Microsoft Windows

FIPS 140 Validation

Microsoft Windows 10 (May 2019 Update, November 2019 Update and May 2020 Update)

Microsoft Windows Server (versions 1903, 1909, and 2004)

Non-Proprietary

Security Policy Document

Document Information	
Version Number	1.2
Updated On	May 19, 2023

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.

Version History

Version	Date	Summary of Changes
1.0	November 4, 2020	Draft sent to NIST CMVP
1.1	October 28, 2022	Updates in response to NIST comments
1.2	May 19, 2023	Added CMVP certificate details for bounded modules

TABLE OF CONTENTS

1 Introduction

BitLocker Drive Encryption is a data protection feature of the Windows 10 operating system which encrypts data on a storage volume.

This security policy document describes the BitLocker Dump Filter cryptographic module which protects hibernation files and crash dump files on BitLocker encrypted computers. Other parts of BitLocker are described in the Security Policy Documents for Boot Manager, Windows OS Loader, and Windows OS Resume.

The BitLocker Dump Filter is part of the system dump stack. Whenever the dump stack is called during a crash or starting the hibernation process, this module ensures that all data is encrypted before written to storage as a dump file or hibernation file.

1.1 List of Cryptographic Module Binary Executables

The BitLocker Dump Filter module contains the following binary. Each binary has a distinct implementation per build for each instruction set (x86, x64, ARM64).

- DUMPFVE.SYS

The Windows builds and instruction sets covered by this validation are:

- Windows 10 and Windows Server version 1903, build 10.0.18362
  - x86
  - x64
- Windows 10 and Windows Server version 1909, build 10.0.18363
  - x86
  - x64
- Windows 10 and Windows Server version 2004, build 10.0.19041
  - x86
  - x64
  - ARM64

Tables 1-3 below present the matrix of hardware platforms, Windows builds, and Windows editions validated.

1.2 Validated Platforms

The Windows editions covered by this validation are:

- Microsoft Windows 10 Home Edition (32-bit version)
- Microsoft Windows 10 Pro Edition (64-bit version)
- Microsoft Windows 10 Enterprise Edition (64-bit version)
- Microsoft Windows 10 Education Edition (64-bit version)
- Windows Server Core Standard
- Windows Server Core Datacenter

The BitLocker Dump Filter components listed in Section 1.1 were validated using the machine configurations and Windows Operating System versions specified in the table below.

All the computers for Windows 10 and Windows Server listed in the table below are all 64-bit Intel architecture and implement the AES-NI instruction set but not the SHA Extensions. The exceptions are:

- Dell Inspiron 660s - Intel Core i3 without AES-NI and SHA Extensions
- HP Slimline Desktop - Intel Pentium with AES-NI and SHA Extensions
- Dell PowerEdge 7425 - AMD EPYC 7251 with AES-NI and SHA Extensions
- Microsoft Surface Pro X - Microsoft SQ1 with Arm Neon

Table 1 Validated Platforms for Windows 10 and Windows Server version 1903

Computer	Windows 10 Home	Windows 10 Pro	Windows 10 Enterprise	Windows 10 Education	Windows Server Core	Windows Serve Core Datacenter
Microsoft Surface Go - Intel Pentium		√				
Microsoft Surface Book 2 - Intel Core i7		√	√			
Microsoft Surface Pro 6 - Intel Core i5		√	√			
Microsoft Surface Laptop 2 - Intel Core i5		√	√	√		
Microsoft Surface Studio 2 - Intel Core i7			√			
Microsoft Windows Server 2019 Hyper-V[1]					√	√
Microsoft Windows Server 2016 Hyper-V[2]					√	

[1] Hardware Platform: Dell PowerEdge R740 Server - Intel Xeon Gold

[2] Hardware Platform: Dell PowerEdge R7425 Server - AMD EPYC 7251

Dell Latitude 12 Rugged Tablet - Intel Core i5		√				
Dell Latitude 5290 - Intel Core i7			√			
Dell PowerEdge R740 - Intel Xeon Gold						√
Dell PowerEdge R7425 - AMD EPYC 7251					√	
Dell Inspiron 660s [with x86 Windows] - Intel Core i3	√					
HP Slimline Desktop - Intel Pentium		√				
HP ZBook15 G5 - Intel Core i5		√				
HP EliteBook x360 830 G5 - Intel Core i5			√			
Samsung Galaxy Book 10.6” - Intel Core m3		√				
Samsung Galaxy Book 12” - Intel Core i5			√			
Panasonic Toughbook - Intel Core i5		√				

