Microsoft Corporation

BitLocker Dump Filter

FIPS 140-3 Non-Proprietary Security Policy Document

Microsoft Windows 11 version 22H2
(Pro, Enterprise, IoT Enterprise, Education, and Home Editions)

Microsoft Windows Server 2022
(Standard and Datacenter Editions)

Prepared By	Microsoft Corporation One Microsoft Way Redmond, WA 98052-6399
Document Version Number	1.0
Updated On	May 4, 2026

COPYRIGHT AND DISCLAIMER

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The example companies, organizations, products, people and events depicted herein are fictitious. No association with any real company, organization, product, person or event is intended or should be inferred.

Microsoft, Active Directory, Azure, Visual Basic, Visual Studio, Windows, the Windows logo, Windows NT, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

BitLocker Drive Encryption is a data protection feature of the Windows operating system that encrypts data on a storage volume. The BitLocker Dump Filter module (the “module”) is a multi-chip standalone software-hybrid cryptographic module that protects crash dump files on BitLocker-encrypted computers. The module implements approved cryptographic algorithms.

This FIPS 140-3 Security Policy contains a specification of the rules under which the module must operate and describes how the module meets the requirements specified in Federal Information Processing Standards Publication 140-3 (FIPS PUB 140-3) and International Standard ISO/IEC 19790:2012 (Information technology – Security techniques – Security requirements for cryptographic modules). This document is intended for the FIPS 140-3 testing lab, the Cryptographic Module Validation Program (CMVP), and administrators and users of the module.

1.2 Security Levels

The overall security rating for the module is level 1. The table below lists the security levels of individual clauses for this validation.

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	1
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	1
	Overall Level	1

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The BitLocker Dump Filter module is a multi-chip standalone software-hybrid cryptographic module that protects crash dump files on BitLocker-encrypted computers. The module is part of the system dump stack. When the dump stack is called during a crash, the module ensures that all data is encrypted before being written to storage as a dump file.

Other aspects of BitLocker and related cryptographic modules are described in the Security Policy documents for the Boot Manager and Windows OS Loader modules

Module Type: Software-hybrid

Module Embodiment: MultiChipStand

Cryptographic Boundary:

The software-hybrid cryptographic boundary for BitLocker Dump Filter consists of disjoint software and hardware components within the same physical perimeter of the host platform. The module's software component is the binary listed in the following table, and its hardware component is the CPU running on the host platform.

Software Component	Description
DUMPFVE.SYS	Binary file that contains the module.

Table 2: Module Software Components

Tested Operational Environment's Physical Perimeter (TOEPP):

The Tested Operational Environment's Physical Perimeter (TOEPP) is the physical perimeter of the computer that contains the module.

The following block diagram illustrates the module's components, physical perimeter (TOEPP), and cryptographic boundary. The cryptographic boundary of the module is the module software component, DUMPFVE.SYS. The DUMPFVE.SYS binary is loaded from the OS volume in physical storage and executes in the computer memory. The control input, data input / output, and status output of the module exist within the computer memory as well.

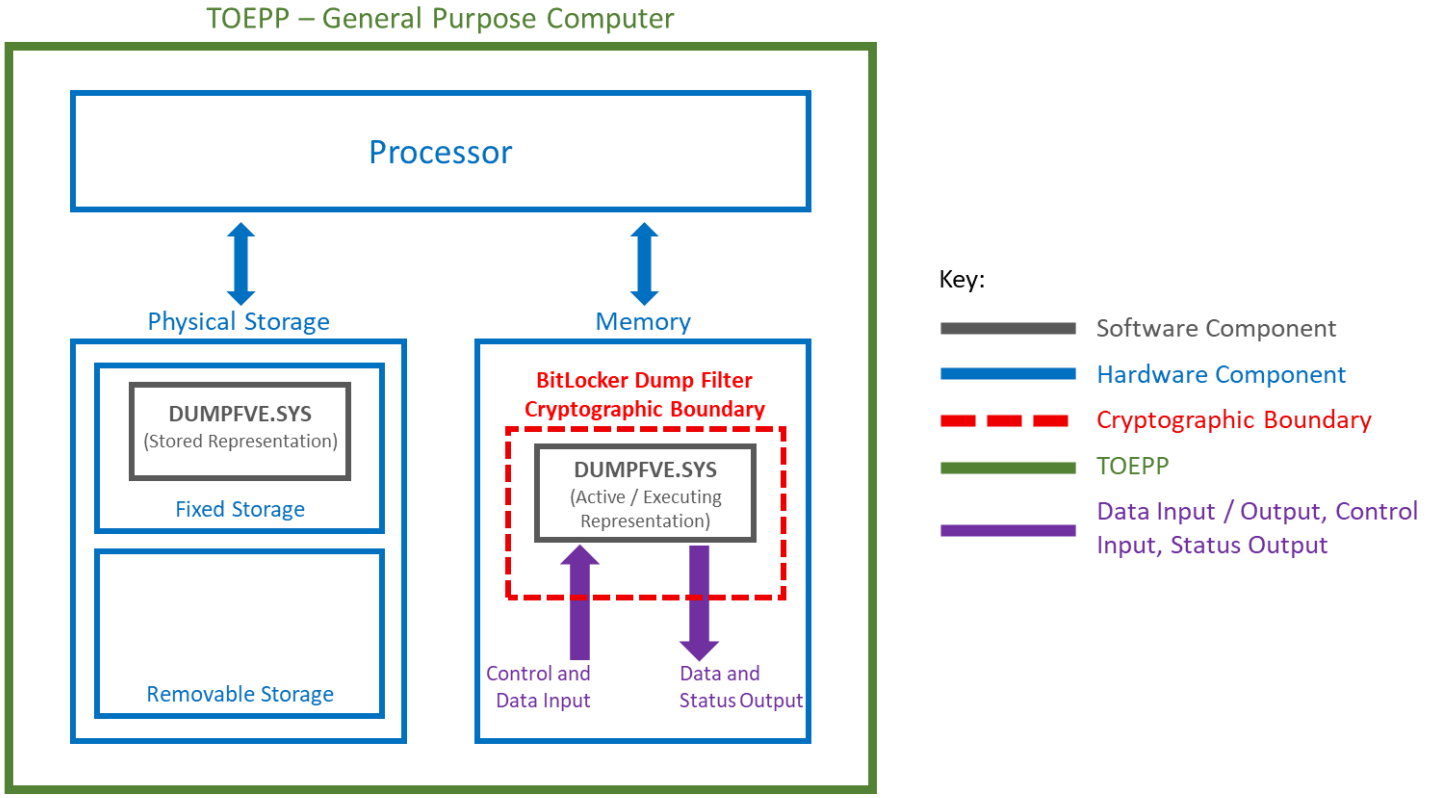
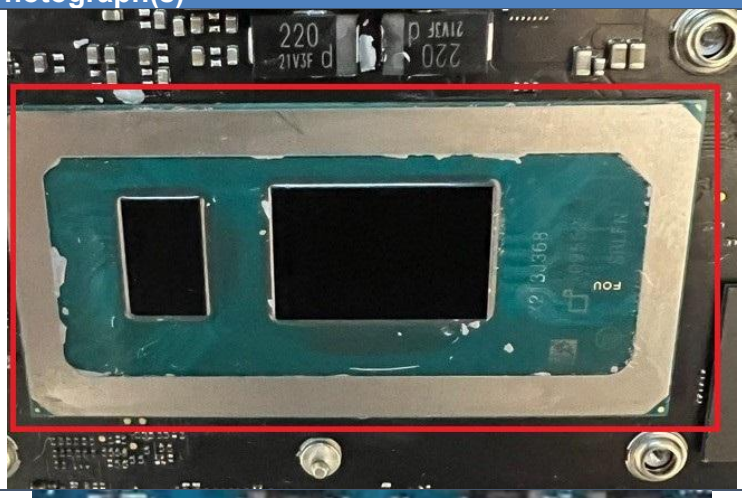
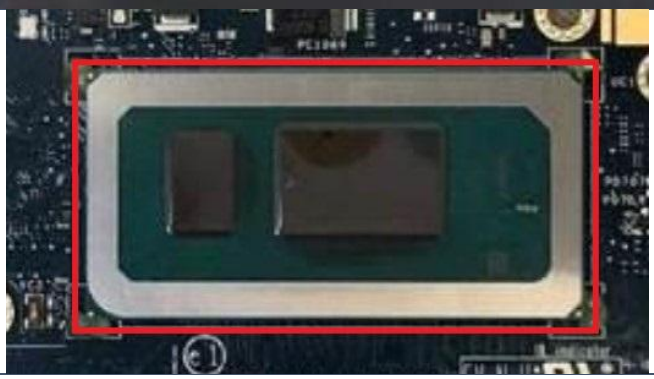


