of the SafeNet cryptographic module, the term "Security Officer" is equivalent to the FIPS 140-2 term of "**Crypto Officer**".

[12] Within the confines of the operational use of the SafeNet cryptographic module, the FIPS 140-2 term of "**User**" encompasses the SafeNet cryptographic module roles of "crypto user" and "crypto officer", which are collectively called the Partition **Users**.

For an operator to assume any role other than Public User, the operator must be identified and authenticated. The following conditions must hold in order to assume one of the authenticated roles:

- No operator can assume the Crypto Officer, Crypto User or Security Officer role before identification and authentication;
- No identity can assume the Crypto Officer or Crypto User plus the Security Officer role.

For additional information regarding roles and authorized services, please refer to Table 4-1 and Table 4-3.

3.3.3 Account Data

The module maintains the following User (which can include both the Crypto Officer and Crypto User role per Partition[13]) and SO account data:

- Partition ID or SO ID number.
- Partition User encrypted or SO encrypted authentication data (checkword).
- Partition User authentication challenge secret (one for each role, as applicable).
- Partition User locked out flag.

An authenticated User is referred to as a Partition User. The ability to manipulate the account data is restricted to the SO and the Partition User. The specific restrictions are as described below:

1. *Only the Security Officer role can create (initialize) and delete the following security attributes:*
    - Partition ID.
    - Checkword.
2. *If "SO reset of partition PIN" is allowed and enabled, the SO role only can modify the following security attribute:*
    - Locked out flag for Partition User.
3. *Only the Partition User can modify the following security attribute:*
    - Checkword for Partition User.
4. *Only the Security Officer role can change the default value, query, modify and delete the following security attribute:*
    - Checkword for Security Officer.

3.3.4 Subject

For purposes of this security policy, the subject is defined to be a module session. The session provides a logical means of mapping between applications connecting to a module and the processing of commands within a module. Each session is tracked by the Session ID, the Partition ID and the Access ID, which is a unique ID associated with the application's connection. It is possible to have multiple open sessions with a module associated with the same Access ID/Partition ID combination. It is also possible for a module to have sessions opened for more than one Partition ID or have multiple Access IDs with sessions opened on a module. Applications running on remote host systems that require data and cryptographic services from a module must first connect via the communications service within the appliance, which will establish the unique Access ID for the connection and then allow the application to open a session with one of the partitions within a module. A

[13] A Partition effectively represents an identity within the module.

local application (e.g., command line administration interface) will open a session directly with the appropriate partition within a module without invoking the communications service.

3.3.5 Operator – Subject Binding

An operator must access a partition through a session. A session is opened with a partition in an unauthenticated state and the operator must be authenticated before any access to cryptographic functions and Private objects within the partition can be granted. Once the operator is successfully identified and authenticated, the session state becomes authenticated and is bound to the Partition User represented by the Partition ID, in the Crypto Officer or Crypto User role. Any other sessions opened with the same Access ID/Partition ID combination will share the same authentication state and be bound to the same Partition User.

3.3.6 Object

An object is defined to be any formatted data held in volatile or non-volatile memory on behalf of an operator. For the purposes of this security policy, the objects of primary concern are private (asymmetric) keys and secret (symmetric) keys.

3.3.7 Object Operations

Object operations may only be performed by a Partition User. The operations that may be performed are limited by the role (Crypto Officer or Crypto User) associated with the user's login state, see section 3.5. New objects can be made in several ways. The following list identifies operations that produce new objects:

- Create;
- Copy;
- Generate;
- Unwrapping; and
- Derive.

Existing objects can be modified and deleted. The values of a subset of attributes can be changed through a modification operation. Objects can be deleted through a destruction operation. Constant operations do not cause creation, modification or deletion of an object. These constant operations include:

- Query an object's size;
- Query the size of an attribute;
- Query the value of an attribute;
- Use the value of an attribute in a cryptographic operation;
- Search for objects based on matching attributes;
- Cloning an object;
- Wrapping an object; and
- Masking and unmasking an object.

Secret keys and private keys are always maintained as Sensitive objects and, therefore, they are permanently stored with the key value encrypted to protect its confidentiality. Key objects held in volatile memory do not have their key values encrypted, but they are subject to active zeroization in the event of a module reset or in response to a tamper event. For additional information about the clearing of sensitive data, see Section 3.13. Operators are not given direct access to key values for any purpose.

3.4 Identification and Authentication

3.4.1 Authentication Data Generation and Entry

The module requires that Partition Users and the SO be authenticated by proving knowledge of a secret shared by the operator and the module. A module configured for Trusted Path Authentication must be initialized using the PED to define the SO authentication data.

For Trusted Path Authentication, a module generates the authentication secret as a 48-byte random value and, optionally for a Partition User, an authentication challenge secret. The authentication secret(s) are provided to the operator via a physically separate trusted path, described in sub-section 3.4.2, and must be entered by the operator via the trusted path and via a logically separate trusted channel (in the case of the response based on the challenge secret) during the login process. If a Partition is created with Crypto Officer and Crypto User roles, a separate challenge secret is generated for each role.

The following types of iKey are used with the SafeNet PED:

- Orange (RPV) iKey – for the storage of the Remote PED Vector (RPV),
- Blue (SO) iKey – for the storage of SO authentication data,
- Black (User) iKey – for the storage of User authentication data,
- Red (Domain) iKey – for the storage of the cloning domain data, used to control the ability to clone from a cryptographic module to a backup module,./'..

Figure 3-1 depicts the SafeNet PED and the associated iKeys.

Any iKey, once data has been written to it, is an Identification and Authentication device and must be safeguarded accordingly by the administrative or operations staff responsible for the operation of the module within the customer's environment.

