AKCode Cryptographic Module

Anonymous Key Technology (AKT)

C++ Module (CPP)

and

Anonymous Key Technology (AKT)

Java Module (JAVA)

Security Policy

FIPS 140-2 Level 1

Version 1.5

By: Lynn Spraggs, Ph.D., P.Eng
AKCode Inc.
http://www.akcode.com

Table of Contents

1. Background

1.1. Purpose

This is the non-proprietary FIPS 140-2 security policy for the Anonymous Key Technology (AKT) software Cryptographic Module (Module) developed by AKCode Inc. (AKCode). The AKCode Security Policy details the secure operation of the AKT CPP Module and the JAVA module as required in Federal Information Processing Standards Publication 140-2 (FIPS 140-2) as published by the National Institute of Standards and Technology (NIST) of the United States Department of Commerce.

1.2. References

For more information on the AKT CPP Module and the JAVA module, please visit http://www.akcode.com. For more information on the FIPS 140-2 cryptographic module validation program and the validation process or information on NIST, please visit http://csrc.nist.gov/cryptval.

2. Overview

The AKT CPP Module is a software module packaged as a Dynamic-link Library (DLL) on Windows, or as a shared object (.so) on Linux/Solaris. The AKT JAVA Module is a software module packaged as a jar or cab container of class files that are dynamically loaded by other executables. The packaged library is for use on specified operating systems (see section 3.0). The AKT CPP and JAVA Modules provide an intuitive, high-level approach that can be used to develop new secure applications, customize existing applications or by utilizing the supplied functionality to do much of the secure processing required in distributed systems today.

2.1. FIPS 140-2 Compliance:

The AKT CPP and JAVA Modules were designed and implemented to meet FIPS 140-2 requirements. As such, there are no special steps required to ensure applications using the modules are FIPS 140-2 compliant. The functionality delivered with the AKT CPP and JAVA Modules will allow most organizations to utilize the suites as validated.

2.2. Access to the Module:

The modules' authentication strength varies according to the particular method chosen for authentication. An evaluation of the authentication strength was not conducted at this time, thus AKCode makes no particular claim as pertains to the FIPS 140-2 identity or role based authentication requirements.

The AKT Module allows access to the module for operators using usernames, passwords and optional biometrics. Once access is granted, operators assume one of two primary roles: Crypto Officer (CO) or User. COs have access to user management and key database management functionality. Users have access to general cryptographic functionality. The access control procedure is performed inside the cryptographic module.

2.3. FIPS-Approved Algorithms:

The AKT Module provides confidentiality, integrity and message digest services. The AKT Module natively supports the following algorithms: AES, SHA-1 and HMAC-SHA-1. The module performs random number generation using a compliant pseudo-random number generation as specified in Appendix 3.1 of FIPS 186-2, using SHA-1 as the G function.

2.4. Data Management:

Data required by the cryptographic module is maintained within the cryptographic boundary or it is stored in an encrypted state outside the module or it is entered through an input device. All keys remain inside the module or they are exported in an encrypted state. All encryption and/or decryption of data is performed within the cryptographic boundary of the module.

Table 2 DEK Management Table

Secret Encryption Keys

KEY	Description/Usage	Generation	Storage	Entry/Output	Destruction	Establishment
Enterprise Server DEK 256 bit AES	Key used to encrypt or decrypt all data being secured for transmission outside the module.	Generated at time of installation using PRNG	Stored in plaintext in SRAM	Not exported.	Destroyed if used in SRAM	Generated inside the module and stored encrypted
Application Server DEK 256 bit AES	Key used to encrypt or decrypt all data being secured for transmission outside the module.	Generated at time of installation using PRNG	Stored in plaintext in SRAM	Not exported.	Destroyed if used in SRAM	Generated inside the module and stored encrypted
Operator DEK 256 bit AES	Key used to encrypt operator transmissions	Generated inside module via PRNG	Stored in plaintext in SRAM while user is active	Output encrypted with the PPP KEK	Is destroyed when the module is closed	Negotiated using PPP as per Annex D in FIPS 140-2
PPP KEK 256 bits AES	Key used to encrypt operator key during key distribution.	Generated internally via PRNG	Stored in plaintext in SRAM	Output in protected form in the PPP key transport protocol	Is destroyed after usage.	Temporary Transport Protocol

3. Security Specification

This document specification describes the UIS AKT Module version 1.1 Security Policy submitted for validation in accordance with the FIPS publication 140-2. It is implemented as a multi-chip stand-alone module.

