Advanced Micro Devices (AMD)

AMD ASP Cryptographic CoProcessor ("Phoenix")

FIPS 140-3 Non-Proprietary Security Policy

Prepared for:

Advanced Micro Devices (AMD)
2485 Augustine Drive
Santa Clara, CA 95054
www.amd.com

Prepared by:

atsec information security corporation
4516 Seton Center Parkway, Suite 250
Austin, TX 78759

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This section is informative to the reader to reference cryptographic services and other services of AMD Ryzen PRO 7000 Series ASP Cryptographic CoProcessor (the "module") from Advanced Micro Devices (AMD) (the "vendor"). Only the components listed in Section 2.1 are subject to the FIPS 140-3 validation. The CMVP (Cryptographic Module Validation Program) makes no statement as to the correct operation of the module or the security strengths of the generated keys (when supported) if the specific operational environment is not listed on the validation certificate.

The vendor has provided the non-proprietary Security Policy of the cryptographic module, which was further consolidated into this document by atsec information security together with other vendor-supplied documentation. In preparing the Security Policy document, the laboratory formatted the vendor-supplied documentation for consolidation without altering the technical statements therein contained. The further refining of the Security Policy document was conducted iteratively throughout the conformance testing, wherein the Security Policy was submitted to the vendor, who would then edit, modify, and add technical contents. The vendor would also supply additional documentation, which the laboratory formatted into the existing Security Policy, and resubmitted to the vendor for their final editing.

1.2 Security Levels

Section	Security Level
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	N/A
9	1
10	1
11	1
12	N/A

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The AMD Ryzen PRO 7000 Series ASP Cryptographic CoProcessor (hereafter referred to as "the module") supports the Ryzen PRO 7000 Series SoC (System on a Chip) by providing digital signature verification of the key database during secure boot procedures.

Module Type: Firmware-hybrid

Module Embodiment: SingleChip

Module Characteristics:

Cryptographic Boundary:

The cryptographic boundary of the module is defined as the fips_module binary, which performs self-tests, provides the service indicator, and shows status service, as well as the hardware implementations of RSA and SHA2-384 in the CCP, which are used to perform signature verification and verify the integrity of the fips_module binary.

Tested Operational Environment's Physical Perimeter (TOEPP):

The TOEPP of the module is defined as the Ryzen PRO 7000 Series SoC in which the module operates.