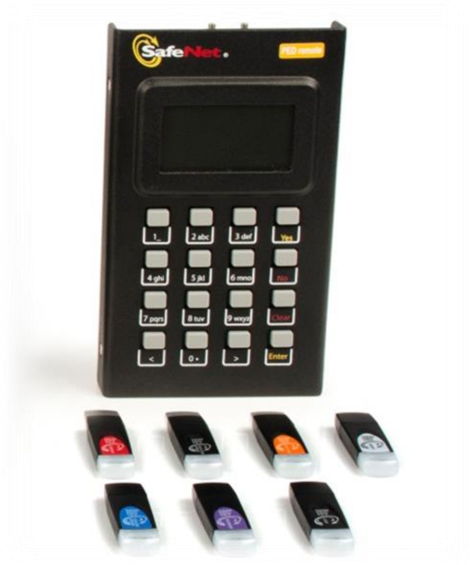

Figure 3-1. SafeNet PED and iKeys

3.4.2 Trusted Path

In Trusted Path mode, user authentication is, by default, a two-stage process. The first stage is termed "Activation" and is performed using a trusted path device (PED) which connects to the cryptographic module either directly over a physical wire or remotely over a secure network connection. The primary form of authentication data used during Activation is the 48-byte value that is randomly generated by a module and stored on the Black (User) iKey[14] via the trusted path. The data on the iKey must then be entered into a module via the trusted path as part of each Activation process. Once Activation has been performed, the user's Partition data is ready for use within a module. Access to key material and cryptographic services, however, is not allowed until the second stage of authentication, "User Login", has been performed. This typically requires the input of a partition's challenge secret as part of a login operation. However, for SO authentication the presentation of the iKey data (i.e., equivalent to Activation) is all that is required to complete authentication.

The default Partition Policy enables the use of challenge/response authentication for the "User Login" stage. The authentication challenge secret (or secrets if the Crypto Officer and Crypto User roles are used) for the partition is generated by the module as a 75-bit value that is displayed as a 16-character alphanumeric string on the visual display of the trusted path device. The challenge secret is then provided, via a secure out-of-band means, to each external entity authorized to connect to the partition and is used by the external entity to form the response to a random one-time challenge from a module. The encrypted one-time response is returned to the cryptographic module where it is verified to confirm the "User Login". Thus, when the challenge secret is required, both the trusted path Activation and the successful completion of the challenge/response process by the external entity is required to authenticate to a partition and have access to its cryptographic material and functions.

3.4.2.1 Remote PED Operation

The user has the option of operating the PED in the conventional manner (i.e., locally connected to the cryptographic module) or remotely, connected to a management workstation via USB. Remote PED operation extends the physical trusted path connection by the use of a protocol that authenticates both the remote PED and the module and establishes a one-time AES key to encrypt the communications between the module and the Remote PED. Once secure communications have been established, all interactions between the cryptographic module, PED and iKeys are performed in exactly the same way as they would be when locally connected.

The logical path between the module and the Remote PED is secured in the manner described below.

At the time it is initialized, the module generates a random 256-bit secret, known as the Remote PED Vector (RPV), stores it in its secure parameters area, and writes it to the "Orange" iKey, also known as the Remote PED Key (RPK).

To establish the secure connection, the RPK must be inserted into the PED. The PED extracts the RPV, and the PED and the cryptographic module then participate in an ephemeral Diffie-Hellman key agreement session. The derived shared secret is then XORed with the RPV to produce the key to be used for the session. An exchange of encrypted random nonces is performed to authenticate both ends of the transmission. All traffic between the PED and the cryptographic module is encrypted using AES 256.

[14] Or Black (User) PED key. Within this document the terms "iKey" and "PED" key are interchangeable unless otherwise indicated.

3.4.3 M of N Authentication

The SafeNet cryptographic module supports the use of an **M of N secret sharing** authentication scheme for each of the module roles. M of N authentication provides the capability to enforce multi-person integrity over the functions associated with each role.

The module's capabilities prevent the use of M of N for SafeNet IS. This capability is disabled and cannot be changed.

3.4.4 Limits on Login Failures

The module also implements a maximum login attempts policy. The policy differs for an SO authentication data search and a Partition User authentication data search.

In the case of an SO authentication data search:

- If "m" consecutive SO logon attempts fail, a module is zeroized. "m" is set at the time the cryptographic module is initialized, and can be modified by the SO. The valid range is 3-10.

In the case of a Partition User authentication data search:

- "SO reset of partition PIN" is Allowed and Enabled, so if "n" consecutive operator logon attempts fail, the module flags the event in the Partition User's account data, locks the Partition User and clears the volatile memory space. The SO must unlock the partition in order for the Partition User to resume operation. "n" is set at the time the cryptographic module is initialized, and can be modified by the SO. The valid range is 3-15.

3.5 Access Control

The Access Control Policy is the main security function policy enforced by a module. It governs the rights of a subject to perform privileged functions and to access objects stored in a module. It covers the object operations detailed in section 3.3.7.

A subject's access to objects stored in a module is mediated on the basis of the following subject and object attributes:

- Subject attributes:
  - Session ID
  - Access ID and Partition ID associated with session
  - Session authentication state (binding to authenticated Partition identity and role)
- Object attributes:
  - **Owner.** A Private object is owned by the Partition User associated with the subject that produces it. Ownership is enforced via internal key management.
  - **Private.** If True, the object is Private. If False, the object is Public.
  - **Sensitive.** If True, object is Sensitive. If False, object is Non-Sensitive.
  - **Extractable[15].** If True, object may be extracted. If False, object may not be extracted.

[15] Extract means to remove the key from the control of the module. This is typically done using the Wrap operation, but the Mask operation is also considered to perform an extraction when cloning is enabled for the container. The Wrap Operation is not available in FIPS mode.

- **Modifiable.** If True, object may be modified. If False, object may not be modified.

Objects are labelled with a number corresponding to their partition and are only accessible by a subject associated with the owning Partition ID. Only generic data and certificate objects can be non-sensitive. Private key and secret key objects are always created as Sensitive, Private objects. Sensitive objects are encrypted using the partition's secret key to prevent their values from ever being exposed to external entities. Private objects can only be used for cryptographic operations by a logged in Partition User. Key objects that are marked as extractable may be exported from a module using the Wrap operation if allowed and enabled in the partition's policy set. Table 3-3 summarizes the object attributes used in Access Control Policy enforcement.

