14.10.19

RSA BSAFE® Crypto Module 1.0.1 Security Policy

This document is a non-proprietary security policy for the RSA BSAFE Crypto Module 1.0.1 (BSAFE Crypto Module).

This document may be freely reproduced and distributed whole and intact including the copyright notice.

Contents:

Copyright © 2019 Dell Inc. or its subsidiaries. All rights reserved.

Preface

This document is a non-proprietary security policy for the BSAFE Crypto Module from RSA Security LLC, a Dell Technologies company (RSA).

This security policy describes how the BSAFE Crypto Module meets the Level 2 security requirements of FIPS 140-2 for Design Assurance, and Level 1 security requirements for all other aspects of FIPS 140-2, and how to securely operate it.

Federal Information Processing Standards Publication 140-2 - Security Requirements for Cryptographic Modules details the U.S. Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the NIST website.

This document deals only with operations and capabilities of the BSAFE Crypto Module in the technical terms of a FIPS 140-2 cryptographic module security policy.

Terminology

In this document, the term BSAFE Crypto Module denotes the BSAFE Crypto Module FIPS 140-2 validated Cryptographic Module for Overall Security Level 1 with Level 2 Design Assurance.

The BSAFE Crypto Module is also referred to as:

- The Cryptographic Module
- The BCM
- The module.

Document Organization

This document explains the BSAFE Crypto Module features and functionality relevant to FIPS 140-2, and contains the following sections:

- This section, Preface, provides an overview and introduction to the Security Policy.
- The Cryptographic Module describes the module and how it meets the FIPS 140-2 Security Level 1 requirements.
- Secure Operation of the Module addresses the required configuration for the FIPS 140-2 mode of operation.
- Acronyms lists the definitions for the acronyms used in this document.

With the exception of the Non-Proprietary *RSA BSAFE Crypto Module Security Policy*, the FIPS 140-2 Security Level 2 Design Assurance, and Security Level 1 overall validation submission documentation is proprietary to Dell Inc. and is releasable only under appropriate non-disclosure agreements. For access to the documentation, please contact RSA.

1 The Cryptographic Module

This section provides an overview of the module, and contains the following topics:

- Module Characteristics
- Module Interfaces
- Roles and Services
- Cryptographic Key Management
- Cryptographic Algorithms
- Self-tests.

1.1 Module Characteristics

BSAFE Crypto Module is classified as a multi-chip standalone cryptographic module for the purposes of FIPS 140-2. As such, BSAFE Crypto Module must be tested on a specific operating system and computer platform. The cryptographic boundary includes the module running on selected platforms running selected operating systems. The module is packaged as a Microsoft® Windows® kernel-mode dynamic library containing the module's entire executable code. The BSAFE Crypto Module toolkit relies on the physical security provided by the host PC in which it runs.

The module is validated as meeting all FIPS 140-2 Security Level 2 for Design Assurance, and Level 1 overall security requirements.

Table 1 Certification Levels

Section of the FIPS 140-2 Specification	Level
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	2
Mitigation of Other Attacks	1
Overall	1

The module relies on the physical security provided by the host on which it runs.

1.1.1 Laboratory Validated Operating Environments

For FIPS 140-2 validation, the module is tested by an accredited FIPS 140-2 testing laboratory on the following operating environments:

Operating Environment		Compiler
Microsoft Windows 10 Enterprise		
32-bit	Dell Optiplex 7040, Intel i5-6500T	• Microsoft Visual Studio 2015 • Microsoft Visual Studio 2017
64-bit	Dell Optiplex 5060, Intel i7-8700	• Microsoft Visual Studio 2015
	Dell Latitude 5590, Intel i7-8650U	• Microsoft Visual Studio 2017
Microsoft Windows 7 Enterprise SP1		
32-bit, 64-bit	Dell Optiplex 7040, Intel i5-6500T	• Microsoft Visual Studio 2015 • Microsoft Visual Studio 2017
Microsoft Windows Server 2016		
64-bit	Dell Latitude 5590, Intel i7-8650U	• Microsoft Visual Studio 2015
	Dell Optiplex 5060, Intel i7-8700	• Microsoft Visual Studio 2017

Note: All environments were tested with and without the Intel AES-NI Processor Algorithm Accelerator (PAA)

1.1.2 Affirmation of Compliance for other Operating Environments

Affirmation of compliance is defined in Section G.5, “Maintaining Validation Compliance of Software or Firmware Cryptographic Modules,” in Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program. Compliance is maintained in all operational environments for which the binary executable remains unchanged.