Figure 1: Module Boundary Diagram

The following table includes a photograph of the CPU of each computer listed in section 2.2 Tested and Vendor Affirmed Module Version and Identification. The processor is highlighted by a red box for clarity. For laptop devices, the processor is shown as integrated into the motherboard. For server devices, the processor is shown both independently and as installed in the computer with its integral heat sink.

CPU	Photograph(s)
12th Gen Intel Core i7-1265U (Microsoft Surface Laptop 5)	
11th Gen Intel Core i5-11500H (HP ZBook Power G8)	
11th Gen Intel Core i7-1185G7 (Dell Latitude 7420)	

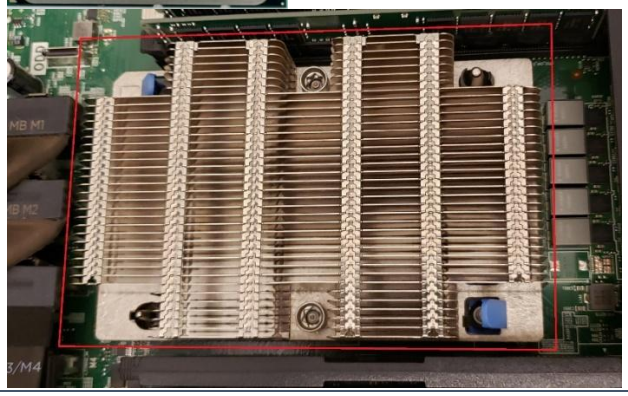
CPU	Photograph(s)
Intel Xeon Gold 6130 (Dell PowerEdge R640)	 

Table 3: CPU Photographs

2.2 Tested and Vendor Affirmed Module Version and Identification

This validation includes the following Windows products and versions, each of which can be identified by its build number. The cryptographic module is a distinct implementation in each product build and for each processor architecture. Some Windows products may be installed as different editions; however, the cryptographic module is the same implementation in different editions of the same product.

Windows Product	Build	Edition(s) in Scope
Windows 11 version 22H2	10.0.22621.1	Enterprise Edition Home Edition IoT Enterprise Edition Pro Edition Education Edition
Windows Server 2022	10.0.20348.1668 (including the March 2023 updates)	Standard Edition Datacenter Edition

Table 4: Version Information

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
DUMPFVE.SYS (Windows 11 version 22H2)	Windows 11 version 22H2, build 10.0.22621.1	N/A	Yes
DUMPFVE.SYS (Windows Server 2022)	Windows Server 2022, build 10.0.20348.1668	N/A	Yes

Table 5: 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
Dell Latitude 7420	11th Gen Intel Core i7-1185G7	N/A	11th Gen Intel Core i7-1185G7	N/A
Dell PowerEdge R640	Intel Xeon Gold 6130	N/A	Intel Xeon Gold 6130	N/A
HP ZBook Power G8	11th Gen Intel Core i5-11500H	N/A	11th Gen Intel Core i5-11500H	N/A
Microsoft Surface Laptop 5	12th Gen Intel Core i7-1265U	N/A	12th Gen Intel Core i7-1265U	N/A

Table 6: Tested Module Identification – Hybrid Disjoint Hardware

Tested Operational Environments - Software, Firmware, Hybrid:

The operational environment for the module is the Windows operating system running on a supported hardware platform, as listed in the table below. All hardware platforms in the table below are 64-bit Intel architecture. The tested operational environments provide Processor Algorithm Acceleration (PAA) in the form of the Advanced Encryption Standard New Instructions (AES-NI).

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Windows 11 version 22H2, Education Edition	Dell Latitude 7420	11th Gen Intel i7-1185G7	Yes	N/A	Windows 11 version 22H2, build 10.0.22621.1
Windows 11 version 22H2, Enterprise Edition	Microsoft Surface Laptop 5	12th Gen Intel Core i7-1265U	Yes	N/A	Windows 11 version 22H2, build 10.0.22621.1
Windows 11 version 22H2, Home Edition	Microsoft Surface Laptop 5	12th Gen Intel Core i7-1265U	Yes	N/A	Windows 11 version 22H2, build 10.0.22621.1
Windows 11 version 22H2, IoT Enterprise Edition	Microsoft Surface Laptop 5	12th Gen Intel Core i7-1265U	Yes	N/A	Windows 11 version 22H2, build 10.0.22621.1
Windows 11 version 22H2, Pro Edition	HP ZBook Power G8	11th Gen Intel i5-11500H	Yes	N/A	Windows 11 version 22H2, build 10.0.22621.1
Windows Server 2022 Datacenter, including the March 2023 Updates	Dell PowerEdge R640	Intel Xeon Gold 6130	Yes	N/A	Windows Server 2022, build 10.0.20348.1668
Windows Server 2022 Standard, including the March 2023 Updates	Dell PowerEdge R640	Intel Xeon Gold 6130	Yes	N/A	Windows Server 2022, build 10.0.20348.1668

Table 7: Tested Operational Environments - Software, Firmware, Hybrid

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

Operating System	Hardware Platform
Any Microsoft operating system which is a relabeled version of the operating systems listed in Section 2.2.	Any UEFI-based x64 computer.

