MFP Cryptographic Module(A)

FIPS 140-2 Non-Proprietary Security Policy

Document Revision:1.6

Document Date: Aug 2021

Prepared by:

atsec information security Corp.

9130 Jollyville Road, Suite 260

Austin, TX 78759

www.atsec.com

Kyocera Document Solutions Inc.
1-2-28 Tamatsukuri
Chuo-ku
Osaka, Osaka 540-8585
Japan
+81-6-6764-3355
https://www.kyoceradocumentsolutions.com/

For further information contact:
Masaki Sone masaki.sone@dc.kyocera.com
Akihiro Kanekawa akihiro.kanekawa@dc.kyocera.com

Table of Contents

1 Introduction

This document is the non-proprietary FIPS 140-2 Security Policy for the MFP Cryptographic Module(A) cryptographic module. Hereafter, “MFP Cryptographic Module(A)”, “the MFP module”, “the module”,” sub-chip module” are used interchangeably.

The document contains a specification of the rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-2 (Federal Information Processing Standards Publication 140-2) for a Security Level 2 hardware module.

1.1 Purpose of the Security Policy

There are three major reasons that a security policy is needed:

- it is required for FIPS 140-2 validation,
- it allows individuals and organizations to determine whether a cryptographic module, as implemented, satisfies the stated security policy, and
- it describes the capabilities, protection and access rights provided by the cryptographic module, allowing individuals and organizations to determine whether it will meet their security requirements.

1.2 Target Audience

This document is part of the package of documents that are submitted for FIPS 140-2 conformance validation of the module. It is intended for the following audience:

- Developers.
- FIPS 140-2 testing lab.
- The Cryptographic Module Validation Program (CMVP).
- Customers using or considering integration of the MFP Cryptographic Module(A) or its single-chip embodiment.

2 Cryptographic Module Specification

2.1 Module Overview

The MFP module is a cryptographic security module for encrypting data written to a storage device and other security functions of a Kyocera Multi-Function Printer (MFP). Secure key generation and fast AES encryption/decryption are offered through a SATA interface. SecureBoot verifies the integrity of the firmware when the product starts up.

This module is a single-chip hardware module implemented as a sub-chip in the Kyocera SCH114C SoC (system-on-chip). The module consists of two major components:

- VaultIP: a sub-block containing most of the cryptographic services
- EIP-38: a sub-block used to encrypt/decrypt data stored in the hard drive (or SSD) of the multi-functional printer.

The module offers key management and crypto functions needed for platform and application security and can be used stand-alone or as a 'Root of Trust' to support a TEE-based platform.

The module completely shields all key and security sensitive data from all CPUs, interfaces and memory. Security sensitive materials are stored as assets that never leave the module in unencrypted and/or non-authenticated form.

Additionally, the module offers hardware security features that are needed when operating in a Trusted Execution Environment (TEE). These features include One-Time-Programmable memory (OTP) access and management, Random Number Generation / entropy source, timers, (short) monotonic/non-volatile counters and import and export of keys and other assets.

2.2 Intended Usage

The primary application of the MFP module is to control and manage the operation of a Multi-function Printer where authentication, encrypted content processing using standard protocols, and protection of keys and other sensitive assets are required. The features available in the MFP are:

- Small footprint IP.
- Internal storage for protection and management of sensitive keys and assets.
- Root of Trust as true hardware interface to on chip One-Time Programmable (OTP) memory.
- Secure Timers (hardware counters).
- Hash engine to offload computationally intensive hash algorithms: SHA-1, SHA-2.
- Public Key Encryption, supporting RSA, ECDSA (sub-)functions.
- True random number generator (TRNG), also known as Non-deterministic random number generator (NDRNG).
- Embedded DMA controller for high speed symmetric crypto and hash data transfer.
- Encrypt/Decrypt services for external storage devices.

2.3 FIPS 140-2 Module Information

For the purpose of this Cryptographic Module Validation, the MFP Cryptographic Module(A) is synthesized in silicon as a single-chip hardware module validated at Security Level 2.

Table 1 below shows the security level claimed for each of the eleven sections that comprise the FIPS 140-2:

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

Table 1 - Security Levels

The MFP Cryptographic Module(A) has been tested as a single chip embodiment.

2.4 Approved Modes of Operation

The MFP module has two modes of operation: FIPS mode and non-FIPS mode.

- In FIPS mode of operation, only FIPS-Approved or FIPS-Allowed cryptographic algorithms with specific modes and key lengths can be requested. Table 12, Table 12a show all algorithms supported by the module in FIPS mode.
- In non-FIPS mode of operation, only the non-approved cryptographic algorithms listed in Table 13 are available.

The mode of operation is implicitly assumed depending on the service invoked. If the user is requesting a FIPS service, then the module will implicitly be in FIPS mode. If the user is requesting a non-FIPS service, then the module is implicitly in non-FIPS mode.

2.5 System Block Diagram

The figure below shows a system diagram in which MFP Cryptographic Module(A) is integrated in a SoC connected to a common bus system.

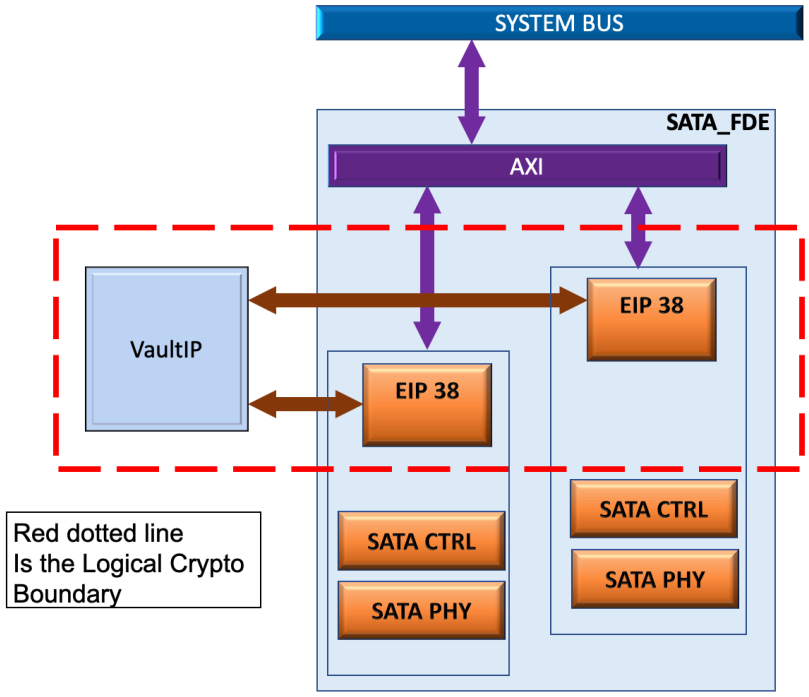

Figure 1 - System Block Diagram

For the purpose of this validation, the physical cryptographic boundary is enclosed in the single chip embodiment i.e. the Kyocera SCH114C SoC on which module is implemented. The logical cryptographic module boundary is represented within the red line.

2.6 Hardware Block Diagram

The MFP Cryptographic Module(A) consists of two major components: the VaultIP sub-component and EIP-38 sub-component. The VaultIP sub-component in turn consists of two sub-components: the Verilog RTL and the Firmware running from a ROM on the sequencer. The RTL implements the cryptographic algorithms and basic public key big number mathematics. The Firmware handles the higher level operations, manages the keys and takes care of the data transfers by setting up DMAs.

The EIP-38 sub-component consists of two AES-XTS engines and is implemented in Verilog RTL: one engine is used to encrypt data stored in the hard drive and/or SSD of the multi-function printer while the other engine is used to decrypt the same data. The breakdown of MFP Cryptographic Module(A) is shown in Figure 2; it shows the details of all interfaces that cross the security boundary and the first hierarchy levels of the MFP module RTL. Firmware is located in the Program ROM. The firmware has many routines; typically, the sources for each routine are located in an individual assembly file. All firmware routines are located on the same hierarchical level.

