Unisys OS 2200 Cryptographic Library

Version 1R1

Unisys Corporation

Unisys OS 2200 Cryptographic Library FIPS 140-2 Security Policy

Version 1.08

Copyright Notice

Acknowledgements

The principal author of this document is:

James R. Heit
Consultant james.heit@Unisys.com
2470 Highcrest Road
Roseville, MN 55113

For further information contact

Mary Ann Bucher
Manager mary.bucher@Unisys.com
2470 Highcrest Road
Roseville, MN 55113

Judith Kruse
Director judith.kruse@Unisys.com
2470 Highcrest Road
Roseville, MN 55113

Unisys Cryptographic Library Module
FIPS 140-2 Level 1
Security Policy
Version 1.08

Version Table

Version #	Date	Author	Description
1.0	Dec. 1, 2009	Unisys	Initial CMVP Lab submission.
1.01	Jan. 5, 2010	Unisys	Revisions from initial CMVP Lab review.
1.02	Jan. 14, 2010	Unisys	Added table 3.1.1.
1.03	Jan. 21, 2010	Unisys	Revision from third CMVP Lab review.
1.04	Feb. 22, 2010	Unisys	Minor wording revisions.
1.05	Mar. 1, 2010	Unisys	Added CAVP Certificate Numbers.
1.06	Mar. 1, 2010	Unisys	CMVP requested corrections.
1.07	Mar. 8, 2010	Unisys	Add note signature gen/ver only approved with SHA-1.
1.08	June 8, 2010	Unisys	Requested CMVP corrections

Table of Figures

1. Introduction

This document specifies the non-proprietary Security Policy for the Unisys OS 2200 Cryptographic Library cryptographic module version 1R1. This Security Policy describes the compliance of Cryptographic Library (CryptoLib) with Federal Information Processing Standards Publication 140-2 (FIPS 140-2).

This document also specifies the required actions to use CryptoLib in a FIPS approved mode of operation. This document may be freely distributed in-whole and without modification.

1.1 Audience

This document is required as part of FIPS 140-2 validation. It describes how the Cryptographic Library Module meets the requirements of FIPS 140-2 validation.

1.2 References

This document deals only with operations and capabilities of the Module in the technical terms of a FIPS 140-2 Cryptographic Module Security Policy. More information is available on the Module from the following sources:

The Unisys website (http://www.unisys.com) contains information on the full line of products from Unisys.

The CMVP website (http://csrc.nist.gov/groups/STM/cmvp/) contains information on NIST and the cryptographic module validation program.

Please contact Unisys Corporation for access to proprietary product documentation for the Unisys OS 2200 Cryptographic Library.

1.3 Documents

FIPS 140-2 requires a submission package containing several documents, they include:

- Security Policy, this document
- Finite State Model
- Design Documents
- Block Diagram
- Source Listings
- Vendor Evidence

Please contact Unisys Corporation for access to proprietary product documentation for the Unisys OS 2200 Cryptographic Library.

2. Specification

2.1 Overview

CryptoLib is an OS2200 system software library product that has been validated to the FIPS 140-2 standard. Access to the library is provided through an Application Program Interface (API). The U.S. Government and some commercial customers require FIPS 140 validation of products that use cryptography.

2.2 Specification

This module is classified by the FIPS 140-2 standard as a multi-chip standalone cryptographic module. This module is validated to the following FIPS 140-2 levels.

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

Table 2.2.1 Security Level per FIPS 140-2 Sections

2.3 Boundary

This module is a software component only, so the logical cryptography boundary contains the software module that makes up the cryptographic module. No source code is provided to the Security Officer or User, only the binary object Module. The physical boundary includes the mainframe containing general purpose hardware including: the CPU, cache, RAM, disk drives, NICs, and other internal components of the system.