Specifically, RSA affirms compliance for the following operational environments:

- Microsoft Windows 8.1 Enterprise
  - 32-bit built with Visual Studio 2015
  - 64-bit built with Visual Studio 2017
- Microsoft Windows Server 2012 R2 Standard
  - 64-bit built with Visual Studio 2015
  - 64-bit built with Visual Studio 2017

- Microsoft Windows Server 2008 R2 SP1
  - 64-bit built with Visual Studio 2015
  - 64-bit built with Visual Studio 2017.

Note: The Cryptographic Module Validation Program (CMVP) makes no statement as to the correct operation of the module or the security strengths of the generated keys when the specific operational environment is not listed on the validation certificate.

1.1.3 Single Operator Mode

The module is a Windows kernel-mode dynamic link library. It is loaded into the memory space of the Windows kernel when a kernel-mode driver that references the module is loaded. The driver is the single operator of the cryptographic module and makes calls into the cryptographic module.

1.2 Module Interfaces

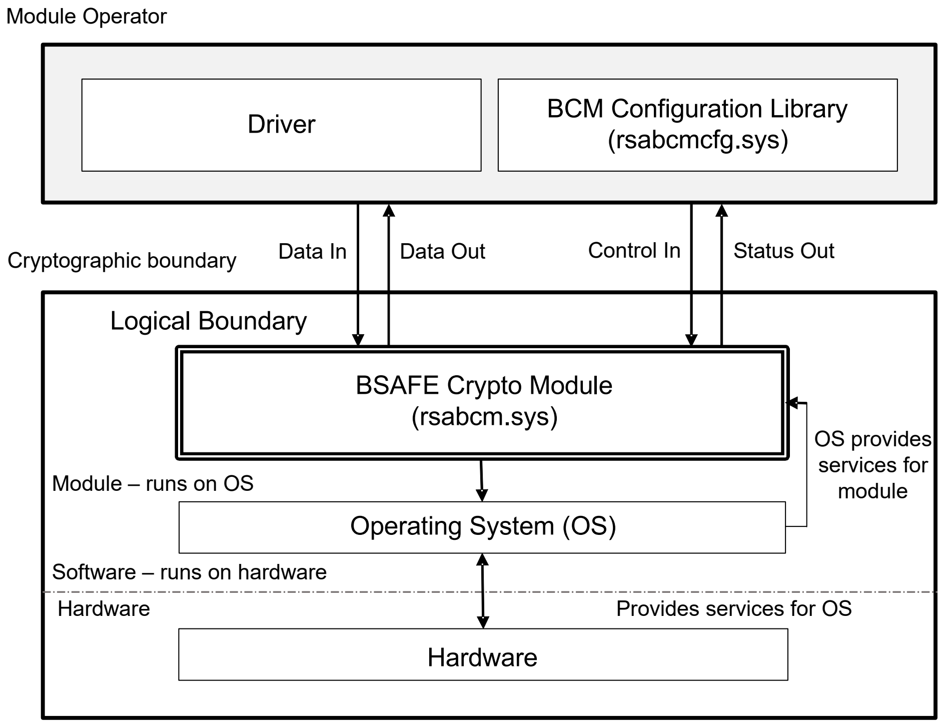
BSAFE Crypto Module is validated as a multi-chip standalone software cryptographic module. The physical cryptographic boundary of the module is the case of the general-purpose computer, which encloses the hardware running the module. The physical interfaces for the module are the physical interfaces of the computer running the module, such as the keyboard, monitor and network interface.

The logical boundary of the module is the Windows kernel-mode dynamic library comprising the module:

- *rsabcm.sys*

The underlying logical interface to the module is the API, documented in the *RSA BSAFE Crypto Module Developers Guide*. BSAFE Crypto Module provides Control Input through the API calls. Data Input and Output are provided in the variables passed with the API calls, and Status Output is provided through the returns and error codes documented for each call. This is illustrated in the following diagram.

