Samsung Flash Memory Protector

Version 1.2.1

FIPS 140-2 Non-Proprietary Security Policy

Version 1.4

Last Update: 2017-10-03

TABLE OF CONTENTS

1.Introduction

This document is the non-proprietary FIPS 140-2 Security Policy for the Samsung Flash Memory Protector cryptographic module. It 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 1 Software-Hybrid cryptographic module. This security policy is for the validation of the Samsung Flash Memory Protector module to include information about new test platform and minor code changes for performance enhancement.

In this document, the terms “Samsung Flash Memory Protector”, “cryptographic module” or “module” are used interchangeably to refer to the Samsung Flash Memory Protector.

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 the 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 people:

- Developers working on the release
- FIPS 140-2 testing laboratory
- Cryptographic Module Validation Program (CMVP)
- Consumers

1.3 Document Organization / Copyright

This non-proprietary security policy document may be reproduced and distributed only in its original entirety without any revision, ©2017 Samsung Electronics Co., Ltd.

2.Cryptographic Module Specification

2.1.Description of Module

The Samsung Flash Memory Protector is classified as a multi-chip standalone software-hybrid module for FIPS 140-2 purposes. It is designed for the on-the-fly hardware encryption to flash memory including Disk Encryption, File Encryption and Dual Encryption (i.e. both File and Disk Encryption.) The logical cryptographic boundary for the module includes the cryptographic module software backed by hardware cryptography to support hardware-based cryptographic algorithms. The module includes the following components:

- Flash Memory Protector Driver
- Flash Memory Protector

The Flash Memory Protector Driver (FMP Driver) is the software component of the cryptographic module running in the Linux Kernel which calls the cryptographic algorithms implemented in the FMP hardware module for power-up self-tests, and also holds the FIPS status of the entire cryptographic module once the power-up self-test is completed successfully. All actual cryptographic functions are implemented within the hardware.

The Flash Memory Protector (FMP) is the hardware component of the cryptographic module which supports AES with CBC mode and XTS-AES encryption and decryption for confidentiality on storage devices. It resides in the Samsung Exynos Processor. It is located between the external DRAM and Flash memory Device so that it can encrypt and decrypt the data quickly and reduce power consumption.

The hybrid cryptographic module is specified in the following table:

Component	Type	Version Number	Parts Number / File Name
FMP Driver	Software	1.3.1	boot.img
FMP	Hardware	3.0	Samsung Exynos Processor 7570
			Samsung Exynos Processor 7870

Table 1: Components of the Hybrid Cryptographic Module

The module has been tested on the following platforms:

Device	Processor	O/S & Ver.
Samsung Galaxy J3	Exynos 7570	Android 7.0
Samsung Galaxy Tab Active2	Exynos 7870	Android 7.1

Table 2: Tested Platforms

The module is intended to meet the requirements of FIPS 140-2 Security Level 1 Overall. The table below shows the security level claimed for each of the eleven sections that comprise the validation:

FIPS 140-2 Sections	Security Level				
	N/A	1	2	3	4
Cryptographic Module Specification		X			
Cryptographic Module Ports and Interfaces		X			

Roles, Services and Authentication		X			
Finite State Model		X			
Physical Security		X			
Operational Environment		X			
Cryptographic Key Management		X			
EMI/EMC				X	
Self Tests		X			
Design Assurance				X	
Mitigation of Other Attacks	X				

Table 3: Security Levels

2.2.Description of Approved Mode

The module supports only FIPS mode.

When the module is initialized during the kernel boot-up, the power-up self-test (including both software and hardware components) is executed automatically without any operator intervention. The module enters FIPS mode automatically if the power-up self-test completes successfully.

If any self-tests fail during power-up, the module sets a global flag to FMP_FIPS_ERR_STATE and goes into Error state. All cryptographic services are prohibited in error state.

The status of the module can be determined using the following ADB command:

```
adb shell cat /sys/devices/fips-fmp/fips-fmp/fips_fmp_status
```

If the module is in FIPS mode, the above command returns the string "passed" to indicate that the power-up self-tests completed successfully. If the module is in an Error state, the above command returns the string "failed".