The AKT Module consists of the following generic components:

- A commercially available general-purpose hardware-computing platform that is shown schematically in Figure 2.0. This platform must conform to FCC regulations.
- A commercially available Operating System (OS) that is consistent with and runs on the above platform. For the purposes of this validation, the module has been tested on four servers running LINUX Kernel version 2.4, Solaris 8, Windows NT, Windows XP and Windows 2000 and a client running Microsoft Windows 2000, Windows XP, Windows 7 and Windows 98 running in single user mode (supported in version 1.0.0), and on both a Samsung Pocket PC and Compaq IPAQ both running Windows CE (supported in version 1.0.2), though these were not all tested during operational testing.

The AKT CPP Module has been operationally tested on the following hardware computing platform and Operating Systems:

- HP Pavilion –
    - Linux 2.2 with Slackware 7.1
    - Windows 2000
        - Internet Explorer version 5.0
- COMPAQ Presario
    - Windows 2000
    - Linux 2.4.18 with Slackware 8.1
        - Internet Explorer version 5.0
        - Netscape version 7.01
- Future Shop CICERO
    - Windows NT 4.00
    - Windows XP
- SUN Server, Solaris version 8
- AKCode Kalavista, general purpose PC
    - Linux 2.4 with Suse 8.1
- 4th Generation Intel® Core Processor (Intel Core i5)
    - Windows 7 x64

The AKT JAVA Module has been operationally tested on the following hardware computing platform and Operating Systems:

- HP Pavilion –
  - Windows 2000
    - Internet Explorer version 5.0
- COMPAQ Presario
  - Windows 2000
    - Internet Explorer version 5.0
    - Netscape version 7.01
- 4th Generation Intel® Core Processor (Intel Core i5)
  - Windows 7 x64

Table 3 – Module Security Level Specification

Security Requirements Section	**Level**
Cryptographic Module	**1**
Module Interfaces	**1**
Roles and Services	**1**
Finite State Model	**1**
Physical Security	**N/A**
Operational Environment	**1**
Key Management	**1**
Cryptographic Algorithms	**1**
EMI/EMC	**3**
Self Test	**1**
Design Assurance	**1**
Mitigation of Other Attacks	**N/A**

3.1. Cryptographic Boundary and Physical Interfaces

The AKT Module is a software cryptographic system designed to comply with FIPS 140-2 Level 1 for a single user stand-alone cryptographic module. Single user mode is configured on the host PC at installation which enforces this. The module's embodiment is "Multi-chip Standalone".

The physical cryptographic boundary is the host PC and the address space of the process within which the module is loaded, the logical boundary, is the software.

Figure 1.0 provides a schematic of the cryptographic boundary and Figure 2.0 illustrates the Module resident in the hardware. The hardware interfaces are defined to be:

3.2.1 Interfaces into the Cryptographic Boundary

- Keyboard Input Device
- PAD storage devices

- Wireless Storage Device

3.2.2 Biometrics Input Device(s)

- Fingerprint Reader
- Keyboard Cadence

3.2.3 Interfaces Out of the Cryptographic Boundary

- Standard PC Ports
- TCP Port

3.2.4 Interfaces that are both Input and Output

- Smart Card Reader/Writer
- USB storage device
- Database Storage Device
- Wireless Reader/Writer

Being a software module, the AKT defines its logical interfaces in terms of the API that it provides. The Data Input Interface is defined to be all the API calls that accept, as their arguments, data to be used or processed by the Module. Data Output Interfaces are defined to be the API calls that return, by means of return value or arguments of appropriate types, data generated or otherwise processed by the Module to the caller. Finally, Control Input Interfaces are comprised of the calls used to initiate the Module and the API calls used to control the operation of the Module. The Status Output Interface is defined as special API calls that provide information about return values and the status of the Module to the requesting user. A functional block diagram is shown in Figure 3.0.