Figure 1 BSAFE Crypto Module Logical Interfaces



1.3 Roles and Services

The module meets all FIPS 140-2 Level 1 requirements for roles and services, implementing both a Crypto User role and Crypto Officer role. As allowed by FIPS 140-2, the module does not support user identification or authentication for these roles. Only one role can be active at a time. There is no maintenance role or cryptographic bypass capability, and the module does not allow concurrent operators.

1.3.1 Crypto Officer Role

The Crypto Officer is responsible for installing and loading the cryptographic module. The module self-tests run automatically when the module is loaded. The Crypto Officer can rerun the module self-tests manually by calling `BCM_module_selftest()`.

An operator assuming the Crypto Officer role can call any module function. The Services section provides a list of functions available to the Crypto Officer role.

1.3.2 Crypto User Role

An operator assuming the Crypto User role can use the entire BSAFE Crypto Module API except for `BCM_module_selftest()`, which is reserved for the Crypto Officer. The Services section provides a list of functions available to the Crypto User Role.

1.3.3 Services

The following is the list of services provided by the module. For more information about individual functions, see the *RSA BSAFE Crypto Module Developers Guide*. .

Table 2 Services available to the Crypto Officer Role only

Services Available to the Crypto Officer Role only
Module Utility Function
`BCM_module_selftest()`

Table 3 Services available to the Crypto User and Crypto Officer Roles

Services Available to the Crypto User and Crypto Officer Roles	
Symmetric Ciphers	
`BCM_cipher_new()`	`BCM_cipher_get_iv()`
`BCM_cipher_new_iv()`	`BCM_cipher_set_iv()`
`BCM_cipher_new_XTS()`	`BCM_cipher_encrypt()`
`BCM_cipher_new_AEAD()`	`BCM_cipher_encrypt_with_padding()`
`BCM_cipher_new_CCM()`	`BCM_cipher_encrypt_with_tag()`
`BCM_cipher_delete()`	`BCM_cipher_decrypt()`
`BCM_cipher_get_block_size()`	`BCM_cipher_decrypt_with_padding()`
`BCM_cipher_get_iv_size()`	`BCM_cipher_decrypt_with_tag()`
Digests	
`BCM_digest_new()`	`BCM_digest()`
`BCM_digest_delete()`	`BCM_digest_update()`
`BCM_digest_get_size()`	`BCM_digest_final()`
Message Authentication Codes	
`BCM_mac_new()`	`BCM_mac_verify()`
`BCM_mac_delete()`	`BCM_mac_update()`
`BCM_mac_get_size()`	`BCM_mac_final()`
`BCM_mac()`	`BCM_mac_final_verify()`
Key Wrapping and Unwrapping	
`BCM_keywrap_wrap()`	`BCM_keywrap_wrap_data()`
`BCM_keywrap_unwrap()`	`BCM_keywrap_unwrap_data()`
Signing and Verifying	
`BCM_sign_get_size()`	`BCM_verify()`
`BCM_sign()`	`BCM_verify_RSA_PSS()`
`BCM_sign_RSA_PSS()`	
Asymmetric Encryption and Decryption	
`BCM_asym_get_size()`	`BCM_asym_encrypt_RSA_OAEP()`
`BCM_asym_encrypt()`	`BCM_asym_decrypt_RSA_OAEP()`
`BCM_asym_decrypt()`	
Random Number Generation	
`BCM_random_bytes()`	`BCM_random_seed()`
`BCM_secure_random_bytes()`	
Key Derivation Functions	
`BCM_pbkdf2()`	

Table 3 Services available to the Crypto User and Crypto Officer Roles (continued)

Services Available to the Crypto User and Crypto Officer Roles	
Key Functions	

`BCM_key_import()`
`BCM_key_export()`
`BCM_key_delete()`
`BCM_key_ref_inc()`
	`BCM_skey_new()`
	`BCM_skey_new_XTS()`
	`BCM_pkey_generate_RSA()`
	`BCM_pkey_validate()`

Module Information and Utility Functions	

`BCM_module_version()`
`BCM_module_state()`
`BCM_module_configure()`
`BCM_STATUS_to_string()`
	`BCM_ALG_to_string()`
	`BCM_ctx_new()`
	`BCM_ctx_delete()`
	`BCM_ctx_ref_inc()`

1.4 Cryptographic Key Management

Cryptographic key management is concerned with generating and storing keys, key assurance, managing access to keys, protecting keys during use, and zeroizing keys when they are no longer required.