Table 8: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid

The CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

2.3 Excluded Components

No components within the cryptographic boundary are claimed as excluded.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	Normal operation of the computer, Windows OS, and module. This is the only mode claimed by the module.	Approved	N/A

Table 9: Modes List and Description

2.5 Algorithms

Approved Algorithms:

The tables below list the approved algorithms used in the module. The module may not use some of the capabilities described in each CAVP certificate. As the module has separate CAVP certificates for Windows 11 and Windows Server 2022, each approved algorithm is listed twice, with CAVP #A4008 for Windows 11 and CAVP #A4009 for Windows Server 2022. See Section 13 Standards References for links to the standards referenced in the tables below.

The BitLocker Dump Filter module relies on functionality that is implemented in the Code Integrity and Secure Kernel Code Integrity cryptographic modules. All modules are installed together as described in section 11 Life-Cycle Assurance. FIPS 140-3 deems the BitLocker Dump Filter module as binding to the Code Integrity or Secure Kernel Code Integrity module (certificates #5406 and #5407), depending on whether or not Memory Integrity is enabled on the host platform. These module dependencies are described as Existing Validated Modules (EVMs) in this document. See section 5.1 Integrity Techniques for more information on the dependencies between Windows modules.

Table 10 below lists the approved algorithms in the BitLocker Dump Filter module. Table 11 below lists the approved cryptographic algorithms in the Code Integrity and Secure Kernel Code Integrity modules that are used by the BitLocker Dump Filter module.

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A4008	Direction - Encrypt Key Length - 128, 256	SP 800-38A
AES-CBC	A4009	Direction - Encrypt Key Length - 128, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A4008	Direction - Encrypt Key Length - 128, 256	SP 800-38E
AES-XTS Testing Revision 2.0	A4009	Direction - Encrypt Key Length - 128, 256	SP 800-38E

Table 10: Approved Algorithms -

EVM

Algorithm	CAVP Cert	Properties	Reference
RSA SigVer (FIPS186-4)	A4008	Signature Type - PKCS 1.5 Modulo - 2048	FIPS 186-4
RSA SigVer (FIPS186-4)	A4009	Signature Type - PKCS 1.5 Modulo - 2048	FIPS 186-4
SHA2-256	A4008	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A4009	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 11: Approved Algorithms - EVM

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

The table below lists the module's Security Function Implementations. The Algorithms column presents algorithm and key size information for each CAVP certificate listed in Section 2.5 Algorithms. Blank cells are either not applicable or optional according to the CMVP.

Name	Type	Description	Properties	Algorithms
BC1	BC-UnAuth	Symmetric block cipher encryption used by the Write Encrypted Crash Dump Data service to write the crash dump file to an encrypted volume.		AES-CBC: (A4008, A4009) Key Length: 128 and 256 bits AES-XTS Testing Revision 2.0: (A4008, A4009) Key Length: 128 and 256
DigSig1	DigSig-SigVer	RSA signature verification used for the module pre-operational software integrity check only, executed by either the [EVM] Code Integrity or [EVM] Secure Kernel Code Integrity module (IG 1.A Documentation Requirements 5). SHA secure hash functions support this by verifying the RSA signature.		RSA SigVer (FIPS186-4): (A4008, A4009) Modulus Size: 2048 bits SHA2-256: (A4008, A4009) SHA Size: 256 bits
SHS1	SHA	Secure hash function used for the module pre-operational software integrity check only, executed by either the [EVM] Code Integrity or [EVM] Secure Kernel Code Integrity module (IG 1.A Documentation Requirements 5).		SHA2-256: (A4008, A4009) SHA Size: 256 bits

Table 12: Security Function Implementations

2.7 Algorithm Specific Information

2.7.1 FIPS 186-4 and 186-5

The module claims compliance with FIPS 186-5 for RSA signature verification, although the CAVP testing for RSA was completed against FIPS 186-4. Because the FIPS 186-4 RSA CAVP tests are mathematically identical to the FIPS 186-5 RSA CAVP tests, the module can claim FIPS 186-5 compliance for these tests. The scope of FIPS 186-4 testing complies with IG C.K. Additional Comment 3. Although FIPS 186-4 has been superseded by FIPS 186-5, algorithms implemented under FIPS 186-4 remain approved for use under NIST SP 800-131A Rev. 2.

2.7.2 AES-XTS

AES-XTS is approved only for storage applications such as BitLocker. The length of data encrypted does not exceed 2^{20} AES blocks. Key 1 and Key 2 are generated independently, as required by IG C.I., and the two keys are explicitly checked to ensure that they are not equal before use.

2.8 RBG and Entropy

N/A for this module.

N/A for this module.

2.9 Key Generation

N/A because the module does not generate cryptographic keys.

2.10 Key Establishment

N/A because the module does not perform key establishment.

2.11 Industry Protocols

N/A as none are claimed.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

As a software-hybrid module, the module has no physical ports of its own. The physical ports of the module are interpreted as those on the underlying hardware platform and control of them is outside the scope of the module.

Physical Port	Logical Interface(s)	Data That Passes
N/A	Data Input	The Data Input Interface includes the GetFveContext function and DumpWrite function. GetFveContext is responsible for reading the Full Volume Encryption Key (FVEK). DumpWrite accepts the memory blocks to encrypt with the FVEK and the target disk locations for the blocks as input.
N/A	Data Output	The Data Output Interface is the data returned from the DumpWrite function. This function is responsible for providing the encrypted content for the crash dump file. Data exits the module in the form of encrypted blocks that may be written to a crash dump file on an encrypted volume.
N/A	Control Input	The BitLocker Dump Filter module's control input interface consists of parameter interfaces for the GetFveContext and DumpWrite functions. These interfaces are not exported, but rather, are internal to the cryptographic module.
N/A	Status Output	The BitLocker Dump Filter status output is a return value of type NTSTATUS that indicates whether the function completed successfully or not. The BitLocker Dump Filter has no status output interface for self-test errors. If the self-tests pass, the module is loaded. If not, the module unloads itself.
N/A	Power	N/A

Table 13: Ports and Interfaces

3.4 Additional Information

3.4.1 Module Function Details

The table below provides additional details on the module functions, GetFveContext and DumpWrite. These interfaces are not exported and are internal to the module.