Figure 1 AKT Module

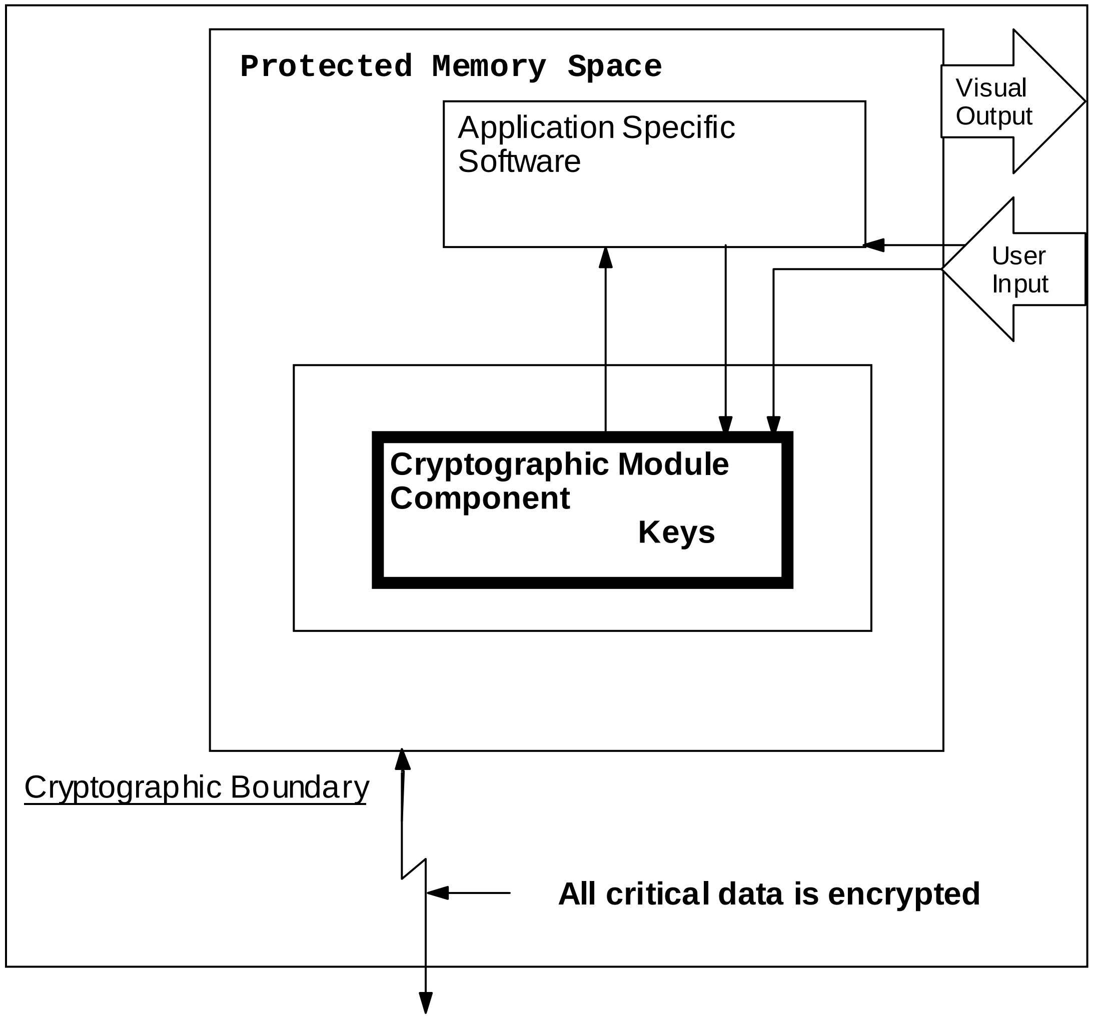

Protected Memory Space
Application Specific Software
Visual Output
User Input
Cryptographic Module Component
Keys
Cryptographic Boundary
All critical data is encrypted