1.4.1 Key Generation

The module supports the generation of RSA public and private keys. The module uses the approved CTR Deterministic Random Bit Generator (CTR DRBG) as the pseudo-random number generator (PRNG) for RSA keys, and the CTR DRBG can be used to generate symmetric keys for use with algorithms such as AES, Triple-DES and HMAC. RSA keys are generated using the approved FIPS 186-4 RSA key generation method. The output of the approved DRBG is used unmodified when symmetric keys are generated. It is also used unmodified as the random input for RSA key generation. RSA affirms compliance with the SP 800-133 standard for key generation.

1.4.2 Key Protection

All key data resides in internally allocated data structures and can be output only using the BSAFE Crypto Module API. The operating system protects memory and process space from unauthorized access.

1.4.3 Key Zeroization

The operator should follow the steps outlined in the *RSA BSAFE Crypto Module Developers Guide* to ensure sensitive data is protected by zeroizing the data from memory when it is no longer needed. All volatile keys and CSPs listed in Key Storage are zeroized by unloading the module from memory.

1.4.4 Key Storage

The module does not provide long-term cryptographic key storage. If a user chooses to store keys, the user is responsible for storing keys exported from the module.

The following table lists all keys and CSPs in the module and where they are stored.

Table 4 Key Storage

Key or CSP	Generation/Input	Storage
Hardcoded HMAC key (144-bit)	Built into the module	Persistent storage embedded in the module binary (plaintext)
RSA public/private keys (2048 to 4096-bit key sizes, less than 2048 bits for legacy signature verification only)	Entered in plaintext through the API or generated by an explicit API call	Volatile memory only (plaintext)
AES keys (128, 192, and 256-bit key sizes)	Entered in plaintext through the API	Volatile memory only (plaintext)
Triple-DES keys (192-bit key size)	Entered in plaintext through the API	Volatile memory only (plaintext)
HMAC with SHA-1 and SHA-2 (greater than 112-bit key size)	Entered in plaintext through the API	Volatile memory only (plaintext)
CTR DRBG entropy (128 bits)	Generated internally	Volatile memory only (plaintext)
CTR DRBG V value (128 bits)	Generated internally	Volatile memory only (plaintext)
CTR DRBG key (256 bits)	Generated internally	Volatile memory only (plaintext)
CTR DRBG init_seed (384 bits)	Generated internally	Volatile memory only (plaintext)

1.4.5 Key Access

An authorized operator of the module has access to all key data created during BSAFE Crypto Module operation.

Note: The Crypto User and Crypto Officer roles have equal and complete access to all keys.

The following table lists the different services provided by the toolkit with the type of access to keys or CSPs.

Table 5 Key and CSP Access

Service	Key or CSP	Type of Access
Encryption and decryption	Symmetric keys (AES and Triple-DES) Asymmetric key (RSA)	Read/Execute
Digital signature and verification	Asymmetric key (RSA)	Read/Execute
Message digest	None	N/A
MAC	HMAC keys	Read/Execute
Random number generation	CTR DRBG entropy, V, key, and init_seed	Read/Write/Execute
Key generation	Symmetric keys (AES and Triple-DES) Asymmetric key (RSA) MAC keys (HMAC)	Write
Key assurance	Asymmetric key (RSA)	Read
Self-test (Crypto Officer service)	Hardcoded key (HMAC)	Read/Execute
Show status	None	N/A
Zeroization	All	Read/Write

1.4.6 Key Wrapping

The module supports wrapping of raw key data and asymmetric and symmetric keys, with symmetric keys using the AES algorithm.

1.5 Cryptographic Algorithms

The module offers a wide range of cryptographic algorithms. This section describes the FIPS 140-2-approved or allowed algorithms that can be used when operating the module in a FIPS 140-2 compliant manner.

