Suite B Cryptographic Module v2.3.1

FIPS 140-2 Security Policy

Revision: 1.0

Prepared by: KEYW Corporation
7740 Milestone Parkway, Suite 500
Hanover, MD 21076
443-733-1600 *Phone*
443-733-1601 *Fax*

Contents

Revision History 4

Abbreviations 5

1. Introduction 6
   1.1. Identification 6
   1.2. Overview 6
   1.3. FIPS 140-2 Security Levels 8
2. Cryptographic Module Specification 9
   2.1. Security Functions 9
   2.2. Modes of Operation 9
   2.3. Cryptographic Boundary 10
   2.4. Determining Module Version 10
3. Cryptographic Module Ports and Interfaces 11
4. Roles, Services, and Authentication 13
   4.1. Roles 13
   4.2. Services 14
   4.3. Authentication 16
5. Physical Security 17
6. Cryptographic Keys and Critical Security Parameters 18
   6.1. Key Zeroization 20
7. Self-Tests 21
   7.1. Invoking Self-Tests 23
   7.2. Self-Tests Results 23
8. Mitigation of Other Attacks 24
9. Referenced Documents 25

Tables and Figures

Figure 1 – Module Message Encryption/Decryption Flow 6
Table 1 – Summary of Achieved FIPS 140-2 Security Levels 8
Table 2 – FIPS-Approved Security Functions 9
Figure 2 – Module Cryptographic Boundary 10
Table 3 – Module Ports and Interfaces 11
Figure 3 – Module I/O 12
Figure 4 – Module Cryptographic Boundary I/O 12
Table 4 – Module Services for Cryptographic Officer Role 14
Table 5 – Module Services for User Role 15
Table 6 – Module Authentication 16
Table 7 – Module Cryptographic Keys and Critical Security Parameters 19
Table 8 – Module Self-Tests 22
Table 9 – Module Self-Test Error Codes 23

Revision History

Revision	Date	Author	Changes
1.0	July 11, 2014	R. Glenn D. Mackie C. Constantinescu D. Wolff E. Hufford	Initial Release

Abbreviations

AAD	Additional Authentication Data
AC	Alternating Current
AES	Advanced Encryption Standard
API	Application Programming Interface
BAS	BlackBerry Administration Service
BES	BlackBerry Enterprise Server
BIN	Binary
CA	Certification Authority
CAVP	Cryptographic Algorithm Validation Program
CSP	Critical Security Parameters
CVL	Component Validation List
DEP	Default Entry Point
DLL	Dynamic Link Library
EC	Elliptic Curve
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standard
GCM	Galois/Counter Mode
HMAC	Keyed-hash Message Authentication Code
HRNG	Hardware Random Number Generator
I/O	Input/Output
IV	Initialization Vector
KAM	Key Agreement Manager
KAS	Key Agreement Scheme
KASVS	Key Agreement Schemes Validation System
KAT	Known Answer Test
KDF	Key Derivation Function
KEK	Key Encryption Key
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MDS	Mobile Data System
MS	Mobile Set
NIST	National Institute of Standards and Technology
OS	Operating System
PKI	Public Key Infrastructure
PKV	Public Key Validation
PT	Plaintext
RBG	Random Bit Generator
S/MIME	Secure/Multipurpose Internet Mail Extensions
SHA	Secure Hash Algorithm
SSL	Secure Sockets Layer
TLS	Transport Layer Security
µSC	Micro Smart Card
USB	Universal Serial Bus
USSOCOM	United States Special Operations Command
VS	Validation Specification
XTS	**X**EX **T**weakable Block Cipher with Ciphertext **S**tealing

1. Introduction

1.1. Identification

The following information identifies this document:

- Title: Suite B Cryptographic Module FIPS 140-2 Security Policy
- Version: 2.3.1

1.2. Overview

KEYW, in coordination with the United States Special Operations Command (USSOCOM), has developed a Suite B-compliant, standards based, Federal Information Processing Standard (FIPS) 140-2 Level 1 certified Cryptographic Library that is utilized by the Suite B Cryptographic Module. The Suite B Cryptographic Module implements an AES/GCM-256 layer of encrypted communications between a BlackBerry Enterprise Server (BES) and a BlackBerry Mobile Set (MS) with Elliptic Curve (EC) key exchange used to negotiate symmetric keys, which is initiated by a Key Agreement Manager (KAM).

An "in-band" Elliptic Curve Diffie-Hellman (ECDH) Key Agreement Scheme (KAS) implementing the Full Unified Model, C(2, 2, ECC CDH) as described in National Institute of Standards and Technology (NIST) publication SP 800-56A (Reference [1]) is used by the Suite B Cryptographic Module as it provides an optimal encryption and keying solution on the BES and the BlackBerry MS. The in-band KAS solution integrates with existing BlackBerry Application Programming Interfaces (APIs), which includes a C++ API for the BES and a Java API for the BlackBerry MS, and fits within the BES infrastructure -- making it the simplest solution to manage.