Table 3-3. Object Attributes Used in Access Control Policy Enforcement

Attribute	Values	**Impact**
PRIVATE	TRUE – Object is private to (owned by) the operator identified as the Access Owner when the object is created.	Object is only accessible to subjects (sessions) bound to the operator identity that owns the object.
	FALSE – Object is not private to one operator identity.	Object is accessible to all subjects associated with the partition in which the object is stored.
SENSITIVE	TRUE – Attribute values representing plaintext key material are not permitted to exist (value encrypted).	Key material is stored in encrypted form.
	FALSE – Attribute values representing plaintext data are permitted to exist.	Plaintext data is stored with the object and is accessible to all subjects otherwise permitted access to the object.
MODIFIABLE	TRUE – The object's attribute values may be modified.	The object is "writeable" and its attribute values can be changed during a copy or set attribute operation.
	FALSE – The object's values may not be modified.	The object can only be read and only duplicate copies can be made.
EXTRACTABLE	TRUE – Key material stored with the object may be extracted from the SafeNet cryptographic module using the Wrap operation.	The ability to extract a key permits sharing with other crypto modules and archiving of key material.
	FALSE – Key material stored with the object may not be extracted from the SafeNet cryptographic module.	Keys must never leave a module's control.

The module does not allow any granularity of access other than owner or non-owner (i.e., a Private object cannot be accessible by two Partition Users and restricted to other Partition Users). Ownership of a Private object gives the owner access to the object through the allowed operations but does not allow the owner to assign a subset of rights to other operators. Allowed operations are those permitted by the cryptographic module and Partition Capability and Policy settings.

The policy is summarized by the following statements:

- A subject may perform an allowed operation on an object if the object is in the partition with which the subject is associated and one of the following two conditions holds:
  1. The object is a "Public" object, i.e., the PRIVATE attribute is FALSE, or
  2. The subject is bound to the Partition User that owns the object.
- Allowed operations are those permitted by the object attribute definitions within the following constraints:
  1. A Partition User in the Crypto User role has access to only the User operations, and
  2. The restrictions imposed by the cryptographic module and Partition Capability and Policy settings.

3.5.1 Object Protection

The module cryptographically protects the values of sensitive objects stored in its internal flash memory. Sensitive values protected using AES 256 bit encryption with three different keys – each having a separate protection role. The three keys used to protect sensitive object values are the following:

- **User Storage Key (USK)** – this key is created by the cryptographic module when the User or SO is created. It is used to maintain cryptographic separation between users' keys.
- **Master Tamper Key (MTK) –** this key is securely stored in the battery-backed RAM. It encrypts keys as they are generated to ensure that they can only be used by the co-processor itself or with authorization from it.
- **Key Encryption Key (KEK) –** this key is stored in battery-backed RAM in the module. It also encrypts all sensitive object values and is used to provide the "decommissioning" feature. The KEK is erased in response to an external decommission signal. This provides the capability to prevent access to sensitive objects in the event that the module has become unresponsive or has lost access to primary power.

3.5.2 Object Re-use

The access control policy is supported by an object re-use policy. The object re-use policy requires that the resources allocated to an object be cleared of their information content before they are re-allocated to a different object.

3.5.3 Privileged Functions

The module shall restrict the performance of the following functions to the SO role only:

- Module initialization
- Partition creation and deletion
- Configuring the module and partition policies
- Firmware update

3.6 Cryptographic Material Management

Cryptographic material (key) management functions protect the confidentiality of key material throughout its life-cycle. The FIPS PUB 140-2 approved key management functions provided by the module are the following:

1. Deterministic Random Bit Generation (DRBG) in accordance with NIST SP 800-90A section 10.2.1.
2. Cryptographic key generation in accordance with the following indicated standards:
    a. RSA 2048-4096 bits key pairs in accordance with FIPS PUB 186-4 and ANSI X9.31.
    b. Triple-DES 192 bits (SP 800-67).
    c. AES 128, 192, 256 bits (FIPS PUB 197).
    d. DSA 2048 and 3072 bit key pairs in accordance with FIPS PUB 186-4.

   e. Elliptic Curve key pairs (curves in accordance with SP 800-57) in accordance with FIPS PUB 186-4.
   f. Diffie-Hellman key pairs.
   g. Key Derivation in accordance with NIST SP 800-108 (Counter mode).
3. Diffie-Hellman (2048 bits) (key agreement; key establishment methodology provides 112 bits of encryption strength.[16])
4. EC Diffie-Hellman (ECDH) (curves in accordance with SP 800-57) key establishment in accordance with NIST SP 800-56A.
5. Symmetric key unwrap: Triple-DES 192 bits and AES 128, 192 and 256 bits in accordance with PKCS #11 (key transport provides 112 bits of security strength with Triple-DES and between 128 and 256 bits of security strength with AES).
6. Asymmetric key wrap / unwrap: RSA 2048 – 4096 (PKCS #1 V1.5 and OAEP) (key transport provides between 112 and 150 bits of security strength).
7. Encrypted key storage (using AES 256 bit encryption, see Section 3.5.1) and key access following the PKCS #11 standard.
8. Destruction of cryptographic keys is performed in one of three ways as described below in accordance with the PKCS #11 and FIPS PUB 140-2 standards:
   a. An object on a SafeNet cryptographic module that is destroyed using the PKCS #11 function C_DestroyObject is marked invalid and remains encrypted with the Partition User's key or a SafeNet cryptographic module's general secret key until such time as its memory locations (flash or RAM) are re-allocated for additional data on a SafeNet cryptographic module, at which time they are purged and zeroized before re-allocation.
   b. Objects on a SafeNet cryptographic module that are destroyed as a result of authentication failure are zeroized (all flash blocks in the Partition User's memory turned to 1's). If it is an SO authentication failure, all flash blocks used for key and data storage on a SafeNet cryptographic module are zeroized.
   c. Objects on a SafeNet cryptographic module that are destroyed through C_InitToken (the SO-accessible command to initialize a SafeNet cryptographic module available through the API) are zeroized, along with the rest of the flash memory being used by the SO and Partition Users.

