FIPS 140-2 Non-Proprietary Security Policy

for the

Safend Cryptographic Library

Versions 3.3 and 3.4

Level 1 Validation

Document Version: Version 1.52

June 22, 2011

TABLE OF CONTENTS

DOCUMENT INTRODUCTION

Background

Federal Information Processing Standards Publication 140-2 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic products to be deployed in a Sensitive but Unclassified environment. More information about the FIPS 140-2 standard and validation program is available on the CMVP website at http://csrc.nist.gov/groups/STM/cmvp/index.html.

This non-proprietary Cryptographic Module Security Policy for the Safend Cryptographic Library ("the module") from Safend provides an overview of the product line using the module and a high-level description of how the module meets the security requirements of FIPS 140-2. This document contains details on the module's cryptographic keys and critical security parameters. This Security Policy concludes with instructions and guidance on running the module in FIPS 140-2 mode of operation.

Further Information

The Safend website (www.safend.com) contains information on the full line of products from Safend, including a detailed overview of the Safend product line that is using the module. The Cryptographic Module Validation Program website (http://csrc.nist.gov/groups/STM/cmvp/index.html) contains Safend contact information for answers to technical or sales-related questions.

Copyright Notice

With the exception of this Non-Proprietary Security Policy, FIPS 140-2 documentation is proprietary to Safend and is releasable only under appropriate non-disclosure agreements with Safend. This document may be freely reproduced and distributed in its entirety without modification (including this Copyright Notice).

Relationship to the Safend Product Line

The Safend Cryptographic Library from Safend is designed for use inside the full range of Safend products, including the Protector solution for prevention of information leakage via endpoints. As such the module implements the security relevant functions of Safend's Data Protection Suite of products. The typical "users" of the module are therefore other Safend software processes referred to as "calling daemons" in the following text. The Safend Cryptographic Library is FIPS 140-2 Level 1 validated.

SAFEND CRYPTOGRAPHIC LIBRARY

Safend Product Line Overview

Safend Protector provides a solution which enables organizations to see what ports and devices are being used in their organization (visibility), to define a policy that controls their usage and to protect data in motion. Safend Protector controls every endpoint and every device, over every network or interface. It monitors real-time traffic and applies customized, highly-granular security policies over all physical, wireless and storage device interfaces.

Safend Protector detects and allows restriction of devices by device type, model or even specific device serial number. For storage devices, Safend Protector allows security administrators to either block all storage devices completely, permit read-only, encrypt all data on devices as well as monitoring, blocking and logging files that are downloaded to or read from these devices

Safend Encryptor is a hard-disk encryption solution that leverages the security of full-disk encryption and the flexibility of file-based encryption to protect sensitive data residing on PCs and laptops. By encrypting only data files and avoiding encrypting operating system and program files, organizations benefit from their system performance and productivity remaining intact.

The Safend Cryptographic Library (or "module") provides cryptographic operations utilized by the Protector and Encryptor applications via executable called Sphinx.sys. The module provides message digest, random number generation, and data encryption/decryption functions to the Protector and Encryptor applications.

Validation Level

The following table lists the level of validation for each of the areas in FIPS 140-2 for the module:

Section No.	Area Title	Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key Management	1
8	Electromagnetic Interface/Electromagnetic Compatibility	1
9	Self-Tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A

Table 1 – Validation Level by Section

Physical security is not relevant as the module is a pure software-based solution. The "Mitigation of Other Attacks" section is also not relevant as the module does not implement any countermeasures towards specific attacks.

The module's cryptographic algorithm implementations have received the following certificate numbers from the Cryptographic Algorithm Validation Program (CAVP):

ALGORITHM TYPE	ALGORITHM	STANDARD	ALGORITHM VALIDATION CERTIFICATE	USE
Hashing	SHA-1, SHA-256	FIPS 180-2	870	Message digest
RNG	ANSI X9.31 Appendix A.2.4	ANSI X9.31	504	Random number generation Key generation
Symmetric Key	AES CBC mode with 256-bit keys	FIPS 197	879	Data encryption and decryption
Keyed Hash	HMAC-SHA1	FIPS 198	492	Module Integrity

Table 2 – Algorithm Certificates

Module Definition

The module is classified as a multi-chip standalone cryptographic module. As such, it has a well-defined physical and logical boundary. The physical boundary includes the generic, General Purpose Computer (GPC) upon which the module executes. The module's logical cryptographic boundary includes the executable image of the module stored on the PC hard drive and running from Random Access Memory (RAM). There exists a distinct set of interfaces for each boundary definition.