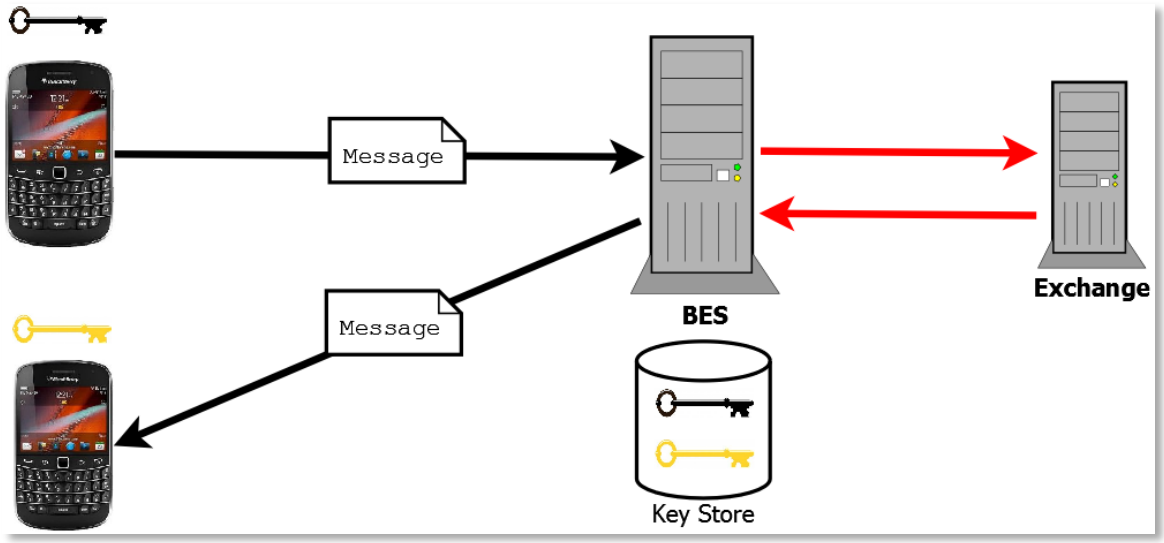

Figure 1 – Module Message Encryption/Decryption Flow

The Suite B Cryptographic Module, hereafter referred to as the Module, operates as one of several layers of encryption within the BlackBerry infrastructure. The BlackBerry encryption is invoked automatically when the Module is instantiated, providing an additional layer of encryption and obfuscation above the Module. Additional encryption at the application layer can be added by enabling S/MIME encryption on emails and SSL/TLS encryption on web traffic via the use of the BlackBerry

S/MIME Support Package and the BlackBerry MDS Connection Service. All of these additional layers of encryption use FIPS 140-2 Level 1 certified cryptographic libraries, either from BlackBerry (FIPS 140-2 Validation Certificate #1669) on the BlackBerry MS or Microsoft (FIPS 140-2 Validation Certificate #1335) on the BES. The Module has been developed to operate on Microsoft Windows Server 2008 with BES version 5.0 (Service Pack 3) or later and the BlackBerry Operating System (OS) version 7.0.0/7.1.0. The Module has been tested on Microsoft Windows Server 2008 with BES version 5.0 (Service Pack 3) and the BlackBerry OS version 7.0.0. The Module has been developed in Microsoft Visual C++ 2010 for the BES portion of the solution and in Java BlackBerry OS 7.0.0 for the BlackBerry MS. The Module relies on the Cryptographic Library, which has been developed from the same source code base and performs the same cryptographic functions end-to-end.

The Module must be installed on the server hosting the BES and on the BlackBerry MS during initial provisioning. Once installed with the appropriate BES security policy, the Module cannot be removed from the BlackBerry MS without performing a complete wipe of the BlackBerry MS. The Module key exchange functions are managed from the BES by the KAM web application, which is developed by KEYW but does not perform any cryptographic functions, only the management of those functions.

The Module meets the requirements of the FIPS 140-2 Security Level 1 specification. The Module Cryptographic Library provides the following cryptographic services:

- Data encryption and decryption
- Message digest and authentication code generation
- Digital signature verification
- Elliptic curve key agreement

The Module leverages Random Bit Generators (RBGs) from the FIPS 140-2 certified environments on which it runs based upon configuration.

- BlackBerry MS
  - BlackBerry RBG – FIPS 140-2 Level 1 (Certificates #132 and #133)
  - Supports FIPS 140-2 certified microSD Smart Card HRNG (Compatible with SafeNet microSD Smart Card 650 (µSC650) HRNG)
- BES
  - Microsoft Cryptographic Library RBG – FIPS 140-2 Level 1 (Certificates #23 and #27)
  - SafeNet Luna SA5 with Luna SA 7000 PCI card HRNG – FIPS 140-2 Level 3 (Certificate #998)

1.3. FIPS 140-2 Security Levels

The Module meets the overall requirements applicable to Level 1 security for FIPS 140-2 as shown in the table below:

Cryptographic Module Specification	1
Cryptographic 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
Cryptographic Module Security Policy	1