Keys are always stored as secret key or private key objects with the Sensitive attribute set. The key value is, therefore, stored in encrypted form using the owning Partition User's Storage Key (USK) and the Master Tamper Key (MTK) stored in the battery-backed RAM. Access to keys is never provided directly to a calling application. A handle to a particular key is returned that can be used by the application in subsequent calls to perform cryptographic operations.

Private key and secret key objects may be imported into a module using the Unwrap, Unmask (if cloning and unmasking are enabled at the module level) or Derive operation under the control of the Access Control Policy. Any externally-set attributes of keys imported in this way are ignored by a module and their attributes are set by a module to values required by the Access Control Policy.

[16] Non-Approved but allowed method in FIPS mode.

3.6.1 Key Cloning

Key cloning is a SafeNet product feature that uses a one-time, 256-bit AES key as a session key to encrypt an object being transferred from one SafeNet module to another. Objects transferred using the cloning protocol may be keys, user data, or module data. The AES session encrypting key is obtained by combining the 48 byte cloning domain value (randomly generated by the module) with random one-time data generated by source and target modules and exchanged using RSA 4096-based transport.

3.6.2 Key Mask/Unmask

Key masking is a SafeNet product feature that uses a 256-bit AES key, which is unique to the module, to encrypt a key object for output in a way that ensures the key can only be imported, by unmasking, into the module from which it originally came or one that has been initialized to contain the same “master” key for the module. The key mask operation takes a key handle as input and uses the module’s validated AES implementation to create the masked key output.

The key unmask operation takes a masked (encrypted) key object as input, performs the necessary decryptions inside the module and returns a handle to the imported key.

Note that for both mask and unmask operations, the user (or calling application acting on the user’s behalf) never has access to the actual key values – only handles assigned to the key objects in the module.

3.6.3 Key Wrap/Unwrap

The key wrap operation encrypts a key value for output, using an RSA public key. Symmetric key wrapping is not available in FIPS mode.

The unwrap operation takes as input an encrypted key value and a handle to a local copy of the key that was originally used to do the wrapping. It decrypts the key value, stores it in the module as a key object, and returns the handle to the imported key.

Note that for both wrap and unwrap operations, the user (or calling application acting on the user’s behalf) never has access to the actual key values – only handles assigned to the key objects in the module.

3.7 Cryptographic Operations

Because of its generic nature, the module’s cryptographic co-processor and firmware support a wide range of cryptographic algorithms and mechanisms. The approved cryptographic functions and algorithms that are relevant to the FIPS 140-2 validation are in the following tables:

Table 3-4. Approved Security Functions for SafeXcel 3120

Approved Security Functions	Certificate No.
Symmetric Encryption/Decryption	
AES: (ECB, CBC, GCM[17]); Encrypt/Decrypt; Key Size = 128, 192, 256)	#4849
Triple-DES: (TECB, TCBC); Decrypt KO 1)	#2552
Hashing	
SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 (Byte Only)	#3988
Message Authentication Code	
HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	#3306
Asymmetric	
RSA: Key Generation 2048-3072 bits, Signature Generation 2048-4096 bits, Signature Verification 1024-4096 bits	#2691
DSA: PQG Generation 2048-3072 bits, Key Generation 2048-3072 bits, Signature Generation 2048-3072 bits, Signature Verification 1024-3072 bits	#1298
ECDSA: Key Generation P-224, P-256; Signature Generation P-224, P-256, P-384; Signature Verification P-192, P-224, P-256, P-384	#1242
Random Number Generation	
NIST SP 800-90A DRBG (CTR) AES-256	#1704

Table 3-5. Approved and Allowed Security Functions for SafeXcel 1746

Approved and Allowed Security Functions	Certificate No.
Symmetric Encryption/Decryption	
AES: (ECB, CBC, CFB8); Encrypt/Decrypt; Key Size = 128, 19/2, 256)	#4960
Triple-DES: (TECB, TCBC, TCFB8); Decrypt KO 1)	#2573
Asymmetric	
DSA: Key Generation 2048-3072 bits, Signature Generation 2048-3072 bits, Signature Verification 1024-3072 bits	#1307
ECDSA: Key Generation P-224, P-256, P-384, P-521; Signature Generation P-224, P-256, P-384, P-521, Signature Verification P-192, P-224, P-256, P-384, P-521	#1270

[17] The module generates IVs internally using the Approved DRBG which are at least 96-bits in length.

Table 3-6. Approved Security Functions for Firmware Implementation

Approved Security Functions	Certificate No.
Symmetric Encryption/Decryption	
AES: (ECB, CBC, OFB, CFB8, CFB128, CTR, GCM[17]); Encrypt/Decrypt; Key Size = 128, 192, 256	#4977
Triple-DES: (ECB, CBC, OFB, CFB8, CFB64; CMAC; CTR); Decrypt KO 1	#2574
Hashing	
SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 (Byte Only)	#4050
Message Authentication Code	
HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	#3307
AES CMAC (Key Sizes Tested: 128, 192, 256)	#4977
Asymmetric	
RSA: Key Generation 2048-3072 bits, Signature Generation 2048-4096 bits, Signature Verification 1024-4096 bits	#2692
DSA: Key Generation 2048-3072 bits, Signature Generation 2048-3072 bits, Signature Verification 1024-3072 bits	#1306
ECDSA: Key Generation, Signature Generation, Signature Verification B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521	#1266
ECDSA: Signature Generation Component B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521	CVL #1577
Key Agreement Scheme	
KAS-ECC: Ephemeral Unified (KAS Role(s): Initiator / Responder) OnePassDH (KAS Role(s): Initiator / Responder)	#151
Key Derivation	
KBKDF: NIST SP 800-108 (Counter Mode)	#163
Key Transport	
KTS (AES Cert. #4977 and HMAC Cert. #3307; key establishment methodology provides between 128 and 256 bits of encryption strength)	AES Cert. #4977 and HMAC Cert. #3307
Key Generation	
CKG[18]	Vendor Affirmed using IG D.12

Table 3-7. Allowed Security Functions for Firmware Implementation

Allowed Security Functions
Key Agreement

[18] Symmetric keys and seeds used for asymmetric key generation are an unmodified output from approved DRBG (Cert. #1704)

Allowed Security Functions
Diffie-Hellman (key agreement; key establishment methodology provides 112 bits of encryption strength)
EC Diffie-Hellman (key agreement; key establishment methodology provides between 112 and 256 bits of encryption strength)
Key Transport
RSA (key wrapping; key establishment methodology provides between 112 and 150 bits of encryption strength)
AES (key unwrapping; key establishment methodology provides between 128 and 256 bits of encryption strength)
Triple-DES (key unwrapping; key establishment methodology provides 112 bits of encryption strength)
Entropy Source
Hardware Non-Deterministic Random Number Generator (free-running local oscillators)[19]

Non-FIPS Approved security functions are not available for use when the module has been configured to operate in FIPS-approved mode, see Section 3.2.