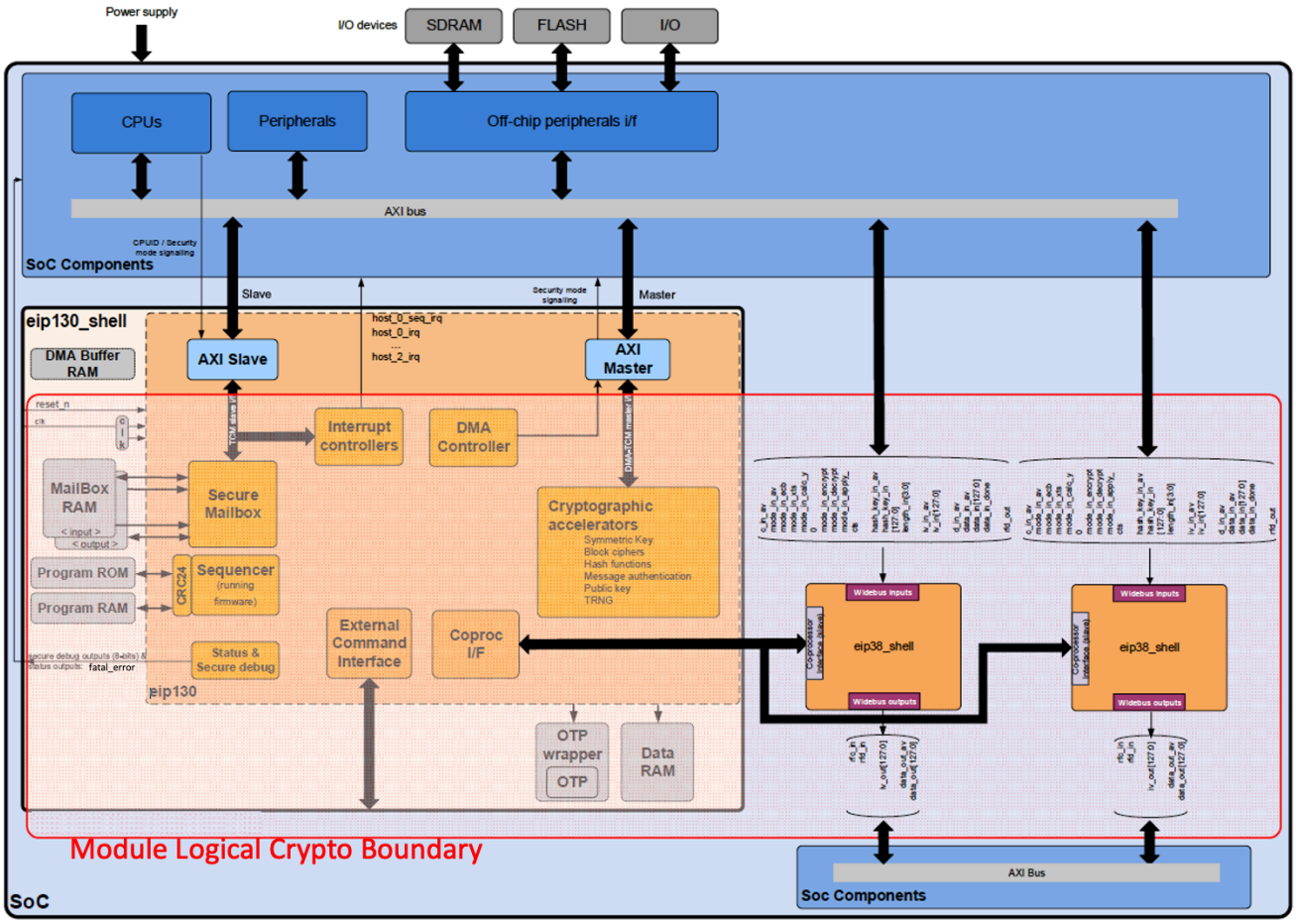

Figure 2 - Hardware Block Diagram

2.7 MFP module breakdown

The next list shows all (sub-)module levels of the MFP Cryptographic Module(A) and their corresponding version numbers. The levels provide an overview starting at the MFP Cryptographic Module(A) shell level.

Top-level MFP Cryptographic Module(A):

- EIP-38 sub-component using hardware version ‘EIP38-3.2’
  - Note: for security reasons, this block can only be controlled from the VaultIP using the Co-Proc I/F as shown in Figure 2
  - Two AES-XTS engines for encrypting/decrypting data. Keys and control are provided by the VaultIP sub-component.
- VaultIP TCM (Tightly Coupled Memory) module using hardware version ‘VaultIP-2.1.10’

Implementation of VaultIP TCM level. The TCM level embeds many sub-components. Several components are not implemented as dedicated sub-components from a design perspective, but do perform a specific operation and have their own hierarchy level:

- CRC32
- CRC24

- Counters
- Mailboxes
- OTP Interface
- Bus manager
- Internal DMA

The individual sub-components from a hardware design perspective are listed below.

- AES with ECB, CBC, CTR, CMAC, XTS (EIP-38), and CCM.

  AES data path (ECB)

- PKCP (public key co-processor interface). Used to communicate with the EIP-38 for operations on the multi-function printer's data storage hard drive (or SSD).
- HASH, SHA-1 and SHA-2, including SHA-224, SHA-256, SHA-384 and SHA-512.
- Multi-input 32-bit wide adders
- TRNG (generate and capture the entropy).
- Sequencer ('tiny' RISC processor), running the Firmware code.
- DMA controller, requesting data reads or writes to or from the module.
- Interrupt controller, captures the various interrupt sources and manages these to a single host interrupt. Multiple instantiations.
- The following are available, but not FIPS approved: AES-XTS (Vault), AES-GCM, Triple-DES with ECB and CBC.

Technology specific cells in the MFP Cryptographic Module(A) shell module.

- Memories
  - Mailbox RAM (2 instantiations, one input mailbox and one output mailbox)
  - Program ROM
  - OTP wrapper
  - Data RAM
- Clock gates
- FRO cells and related components / standard cells

Firmware VaultIP with version 2.2.18

- The MFP Cryptographic Module(A) Boot Firmware located in the Program ROM.

This includes code required to securely load & decrypt the main RAM firmware and verify its integrity. Also, this includes token management from and to the mailboxes (external interface), higher-level crypto operations, key generation, CTR-DRBG engine, asset management, DMA setup and generic engine control.

- PKA Firmware

Interfaces located outside the logical boundary, but inside the physical boundary of the MFP Cryptographic Module(A) .

- AXI Slave
- AXI Master: Interface module converts a DMA request into a data transaction from TCM to external AXI or reverse.

3 Ports and Interfaces

The MFP Cryptographic Module(A) module embeds a single slave and master interfaces. The slave interface is used to receive commands from one or more host CPUs and send the appropriate response. The master interface is used for autonomous data reads and writes from and to an external memory, flash or interface.

Additionally, the MFP module includes physical ports for showing the crypto module status, establishing the role that is requesting services, and resetting the crypto module.

3.1 Physical ports

The summary of interface pins located on the boundary of the MFP Cryptographic Module(A) is given in the tables below and shown in Figure 2. For clarity, signals are grouped by function in separate tables. Each port pin provides its name, direction, clock domain and function.

The first set of signals in the table below is hardware related and drives the various clock and reset signals.

Port Name	Direction	Clock Domain	Function
Clocks			
slv_clk	IN	slv_clk	Host interface clock.
ctr_clk	IN	ctr_clk	System counter clock signal. This clock signal may not be gated and must be connected to a fixed frequency, while the other clock speed could vary.
clk	IN	clk	Internal crypto-module clock.
Reset			
slv_reset_n	IN	slv_clk	Host interface reset.
ctr_reset_n	IN	ctr_clk	Counter reset. This reset signal may only be active (set to ‘0’) when the system is reset and must remain inactive after that, such that the counters remain counting.
reset_n	IN	clk	Module reset.
clk_man_reset_n	IN	n/a	This signal provides a means to reset all flip-flops inside the clock gate modules, e.g. for DFT and for the simulation processes to start in a known state. The clock gates are typically not connected to the global **reset_n** signal because it may be required to have the clocks running during a system reset.
External clock signals for dynamic clock control			
slv_clk_busy	OUT	clk	Indicates that the Host interface is busy with Host bus transfers. When 1b, indicates active transfer on Host bus.
ctr_clk_busy	OUT	ctr_clk	When 1b, indicates that the counter clock domain is active. This signal is always asserted (set to ‘1’), except when the counter module is in reset (**ctr_reset_n** set to ‘0’).
clk_busy	OUT	clk	When 1b, indicates that the module is active and busy with processing data and tokens.

Table 2 - Clock and Reset ports

The second group is related to software reset and is designed only for testing purposes: this

functionality is tied-off in the production cryptographic module and they are only provided for reference[1].

Port Name	Direction	Clock Domain	Function
soft_reset [tied-off to zero: 1'b0]	IN	slv_clk	Soft reset input. When this signal is made high, internal (state) registers is cleared and crypto engines are reset.
abort_req [tied-off to zero: 1'b0]	IN	slv_clk	Abort request signal. A high level of this signal indicates a request for a soft reset of the module.
abort_ack [unconnected]	OUT	slv_clk	Abort acknowledge signal. A high level of this signal indicates that the module is ready to receive a soft reset.

Table 3 - Soft reset ports

The third group provides signals to indicate the status of the MFP Module:

Port Name	Direction	Clock Domain	Function
fatal_error	OUT	clk	When active (set to '1'), the MFP Module detected an error. The errors can happen when power on self-test or conditional test fails or a self-test failure is detected when resuming from sleep. The value of this bit equals the value of the corresponding bit in the MODULE_STATUS register (bit 31).
debug_ctrl_out[7:0]	OUT	clk	Secure debug control output bus.

Table 4 - Status Signal ports

The fourth group provides signals to indicate the Power mode of the MFP Module:

Port Name	Direction	Clock Domain	Function
power_mode_out	OUT	clk	0b: MFP Module is operational. 1b: MFP Module is ready to enter 'Sleep Mode'. Only valid in combination with an active **power_mode_write** signal.
power_mode_write	OUT	clk	When active it indicates the status of **power_mode_out** is valid.
power_mode_in	IN	clk	The status of this signal is checked by the core during power up after Sleep Mode to see if the state information from Data RAM must be restored. 0b: MFP Module must not restore the state information from Data RAM after reset. 1b: MFP Module must restore the state information from Data RAM after reset.

Table 5 - Power Mode Control Signals

The following table provides the signals of the target Host interface. Besides global configuration, this physical interface provides the token interface to the mailboxes. Writing

[1] Can be completely reset by a hard reset. The firmware can be reset using a reset token, provided via the Control Input Interface.

to the mailbox triggers processing. After processing, the results can be read from the output mailbox using this same physical interface.