Table 2 Validated Platforms for Windows 10 and Windows Server version 1909

Computer	Windows 10 Home	Windows 10 Pro	Windows 10 Enterprise	Windows 10 Education	Windows Server Core	Windows Server Core Datacenter
Microsoft Surface Go - Intel Pentium				√		
Microsoft Surface Go LTE - Intel Pentium			√			

Microsoft Surface Book 2 - Intel Core i7			√			
Microsoft Surface Pro LTE - Intel Core i5		√				
Microsoft Surface Pro 6 - Intel Core i5			√			
Microsoft Surface Laptop 2 - Intel Core i5		√				
Microsoft Surface Studio 2 - Intel Core i7		√				
Microsoft Windows Server 2019 Hyper-V[3]						√
Microsoft Windows Server 2016 Hyper-V[4]					√	
Dell Latitude 7200 2-in-1 - Intel Core i7		√				
Dell Latitude 5300 2-in-1 - Intel Core i7			√			
Dell PowerEdge R740 - Intel Xeon Platinum						√
Dell PowerEdge R7425 - AMD EPYC 7251					√	
Dell Inspiron 660s [with x86 Windows] - Intel Core i3		√				
HP ProBook 650 G5 - Intel Core i7		√				
HP EliteBook x360 830 G6 - Intel Core i7			√			

[3] Hardware Platform: Dell PowerEdge R740 Server - Intel Xeon Platinum
[4] Hardware Platform: Dell PowerEdge R7425 Server - AMD EPYC 7251

HP Slimline Desktop - Intel Pentium	√					
Panasonic Toughbook CF-33 - Intel Core i5			√			
Samsung Galaxy Book 10.6" - Intel Core m3		√				
Samsung Galaxy Book 12" - Intel Core i5			√			
Microsoft Surface Pro 7 - Intel Core m3		√				
Microsoft Surface Laptop 3 - Intel Core i5			√			

Table 3 Validated Platforms for Windows 10 and Windows Server version 2004

Computer	Windows 10 Home	Windows 10 Pro	Windows 10 Enterprise	Windows 10 Education	Windows Server Core	Windows Server Core Datacenter
Microsoft Surface Pro LTE - Intel Core i5		√				
Microsoft Surface Pro 7 - Intel Core i3			√			
Microsoft Surface Pro 6 - Intel Core i7			√			
Microsoft Surface Pro X - Microsoft SQ1			√			
Microsoft Surface Go - Intel Pentium				√		
Microsoft Surface Go LTE - Intel Core i7		√				
Microsoft Surface Go 2 - Intel Core m3		√				

Microsoft Surface Go 2 LTE - Intel Pentium			√			
Microsoft Surface Laptop 2 - Intel Core i5		√				
Microsoft Surface Laptop 3 - Intel Core i5		√				
Microsoft Surface Book 2 - Intel Core i7			√			
Microsoft Surface Studio 2 - Intel Core i7		√				
Microsoft Windows Server 2019 Hyper-V[5]					√	√
Microsoft Windows Server 2016 Hyper-V[6]					√	
Dell Latitude 7200 2-in-1 - Intel Core i7		√				
Dell Latitude 5300 2-in-1 - Intel Core i7			√			
Dell PowerEdge R640 - Intel Xeon Gold						√
Dell PowerEdge R740 - Intel Xeon Platinum						√
Dell Inspiron 660s [with x86 Windows] - Intel Core i3		√				
Dynabook TECRA-X50-F - Intel Core i7		√				
HP Slimline Desktop - Intel Pentium	√					

[5] Hardware Platform: Dell Precision 5810 - Intel Xeon E5
[6] Hardware Platform: Dell PowerEdge R740 - Intel Xeon Platinum

HP ZBook 15G6 - Intel Core i7		√				
HP EliteBook x360 830 G6 - Intel Core i7			√			
HP ProBook 650 G5 - Intel Core i7		√				
Panasonic Toughbook FZ-55 - Intel Core i5			√			
Dell PowerEdge R7515 - AMD EPYC 7702P					√	

2 Cryptographic Module Specification

BitLocker Dump Filter is a multi-chip standalone module that operates in FIPS-approved mode during normal operation of the computer and Windows operating system.