Table 3-8. Non-FIPS Approved Security Functions

Non-FIPS Approved Security Functions
Symmetric Encryption/Decryption
DES
Triple-DES (2-Key)
RC2
RC4
RC5
CAST5
SEED
ARIA
Hashing
MD2
MD5
HAS-160
Message Authentication Code
AES MAC (non-compliant)
DES-MAC
RC2-MAC
RC5-MAC
CAST5-MAC
SSL3-MD5-MAC[20]
SSL3-SHA1-MAC[20]
HMAC (Certs. #3306 and #3307 – non-compliant less than 112 bits of encryption strength)
Triple-DES MAC
Asymmetric
KCDSA
RSA X-509
RSA (Certs. #2691 and #2692 – non compliant less than 112 bits of encryption strength)
DSA (Certs. #1298, #1306, #1307 – non-compliant less than 112 bits of encryption strength)
ECDSA (Certs. #1242, #1266, #1270 – non-compliant less than 112 bits of encryption strength)
Generate Key

[19] The module's NDRNG generates 382.83 bits of entropy for key and seed generation

[20] Used by the TLS protocol. TLS has not been reviewed or tested by the CAVP or the CMVP.

Non-FIPS Approved Security Functions
DES
RC2
RC4
RC5
CAST5
SEED
ARIA
GENERIC-SECRET
SSL PRE-MASTER
Key Agreement
ECC (non-compliant less than 112 bits of encryption strength)
Diffie-Hellman (key agreement; key establishment methodology; non-compliant less than 112 bits)
Key Transport
RSA (key wrapping; key establishment methodology; non-compliant less than 112 bits of encryption strength)

3.8 Self-tests

The module provides self-tests on power-up and on request to confirm the firmware integrity, and to check the random number generator and each of the implemented cryptographic algorithms. The module also performs conditional self-tests in accordance with FIPS 140-2, section 4.9.2.

Table 3-9. Module Self-Tests

Test	When Performed	Where Performed	Indicator
Boot loader performs a SHA-256 integrity check of the firmware prior to firmware start	Power-on	Firmware	Module halt[21]
ECDSA integrity check of the binary running on the hardware.	Power-on	Hardware	Module halt
DRBG Instantiate Function Known Answer Test (KAT)	Power-on	Hardware	Module halt
DRBG Generate Function KAT	Power-on	Hardware	Module halt
DRBG Reseed Function KAT	Power-on	Hardware	Module halt
DRBG Uninstantiate Function KAT	Power-on	Hardware	Module halt
Triple-DES KATs (e / d)	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt[22]
SHA-1 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
SHA-224 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
SHA-256 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
SHA-384 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
SHA-512 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
HMAC SHA-1 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
HMAC SHA-224 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
HMAC SHA-256 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
HMAC SHA-384 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
HMAC SHA-512 KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
RSA sig-gen KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt

[21] Details of the failure can be obtained from the dual-port following a module halt.

[22] An error message is output, the cryptographic module halts, and data output is inhibited.

Test	When Performed	Where Performed	Indicator
RSA sig-ver KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
DSA sig-gen KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
DSA sig-ver KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
Diffie-Hellman KAT	Power-on/Request	Firmware	Module halt / Error - Halt
AES KATs (e /d)	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
AES-GCM KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
ECDH KAT	Power-on/Request	Firmware	Module halt / Error - Halt
ECDSA sig-gen KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
ECDSA sig-ver KAT	Power-on/Request	Firmware / Hardware	Module halt / Error - Halt
KDF KAT	Power-on/Request	Firmware	Module halt / Error - Halt
DRBG continuous test	Continuous	Firmware / Hardware	Error - Halt
NDRNG continuous test	Continuous	Firmware / Hardware	Error - Halt
RSA – Pair-wise consistency test (asymmetric key pairs)	On generation	Firmware / Hardware	Error
DSA – Pair-wise consistency test (asymmetric key pairs)	On generation	Firmware / Hardware	Error
ECDSA – Pair-wise consistency test (asymmetric key pairs)	On generation	Firmware / Hardware	Error
Firmware load test (4096-bit RSA sig ver)	On firmware update load	Firmware	Error – module will continue with existing firmware

While the module is running Power-On Self Tests (POST) all interfaces are disabled until the successful completion of the self-tests.

3.9 Firmware Security

The Firmware Security Policy assumes that any firmware images loaded in conformance with the policy have been verified by SafeNet to ensure that the firmware will function correctly. The policy applies to initial firmware loading and subsequent firmware updates. Any firmware loaded into this module that is not shown on the module certificate, is out of the scope of this validation and requires a separate FIPS 140-2 validation.

The module shall not allow external software[23] to be loaded inside its boundary. Firmware is digitally signed using a Firmware Signing Key (FSK) held by Thales. And corresponding to the Firmware Signing Certificate (FSC) delivered as part of the firmware signing package. The FSC is signed by the Manufacturer Root Signing Key with the corresponding Public component embedded in the module at manufacture. RSA (4096 bits) PKCS #1 V1.5 with SHA-384 is used as the approved signature method.

The Boot Loader shall provide an integrity check to ensure the integrity of the firmware and to ensure the integrity of any permanent security-critical data stored within a cryptographic module.