Table 1 – Summary of Achieved FIPS 140-2 Security Levels

2. Cryptographic Module Specification

2.1. Security Functions

The Module Cryptographic Library is software that implements the following FIPS-approved security functions:

Algorithm	Description	CAVP Certificate No.
AES-128, AES-192, AES-256	FIPS Publication 197, The Advanced Encryption Standard (AES), U.S. DoC/NIST, November 26, 2001, Natl. Inst. Stand. Technol. (Reference [3])	#2603
GCM	NIST SP 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007, Natl. Inst. Stand. Technol. (Reference [4])	#2603
ECDSA	ANS X9.62-2005: Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA). Per the NIST SP 800-131A transition: curve sizes less than P-224 shall not be used. (Reference [8]/[15])	#448
ECDH	ECDH Key Agreement Scheme (KAS) implementing the Full Unified Model, C(2, 2, ECC CDH) as described in NIST SP 800-56A, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, Revision 1, March 2007, Natl. Inst. Stand. Technol. Per the NIST SP 800-131A transition: curve sizes less than P-224 shall not be used. (Reference [1]/[2]/[15])	#98 and #259 (CVL Certificate Nos.)
SHA-1, 224, 256, 384, 512, 512/224, 512/256	FIPS Publication 180-4, Secure Hash Standard (SHS), March 2012, Natl. Inst. Stand. Technol. (Reference [5])	#2187
HMAC-SHA-1, 224, 256, 384, 512, 512/224, 512/256	FIPS Publication 198-1, The Keyed-Hash Message Authentication Code (HMAC), July 2008, Natl. Inst. Stand. Technol. (Reference [6])	#1610
XTS	NIST SP 800-38E, Recommendation for Block Cipher Modes of Operation: the XTS-AES Mode for Confidentiality on Storage Devices, January 2010, Natl. Inst. Stand. Technol. (Reference [7])	#2603

Table 2 – FIPS-Approved Security Functions

2.2. Modes of Operation

The Module must be installed on the BES and the BlackBerry MS manually, and once installed the Module Cryptographic Library runs all algorithms in FIPS 140-2 compliant mode. There are no algorithms or "expanded" cryptographic modes within the Module that are not FIPS 140-2 compliant.

As mentioned in Section 1.2, the Module leverages RBGs from the FIPS 140-2 certified environments that shall be configured for FIPS mode.

- BlackBerry MS
  - Enable the Enforce FIPS Mode of Operation IT policy rule via the BAS to guarantee generating FIPS-validated random bytes for the ephemeral keys and initialization vectors
- BES
  - Enable the FIPS compliant algorithms mode via the Local Security Policy to guarantee generating FIPS-validated random bytes for the ephemeral keys, nonces and initialization vectors

2.3. Cryptographic Boundary

The physical boundary of the Module is the physical boundary of the BlackBerry MS or BES hardware device that executes the Module as shown in the following figure. Consequently, the embodiment of the Module is a multiple-chip standalone.

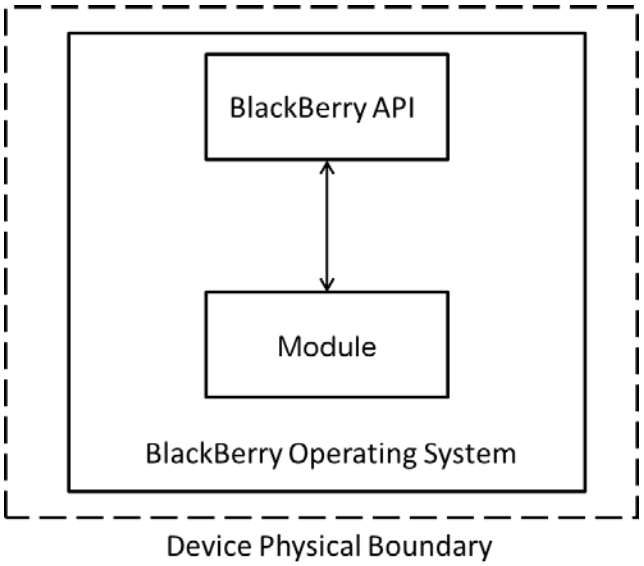

Figure 2 – Module Cryptographic Boundary

2.4. Determining Module Version

The operator can determine the version of the Module by performing the following steps:

On the BlackBerry MS:

1. On the BlackBerry MS Home screen, click the Options icon
2. Click Device > Application Management
3. The Applications screen displays the KEYWxcoder version as v2.3.1

On the BES:

1. On the BES, right-click the KEYWxcoder.dll file and click view Properties
2. Click Details tab
   The File version property displays the KEYWxcoder version as v2.3.1

3. Cryptographic Module Ports and Interfaces

The Module ports correspond to the physical ports of the BlackBerry MS and BES executing the Module, and the Module interfaces correspond to the logical interfaces to the Module. The following table and figures describe the Module ports and interfaces.

