Security Policy

EFJ Communication Cryptographic Library (CCL)

Author: Angel G. Ortiz

Software Version: 2.0

Document Version 2.1
February 24, 2005

Contents

1 Introduction

1.1 Scope

This document is a FIPS 140-2 Security Policy for the E.F. Johnson Communication Cryptographic Library™ (CCL), version 2.0 cryptographic module. The CCL is a Level 1 software cryptographic module. In terms of the FIPS 140-2 standard, the CCL is a multi-chip standalone FIPS 140-2 module.

The module's logical boundary consists of a Windows™ 2003/XP/2000 or Pocket PC 2003 DLL, and an Application Programming Interface to access the FIPS 140-2 security functions within the CCL DLL.

This DLL can be used in any Windows™ 2003/XP/2000 or Pocket PC 2003 operating system application that requires FIPS 140-2 security functionality.

1.2 CCL Implementation

The CCL is implemented as a multi-chip standalone module meeting Level 1 requirements of the FIPS 140-2 standard. The CCL is a DLL implemented using the C programming language. Application developers wishing to use the CCL can use the CCL's Application Programming Interface (API) to perform AES, DES[1], DSA, HMAC, PRNG, and SHA-1 security related functions.

The services of the CCL module are accessible through the CCL's API. There are two CCL versions. One is for the Windows™ 2003/XP/2000 operating systems. The other is for the Microsoft™ Pocket PC 2003 operating system. The product number and version are:

1. Pocket PC 2003 CCL version 2.0 ~ Product Number 039-5733-200
2. Microsoft® Windows 2003/XP/2000 CCL version 2.0 ~ Product Number 039-5734-200

1.3 Cryptographic Boundary

The cryptographic boundary for the CCL consists of the Personal Computer or Personal Data Assistant (PDA) in which the CCL module is installed. The software logical boundary surrounds the CCL DLL.

The CCL module runs as a dynamically linked export library (DLL) under the Windows™ 2000/2003/XP and Pocket PC 2003 operating systems.

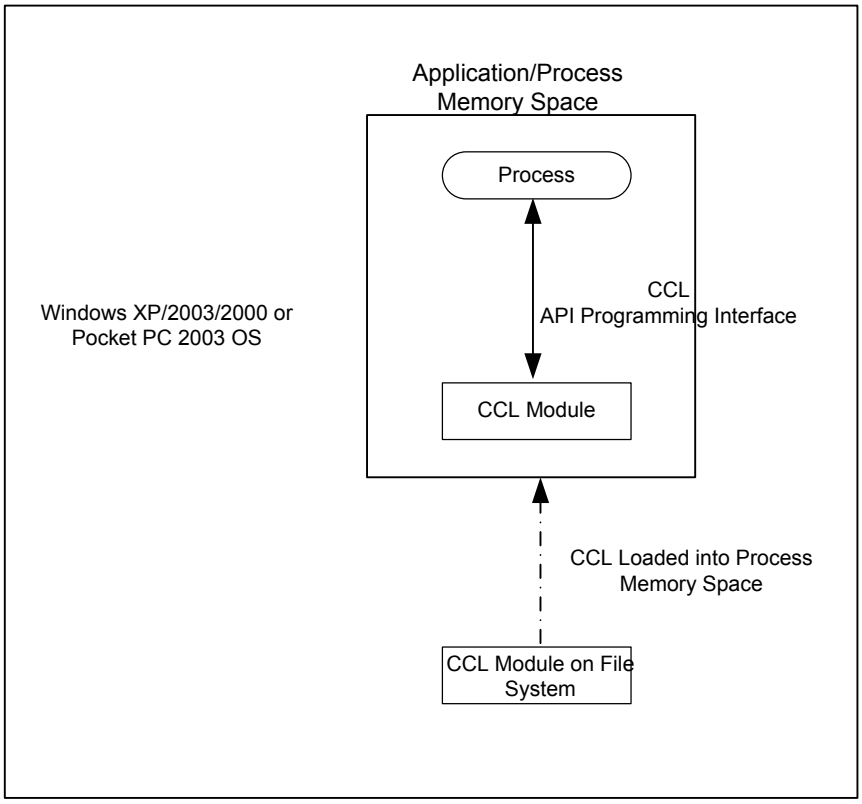
All applications running on the Windows™ 2000/2003/XP and Pocket PC 2003 operating systems run as processes. Every process running on the operating

