Windows Vista Winload OS Loader (winload.exe) Security Policy

For FIPS 140-2 Validation

v 3.3
04/14/08

1 Introduction

The Windows OS Loader (WINLOAD.exe, versions 6.0.6001.18000, 6.0.6001.18027, and 6.0.6001.22125) is an operating system loader which loads the Windows Vista operating system kernel (ntoskrnl.exe) and other boot start binary image files.

1.1 Cryptographic Boundary for WINLOAD.EXE

The Windows Vista WINLOAD.EXE consists of a single executable (EXE). The cryptographic boundary for WINLOAD.EXE is defined as the enclosure of the computer system, on which WINLOAD.EXE is to be executed. The physical configuration of WINLOAD.EXE, as defined in FIPS-140-2, is multi-chip standalone.

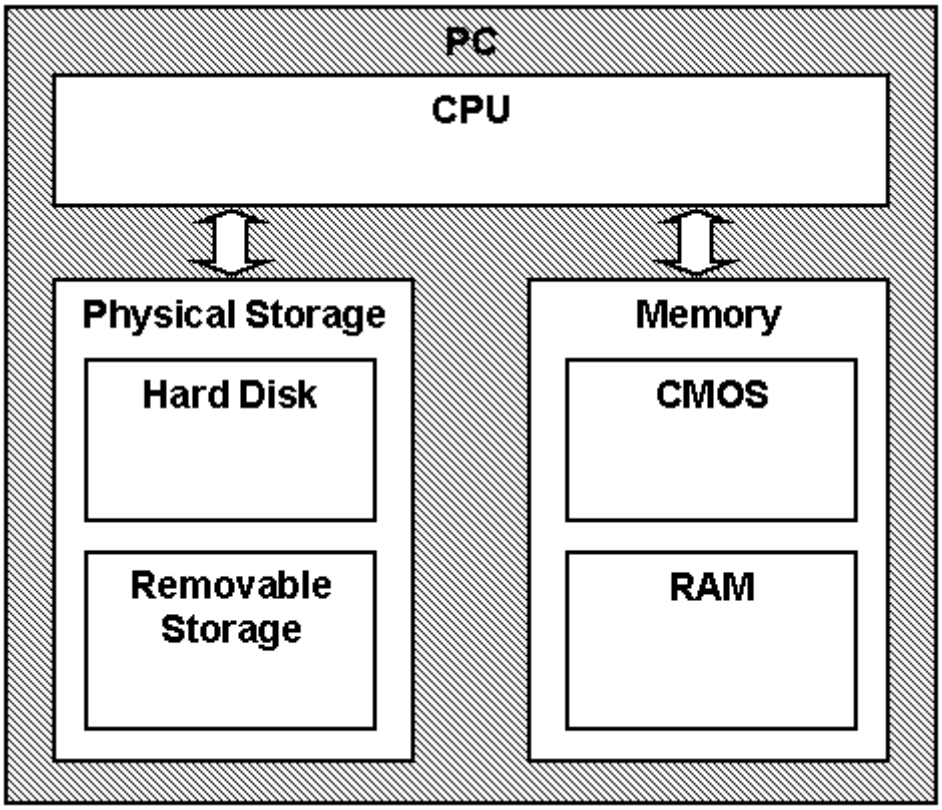
2 Security Policy

2.1 WINLOAD.EXE Security Policy

WINLOAD.EXE operates under several rules that encapsulate its security policy.

- WINLOAD.EXE is supported on Windows Vista Service Pack 1.
- WINLOAD.EXE operates in FIPS mode of operation only when used with the FIPS approved version of BOOTMGR (FIPS 140-2 Cert. [to be added once cert. # is available]) operating in FIPS mode
- Windows Vista is an operating system supporting a "single user" mode where there is only one interactive user during a logon session.
- WINLOAD.EXE is only in its Approved mode of operation when Windows is booted normally, meaning Debug mode is disabled and Driver Signing enforcement is enabled.