3.10 Physical Security

The SafeNet cryptographic module is a multi-chip embedded module as defined by FIPS PUB 140-2 section 4.5. The module is enclosed in a strong metal enclosure that provides tamper-evidence. Any tampering that might compromise a module’s security is detectable by visual inspection of the physical integrity of a module.

[23] External software means any form of executable code that has been generated by anyone other than SafeNet and has not been properly formatted and signed as a legitimate SafeNet firmware image.

The Security Officer should perform a visual inspection of the module at regular intervals. Within the metal enclosure, a hard opaque epoxy covers the circuitry of the cryptographic module. Attempts to remove this epoxy will cause sufficient damage to the cryptographic module so that it is rendered inoperable.

The module's enclosure is opaque to resist visual inspection of the device design, physical probing of the device and attempts to access sensitive data on individual components of the device.

The module is designed to operate between 0° and 65° Celsius, and to sense and respond to out-of-range temperature conditions. The module also senses and responds to out-of-range voltage conditions. In the event that the module senses an out-of-range temperature or voltage, it will clear all working memory and halt operations. It can be reset and placed back into operation when proper operating conditions have been restored.

The epoxy hardness was tested at room temperature and at the high and low temperatures which would cause the active tamper (0° to 65° Celsius).

3.11 EMI / EMC

The module conforms to FCC Part 15 Class B requirements for home use.

3.12 Fault Tolerance

If power is lost to a module for whatever reason, the module shall, at a minimum, maintain itself in a state that it can be placed back into operation when power is restored without compromise of its functionality or permanently stored data.

A module shall maintain its secure state[24] in the event of data input/output failures. When data input/output capability is restored the module will resume operation in the state it was prior to the input/output failure.

3.13 Mitigation of Other Attacks

Timing attacks are mitigated directly by a module through the use of hardware accelerator chips for modular exponentiation operations. The use of hardware acceleration ensures that all RSA signature operations complete in very nearly the same time, therefore making the analysis of timing differences irrelevant. RSA blinding may also be selected as an option to mitigate this type of attack.

[24] A secure state is one in which either a SafeNet cryptographic module is operational and its security policy enforcement is functioning correctly, or it is not operational and all sensitive material is stored in a cryptographically protected form on a SafeNet cryptographic module.

4 Security Policy Checklist Tables

Table 4-1 Roles and Required Identification and Authentication

Role	Type of Authentication	Authentication Data
Security Officer	Identity-based	Level 3 – Authentication token (PED Key – one per module) plus optional PED PIN
Crypto Officer	Identity-based[25]	Level 3 – Authentication token (PED Key – one per user) plus optional PED PIN, plus optional Challenge Secret for the role[26]
Crypto User	Identity-based	Level 3 – Authentication token (PED Key – one per user) plus optional PED PIN, plus optional Challenge Secret for the role
Public User	Not required	N/A

Table 4-2 Strengths of Authentication Mechanisms

Authentication Mechanism	Strength of Mechanism
PED Key (Level 3) plus PIN	48-byte random authentication data stored on PED key plus PIN entered via PED key pad (minimum 4 bytes). It is obvious that the probability of guessing the authentication data in a single attempt is 1 in 2^{384}. With login failure thresholds of 3 to 10 for SO and configurable from 1 to 15 (default 15) for users, this ensures the FIPS 140-2 required thresholds can never be reached.
Challenge Secret (Level 3)	Default 16 character random string (minimum 7-character string). The probability of guessing the challenge secret in a single attempt is 1 in 62^7 (approximately 3.5 x 10^{12}). With login failure thresholds of 3 to 10 for SO and configurable from 1 to 10 (default 10) for users, this ensures the FIPS 140-2 required thresholds can never be reached.

All services listed in Table 4-3 can be accessed in FIPS and non-FIPS mode. The services listed in Table 4-3 use the security functions listed in Table 3-4, Table 3-5, Table 3-6, and Table 3-7. When the module is operating in FIPS-approved mode as described in Section 3.2, the Non-FIPS Approved Security Functions in Table 3-8 are disabled and cannot be used for these services. The non-Approved functions in Table 3-8 can only be accessed through the services when the module is in non-FIPS Approved mode.

Table 4-3. Services Authorized for Roles

Role	Authorized Services
Security Officer	Show Status, Self-test, Initialize Module, Configure Module Policy, Create Partition, Configure Partition Policy, Zeroize, Firmware Update, Key Generation, Key Pair

[25] The Crypto Officer and Crypto User both apply to the same partition, i.e., identity. They are distinguished by different challenge values representing the two different roles.

[26] If activation or auto-activation is enabled, challenge secret is required in FIPS mode.

Role	Authorized Services
	Generation, Symmetric Key Wrap/Unwrap, Asymmetric Key Unwrap, Symmetric Key Mask/Unmask, Symmetric Encrypt/Decrypt, Asymmetric Signature/Verification, Store Data Object, Read Data Object
Crypto Officer	Show Status, Self-test, Zeroize, Key Generation, Key Pair Generation, Symmetric Encrypt/Decrypt, Asymmetric Signature/Verification, Symmetric Key Wrap/Unwrap, Asymmetric Key Unwrap, Symmetric Key Mask/Unmask, Store Data Object, Read Data Object,
Crypto User	Show Status, Self-test, Zeroize, Symmetric Encrypt/Decrypt, Asymmetric Signature/Verification, Store Data Object, Read Data Object
Public User	Show Status, Self-test, Zeroize, Store Public Data Object, Read Public Data Object