[1] Please see rule 8 in section 4.2 for rules governing the module's use of DES.

system will execute in its own separate memory space. When a process which makes use of the CCL is executed, the operating system maps the CCL's DLL code into the process' memory space. Multiple applications can load, and make use of the CCL DLL at the same time. Each process will have its own copy of the CCL loaded into its own memory space. Any data passed between the process and the CCL is specific to that process and never leaves the process' memory space.

Below is the CCL's block diagram as described above.

Figure 1-1 CCL Block Diagram



The CCL module was tested on the following computing platforms:

Platform 1
Hardware: IBM Compatible PC
Processor: Intel P4 1.6 GHz
Operating System: Windows XP Service Pack 1

Platform 2
Hardware: Hewlett Packard iPAQ h5555 Personal Data Assistant
Processor: Intel PXA255
Operating System: Pocket PC 2003

2 Intended FIPS 140-2 Security Levels

The CCL is validated to meet FIPS 140-2 security requirements for the levels shown in the Table 1. The overall module is validated for Security Level 1.

Table 1.3-1 CCL Security Levels

Area	**FIPS 140-2 Intended Security Level**
Cryptographic Module Specification	Level 1
Cryptographic Module Ports and Interfaces	Level 1
Roles, Services, and Authentication	Level 1
Finite State Model	Level 1
Physical Security	N/A
Operational Environment	Level 1
Cryptographic Key Management	Level 1
EMI/EMC	Level 1
Self Tests	Level 1
Design Assurance	Level 1
Mitigation of Other Attacks	N/A

3 FIPS 140-2 Approved Operational Modes

The CCL provides FIPS 140-2 security functions via an API interface.

The following security functions are available to the user when the CCL is operating in a FIPS 140-2 approved mode of operation.

1. AES-128 OFB
2. AES-128 ECB
3. AES-128 CBC
4. AES-192 OFB
5. AES-192 ECB
6. AES-192 CBC
7. AES-256 OFB
8. AES-256 ECB
9. AES-256 CBC
10. DES[2] OFB
11. DES[2] ECB
12. DES[2] CBC
13. DES[2]CFB64
14. DSA Signature Generation
15. DSA Signature Verification
16. HMAC-SHA-1
17. PRNG

Note: The CCL contains two PRNGs. One PRNG is specific to generating random numbers for the DSA algorithm, and the other PRNG is for general purpose random number generation.

The CCL can be operated in a non FIPS 140-2 mode by using only the following ciphers:

1. SecureNet DES 1 bit CFB with differential encoding and decoding
2. DES[2] 1 bit CFB
3. DES 8 bit CFB
4. DES 8 bit OFB

[2] Please see rule 8 in section 4.2 for rules governing the module's use of DES.

4 Security Rules

The CCL meets the requirements of a multi-chip standalone module. Since the CCL is a software module, the module interfaces are defined in terms of the API performing FIPS 140-2 security functions.

4.1 Operating Environment

The CCL requires the Microsoft® Windows 2000/2003/XP or Pocket PC 2003 operating systems. The Pocket PC 2003 operating system is inherently restricted to a single operator mode of operation. For the Windows NT based systems (Windows 2000/2003/XP), the operator must disable all non-administrator accounts and all network services in order to restrict the operating system to a single operator mode of operation.

The CCL was tested under the Pocket PC 2003 operating system using the HP iPAQ h5555 PDA and the Windows XP service pack 1 operating system.

4.2 FIPS 140-2 Related Security Rules

1. The CCL has the following interfaces:

   The CCL's physical interfaces consist of those found on a Personal Computer or PDA. These physical interfaces are the PDA's or computer's hardware such as keyboard, mouse, hard drive, PDA keypad, serial and USB ports. The CCL's logical interface is provided through the CCL's Application Programming Interface.