The module provides the following CAVP validated algorithms implemented in the FMP hardware module and FMP Driver:

Algorithm	CAVP Cert	Standard	Mode/Method	Key Lengths	Use
AES	4407	FIPS 197	CBC	128 and 256 bits	Hardware encryption and decryption by FMP
		SP 800-38E	XTS-AES	128 and 256 bits	Hardware encryption and decryption by FMP
HMAC	3126 and 3127	FIPS 198-1	SHA-256	344	Firmware integrity test for FMP Driver
SHA	3854 and 3855	FIPS 180-4	SHA-256		Firmware integrity test

Table 4: Approved Algorithms

Note: According to SP 800-38E, the XTS-AES mode was designed only for the cryptographic protection of data on storage devices. This mode is only approved for storage application. The module also checks the XTS-AES key (i.e., concatenation of two AES keys: Key_1 and Key_2) to ensure Key_1 is different than Key_2.

Note: The SHA and HMAC implementations in the module are only used for Integrity Test of the module during power-on.

2.3.Cryptographic Module Boundary

The physical boundary of the module is the physical boundary of the device that contains the module. Consequently, the embodiment of the module is a multi-chip standalone cryptographic module.

2.3.1.Software Block Diagram

In the following diagram, the bidirectional arrows depict the flow of the status, control and data. The operations within the logical cryptographic boundary (the blue dotted region) use the cipher from the hardware that is included in the logical boundary.

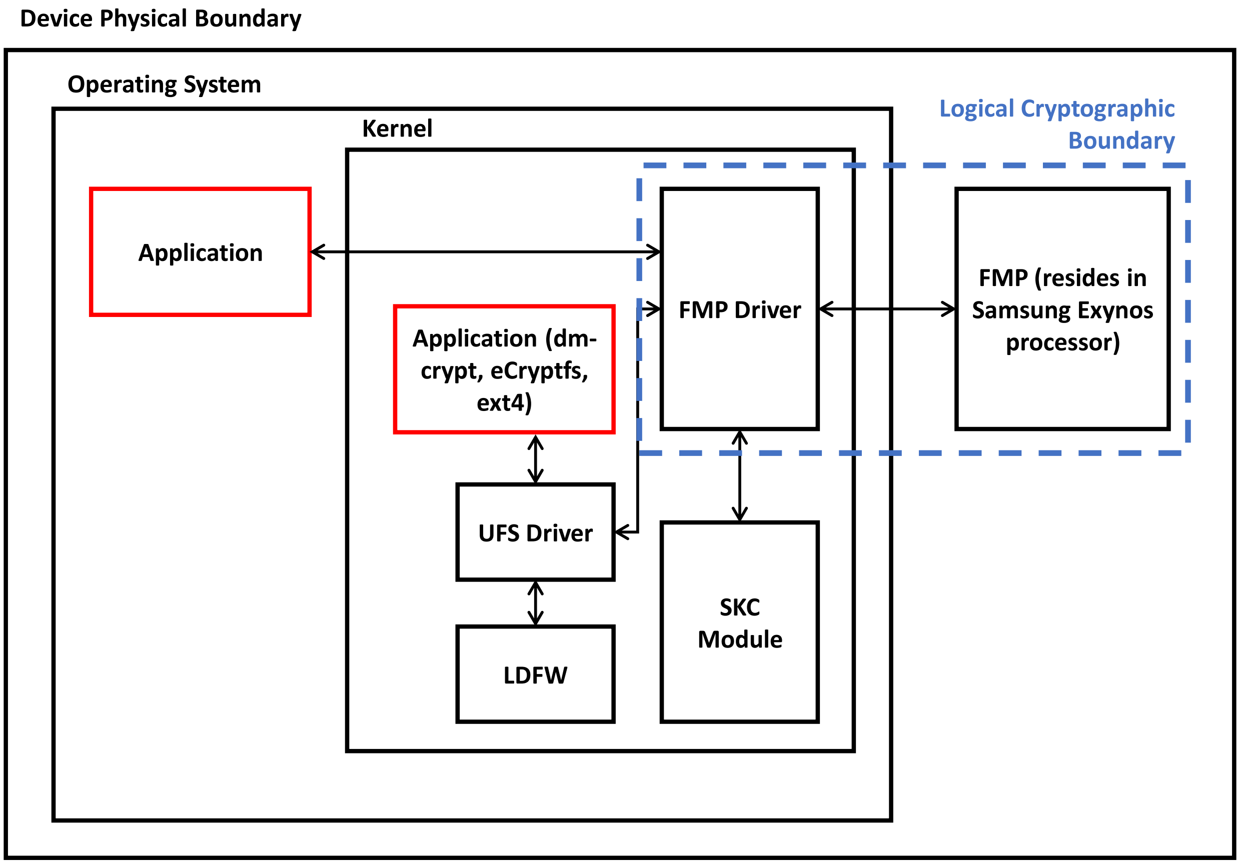
Device Physical Boundary
Operating System
Kernel
Logical Cryptographic Boundary
Application
FMP Driver
FMP (resides in Samsung Exynos processor)
Application (dm-crypt, eCryptfs, ext4)
UFS Driver
LDFW
SKC Module