1.5.1 FIPS 140-2-approved Algorithms

The following table lists the BSAFE Crypto Module FIPS 140-2-approved algorithms, with their appropriate standards and CAVP validation certificate numbers.

Table 6 BSAFE Crypto Module FIPS 140-2-approved Algorithms

Algorithm Type	Algorithm	Standard	Validation Certificate
Symmetric Key	AES in:		#5941
	• CBC, ECB, and CTR modes (with 128, 192, and 256-bit key sizes)	SP 800-38A	
	• CCM mode (with 128, 192, and 256-bit key sizes)	SP 800-38C	
	• GCM mode with automatic Initialization Vector (IV) generation (with 128, 192, and 256-bit key sizes)	SP 800-38D	
	• XTS mode[1] (with 128 and 256-bit key sizes)	SP 800-38E	
	Triple-DES (three key) in ECB and CBC modes	SP 800-67 and SP 800-38A	#2895
Asymmetric Key	RSA (2048 to 4096-bit key size)		#3119
	• Key generation, signature and signature verification Key (modulus) sizes: 2048, 3072 bits	FIPS 186-4	
	• Signature verification only Key (modulus) sizes: 1024, 1536, 2048, 3072, 4096 bits	FIPS 186-2	
	• Key validation	SP 800-56B Rev 1	VA[3]
	RSASP1 (RSA signature primitive 1) component	FIPS 186-4	#2172
	RSAEP (RSA encryption primitive) component[2]	SP 800-56B Rev 1	VA
	RSADP (RSA decryption primitive) component	SP 800-56B Rev 1	#2173
	RSA-OAEP (RSA with Optimal Asymmetric Encryption Padding)	SP 800-56B Rev1	VA
KDFs	Password-based Key Derivation Function 2 (PBKDF2)[4]	SP 800-132	VA
Key Wrap	AES and AES padded (with 128, 192, and 256-bit key sizes)	SP 800-38F	#5941
Random Number	CTR DRBG (with AES-128, AES-192 and AES-256)	SP 800-90A rev1	#2494
Message Digest	SHA-1, SHA-2 (224, 256, 384, 512, 512-224, 512-256)	FIPS180-4	#4694
MAC	HMAC-SHA1, and HMAC-SHA2 (256, 384, and 512)	FIPS 198-1	#3916

[1] AES in XTS mode is only approved for hardware storage applications. The two keys concatenated to create the single double-length key must be checked to ensure they are different.

[2] This algorithm provides 112-128 bits of encryption strength.

[3] Vendor Affirmed - Not yet tested by the CAVP, but approved for use in FIPS 140-2 mode. RSA affirms correct implementation of the algorithm.

[4] As defined in NIST Special Publication 800-132, PBKDF2 can be used in FIPS 140-2 mode when used with FIPS 140-2-approved symmetric key and message digest algorithms. For more information, see Crypto User Guidance.

1.5.2 FIPS 140-2-allowed Algorithms

The following table lists the BSAFE Crypto Module FIPS 140-2-allowed algorithms, with appropriate standards.

Table 7 BSAFE Crypto Module FIPS 140-2-allowed Algorithms

Algorithm Type	Algorithm	Standard
Random Number	Non-deterministic Random Number Generator (NDRNG) Entropy source to seed the random number generator.	IG G.13

The following algorithm available in the module is **not allowable** for FIPS 140-2 usage. This algorithm **must not be used** when operating the module in a FIPS 140-2 compliant way:

- MD5

For more information about using the module in a FIPS 140-2-compliant manner, see Secure Operation of the Module.

1.6 Self-tests

The module performs a number of power-up and conditional self-tests to ensure proper operation.

If a power-up self-test fails, all cryptographic services for the library are disabled. Cryptographic services for the module can be re-enabled only by reloading the FIPS 140-2 module. If a conditional self-test fails, the operation fails but no services are disabled.

For self-test failures, power-up or conditional, the library notifies the user through the returns and error codes for the API.

1.6.1 Conditional Self-tests

The module performs the following conditional self-tests:

- A Continuous Random Number Generation (CRNG) test each time the toolkit produces random data, as per the FIPS 140-2 standard. The CRNG test is performed for the CTR DRBG and NDRNG (Entropy).
- A repetition count test and adaptive proportion test for the NDRNG (Entropy), as defined in SP 800-90B.
- A pair-wise consistency test each time the module generates an RSA public/private key pair.