Figure 2 Module Hardware and Software

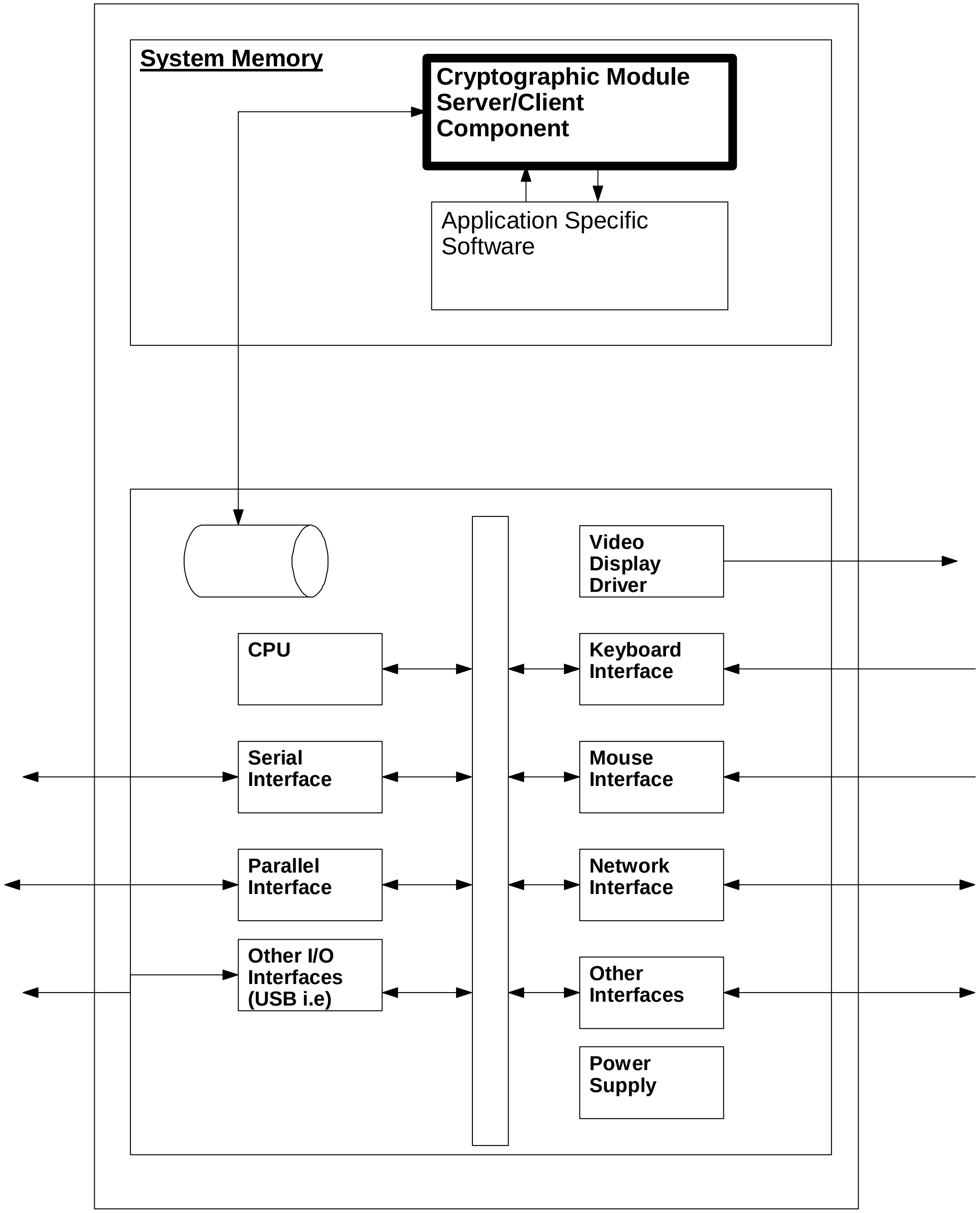