Function	Description
GetFveContext	This function gets the BitLocker Full Volume Encryption Key (FVEK) for the storage volume. The Context parameter supplies the dump stack filter context. The FveContext parameter supplies the internal BitLocker context, which includes the BitLocker status and FVEK in this context, so it can be used later when writing data to the volume.
DumpWrite	This function uses the FVEK from the Context parameter that is provided by the GetFveContext interface. The DiskByteOffset parameter is used to specify the location on the volume to receive the encrypted output data. The Mdl parameter points to the input data to be encrypted.

Table 14: Module Function Details

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

4.2 Roles

The module claims a single role, Cryptographic Officer (CO). All services are accessible by this role. BitLocker Dump Filter is a kernel-mode driver that does not interact with the user through any service. The module's functions are fully automatic and not configurable.

Name	Type	Operator Type	Authentication Methods
Cryptographic Officer (CO)	Role	CO	None

Table 15: Roles

4.3 Approved Services

The module provides approved services only. The indicator for each service is the successful completion of the service. The table below provides additional indicator details to enable the operator to verify each service's successful completion.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Perform Cryptographic Algorithm Self-Tests	The module provides a power-up self-test service that is automatically executed.	Self-test success is indicated by module and algorithm availability; failure is indicated by the Driver Load Failure error and the module unloading itself.	Automatically executed when the module is loaded into memory.	Implicit in module availability: if the module is available, the self-tests have passed.	BC1	Cryptographic Officer (CO)
Perform Pre-Operational Software Integrity Test [EVM]	The pre-operational integrity test is executed by the [EVM] Code Integrity or [EVM] Secure Kernel Code Integrity module (IG 1.A Documentation Requirements 5).	Integrity test success is indicated by the BitLocker Dump Filter module being loaded into memory.	This service is fully automatic and executed before the BitLocker Dump Filter module is loaded into memory.	The BitLocker Dump Filter module is loaded into memory.	DigSig1 SHS1	Cryptographic Officer (CO)
Perform Zeroization	Zeroizes cryptographic material.	SSPs are zeroized. See Section 9.3 SSP Zeroization Methods for more information.	This service is fully automatic.	Keys are zeroized.	BC1 DigSig1 SHS1	Cryptographic Officer (CO) - Embedded X.509 Certificate for DUMPFVE.SYS (This is not an SSP): Z - Full Volume Encryption Key (FVEK): Z - Hash Value for DUMPFVE.SYS (This is not an SSP): Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Show Status	The module provides a show status service that is automatically executed by the module to provide the status response of the module either via output to the computer monitor or to log files.	A return value of type NTSTATUS that indicates whether the function completed successfully or not.	Automatically executed by the module after each operation.	Status is output across the module's Status Output logical interface and on to the computer monitor or to log files.	None	Cryptographic Officer (CO)
Show Version	Provides the module version number.	Version information is provided in the Portable Executable (PE) header of each module binary. The PE header contains Windows-specific fields such as the major and minor version, which equate to the module version. See the PE Format public documentation more information.	Module binary file.	Version information is embedded in the PE header of the binary.	None	Cryptographic Officer (CO)
Write Encrypted Crash Dump Data	This service is executed when the system crashes and must write the crash dump file to an encrypted volume.	Functions related to the service return a status after execution.	Automatically executed when the system crashes and must write the crash dump file to an encrypted volume.	The cryptographic operation is completed and status is returned.	BC1	Cryptographic Officer (CO) - Full Volume Encryption Key (FVEK): W,E

Table 16: Approved Services

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

The secure installation, generation, and startup procedures of this module are part of the secure installation, configuration, and startup procedures of the Windows operating systems named in Section 2.2 Tested and Vendor Affirmed Module Version and Identification.

In the Windows operating system, all kernel-mode modules, including DUMPFVE.SYS, are loaded into the Windows Kernel (ntoskrnl.exe), which executes as a single process. The Windows operating system environment enforces process isolation from user-mode processes including memory and processor scheduling between the kernel and user-mode processes.

5.1 Integrity Techniques

Windows uses several mechanisms to provide integrity verification depending on the stage in the boot sequence and also on the hardware and configuration. The algorithms used for integrity verification are included in section 2.5 Algorithms, Table 11.

Either the [EVM] Code Integrity module or [EVM] Secure Kernel Code Integrity module (IG 1.A Documentation Requirements 5) checks the integrity of the DUMPFVE.SYS component of the BitLocker Dump Filter module before it is loaded. The EVM uses approved RSA SigVer (FIPS 186-4, CAVP certificates A4008 and A4009) and SHA2-256 (FIPS 180-4, CAVP certificates A4008 and A4009) to perform the integrity check. See the table below and the Security Policy documents for Code Integrity and Secure Kernel Code Integrity modules for more information on the EVMs. Windows binaries include a SHA2-256 hash of the binary signed with the 2048-bit Microsoft RSA code-signing key (the key associated with the Microsoft code-signing certificate). The integrity check uses the public key component of the Microsoft code signing certificate to verify the signed hash of the binary.

Whether the Code Integrity module or the Secure Kernel Code Integrity module performs the integrity check depends on the following:

- If Memory Integrity (also known as Hypervisor Code Integrity, HVCI) is not enabled, then the Code Integrity module performs the integrity check.
- If Memory Integrity is enabled, then the Secure Kernel Code Integrity module performs the integrity check.

Core Isolation and Memory Integrity are enabled by default when Windows runs on a Secured-core PC or a Secured-core server.

EVM Name	CMVP Certificate	Version Details
Code Integrity (Windows 11 v 22H2 and Windows Server 2022)	#5406	Windows 11 version 22H2, build 10.0.22621.1 Windows Server 2022, build 10.0.20348.1668
Secure Kernel Code Integrity (Windows 11 v 22H2 and Windows Server 2022)	#5407	Windows 11 version 22H2, build 10.0.22621.1 Windows Server 2022, build 10.0.20348.1668

Table 17: EVM Details

The figure below shows the Integrity Chain of trust for the Windows builds and modules in scope for this validation.