FIPS 140-2 Interface	Module Ports	Module Interfaces
Data Input	BlackBerry MS: All data traffic (email, contacts, calendars, web traffic, management traffic). Cellular Voice traffic is excluded.	Input parameters of Module function calls
	BES: All data traffic (email, contacts, calendars, web traffic, management traffic). Cellular Voice traffic is excluded.	
Data Output	BlackBerry MS: All data traffic (email, contacts, calendars, web traffic, management traffic). Cellular Voice traffic is excluded.	Output parameters of Module function calls
	BES: All data traffic (email, contacts, calendars, web traffic, management traffic). Cellular Voice traffic is excluded.	
Control Input	BlackBerry MS: Touch Screen, BlackBerry Buttons	Module function calls
	BES: Keyboard, Mouse	
Status Output	BlackBerry MS: LCD, LED	Return codes of Module function calls
	BES: BlackBerry Dispatcher Logs, KAM web application	
Power Input	BlackBerry MS: USB Port, Battery	N/A
	BES: AC Power Supply	
Maintenance	N/A	N/A

Table 3 – Module Ports and Interfaces

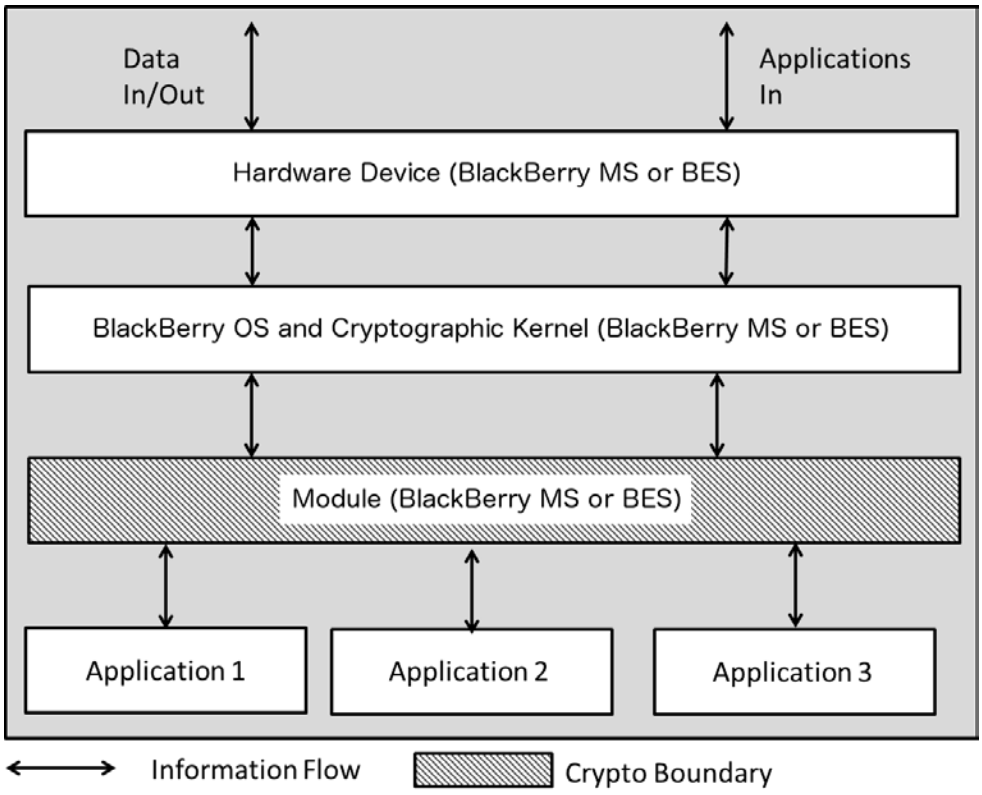

Figure 3 – Module I/O

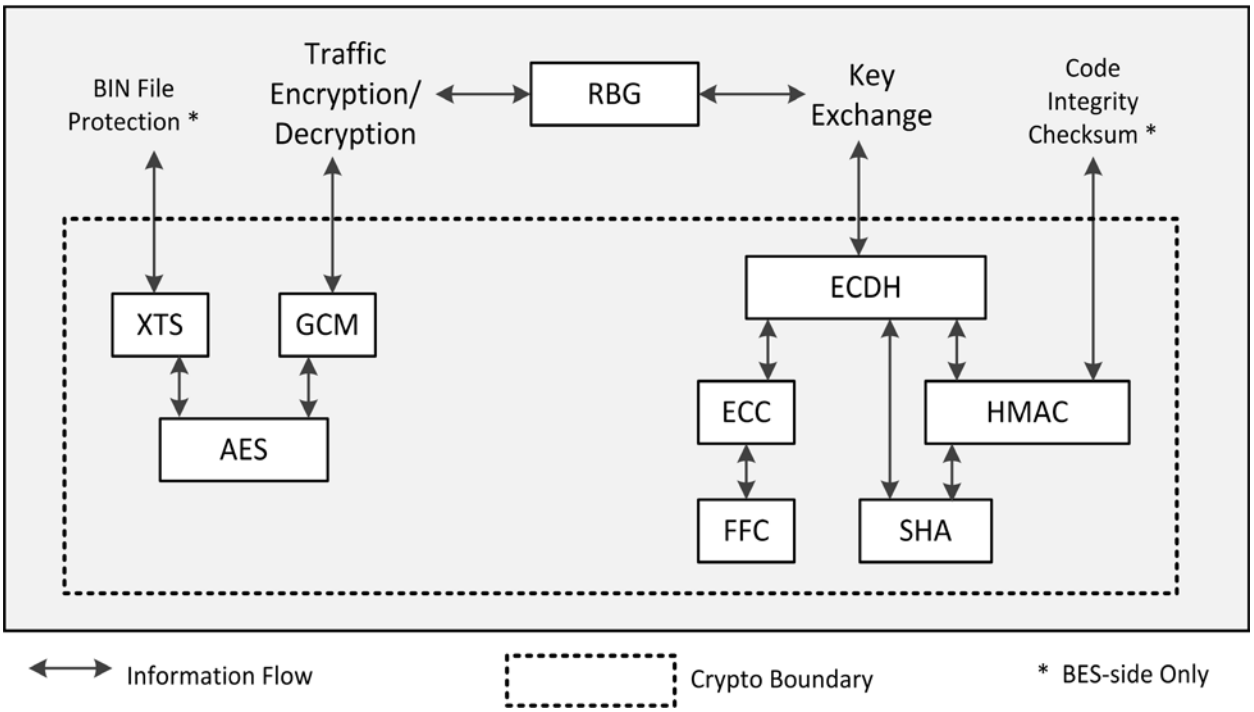

Figure 4 – Module Cryptographic Boundary I/O

4. Roles, Services, and Authentication

4.1. Roles

The Module supports user and cryptographic officer roles. The Module does not support a maintenance role. The Module does not support multiple or concurrent operators and is intended for use by a single operator, thus it always operates in a single-user mode of operation.

4.2. Services

The services described in the following tables are available to the operator roles:

Cryptographic Officer Role (BES/KAM)		
Service	**Description**	**Input/Output**
Start Transcoding	Performed by the KAM during BlackBerry MS provisioning, which initializes the Module and allows encryption and decryption of data traffic. The Module will continue Transcoding until the KAM performs Stop Transcoding or as a result of a fault in Key Exchange.	Input: The state of the Module BIN file for a BlackBerry MS is modified to Start Transcoding.
Stop Transcoding	Performed by the KAM or as a result of a fault in Key Exchange, which disallows Module encryption and decryption of data traffic.	Input: The state of the Module BIN file for a BlackBerry MS is modified to Stop Transcoding.
Key Exchange	Initiated by the KAM on a schedule and/or manually and also by a custom email message from an email server, which performs key agreement between the BES and a BlackBerry MS using ECDH to negotiate new symmetric keys for data traffic encryption and decryption.	Input: The state of the Module BIN file for a BlackBerry MS is modified to perform Key Exchange.