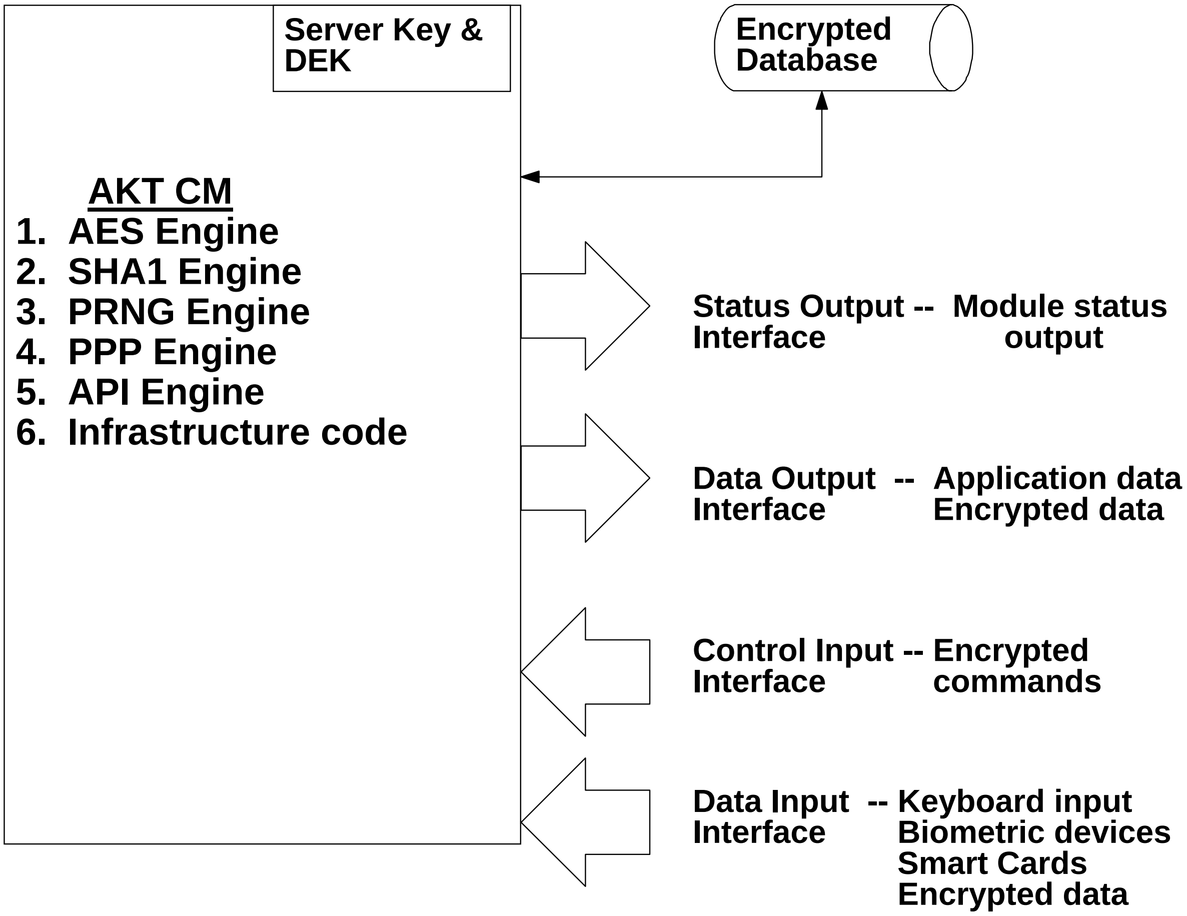