Port Name	Direction	Clock Domain	Function
cpu_id[2:0]	IN	slv_clk	Processor Host ID bits. These bits must be valid together with paddr.
psel	IN	slv_clk	AXI Select. Selects the TCM port for a bus transfer.
pwrite	IN	slv_clk	AXI Write. Indicates the direction of a bus transfer.
penable	IN	slv_clk	AXI enable. When HIGH, and asserted with psel, this means the AXI interface is being requested to complete a bus transfer. This signal indicates the second and subsequent cycles of an AXI transfer.
pprot[2:0]	IN	slv_clk	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access. The module uses the Host Secure/Non-Secure bits to accept or ignore accesses to specific asserts and operations.
paddr[13:0]	IN	slv_clk	AXI Address bus. Indicates the target address for the bus transfer (byte addressable).
pwdata[31:0]	IN	slv_clk	AXI Write Data bus. Transfers data from Master to Slave.
pready	OUT	slv_clk	AXI Ready Output. Indicates extension of the bus transfer, when de-asserted (‘0’).
prdata[31:0]	OUT	slv_clk	AXI Read Data bus. Transfers data from Slave to Master.
big_end_reg	IN	slv_clk	Big-endian. Indicates the byte order of the transfers. big_end_reg = 0b: Little-endian. big_end_reg = 1b: Big-endian. This input should not change while transfers are active.

Table 6 - AXI Slave interface (Host processor bus)

The next table provides the signals of the master data interface. This physical interface provides DMA signals and a target TCM interface. A request for data is initiated via the DMA interface. It is a requirement from the external system to provide the requested data on the TCM interface. Depending on the indicated direction of the DMA, input data is requested, or result data is available to be read.

Port Name	Direction	Clock Domain	Function
Read Command channel			
dmac_grant	IN	clk / mst_clk	Read address ready. When high, this signal indicates that the AXI slave or interconnect is ready to accept a read address and associated control signals. Make always set to HIGH.
dmac_ready	OUT	clk / mst_clk	DMA ready. When high, this signal indicates that the AXI slave or interconnect is ready to accept a address and associated control signals
dmac_rdata[31:0]	OUT	clk / mst_clk	Read data. The read data bus is 32 bits wide.
dmac_resp[1:0]	OUT	clk / mst_clk	Read response. This signal indicates the status of the read transfer.
dmac_bus_req	OUT	clk / mst_clk	Bus request signal. A high level of this signal indicates a request for BUS
dmac_lock	OUT	clk / mst_clk	This signal indicates to the arbiter that the requesting master is going to perform a number of indivisible transfers. It also indicates that the arbiter must not grant the bus to another bus master when the first of the locked transfers has commenced.

Port Name	Direction	Clock Domain	Function
dmac_write	OUT	clk / mst_clk	HIGH, this signal indicates a write transfer. When LOW, it indicates a read transfer.
dmac_addr[31:0]	OUT	clk / mst_clk	32-bit system address bus
dmac_burst[2:0]	OUT	clk / mst_clk	Indicates if the transfer is a burst transfer
dmac_prot[3:0]	OUT	clk / mst_clk	Protection control. Provides information about a bus access.
dmac_size[2:0]	OUT	clk / mst_clk	Indicates the size of the transfer. This is typically: • byte, 8-bit • halfword, 16-bit • word, 32-bit. The DMAC enables 8-bit, 16-bit, and 32-bit transfer widths: 8-bit: HSIZE[2:0] = 0b000 16-bit: HSIZE[2:0] = 0b001 32-bit: HSIZE[2:0] = 0b010
dmac_trans[1:0]	OUT	clk / mst_clk	Indicates the type of the current transfer. This can be: • NONSEQUENTIAL • SEQUENTIAL • IDLE • BUSY.
dmac_wdata[31:0]	OUT	clk / mst_clk	The write data bus transfers data from the master to the bus slaves during write operations

Table 7 - Master DMA/TCM interface ports (AXI)

The next table provides an overview of the interrupt outputs. By default, the enabled number of interrupts equals the number of mailboxes. If the slave interface supports secure accesses, host 0 has a dedicated IRQ for that.

When accessing the module securely (prot_acc_n = 0b), it can also configure the mailboxes and interrupts. During integration, other optional interrupts can be enabled in the module, dependent on the system host CPUs and co-access requirements to the MFP Module.

Port Name	Direction	Clock Domain	Function
host_1_sec_irq	OUT	slv_clk	Combined interrupt output (active HIGH) for one or more Hosts and Domain. Represents Host1, secure interrupt. The interrupt controller is only accessed when Host ID equals 0 and PROT is zero (secure). This interrupt is only available when the slave interface has a PROT signal.
Host_0_irq	OUT	slv_clck	Combined interrupt output (active HIGH) for one or more Hosts and Domain. Represents Host0, non-secure interrupt. The interrupt controller is only accessed when Host ID equals 0 and PROT is one (non-secure).
host_1_irq	OUT	slv_clk	Combined interrupt output (active HIGH) for one or more Hosts and Domain. Represents Host1, non-secure interrupt. The interrupt controller is only accessed when Host ID equals 1 and PROT is one (non-secure).

Table 8 - Interrupt signal ports

Another set of signals available on the cryptographic boundary is the SCAN and FRO characterization signals. Only the *scan_mode_en* and *tst_fro_iddq* signals are connected for device production test purposes. Other direct FRO characterization signals are tied-off (for input signal direction), or left unconnected (for output signal direction)[2].

Port Name	Direction	Clock Domain	Function
Characterization / FRO Characterization			
scan_mode_en	IN	None	Active HIGH enable signal for scan_test_mode. This signal typically comes from a testmode controller and is used to break unwanted combinatorial loops during scan test.
tst_fro_iddq	IN	None	Active HIGH enable signal for IDDq testing – this forces all fro_enable outputs LOW to shut down all FROs, overruling all other control signals. This is a combinatorial function, TRNG component clocks don't need to run for this to work.
tst_fro_ctrl_en <= 1'b0	IN	None	Active HIGH enable signal for FRO characterization (enables the tst_fro_select, tst_fro_enable and tst_fro_delay inputs). This is a combinatorial function, TRNG component clocks don't need to run for this to work.
tst_fro_select[4:0] <= 5'b00000	IN	None	FRO selection input (valid values 0-7). A selected FRO will have its fro_testin input forced LOW.
tst_fro_enable <= 1'b0	IN	None	Active HIGH enable signal for FRO selected by tst_fro_select.
tst_fro_delay <= 1'b0	IN	None	Delay chain length selection for FRO selected by tst_fro_select. This input should only be changed while tst_fro_enable is LOW.
tst_fro_clk_out [unconnected]	OUT	None	Output clock signal on the FRO shell module from the FRO selected by tst_fro_select, forced 'low' when tst_fro_ctrl_en is 'low'. This output can also be activated for register-controlled characterization.

Table 9 - TRNG control ports

3.2 Logical Interfaces

The Slave Interface ports communicate with the host through the AXI slave interface, providing the token interface to the mailboxes. Writing to the mailbox triggers processing. Once the input token is processed completely, the results can be read from the output mailbox using this same interface. The Firmware running on the embedded sequencer reads the input tokens, starts processing and writes the output token triggering the corresponding interrupt. Based on the interrupt, an external host can read the result output token. Output is not available in the output mailbox until the input token is fully processed.

Input tokens sent through the Slave Interface constitutes data input and control input interfaces; output tokens constitute data output and status output interfaces.

[2] The Module provides a token that can be used to return sampled TRNG outputs. The requested number of bits is returned over the RxT data output interface using DMA.

The Master DMA/TCM interface communicates with the host through the AXI master interface, providing DMA signals and a target TCM interface. A request for data is initiated via the DMA interface. It is a requirement from the external system to provide the requested data on the TCM interface. Depending on the direction of the DMA, input data is requested, or result data is available to be read. The Master DMA/TCM interface provides support for data input and data output interfaces.

The MFP Cryptographic Module(A) also includes additional ports for Control Input and Status Output:

- The Clock and Reset ports provide Control Input and Status Output interfaces.
- The Soft Reset ports (testing only) provide Control Input and Status Output interfaces.
- The Status Signal ports provide the Status Output interface.
- Interrupt signal ports provide the Status Output interface.
- Coprocessor interface port. This port is used to export key material to the EIP-38 AES-XTS sub-component.
- The TRNG control ports (testing only) provide Control Input and Status Output.
- The power port provides power input to the MFP Cryptographic Module(A).

The following table shows the mapping between the ports available in the MFP Cryptographic Module(A) and the logical interfaces:

Port Groups	Data Input	Data Output	Control Input	Status Output	Power Input
Clock and Reset ports			✓	✓	
Soft Reset ports			✓	✓	
Status Signal ports				✓	
Slave interface ports (AXI Slave Interface)	✓	✓	✓	✓	
Master DMA/TCM interface ports (AXI Master Interface)	✓	✓			
Interrupt signal ports				✓	
TRNG control ports			✓	✓	
Power port					✓

Table 10 - Logical Interfaces

4 Roles, Services and Authentication

4.1 Roles

The MFP Cryptographic Module(A) module is usually installed as part of a System on Chip (SoC), where applications running on the system can use the MFP Cryptographic Module(A) cryptographic services. Applications must identify and authenticate to the MFP Cryptographic Module(A) through one of the following roles:

- **User Role**: This role performs general services, cryptographic operations, and asset managements.
- **Crypto Officer Role**: This role can perform the same functionality as the user role, but it also performs initialization services in the cryptographic module (e.g. configuration of the TRNG engine, write operations in OTP and registers).