Table 4-4 Access Rights within Services

Service	Cryptographic Keys and CSPs Access	Role
Show Status[27]	N/A	All
Self-test	N/A	All
Initialize Module	Use – DRBG State Write - Authentication data[28], PSK	SO
Configure Module Policy	N/A	SO
Create Partition	N/A	SO
Configure Partition Policy	N/A	SO
Zeroize	Erase - Authentication data, PSK, SADK, symmetric keys, asymmetric key pairs	All
Firmware Update	Use/Write – Root Certificate, FSC[29]	SO
Key Generation	Use – DRBG State, Storage Keys[30] Write - Symmetric keys	SO, Crypto Officer
Key Pair Generation	Use – DRBG State, Storage Keys Write - Asymmetric key pairs	SO, Crypto Officer
Symmetric Key Wrap/ Unwrap	Use/Write - Symmetric with RSA, Symmetric Unwrap with Symmetric ECB mode	SO, Crypto Officer
Asymmetric Key Unwrap	Use - Asymmetric with Symmetric CBC mode	SO, Crypto Officer
Symmetric Key Mask/ Unmask	Use/Write – Masking Keys	SO, Crypto Officer
Symmetric Encrypt/Decrypt	Use – DRBG State, Symmetric keys	SO, Crypto Officer, Crypto User
Asymmetric Signature	Use – Asymmetric private Keys	SO, Crypto Officer, Crypto User
Asymmetric Verification	Use – Asymmetric public Keys	SO, Crypto Officer, Crypto User

[27] Show status is provided by invoking the “hsm show” command from the administrative interface. It will display identifying information about the module such as label, serial number, firmware version, etc. The “hsm capability list” command indicates whether the module is in FIPS-approved mode.

[28] Authentication data depending on the configuration of the cryptographic module may include access and use of the following CSPs: Challenge Secret, Random Challenge, Challenge Response, PED Key Authentication Data, Optional PIN, RPV

[29] Public key value. See Table 4-5 for its description.

[30] Where generated keys are stored long-term inside the crypto module as token objects the following storage keys may be accessed and used: GSK, USK (CO only), PSK (SO Only)

Service	Cryptographic Keys and CSPs Access	Role
Store Data Object	Non-cryptographic data	SO, Crypto Officer, Crypto User, Public User[31]
Read Data Object	Non-cryptographic data	SO, Crypto Officer, Crypto User, Public User[31]

[31] The Public User has access to Public Data Objects only.

Table 4-5 Keys and Critical Security Parameters

Keys and CSPs	CSP Type	Generation	Input / Output	Storage	Destruction	Use
Challenge Secret	16 character data string	AES-CTR DRBG	Output via direct connection to PED	Flash memory encrypted with PSK	N/A	Used in Trusted Path Authentication configuration only. 16 character random string generated by the cryptographic module and output via the PED display when the user is created. It is input by the operator as the authentication data for a client application login.
Random Challenge	One-time random number	AES-CTR DRBG	Output to host using ICD communication path	Working RAM in plaintext	Power Cycle	Used in Trusted Path Authentication configuration only. A one-time random number generated by the cryptographic module and sent to the calling application for each login. It is combined with the input Challenge Secret to compute the one-time response that is returned to the cryptographic module.
Challenge Response	20-byte value	N/A	Input from host using ICD communication path	Working RAM in plaintext	Power Cycle	A 20-byte value used for authentication in the challenge response scheme. It is generated using the challenge secret and the one-time random challenge value.
PED[32] Key Authentication Data	48-byte random value	AES-CTR DRBG	Input / Output via direct connection to PED	Flash memory encrypted	N/A	Used in Trusted Path Authentication configuration. A 48-byte random value that is generated by the module when the SO or User is created. It is written out to the serial memory device (PED key) via the Trusted Path.
Optional PIN	PIN	N/A	Input on the PED via secure channel. PED does not input or output the PIN.	Not stored On module	N/A	An optional PIN value used for authentication along with the PED key. It must be a minimum of 4-bytes long

[32] Within this document the terms “PED” key and “iKey” are interchangeable unless otherwise indicated.

Keys and CSPs	CSP Type	Generation	Input / Output	Storage	Destruction	Use
Cloning Domain Vector	48-Byte value	AES-CTR DRBG	Output via direct connection to PED	Flash Memory encrypted with PSK	N/A	48-byte value that is used to control a module's ability to participate in the cloning protocol. It is either generated by the module or imprinted onto the module at the time the module is initialized. The value is output from the original module in the domain onto a PED key to enable initializing additional modules into the same domain.
User Storage Key (USK)	AES-256	AES-CTR DRBG	Not Input or Output	Flash memory encrypted	N/A	This key is used to encrypt all sensitive attributes of all private objects owned by the user. Encrypted, as part of the partition data, by the key taken from the PED key data.
Partition Storage Key (PSK)	AES-256	AES-CTR-DRBG	Not Input or Output	Flash memory encrypted	N/A	The storage key for the SO/user partition. This key is used to encrypt the key data for the SO/user partitions. Encrypted as part of the SO/user partition data by the SO/user storage key (USK).
Global Storage Key (GSK)	AES-256	AES-CTR DRBG	Not Input or Output	Flash memory encrypted	N/A	32-byte AES key that is the same for all users on a specific SafeNet cryptographic module. It is used to encrypt permanent parameters within the non-volatile memory area reserved for use by the module. Encrypted, as part of the partition data, by the SO/User partition storage key (PSK).
Token or Module Unwrapping Key (TUK)	RSA-2048 bit private key	ANSI X9.31	Not Input or Output	Flash memory encrypted with GSK	N/A	A 2048-bit RSA private key used in the cloning protocol.
Token or Module Wrapping Certificate (TWC)	RSA-2048 public/private certificate	Loaded at Manufacturing	Public Key Output in Plaintext	Flash memory plaintext	N/A	Used in exchange of session encryption key as part of the handshake during the cloning protocol.
Token or Module Variable Key (TVK)	AES-256	AES-CTR DRBG	Not Input or Output	Tamperable BBRAM in plaintext	Zeroized as part of a tamper event	It is used to encrypt authentication data stored for auto-activation purposes. The non-volatile RAM is actively zeroized in response to a tamper event.
Master Tamper Key (MTK)	AES-256	AES-CTR DRBG	Not Input or Output	Tamperable BBRAM in plaintext	Zeroized as part of a tamper event	The MTK encrypts all sensitive values
Key Encryption Key (KEK)	AES-256	AES-CTR DRBG	Output Encrypted	Tamperable BBRAM in plaintext	Zeroized as part of a decommission signal	The KEK encrypts all sensitive values and is zeroized in response to a decommission signal.