Figure 3 Functional Diagram



4. Roles and Services

The AKT Module supports the following two distinct operator roles:

- User Role
- Cryptographic Operator Role

The AKT Module will enforce the separation of roles using access control to the module. An operator must enter their individual characteristics to access the module. Upon correct entry of their user characteristics, the role is selected based on the identity of the operator. At the end of a session, the operator will be logged-out.

The cryptographic module is being evaluated at Level 1, therefore, no FIPS approved authentication requirements apply. The module's authentication strength varies according to the particular method chosen for authentication. An evaluation of the authentication strength was not conducted at this time, thus AKCode makes no particular claim as pertains to the FIPS 140-2 authentication requirements.

4.1. User Role

The User role shall provide all of the services necessary for the secure transport of data over an insecure network. These services are the following:

4.1.1. Enter User Name:

This service presents the operator with an input dialogue box wherein the operator can input their assigned user name that is used as a key tag in the server database, which is located outside of the logical boundary of the module.

4.1.2. Enter Password:

This service presents the operator with an input dialogue box wherein the operator can input their assigned password. The entry is plain text and is obscured by asterisks from the operator, as it is input.

4.1.3. Input Additional Biometrics Data:

The biometrics data is input to the module through devices that are configured to read an individual biometrics characteristic and subsequently to convert it into a unique digital format that can be transmitted to the module.

4.1.4. Encrypt Data:

This service utilizes the FIPS-197 AES function to perform encryption of the data.

4.1.5. Decrypt Data:

This service utilizes the FIPS-197 AES function to perform decryption of the data.

4.1.6. Transmit data outside the module:

This service allows data to be securely transmitted outside of the module.

4.1.7. Create User:

The user has the ability to create themselves as a new user. A user is created when a profile, a username and a password have been successfully stored in the secure user database, which is located outside of the logical boundary of the module, and the user activity has been initiated.

4.1.8. Show Status:

It is possible for the user to show the current status of the module by invoking the show status command. If an error state is processed the module returns an error code, represented as a 32-bit unsigned long, to the calling application via the status output interface when an error occurs. The application using the module determines how to communicate the error to the operator.

4.1.9. Perform Self Tests:

Self-testing the module can be performed to verify that the components of the module are functioning as per the specification. If an error state is processed, the same methodology is used that is used in 4.1.8 above.

4.2. Cryptographic Officer:

The CO role shall provide all of the services necessary for the secure transport of data over an insecure network. These services are as follows (some services are defined earlier and are the same):

4.2.1. Enter User Name

4.2.2. Enter Password

4.2.3. Input Additional Biometrics Data

4.2.4. Encrypt Data

4.2.5. Decrypt Data

4.2.6. Transmit data outside the module

4.2.7. Create User:

The CO has the ability to create new users. A user is created when a profile, a username and a password have been successfully stored in the secure user database, which is located outside of the logical boundary, and the user activity has been initiated.

4.2.8. Initiate User:

In some instances, it is possible for users to enroll themselves in the system. However, the user must obtain permission from the CO to utilize the system. The user remains in an inert state until the user is initiated.

4.2.9. Deactivate User:

If a user is removed from the system, the CO has the power to deactivate them. In this state, it is possible for the user profile to remain in the database, which is located outside of the module, for archival purposes, but the data would be in an inert state.

4.2.10. Spawn New CO:

The CO has the power to spawn other CO’s to share the duties.

4.2.11. Show Status:

See section 4.1.8 for a discussion of how the error state is processed.

4.2.12. Perform Self Test:

The role of the CO is created when the software is installed the first time in a trusted environment. Thereafter, the CO can elect to spawn other CO's, and/or create users or, have the system established so users are allowed to create their own profiles but the CO activates the user profiles.

5. Security Rules

This section documents the security rules enforced by the AKT Module to implement the security requirements of this FIPS 140-2 Level 1 module when used in conjunction with the specified hardware and software.

5.1. Distinct Operator Roles:

The cryptographic module shall provide two distinct operator roles. These are the User role and the CO role.

5.2. Module Access Control:

The AKT Module shall provide access to the module using a username and password, or additional unique user information. The cryptographic module is being evaluated at Level 1, therefore, no FIPS approved authentication requirements apply. The module's authentication strength varies according to the particular method chosen for authentication. An evaluation of the authentication strength was not conducted at this time, thus AKCODE makes no particular claim as pertains to the FIPS 140-2 identity or role based authentication requirements.

5.3. Keys:

All keys are output in encrypted form.

5.4. Data Encryption Key (DEK):

The AKT module uses the AES algorithm running in 256-bit mode. All data transmitted outside the AKT module is encrypted using one of the following Data Encryption Keys.

5.4.1 Enterprise/Application Server DEK

A DEK generated using the PRNG specified in FIPS 186-2 Section 3.1. These DEKs are utilized on the server to encrypt data transmitted to and from the secure user database, which is located outside of the logical boundary.