The MFP Cryptographic Module(A) module implements a role-based authentication method (See section 4.3). Internal mechanisms of the module ensure that User and Crypto Officer service requests and assets (key material and other CSPs) are properly separated and protected. The next section shows the services provided by the MFP Module and the roles that can request them. All services require an authorized role.

4.2 Services

The following table presents the services provided by the MFP Cryptographic Module(A), including:

- The input token through which the service is requested.
- The authorized roles: a checkmark in the role column indicates that the user authenticated with that role can perform the service.
- The cryptographic algorithms involved in the service. The algorithm can be split in several roles for different modes or key lengths as they may not be available in FIPS mode of operation.
- Whether the service is available in FIPS mode
- The Keys and Critical Security Parameters (CSPs) involved in the service.
- How the service accesses the Keys and CSPs: (C)reate (create and fill the CSP), (R)ead (read an existing CSP), (U)pdate (write an existing CSP), (D)elete (delete and zeroize memory where the CSP was stored). An asterisk (*) indicates that the CSP is transported via the token or the Host memory DMA (assets cannot be used).

Service	Input Token	Roles		Cryptographic Algorithms	FIPS	Keys and CSPs	Access
		User	CO				
EIP-38 sub-component AES-XTS encryption and decryption	Asset Export via the Coprocessor Interface		✓	n/a	✓	n/a	R*
Encryption and Decryption	Encryption	✓	✓	AES (ECB, CBC, CTR)	✓	AES key	R
		✓	✓	AES (CCM)	✓	AES key	R
		✓	✓	AES (XTS) in VaultIP component		AES key	R

Service	Input Token	Roles		Cryptographic Algorithms	FIPS	Keys and CSPs	Access
		User	CO				
		✓	✓	AES (GCM)		AES key	R
		✓	✓	Triple-DES (ECB, CBC)		Triple-DES key	R
Message Digest	Hash	✓	✓	SHA-1	✓	n/a	
		✓	✓	SHA-224, SHA-256	✓	n/a	
		✓	✓	SHA-384, SHA-512	✓	n/a	
MAC Generation	MAC	✓	✓	HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	✓	HMAC key	R
		✓	✓	AES-CMAC	✓	AES key	R
MAC Verification	MAC	✓	✓	HMAC-SHA-1	✓	HMAC key	R
		✓	✓	HMAC-SHA-224, HMAC-SHA-256	✓	HMAC key	R
		✓	✓	HMAC-SHA-384, HMAC-SHA-512	✓	HMAC key	R
		✓	✓	AES-CMAC	✓	AES key	R
		✓	✓	AES-CBC-MAC		AES key	R
ECDH/ECDSA key verification	Public Key	✓	✓	ECDH	✓	EC parameters ECDH private key *(optional)* ECDH public key *(optional)*	R R R
		✓	✓	ECDSA	✓	EC parameters ECDSA private key *(optional)* ECDSA public key *(optional)*	R R R
ECDSA Sig Gen	Public Key	✓	✓	ECDSA (P-224, P-256, P-384, P-521), SHA-224, SHA-256, SHA-384, SHA-512	✓	EC parameters ECDSA private key	R R
ECDSA Sig Verify	Public Key	✓	✓	ECDSA (P-192, P-224, P-256, P-384, P-521), SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	✓	EC parameters ECDSA public key	R R
RSA Sig Gen	Public Key	✓	✓	RSA-PSS and RSA-PKCS#1-v1.5 (n=2048, 3072), SHA-224, SHA-256, SHA-384, SHA-512	✓	RSA-PSS private key RSA-v1.5 private key	R

Service	Input Token	Roles		Cryptographic Algorithms	FIPS	Keys and CSPs	Access
		User	CO				
RSA Sig Verify	Public Key	✓	✓	RSA-PSS and RSA-PKCS#1-v1.5 (n=1024 to 3072), SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	✓	RSA-PSS public key RSA-v1.5 public key	R
ECDH/ECDSA generate public key	Public Key	✓	✓	ECDSA (P-224, P-256, P-384, P-521)	✓	ECDH/ECDSA private key EC parameters ECDH/ECDSA public key	R R C
ECDH/ECDSA generate private and public key	Public Key	✓	✓	ECDSA (P-224, P-256, P-384, P-521)	✓	ECDH/ECDSA private key EC parameters ECDH/ECDSA public key	C R C
ECDH generate shared secrets single key-pair	Public Key	✓	✓	ECDH (P-224, P-256, P-384, P-521)	✓	ECDH private key EC parameters ECDH public key ECDH shared secret	R R R C
ECDH key agreement (single key-pair)	Public Key	✓	✓	ECDH (P-224, P-256, P-384, P-521), CTR_DRBG, one-step KDF [SP800-56C Rev1], section 4.1 (SHA-256)	✓	ECDH private key EC parameters ECDH public key Derived Key	R R R C
ECDH generate shared secrets dual key-pair	Public Key	✓	✓	ECDH (P-224, P-256, P-384, P-521), CTR_DRBG	✓	ECDH private key (2x) EC parameters ECDH public key (2x) ECDH shared secret	R R R C
ECDH key agreement (dual key-pair)	Public Key	✓	✓	ECDH (P-224, P-256, P-384, P-521), CTR_DRBG, one-step KDF [SP800-56C Rev1], section 4.1 (SHA-256)	✓	ECDH private key (2x) EC parameters ECDH public key (2x) Derived Key	R R R C
AES Key wrapping	AES (Un)Wrap	✓	✓	AES Key Wrap with or without padding as specified in [SP800-38F]	✓	AES key	R/R*
AES Key unwrapping	AES (Un)Wrap	✓	✓	AES Key Wrap with or without padding as specified in [SP800-38F]	✓	AES key	R/R*
TRNG Configuration	TRNG Configuration		✓		✓	n/a	
Get TRNG Random Number	TRNG Get Random	✓	✓	TRNG	✓	Raw random data to seed DRBG	C*

Service	Input Token	Roles		Cryptographic Algorithms	FIPS	Keys and CSPs	Access
		User	CO				
	Number						
Get DRBG Random Number	TRNG Get Random Number	✓	✓	CTR_DRBG	✓	Entropy Input Internal DRBG state (seed, V and K)	R RU
TRNG Post-processing Verification	TRNG Post-Processing Verification	✓	✓		✓	n/a	
TRNG Hardware Self-test	TRNG Hardware Self-test	✓	✓		✓	n/a	
Dynamic Asset Creation	Asset Create	✓	✓		✓	Asset	C
Static Asset Search	Static Asset Search	✓	✓		✓	n/a	
Dynamic Asset Key Derivation	Asset Load (derive)	✓	✓	SP800-108 KDF SP800-56C Rev1 KDF	✓	Asset Key Derivation Key (KDK)	U R
Dynamic Asset Import as keyblob	Asset Load (import)	✓	✓	AES-CMAC, AES-CTR	✓	Asset Key Encryption Key (KEK)	U R
Dynamic Asset Import as AES wrapped keyblob	Asset Load (AES Unwrap)	✓	✓	AES-KW, AES-KWP	✓	Asset AES Wrap Key	U R
Dynamic Asset Generation	Asset Load (random)	✓	✓	CTR_DRBG, AES-CMAC, AES-CTR	✓	Asset Key Encryption Key (KEK)[3]	U R
Dynamic Asset Import as plaintext	Asset Load (plaintext)	✓	✓	AES-CMAC, AES-CTR	✓	Asset Key Encryption Key(KEK)[3]	U R
Dynamic Asset Deletion	Asset Delete	✓	✓		✓	Asset	D
Public Data Read	Public Data Read	✓	✓		✓	Asset	R*
Monotonic Counter Read	Monotonic Counter Read	✓	✓		✓	Asset	R*
Monotonic Counter Increment	Monotonic Counter Increment	✓	✓		✓	Asset	RU
OTP Data Write	One-Time-Programmable Data Write		✓		✓	Asset Provisioning Key	CU R

[3] The Key Encryption Key (KEK) is required when the asset needs to be exported as a keyblob.

Service	Input Token	Roles		Cryptographic Algorithms	FIPS	Keys and CSPs	Access
		User	CO				
Generate Random HUK	Provision Random HUK		✓	DRBG	✓	Trusted Root Key (HUK) Crypto Officer Identity	C C
Show Status	System Information	✓	✓		✓	n/a	
On Demand[4] Self-Tests	N/A	✓	✓	All relevant FIPS algorithms	✓	Asset Store	D
Module Reset	Reset	✓	✓		✓	Asset Store	D
Authenticated Unlock Start	Authenticated Unlock Start	✓	✓	DRBG	✓	RSA Authentication Key Authentication State	R U
Authenticated Unlock Verify	Authenticated Unlock Verify	✓	✓	ECDSA P-256	✓	RSA Authentication Key Authentication State	R RU
Activate / Deactivate Debug port signals	Set Secure Debug	✓	✓		✓	RSA Authentication Key Authentication State	R R
Register Read	Register Read	✓	✓		✓	n/a	
Register Write	Register Write		✓		✓	n/a	
Zeroize Output Mailbox	Zeroize Output Mailbox	✓	✓		✓	CSPs in output mailbox	D
Create / Update User Identity	Define Users		✓		✓	User Identity	CU
Select OTP Zeroize	Select One-Time-Programmable Zeroize		✓		✓	n/a	
Zeroize OTP	Zeroize One-Time-Programmable		✓		✓	CSPs in the OTP and Asset Store	D
Go to Sleep mode	Sleep Mode	✓	✓		✓	n/a	
Resume from Sleep mode	Resume from Sleep	✓	✓		✓	n/a	
Set the System timer	Set System Time		✓		✓	n/a	
Module Initialization	Self-test		✓		✓	n/a	

Table 11 - MFP Cryptographic Module(A) Services

[4] This service is performed by power-off and power-on of the module.