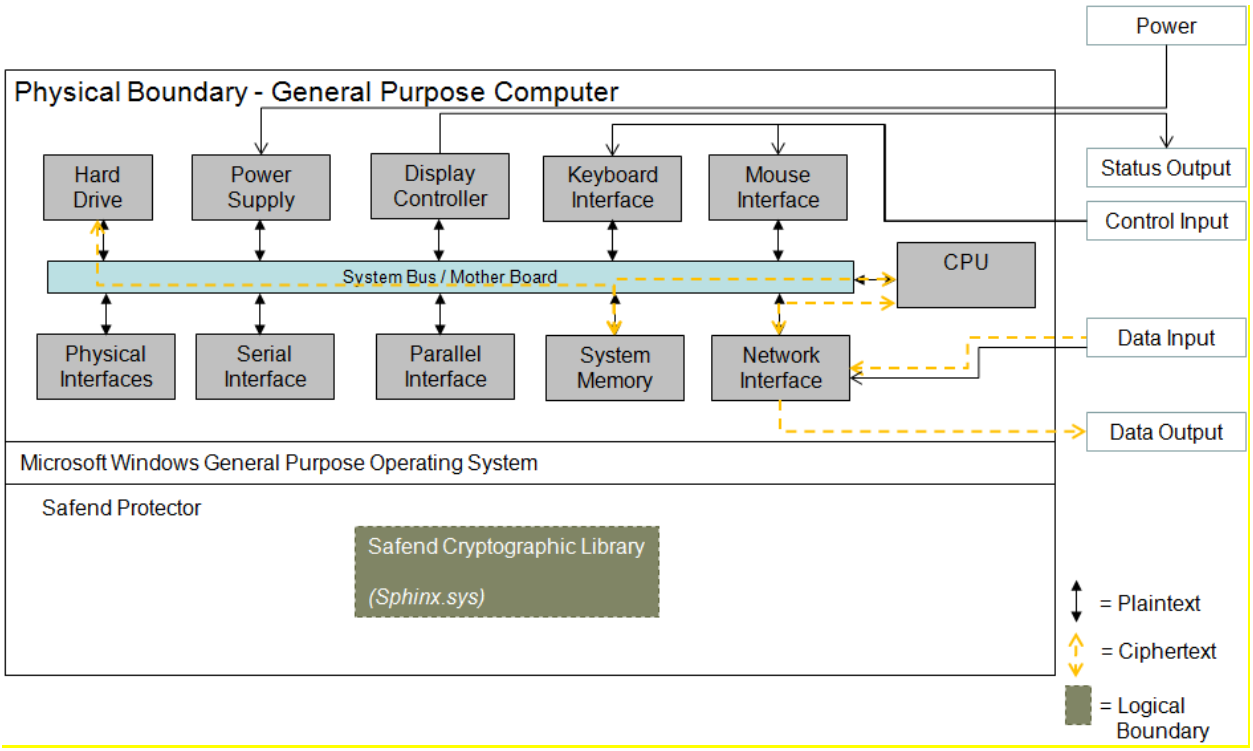

Figure 1: Physical and Logical Boundaries

The module runs as an executable called Sphinx.sys. The interface to the module is an Application Programming Interface (API) function calls, and these function calls provide the interface to the cryptographic services, for which the calling functions and some input parameters provide the control input and the return codes provide the status output.

Module Ports and Interfaces

The interfaces (ports) for the physical boundary include the computer keyboard port, CDROM drive, floppy disk, mouse, network port, parallel port, USB ports, monitor port and power plug. When operational, the module does not transmit any information across these physical ports because it is a software cryptographic library. Therefore, the module's interfaces are purely logical and are provided through the API that a calling daemon can utilize the logical interfaces expose services that applications directly call, and the API provides functions that may be called by a referencing application (see *Roles and Services* for the list of available functions).

