Apani Kernel Crypto Module Security Policy for FIPS 140-2 Validation

Module version V1.0.1
Document version 1.0.5
June 7, 2010

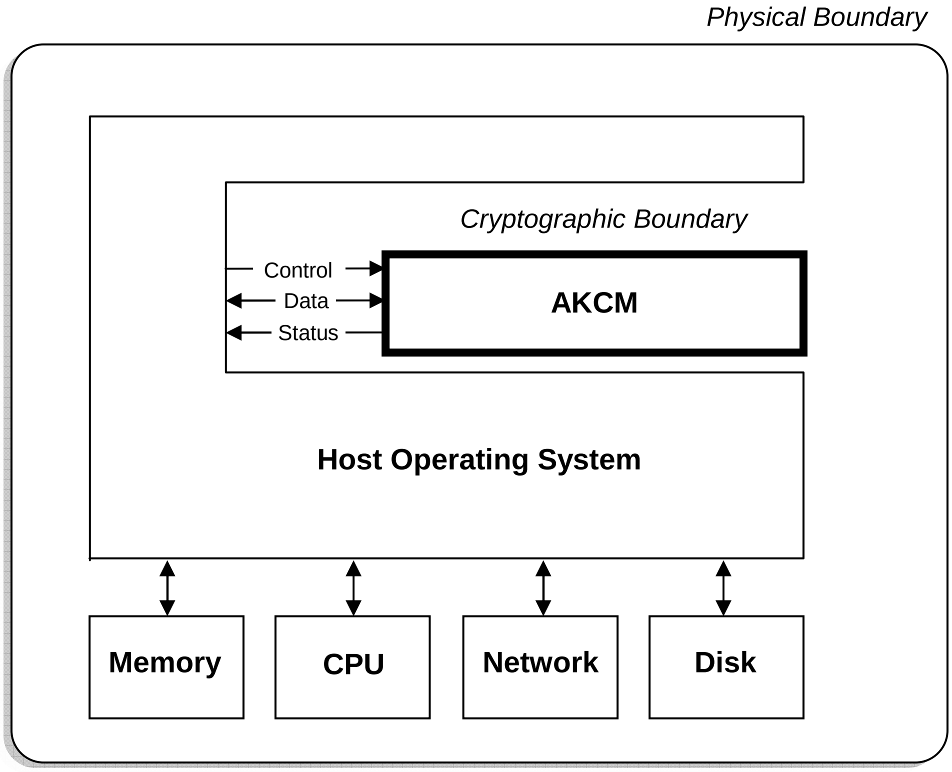
Table of Contents

1. Introduction

This document describes the FIPS 140-2 Security Policy for the Apani Kernel Crypto Module (AKCM) and how the AKCM meets all the requirements as specified in the FIPS 140-2 Level 1 requirements. This Security Policy forms part of the submission to the cryptographic module testing lab.

2. Module Overview

The AKCM is classified as a multi-chip standalone cryptographic module consisting of 1) a commercially available general purpose computer, 2) a commercially available operating system and 3) the AKCM. The AKCM is a software library that runs on a wide variety of computing platforms and performs encryption, hashing and message authentication generation functions. The cryptographic boundary is defined as the AKCM itself: a binary software library for general purpose computers. The format of this library on Windows is a kernel dynamic link library (akcm.sys). The physically contiguous cryptographic boundary is defined as the outer enclosure of the general purpose computing system. The block diagram for the module is shown below.



The AKCM is validated at Security Level 1 for all the *Security Requirements* sections documented in the FIPS 140-2 *Security Requirements For Cryptographic Modules* document except for *Physical Security* and *Mitigation of Other Attacks* which do not apply.