The following table shows the FIPS-Approved. The FIPS-approved algorithms are validated under the CAVP with certificate numbers C1892 and C1933 as noted in the table.

Note: Not all of the algorithms/modes verified through the CAVP certificates are available from the module.

Algorithm	Usage	Key lengths	Modes	Standards
EIP-38 component obtained the CAVP C1933 certificate for the below algorithm capabilities				
AES (EIP-38)	Encryption Decryption	128, 256 bits	XTS	[SP800-38E]
VaultIP component obtained the CAVP C1892 certificate for the below algorithm capabilities				
AES (VaultIP)	Encryption Decryption	128, 192, 256 bits	ECB, CBC, CTR	[FIPS197] [SP800-38A]
	Encryption Decryption	128, 192, 256 bits	CCM	[SP800-38C]
	MAC	128, 192, 256 bits	CMAC	[SP800-38B] [SP800-38D]
AES Key Wrapping (IG D.9)	Key Wrapping (KTS) provides between 128 and 256 bits of strength	128, 192, 256 bits	KW, KWP	[SP800-38F]
AES Key Wrapping with AES-CMAC and AES-CTR	Key Wrapping (KTS) provides 256 bits of strength	256-bit AES keys	AES-CMAC and AES-CTR	[SP800-38B] [SP800-38A] IG D.9
DRBG	Random Number Generation		AES-256 in CTR mode, no derivation function, prediction resistance disabled	[SP800-90A] [SP800-38A]
ECDSA	Key Verification Signature Verification	P-192, P-224, P-256, P-384, P-521		[FIPS186-4]
	Integrity test of Firmware RAM (Signature Verification)	P-256		[FIPS186-4]
	Key Generation Signature Generation	P-224, P-256, P-384, P-521		[FIPS186-4]
KAS ECC CDH Component (CVL) (approved per IG D.8 scenario 5)	Shared Secret Computation used in Key Agreement Scheme	P-224, P-256, P-384, P-521		[SP800-56Ar1] Section 5.7.1.2
HMAC-SHA-1	MAC Verification	112-512 bits		[FIPS198-1]
HMAC-SHA-224 HMAC-SHA-256 HMAC-SHA-384 HMAC-SHA-512	MAC Generation, MAC Verification	112-512 bits 128-512 bits 128-512 bits 128-512 bits		[FIPS198-1]

Algorithm	Usage	Key lengths	Modes	Standards
KBKDF	Key Derivation	128-512 bits	Counter and feedback modes using CMAC-AES-256 and HMAC-SHA-256	[SP800-108] [FIPS198-1] [SP800-38B]
KDA (vendor affirmed, no Cert.)	Key Derivation	128-512 bits	Counter and feedback modes using CMAC-AES-256 and HMAC-SHA-256	[SP800-56C Rev1] [SP800-108] [FIPS198-1] [SP800-38B]
Key Generation (vendor affirmed, no Cert.)	Cryptographic Key Generation (CKG)	128, 192, 256 bit AES key 112-512 bit HMAC key		[SP800-133]
RSA	Signature Verification	n=(1024 to 3072)	RSA-PSS (no CRT)	[PKCS#1] [FIPS186-4]
	Signature Generation	n=(2048 and 3072)		
	Signature Verification	n=(1024 to 3072)	RSA-PKCS#1v1.5 (no CRT)	
	Signature Generation	n=(2048 and 3072)		
SHA-1	Message Digest	N/A	N/A	[FIPS180-4]
SHA-224, SHA-256, SHA-384, SHA-512	Message Digest	N/A	N/A	[FIPS180-4]

Table 12 - FIPS approved cryptographic algorithms

The following table shows the FIPS-Allowed algorithms.

Algorithm	Usage	Key lengths	Modes	Standards
ECDH (allowed per IG D.8 scenario 3) Cert. #C1892 CVL	Key Agreement	P-224, P-256, P-384, P-521	one-step KDF using SHA-256	[SP800-56A] section 5.7.1.2 [SP800-56C Rev1] section 4
NDRNG (non-approved but allowed per IG 7.15, no Cert.)	Non-Deterministic Random Number Generator	N/A	N/A	

Table 12a - FIPS allowed algorithms

The following table shows the cryptographic algorithms and their key lengths that are not FIPS-Approved.

Algorithm	Usage	Key lengths	Modes
VaultIP Component algorithms			
AES	Encryption / Decryption	128, 192, 256 bits	GCM
AES	Encryption / Decryption	128, 256 bits	XTS (#C1892)

Triple-DES	Encryption / Decryption	192 bits	ECB, CBC

Table 13 - Non-Approved FIPS algorithms

The MFP Cryptographic Module(A) also supports the following algorithms that are used for firmware and data integrity:

Algorithm	Usage
CRC-24	ROM Firmware Integrity Test
CRC-32	Asset Integrity Test

Table 14 - Other Algorithms

When a CSP is loaded or updated in the asset store, the MFP Cryptographic Module(A) generates a CRC-32 checksum. Whenever the CSP is used by a service (referenced through its asset ID), the MFP Cryptographic Module(A) verifies the integrity of the CSP comparing the existing and re-calculated checksums.

Note: Integrity of the RAM Firmware is performed using ECDSA signature verification at FW load time. When resuming from Sleep mode, RAM Firmware integrity is verified by SHA-256.

4.3 Identification and Authentication

Role-based identification is indicated through the use of a single bit hardware signal (*prot_acc_n*) which is provided as side band information for all services. When the input token is written (the token data is available on the data input bus) the sideband signal must be valid.

Role-based authentication is performed through the use of a 32-bit identity value provided in the input token for all services. The MFP Cryptographic Module(A) requires the identification and authentication of the user and Crypto Officer roles for each service request; role identification and authentication status is not internally maintained in the cryptographic module.

The module is initialized via the “Provision Random HUK” token, which instructs the module to generate the Trusted Root key (HUK) and assigns the Crypto Officer 32-bit ID. The module supports only one Crypto Officer role identity; this value is stored in One Time Programmable (OTP) memory and cannot be altered unless the whole module is zeroized.

To access the module to request services, the Crypto Officer role is explicitly selected using the role selection bit (*prot_acc_n=0b*). For each individual token received with the Crypto Officer role selected, the MFP Cryptographic Module(A) compares the *32-bit ID field* provided in the input token with a predefined 32-bit value stored inside the OTP of the MFP Cryptographic Module(A) (this identity value is pre-defined by vendor and loaded to OTP during the module initialization). If the comparison succeeds, then the input token is processed. Otherwise, the service request is rejected indicating the error in the output token.

Likewise, the User role is explicitly selected using the role selection bit (*prot_acc_n=1b*). For each individual token received with the user role selected, the MFP Cryptographic Module(A) compares the *32-bit ID field* provided in the input token with the stored user IDs (up to four) inside the MFP Cryptographic Module(A). If the identity matches one of the stored user IDs, access is granted and the input token is processed. Otherwise, the request is rejected indicating the error in the output token. User role identities are created by the Crypto Officer

role and reside in volatile memory.

If the authentication fails, the MFP Cryptographic Module(A) waits at least 15ms (minimum value set for the highest chip frequency) before it returns the output token rejecting the service request.

For the User role, the *Define Users* token allows the Crypto Officer to create or modify up to four identities, whose values are stored internally in MFP Cryptographic Module(A)'s internal memory. These identities do not survive a power-cycle; the Crypto Officer must define the users again anytime the MFP Cryptographic Module(A) is reset or powered-up and invoke the "Self-Test" token to initialize user authentication.

4.4 Mechanism and Strength of Authentication

The probability of successfully guessing the Crypto Officer Identity is: $P_{co} = \frac{1}{2^{32}}$,

and the probability of successfully guessing one of the four User Identities is: $P_u = \left(4 \times \frac{1}{2^{32}}\right)$.

In both cases, the FIPS 140-2 requirements are satisfied that the probability is less than $1/1{,}000{,}000$ that a random attempt will succeed or a false acceptance occurring since

$$P_{co} = \frac{1}{2^{32}} < \frac{1}{1{,}000{,}000} \quad \text{and} \quad P_u = \frac{4}{2^{32}} < \frac{1}{1{,}000{,}000}$$

In addition, since there is a $15ms$ delay for a failed authentication, the MFP Cryptographic Module(A) can process at most 4000 consecutive failed authentication attempts every minute since

$4000\ attempts = {}^{60s}/_{0.015s}$

This means, in the case of the Crypto Officer identity, the overall success rate is equal to

$4000 \times P_{co} = \frac{4000}{2^{32}} \approx \frac{1}{1{,}073{,}742}$ which is $strictly\ less\ than\ \frac{1}{100{,}000}$.

Likewise in the case of the User identity, the overall success rate is

$4000 \times P_u = \frac{16000}{2^{32}} \approx \frac{1}{268{,}435}$ which is $strictly\ less\ than\ \frac{1}{100{,}000}$.

In both scenarios, the FIPS 140-2 requirement is meet that the probability of multiple attempts within a minute is less than $1/100{,}000$ that a false attempt will succeed or a false acceptance will occur.

4.5 Authentication Data protection

The Crypto Officer Identity is stored in the OTP and cannot be modified. The four identities assigned to users with the User role reside in internal memory and can be only created or updated by the Crypto Officer, who needs to authenticate with the correct Identity.