The API provided by the module is mapped onto the FIPS 140- 2 logical interfaces: data input, data output, control input, and status output. Each of the FIPS 140- 2 logical interfaces relates to the Module's callable interface, as follows:

FIPS 140-2 LOGICAL INTERFACES	MODULE LOGICAL INTERFACES	GPC PHYSICAL PORTS
Data Input	Input arguments to API functions	Standard GPC input ports (e.g., keyboard)
Data Output	Return values or output parameters from API functions	Standard GPC output ports (e.g., monitor)
Control Input	Calls to API functions and some parameters to these functions	Standard GPC input ports (e.g., keyboard)
Status Output	Information returned via exceptions (such as return or exit codes)	Standard GPC output ports (e.g., monitor)
Power	N/A	Standard GPC power port (e.g., power connector)

Table 3 – Safend Cryptographic Library Logical Interface Mapping

Roles and Services

The module supports a *Crypto Officer* role and a *User* role. The module supports no *Maintenance* roles.

The module does not implement authentication for the two roles. Roles are implicitly selected via the services being called and the situation in which they are called as described below. There are also no internal audit trails tracking any of the events or data generated or used by the module during FIPS approved mode of operation.

Crypto Officer Role

The Crypto Officer can access all services in the module and perform initialization. The Crypto Officer role is implicitly selected or assumed when initializing the module.

User Role

The User role includes all the calling daemons (other software modules outside the Safend Cryptographic Library) using the module as part of their normal operation.

The User role is implicitly selected or assumed when calling any services on the module API when using the module during normal operations.

Available Services

The services available to the User and Crypto Officer roles in the Module consist of the following:

SERVICE	DESCRIPTION	CSP	ALGORITHM	ROLES	ACCESS
Encrypt Data	When the policy for removable media is set to "Allow Encrypted," the module will encrypt data on the removable media.	AES Symmetric Key	AES	Crypto Officer / User	Read Write Execute
Decrypt Data	The module will decrypt data on the removable media.	AES Symmetric Key	AES	Crypto Officer / User	Read Write Execute
Random Number Generation	Generate random numbers for use by host application	PRNG Seed and Seed key	X9.31	Crypto Officer / User	Read Write Execute
Initialization of module	Initializes and tests the module upon power-on and includes calls that implement on-demand status verification of the module.	None	N/A	Crypto Officer	N/A
Show Status	View status of module (i.e., FIPS mode enabled, module version, self-test success/failure)	None	N/A	Crypto Officer / User	N/A
Self Tests	Includes Integrity and known answer tests	None	N/A	Crypto Officer / User	N/A
	Module integrity check	HMAC Key	HMAC	Crypto Officer / User	Read Write Execute
Message Digest	Data hashing	None	N/A	Crypto Officer / User	N/A
Zeroization	The module calls the FIPS_RAND_CLEANUP() function, which zeroizes the PRNG Seed and Seed Key	PRNG Seed and Seed Key	N/A	Crypto Officer / User	Write

Table 4 – Module Services

Note that the conditional self-test is not a separate service call through the external API but is embedded in the Random Number Generation service specified in Table 5.

Physical Security

The module is a software-only module and does not provide any physical security mechanisms. Therefore, this section is not applicable.

Operational Environment

The module operates on a general purpose computer running on a modern version of the Microsoft Windows general purpose operating system (GPOS), including Microsoft Windows 2000 SP4, Microsoft Windows 7, Microsoft Windows 2008, Microsoft Windows Vista, Windows XP and Windows 2003 Server. For FIPS 140-2 purposes, the module runs on the Microsoft

Windows XP Professional operating system in single user mode. The operating system does not require any additional configuration to meet FIPS 140-2 requirements.

The module was tested on Microsoft Windows XP Professional running on an Intel Pentium 4 processor. The GPC used during testing met Federal Communications Commission (FCC) Electromagnetic Interference (EMI) and Electromagnetic Compatibility (EMC) requirements for home use as defined by 47 Code of Federal Regulations, Part 15, Subpart B. FIPS 140-2 validation compliance is maintained when the module is operated on other versions of the Microsoft Windows GPOS running in single user mode, assuming that the requirements outlined in CMVP IG G.5 are met.

Cryptographic Key Management

A Critical Security Parameter (CSP) is information, such as passwords, symmetric keys, asymmetric private keys, etc., that must be protected from unauthorized access, disclosure, modification, or substitution. The module is used for cryptographic operations (random number generation, encryption, decryption, etc.). The module does not support key generation, and the module does not output cryptographic keys or CSPs. The PRNG Seed and Seed Key are created externally and input in plaintext into the module for use in the ANSI X9.31 Appendix A.2.4 pseudo-random number generator.