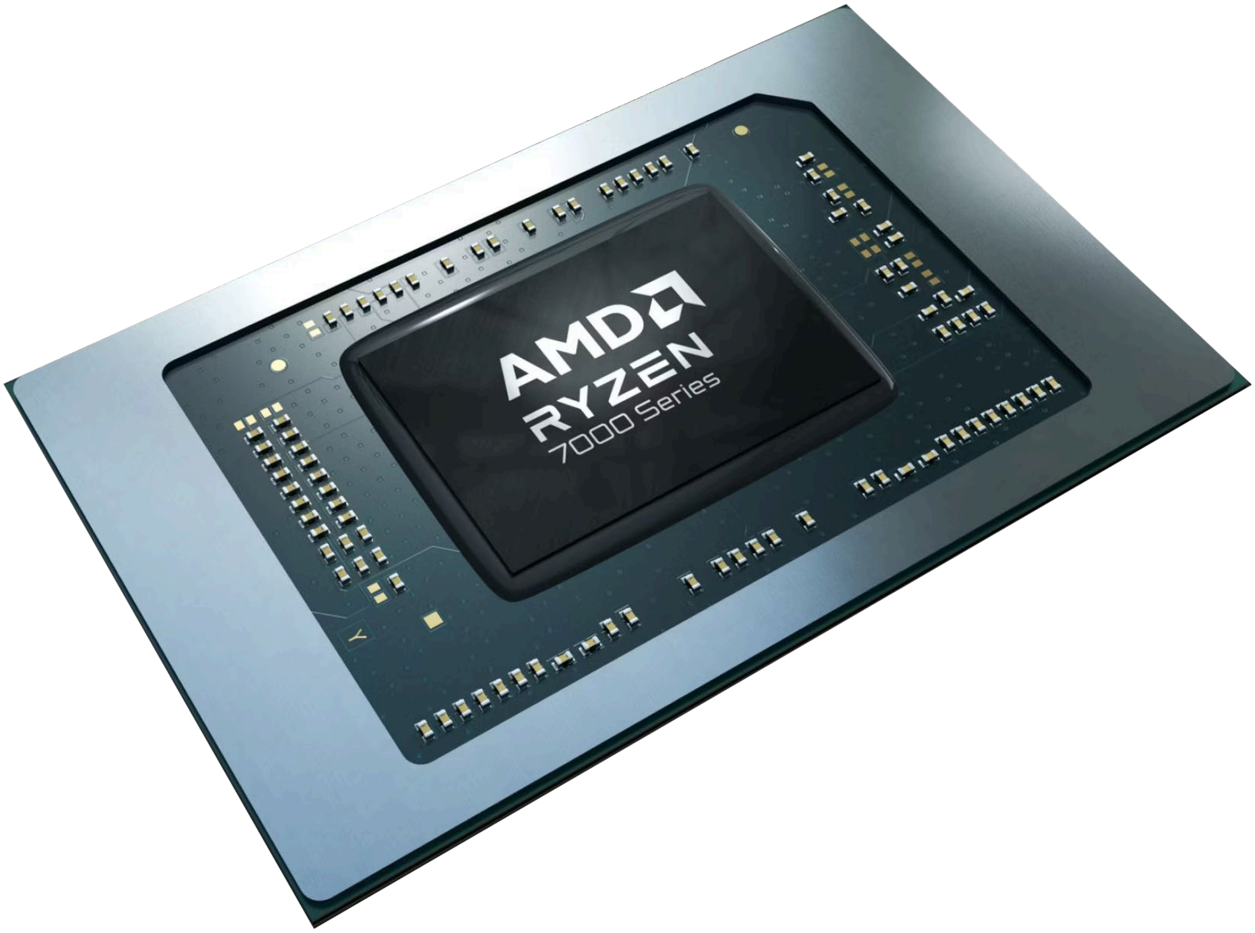


Figure 1: The AMD Ryzen PRO 7000 Series SoC, representing all versions of the tested platforms.

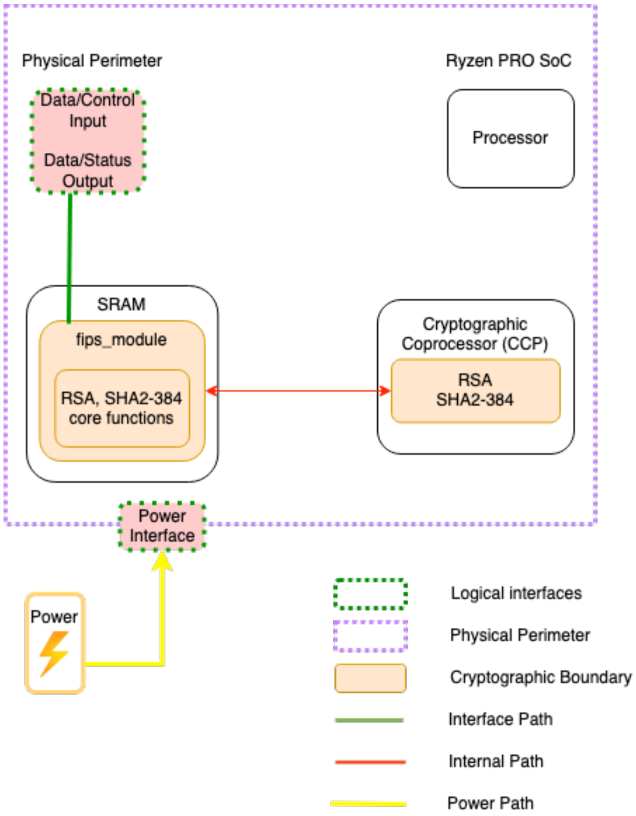


Figure 2: Block Diagram

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

N/A for this module.

Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets):

Package or File Name	Software/ Firmware Version	Features	Integrity Test
fips_module.bin	bc0d0443FIPS001		SHA2-384

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

Tested Module Identification – Hybrid Disjoint Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
RSA and SHA2-384 implementations in CCP	bc0d0346FIPS001		AMD Ryzen PRO 7640HS (100-000000960)	

Table 3: Tested Module Identification – Hybrid Disjoint Hardware

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
N/A	AMD Ryzen PRO 7640HS (100-000000960)	AMD Ryzen PRO 7640HS (100-000000960)	No	N/A	bc0d0346FIPS001

Table 4: Tested Operational Environments - Software, Firmware, Hybrid

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:

N/A for this module.