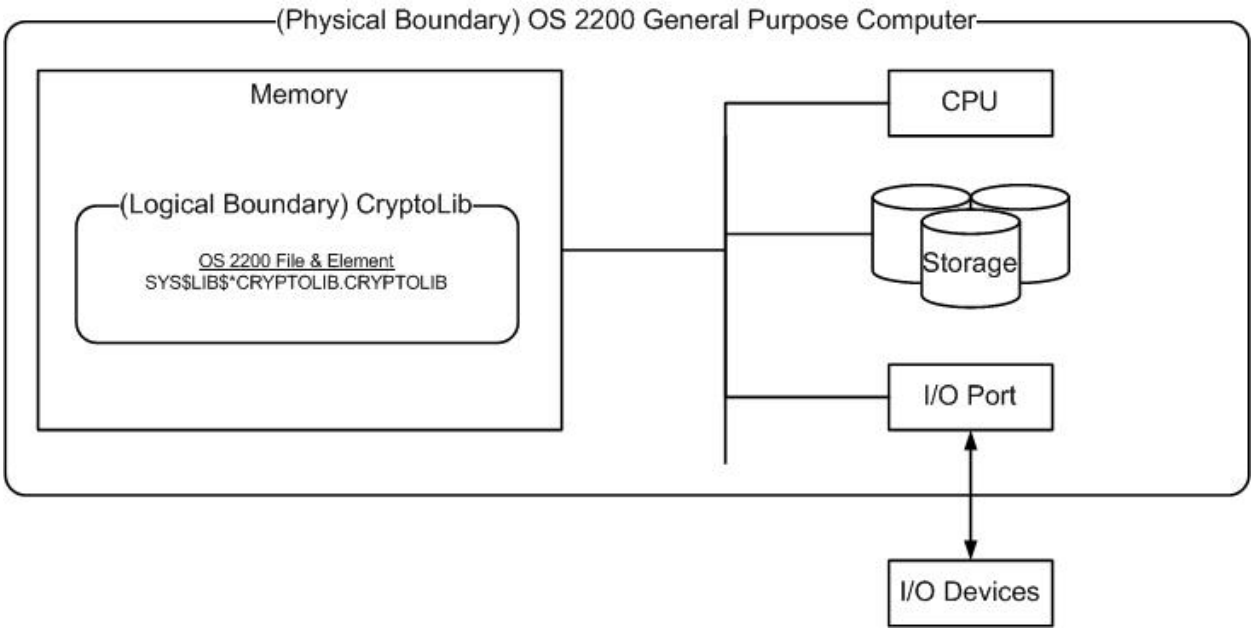

Figure 2.3.1 Logical Block Diagram

2.4 Operational Mode

When the module is uninitialized it is considered in non-FIPS mode and prevents any calls to cryptographic functions. Prior to the caller calling CL$init, any calls to cryptographic functions will return with an uninitialized status. The Module must be initialized by calling CL$init. The caller can specify whether they wish to run in FIPS Approved mode or non-FIPS Approved mode by a parameter on CL$init. The CL$init function will perform the Power Up tests, that is, the Known Answer Tests and the module integrity check. The module integrity check is done by using an RSA signature which was computed at build time. If the CL$init computed signature does not match the build signature that is distributed with the Module, the Module will not initialize and no cryptographic functions will be made available. If the Power Up tests fail, the Module will not initialize and no cryptographic functions will be made available and no data output will be output on the Data Output Interface. If the Power Up tests and Module integrity check complete successfully, cryptographic function calls become available to the User or Crypto-officer.

To summarize, to operate CryptoLib in a FIPS Approved mode of operation the following are required:

- The installed version of CryptoLib must be FIPS validated, and have no software corrections applied. Not all versions will be FIPS validated. The installation of CryptoLib is performed via SOLAR. The caller can also check a returned parameter on the CL$init call to see if the CryptoLib version in use is FIPS validated.
- The caller must specify that it wishes to run in FIPS Approved mode on the CL$init API call.
- The caller must use only approved cryptographic algorithms. See section 2.7.

2.5 Validated Platform

The FIPS 140-2 Lab tested this module on the following: UNISYS 2200 36 bit processor/ OS 2200 IOE (Integrated Operating Environment) 13.0

2.6 Ports and Interfaces

The module is a software component and utilizes Application Program Interface (API) as interfaces to the module. The module's API uses the four logical interfaces (Data Input, Data Output, Control Input, Status Output) defined by FIPS 140-2 in the following matter:

Data Input Interface
All data to all functions that is input to and processed by the Module, from the User or Crypto-officer enters via the Data Input Interface.

Data Output Interface
All functions output data (excluding statuses and return codes which are returned via the Status Output Interface) to the User or Crypto-officer via the Data Output Interface. Upon any Self-Test (Conditional or Power Up) failure the module enters an error state and the Data Output Interface is prohibited from output.

Control Input Interface
All input functions that are used to control the operation of the module enter via the Control Input Interface.

Status Output Interface
All functions provide status information back in statuses and return codes from the Module to the User or Crypto-Officer via the Status Output Interface. Some functions, such as CL$init, also provide output parameters that are defined for status output.

Power Interface
This Module is a software only cryptographic module and does not provide power or maintenance access interface beyond the power provided by the computer.

2.7 Approved Cryptographic Algorithms

The Module supports the following FIPS 140-2 algorithms in approved FIPS mode. Note: All Module algorithms, both approved and non-approved, are available for use. To run in FIPS approved mode, only FIPS approved algorithms should be used.