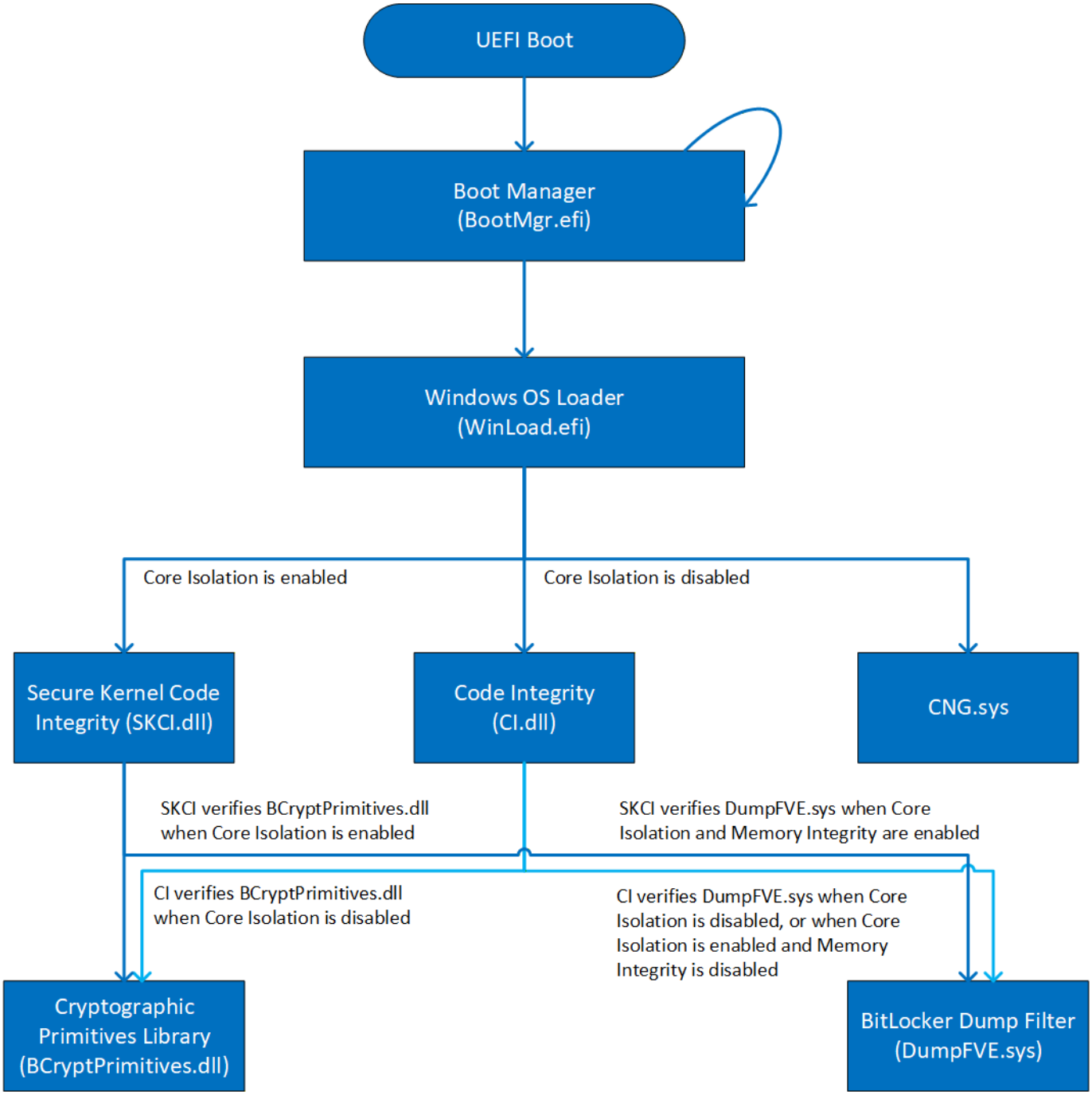


Figure 2: Integrity Chain of Trust

5.2 Initiate on Demand

To initiate the integrity test on demand, the operator may restart the computer.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied:

The modifiable operational environment for the module is the Windows operating system running on a supported hardware platform, as listed in Section 2.2 Tested and Vendor Affirmed Module Version and Identification. The BitLocker Dump Filter is loaded into kernel memory as part of the boot process before the logon component is initialized.

7 Physical Security

7.1 Mechanisms and Actions Required

BitLocker Dump Filter is a multi-chip standalone software-hybrid module whose host platforms meet the Level 1 physical security requirements. The host platform consists of production-grade physical security components that include standard passivation techniques and is entirely contained within a metal or hard plastic production-grade enclosure that may include doors or removable covers.

Tables identifying the voltage and temperature boundaries that trigger zeroization or shutdown are not included in this Security Policy as they are N/A for a Level 1 validation of a hybrid module.

8 Non-Invasive Security

N/A for this module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
Hard Disk	Persistent SSPs are stored on the operating system volume (see: Hard Disk in the block diagram).	Static
RAM	Volatile SSPs are temporarily stored in the computer's memory while the module is powered-on (see: RAM in the block diagram).	Dynamic

Table 18: Storage Areas

9.2 SSP Input-Output Methods

The table below lists the input method for the single SSP input into the module (the Full Volume Encryption Key). The FVEK is input from the Windows Kernel, which received the FVEK from the Windows OS Loader module.

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Resides in RAM as plaintext	Imported from the Windows Kernel RAM space	RAM	Plaintext	Manual	Electronic	

Table 19: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Procedural zeroization	Operators may choose to overwrite SSPs in non-volatile storage by reformatting and overwriting the storage media with zeroes for the operating system. Operators may zeroize SSPs in volatile storage by powering off the host General Purpose Computer (GPC) as the SSPs in volatile storage are effectively overwritten with zeros when power is lost.	SSPs are non-recoverable after being overwritten with zeroes.	Operators may use the Format command together with the /P parameter which specifies the number of overwrite passes. See the public documentation for the Format command for more information. Operators may power off or reboot the host GPC to zeroize SSPs stored in volatile RAM.
Zeroization after module is unloaded.	The module overwrites the temporary storage area for the FVEK SSP with zeroes when the module is unloaded from memory as part of shutting down Windows.	SSPs are non-recoverable after being overwritten with zeroes.	None (programmatically executed by the module as part of Windows shutdown).

Table 20: SSP Zeroization Methods

9.4 SSPs

The tables below list the keys and SSPs used by the module. Per the CMVP, public keys and file hashes used for the module integrity check are not considered SSPs and, as such, have no input method assigned and are categorized as neither PSP nor CSP. Blank cells are either not applicable or optional according to the CMVP.

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Embedded X.509 Certificate for DUMPFVE.SYS (This is not an SSP)	Public key used for RSA PKCS #1 (v1.5) integrity verification of DUMPFVE.SYS.	Size: 2048 bits - Strength: 112 bits	Asymmetric Public Key (RSA) - Neither	Generated external to the module by the Microsoft Windows build process.		DigSig1
Full Volume Encryption Key (FVEK)	AES key used to encrypt dump files on disk.	Size: 128 bits or 256 bits (administrator configurable) - Strength: 128 bits or 256 bits	Symmetric Key (AES) - CSP	Externally generated by BitLocker runtime components outside of the BitLocker Dump Filter cryptographic boundary.		BC1
Hash Value for DUMPFVE.SYS (This is not an SSP)	File hash used to verify the integrity of DUMPFVE.SYS.	Size: 256 bits - Strength: 128 bits	File Hash - Neither	Generated external to the module by the Microsoft Windows build process.		SHS1

Table 21: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Embedded X.509 Certificate for DUMPFVE.SYS (This is not an SSP)		Hard Disk:Plaintext		Procedural zeroization	
Full Volume Encryption Key (FVEK)	Resides in RAM as plaintext	RAM:Plaintext	Until zeroized	Zeroization after module is unloaded.	
Hash Value for DUMPFVE.SYS (This is not an SSP)		Hard Disk:Plaintext		Procedural zeroization	

Table 22: SSP Table 2

9.5 Transitions

The following transition timeline applies to the approved algorithm named below:

- FIPS 186-4 has been superseded by FIPS 186-5. This transition began on July 25, 2023, and concluded on February 3, 2024. Although testing for this module was completed against FIPS 186-4, it claims compliance with FIPS 186-5 for RSA signature verification – see Section 2.7.1 FIPS 186-4 and 186-5 for more information.

10 Self-Tests

Windows performs tests automatically to ensure integrity and correct functionality. The module will not perform cryptographic functions while in its self-test or error states. If a self-test fails, the module enters an error state. If the self-test passes, cryptographic functions are available for use. As the module has separate CAVP certificates for Windows 11 and Windows Server 2022, the self-test tables below list two identical rows for each algorithm self-test, one for Windows 11 and one for Windows Server 2022.

10.1 Pre-Operational Self-Tests

As described above, either the [EVM] Code Integrity module or the [EVM] Secure Kernel Code Integrity module checks the integrity of DUMPFVE.SYS before it is loaded. The algorithms used for the pre-operational self-tests pass their own algorithm self-tests before integrity verification is performed. See section 5.1 Integrity Techniques for details on the EVM and how the module's integrity is checked.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
RSA SigVer (FIPS186-4) (A4008)	RSA PKCS#1v1.5 with 2048-bit key and SHA2-256 (Windows 11 version 22H2)	Software Integrity	SW/FW Integrity	The Perform Cryptographic Algorithm Self-Tests service runs.	[EVM] signature verification.
RSA SigVer (FIPS186-4) (A4009)	RSA PKCS#1v1.5 with 2048-bit key and SHA2-256 (Windows Server 2022)	Software Integrity	SW/FW Integrity	The Perform Cryptographic Algorithm Self-Tests service runs.	[EVM] signature verification.

Table 23: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

The module performs the conditional cryptographic algorithm self-tests automatically when the module is loaded into memory, after the pre-operational software integrity tests described above have completed. The module implements Known Answer Test (KAT) functions each time the module is loaded by the Windows kernel. The BitLocker Dump Filter has no status output interface for self-test errors. If the self-tests pass, the module is loaded.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
[EVM] RSA SigVer (FIPS186-4) (A4008)	RSA PKCS#1v1.5 with 2048-bit key and SHA2-256 (Windows 11 version 22H2)	KAT	CAST	The module's integrity verification is performed.	[EVM] signature verification.	Runs before module's integrity verification is performed.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
[EVM] RSA SigVer (FIPS186-4) (A4009)	RSA PKCS#1v1.5 with 2048-bit key and SHA2-256 (Windows Server 2022)	KAT	CAST	The module's integrity verification is performed.	[EVM] signature verification.	Runs before module's integrity verification is performed.
[EVM] SHA2-256 (A4008)	256 bit hash (Windows 11 version 22H2)	KAT	CAST	The module's integrity verification is performed.	[EVM] Secure Hash	Runs before module's integrity verification is performed.
[EVM] SHA2-256 (A4009)	256 bit hash (Windows Server 2022)	KAT	CAST	The module's integrity verification is performed.	[EVM] Secure Hash	Runs before module's integrity verification is performed.
AES-CBC (A4008) - Decrypt	AES-CBC with 128- or 256-bit key (Windows 11 22H2)	KAT	CAST	BitLocker Dump Filter is initialized.	Decrypt KAT	Run at every module initialization after the module integrity is verified.
AES-CBC (A4008) - Encrypt	AES-CBC with 128- or 256-bit key (Windows 11 22H2)	KAT	CAST	BitLocker Dump Filter is initialized.	Encrypt KAT.	Run at every module initialization after the module integrity is verified.
AES-CBC (A4009) - Decrypt	AES-CBC with 128- or 256-bit key (Windows Server 2022)	KAT	CAST	BitLocker Dump Filter is initialized.	Decrypt KAT	Run at every module initialization after the module integrity is verified.
AES-CBC (A4009) - Encrypt	AES-CBC with 128- or 256-bit key (Windows Server 2022)	KAT	CAST	BitLocker Dump Filter is initialized.	Encrypt KAT.	Run at every module initialization after the module integrity is verified.
AES-XTS Testing Revision 2.0 (A4008) - Encrypt	AES-XTS with 128- or 256-bit key (Windows 11 22H2)	KAT	CAST	BitLocker Dump Filter is initialized.	Encrypt KAT.	Run at every module initialization after the module integrity is verified.
AES-XTS Testing Revision 2.0 (A4008) - Key Equivalence	AES-XTS with 128- or 256-bit key (Windows 11 22H2)	Fault-detection	CAST	Write Encrypted Crash Dump Data service runs.	Key equivalence test in compliance with FIPS 140-3 IG C.I.	Runs before AES-XTS algorithm is used.
AES-XTS Testing Revision 2.0 (A4009) - Encrypt	AES-XTS with 128- or 256-bit key (Windows Server 2022)	KAT	CAST	BitLocker Dump Filter is initialized.	Encrypt KAT.	Run at every module initialization after the module integrity is verified.
AES-XTS Testing Revision 2.0 (A4009) - Key Equivalence	AES-XTS with 128- or 256-bit key (Windows Server 2022)	Fault-detection	CAST	Write Encrypted Crash Dump Data service runs	Key equivalence test in compliance with FIPS 140-3 IG C.I.	Runs before AES-XTS algorithm is used.