2.3 Excluded Components

There are no components within the cryptographic boundary that are excluded from the FIPS 140-3 security requirements.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved mode	Whenever the module is operational	Approved	The module always operates in the approved mode

Table 5: Modes List and Description

The module implements only one mode of operation, the approved mode, in which the approved services are available. No configuration is necessary for the module to operate and remain in the approved mode.

After passing all pre-operational self-tests and cryptographic algorithm self-tests executed on start-up, the module automatically transitions to the approved mode.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
RSA SigVer (FIPS186-4)	A4201	Signature Type - PKCSPSS Modulo - 4096	FIPS 186-4
SHA2-384	A4201	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 6: Approved Algorithms

Vendor-Affirmed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

N/A for this module.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Digital Signature Verification	DigSig-SigVer	Verify a digital signature		RSA SigVer (FIPS186-4) Key size: 4096 bits SHA2-384
Message Digest	SHA	Compute a message digest		SHA2-384

Table 7: Security Function Implementations

2.7 Algorithm Specific Information

There is no algorithm specific information.

2.8 RBG and Entropy

N/A for this module.

N/A for this module.

2.9 Key Generation

The module does not implement any key generation methods.

2.10 Key Establishment

The module does not implement any automated key establishment methods.

2.11 Industry Protocols

The module does not implement any industry protocols.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
SRAM	Data Input	API input parameters for data
SRAM	Data Output	API output parameters for data
SRAM	Control Input	API function calls, API input parameters for control
SRAM	Status Output	API return codes, status values
Power port	Power	Power

Table 8: Ports and Interfaces

The logical interfaces are logically separated from each other by the API design. The power interface is physically separated from any other interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	None

Table 9: Roles

The module supports the Crypto Officer role only. This sole role is implicitly and always assumed by the operator of the module. No support is provided for multiple concurrent operators.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Digital Signature Verification	Verify a digital signature	1	Message, public key, signature	Pass/fail	Digital Signature Verification	Crypto Officer - RSA public key: W,E
Show Version	Return the module	None	None	Module version	None	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	version information					
Show Status	Return the module status	None	None	Module status	None	Crypto Officer
Self-test	Initiate on-demand self-tests by reset	None	None	Pass/fail	Digital Signature Verification Message Digest	Crypto Officer
Zeroization	Zeroize all SSPs	None	None	None	None	Crypto Officer - RSA public key: Z

Table 10: Approved Services

The approved service indicator can be retrieved using Microsoft HSTI and through the UEFI interactive shell tool. As the module only offers approved services, the indicator is always set when the module is operational. This is shown by the “FIPS mode: 1” output.

4.4 Non-Approved Services

N/A for this module.

4.5 External Software/Firmware Loaded

The module does not load external software or firmware.

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the firmware component of the module (“fips_module.bin”) is verified by comparing a SHA2-384 digest value calculated at run time with the SHA2-384 digest value stored in the module that was computed at build time.

5.2 Initiate on Demand

The integrity test is performed as part of the pre-operational self-tests, which are executed when the module is initialized. The integrity tests can be invoked on demand by powering off and subsequently re-initializing the module or SoC, which will perform (among others) the firmware integrity test.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Non-Modifiable

How Requirements are Satisfied:

The operational environment provides context separation for the memory and registers utilized by the module. When these components are used by the module, no other process or sub-component can access the information concurrently.

6.2 Configuration Settings and Restrictions

No configuration of the operational environment is required for the module to operate in an approved mode. Therefore, there are no rules, settings, or restrictions to the configuration of the operational environment.

7 Physical Security

7.1 Mechanisms and Actions Required

N/A for this module.

The embodiment of the module is a single chip consisting of production-grade components. The coating is a standard sealing coat applied over the single chip. The module provides no additional physical security techniques. No actions are required to maintain the physical security of the module.

8 Non-Invasive Security

This module does not implement any non-invasive security mechanism and therefore this section is not applicable.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
SRAM	Temporary storage for SSPs used by the module as part of service execution	Dynamic

Table 11: Storage Areas

SSPs are provided to the module by the calling process and are destroyed when released by the respective functions.