   - Data Input Interface: The Data Input Interface is the parameters of the API function calls defined as input.
   - Data Output Interface: The Data Output Interface is the parameters of the API function calls defined as output.
   - Control Input Interface: Control inputs are the function calls.
   - Status Output Interface: The return values of the CCL APIs are the status interface for the CCL.
   - Power Interface: The Battery and Charging system of the Personal Computer or Personal Data Assistant is the power port.

2. All data output via the CCL Data Output Interface is disabled when an error state exists.

3. The CCL supports a User role and a Crypto Officer role. The role is selected implicitly by the service that is invoked.

4. The CCL supports the following FIPS approved algorithms.

- AES-128 OFB
- AES-128 ECB
- AES-128 CBC
- AES-192 OFB
- AES-192 ECB
- AES-192 CBC
- AES-256 OFB
- AES-256 ECB
- AES-256 CBC
- DES OFB
- DES ECB
- DES CFB64
- DES CBC
- SHA-1 hash function algorithm
- Digital Signature Algorithm (DSA)
- HMAC-SHA-1 ~ Keyed-Hash Message Authentication Code (HMAC) Using SHA-1.
- FIPS 186-2 Appendix 3.1 Pseudo Random Number Generator (PRNG)

  The CCL's PRNG must be seeded with a user provided seed. Note that the input must have at least 128 bits of entropy. Irrespective of the size of the seed provided, all generated keys will have a minimum of 128 bits of entropy and a maximum of 160 bits of entropy.

5. The CCL performs the following self-tests on power-up:

   - Software Integrity Test
     All CCL software releases are digitally signed using the DSA algorithm at the E.F. Johnson Co. facility. During self-tests, the CCL verifies the integrity of the loaded software using the DSA algorithm.

   - DES Algorithm Test
     The DES algorithm is tested for encrypt and decrypt using a Known Answer Test in the Electronic Code Book (ECB) mode of operation.

   - AES Algorithm Test
     The AES algorithm is tested for encrypt and decrypt using a Known Answer Test in the Electronic Code Book (ECB) mode of operation.

   - SHA-1 Algorithm Test
     The SHA-1 hash algorithm is tested using a Known Answer Test.

   - HMAC-SHA-1 Algorithm Test
     The HMAC-SHA-1 algorithm is tested using a Known Answer Test.

- DSA Algorithm Test
  The DSA Algorithm test is a DSA signature generation and signature verification test using a Known Answer Test (KAT).

- Pseudorandom Number Generator Test
  Both variants of the CCL's Pseudorandom Number Generators (PRNGs) are tested using their own Known Answer Test.

6. The CCL enters an error state upon failure of any of the self-test routines.

7. The CCL outputs a successful status indicator via the Status Indicator interface only when all tests have passed. The status indicator is that the module is properly loaded. The module does not perform any cryptographic functions while in an error state.

8. The CCL supports DES only for support of pre-existing legacy systems.

5 Self-Testing

In addition to the CCL Power-up Self tests that the CCL performs at power-up, the CCL implements additional tests to verify the proper functioning of the CCL module. The additional self-tests are:

1. On-Demand Self-Testing
2. Conditional Self-Testing

5.1 On-Demand Self-Testing

The CCL module exports an API function called ***DLL_PostPowerUpSelfTest(void)*** which is used to initiate all the power-up self tests of the CCL module.

5.2 Conditional Self-Testing

The CCL's PRNG implements continuous conditional tests that compare a newly generated 160 bit block of random data with the previously generated 160 bit block of random data. If the two blocks are equal, the CCL enters an error state. In this error state, the CCL module will not perform any cryptographic operations.

6 Identification and Authentication Policy

The CCL does not support authentication for either the User or Crypto Officer Roles.

7 Access Control Policy

7.1 Roles Supported

The CCL cryptographic module supports the User and Crypto-Officer role only. There is no Maintenance role in the CCL.

The User and Crypto-Officer roles are mutually exclusive and cannot both be active concurrently.

7.1.1 User Role

This role is implicitly assumed when an operator uses one of the services of the CCL module.

7.1.2 Crypto-Officer Role

This role is implicitly assumed when an operator uses one of the Crypto-Officer services.

The CCL DLL will perform a Digital Signature verification of itself when the underlying operating system attempts to load the module into the address space of an application.

7.2 Services Provided

The services provided by the CCL are the actual security functions that will be made available to the User via API calls.