View Status	The Module status for a BlackBerry MS is displayed by the KAM. The Module status for the BES is displayed by Windows Services via the BlackBerry Dispatcher service in the Status column. The BlackBerry Dispatcher service can be started, stopped or re-started via Windows Services.	Output: The state of the Module BIN file for a BlackBerry MS determines status. The running state of the Module for the BES determines status.
Zeroize	Performed by the BlackBerry “Wipe Handheld” or “Security Wipe” function via the BES, which remotely erases all user data, certificates and keys on a BlackBerry MS. The KAM can also delete the Module BIN file for a BlackBerry MS, which erases the keying material on the BES for that BlackBerry MS.	Input: BlackBerry Encrypted Key Store is erased and BIN file is deleted.

Table 4 – Module Services for Cryptographic Officer Role

User Role (BlackBerry MS)		
Service	**Description**	**Input/Output**
View Status	Displayed in the BlackBerry notification area and BlackBerry MS logs.	Output: The state of the status flag determines status.
Encrypt Data Traffic	Encrypts all data traffic that is sent from the BlackBerry MS or BES with a symmetric key that was negotiated during Key Exchange.	Output: Plaintext data is encrypted into ciphertext.
Decrypt Data Traffic	Decrypts all data traffic that is received by the BlackBerry MS or BES with a symmetric key that was negotiated during Key Exchange.	Input: Ciphertext data is decrypted into plaintext.
Zeroize	Performed by the BlackBerry "Wipe Handheld" or "Security Wipe" function on the BlackBerry MS, which erases all user data, certificates and keys on that BlackBerry MS.	Input: BlackBerry Encrypted Key Store is erased.

Table 5 – Module Services for User Role

4.3. Authentication

The Module does not support operator authentication. Roles are implicitly selected based on the service performed by the operator.

Role	Type of Authentication	Authentication Data
Cryptographic Officer (BES/KAM)	Module: None BES: Keyboard Login	Module: None BES: Username and Password as required by IT Policy
User (BlackBerry MS)	Module: None BlackBerry MS: Keyboard Login	Module: None BlackBerry MS: User PIN or Password as required by IT Policy

Table 6 – Module Authentication

5. Physical Security

The Module is implemented entirely in software, thus it is not subject to the FIPS 140-2 Physical Security requirements.