The user role identities are zeroized when the MFP Cryptographic Module(A) is powered-off; the Crypto Officer Identity remains in the OTP but can be zeroized through OTP zeroization.

No authentication data can be output by any of the available services.

5 Physical Security

For the purpose of this validation, the MFP Cryptographic Module(A) was synthesized in silicon as part of the Kyocera SCH114C SoC. This single chip presents a metallic integrated heat spreader (IHS) production-grade opaque cup lid. This lid serves as a protective shell around the silicon die and a pathway for heat exchange between the SoC and SoC cooler. The IHS lid provides opacity in the visible spectrum and prevents any tamper-evident access to the interior of the single-chip. The single-chip conforms to Level 2 requirements for physical security.

6 Operational Environment

The module operates in a non-modifiable environment.

7 Cryptographic Key and CSP Management

The Asset Store provides the core mechanism for secure handling of key material and other sensitive data like the IV, digest, MAC and state. The Asset Store is designed to store asset objects and allows them to be used, while never revealing their contents outside of the MFP Cryptographic Module(A). The Asset Store identifies the following three types of assets:

- *Static Asset*: this type of asset is key material that is available immediately after power-up and is located in the OTP. This asset cannot be modified or output (nevertheless, the whole OTP can be zeroized by the Crypto-Officer, see section 7.6).
- *Dynamic Asset*: this type of asset can be key material or any other sensitive data like the IV, digest, MAC and state. This asset needs to be loaded into the Asset Store before its use because it is located in the internal Data RAM memory of the MFP Cryptographic Module(A)and set to zero at powered-off. This asset cannot be modified, only deleted from the asset store.
- *Public data object*: this type of asset is data that can be retrieved from the Asset Store in plaintext for use outside the MFP Cryptographic Module(A) and can reside either in the OTP or the internal memory. This asset cannot be modified, only deleted from the asset store.

Static assets are used in the same way as dynamic assets, the only difference is their lifespan. Both are stored in plaintext within the MFP Cryptographic Module(A) and protected from disclosure and modification.

Whenever an asset is loaded or updated, the MFP Cryptographic Module(A) generates a CRC-32 checksum for that asset; a read of an asset will verify its integrity re-computing the CRC-32 checksum and comparing it to the stored checksum.

The following table summarizes the cryptographic keys used in the MFP Cryptographic Module(A) with the key lengths supported, the available methods for key generation, entry and output and the way they are stored. Notice that for Key Entry:

- "Firmware" means that the keys are determined during the delivery process and are unique to a given customer. See section 7.3.2 for more information.
- "KeyBlob" is a block of binary data encrypted through a key wrapping method using the AES-CMAC and AES-CTR algorithms. See section 7.3.1 for more information.

Name	Key Length / CSP Size	Generation			Entry					Storage		Output	
		KBKDF	Random	Key Pair	Firmware	OTP	Plaintext	AES-KeyWrap	KeyBlob	Static	Dynamic	KeyBlob	Plaintext
Trusted Root Key (HUK) (HMAC-SHA-256)	128, 256 bits		✓			✓			✓	✓		✓	
Trusted Key Encryption Keys (KEK) (AES-CMAC or AES-CTR)	256 bits	✓								✓	✓		
Trusted Key Derivation Keys (KDK) (HMAC-SHA-256 or AES-CMAC)	128, 256 bits	✓								✓	✓		
Authentication Keys (RSA public key)	2048-3072 bits				✓					✓			

Provisioning Key (AES-CMAC, AES-CTR)	2x256 bits				✓					✓			
AES keys	128, 192, 256 bits	✓	✓				✓	✓	✓		✓	✓	
HMAC keys	SHA-1: 112-512 bits SHA-224: 112-512 bits SHA-256: 128-512 bits SHA-384: 1024 bits SHA-512: 1024 bits	✓	✓				✓	✓	✓		✓	✓	
RSA key pairs	2048-3072 bits						✓	✓	✓		✓	✓[5]	
ECDSA key pairs	224-521 bits			✓			✓	✓	✓		✓	✓[6]	✓[6]
EC Diffie-Hellman key pairs	224-521 bits			✓			✓	✓	✓		✓	✓[6]	✓[6]
Crypto Officer Role	32 bits						✓			✓			
User Identity	32 bits						✓				✓		
Entropy input string											✓		
DRBG internal state (seed, V and K)											✓		

Table 15 - Life Cycle of Keys and Critical Security Parameters (CSPs)

The Provisioning Key is a key that is available for OTP initialization only. When the complete OTP is initialized the Provisioning Key cannot be used anymore.

7.1 Key Generation

The MFP Cryptographic Module(A) provides services for generating asymmetric and symmetric keys. The key generation methods implemented in the module are compliant with [SP800-133] (vendor affirmed).

The MFP Cryptographic Module(A) implements symmetric key generation for AES and HMAC keys directly using random data obtained from a Deterministic Random Bit Generator (DRBG) compliant with [SP800-90A].

The MFP Cryptographic Module(A) implements Elliptic Curve DSA (ECDSA) asymmetric key generation services compliant with [FIPS186-4]. A seed (i.e. the random value) used in asymmetric key generation is directly obtained from the [SP800-90A] DRBG.

Elliptic Curve Diffie-Hellman (ECDH) key pairs are generated by ECDSA.

Intermediate key generation values are not output from the cryptographic module during or after processing the service.

7.1.1 Key Derivation

The MFP Cryptographic Module(A) provides services for deriving keys from the Trusted Root Key (HUK), or any asset indicated as a Trusted Key Derivation Key (KDK). The MFP Cryptographic Module(A) provides key-based derivation in compliance with [SP800-108] and [SP800-56C Rev1] (vendor affirmed).

[5] The RSA key pair only when loaded as plaintext can be output as Keyblob.
[6] The EC Private Key (loaded as plaintext or generated with key EC key generation service in table 11) can be output as Keyblob whereas the EC Public Key (loaded as plaintext or generated with key EC key generation services in table 11) can be output either as a Keyblob when loaded as plaintext or as plaintext when generated.

7.2 Key Entry and Output

Electronic key entry method is used to import the private/secret keys; there is no manual key import or export method used in the MFP Cryptographic Module(A) .

7.2.1 Dynamic assets

Dynamic assets can be input into the MFP Cryptographic Module(A) through the following methods:

- Load plaintext from the *Host* (*Asset Load Plaintext* token).
- AES Key Wrapping as specified in NIST SP 800-38F (*Asset Load AES-Wrap* token).
- AES Key Wrapping with AES-CMAC, AES-CTR *(Asset Load Import token).*

Keys can also be initialized in the asset store using the built-in key generation features provided by the MFP Cryptographic Module(A) (see section 7.1).

When plaintext or random number loaded assets must survive a power cycle, they must be stored outside the MFP Cryptographic Module(A). The Asset Store is able to generate a block of binary data known as a *Key Blob* in which the asset is exported to the *Host*. MFP Cryptographic Module(A) utilizes the AES-CMAC, AES-CTR algorithms to protect the asset inside the *Key Blob* from disclosure and modification, using a Trusted Key Encryption Key (KEK). An important part of the protection provided by the *Asset Store* is to allow a *Key Blob* to be created only once, when the asset is filled using the *Asset Load Plaintext* (key provided in plaintext by the host) or *Asset Load Random* (key generated by MFP Cryptographic Module(A) using the DRBG) services.

7.2.2 Static assets

The MFP Cryptographic Module(A) provides functionality to program *static assets* or *public data objects* into the OTP via a special service only available to the Crypto Officer role and using an AES Key Blob using AES-CMAC and AES-CTR that is intended for provisioning. The provisioning Key Blob can be created outside MFP Cryptographic Module(A) using a special shared secret. Ownership is covered through the policy of assets that are intended to be written to OTP.

Static assets stored in OTP cannot be output. The OTP can also contain Public data objects which are accessible through its identifier.

Additionally, the MFP Cryptographic Module(A) includes two types of keys that are stored in the Firmware and can be input during the integration of the cryptographic module:

- Two RSA Authentication Public Keys with 2048 and 3072 bits.
- One 512-bit AES Provisioning Key (AES-CMAC, AES-CTR).

Note: The program RAM area for these keys in the Firmware is not considered for the integrity verification of the Firmware component of the MFP Cryptographic Module(A) .

7.3 Key access control and usage

The MFP Cryptographic Module(A) manages the concept of asset ownership and usage policy based on the following information provided during asset creation:

- *Host ID*: host that owns the asset
- *Protection bit*: indicates whether the user is a Crypto Officer or User role

- *Identity*: user ID that owns the asset
- *Usage Policy*: defines how the asset may be used (i.e. algorithm, mode and cryptographic operation)

Once created, the ownership and usage attributes of the key remains until the key is deleted from the asset store or the asset store is zeroized.

7.4 Key Agreement / Key Transport

The MFP Cryptographic Module(A) provides:

- the EC Diffie-Hellman key agreement scheme compliant with [SP800- 56A]. This key agreement scheme provides between 112 and 256 bits of security strength.
- EC Diffie-Hellman shared secret computation, compliant with SP800-56A and scenario 5 primitive only of IG D.8.