The following configurations and modes of operation will cause BitLocker Dump Filter to operate in a non-approved mode of operation:

- Boot Windows in Debug mode
- Boot Windows with Driver Signing disabled
- Windows enters the ACPI S4 power state[7]

2.1 Cryptographic Boundary

The software binary that comprises the cryptographic boundary for BitLocker Dump Filter is DUMPFVE.SYS.

2.2 FIPS 140-2 Approved Algorithms

BitLocker Dump Filter implements the following FIPS 140-2 Approved algorithms:[8]

Table 4

Algorithm	Windows 10 and Windows Server version 1903	Windows 10 and Windows Server version 1909	Windows 10 and Windows Server version 2004
FIPS 197 AES CBC 128 and 256	#C785	#C1363	#C1897
NIST SP 800-38E AES XTS 128 and 256	#C785	#C1363	#C1897
NIST SP 800-38C AES CCM 256	#C798	#C1364	#C1946

[7] The ACPI S4 power state applies only for editions / versions where Windows Resume is not certified.
[8] This module may not use some of the capabilities described in each CAVP certificate.

2.3 Non-Approved Algorithms

BitLocker Dump Filter implements only Approved algorithms.

2.4 FIPS 140-2 Approved Algorithms from Bounded Modules

A bounded module is a FIPS 140 module which provides cryptographic functionality that is relied on by a downstream module. As described in the Integrity Chain of Trust section, the BitLocker Dump Filter depends on the following modules and algorithms:

When Memory Integrity, called HVCI in previous Windows 10 versions, is not enabled, Code Integrity versions 1903, 1909, and 2004 (module certificate #4511) provides:

- CAVP certificates #C785, #C1363, #C1897 (Windows 10 and Windows Server for FIPS 186-4 RSA PKCS#1 (v1.5) digital signature verification with 2048 moduli; supporting SHA-256
- CAVP certificates #C785, #C1363, #C1897 (Windows 10 and Windows Server) for FIPS 180-4 SHS SHA-256

When Memory Integrity is enabled, Secure Kernel Code Integrity versions 1903, 1909, and 2004 (module certificate #4512) provides:

- CAVP certificates #C785, #C1363, #C1897 (Windows 10 and Windows Server for FIPS 186-4 RSA PKCS#1 (v1.5) digital signature verification with 2048 moduli; supporting SHA-256
- CAVP certificates #C785, #C1363, #C1897 (Windows 10 and Windows Server) for FIPS 180-4 SHS SHA-256

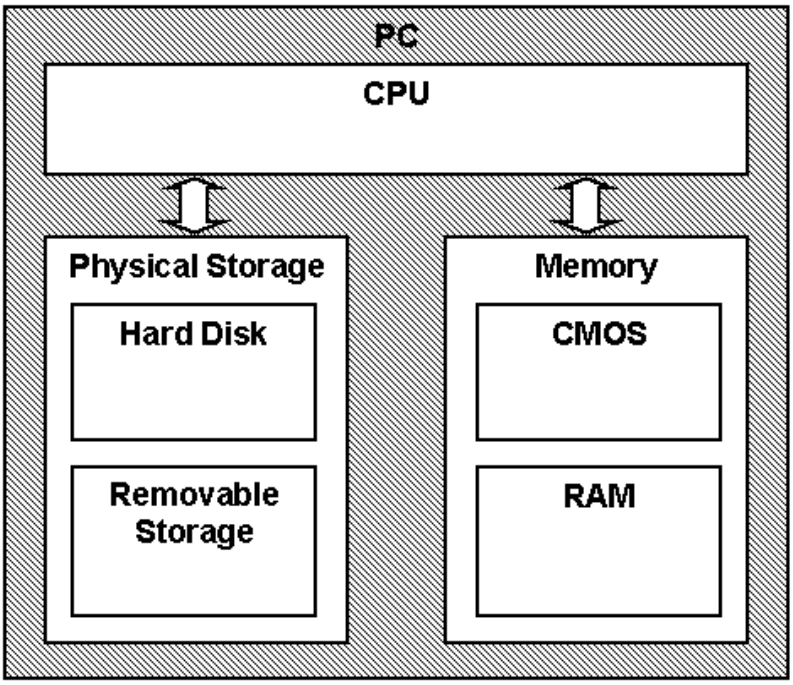
Note that the validated platforms listed in section 1.2 include processors that support the SHA Extensions. This module does not implement SHA, but the bounded modules may implement SHA and, therefore, use the SHA Extensions.