The BES that executes the Module is located on production grade equipment within the backend network infrastructure and is expected to be secure by best practices. Similarly, on the BlackBerry MS, physical security is provided as a basic requirement for BlackBerry production-grade components that host the Module.

6. Cryptographic Keys and Critical Security Parameters

The following table describes the cryptographic keys, key components and Critical Security Parameters (CSPs) utilized exclusively by the Module.

Key/CSP	Type(s) of Access	Input/Output	Storage	Destruction
HMAC Integrity Check Key *Used for Software Integrity Checksum.*	Cryptographic Officer Role (BES/KAM): Read & Write	Generated (using a KEYW proprietary method) during each Module registration. A new key is generated after each build.	HMAC key not stored, only briefly generated (in RAM) during Module registration	Destroyed (zeroized) immediately after each Module registration
ECDH Key Establishment Keys *The BES and each BlackBerry MS are acting as independent entities within a PKI framework and use approved methods of (traffic) key establishment.*	User Role (BlackBerry MS): Read & Write	When executing a key agreement scheme, each side imports its own PKI static key pair (private and public keys) and imports the other side's public key, accessing PKI Certificates issued (and signed) by a Certification Authority (CA). Additionally, each side outputs internally a Private Ephemeral key and then outputs the corresponding Public Ephemeral key to the other side.	BlackBerry MS: BlackBerry Encrypted Key Store	BlackBerry MS: Erased on "Wipe Handheld" or "Security Wipe" command
	Cryptographic Officer Role (BES/KAM): Read & Write		BES: since retrieving static keys from PKI Certificates is repetitive and time consuming, the static keys are cached (and KEK- encrypted) in the BIN files in order to expedite subsequent key exchanges	BES: Delete BIN file
XTS-AES Keys *Serve as Key Encryption Keys (KEKs) to protect (encrypt) the contents of BIN files on the BES.*	Cryptographic Officer Role (BES/KAM): Read & Write	Generated (using a KEYW proprietary method) each time CSPs are accessed and/or updated on a BIN file. Each BIN file has its own KEK.	XTS-AES keys not stored, only briefly generated (in RAM) during CSPs access	Destroyed (zeroized) immediately after each usage

AES (Traffic) Keys *Symmetric keys used for encryption and decryption of traffic packets.*	User Role (BlackBerry MS): Read & Write	Negotiated after a BES-initiated Key Agreement process, using an approved ECDH scheme. Distinct keys are used for incoming and outgoing traffic.	BlackBerry MS: BlackBerry Encrypted Key Store	BlackBerry MS: Erased on "Wipe Handheld" or "Security Wipe" command
	Cryptographic Officer Role (BES/KAM): Read & Write		BES: BIN files on server protected by KEKs	BES: traffic keys not archived; existing keys discarded and substituted by newly negotiated keys
GCM IVs and Tags *Used during GCM authenticated encryption of traffic packets.*	User Role (BlackBerry MS): Read & Write	BlackBerry MS: IV provided by RBG on MS before packet encryption, GCM authentication Tag computed after packet encryption	BlackBerry MS: IVs and Tags not stored, only briefly generated (in RAM) during packet transmission	BlackBerry MS: Erased after transmission of outgoing packets and reception/ verification of incoming packets
	Cryptographic Officer Role (BES/KAM): Read & Write	BES: IV provided by RBG on BES before packet encryption, GCM authentication Tag computed after packet encryption	BES: IVs and Tags not stored, only briefly generated (in RAM) during packet transmission	BES: Erased after transmission of outgoing packets and reception/ verification of incoming packets
HMAC Keys *Used during HMAC-SHA-1, 256, 384, 512 operations executed during the Key Confirmation phase of key exchange.*	User Role (BlackBerry MS): Read & Write	BlackBerry MS: Split from the front part of the Derived Key Material built during key exchange	BlackBerry MS: HMAC keys not stored, only briefly generated (in RAM) during key exchange	BlackBerry MS: Erased after the Key Confirmation phase of key exchange is completed
	Cryptographic Officer Role (BES/KAM): Read & Write	BES: Split from the front part of the Derived Key Material built during key exchange	BES: HMAC keys not stored, only briefly generated (in RAM) during key exchange	BES: Erased after the Key Confirmation phase of key exchange is completed

Table 7 – Module Cryptographic Keys and Critical Security Parameters

6.1. Key Zeroization

The Module leverages the built-in BlackBerry security solution to ensure algorithmic keys and key components are protected. Similarly, data and specifically key removal via zeroization, is an integral part of the BlackBerry security solution. A user can request a zeroization at any time by navigating to Options and selecting “Wipe Handheld” or “Security Wipe” on the BlackBerry MS, which erases all user data, certificates and keys on that BlackBerry MS. The BES administrator may also zeroize the BlackBerry MS remotely via the “Wipe Handheld” or “Security Wipe” command through the BlackBerry Administration Service (BAS) interface on the BES as well.