The AKCM is validated on the following platforms:

Operating System	Processor	Configuration
Microsoft® Windows® XP®	Intel® Core™ 2 Duo	32 bit
Microsoft Windows XP	Intel Core 2 Duo	64 bit
Microsoft Windows Server® 2003	Intel Core 2 Duo	32 bit
Microsoft Windows Server 2003	Intel Core 2 Duo	64 bit
Microsoft Windows Server® 2008	Intel Core 2 Duo	32 bit
Microsoft Windows Server 2008	Intel Core 2 Duo	64 bit

Microsoft, Windows, Windows XP and Windows Server are a registered trademarks of Microsoft Corporation in the United States and other countries.
Intel and Core are registered trademarks of Intel Corporation in the United States and other countries.

The AKCM always operates in a FIPS approved mode of operation. The AKCM is validated for each operating system when running in a single-user mode of operation.

3. Roles and Services

The AKCM implicitly recognizes the Crypto Officer and User roles where specific services are available as described in the following table:

Role	Service	Description
Either	Module Load	Instruct the operating system to load

		the AKCM from a disk file
	Initialization	Initialize the AKCM
	Self-Test	Perform the AKCM self-test and integrity test functions
	Module Unload	Instruct the operating system to unload the AKCM
	Show Status	Return the AKCM device state
Crypto Officer	Connect	Attach an AKCM session to the AKCM
	Disconnect	Detach an AKCM session from the AKCM
User	Encryption	Encrypt data using AES or Triple DES with a key
	Decryption	Decrypt data using AES or Triple DES with a key
	Message Digest	Compute SHA digest of data
	MAC	Compute HMAC digest of data with a key
	Zeroization	Set an AKCM token, key, digest or data buffer to 0

Services that do not access critical security parameters can be executed in either Crypto Officer role or the User role.

Authentication of the implicit Crypto Officer role is made by calling the connect function. Authentication of the implicit User role is accomplished by the Crypto Officer supplying a valid token to the connect function. All encryption, decryption, digest and MAC services require a valid token. AKCM does not meet all the requirements for role-based authentication.

For the AKCM, token is required to be 20 bytes in length yielding a single random login attempt probability of success of 1 in 10^48. This exceeds a one minute random login probability of one in 100,000 for as long as the host computer cannot attempt more than 10^43 login attempts per minute.

4. Physical Security

The FIPS 140-2 Physical Security requirements are not applicable to this software only module.

5. Cryptographic Key Management

All keys are imported in plain text from the invoking program running on the same computer. All keys are kept only in memory (RAM) and are not stored between invocations of the cryptographic module.

Service	Critical Security Parameter	Role, Type of Access
Encryption	Symmetric keys	User, RW
Decryption	Symmetric keys	User, RW
Message Digest	None	User, NA
MAC	HMAC Keys	User, RW
Self-Test	None	Either, NA
Show Status	None	Either, NA
Zeroization	All	User, W
Connect	Token	CO, RW or User, RO
Disconnect	Token	CO, RW
Initialization	None	Either, NA

Module Load	None	Either, NA
Module Unload	None	Either, NA

Zeroization functions are made available for externally managed keys and tokens. Zeroization functions overwrite memory containing keys with a constant.

The following events are generated by the crypto module for audit by the caller:

- Malformed requests
- Authentication failures
- Failed crypto operation attempts

The following algorithms are implemented by the Cryptographic Module:

Algorithm	Key Size	FIPS Approved
Encryption and Decryption		
Triple DES (#915)	168	X
AES (#1313)	128	X
AES (#1313)	192	X
AES (#1313)	256	X
Message Digest		
SHA-1 (#1201)		X
SHA-256 (#1201)		X
SHA-384 (#1201)		X
SHA-512 (#1201)		X
MAC		
HMAC SHA-1 (#764)	≥ 8	X
HMAC SHA-256 (#764)	≥ 8	X
HMAC SHA-384 (#764)	≥ 8	X
HMAC SHA-512 (#764)	≥ 8	X

The AKCM does not create secret, public or private keys. The following table provides a detailed list of key establishment methods:

Service	Cryptographic Keys and CSPs	Key Length	Key Strength	FIPS Approved Establishment Mechanism	Types of Access	Key State Within Module
Encryption	Triple DES	168	112	NA	RW	Ephemeral
Decryption	Triple DES	168	112	NA	RW	Ephemeral
Encryption	AES	128	128	NA	RW	Ephemeral
Decryption	AES	128	128	NA	RW	Ephemeral
Encryption	AES	192	192	NA	RW	Ephemeral
Decryption	AES	192	192	NA	RW	Ephemeral
Encryption	AES	256	256	NA	RW	Ephemeral
Decryption	AES	256	256	NA	RW	Ephemeral
MAC	HMAC Keys	≥ 8	≥ 8	NA	RW	Ephemeral

6. Self-Tests

6.1 Power-Up Self-Tests

The following power-up self-tests are run at module load. If any self-test fails, the module will enter an error state. That error state is detectable through status functions and the cryptographic module will not perform any cryptographic functions while in the error state. The error state is cleared by reloading the module.

6.1.1 Cryptographic Algorithm Known Answer Tests

Known answer tests (KAT) for encryption/decryption or hashing process a buffer for which the calculated output is known and stored within the cryptographic module. An encryption or hashing test passes when the freshly calculated output matches the expected value. A test fails when the calculated output does not match the expected value. A decryption test passes when the freshly calculated output matches the plaintext value or fails when the calculated output does not match the expected value.

The AKCM performs the following KATs:

Algorithm	Operation	Key Size	Message Size	Output Size
Triple DES	Encrypt	168	64	64
Triple DES	Decrypt	168	64	64
AES	Encrypt	128	128	128
AES	Decrypt	128	128	128
AES	Encrypt	192	128	128
AES	Decrypt	192	128	128
AES	Encrypt	256	128	128
AES	Decrypt	256	128	128
SHA-1			512	160
SHA-256			512	256
SHA-384			1024	384
SHA-512			1024	512
HMAC SHA-1		1024	1024	96 (truncated)
HMAC SHA-256		3200	1024	192 (truncated)
HMAC SHA-384		2048	1024	384
HMAC SHA-512		3072	1024	512

6.1.2 Software Integrity Tests

The module performs an integrity test using a FIPS approved algorithm to verify the module is the same as when it was delivered. The test uses the FIPS validated HMAC SHA-512 algorithm with a 2048 bit key.

6.2 Conditional Self-Tests

The Apani Kernel Crypto Module does not perform conditional self-tests since the AKCM does not perform the following cryptographic operations:

- Generate public or private keys
- Load software or firmware components into the Crypto Module
- Accept manually entered keys
- Generate random numbers
- Implement a bypass capability

7. Mitigation of Attacks

The Apani Kernel Crypto Module has not been designed to mitigate specific attacks outside the scope of FIPS 140-2.

8. Crypto Officer Guidance

8.1 Crypto Officer Responsibilities

The Crypto Officer is responsible for the following steps:

- Installing the AKCM in EpiForce Agents
- Configuring the AKCM EpiForce Systems

8.2 Installing the AKCM in EpiForce Agents

The AKCM is installed as part of the EpiForce Agent installation. No special steps are required to install the AKCM beyond the steps required to install the standard EpiForce Agent product.

9. User Guidance

9.1 Use of the AKCM in EpiForce Agents

In EpiForce systems where a supported FIPS 140-2 verified cryptographic algorithm is configured, Agents that participate in an cryptographic exchange with other Agents that support the AKCM will automatically use the AKCM FIPS 140-2 verified cryptographic algorithms. Keys and tokens should be zeroized when they are not longer needed.

10. Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
AKCM	Apani Kernel Crypto Module
CPU	Central Processing Unit
CO	Crypto Officer
CSP	Critical Security Parameter
DES	Data Encryption Standard
FIPS	Federal Information Processing Standard
HMAC	Hash-Based Message Authentication Code
KAT	Known Answer Test
MAC	Message Authentication Code
R	Read Access
RAM	Random Access Memory
RO	Read Only Access
SHA	Secure Hash Algorithm
W	Write Access