The CCL APIs made available to the user are the interfaces.

The table below lists all the security services and functions that are performed by the CCL. The operator using the CCL service is also listed in the table.

Table 7.2-1 CCL Services vs. Security Functions

Service	Security Function(s) Used	Key Type and Length	General Mode of Operation	Operator Using Service.
Generate Random Key	PRNG	256 bit AES 192 bit AES 128 bit AES or 56 bit DES	Key Generation	Crypto-Officer
AES Encryption	AES	256 bit AES 192 bit AES 128 bit AES	Encryption	User
AES Decryption	AES	256 bit AES 192 bit AES	Decryption	User

Service	Security Function(s) Used	Key Type and Length	General Mode of Operation	Operator Using Service.
		128 bit AES		
AES Key Wrapping Encryption	AES	256 bit AES 192 bit AES 128 bit AES	Encryption	User
AES Key Unwrapping Decryption	AES	256 bit AES 192 bit AES 128 bit AES	Decryption	User
DES Encryption	DES	56 bit DES	Encryption	User
DES Decryption	DES	56 bit DES	Decryption	User
DSA Signature Generation	DSA	1024 bits	DSA Signature Generation	User
DSA Signature Verification	DSA	1024 bits	DSA Signature verification	User
Self Tests	AES, DES, DSA, HMAC, PRNG, SHA-1	Security Function Specific	Service Specific	Crypto-Officer
Show Status	N/A	N/A	CCL operation State	User
SHA-1	SHA-1	SHA-1 160 bit	Hashing	User
HMAC-SHA-1	HMAC-SHA-1	HMAC 160 bit	MAC Generation	User
Zeroize Keys	N/A	N/A	Clear Keys	Crypto-Officer

7.2.1 Generate Random Key

This service uses the CCL PRNG to generate a random key. This service makes use of the module's general purpose FIPS 186-2 PRNG.

7.2.2 AES Key Wrapping Encryption

All keys that are stored outside of the CCL boundary can be stored in encrypted format using the AES Key Wrapping service. A 256, 192, or 128 bit AES key can be used.

This service is provided by the CCL's FIPS approved AES.

7.2.3 AES Key Unwrapping Decryption

All keys that are stored outside of the CCL boundary can be retrieved and decrypted using the AES Key Unwrapping service. A 256, 192, or 128 bit AES key can be used.

This service is provided by the CCL's FIPS approved AES.

7.2.4 AES Encryption

This service is provided by the CCL's FIPS approved AES algorithm to perform encryption.

7.2.5 AES Decryption

This service is provided by the CCL's FIPS approved AES algorithm to perform decryption.

7.2.6 DES Encryption

This service is provided by the CCL's FIPS approved DES algorithm to perform encryption.

7.2.7 DES Decryption

This service is provided by the CCL's FIPS approved DES algorithm to perform decryption.

7.2.8 DSA Signature Generation

This service is provided by the CCL's FIPS approved DSA algorithm to perform DSA Signature generation.

7.2.9 DSA Signature Verification

This service is provided by the CCL's FIPS approved DSA algorithm to perform DSA Signature verification.

7.2.10 HMAC-SHA-1

This service is provided by the CCL's FIPS approved SHA-1/HMAC algorithm to perform a MAC given an encryption key.

7.2.11 SHA-1

This service is provided by the CCL's FIPS approved SHA-1 algorithm to generate a 160 bit message digest.

7.2.12 Self Test

This service provides power up and continuous tests to verify the secure state and operation of the CCL. All of the CCL's cryptographic and security functions are tested using known answer tests. The user initiates this service by power cycling or resetting the module.

7.2.13 Show Status

This service provides information on the CCL state such as the Fatal Error State.

7.2.14 Zeroize Keys

This service zeroizes all CSPs present in the module.

7.3 Access Rights within Services

An operator requiring a service within any role can read and/or write cryptographic keys and Critical Security Parameters (CSP) only through the invocation of the CCL module security service. Access to the module's security services is via the CCL's APIs.

The services within each role can only access the cryptographic keys and CSP that the service's API specifies.