1.6.2 Power-up Self-tests

Power-up self-tests are executed automatically when the module is loaded into memory. The power-up self-tests include the FIPS140-2 required Software Integrity Test and a set of Cryptographic Algorithms tests. The following Cryptographic Algorithm tests are implemented in the module:

- AES in CBC, CCM, CTR, ECB, GCM, and XTS mode, encrypt and decrypt KATs
- AES key wrap in KW and KWP mode, wrap and unwrap KATs
- CTR DRBG KAT and SP800-90A health tests
- HMAC SHA-1, and HMAC SHA-2 (256, 384, and 512) KATs
- RSA encrypt and decrypt KATs
- RSA sign and verify KATs
- SHA-1 KATs
- SHA-2 (256, 384, 512, 512-224, 512-256) KATs
- Software integrity test using HMAC verification.
- Triple-DES in ECB and CBC mode, encrypt and decrypt KAT.

Power-up self-tests are executed automatically when the module loads into memory.

1.6.3 Mitigation of Other Attacks

RSA key operations implement blinding, a reversible way of modifying the input data, so as to make the RSA operation immune to timing attacks. Blinding has no effect on the algorithm other than to mitigate attacks on the algorithm.

RSA signing operations implement a verification step after private key operations. This verification step, which has no effect on the signature algorithm, is in place to prevent potential faults in optimized Chinese Remainder Theorem (CRT) implementations. For more information, see https://eprint.iacr.org/2011/388.

2 Secure Operation of the Module

The following guidance must be followed in order to operate the module in a FIPS 140-2 mode of operation, in compliance with FIPS 140-2 requirements.

Note: The module operates as a Validated Cryptographic Module only when the rules for secure operation are followed.

2.1 Crypto User Guidance

The Crypto User must only use algorithms approved for use in a FIPS 140 mode of operation, as listed in BSAFE Crypto Module FIPS 140-2-approved Algorithms. The requirements for using the approved algorithms in a FIPS 140 mode of operation are as follows:

- The key length for an HMAC generation or verification must be between 112 and 4096 bits, inclusive. For HMAC verification, a key length greater than or equal to 80 and less than 112 is allowed for legacy-use.
- When using an approved DRBG to generate keys, the requested security strength for the DRBG must be at least as great as the security strength of the key being generated.
- When using GCM feedback mode for symmetric encryption, the authentication tag length and authenticated data length may be specified as input parameters, but the Initialization Vector (IV) must not be specified. It must be generated internally.

  The IV generated internally is fully random, generated by an approved PRNG, with a default length of 96 bits. No special considerations are required provided the system has sufficient entropy.
- In the case where the module is powered down, a new key must be used for AES GCM encryption/decryption.
- AES in XTS mode is approved only for hardware storage applications.
- The two keys used for XTS must be checked to ensure they are different. This check is performed automatically for the module.
- Keys used for digital signature generation and verification shall not be used for any other purpose.
- The length of an RSA key pair for digital signature generation must be greater than or equal to 2048 bits. For digital signature verification, the length must be greater than or equal to 2048 bits, however 1024 bits is allowed for legacy-use only. RSA keys shall have a public exponent of an odd number, equal to or greater than 65537.
- SHA1 is disallowed for the generation of digital signatures.

- For Password-based Key Derivation, the following restrictions apply:
  - Keys generated using PBKDF2 shall only be used in data storage applications.
  - The minimum password length is 14 characters, which has a strength of approximately 112 bits, assuming a randomly selected password using the extended ASCII printable character set is used.
  - For random passwords (that is, a string of characters from a given set of characters in which each character is equally likely to be selected), the strength of the password is given by: `S=L*(log N/log 2)` where `N` is the number of possible characters (for example, for the ASCII printable character set `N = 95`, for the extended ASCII printable character set `N = 218`) and `L` is the number of characters. A password of the strength `S` can be guessed at random with the probability of 1 in 2^S.
  - The minimum length of the randomly-generated portion of the salt is 16 bytes.
  - The iteration count is as large as possible, with a minimum of 1000 iterations recommended.
  - The maximum key length is $(2^{32} - 1) * b$, where `b` is the digest size of the message digest function.
  - Derived keys can be used as specified in NIST Special Publication 800-132, Section 5.4, options 1 and 2.
- The following restrictions apply to the use of three-key Triple-DES:
  - The use of three-key Triple-DES is approved.
  - The user is responsible for ensuring the same Triple-DES key has a limit of:
    - 2^{20} 64-bit data block encryptions when keys are generated as part of one of the recognized IETF protocols.
    - 2^{16} 64-bit data block encryptions otherwise.