5.4.2 Operator DEK

A DEK generated using the PRNG specified in FIPS 186-2 Section 3.1. This DEK is used to transmit data outside of the logical boundary or to transmit data from one AKT module to another**.**

5.5. Key Generation:

All keys that are generated within the module are generated using the FIPS 186-2 appendix 3.1 PRNG, with the SHA-1 based 'G' function.

5.6. PRNG Continuous Test:

During operation a PRNG routine compares each 160-bit random string with the previously generated 160-bit random string. An error is returned via the status output interface and the module is locked if the two 160-bit values are identical.

5.7. Pre - Validation State:

When the AKT Module has not been placed in a valid role, the operator shall not have access to any cryptographic services.

5.8. Power Up:

Upon the application of power, or when commanded by the operator, the AKT Module shall perform the following tests:

- Integrity test:

  The module compares a HMAC-SHA-1 of its on-disk image against the HMAC-SHA-1 generated on the module when it was built by AKCode. The specific procedure is:

  Performed by AKCode:

  1) Build (compile) the AKT Module library.
  2) A post-build step runs GenEDC.exe on the module built in step 1.
  3) GenEDC calculates a HMAC-SHA-1 from the module and writes the result to a plaintext output file in the same directory as the module. This file is shipped with the module.

  When the module is loaded by an application:

  1) The module calculates a HMAC-SHA-1 from its image on disk.
  2) The module loads the HMAC-SHA-1 from the plaintext file and compares it to the calculated HMAC-SHA-1.
  3) An error is returned to the application via the status output interface and the module locks its interface if the HMAC-SHA-1s are not equal.

- SHA-1 Algorithm Known Answer Test (KAT)
- SHA-1 HMAC Known Answer Test
- AES Encryption and Decryption Algorithm KAT
- PRNG algorithm KAT – in this case, a known seed is passed to the PRNG and the result is compared internally to a known answer.
- Conditional continuous random number generator test comparing 160 bits
- PPP Algorithm KAT

Upon successful completion of the self-tests, the module indicates that it is in the “Ready” state. If any of the above tests fail then the AKT Module will not continue to function, any keys that have been generated will be zeroized and the error indicator, “"Power-up Self Test Failed." ” will be displayed on the Video Display Unit (VDU). To exit the error state, the module can be power cycled.

At any time the AKT Module is in an idle state, the operator, if logged in, will be capable of commanding the module to perform the power-up self-test.

5.9. Zeroized Transitions:

The AKT Module will have all cryptographic keys zeroized under the following conditions:

5.9.1. Upon Module Destruction:

Module destruction refers to the unloading of the library from memory.

5.9.2. Upon CO command: The command to zeroize the module is “ZeroizeCipherKey”.

5.9.3. Upon any Error Condition:

Any error that causes the Module to have to be shut down and unloaded will result in the Module being zeroized.

5.10. Module Status:

The AKT Module status can be determined from the status output display area on the VDU. In the majority of cases, the status will be normal, a state which will be indicated by a persistent message on the screen. If the status is abnormal, the message will be changed to indicate that a problem has arisen.

5.11. Concurrent Users:

The AKT Module is intended to be a single user module per instance and it does not support concurrent users of each instance.

5.12. Definition of Critical Security Parameters (CSP):

The following are critical security parameters contained in the AKT Module:

- Operator DEK
- Application Server DEK
- Enterprise Server DEK
- PRNG Seed Keys
- PRNG Internal State
- PPP Key Encryption Key (KEK)
- HMAC-SHA-1 Key

5.13. Error State:

When an error state is entered (due to a self test failure, for example) the following steps occur:

5.13.1. CSPs cleared:

The CSPs of the faulting module are cleared.

5.13.2. Set Fault Flag:

A ‘fault’ flag is set in the module.

5.13.3. Error Code:

An error code relating to the encountered fault is returned to the user via the status output interface.

5.13.4. Fault Flag:

All data input interface methods of the module check the fault flag before processing the data request. The error code is returned to the user via the status output interface if the fault flag is set.

5.13.5. Show Status:

The only method available when the fault flag is set is the ‘show status’ method.

5.13.6. Performing Self Tests:

The fault flag is set to ‘performing self test’ when the self-tests are in progress. This effectively inhibits the data input and output interface of the module.