Figure 1: Cryptographic Boundary

2.3.2.Hardware Block Diagram

The following figure illustrates the various data, status and control paths through the cryptographic module. Inside, the physical boundary of the module, the mobile device consists of standard integrated circuits, including processors and memory. These do not include any security-relevant, semi- or custom integrated circuits or other active electronic circuit elements.

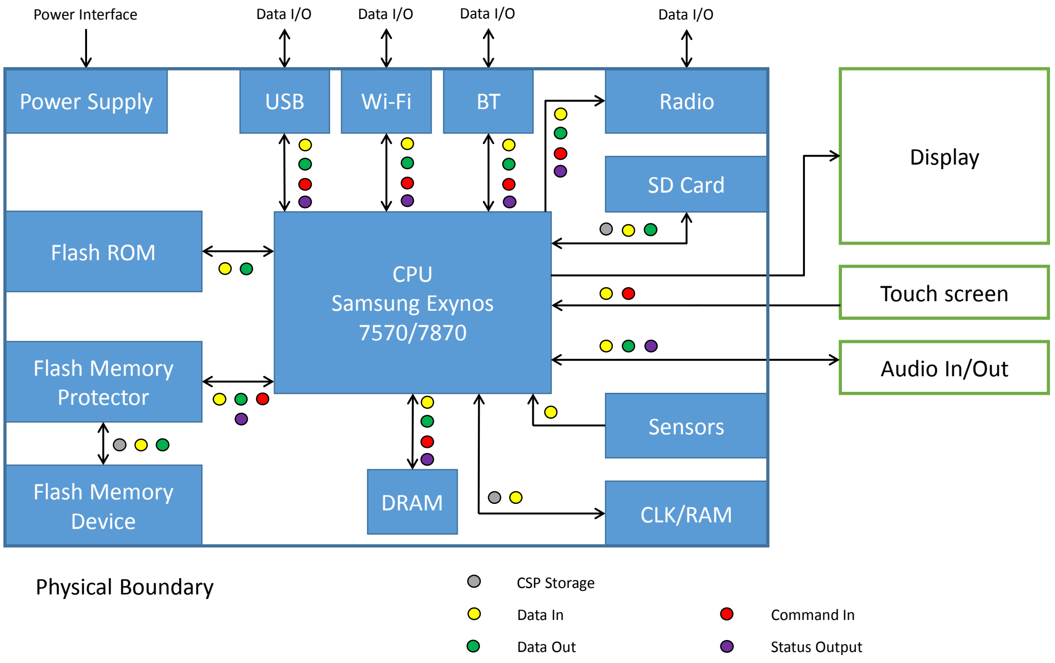

Figure 2: Device Physical Boundary

Figure 3: Samsung Exynos Processor 7570

Figure 4: Samsung Exynos Processor 7870

In the following diagram, the bidirectional arrows depict the flow of the status, control and data. The CSPs and settings are passed via memory and processed by the FMP hardware component.