The following diagram illustrates the master components of the WINLOAD.EXE module



- WINLOAD.EXE's main service is to load the Windows Vista operating system kernel (ntoskrnl.exe) and other boot start binary image files, including CI.DLL, after it determines their integrity using its cryptographic algorithm implementations using the FIPS 140-2 approved algorithms mentioned below. After the verified kernel and boot start binary image files, including CI.DLL, are loaded, WINLOAD.EXE passes the execution control to the kernel and it terminates its own execution. In addition to this service, WINLOAD.EXE also provides status services. The Crypto office and User have access to the service WINLOAD supports.
- If the integrity of the kernel or CI.DLL is not verified, WINLOAD.EXE does not transfer the execution to the kernel.
- WINLOAD.EXE implements the following FIPS-140-2 Approved algorithms.
    - RSA PKCS#1 (v1.5) digital signature verification (Cert. #354)
    - SHS (Cert. #753)
    - AES (Certs. #739 and 760)

Cryptographic bypass is not supported by WINLOAD.EXE.

WINLOAD.EXE was tested using the following machine configurations:

x86	Microsoft Windows Vista Ultimate Edition SP1 (x86 version) – Dell SC430 (Intel Pentium D 2.8GHz)
x64	Microsoft Windows Vista Ultimate Edition SP1 (x64 version) – Dell SC430 (Intel Pentium D 2.8GHz)

3 WINLOAD.EXE Ports and Interfaces

3.1 Control Input Interface

The WINLOADE.EXE Control Input Interface is the set of internal functions responsible for intercepting control input. These functions are:

- BlBdInitialize – Determines if a boot debugger is attached.
- OslMain – This function receives and parses the Boot Application parameters.
- BlInitializeLibrary – Also parses Boot Application parameters.
- BlXmiRead – Reads the operator selection from the Winload user interface.

3.2 Status Output Interface

The Status Output Interface is the BlXmiWrite function that is responsible for displaying the integrity verification errors to the screen. The Status Output Interface is also defined as the BlLogData responsible for writing the name of the corrupt driver to the bootlog.

3.3 Data Output Interface

The Data Output Interface is represented by the OslArchTransferToKernel function and the AhCreateLoadOptionsString function. OslArchTransferToKernel is responsible for transferring the execution from Winload to the initial execution point of the Vista kernel. Data exits the module in the form of the initial instruction address of the Vista kernel.

Data exits the module from the AhCreateLoadOptionsString function in the form of boot application parameters passed to the Vista kernel.

3.4 Data Input Interface

The Data Input Interface is represented by the BlFileReadEx function and the BlDeviceRead function. BlFileReadEx is responsible for reading the binary data of unverified components from the computer hard drive. In addition the FVEK key can also be entered into the module over the module's data input interface. BlDeviceRead is responsible for reading data directly from devices.

4 Specification of Roles

WINLOAD.EXE supports both User and Cryptographic Officer roles (as defined in FIPS-140-2). Both roles have access to all services implemented in WINLOAD.EXE. The module does not implement any authentication services. Therefore, roles are assumed implicitly by booting the Windows Vista operating system.

4.1 Maintenance Roles

Maintenance roles are not supported by WINLOAD.EXE.

4.2 Multiple Concurrent Interactive Operators

There is only one interactive operator during a logon session. Multiple concurrent interactive operators sharing a logon session are not supported.

5 Cryptographic Key Management

WINLOAD.EXE does not store any secret or private cryptographic keys across power-cycles. However, it does use two AES keys in support of the BitLocker feature. These keys are:

- Volume Master Key (VMK) – 256-bit AES key used to decrypt the Full Volume Encryption Key.

- Full Volume Encryption Key (FVEK) - 128 or 256-bit AES key that is used to decrypt data on disk sectors of the hard drive.

Both keys are stored in memory and are zeroized by power-cycling the OS.

WINLOAD.EXE also uses public keys stored on the computer hard disk to verify digital signatures using its implementation of RSA PKCS#1 (v1.5) verify. These public keys are available to both roles. Zeroization is performed by deleting the Winload module.

All the keys (mentioned above) are accessed only by the WINLOAD.EXE service that loads the Windows Vista operating system kernel (ntoskrnl.exe) and other boot start binary image files, including CI.DLL. This service only has execute access to the keys mentioned above.

6 WINLOAD.EXE Self Tests

WINLOAD.EXE performs the following power-on (start up) self-tests.

- SHS (SHA-1) Known Answer Test
- RSA PKCS#1 (v1.5) verify with public key
- AES Known Answer Tests

7 Additional details

For the latest information on Windows Vista, check out the Microsoft web site at http://www.microsoft.com.

CHANGE HISTORY			
AUTHOR	**DATE**	**VERSION**	**COMMENT**
Stefan Santesson	2/15/2008	3.2	Based upon Gold with changes for SP1 and merged changes from CMVP review