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
API input parameters	Operator residing on TOEPP	SRAM	Plaintext	Manual	Electronic	

Table 12: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Remove power from the SoC	De-allocates the volatile memory used to store SSPs	Volatile memory used by the module is overwritten within nanoseconds when power is removed	By removing power

Table 13: SSP Zeroization Methods

All data output is inhibited during zeroization.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
RSA public key	Public key used for RSA signature verification	4096 bits - 150 bits	Public key - PSP			Digital Signature Verification

Table 14: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
RSA public key	API input parameters	SRAM:Plaintext	For the duration of the service	Remove power from the SoC	

Table 15: SSP Table 2

9.5 Transitions

The RSA algorithm as implemented by the module conforms to FIPS 186-4, which has been superseded by FIPS 186-5. FIPS 186-4 was withdrawn on February 3, 2024.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
SHA2-384 (A4201)	N/A	Message digest	SW/FW Integrity	Module is operational	Performed on fips_module.bin

Table 16: Pre-Operational Self-Tests

The pre-operational firmware integrity test is performed automatically when the module is powered on before the module transitions into the operational state. While the module is executing the self-test, services are not available, and data output (via the data output interface) is inhibited until the tests are successfully completed. The module transitions to the operational state only after the pre-operational self-test passed successfully.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-384 (A4201)	32-bit message	KAT	CAST	Module is operational	Message digest	Module initialization (before integrity test)
RSA SigVer (FIPS186-4) (A4201)	PSS using 4096-bit key with SHA2-384	KAT	CAST	Module is operational	Signature verification	Module initialization (before integrity test)

Table 17: Conditional Self-Tests

The module performs self-tests on all approved cryptographic algorithms as part of the approved services supported in the approved mode of operation, using the tests shown in the table above. These self-tests are performed automatically before the firmware integrity test. Services are not available, and data output (via the data output interface) is inhibited during the self-tests. If any of these tests fails, the module transitions to the error state.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-384 (A4201)	Message digest	SW/FW Integrity	On demand	By resetting the module

Table 18: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-384 (A4201)	KAT	CAST	On demand	By resetting the module
RSA SigVer (FIPS186-4) (A4201)	KAT	CAST	On demand	By resetting the module

Table 19: Conditional Periodic Information

The module does not implement any periodic self-tests.

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error	Module immediately stops executing	SHA2-384 CAST error RSA CAST error Integrity test error	Resetting the module	Error code (AA0000FB, AA0000FC, or AA0000FD)

Table 20: Error States

In the error state, the output interface is inhibited, and the module accepts no more inputs or requests (as the module is no longer running). The error code is output through the FW status register, which explains the error that has occurred.

10.5 Operator Initiation of Self-Tests

All self-tests can be invoked on demand by unloading and subsequently re-initializing the module.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The procedures herein described are directed at OEMs for producing and configuring their BIOS so that the FIPS module is properly enabled to operate as the validated module in conformance with the rules in this Security Policy document.

Once properly installed and enabled, no configuration is necessary for the module to operate and remain in the approved mode, as it is the only mode of operation of the module.