5.13.7. Fault Flag Reset:

The only way to reset the fault flag is to reload the module (i.e. unload the module from memory and reload it).

6. Roles, Access Control and Services

6.1. Roles and Access Control:

Access control will take the form of one of three possible mechanisms:

6.1.1. User Based Access:

User based access, uses the password and/or the username.

6.1.2. Biometrics Based Access:

Biometrics access uses the biometric template.

6.1.3. Device Based Access:

Device access uses a Personal Authentication Device, such as a smart card, an RF device, or a USB device, in conjunction with the other access methods discussed earlier.

The following table summarizes the roles and access methodologies:

Table 6.1 Roles and Required Identification and Access to the Module

Role	Type of Access	Access Data
All Operators	Password	User password
All Operators	Biometrics	Biometrics sample
All Operators	Device	Ownership of device

The cryptographic module is being evaluated at Level 1, therefore, no FIPS approved authentication requirements apply. The module's authentication strength varies according to the particular method chosen for authentication. An evaluation of the authentication strength was not conducted at this time, thus AKCode makes no particular claim as pertains to the FIPS 140-2 identity or role based authentication requirements.

6.2. Roles and Services with Access Rights:

Referring to section 4.0 and 5.0 the Roles and Services can be summarized as follows (see section 6.3 for definitions):

Table 6.2(a) Roles, Services and Access Rights

Role		Service	CSP Modes of Access
C.O.	User		
X	X	Enter User Name	
X	X	Enter Password	
X	X	Input Additional Biometrics Data	
X	X	Encrypt Data	Use - Operator DEK Use - Application Server DEK Use - Enterprise Server DEK Use - PRNG Seed Key Use and Update - PRNG State
X	X	Decrypt Data	Use - Operator DEK Use - Application Server DEK Use - Enterprise Server DEK Use - PRNG Seed Key Use and Update - PRNG State
X	X	Transmit Data Outside the Module	Use - Operator DEK Use - Application Server DEK Use - Enterprise Server DEK
X	X	Create User	Generate - Operator DEK Generate - Application Server DEK Generate - Enterprise Server DEK Create - PRNG Seed Key Use and Update - PRNG State
X	X	Show Status	
X	X	Perform Self-Tests	
X		Initiate User	Generate - Operator DEK Generate - Application Server DEK Generate - Enterprise Server DEK Create - PRNG Seed Key Use and Update - PRNG State
X		Deactivate User	Zeroize - Operator DEK Zeroize - Application Server DEK Zeroize - Enterprise Server DEK
X		Spawn New CO	Generate - Operator DEK Generate - Application Server DEK Generate - Enterprise Server DEK Create - PRNG Seed Key Use and Update - PRNG State

6.3. Physical Security Mechanisms:

Physical security of the AKT Module is currently specified as production grade hardware. The AKT Module is a software component and as such is not dependent on the physical security of the device that houses the software. Never the less, the user is expected to take reasonable measures to protect the physical security. The AKT Module needs to be run in a secure computational environment and as much as possible, it should be run in a secure physical environment although, this latter requirement is not as important as the computational environment. It is intended that the AKT Module be run under a single user operating system environment. The physical device should be in a secure location so that it is protected under normal operating conditions.

6.4. Mitigation of Other Attacks:

The module has not been designed to mitigate any other attacks.

7. Appendix A

7.1. Definitions

AES	Advanced Encryption Standard
CO	Crypto Officer
CPP	C++ Programming Language
CSP	Critical Security Parameter
d	Session Data
DEK	Data Encryption Key
DLL	Dynamic-link Libraries
FIPS	Federal Information Processing Standard
KAT	Known Answer Test
KEK	Key Encryption Key (PPP Transport)
NIST	National Institute of Standards and Technology
OS	Operating System
p	Operator Password
PAD	Personal Authentication Device
PC	Personal Computer
PPP	AKCode Ping Pong Ping algorithm
PRNG	Pseudo Random Number Generator
SHA	Secure Hash Algorithm
AKCode	AKCode Inc.
USB	Universal Serial Bus
VDU	Video Display Unit