The definition of Read/Write access for the CCL module is defined as follows:

- For Read/Write, the module both reads and writes to the internally stored cryptographic keys or CSPs.
- For Read access, the module will only Read the internally stored cryptographic keys or CSPs.
- For Delete access, the module will only delete a cryptographic key or CSPs.

Table 7.3-1 CCL Access Rights of CSPs

Service	Cryptographic Keys And CSPs	Type of Access (e.g. Read, Write, Delete)
Generate Random Key	PRNG State PRNG Seed	Read/Write
AES Encryption	AES Encryption Key	Read/Write
AES Decryption	AES Encryption Key	Read/Write
AES Key Wrap Encryption	AES Encryption Key	Read/Write
AES Key Unwrap Decryption	AES Encryption Key	Read/Write
DES Encryption	DES Encryption Key	Read/Write
DES Decryption	DES Encryption Key	Read/Write
DSA Signature Generation	DSA Asymmetric Key Pair	Read/Write
DSA Signature Verification	DSA Asymmetric Key Pair	Read/Write
HMAC-SHA-1	160 bit HMAC Key	Read/Write
SHA-1	SHA-1 160 bit message digest.	Read/Write
Power-up Self Test	None	NA
Show Status	AES, DES, DSA Key Pair, HMAC Key, SHA-1 message digest	Read
Zeroize Keys	PRNG State	Delete

Table 7.3-2 CSP Description

CSP Identifier	Description
AES Encryption Key	A 256, 192, or 128 bit key used to encrypt and decrypt.
DES Encryption Key	A 56 bit key used to encrypt and decrypt.
DSA Private Key	DSA private key used to generate/verify a 1024 bits digital signature.
160 bit HMAC Key	A 160 bit key used with SHA-1 to authenticate messages.

8 Key Management

The CCL module does not provide long-term key storage. All keys generated or processed by the CCL module reside in the application space in which the CCL DLL module has been loaded.

8.1 Key Generation

The CCL provides two services that are used to generate a random key via a Pseudo Random Number Generator (PRNG).

The CCL's implements two PRNGs which are used differently. One version of the PRNG is used for the generation of keys for symmetric ciphers and HMAC-SHA-1. The other version is specifically designed for the generation of keys for the DSA algorithm.

8.2 Key Distribution and Storage

The CCL module does not provide long-term key storage or any protocol such as Diffie-Hellman for key agreement or distribution.

The CCL module does support the import and export of keys from outside the cryptographic boundary. All key used by the CCL module reside within the module's cryptographic boundary in the application space of the application which loaded the CCL module.

9 Physical Security Policy

The CCL is a cryptographic module that is implemented completely in software such that the physical security is provided solely by the host platform. Therefore, the Physical Security section of FIPS 140-2 is not applicable.

10 Mitigation of Other Attacks Policy

The CCL module is not designed for the mitigation of any attacks outside the scope of FIPS 140-2.

11 CCL API Functions

The CCL's API functions are specified in the *EFJ Communication Cryptographic Library (CCL) SDK Manual, version 2.0*.

12 References

The following standards and documents where used in the development of the CCL module.

1. FIPS 140-2: Security Requirements For Cryptographic Modules
2. FIPS 180-1: Secure Hash Standard
3. FIPS 197: Advanced Encryption Standard (AES)
4. AES Key Wrapping Specification from NIST (November 16, 2001)
5. FIPS 198: The Keyed-Hash Message Authentication Code (HMAC)
6. FIPS 46-4: Data Encryption Standard (DES)
7. FIPS 186-2: Digital Signature Standard (DSS)
8. SP 800-38a: Recommendation for Block Cipher Modes of Operation
9. FIPS 81: DES Modes of Operation

13 Acronym List

AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
CCL	Communication Cryptographic Library
CFB	Cipher-Feedback
CSP	Critical Security Parameter
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DSP	Digital Signal Processor
ECB	Electronic Codebook
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standards
HMAC	Keyed-Hashing for Message Authentication Code
OFB	Output-Feedback
OTAR	Over-The-Air-Rekeying
PC	Personal Computer
PDA	Personal Data Assistant
PRNG	Pseudo Random Number Generator
ROM	Read Only Memory
RAM	Random Access Memory
SHA-1	Secure Hash Algorithm-1