Keys and CSPs	CSP Type	Generation	Input / Output	Storage	Destruction	Use
Masking Keys	AES-256 and HMAC SHA-256	AES-CTR DRBG	Not Input or Output	Flash memory encrypted with PSK	N/A	AES 256-bit and HMAC keys used during KTS masking operations. Stored encrypted using the PSK.
Manufacturer's Integrity Certificate (MIC)	RSA-4096 public key certificate	Loaded at manufacturing	Not Input or Output	Flash memory in plaintext	N/A	Used in verifying Hardware Origin Certificates (HOCs), which are generated in response to a customer function call to provide proof of hardware origin.
Manufacturer's Verification Key (MVK)	RSA-1024 public key	Loaded at manufacturing	Not Input or Output	Flash memory in plaintext	N/A	1024-bit public key counterpart to the Manufacturer's Signature Key (MSK) held at SafeNet. Used for key migration support for legacy HSMs.
Device Authentication Key (DAK)	RSA 2048 bit private key	ANSI X9.31	Not Input or Output	Flash memory encrypted with GSK	N/A	2048-bit RSA private key used for a specific PKI implementation requiring assurance that a key or a specific action originated within the hardware crypto module.
Hardware Origin Key (HOK)	RSA 4096 bit private key	ANSI X9.31	Not Input or Output	Flash memory encrypted with GSK	N/A	A 4096 bit RSA private key used to sign certificates for other device key pairs, such as the TWC. It is generated at the time the device is manufactured.
Hardware Origin Certificate (HOC)	RSA-4096 public key certificate	Loaded at manufacturing	Not Input or Output	Flash memory in plaintext	N/A	The X.509 public key certificate corresponding to the HOK. It is signed by the Manufacturer's Integrity Key (MIK) at the time the device is manufactured.
Remote PED Vector (RPV)	256-bit secret value	AES-CTR DRBG	Input / Output via direct connection to PED	Flash memory in plaintext	Zeroized via ICD command	A randomly generated 256-bit secret, which must be shared between a remote PED and a cryptographic module in order to establish a secure communication channel between them.
DRBG Key	AES-256	Hardware Random Source	Not Input or Output	Tamperable BBRAM in plaintext	Zeroized as part of a tamper event	32 bytes AES key stored in the BBRAM of the internal security co-processor. Used in the implementation of the NIST SP 800-90A CTR (AES) DRBG.
DRBG Seed	384 bits	Hardware Random Source	Not Input or Output	Tamperable BBRAM in plaintext	Zeroized as part of a tamper event	Random seed data drawn from the Hardware RBG in the security co-processor and used to seed the implementation of the NIST SP 800-90A CTR (AES) DRBG.

Keys and CSPs	CSP Type	Generation	Input / Output	Storage	Destruction	Use
DRBG V	128 bits	H/W Random Source	Not Input or Output	Tamperable BBRAM in plaintext	Zeroized as part of a tamper event	Part of the secret state of the approved DRBG. The value is stored in the security co-processor as plaintext and is generated using the methods described in NIST SP 800-90A.
DRBG Entropy Input	384 bits	H/W Random Source	Not Input or Output	Tamperable BBRAM in plaintext	Zeroized as part of a tamper event	The entropy value used to initialize the approved DRBG. The 48-byte value is stored ephemerally in memory of the security co-processor.
Secure Audit Logging Key (SALK)	256 bit HMAC	AES-CTR DRBG	Input / Output encrypted	Flash memory encrypted with SADK	N/A	A 256-bit key used to verify data integrity and authentication of the log messages. Saved in the parameter area of Flash memory.
Secure Audit Domain Keys (SADK)	AES-256	Derived (concatenation KDF from SP 800-56A) from Cloning Domain Vector	Not Input or Output	Working RAM in plaintext	Power Cycle	A 256-bit KTS key that is used to wrap/unwrap the SALK when it is exported / imported from / to the module.
Secure Audit Domain Authentication Key (SADAK)	256 bit HMAC	Derived (concatenation KDF from SP 800-56A) from Cloning Domain Vector	Not Input or Output	Working RAM in plaintext	Power Cycle	An HMAC KTS key that provides authentication for the wrap/unwrap of the SALK
RSA Asymmetric Key Pairs (general partition or session keys)	RSA	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use asymmetric key pairs that can be exported/imported from/to the module or generated by the module.
DSA Asymmetric Key Pairs (general partition or session keys)	DSA	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use asymmetric key pairs that can be exported/imported from/to the module or generated by the module.
ECC/ECDH Asymmetric Key Pairs (general partition or session keys)	ECC/ECDH	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use asymmetric key pairs that can be exported/imported from/to the module or generated by the module.
Diffie-Hellman Key Pairs (general partition or session keys)	DH	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use asymmetric key pairs that can be exported/imported from/to the module or generated by the module.

Keys and CSPs	CSP Type	Generation	Input / Output	Storage	Destruction	Use
AES Symmetric Keys (general partition or session keys)	AES	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use symmetric keys that can be exported/imported from/to the module or generated by the module.
Triple-DES Symmetric Keys (general partition or session keys)	Triple-DES	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use symmetric keys that can be exported/imported from/to the module or generated by the module.
MAC Keys (general partition or session keys)	MAC and HMAC	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use symmetric keys that can be exported/imported from/to the module or generated by the module.
KDF Symmetric Keys (general partition or session keys)	KDF	N/A (user imported) or AES-CTR DRBG (module generated)	Input (user imported) / Output encrypted	Flash Memory encrypted with USK (module keys) or RAM plaintext (session keys)	N/A (module keys) or zeroized when session ends (session keys)	General use symmetric keys that can be exported/imported from/to the module or generated by the module.
Firmware Signing Certificate (FSC)	RSA 4096 public key certificate	Loaded at manufacturing	Certificate Output in plaintext	Flash Memory in plaintext	N/A	The X.509 public key certificate corresponding to the Firmware Signing Key. Used in verifying firmware updates
Root Certificate	RSA 4096 public key certificate	Loaded at manufacturing	Certificate Output in plaintext	Flash Memory in plaintext	N/A	The X.509 public key certificate corresponding to the Root Key. It is self-signed. Used in verifying Manufacturing Integrity Certificate (MIC) and firmware updates