  For more information about the use of three-key Triple-DES, see NIST Special Publication 800-67 revision 2 “Recommendations for The Triple Data Encryption Block Cipher”.

2.2 Roles

If a user of BSAFE Crypto Module needs to operate the toolkit in different roles, then the user must ensure all instantiated cryptographic objects are destroyed before changing from the Crypto User role to the Crypto Officer role, or unexpected results could occur.

The following table lists the roles a user can operate in.

Table 8 Crypto Module Roles

Role	Description
`R_FIPS140_ROLE_OFFICER`	An operator assuming the Crypto Officer role can call any BSAFE Crypto Module function. The complete list of the functionality available to the Crypto Officer is outlined in Services.
`R_FIPS140_ROLE_USER`	An operator assuming the Crypto User role can use the entire Crypto Module API except for `R_PROV_FIPS140_self_test_full()`, which is reserved for the Crypto Officer. The complete list of BSAFE Crypto Module functions is outlined in Services.

2.3 Modes of Operation

The module can operate in FIPS 140-2 mode or non-FIPS mode. The mode selected affects which algorithms are available for use:

- In FIPS 140-2 mode (`BCM_MODE_FIPS`), the module allows the cryptographic algorithms listed in BSAFE Crypto Module FIPS 140-2-approved Algorithms.
- In non-FIPS mode (`BCM_MODE_NON_FIPS`), the module allows all available cryptographic algorithms.

In each mode of operation, the complete set of services, which are listed in this Security Policy, are available to both the Crypto Officer and Crypto User roles, with the exception of `BCM_module_selftest()`, which is always reserved for the Crypto Officer.

Note: Cryptographic keys must not be shared between modes. For example, a key generated FIPS 140-2 mode must not be shared with an application running in a non-FIPS 140-2 mode.

2.4 Operating the Cryptographic Module

A driver using module is dynamically linked to the module file `rsabcm.sys`.

When the driver is loaded, the module file is automatically loaded by the operating system. `rsabcm.sys` is itself dynamically linked to a configuration library, `rsabcmcfg.sys` that is also automatically loaded by the operating system.

The configuration library can be customized by the module user. It returns configuration information for the module. This information includes the startup mode for the module, whether the module is to be in FIPS 140-2 mode or non-FIPS mode.

When started in FIPS 140-2 mode, the module's FIPS power-up self-tests run, the module is in FIPS 140-2 approved state and all FIPS 140-2 approved cryptographic algorithms are available.

When started in non-FIPS mode, the FIPS power-up self-tests are not run, the module is in a non-FIPS 140-2 approved state, and all cryptographic algorithms supported are available. Once loaded in non-FIPS mode the module cannot be used in a FIPS approved manner until it is unloaded then reloaded in FIPS 140-2 mode.

2.5 Deterministic Random Bit Generator

In all modes of operation, the module provides the CTR DRBG.

The module includes an allowed non-deterministic random number generator (NDRNG) (Entropy) used to generate seed material for the DRBG.

2.5.1 DRBG Seeding

The quality of the random data output from the DRBG depends on the quality of the supplied seeding (entropy). The module provides internal entropy collection from the CPU cycle counter and system time when available, for example. The module allows entropy from external sources to be supplied by an application callback. For information see `BCM_CONFIG.entropy_cb` in the *RSA BSAFE Crypto Module Developers Guide*.

When the DRBG is seeded, the quantity of entropy obtained depends on the DRBG configured for the module:

DRBG	Entropy Obtained (bits)
CTR DRBG with AES-256	256
CTR DRBG with AES-192	192
CTR DRBG with AES-128	128

Note: If entropy from external sources is added to an application, no assurances are made about the minimum strength of generated keys.

3 Acronyms

The following table lists the acronyms used with the module and their definitions:

Table 9 Acronyms and Definitions

Term	Definition
AES	Advanced Encryption Standard. A fast symmetric key algorithm with a 128-bit block, and keys of lengths 128, 192, and 256 bits. Replaces DES as the US symmetric encryption standard.
API	Application Programming Interface.
Attack	Either a successful or unsuccessful attempt at breaking part or all of a cryptosystem. Various attack types include an algebraic attack, birthday attack, brute force attack, chosen ciphertext attack, chosen plaintext attack, differential cryptanalysis, known plaintext attack, linear cryptanalysis, and middle person attack.
CBC	Cipher Block Chaining. A mode of encryption in which each ciphertext depends upon all previous ciphertexts. Changing the Initialization Vector (IV) alters the ciphertext produced by successive encryptions of an identical plaintext.
CMVP	Cryptographic Module Validation Program
CRNG	Continuous Random Number Generation.
CTR	Counter mode of encryption, which turns a block cipher into a stream cipher. It generates the next keystream block by encrypting successive values of a counter.
CTR DRBG	Counter mode Deterministic Random Bit Generator.
DES	Data Encryption Standard. A symmetric encryption algorithm with a 56-bit key. See also Triple DES.
DRBG	Deterministic Random Bit Generator.
ECB	Electronic Codebook. A mode of encryption, which divides a message into blocks and encrypts each block separately.
Encryption	The transformation of plaintext into an apparently less readable form (called ciphertext) through a mathematical process. The ciphertext can be read by anyone who has the key and decrypts (undoes the encryption) the ciphertext.
FIPS	Federal Information Processing Standards.
GCM	Galois/Counter Mode. A mode of encryption combining the Counter mode of encryption with Galois field multiplication for authentication.
HMAC	Keyed-Hashing for Message Authentication Code.

Table 9 Acronyms and Definitions

Term	Definition
IV	Initialization Vector. Used as a seed value for an encryption operation.
KAT	Known Answer Test.
Key	A string of bits used in cryptography, allowing people to encrypt and decrypt data. Can be used to perform other mathematical operations as well. Given a cipher, a key determines the mapping of the plaintext to the ciphertext. The types of keys include distributed key, private key, public key, secret key, session key, shared key, subkey, symmetric key, and weak key.
Key wrapping	A method of encrypting key data for protection on untrusted storage devices or during transmission over an insecure channel.
MD5	A message digest algorithm, which hashes an arbitrary-length input into a 16-byte digest. Designed as a replacement for MD4.
NDRNG	Non-deterministic random number generator.
NIST	National Institute of Standards and Technology. A division of the US Department of Commerce (formerly known as the NBS) which produces security and cryptography-related standards.
OS	Operating System.
PBKDF2	Password-based Key Derivation Function 2. A method of password-based key derivation, which applies a Message Authentication Code (MAC) algorithm to derive the key.
PC	Personal Computer.
privacy	The state or quality of being secluded from the view or presence of others.
private key	The secret key in public key cryptography. Primarily used for decryption but also used for encryption with digital signatures.
PRNG	Pseudo-random Number Generator.
RNG	Random Number Generator.
RSA	Public key (asymmetric) algorithm providing the ability to encrypt data and create and verify digital signatures. RSA stands for Rivest, Shamir, and Adleman, the developers of the RSA public key cryptosystem.
SHA	Secure Hash Algorithm. An algorithm, which creates a unique hash value for each possible input. SHA takes an arbitrary input, which is hashed into a 160-bit digest.
SHA-1	A revision to SHA to correct a weakness. It produces 160-bit digests. SHA-1 takes an arbitrary input, which is hashed into a 20-byte digest.

Table 9 Acronyms and Definitions

Term	Definition
SHA-2	The NIST-mandated successor to SHA-1, to complement the Advanced Encryption Standard. It is a family of message digest algorithms (SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, and SHA-512/256), which produce digests of 224, 256, 384, 512, 224, and 256 bits respectively.
Triple-DES	A variant of DES, which uses three 56-bit keys.
XTS	XEX-based Tweaked Codebook mode with ciphertext stealing. A mode of encryption used with AES.