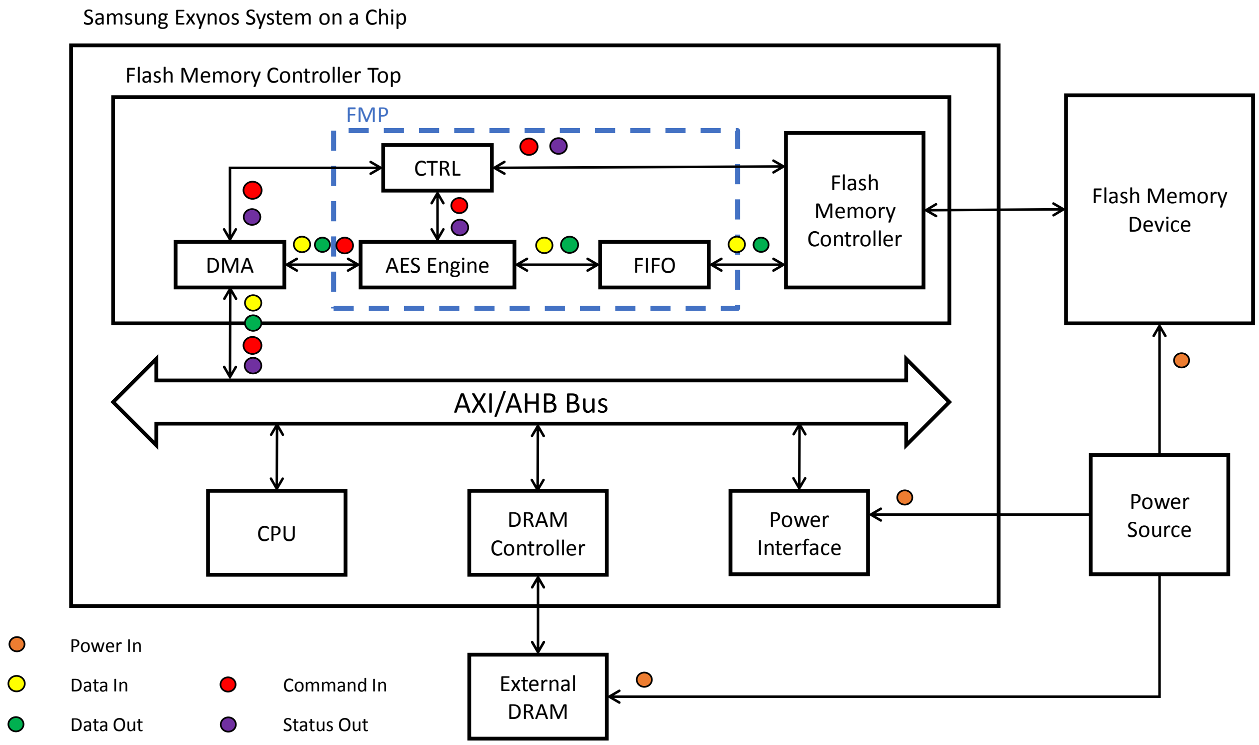

Figure 5: FMP Hardware Block Diagram

3.Cryptographic Module Ports and Interfaces

FIPS Interface	API Interface	Physical Interface
Data Input	API input parameters	FIFO
Data Output	API output parameters	FIFO
Control Input	API function calls	CTRL
Status Output	API return codes, kernel log messages, `/sys/devices/fips-fmp/fips-fmp/fips_fmp_status`	CTRL
Power Input	Physical power connector	CPU power pins

Table 5: Ports and Interfaces

4.Roles, Services and Authentication

4.1.Roles

Role	Description
User	Perform general security services, including cryptographic operations and other Approved security functions.
Crypto Officer (CO)	Perform initialization of Module.

Table 6: Roles

The module meets all FIPS 140-2 level 1 requirements for Roles and Services, implementing both User and Crypto Officer roles. The module does not allow concurrent operators.

The User and Crypto Officer roles are implicitly assumed by the entity accessing services implemented by the module. No further authentication is required. The Crypto Officer can initialize the module.