Furthermore, new symmetric keys for data traffic encryption and decryption can be negotiated using ECDH via the KAM as frequently as possible by schedule and/or manually and also by a custom email message from an email server. The KAM can also delete the Module BIN file for each individual BlackBerry MS, which erases the keying material on the BES for that BlackBerry MS.

7. Self-Tests

The Module implements a series of self-tests that are described in the following table:

Test	Description
Software Integrity	The BES validates the software integrity during registration of the Module DLL file on the BES on power-up. The integrity check is a two-step process consisting of an HMAC verification (based on the NIST-approved HMAC-160 algorithm), applied to the whole Module DLL image processed as a binary data file. In the first step, the 160-bit (20-byte) HMAC key for the HMAC verification is derived (in a KEYW proprietary manner) from several build-specific data fields including the current version string and build date. This HMAC key customization is aimed at preventing malicious Module DLL rebuilds and authenticating the original build only. In the second step, the 160-bit HMAC key is used to perform an HMAC-160 integrity check of the whole Module DLL image. This computation produces a 160-bit checksum that is compared against a hexadecimal value pre-stored in the KEYWxcoder.ini file.
	The BlackBerry MS validates the software integrity during registration of the Module COD file on a BlackBerry MS on power-up, which is only allowed if the BES has deployed the Security Transcoder Cod File Hashes IT policy rule and the Primary Transcoder IT policy rule (only applicable for BES 5.0 SP4 or later and BlackBerry OS 7.1.0.9 or later) via IT Policy to the BlackBerry MS, which contains the SHA-1 hash of the Module COD file.
GCM Encrypt/Decrypt	Exercises a set of Known Answer Tests (KATs) extracted from the GCM test vectors published by NIST in the GCMVS specification (Reference [9]) on all three GCM encryption modes corresponding to AES key sizes of 128, 192 and 256 bits featuring the largest combinations of PT, IV and AAD.
SHA	Exercises a set of Known Answer Tests (KATs) extracted from the SHA test vectors published by NIST in the SHAVS specification (Reference [10]) on all SHA versions specified in FIPS Publication 180-4 including the new SHA-512/224 and SHA-512/256 featuring mixed hash/digest size combinations with the longest input data. The comprehensive SHA KATs implicitly provide assurance about the validity of the Key Derivation Function (KDF) employed by the ECDH Key Agreement Scheme (as recommended in NIST SP 800-56A - Reference [1], a SHA-based concatenation KDF is being used).
HMAC	Exercises a set of Known Answer Tests (KATs) extracted from the HMAC test vectors published by NIST in the HMACVS specification (Reference [11]) featuring the largest combinations of key and tag sizes covering all versions of the underlying hashing algorithm (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224 and SHA-512/256). The comprehensive HMAC KATs implicitly provide assurance about the validity of the Bilateral Key Confirmation method employed by the ECDH Key Agreement Scheme (Reference [1], Section 8.4).

ECDSA KeyPair/PKV	Exercises a set of Known Answer Tests (KATs) adapted from the ECDSA KeyPair (private/public key verification) and PKV (Public Key Validation) test vectors published by NIST in the ECDSA2VS specification (Reference [14]) covering each version of the underlying prime-field EC (P-192, P-224, P-256, P-384 and P-521). The ECDSA KeyPair tests include multiple KAT verifications of ECC point multiplication, which is the ECC primitive used for shared-secret ("Z") computation by the ECDH Key Agreement Scheme.
Key Agreement Scheme (KAS)	Exercises a set of Known Answer Tests (KATs) adapted from the ECDH test vectors published by NIST in the KASVS specification (Reference [12]) featuring the Full Unified model of ECDH covering each version of the underlying prime-field EC (P-192, P-224, P-256, P-384 and P-521). Each test run includes both Initiator-side and Responder-side functions. Every invocation of the Key Agreement Scheme involves (within the BES and BlackBerry MS class constructors) a verification of the arithmetic validity of the selected set of ECC domain parameters (Reference [1], Section 5.5.2). The KAS implementation provides built-in assurance (verification) of the arithmetic validity of a public key, by performing a full ECC public key validation each time such a key is being used: each side verifies both own and opposite static public keys, each side verifies opposite side's ephemeral public key (Reference [1], Section 5.6.2). Also, during key agreement, each side renews its assurance of possessing the correct private key by using the Key Regeneration method (Reference [1], Section 5.6.3), while the ephemeral (generated) private key is subjected to the constraints specified in Reference [1], Section 5.6.1.2. The underlying cryptographic algorithms used during ECDH key agreement are fully validated via individual power-on self-tests: • ECC point multiplication is validated via ECDSA KeyPair KATs • The Key Derivation Function is validated via SHA KATs • The Key Confirmation function is validated via HMAC KATs
XTS Encrypt/Decrypt	Exercises a set of Known Answer Tests (KATs) extracted from the XTS test vectors published by NIST in the XTSVS specification (Reference [13]). Both formats specified for the tweak value input (128-bit hexadecimal string or 64-bit Data Unit Sequence Number) are being tested with various, non-trivial Data Unit bit sizes in encrypt and decrypt mode.