To enable the FIPS capability

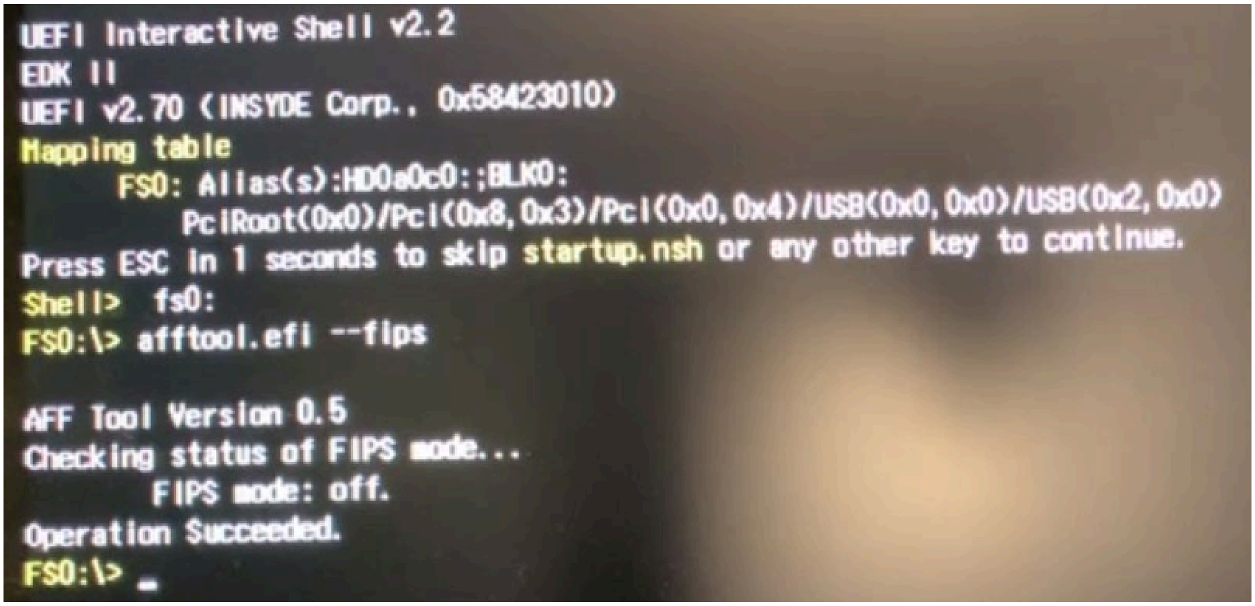
1. Reserve 16KiB at least for AMD Secure Processor level 1 directory, as the FIPS module requires additional 8KiB of ROM space for the AMD Secure Processor L1 Bootloader.

2. The Platform BIOS must include the file with "_FIPS" postfix in the file name as AMD Secure Processor entry 0x1. For example, the file PspBootLoader_stage1_prod_AB_RN_FIPS.sbin has "_FIPS" postfix in the file name. This file is thus a FIPS capable AMD Secure Processor boot loader. Conversely, the file PspBootLoader_stage1_prod_AB_RN.sbin does not have "_FIPS" postfix in the file name, making this file a non-FIPS capable AMD Secure Processor boot loader.
3. Set BIT 32 of AMD Secure Processor soft fuse chain (AMD Secure Processor entry 0xB) to enable FIPS capability.
    a. The BIT32 in AMD Secure Processor entry 0xB is defined as FIPS capability enablement. If 0, the FIPS capability is OFF; if 1, the FIPS capability is ON (i.e., the module is properly installed as the validated module described in this document).

To verify whether FIPS capability is on

1. Boot the system into UEFI shell with secure boot disabled.
2. Use the UEFI shell version of the AFF Tool version 0.3 and beyond. This tool is provided by the vendor. Run the AFF Tool with the command: afftool --fips from the interactive UEFI shell provided by the BIOS.
    a. If it shows "FIPS mode: on", this is the FIPS capable module installed.
    b. If it shows "FIPS mode: off", the module (described in this document) is not installed.

The screenshot in Figure 3 shows the usage of the AFF Tool. The output indicates that the FIPS module is not installed. In this condition, the module does not operate in conformance with this Security Policy document.