2.5 Cryptographic Bypass

Cryptographic bypass is not supported by BitLocker Dump Filter.

2.6 Hardware Components of the Cryptographic Module

The physical boundary of the module is the physical boundary of the computer that contains the module. The following diagram illustrates the hardware components used by the BitLocker Dump Filter module:



3 Cryptographic Module Ports and Interfaces

3.1 Control Input Interface

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.

3.1.1 GetFveContext

```
NTSTATUS  GetFveContext(
        __in PFILTER_EXTENSION Context,
        __in ULONG MaxPagesPerWrite,
        __inout_xcount(FveContext->StructureSize) PFVE_CONTEXT FveContext
        )
```

This function gets the BitLocker Full Volume Encryption Key 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.

3.1.2 DumpWrite

```
NTSTATUS  DumpWrite(
        PFILTER_EXTENSION Context,
        PLARGE_INTEGER DiskByteOffset,
```


PMDL Mdl
)

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.

3.2 Status Output Interface

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 dump filter securely zeroes out memory for any keys handed to it and unloads itself.

3.3 Data Output Interface

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 or hibernate file. Data exits the module in the form of encrypted blocks that may be written to a crash dump file or a hibernation file on an encrypted volume.

3.4 Data Input Interface

The Data Input Interface includes the GetFveContext function and DumpWrite function. GetFveContext is responsible for reading the FVEK. DumpWrite accepts the memory blocks to encrypt with the FVEK and the target disk locations for the blocks as input.

4 Roles, Services and Authentication

4.1 Roles

BitLocker Dump Filter is a kernel-mode driver that does not interact with the user through any service therefore the module's functions are fully automatic and not configurable. FIPS 140 validations define formal "User" and "Cryptographic Officer" roles. Both roles can use any BitLocker Dump Filter service.

4.2 Services

BitLocker Dump Filter services are described below. This module does not export any cryptographic functions.

1. **Writing encrypted crash dump data** – This service is executed when the system crashes and must write the crash dump file to an encrypted volume.
2. **Writing encrypted hibernation file data** - This service is executed when the system enters the hibernation (S4) power state and must write the hibernation file to an encrypted volume.

3. **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.
4. **Self-Tests** - The module provides a power-up self-tests service that is automatically executed when the module is loaded into memory.
5. **Zeroizing Cryptographic Material -** This service is executed as part of the module shutdown. See Cryptographic Key Management

The following table maps the services to their corresponding algorithms and critical security parameters (CSPs) as described in Cryptographic Key Management.

Table 5

Service	Algorithms	CSPs	Invocation
Writing encrypted crash dump data	FIPS 197 AES: AES CBC 128 and 256 bits AES XTS 128 and 256 bits[9] AES CCM 256	Full Volume Encryption Key (FVEK)	This service is fully automatic.
Writing encrypted hibernation data	FIPS 197 AES: AES CBC 128 and 256 bits AES XTS 128 and 256 bits[10] AES CCM 256	Full Volume Encryption Key (FVEK)	This service is fully automatic.
Show Status	None	None	This service is fully automatic. This service is executed upon completion of the Control Input Interfaces.
Self-Tests	AES-CBC - Encrypt/Decrypt KATs AES-CCM - Encrypt/Decrypt KATs Software Integrity Test (2048-bit RSA with SHA-256) AES XTS KAT	None	This service is fully automatic.
Zeroizing Cryptographic Material (see Section 9)	None	Full Volume Encryption Key (FVEK)	This service is fully automatic.

[9] The length of the data unit does not exceed 2^{20} AES blocks for storage applications such as BitLocker.
[10] The length of the data unit does not exceed 2^{20} AES blocks for storage applications such as BitLocker.

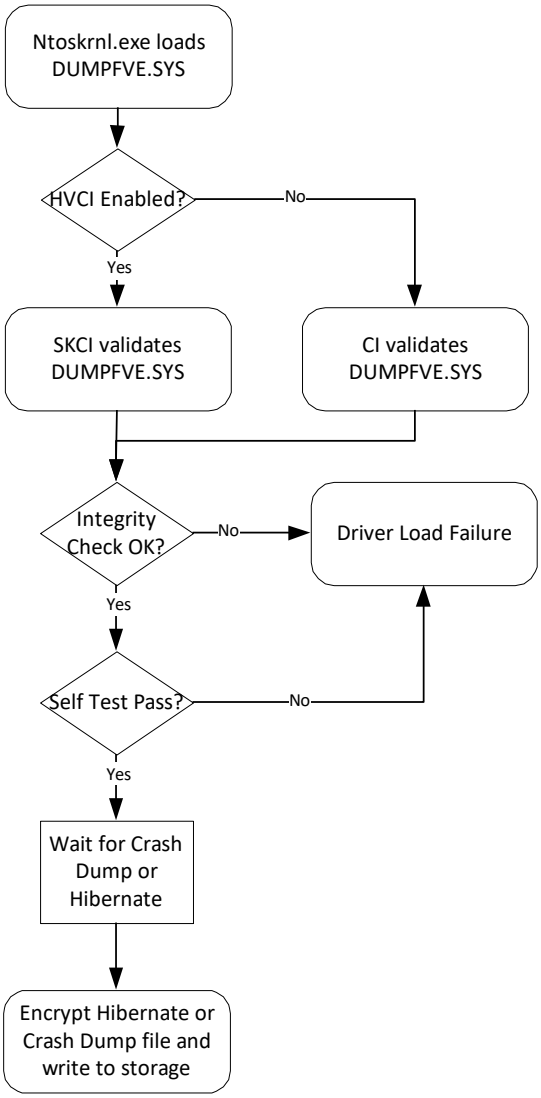
4.3 Authentication

The module does not provide authentication. Roles are implicitly assumed based on the services that are executed.

5 Finite State Model

5.1 Specification

The following diagram shows the finite state model for BitLocker Dump Filter:



6 Operational Environment

The operational environment for BitLocker Dump Filter is the Windows 10 operating system running on a supported hardware platform.

6.1 Single Operator

The BitLocker Dump Filter is loaded into kernel memory as part of the boot process before the logon component is initialized, and so there necessarily is a single operator for the module.

6.2 Cryptographic Isolation

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 CPU scheduling between the kernel and user-mode processes.

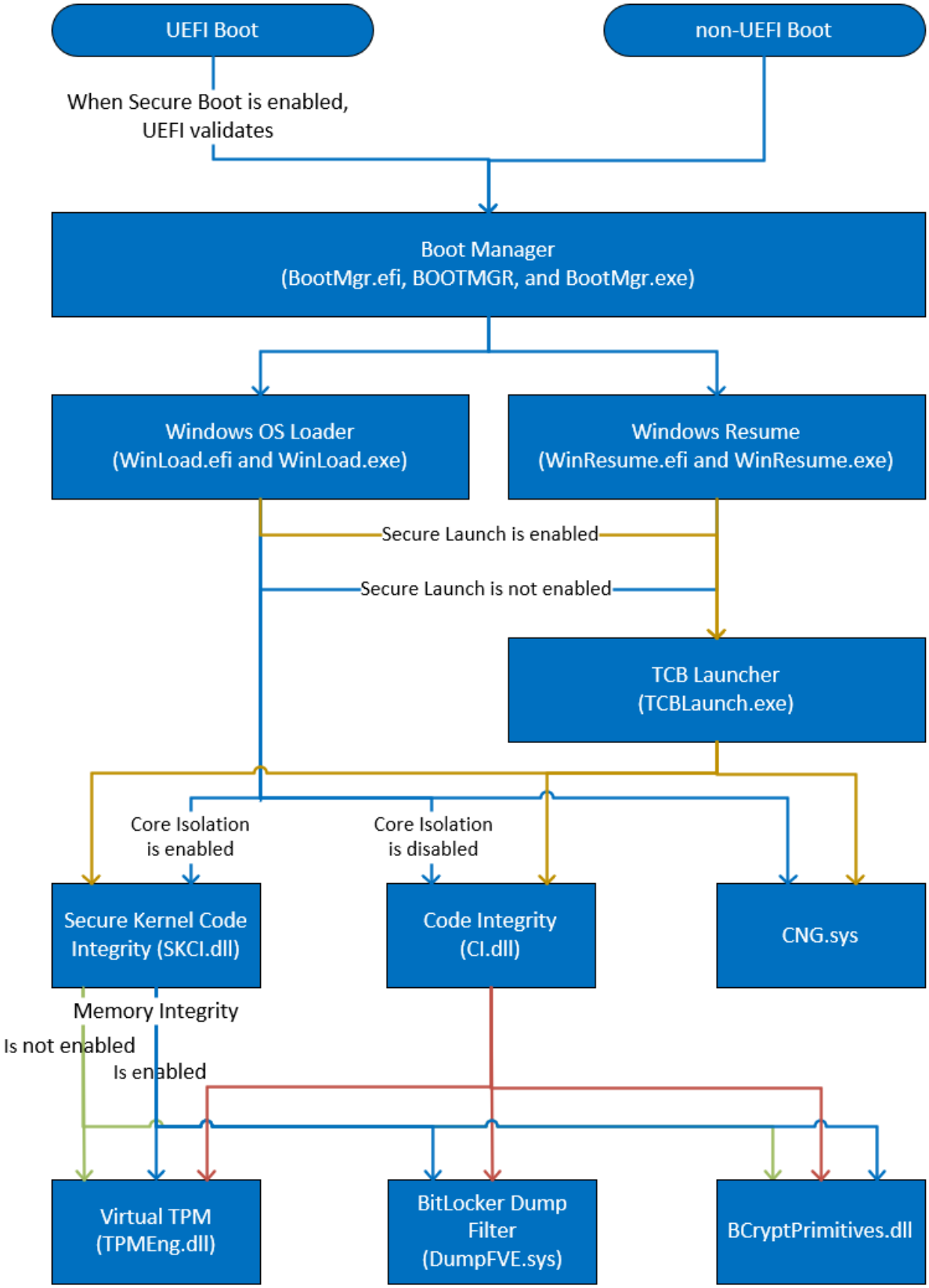
6.3 Integrity Chain of Trust

Windows uses several mechanisms to provide integrity verification depending on the stage in the boot sequence and also on the hardware and configuration. The following diagram describes the Integrity Chain of Trust for each supported configuration for the following Windows versions:

- Windows 10 and Windows Server version 1909 build 10.0.18363
- Windows 10 and Windows Server version 2004 build 10.0.19041

For the supported configurations of the following Windows version, TCB Launcher is excluded, but the remainder of the Integrity Chain of Trust diagram applies.

- Windows 10 and Windows Server version 1903 build 10.0.18362



Note that TCB Launcher was not tested for Windows 10 and Windows Server version 1903 (build 10.0.18362). The scope of this validation only includes TCB Launcher in the Integrity Chain of Trust for the applicable tested configurations for Windows 10 and Windows Server version 1909 (build 10.0.18363) and Windows 10 and Windows Server version 2004 (build 10.0.19041).

The integrity of the BitLocker Dump Filter module is checked according to the following:

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

Windows binaries include a SHA-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.

7 Cryptographic Key Management

7.1 Critical Security Parameters

When the System Volume is encrypted with Bitlocker, BitLocker Dump Filter uses this critical security parameter (CSP):

- Full Volume Encryption Key (FVEK) - 128 or 256-bit AES key that is used to encrypt dump and hibernation files.

The FVEK is passed to BitLocker Dump Filter by the Windows Kernel which received the FVEK from the Windows OS Loader or Windows Resume modules.