Algorithm	Type	Standard	Algorithm Mode and Use	Algorithm Certificate
AES (128,192,256)	Symmetric	FIPS 197	CBC,ECB encrypt/decrypt	1293
Triple-DES	Symmetric	FIPS 46-3	CBC, ECB encrypt/decrypt	910
DSA	Asymmetric	FIPS 186-2; PQG(gen) MOD(1024); PQG(ver) MOD(1024); KEYGEN(Y) MOD(1024); SIG(gen) MOD(1024); SIG(ver) MOD(1024);	Parameter generation and verification; key generation using a FIPS Approved RNG; signature	418

			generation and verification using SHA-1 only.	
RSA	Asymmetric	PKCS#1V1.5; SIG(gen); SIG(ver)	Signature generation and verification using SHA-1 only.	619
SHA (1,224,256,384,512)	Message digest	FIPS 180-3	Message integrity	1187
HMAC-SHA-1	HMAC	FIPS 198-1	Message integrity	753
RNG	Random number generator	FIPS 186-2 General Purpose x-Change Notice w/SHA-1;	Random number Generator	721

Table 2.7.1 FIPS 140-2 Approved Algorithms

2.8 Non-Approved Cryptographic Algorithms

The Module supports the following FIPS 140-2 algorithms in non-approved FIPS mode.

Algorithm	Type	Standard	Algorithm Mode and Use
DES (56)	Symmetric	FIPS 46-3	CBC, ECB encrypt/decrypt
MD2	Message digest	RFC1115	Message integrity
MD5	Message digest	RFC1321	Message integrity
HMAC-MD5	HMAC	RFC2104	Message integrity
RC4	Symmetric		Encrypt/decrypt
Diffie-Hellman[1]	Key agreement		
Non-approved RNG	Random number generator		Random number Generator

Table 2.8.1 FIPS 140-2 Non-Approved Algorithms

3. Roles, Services, and Authentication

3.1 Roles and Services

There are two roles supported by Cryptographic Library, Crypto Officer and User, as defined in the FIPS 140-2 standard. The Crypto Officer and User are defined as any entity that can access the services provided by the module and each role is implicitly assumed based on the service being executed. There are no restrictions on this access. The Crypto Officer role may perform the install and uninstall of the module on the host system. The User role has access to load the module and call any API functions provided by the module.

[1] The module only provides Diffie-Hellman primitives, which a calling application can use in a FIPS approved mode of operation as part of an allowed key establishment scheme.

Service	Role	Approved/Non-Approved
General Services		
Installation	Crypto Officer	Approved
Initialization	User	Approved
Self-Tests	User	Approved
Show status	User	Approved
Uninstall	Crypto Officer	Approved
Diffie Hellman (DH)		
Generation of key pair	User	Non-Approved
Generation of shared secret	User	Non-Approved
Digital Signature (RSA &DSA)		
DSA Key generation	User	Approved
DSA Signature	User	Approved
DSA Verification	User	Approved
RSA Key generation	User	Non-Approved
RSA Signature	User	Approved
RSA Verification	User	Approved
Digest Algorithms and Message Authentication (SHA, HMAC)		
MD2	User	Non-Approved
MD5	User	Non-Approved
MD5 HMAC	User	Non-Approved
SHA 1 Digest	User	Approved
SHA 224 Digest	User	Approved
SHA 256 Digest	User	Approved
SHA 384 Digest	User	Approved
SHA 512 Digest	User	Approved
SHA 1 HMAC	User	Approved
Non-Approved Random Number Generation		
Non-Approved RNG Seeding	User	Non-Approved
Non-Approved RNG Random number request	User	Non-Approved
Random Number Generation - RNG FIPS 186-2 General Purpose x-Change Notice w/ SHA-1		
RNG Seeding	User	Approved
RNG Random number request	User	Approved
Symmetric Encryption		
AES Decryption	User	Approved
AES Encryption	User	Approved
DES Decryption	User	Non-Approved
DES Encryption	User	Non-Approved
RC4 Decryption	User	Non-Approved
RC4 Encryption	User	Non-Approved
Triple DES Decryption	User	Approved
Triple DES Encryption	User	Approved
Zeroization		
AES Zeroization	User	Approved

DES Zeroization	User	Non-Approved
Triple DES Zeroization	User	Approved
DSA Zeroization	User	Approved
MD5 HMAC Zeroization	User	Non-Approved
RC4 Zeroization	User	Non-Approved
RSA Zeroization	User	Approved
SHA 1 HMAC Zeroization	User	Approved

Table 3.1.1 Roles and Services

3.2 Authentication

This Module does not support any authentication or identification services to determine the user.

4. Physical Security

This module is a software library solution, and thus claims no physical security.

5. Cryptographic Key Management

5.1 Key Generation

This module provides cryptographic functions for key generation. These APIs are called by applications that reside outside the cryptographic boundary. All asymmetric keys can be created for PKI, digital signing, and encryption/decryption. All keys are generated by using the approved FIPS 186-2 General Purpose [(x-Change Notice); (SHA-1)] Random Number Generator. The module does not implement a FIPS-approved RSA key generation method; however, it does make use of the approved RNG for key generation.