The MFP Cryptographic Module(A) also provides key wrapping. As shown in Table 15 and explained in the section on Dynamic assets, keys can be entered in encrypted form through the following key wrapping methods and key lengths:

- Key Wrapping with AES in KW and KWP modes with 128, 192 or 256-bit keys, compliant with [SP800- 38F]. Security strength ranges from 128 to 256 bits, according to the AES key length.
- Key Wrapping using AES in CMAC and CTR modes with two 256-bit keys (the first key for the AES-CMAC operation and the second key for the AES-CTR operation), compliant with [SP800-38F]. It provides a security strength of 256 bits.

The following table shows the maximum lengths and security strength of the keys that can be imported to and exported from the MFP Cryptographic Module(A) using the key agreement or wrapping services:

Key	Maximum Length	Security Strength
Trusted Root Key	256 bits	256 bits
AES key	256 bits	256 bits
HMAC key	1024 bits	256 bits
RSA key pair	3072 bits	128 bits
ECDSA key pair	521 bits	256 bits
EC Diffie-Hellman key pair	521 bits	256 bits

Table 16 - Security Strength of Cryptographic Keys

The maximum strength of the key wrapping schemes provided by the MFP Cryptographic Module(A) is greater than or equal to the security strength of the keys that need to be wrapped while entering or exiting the module. The maximum strength of the key agreement scheme provided by the MFP Cryptographic Module(A) is 256 bits.

Nevertheless, it is the user's responsibility to use the establishment method with an appropriate key size to ensure FIPS compliance. Using an insufficient AES key size for AES Key Wrapping or an insufficient ECDH key size for key agreement will reduce the security strength of the wrapped key, agreed upon key.

7.5 Key / CSP Zeroization

A dynamic asset (key or CSP) is deleted from the asset store using the *Asset Delete* service

and the memory where the asset was stored is zeroized.

Additionally, the whole asset store is zeroized when the MFP Cryptographic Module(A) transitions to the error state or when the module is powered-off.

Keys and CSPs considered static assets are stored in OTP; the MFP Cryptographic Module(A) provides a two-step process to zeroize the OTP memory. First, the service *Select One-Time-Programmable Zeroize* must be called to enable OTP zeroization. After OTP zeroization is enabled, the *Zeroize One-Time-Programmable* service fills the complete OTP with ones, and zeroizes the asset store.

The MFP Cryptographic Module(A) also provides the *Zeroize Output Mailbox* service can be used to zeroize the output mailbox before the mailbox is unlinked. This operation prevents sensitive material leaking to other Hosts applications.

Zeroization is performed filling the memory area with zeroes. The operation is performed in a time that is not sufficient to compromise CSPs. Additionally:

- The MFP Cryptographic Module(A) processes one input message at a time, when a zeroization service (Asset Delete, Zeroize Output Mailbox) is processed no other input message accessing the asset store can be executed.
- No information is output when the MFP Cryptographic Module(A) transitions to or is in the Error state.

7.6 Random Number Generation

The MFP Cryptographic Module(A) includes a Deterministic Random Bit Generator (DRBG) based on the CTR_DRBG (without prediction resistance; without derivation function) algorithm and AES-256 as the underlying cipher according to [SP800-90A]. The MFP Cryptographic Module(A) uses this engine to:

- Create symmetric keys in the asset store for the *Asset Load* service.
- Provide random data for the *TRNG Get Random* service.

The MFP Cryptographic Module(A) includes a NDRNG to provide entropy input to the DRBG. The Crypto Officer can seed or re-seed the DRBG using the *TRNG Configuration* service (the service can also start the TRNG engine in the same service).

It is also possible to trigger an automatic re-seed of the DRBG using the same service when requesting sufficient random data; in this case the operation can be performed by both the Crypto Officer and the User roles.

The DRBG is seeded with 384 bits from the NDRNG and provides 256 bits of security strength.

The MFP Cryptographic Module(A) performs continuous tests in the DRBG and NDRNG, verifying that the previous and current generated blocks of random data are not equal (section 9.3). The module also performs DRBG health tests according to section 11.3 of SP 800-90A.

7.7 True Random Number Generation

The TRNG engine is a NDRNG containing a hardware entropy source based on free-running ring oscillators. This NDRNG utilizes eight Free-Running Ring Oscillators (FROs) to supply the entropy needed to generate true random numbers.

The set of FROs provided in the Verilog RTL is ready for synthesis after instantiating cells from the chosen target technology. However, it is possible to optimize the FROs for the

specific target technology such that more entropy is generated. Customers should follow the instructions provided in the MFP Cryptographic Module(A) Integration Manual for this purpose.

Using the *TRNG Configuration* service, the Crypto Officer can configure and start the NDRNG to initialize the DRBG.

8 Electromagnetic Interference / Electromagnetic Compatibility (EMI/EMC)

According to 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, the MFP Cryptographic Module(A) is not subject to EMI/EMC regulations because it is a subassembly that is sold to an equipment manufacturer for further fabrication. That manufacturer is responsible for obtaining the necessary authorization for the equipment with the MFP Cryptographic Module(A) embedded prior to further marketing to a vendor or to a user.

9 Self-Tests

9.1 Power-Up Tests

After successful initialization of the SoC in which the MFP Cryptographic Module(A) is integrated, the MFP Cryptographic Module(A) automatically performs power-up tests without user intervention to ensure that the module is not corrupted and that the cryptographic algorithms within the module work as expected. During the execution of the power-up tests, services are not available and no data output is possible.

If the power-up tests succeed, then the MFP Cryptographic Module(A) becomes operational and crypto services are available; the fatal_error signal is set to 0.

If the power-up tests fail, then the MFP Cryptographic Module(A) transitions to the Error state and crypto services are not available or no data output is possible; the fatal_error signal is set to 1.

9.1.1 Firmware ROM Integrity tests

The MFP Cryptographic Module(A) verifies the integrity of the firmware in ROM using a dedicated 24-bit CRC algorithm. After power-up, the MFP Cryptographic Module(A) calculates the CRC of the firmware image and compares it against the last word of the firmware code image. If the CRC verification succeeds, the MFP Cryptographic Module(A) continues with the rest of the power-up tests; if the CRC verification fails, the module transitions into the Error state and no crypto services are available.

9.1.2 Firmware ROM Cryptographic Algorithm tests

Once the ROM contents have been successfully verified, ROM-based self-tests (table below) are automatically executed in preparation for the execution of the boot process.

Cryptographic Algorithm	Test
SHS	KAT SHA-256
ECDSA	KAT for ECDSA (NIST P-256) signature verification
AES	KAT AES-CTR, 256-bit, decrypt

Table 17 - ROM Power-up Tests

If any of the known answer tests fail (the calculated output does not equal the known answer), MFP Cryptographic Module(A) transitions to the Error state and no crypto services are available.

9.1.3 Firmware RAM Integrity tests

The MFP Cryptographic Module(A) verifies the integrity of the firmware image received by the host using ECDSA signature verification with a P-256 public key. If verification of the ECDSA digital signature succeeds, MFP Cryptographic Module(A) continues with the rest of the power-up tests; if the integrity test fails, the module does not proceed with the power-up. Instead the module enters the Error state and waits for a reset after which the module waits for a valid firmware image to be loaded.

9.1.4 Firmware RAM Cryptographic algorithm tests

The MFP Cryptographic Module(A) performs known answer tests (KAT) (table below) on all FIPS-Approved cryptographic algorithms that can be used for the VaultIP component except

for when it is noted that the KAT is performed for the EIP-38 component.

Cryptographic Algorithm	Test
AES (VaultIP component)	KAT AES-CBC, 128-bit, encryption KAT AES-CBC, 128-bit, decryption KAT AES-CCM, 192-bit, encryption KAT AES-CCM, 192-bit, decryption KAT AES-CMAC, 256-bit, MAC generation
AES (EIP-38 component)	KAT AES-XTS, 128-bit Encryption KAT AES-XTS, 128-bit Decryption KAT AES-XTS, 256-bit Encryption KAT AES-XTS, 256-bit Decryption
SHS	KAT SHA-1 KAT SHA-224 KAT SHA-384 KAT SHA-512
HMAC	KAT HMAC-SHA-256
RSA	KAT RSA 2048-bit (PKCS#1 v1.5), signature generation KAT RSA 2048-bit (PKCS#1 v1.5), signature verification
ECDH	KAT for Z computation (NIST P-224), HMAC-SHA-256, AES-CMAC-KAT (for Key Derivation Function).
ECDSA	KAT ECDSA (NIST P-224) signature generation KAT ECDSA (NIST P-224) signature verification
DRBG	KAT AES-CTR-256 DRBG
KBKDF	KAT SP800-108 KDF (PRF HMAC-SHA-256 in Feedback mode)
CRC32	KAT CRC32

Table 18 - Power-Up Tests

9.2 On-demand self-tests

In order to perform the on-demand self-tests as required by FIPS 140-2, the MFP Cryptographic Module(A) shall be power-off and power-on by doing a hard reset.

9.3 Conditional Tests

MFP Cryptographic Module(A) performs conditional tests on the cryptographic algorithms shown in the following table:

Algorithm	Test
DRBG	Continuous test
ECDSA key generation	Pair-wise consistency test (PCT)
NDRNG	Continuous test

Table 19 - Conditional Tests

If any of these tests fail, MFP Cryptographic Module(A) transitions to the Error state and no

crypto services are available.