Table 8 – Module Self-Tests

7.1. Invoking Self-Tests

The operator can invoke the power-on self-tests on the BlackBerry MS by hard resetting the BlackBerry MS (soft resetting the BlackBerry MS will not invoke power-on self-tests). At power-on the BlackBerry OS executes the Module's Default Entry Point (DEP) automatically, which invokes the self-tests listed in Table 8 and does not require operator intervention.

```
public static void main(String[] args) // Default Entry Point (DEP)
```

The operator can invoke the power-on self-tests on the BES by restarting the BlackBerry Dispatcher service via Windows Services. At power-on the BlackBerry Dispatcher service executes the Module's DEP automatically, which invokes the self-tests listed in Table 8 and does not require operator intervention.

```
int __cdecl LoadDLL() // Default Entry Point (DEP)
```

If the Software Integrity self-test fails the Module will not load and an error is logged. If the KAT self-tests fail the Module will prohibit any subsequent cryptographic operations and an error is logged. Subsequent self-tests on both the BES and BlackBerry MS exercise all Suite B cryptographic algorithms used by the Module, either via regular traffic encryption/decryption or during key exchange.

The Module does not rely on any other external service to initiate the power-on self-tests.

7.2. Self-Tests Results

Upon successful self-test completion, the Module will complete its initialization and transition to normal operational state. In the event of a self-test failure, the Module will enter an error state and a specific error code will be returned indicating which self-test has failed. The Module will not provide any cryptographic services while in this state.

Self-Test	Possible Error Code
Software Integrity Checksum	444
GCM Encrypt	2100 + Test Count
GCM Decrypt	2200 + Test Count
SHA	2300 + Test Count
HMAC	2400 + Test Count
ECDSA Key	2800 + Test Count
KAS	2500 + Test Count (combined indicator of the EC type and failing sub-test)
XTS Encrypt	2600 + Test Count
XTS Decrypt	2700 + Test Count

Table 9 – Module Self-Test Error Codes

8. Mitigation of Other Attacks

The Module has not been designed to mitigate any specific attacks outside the scope of the FIPS 140-2 requirements. The Module resides within the FIPS 140-2 BlackBerry Cryptographic Kernel operating environment, which provides an additional layer of protection to attacks of the Module.

Furthermore, any concerns related to the recently discovered (April 2014) bug in some TLS implementations, publicized as the "heartbleed bug" and referenced as **CVE-2014-0160** in the National Vulnerability Database, are not applicable to the Module implementation. This is because the Module, either during validation testing or regular operation, does not employ any services, nor does it import any software components, pertaining to affected OpenSSL implementations.

9. Referenced Documents

[1] NIST SP 800-56A, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, Revision 1, March 2007, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/nistpubs/800-56A/SP800-56A_Revision1_Mar08-2007.pdf

[2] NIST SP 800-56A, Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography (Draft Revision), August 2012, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/drafts/800-56a/draft-sp-800-56a.pdf

[3] FIPS Publication 197, The Advanced Encryption Standard (AES), U.S. DoC/NIST, November 26, 2001, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[4] NIST SP 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

[5] FIPS Publication 180-4, Secure Hash Standard (SHS), March 2012, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf

[6] FIPS Publication 198-1, The Keyed-Hash Message Authentication Code (HMAC), July 2008, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

[7] NIST SP 800-38E, Recommendation for Block Cipher Modes of Operation: the XTS-AES Mode for Confidentiality on Storage Devices, January 2010, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf

[8] ANS X9.62-2005: Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), November 2005

[9] The Galois/Counter Mode (GCM) and GMAC Validation System (GCMVS), National Institute of Standards and Technology, Updated: August 30, 2012, [Web page], http://csrc.nist.gov/groups/STM/cavp/documents/mac/gcmvs.pdf

[10] The Secure Hash Algorithm Validation System (SHAVS), Updated: July 23, 2012, National Institute of Standards and Technology, [Web page], http://csrc.nist.gov/groups/STM/cavp/documents/shs/SHAVS.pdf

[11] The Keyed-Hash Message Authentication Code Validation System (HMACVS), Updated: July 23, 2012, National Institute of Standards and Technology, [Web page], http://csrc.nist.gov/groups/STM/cavp/documents/mac/HMACVS.pdf

[12] The Key Agreement Schemes Validation System (KASVS), Updated September 2011, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/groups/STM/cavp/documents/keymgmt/KASVS.pdf

[13] The XTS-AES Validation System (XTSVS), Updated: March 2, 2011, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/groups/STM/cavp/documents/aes/XTSVS.pdf

[14] The FIPS 186-3 Elliptic Curve Digital Signature Algorithm Validation System (ECDSA2VS), Updated: January 9, 2013, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/groups/STM/cavp/documents/dss2/ecdsa2vs.pdf

[15] NIST SP 800-131A, Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths, January 2011, Natl. Inst. Stand. Technol. [Web page], http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf