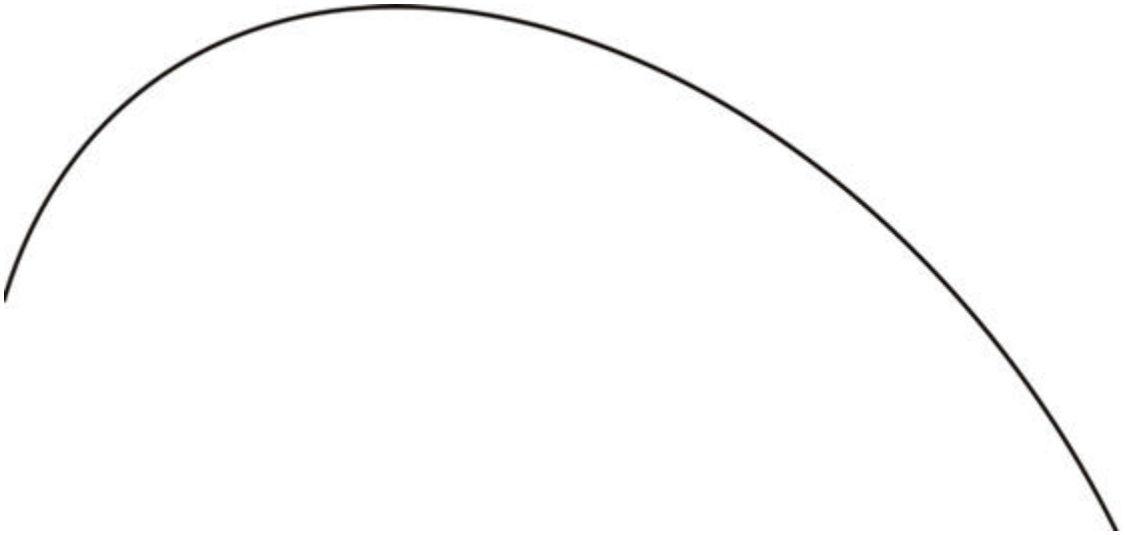

FIPS 140-2 Security Policy

BlackBerry Enterprise Server™ Cryptographic Kernel version 1.0.1.6

Document Version 1.6

BlackBerry Security Team

Research In Motion®

Document and Contact Information

Version	Date	Author	Description
1.0	30 June 2004	Michael K. Brown	Document creation.
1.1	06 July 2004	Michael K. Brown	
1.2	15 July 2004	Michael K. Brown	
1.3	12 August 2004	Michael K. Brown	
1.4	09 September 2004	Michael K. Brown	
1.5	04 October 2004	Michael K. Brown	Algorithm validation certificate numbers included.
1.6	14 December 2004	David MacFarlane	Updated per CMVP review comments.

Contact	Corporate Office
BlackBerry Security Team	**Research In Motion Limited**
BlackBerrySecurity@rim.com (519) 888-7465 ext. 2921	175 Columbia Street West Waterloo ON Canada N2L 5Z5 www.rim.com

Contents

List of Tables

List of Figures

Introduction

The BlackBerry Enterprise Server™ centralises email redirection for BlackBerry Wireless Handheld™ users in an organisation and performs the following functions for each user:

- Monitors the user's Inbox for new mail;
- Applies filters to new messages to determine if and how to redirect them to a user's BlackBerry handheld;
- Compresses and encrypts new messages and delivers them to the BlackBerry handheld over the Internet; and
- Receives, decompresses, and decrypts new messages composed on the BlackBerry handheld and places them in the user's Outbox for delivery by the corporate mail server.

The BlackBerry Enterprise Server™ operates in Microsoft® Exchange, IBM® Lotus® Domino®, and Novell® GroupWise® messaging environments.

The BlackBerry Enterprise Server™ Cryptographic Kernel, hereafter referred to as *cryptographic module* or *module*, is a software cryptographic module that provides data encryption and decryption and other cryptographic services to the BlackBerry Enterprise Server™. The module has been validated to FIPS 140-2 Security Level 1, and the Security Level achieved for each of the eleven sections of FIPS 140-2 is identified in the following table.

Table 1. Security Level Achieved by FIPS 140-2 Section

FIPS 140-2 Section	Security Level Achieved
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 Keys Management	1
EMI/EMC	3
Self-Tests	1
Design Assurance	3
Mitigation of Other Attacks	N/A

Cryptographic Module Specification

The module is a Microsoft Windows®-compatible dynamically linked library (DLL) that performs data encryption and decryption and calculates message digests and authentication codes.

The module implements the following FIPS-Approved[1] security functions:

- **AES-128**, **AES-192**, and **AES-256** (encrypt and decrypt), as specified in FIPS 197. The implementation supports the ECB and CBC modes of operation and has been awarded AES validation certificate #104, http://csrc.nist.gov/cryptval/aes/aesval.html.
- **Triple DES** (encrypt and decrypt), as specified in FIPS 46-3. The implementation supports the ECB and CBC modes of operation and has been awarded Triple DES validation certificate #216, http://csrc.nist.gov/cryptval/des/tripledesval.html.
- **SHA-1, SHA-224, SHA-256, SHA-384,** and **SHA-512**, as specified in FIPS 180-2. The implementation has been awarded SHS validation certificate #265, http://csrc.nist.gov/cryptval/shs/shaval.htm.
- **HMAC SHA-1, HMAC SHA-224, HMAC SHA-256, HMAC SHA-384,** and **HMAC SHA-512**, as specified in FIPS 198. The implementation has been awarded HMAC validation certificate #2, http://csrc.nist.gov/cryptval/hmac/hmacval.htm.
- **FIPS 186-2 RNG**, as specified in FIPS 186-2 Appendix 3.1. The implementation uses SHA-1 as the function *G* and has been awarded RNG validation certificate #28, http://csrc.nist.gov/cryptval/rng/rngval.htm.

The module implements the following non-Approved security functions:

- **Rijndael** (encrypt and decrypt). The implementation supports the ECB and CBC modes of operation; key lengths of 128, 160, 192, 224 and 256 bits; and block lengths of 128[2], 160, 192, 224 and 256 bits.

The physical boundary of the module is the physical boundary of the general purpose computer (GPC) that executes the module and is shown in the following figure:

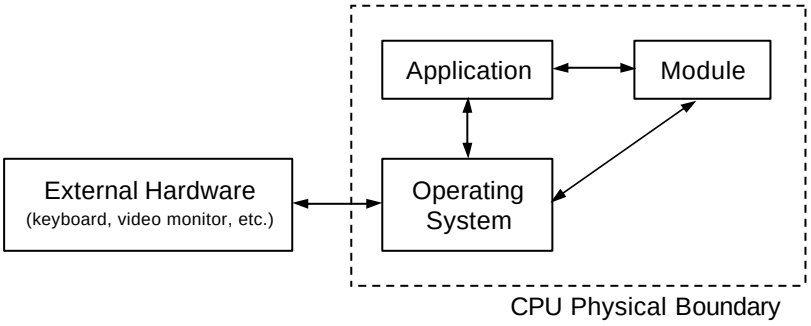

Figure 1. Physical Boundary

[1] A security function is FIPS-Approved if it is explicitly listed in *FIPS 140-2 Annex A: Approved Security Functions for FIPS PUB 140-2*.
[2] Supported for key lengths of 160 and 224 bits only.

Cryptographic Module Ports and Interfaces

The physical ports of the module correspond to the ports of the GPC that executes the module, and the logical interface of the module is its application programming interface (API). The module implements the required FIPS 140-2 interfaces as shown in the following table:

Table 2. Implementation of FIPS 140-2 Interfaces

FIPS 140-2 Interface	Module Implementation
Data Input	The module implements the Data Input Interface via the input parameters of each API function call.
Data Output	The module implements the Data Output Interface via the output parameters of each API function call.
Control Input	The module implements the Control Input Interface via the API function calls.
Status Output	The module implements the Status Output Interface via specific API function calls that return status information and the return code provided by each API function call after execution.

Roles, Services, and Authentication

Services

The following table describes the services that are available to the operator:

Table 3. Module Services

Service	API Function Calls	Description
Show Status	*ShowStatus, GetLastCryptoError*	Displays the status of the module.
Perform Self-Tests	*SelfTest*	Executes the cryptographic algorithm known answer tests (KATs), a subset of the power-up self-tests.
Encrypt Data	*TDES_ECB_Encode, TDES_CBC_Encode, AES_ECB_Encode, AES_CBC_Encode*	Encrypts data using AES, Triple DES, or Rijndael, as specified by the operator.
Decrypt Data	*TDES_ECB_Decode, TDES_CBC_Decode, AES_ECB_Decode, AES_CBC_Decode*	Decrypts data using AES, Triple DES, or Rijndael, as specified by the operator.
Create Message Digest	*SHA1_DigestMessage, SHA224_DigestMessage, SHA256_DigestMessage, SHA384_DigestMessage, SHA512_DigestMessage*	Calculates a message digest using SHA-1, SHA-224, SHA-256, SHA-384, or SHA-512, as specified by the operator.
Create MAC	*HMAC_SHA1_DigestMessage, HMAC_SHA224_DigestMessage, HMAC_SHA256_DigestMessage, HMAC_SHA384_DigestMessage, HMAC_SHA512_DigestMessage*	Calculates a message authentication code using HMAC SHA-1, HMAC SHA-224, HMAC SHA-256, HMAC SHA-384, or HMAC SHA-512, as specified by the operator.
Generate Random Data	*GetRandomData*	Generates random data using the FIPS 186-2 RNG.

Roles

The module supports a User and Crypto Officer role. The module does not support a Maintenance role. Role selection is performed implicitly and is dependent on the service performed by the operator. The following table summarises implicit role selection based on services and the associated access to critical security parameters (CSPs):

Table 4. Role Selection and CSP Access by Service

Service	Role Implicitly Selected	Affected Keys and CSPs	Access to Keys and CSPs
Show Status	User	N/A	N/A
Perform Self-Tests	Crypto Officer	Software Integrity Key	Read, Execute
Encrypt Data	User	AES Key	Read, Execute

Service	Role Implicitly Selected	Affected Keys and CSPs	Access to Keys and CSPs
		Triple DES Key	Read, Execute
		Rijndael Key	Read, Execute
Decrypt Data	User	AES Key	Read, Execute
		Triple DES Key	Read, Execute
		Rijndael Key	Read, Execute
Create Message Digest	User	N/A	N/A
Create MAC	User	HMAC Key	Read, Execute
Generate Random Data	User	N/A	N/A

Authentication

The module does not require or support operator authentication.

Physical Security

The module is implemented entirely in software, thus the FIPS 140-2 physical security requirements are not applicable.

Operational Environment

The module is designed to execute on a GPC in conjunction with the BlackBerry Enterprise Server™ application, which supports the following operating systems:

- Windows NT® Server 4.0 SP 5 or later (for a Microsoft Exchange environment)
- Windows NT Server 4.0 SP 6a or later (for an IBM Lotus Domino environment)
- Windows® 2000 Server SP 1 or later

The operating system is restricted to a single user mode of operation per FIPS 140-2 Implementation Guidance 6.1, i.e., the BlackBerry Enterprise Server™ application is the single user of the module, even when the server application is serving multiple clients.

For the purposes of FIPS 140-2 conformance testing, the module was tested on Windows NT Server 4.0 SP 6a, however the module may be executed on any of the supported operating systems and remain FIPS-compliant.

Cryptographic Key Management

General

The following table describes the cryptographic keys, key components, and CSPs utilised by the module:

Table 5. Cryptographic Keys and CSPs

Key / CSP	Description
AES Key	A symmetric key used to encrypt and decrypt data using the AES algorithm. The module supports AES key lengths of 128, 192, and 256 bits.
Triple DES Key	A symmetric key used to encrypt and decrypt data using the Triple DES algorithm. Per the specification of Triple DES, all Triple DES keys are 192 bits in length.
HMAC Key	A key used to calculate a MAC using the HMAC algorithm. The length of the HMAC key is dependent on the underlying hash algorithm.
Software Integrity Key	A 128-bit HMAC SHA-1 key used to calculate and verify the integrity of the module.
Rijndael Key	A symmetric key used to encrypt and decrypt data using the Rijndael algorithm. The module supports Rijndael key lengths of 128, 160, 192, 224, and 256 bits.

Random Number Generators

The module implements the RNG specified in FIPS 186-2 Appendix 3.1. No additional RNGs are implemented.

Key Generation

The random data generated by the module is done so using a FIPS-Approved and -validated RNG, thus a symmetric key for use with the module may be generated using the Generate Random Data service. The module, however, does not provide an explicit key generation service.

Key Entry and Output

Keys are entered into the module in plaintext via the module API. The module does not support key output.

Key Storage

The module does not support general-purpose key storage. Operational keys are stored in the GPC memory only as long as they are required for processing by the module. However, the Software Integrity Key that is used during the Software Integrity Test is compiled into the module software.

Key Zeroization

The module zeroizes operational keys once they are no longer needed for processing. The operator may zeroize the Software Integrity Key by unloading the module from the GPC memory.

Self-Tests

The following table describes the self-tests implemented by the module:

Table 6. Module Self-Tests

Test	Description
Software Integrity Test	The Software Integrity Test verifies the integrity of the module software using HMAC SHA-1.
FIPS 186-2 RNG KAT	The FIPS 186-2 RNG KAT verifies that the RNG is operating correctly.
Continuous RNG Test	The module implements a continuous RNG test, as specified in FIPS 140-2, for the implemented RNG.
Triple DES KAT	The Triple DES KAT verifies that the Triple DES encryption and decryption functions are operating correctly.
AES KAT	The AES KAT verifies that the AES encryption and decryption functions are operating correctly.
SHA-1 KAT	The SHA-1 KAT verifies that the SHA-1 hashing function is operating correctly.
SHA-224 KAT	The SHA-224 KAT verifies that the SHA-224 hashing function is operating correctly.
SHA-256 KAT	The SHA-256 KAT verifies that the SHA-256 hashing function is operating correctly.
SHA-384 KAT	The SHA-384 KAT verifies that the SHA-384 hashing function is operating correctly.
SHA-512 KAT	The SHA-512 KAT verifies that the SHA-512 hashing function is operating correctly.
HMAC SHA-1 KAT	The HMAC SHA-1 KAT verifies that the HMAC SHA-1 function is operating correctly.
HMAC SHA-224 KAT	The HMAC SHA-224 KAT verifies that the HMAC SHA-224 function is operating correctly.
HMAC SHA-256 KAT	The HMAC SHA-256 KAT verifies that the HMAC SHA-256 function is operating correctly.
HMAC SHA-384 KAT	The HMAC SHA-384 KAT verifies that the HMAC SHA-384 function is operating correctly.
HMAC SHA-512 KAT	The HMAC SHA-512 KAT verifies that the HMAC SHA-512 function is operating correctly.

When an operator attempts to load the module into GPC memory, the power-up self-tests are executed. The power-up self-tests comprise of all the tests identified above, except the Continuous RNG Test. The Software Integrity Test is the first self-test executed, and if it fails then the attempt to load the module fails. If a cryptographic algorithm KAT fails then the operator may not access the corresponding algorithm until the KAT is executed successfully.

The operator may invoke the power-up self-tests by unloading and reloading the module into GPC memory. The operator may also invoke all of the power-up self-tests, except the Software Integrity Test and the Continuous RNG Test, by invoking the **Perform Self-Tests** service.

Mitigation of Other Attacks

The module is not designed to mitigate any specialised attacks.

Installation and Start-Up

The module is installed as part of the BlackBerry Enterprise Server™ application, thus there are no module-specific installation instructions. The installation instructions for the BlackBerry Enterprise Server™ application should be followed and are given in the following documents, available from http://www.blackberry.com/:

- *BlackBerry Enterprise Server for Microsoft Exchange 5.5: Installation and Getting Started Guide*
- *BlackBerry Enterprise Server for Microsoft Exchange 2000/2003: Installation and Getting Started Guide*
- *BlackBerry Enterprise Server for Lotus® Domino: Installation and Getting Started Guide*

FIPS 140-2 Mode of Operation

In order to operate the module in a FIPS-Approved manner, the following conditions must be met:

1. The Rijndael algorithm is not used for data encryption or decryption. More specifically, the following input parameters are not used in any of the AES API function calls:
   - Keys that are 160 or 224 bits in length
   - Keys that are 128, 192, or 256 bits in length when a block size of 160, 192, 224, or 256 bits is specified.

Glossary

AES	Advanced Encryption Standard
API	Application programming interface
CBC	Cipher block chaining
CSP	Critical security parameter
DES	Data Encryption Standard
DLL	Dynamically linked library
ECB	Electronic code book
FIPS	Federal Information Processing Standard
GPC	General purpose computer
HMAC	Keyed-hashed message authentication code
IG	Implementation Guidance
KAT	Known answer test
MAC	Message authentication code
RIM	Research In Motion
RNG	Random number generator
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SP	Service pack