4.2.Services

The module does not support bypass capability. The DW3 field in the DMA descriptor in memory stores the flag for mode and algorithm that will be used for Disk Encryption and/or File Encryption in the FMP. When security functions are requested from the FMP (i.e. the mobile device is set for file or disk encryption), all plaintext entered in the FMP will be encrypted. When the mobile device is not configured for Disk Encryption nor File Encryption, no security function is requested from the FMP and no plaintext enters the logical boundary or the FMP hardware boundary.

The following table describes the services available in FIPS-Approved mode:

Service	**Roles**		**Keys/CSPs**	**Access (Read "R", Write "W", Execute "X")**
	User	**CO**		
Hybrid				
AES encryption and decryption with CBC mode for File Encryption *Input:* DMA descriptor *Output:* Return success or fail	✓		AES CBC key for File Encryption	R, X
XTS-AES encryption and decryption for File Encryption *Input:* DMA descriptor *Output:* Return success or fail	✓		XTS-AES key for File Encryption	R, X
XTS-AES encryption and decryption for Disk Encryption *Input:* Register *Output:* Return success or fail	✓		XTS-AES key for Disk Encryption	R, X
Software				

Service	Roles		Keys/CSPs	Access (Read "R", Write "W", Execute "X")
	User	CO		
Self-Test (Self-test is executed automatically when device is booted or restarted)	✓		HMAC keys for Integrity Tests, AES keys for Known-Answer Tests	R, X
Check Status/Get State	✓		N/A	X
Zeroization	✓		DMA descriptors to AES CBC key for File Encryption, XTS-AES key for File Encryption and XTS-AES key for Disk Encryption	W
Module Initialization		✓	N/A	N/A

Table 7: Approved Services

4.3.Operator Authentication

There is no operator authentication; assumption of role is implicit by action.

4.4.Mechanism and Strength of Authentication

No authentication is required at security level 1; authentication is implicit by assumption of the role.

5.Physical Security

The Samsung Flash Memory Protector is a software-hybrid module that operates on a multi-chip standalone platform, which conforms to the Level 1 requirements for physical security. The hardware portion of the cryptographic module is a production grade component. The cryptographic module must be used in a commercial off the shelf (COTS) mobile device. The mobile device shall be comprised of production grade components with standard passivation (a sealing coat applied over the chip circuitry to protect it against environmental and other physical damage) and a production grade enclosure that completely surrounds the cryptographic module.

6.Operational Environment

The operating system shall be restricted to a single operator mode of operation. The procurement, build and configuring procedure are controlled. The module is installed into a commercial off-the-shelf (COTS) mobile device by the customer.

6.1.Policy

The operating system shall be restricted to a single operator mode of operation (i.e., concurrent operators are explicitly excluded). The external application that makes calls to the cryptographic module is the single user of the cryptographic module, even when the application is serving multiple clients.

7.Cryptographic Key Management

7.1.Key and CSP List

The CSPs are provided in a DMA descriptor that is setup by the third party applications. The DMA descriptor includes the following information: the address for the data blocks, data length, key length, mode, algorithm, Disk Encryption IV, File Encryption IV, File Encryption Key and File Tweak Key. The Flash Memory Controller (FMC) reads the information in the DMA descriptor and passes it into the FMP module. The Disk Encryption Key is stored in a firmware (LDFW) and loaded into a register for FMP when Disk Encryption services are requested.

The following table lists the Keys and CSPs in the module:

CSP/Key	Size	Entry/ Output	Storage	Zeroization
AES CBC key for File Encryption	128 or 256 bits	CSPs are read in the FMP from flash, via the FMC, at the hardware level.	Third party applications store the keys in the DMA descriptor inside the logical boundary.	Call the zeroization API to clear out the sensitive information in the DMA descriptor.
XTS-AES key for File Encryption				
XTS-AES key for Disk Encryption	128 or 256 bits	The LDFW loads the key to the register for the FMP module to read.	The key is stored in a firmware that is outside of the module. The third party applications load the key in the register before requesting the cryptographic operations provided by the FMP.	Key is zeroized when the host device powers off.
DMA descriptor		Set up by third party application	Stored as plaintext in memory	Call the zeroization API to clear out the sensitive information in the DMA descriptor.
Note: The following keys are not a CSP according to IG 7.4 as they are only used for integrity test and known-answer tests during power-up of the module.				
HMAC key for Integrity Test	344 bits	N/A	Stored as plaintext within the FMP Driver binary.	These keys are not subject to key zeroizatons according to IG 7.4.
AES keys for Known-Answer Test	128 or 256 bits	N/A	Stored as plaintext within the FMP Driver binary.	