The module uses the following CSPs:

CSP NAME	USE/DESRIPTION	GENERATION	STORAGE	DELETION
AES Symmetric Key	Symmetric encryption and decryption	The application provides an AES Symmetric Key to the module.	The key is not stored in the module. It is passed by pointer and is not copied.	The key is not stored in the module. It is the application's responsibility to zeroize it after use.
PRNG Seed and Seed Key	Seed the ANSI X9.31 PRNG	The application inputs the V seed value and the seed key into the module.	The CSPs are stored in the module's memory space in plaintext.	The CSPs are deleted either by overwriting or through the calling of function FIPS_RAND_CLEANUP. The memory is also cleared when the module is unloaded.
HMAC Key	FIPS-approved module integrity check	Generated during the manufacturing process and compiled with the binary.	The CSP is contained within the module.	The CSP is erased from memory when the module is unloaded.

Table 5 – Critical Security Parameters

Self-Tests

Two basic sets of tests are run by the module: Power-on self-tests and conditional self-tests.

Power-on self-tests are run upon every initialization of the module and, if any of the tests fail, the module will not initialize. The module will enter an error state and no services can be accessed by the users. A conditional self-test is a test that is automatically invoked as a result of a particular service being requested and is thus conditional on that service being requested. If any of these tests fail, the module will enter and error state and will power off. No services can be accessed by the users.

The available power-on self-tests are as follows:

POWER-ON SELF TEST	DESCRIPTION
AES Known Answer Test for encryption/decryption	If the test fails, the module will fail and enter an error state. Includes both encrypt and decrypt operations.
SHA-1 / Known Answer Test	If the test fails, the module will fail and enter an error state.
HMAC Known Answer Test	If the test fails, the module will fail and enter an error state.
Random Number Generator Known Answer Test	If the test fails, the module will fail and enter an error state.
Integrity Check	The module performs its own integrity check using HMAC-SHA-1 and enters into error state when/if failing. The integrity test verifies the integrity of the Protector Cryptographic Library.

Table 6 – Power-on Self Tests

The Power-on self-tests can be run on demand by reinitializing the module in the FIPS approved Mode of Operation or by calling the corresponding service on the API.

The available conditional self-tests are as follows:

CONDITIONAL SELF TEST	DESCRIPTION
Continuous Random Number Generator Test	If the test fails, the module will fail and enter an error state.

Table 7 – Conditional Self Tests

The status of all tests is output from the module to the calling daemons via the standard API to enable users to verify results and status and to initiate tests on-demand where relevant. Note that the module halts when a test fails.

Mitigation of Other Attacks

No claims have been made that the module mitigates against any other attacks.

Functions Not to Be Used in the FIPS Approved Mode of Operation

The following functions should not be run in FIPS-approved mode of operation:

- SHA-256
- DES

An attempt by the application to call the SHA-256 implementation or the DES implementation in the module will put the module in a non-FIPS-approved mode of operation. Although the SHA-256 implementation has received SHS certificate 870, the module does not implement a Known Answer Test for this implementation.

SECURE OPERATION OF THE SAFEND CRYPTOGRAPHIC LIBRARY

This section describes how to configure the module for the FIPS-approved mode of operation. Operating the module without maintaining the following settings will remove the module from the FIPS-approved mode of operation.

Crypto Officer Guidance – Module Initialization and Configuration

The Crypto Officer must verify that the software version of the module is 3.3 or 3.4 by viewing the Sphinx.sys file properties via Windows Explorer and ensure that FIPS mode is enabled by following the steps listed in the Developer Guidance section.

Developer Guidance

1. The developer is responsible to provide a zeroization command for the AES Symmetric Key when outside the Safend Cryptographic Library.
2. The developer sets FIPS mode via call of the function "FIPS_mode_set()", passing "1" as a parameter in the function call
3. The developer is responsible for ensuring the source files that comprise the Safend Cryptographic Library are built into Safend application where required.

User Guidance

The Safend Cryptographic Library is not distributed as a standalone library and is only used in conjunction with the Safend Data Protection Suite solution. As such, there is no direct User Guidance.

DEFINITION LIST

AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Programming Interface
CAVP	Cryptographic Algorithm Validation Program
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
EMC	ElectroMagnetic Compatibility
EMI	ElectroMagnetic Interference
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standard
GPC	General Purpose Computer
GPOS	General Purpose Operating System
HMAC	Hashed MAC (see MAC)
KAT	Known Answer Test
MAC	Message Authentication Code
PRNG	Pseudo-Random Number Generator
RAM	Random Access Memory
RNG	Random Number Generator
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
USB	Universal Serial Bus