Table 24: Conditional Self-Tests

10.3 Periodic Self-Test Information

The tables below present the periodic self-test information for the module. The set of self-tests presented in tables 25 and 26 below is identical to the set of self-tests presented in tables 23 and 24 above. Either the [EVM] Code Integrity module or the [EVM] Secure Kernel Code Integrity module checks the integrity of DUMPFVE.SYS before it is loaded. See section 5.1 Integrity Techniques for details on the EVMs and how the module's integrity is checked.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-4) (A4008)	Software Integrity	SW/FW Integrity	Pre-operational integrity test is run at every module initialization before the CASTs (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
RSA SigVer (FIPS186-4) (A4009)	Software Integrity	SW/FW Integrity	Pre-operational integrity test is run at every module initialization before the CASTs (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)

Table 25: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
[EVM] RSA SigVer (FIPS186-4) (A4008)	KAT	CAST	Conditional CASTs are run at every module initialization before the module integrity is verified (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
[EVM] RSA SigVer (FIPS186-4) (A4009)	KAT	CAST	Conditional CASTs are run at every module initialization before the module integrity is verified (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)
[EVM] SHA2-256 (A4008)	KAT	CAST	Conditional CASTs are run at every module initialization before the module integrity is verified (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
[EVM] SHA2-256 (A4009)	KAT	CAST	Conditional CASTs are run at every module initialization before the module integrity is verified (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CBC (A4008) - Decrypt	KAT	CAST	Conditional CASTs are run at every module initialization after the module integrity is verified (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
AES-CBC (A4008) - Encrypt	KAT	CAST	Conditional CASTs are run at every module initialization after the module integrity is verified (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
AES-CBC (A4009) - Decrypt	KAT	CAST	Conditional CASTs are run at every module initialization after the module integrity is verified (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)
AES-CBC (A4009) - Encrypt	KAT	CAST	Conditional CASTs are run at every module initialization after the module integrity is verified (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)
AES-XTS Testing Revision 2.0 (A4008) - Encrypt	KAT	CAST	Conditional CASTs are run at every module initialization after the module integrity is verified (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
AES-XTS Testing Revision 2.0 (A4008) - Key Equivalence	Fault-detection	CAST	Before the FVEK is used (Windows 11 version 22H2).	Manual on-demand (operator initiated by rebooting the computer)
AES-XTS Testing Revision 2.0 (A4009) - Encrypt	KAT	CAST	Conditional CASTs are run at every module initialization after the module integrity is verified (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)
AES-XTS Testing Revision 2.0 (A4009) - Key Equivalence	Fault-detection	CAST	Before the FVEK is used (Windows Server 2022).	Manual on-demand (operator initiated by rebooting the computer)

Table 26: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Driver Load Failure	Driver load failure	The error state occurs if the BitLocker Dump Filter integrity test fails or if the cryptographic algorithm known answer tests fail.	Restart the computer.	The module unloads itself.
Fault Detected	Fault detected	This error state occurs if the AES-XTS fault detection key equivalency test fails.	The operation returns an error.	The module rejects AES-XTS keys.

Table 27: Error States

10.5 Operator Initiation of Self-Tests

To perform the module self-tests on demand, including running the approved services Perform Pre-operational Software Integrity Test [EVM] and Perform Cryptographic Algorithm Self-Tests, the user may reboot the computer.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The Windows operating system must be pre-installed on a computer by an OEM, installed by the end-user, by an organization's IT administrator, or updated from a previous Windows version downloaded from Windows Update. An inspection of authenticity can be made by following the guidance at this Microsoft web site: https://www.microsoft.com/en-us/howtotell/default.aspx.

For Windows Updates, the client only accepts binaries signed by Microsoft certificates. The Windows Update client only accepts content whose SHA2 hash matches the SHA2 hash specified in the metadata. All metadata communication is done over a Transport Layer Security (TLS) port. Using TLS ensures that the client is communicating with the real server and so prevents a malicious TLS server from communicating to the TLS client. The version and digital signature of any new cryptographic module must be verified to match the version that was validated. See Section 11.2 Administrator Guidance for details on how to do this.

Module initialization occurs automatically as part of the Windows boot process after the user enables BitLocker. The finite state model diagram below visualizes the initialization process along with other module states. Every state of the module can transition to the power-off state through power-cycle/rebooting the host machine.