Table 8: Keys and CSPs

7.2.Key/CSP Generation, Entry and Output

The module does not provide any key generation service or perform key generation for any of its Approved algorithms. The keys are provided by the third party applications and stored in a DMA descriptor located in the memory.

The cryptographic module does not provide any asymmetrical algorithms or key establishment methods. Manual key entry or key output capabilities are not provided. All CSPs can only be exchanged in the memory via a DMA descriptor which occurs within the physical boundary of the device and therefore may be passed to the module in plaintext.

7.3.Key/CSP Storage and Zeroization

As all CSPs are stored in a DMA descriptor located in the memory, it is the user's responsibility to destroy the sensitive information in the DMA descriptor using FIPS Pub 140-2 compliant procedures. The cryptographic module itself does not destroy externally stored keys and secrets since it does not own these CSPs. The zeroization API function is provided by the module to clear out the sensitive information stored in PRD table for DMA descriptor with 0s. No CSP is passed in the FMP Driver.

8.Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC)

The module is a software-hybrid module that has been tested on the test platforms listed in section 2.1. The Samsung FMP hardware component cannot be certified by the FCC as it is not a standalone device. The test platforms are accepted by the FCC with the following information:

Test Firm: Samsung Electronics EMC Laboratory

Test Firm Registration Number: #451343

The test platform which runs the module conforms to the EMI/EMC requirements specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class B (i.e., for home use).

FCC ID: A3LSMJ327A (for Samsung Galaxy J3)

FCC ID: A3LSMT395 (for Samsung Galaxy Tab Active2)

9.Power-Up Tests

Power-Up tests consist of software integrity tests and known-answer tests (KAT) of algorithm implementations. The power-up self-tests are automatically performed without any operator intervention during loading of the module. If any of the power-up self-tests fail, the module will return the error codes to the calling application, print the specific error message in the kernel log, set the global variable fmp_fips_state to FMP_FIPS_ERR_STATE and set the global variable fips_fmp_result to 0 which the status Linux command listed in section 2.2 will return the string "failed" to indicate the module is in error state. Data output is prohibited and no further cryptographic operation is allowed in the error state. To recover from the error state, re-initialization is possible by doing a reboot to set the module to the power-on state.

If the power-up self-tests are completed successfully, the module will set the global variable fmp_fips_state to FMP_FIPS_SUCCESS_STATE and set the global variable fips_fmp_result to 1 which the status Linux command will return the string "passed" to indicate the module is in FIPS mode.

FIPS 140-2 explicitly allows that the on-demand test can be fulfilled with a power cycle of the module. Hence, a power cycle and its associated power-on self-test is the methodology used to perform the "on-demand" tests.

9.1.Cryptographic Algorithm Tests

Algorithm	**Test**
AES CBC 128 bits	KAT – Encryption and Decryption are tested separately
AES CBC 256 bits	KAT – Encryption and Decryption are tested separately
XTS-AES 128 bits	KAT – Encryption and Decryption are tested separately
XTS-AES 256 bits	KAT – Encryption and Decryption are tested separately
SHA-256	KAT
HMAC-SHA-256	KAT

Table 9: Power-Up Cryptographic Algorithm Tests

9.2.Integrity Tests

9.2.1.Integrity Tests for FMP Driver

At build time -

- Calculate the HMAC-SHA-256 values of only the FMP Driver code (available in .text, .rodata and .init.text sections of kernel image) with different offset values which define the kernel position at boot.
- Once generated, the build time HMAC-SHA-256 values of the FMP Driver code with their associated offset values are updated in the kernel image itself as a read-only data.