5.2 Key Agreement

This module provides RSA encrypt/decrypt and Diffie-Hellman primitives, which calling applications can use to implement approved/allowed key establishment methods.

5.3 Key Storage, Entry, and Output

This module does not store any critical security parameters (CSPs) in persistent state media. All CSPs generated or passed to the Module remain in the User's (calling application) memory. The User must utilize the API's correctly to guarantee FIPS 140-2 compliance.

5.4 Key Zeroization

This module does not store any CSPs and it is the User's responsibility to ensure all CSPs are deleted in a way that will make them unavailable. This Module provides functions to overwrite memory that contains keying material with zeroes. Once overwritten, the keying material will become unavailable. It is the User's responsibility to ensure the correct API is called to overwrite the keying material.

6. Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC)

This module runs on hardware that meets the applicable EMI/EMC requirements for FIPS 140-2 specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class A (i.e., for business use).

7. Self-Tests

7.1 Power Up Tests

Power Up tests, also known as Known Answer Tests (KATs), are tests where a cryptographic value is calculated and compared against a result that was previously calculated and stored. KATs are not callable by Users and thus provide the User no input/output. Before any User may call into the Module they must call the initialization function CL$init which performs the KATs for approved functions. If any KAT fails, the Module will prevent any cryptographic calls from being performed.

Algorithm	Power Up Self Test
AES	Encrypt/Decrypt KAT
Triple-DES	Encrypt/Decrypt KAT
DSA	Sign/Verify Test
RSA	Sign/Verify Test
SHA	SHA-256 KAT SHA-512 KAT
HMAC	HMAC-SHA-1 KAT
RNG	FIPS 186-2 General Purpose x-Change Notice KAT
Software Integrity	RSA signature verification

Table 6.1.1 Approved Cryptographic Power Up Tests

7.2 Conditional Self-Tests

Conditional self-tests are executed implicitly when they are necessary. This module implements two types of conditional self tests as required by FIPS 140-2 Level 1 requirements, pair-wise consistency self-tests and continuous random number generator tests (CRNG).

Pair-wise Consistency Tests
Whenever the module creates an asymmetric public/private key pair for use by RSA or DSA, a pair-wise consistency test is performed. The module will perform a sign/verify operation on each key pair generated to ensure the key generation is functioning properly. If this operation fails, an error is reported and the key pair is discarded.

Algorithm	Conditional Self Test
RSA	Pairwise Consistency self test
DSA	Pairwise Consistency self test

Table 6.2.1 Pair-wise Consistency Tests

Continuous Random number generator (CRNG)

The module implements two random number generators. A non-approved random number generator (RNG), uses API calls to the OS to gather random data. The approved RNG is seeded by using the non-approved random number generator. The approved RNG is implemented based on FIPS 186-2 General Purpose x-Change Notice with the SHA-1 algorithm. This test is performed for both RNG's and upon any failure of the test an error is reported and the generated number is thrown away.

Algorithm	Conditional Self Test
Non-approved RNG	CRNG test FIPS 140-2 Section 4.9.2
RNG	CRNG test FIPS 140-2 Section 4.9.2

Table 6.2.2 CRNG test FIPS 140-2 Section 4.9.2

The following additional Conditional Self-Tests are not applicable to this module:

Bypass Conditional Self-Test (Not Applicable)
This module does not support a bypass capability.

Firmware Load Conditional Self-Test (Not Applicable)
This module does not reference any externally cryptographic modules or devices.

Manual Key Entry Conditional Self-Test (Not Applicable)
This module does not allow keys to be manually entered.

7.3 Critical Function Tests

This module does not implement any critical function tests for FIPS 140-2 Level 1.

8. Design Assurance

Unisys manages and maintains source code and associated User documentation using the PRIMUS source control system. PRIMUS is also used for product build management, and tracking which versions of the files are used in each release.

9. Mitigation of Attacks

This module has no prevention against specific attacks made on the module.

Appendix A : Glossary

AES	Advanced Encryption Standard
API	Application Programming Interface
CBC	Cipher Block Chaining
CMVP	Cryptographic Module Validation Program
CryptoLib	Unisys OS 2200 Cryptographic Library
CSP	Critical Security Parameter
DES	Digital Encryption Standard
DH	Diffie-Hellman
DSA	Digital Signature Algorithm
ECB	Electronic Code Book
EMC	Electro Magnetic Compatibility
EMI	Electro Magnetic Interference
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standard
HMAC	Hashed MAC
KAT	Known Answer Test
MAC	Message Authentication Code
MD2	Message Digest Algorithm 2
MD5	Message Digest Algorithm 5
NIST	National Institute of Standards and Technology
RSA	Rivest-Shamir-Adleman
RNG	Random Number Generator
SHA	Secure Hash Algorithm