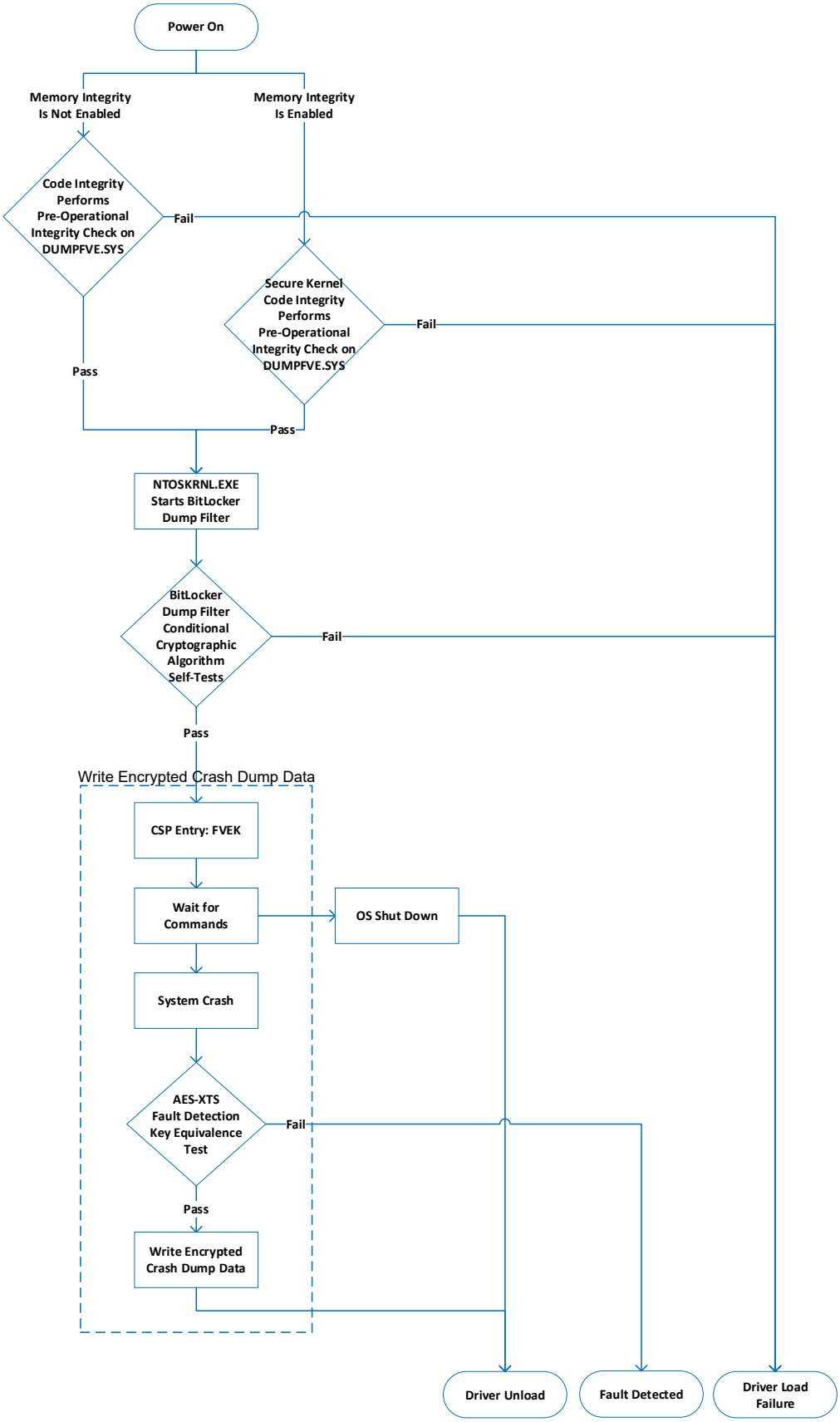


Figure 3: Finite State Model

11.2 Administrator Guidance

The installed version of Windows must be checked to match the version that was validated. See Section 11.2.1 Verifying the Installed Windows Version below for details on how to do this. To sanitize the module, the operator should reformat the hard drive or wipe the device as part of unenrollment for Azure Entra ID (formerly Active Directory.

11.2.1 Verifying the Installed Windows Version

The following methods may be used to check the installed version of Windows against the version number listed in 2.2 Tested and Vendor Affirmed Module Version and Identification.

Using the Windows command prompt or Windows PowerShell (local or remote):

- Open a command prompt or PowerShell window.
- At the prompt, type **systeminfo** and press the Enter key.
- Near the top of the output, information like the following is displayed. The OS Version field lists the installed Windows version. Compare this version number against the version number listed in 2.2 Tested and Vendor Affirmed Module Version and Identification.

```
OS Name:                   Microsoft Windows 11 Enterprise
OS Version:                10.0.xxxxx N/A Build xxxxx
OS Manufacturer:           Microsoft Corporation
```

For Windows installations without a user interface, e.g., Windows Server with the Core Installation option, a server management solution may also be used to validate the installed Windows version. For example, the Overview page of Windows Admin Center Server Manager lists the version number under the Operating System category. For more information, see Manage Servers with Windows Admin Center.

11.2.2 Verifying the Cryptographic Module Version and its Signature

To confirm the version number or digital signature of the module, locate the module binary or binaries named in 2.2 Tested and Vendor Affirmed Module Version and Identification in their default installation location. The list below identifies the default install locations for the Windows cryptographic module binaries for a system where Windows has been installed on the C: drive.

- DUMPFVE.SYS - C:\Windows\System32\

To validate the module version number, use Windows Explorer or PowerShell (local or remote).

- Using Windows Explorer:
  - Open Windows Explorer and navigate to the folder where the binary is installed, referencing the list above for the correct location.
  - Find the file in the folder and **right click** on the file's icon.
  - Select **Properties** from the context menu.
  - Select the **Details** tab.

- Compare the version number in the **File version** field against the version identified in 2.2 Tested and Vendor Affirmed Module Version and Identification.

- Using PowerShell:
  - Open a **PowerShell** window.
  - Use the **Get-ItemProperty cmdlet** together with the path of the cryptographic module binary identified above, formatting the output as a list. For example, if the binary path is `C:\Windows\System32\dumpfve.sys`, the PowerShell command is:

    ```
    Get-ItemProperty -Path "C:\Windows\System32\dumpfve.sys" | Format-List
    ```

  - The cmdlet will return output that summarizes file property details, including the version number. If the version number listed in the **VersionInfo / ProductVersion** field matches one of the version numbers identified in 2.2 Tested and Vendor Affirmed Module Version and Identification, then the module version has been verified.
  - Full documentation for the Get-ItemProperty cmdlet may be found at: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/get-itemproperty.

To validate the Windows digital signature for the module binary, use Windows Explorer or PowerShell (local or remote).

- Using Windows Explorer:
  - Open Windows Explorer and navigate to the folder where the binary is installed, referencing the list at the beginning of this section for the correct location.
  - Find the file in the folder and **right click** on the file's icon.
  - Select **Properties** from the context menu.
  - Select the **Digital Signatures** tab.
  - In the **Signature** list, select the **Microsoft Windows** signer.
  - Click the **Details** button.
  - Under the Digital Signature Information, you should see: "`This digital signature is OK.`" If that condition is true then the digital signature has been verified.

- Using PowerShell:
  - Open a **PowerShell** window.

- Use the **Get-AuthenticodeSignature cmdlet** together with the path the path of the cryptographic module binary identified at the beginning of this section. For example, if the binary path is `C:\Windows\System32\dumpfve.sys`, the PowerShell command is:

```
Get-AuthenticodeSignature -FilePath
"C:\Windows\System32\dumpfve.sys"
```

- The cmdlet will return output that summarizes signature details. If the signature is valid, the **Status** field will show "`Valid`" and the **StatusMessage** field will show "`Signature verified.`"
- Full documentation for the Get-AuthenticodeSignature cmdlet may be found at: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.security/get-authenticodesignature.

11.3 Non-Administrator Guidance

The module implements a single role only, Cryptographic Officer. See the Administrator Guidance above.

11.4 Design and Rules

The module is a multi-chip standalone software-hybrid module that operates in its approved mode during normal operation of the computer and Windows operating system and provides cryptographic services within the Operational Environments listed in Section 2.2 Tested and Vendor Affirmed Module Version and Identification. The other sections of this Security Policy provide additional details on the design of the module and its rules of operation.

12 Mitigation of Other Attacks

12.1 Attack List

The following table lists the mitigations of other attacks for this cryptographic module.

Algorithm	Protected Against	Mitigation	Constraints / Guidance
AES	Timing Analysis Attack	Constant time implementation.	None
	Cache Attack	Memory access pattern is independent of any confidential data.	None
		Module prevents observation of memory for AES rounds.	Effective only for computers that use the AES-NI instruction set.

Table 28: Mitigation of Other Attacks

13 Standards References

- FIPS 140-3, Security Requirements for Cryptographic Modules, https://csrc.nist.gov/publications/detail/fips/140/3/final

- FIPS 180-4, Secure Hash Standard (SHS), https://csrc.nist.gov/publications/detail/fips/180/4/final
- FIPS 186-4, Digital Signature Standard (DSS), https://csrc.nist.gov/publications/detail/fips/186/4/final
- FIPS 186-5, Digital Signature Standard (DSS), https://csrc.nist.gov/pubs/fips/186-5/final
- FIPS 197, Advanced Encryption Standard (AES), https://csrc.nist.gov/publications/detail/fips/197/final
- NIST SP 800-38A, Recommendation for Block Cipher Modes of Operation: Methods and Techniques, https://csrc.nist.gov/publications/detail/sp/800-38a/final
- NIST 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