At run time –

- Calculate a HMAC-SHA-256 value of only the module code, available in running kernel on the device (.text, .init.text and .rodata sections of running kernel)
- Retrieve the build time HMAC-SHA-256 value of the module based on the offset value in the current kernel position.
- If this calculated run-time HMAC-SHA-256 value is the same as the build time HMAC-SHA-256 value, Integrity test is considered passed. If calculated and stored values do not match, set error state, FMP_FIPS_ERR_STATE.

10.Design Assurance

10.1.Configuration Management

10.1.1.Versioning for FMP Driver

Perforce is used as the repository for both source code and documents for the FMP Driver. All source code and documents are maintained in an internal server. Release is based on the Changelist number, which is auto-generated. Every check-in process creates a new Changelist number.

Versions of controlled items include information about each version. For documentation, document version and date inside the document provide the current version of the document. Version control maintains the all the previous versions and the version control system automatically numbers revisions. For source code, unique information is associated with each version such that source code versions can be associated with binary versions of the final product. The source code of the module available in the Samsung internal Perforce repository is used to build the target binary.

10.1.2.Versioning for FMP

IMS IP configuration management system is a Samsung in-house management system used to maintain documentation for the FMP and the delivery of different versions of the FMP hardware design to the chip development team. Each version of IMS is composed as {Module Category}_{Module Name}_{Version Number}.

The individual design files of the FMP hardware are maintained by Perforce in an internal server. Every check-in process creates a new Changelist number, which is auto-generated, to keep track of the changes of the individual design files of the FMP hardware.

10.2.Delivery and Operation

The cryptographic module is never released as source code. The module sources are stored and maintained at a secure development facility with controlled access.

The FMP Driver is a built-in kernel module within the device Linux kernel image. A product that does not need a FIPS 140-2 certified cryptographic module may decide to change the build flag CONFIG_FIPS_FMP in Kernel config. The development team and the manufacturing factory share a secured internal server for exchanging binary software images. The factory is also a secure site with strict access control to the manufacturing facilities. The module binary is installed on the mobile devices (phone and tablets) using direct binary image installation at the factory. The mobile devices are then delivered to mobile service operators. Users cannot install or modify the module.

Samsung vets all service providers and establishes secure communication with them for delivery of tools and software updates. If the binary is modified by an unauthorized entity, the device has a feature to detect the change and thus not accept the binary modified by the unauthorized entity.

11.Mitigation of Other Attacks

No other attacks are mitigated.

12.Glossary and Abbreviations

AES	Advanced Encryption Specification
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
COTS	Commercial Off The Shelf
CSP	Critical Security Parameter
DMA	Direct Memory Access
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standards Publication
FMC	Flash Memory Controller
FMP	Flash Memory Protector
HMAC	Hash Message Authentication Code
IV	Initial Vector
KAT	Known Answer Test
NIST	National Institute of Science and Technology
O/S	Operating System
OTA	Over-The-Air
PRD	Physical Region Description
SHA	Secure Hash Algorithm
UFS	Universal Flash Storage
XTS	XEX Tweakable Block Cipher with Ciphertext Stealing

13.References

[1] FIPS 140-2 Standard, https://csrc.nist.gov/publications/detail/fips/140/2/final

[2] FIPS 140-2 Implementation Guidance, https://csrc.nist.gov/CSRC/media/Projects/Cryptographic-Module-Validation-Program/documents/fips140-2/FIPS1402IG.pdf

[3] FIPS 140-2 Derived Test Requirements, https://csrc.nist.gov/CSRC/media/Projects/Cryptographic-Module-Validation-Program/documents/fips140-2/FIPS1402DTR.pdf

[4] FIPS 197 Advanced Encryption Standard, https://csrc.nist.gov/publications/detail/fips/197/final

[5] FIPS 180-4 Secure Hash Standard, https://csrc.nist.gov/publications/detail/fips/180/4/final

[6] FIPS 198-1 The Keyed-Hash Message Authentication Code (HMAC), https://csrc.nist.gov/publications/detail/fips/198/1/final

[7] SP 800-38E Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices, https://csrc.nist.gov/publications/detail/sp/800-38e/final