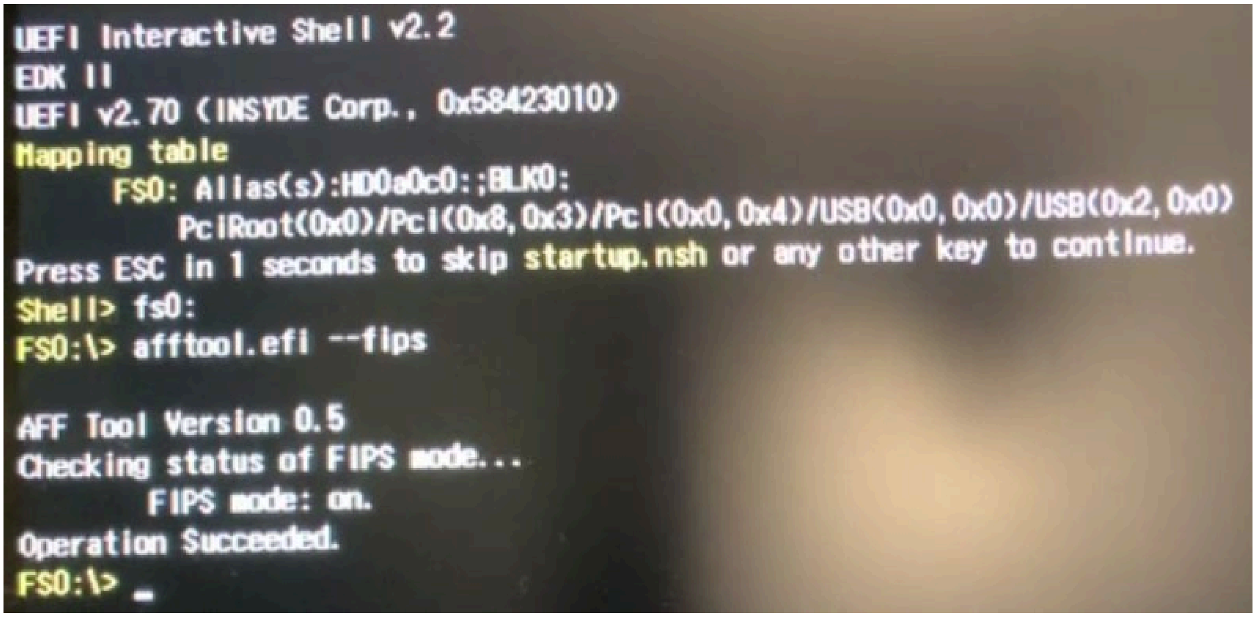


```
UEFI Interactive Shell v2.2
EDK II
UEFI v2.70 (INSYDE Corp., 0x58423010)
Mapping table
      FS0: Alias(s):HD0a0c0::BLK0:
          PciRoot(0x0)/Pci(0x8,0x3)/Pci(0x0,0x4)/USB(0x0,0x0)/USB(0x2,0x0)
Press ESC in 1 seconds to skip startup.nsh or any other key to continue.
Shell> fs0:
FS0:\> afftool.efi --fips

AFF Tool Version 0.5
Checking status of FIPS mode...
        FIPS mode: off.
Operation Succeeded.
FS0:\> _
```

Figure 3: AFF Tool indicates that the module is not installed.

The screenshot in Figure 4 again shows the usage of the AFF Tool. The output demonstrates that the FIPS module is installed and thus will operate as the FIPS validated module according to the rules in this Security Policy document.



```
UEFI Interactive Shell v2.2
EDK II
UEFI v2.70 (INSYDE Corp., 0x58423010)
Mapping table
      FS0: Alias(s):HD0a0c0::BLK0:
          PciRoot(0x0)/Pci(0x8,0x3)/Pci(0x0,0x4)/USB(0x0,0x0)/USB(0x2,0x0)
Press ESC in 1 seconds to skip startup.nsh or any other key to continue.
Shell> fs0:
FS0:\> afftool.efi --fips

AFF Tool Version 0.5
Checking status of FIPS mode...
        FIPS mode: on.
Operation Succeeded.
FS0:\> _
```

Figure 4: AFF Tool indicating that the module is installed.

11.2 Administrator Guidance

All the functions, ports and logical interfaces described in this document are available to the Crypto Officer. The module only provides approved functions, and as such there are no special procedures to administer the approved mode of operation.

11.3 Non-Administrator Guidance

The module implements only the Crypto Officer. There are no requirements for non-administrator operators.

11.4 Design and Rules

The bootloader (which acts as the operator of the module) initializes the fips_module.bin component by loading it into memory upon power-on. After the pre-operational self-tests are successfully concluded, the module automatically transitions to the operational state. In the operational state, the module automatically performs the signature verification of the key database using the RSA signature verification service, which is the sole service provided by the module. The key database, RSA public key, and signature are provided as input by the operator of the module (the bootloader). After the successful signature verification of the key database, the module unloads itself from memory, ceasing its operation.

All the procedures described above are conducted without any human assistance. To perform the procedures again, the module must be reset, which will trigger a new boot.

11.5 Maintenance Requirements

There are no maintenance requirements.

11.6 End of Life

The process for performing "End of Life" occurs at the chronological point of 10 years starting from manufacturing date of the module.

As stated previously, the module does not possess persistent storage of SSPs. The SSP values only exists in volatile memory and those values vanish when the module is powered off. The procedure for secure sanitization of the module at the end of life is simply to power it off, which is the action of zeroization of the SSPs. As a result of this sanitization via power-off, the SSPs are removed from the module, so that the module may either be distributed to other operators or disposed.

12 Mitigation of Other Attacks

The module does not offer mitigation of other attacks and therefore this section is not applicable.