7.2 Zeroization

7.2.1 Volatile Keys

The FVEK is zeroized when the module is unloaded as part of shutting down or hibernating Windows.

7.2.2 Persistent Keys

BitLocker Dump Filter does not have any persistent keys.

7.3 Access Control Policy

The BitLocker Dump Filter does not allow access to the cryptographic keys contained within it, so an access control table is not included in this document. BitLocker Dump Filter receives keys from outside the module and then manages them appropriately once received. BitLocker Dump Filter prevents access to its keys by zeroizing them after use.

8 Self-Tests

8.1 Power-On Self-Tests

The BitLocker Dump Filter implements Known Answer Test (KAT) functions each time the module is loaded. The module performs the following KATs:

- AES-CBC Encrypt/Decrypt Known Answer Tests
- AES-CCM Encrypt/Decrypt Known Answer Tests
- XTS-AES Encrypt/Decrypt Known Answer Tests

If the self-test fails, the module will not load and a status code STATUS_FAIL_CHECK will be returned.

9 Design Assurance

The secure installation, generation, and startup procedures of this cryptographic module are part of the overall operating system secure installation, configuration, and startup procedures for the Windows 10 operating system.

The Windows 10 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 10 version downloaded from Windows Update.

An inspection of authenticity of the physical medium can be made by following the guidance at this Microsoft web site: https://www.microsoft.com/en-us/howtotell/default.aspx

The installed version of Windows 10 must be checked to match the version that was validated. See Appendix A for details on how to do this.

For Windows Updates, the client only accepts binaries signed with Microsoft certificates. The Windows Update client only accepts content whose signed SHA-2 hash matches the SHA-2 hash specified in the metadata. All metadata communication is done over a Secure Sockets Layer (SSL) port. Using SSL ensures that the client is communicating with the real server and so prevents a spoof server from sending the client harmful requests. The version and digital signature of new cryptographic module releases must be verified to match the version that was validated. See Appendix A for details on how to do this.

10 Mitigation of Other Attacks

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

Table 6

Algorithm	Protected Against	Mitigation

AES	Timing Analysis Attack	Constant time implementation
	Cache Attack	Memory Access pattern is independent of any confidential data
		Protected Against Cache attacks only when used with AES NI

11 Security Levels

The security level for each FIPS 140-2 security requirement is given in the following table.

Table 7

Security Requirement	Security Level
Overall	1
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	NA
Operational Environment	1
Cryptographic Key Management	1
EMI/EMC	1
Self-Tests	1
Design Assurance	2
Mitigation of Other Attacks	1

12 Additional Details

For the latest information on Microsoft Windows, check out the Microsoft web site at:

https://www.microsoft.com/en-us/windows

For more information about FIPS 140 validations of Microsoft products, please see:

https://docs.microsoft.com/en-us/windows/security/threat-protection/fips-140-validation

13 Appendix A – How to Verify Windows Versions and Digital Signatures

13.1 How to Verify Windows Versions

The installed version of Windows 10 OEs must be verified to match the version that was validated using the following method:

1. In the Search box type "cmd" and open the Command Prompt desktop app.
2. The command window will open.
3. At the prompt, enter "ver".
4. The version information will be displayed in a format like this:
   `Microsoft Windows [Version 10.0.xxxxx]`

If the version number reported by the utility matches the expected output, then the installed version has been validated to be correct.

13.2 How to Verify Windows Digital Signatures

After performing a Windows Update that includes changes to a cryptographic module, the digital signature and file version of the binary executable file must be verified. This is done like so:

1. Open a new window in Windows Explorer.
2. Type "C:\Windows\" in the file path field at the top of the window.
3. Type the cryptographic module binary executable file name (for example, "CNG.SYS") in the search field at the top right of the window, then press the Enter key.
4. The file will appear in the window.
5. Right click on the file's icon.
6. Select Properties from the menu and the Properties window opens.
7. Select the Details tab.
8. Note the File version Property and its value, which has a number in this format: xx.x.xxxxx.xxxx.
9. If the file version number matches one of the version numbers that appear at the start of this security policy document, then the version number has been verified.
10. Select the Digital Signatures tab.
11. In the Signature list, select the Microsoft Windows signer.
12. Click the Details button.
13. 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.