9.4 Module status

MFP Cryptographic Module(A) provides different interfaces to show its status:

- Once in Error state the indicator is that no crypto services are available and all tokens except Reset Token will be rejected with an invalid token error. If a *Reset* Token is called in Error state, it will put the module in Fatal Error state. In this state none of the services are possible and the module needs to be power-off and power-on by doing a hard reset.
- During operational mode, the *System Info* token (*Show status* service) provides the value of the fatal_error signal (bit 31 of output token word 0 set to '1'), firmware version information (output token word 1 set to '0x00020503') and OTP anomalies in the 5th word of the output token. This service can be requested by both the User and Crypto-Officer roles.
- The output token returned by MFP Cryptographic Module(A) provides the result of processing the service requested by the User or Crypto-Officer role.

9.5 Error state

The MFP Cryptographic Module (A) supports two error states: Fatal Error and Error.

The module transitions to Error state for following reasons:

- Failure of the power-up self-test or on-demand self-tests.
- Failure when resuming from Sleep.
- Failure of the conditional tests.

While in the Error state, the module cannot perform any cryptographic services and no data output is possible. The Asset Store is also Zeroized. All tokens except *Reset Token* (which is a soft reset) will be rejected with an invalid token error.

To recover from the Error state the module shall be power-off and power-on by doing a hard reset.

The module transitions to Fatal Error state for following reason:

- use of a "soft reset operation" via the "Reset" token while in Error state.

In the Fatal Error state, services are not available and no data output is possible.

To recover from the Fatal Error state the module shall be power-off and power-on by doing a hard reset.

10 Design Assurance

10.1 Configuration Management

Kyocera uses a variety of tools for configuration management. These are listed below.

Version Control Tool	Object(s)
CVS (Concurrent Versioning System)	RTL
Git version control	RAM Firmware
Perforce	ROM Firmware
MS Sharepoint	Documentation

Table 20 - Change and Version Control

10.1.1 Cryptographic Module Identification

MFP Cryptographic Module(A) is uniquely identified by the filenames used to ship the IP core Verilog code and the firmware image. The following convention is used:

<nn>[<o>]_HW<x.y.z> [_(alpha|beta|final)]

<nn>[<o>]_Firmware[_cfg<O>]_FW<a.b.c> [_(alpha|beta|final)]

where:

- <nn> is the “EIP number” in Kyocera’s IP catalog.
- <o> or <O> is a code for specific configuration options, if any. Please see the Data Sheet or Hardware Reference Manual for a detailed explanation of this code, see the Release Notes to see the configuration details for this delivery.
- <x.y.z> is the hardware version number of the IP in this delivery. The 3rd digit is only used for patches to the original <x.y> version.
- <a.b.c> is the firmware version number included in this delivery. The 3rd digit is only used for patches to the original <a.b> version.
- (alpha|beta|final) is used to distinguish intermediate drops leading towards the final release. “_final” is omitted if no intermediate drops are done or if it is a generic release.

10.1.2 Guidance Identification

Kyocera uniquely identifies each document with the document name, document number (e.g. 007-130300-201) and Revision (RevA, RevB, etc.).

10.1.3 Source Code Identification

Source code is identified and controlled by the configuration management tool as listed in Table 20. Proper keywords are included in each source code file to provide the filename and revision configuration item.

10.2 Delivery and Operation

MFP Cryptographic Module(A) is synthesized in silicon within the MFP Cryptographic Module(A) embodiment. It is a single chip hardware module. The chip is delivered from the vendor via a trusted delivery courier. Upon reception of MFP Cryptographic Module(A), the customer should verify that the package does not have any irregular tears or openings.

The chip comes preloaded with the following code packages:

- 915-130015-210_VaultIP-130-015_HW2.1.10.zip
- 914-130915-227_VaultIP-130-015_Firmware-cfgA_FW2.2.18.zip
- SafeXcel-IP-38x-2_HW3.2.3.zip
- vaultIP-1xx_eip38_wrapper.zip

The Crypto Officer ID, which is embedded in the Firmware, will be provided in the package and the *Define User* service can be used to create the user roles. On power-up, the module automatically performs power-up self-tests; successful completion of the integrity checks within the power-up tests ensures the integrity of MFP Cryptographic Module(A).

10.3 Guidance

For the purpose of this cryptographic module validation, MFP Cryptographic Module(A) is synthesized in silicon within the MFP Cryptographic Module(A) embodiment and validated as a single-chip hardware module.

Kyocera provides the following documentation describing the functionality of MFP Cryptographic Module(A) :

Document Name	Document Number
SafeXcel-IP-38_HW3.2_Integration-Manual_RevA	007-038320-200
SafeXcel-IP-38_HW3.2_Hardware-Reference-and-Programmer-Manual_RevD	007-038320-207
SafeXcel-IP-38_HW3.2_Hardware-Reference-Manual_Integration-Addendum_RevB	007-038320-207/1
VaultIP-1XX_HW2.1_Integration-Manual_RevC	007-130210-200
VaultIP-1XX_HW2.1_Hardware-Reference-Manual_RevD	007-130210-201
VaultIP-1XX_FW2.2_Firmware-Reference-Manual_RevE	007-130220-204
VaultIP-1XX_FW2.2_Software-Integration-Manual_RevD	007-130220-312
SafeZone-SecureBoot-Solution_v3.8_Developer-Guide_RevD	007-916380-306

Table 21 - MFP Cryptographic Module(A) Documentation

10.3.1 Crypto Officer Guidance

For the validated module, at each initialization, the CO should verify the module is operating in FIPS validated configuration by performing the following steps:

- The CO should verify with *System Info* token that the MODULE_STATUS register value shows 0x00000202. This indicates the successful completion of the power-up tests.
- Next the CO must send a Self-Test token to initialize the User role authentication. The successful completion of this step must be verified with *System Info* token confirming that the MODULE_STATUS register shows 0x00000201 value.

The Use of the module without these steps will be considered as a non-validated module.

10.3.2 AES XTS

The AES algorithm in XTS mode implemented in the EIP-38 sub-module can be only used for the cryptographic protection of data on storage devices, as specified in [SP800-38E].

To meet the requirement in [FIPS140-2_IG] A.9, the module implements a check to ensure that the two AES keys used in the AES-XTS algorithm are not identical.

11 Mitigation of Other Attacks

The cryptographic module does not implement security mechanisms to mitigate other attacks.

A Appendixes

A.1 Glossary and Abbreviations

AES	Advanced Encryption Specification	**IRQ**	Interrupt ReQuest
CAVP	Cryptographic Algorithm Validation Program	**KAT**	Known Answer Test
CBC	Cipher Block Chaining	**KBKDF**	Key-based Key Derivation Function
CCM	Counter with Cipher Block Chaining-Message Authentication Code	**MAC**	Message Authentication Code
CFB	Cipher Feedback	**NIST**	National Institute of Science and Technology
CMVP	Cryptographic Module Validation Program	**NVLAP**	National Voluntary Laboratory Accreditation Program
CSP	Critical Security Parameter	**OFB**	Output Feedback
CTR	Counter Mode	**OTP**	One-Time-Programmable memory
CVL	Component Verification List	**PSS**	Probabilistic Signature Scheme
DES	Data Encryption Standard	**RNG**	Random Number Generator
DMA	Direct Memory Access	**RSA**	Rivest, Shamir, Addleman
DSA	Digital Signature Algorithm	**RTL**	Register Transfer Level
FIPS	Federal Information Processing Standards Publication	**SHA**	Secure Hash Algorithm
GCM	Galois/Counter Mode	**SHS**	Secure Hash Standard
FRO	Free Running Oscillator	**SOC**	System on a Chip
HMAC	Hash Message Authentication Code	**SSH**	Secure Shell
IP	(semiconductor) Intellectual Property (core)	**TCM**	Tightly Coupled Memory
		TEE	Trusted Execution Environment

A.2 References

FIPS140-2 **FIPS PUB 140-2 - Security Requirements For Cryptographic Modules**
May 2001
http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

FIPS140-2_IG **Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program** Aug 2020
http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf

FIPS180-4 **Secure Hash Standard (SHS)**
March 2012
http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

FIPS186-4 **Digital Signature Standard (DSS)**
July 2013
http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

FIPS197 **Advanced Encryption Standard**
November 2001
http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

FIPS198-1 **The Keyed-Hash Message Authentication Code (HMAC)**
July 2008
http://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

PKCS#1 **Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1** February 2003
http://www.ietf.org/rfc/rfc3447.txt

RFC3394 **Advanced Encryption Standard (AES) Key Wrap Algorithm**
September 2002
http://www.ietf.org/rfc/rfc3394.txt

RFC5649 **Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm** September 2009
http://www.ietf.org/rfc/rfc5649.txt

SP800-38A **NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation - Methods and Techniques**
December 2001
http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

SP800-38B **NIST Special Publication 800-38B - Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication**
May 2005
http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf

SP800-38C **NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality**
May 2004
http://csrc.nist.gov/publications/nistpubs/800-38C/SP800-38C_updated-July20_2007.pdf

SP800-38D **NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC**
November 2007
http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

SP800-38E **NIST Special Publication 800-38E - Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices**
January 2010
http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf

SP800-38F **NIST Special Publication 800-38F - Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping**
December 2012
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf

SP800-56A **NIST Special Publication 800-56A Revision 2 - Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography**
May 2013
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Ar2.pdf

SP800-56C **NIST Special Publication 800-56C Revision 1 - Recommendation for Key-DerivationMethods in Key- Establishment Schemes**
April 2018
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Cr1.pdf

SP800-90A **NIST Special Publication 800-90A – Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators**
January 2015
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf

SP800-108 **NIST Special Publication 800-108 - Recommendation for Key Derivation Using Pseudorandom Functions**
October 2009
http://csrc.nist.gov/publications/nistpubs/800-108/sp800-108.pdf