FIPS PUB 140-2 Security Policy for SAP CommonCryptoLib Crypto Kernel v8.4.47.0

Document version: 2.5
Date: 2017-03-24

TABLE OF CONTENT

This document contains the non-proprietary Security Policy according to FIPS PUB 140-2 for the cryptographic module “SAP CommonCryptoLib Crypto Kernel” in its version 8.4.47.0. The cryptographic module is a shared library providing various cryptographic functions, which has been validated according to overall Security Level 1 on various platforms (see section 1.5 hereinafter).

1 CRYPTOGRAPHIC MODULE DESCRIPTION

1.1 Overview

The cryptographic module is the "SAP CommonCryptoLib Crypto Kernel" in its version 8.4.47.0. It is a shared library, i.e. it consists of software only. On Windows platforms the cryptographic module consists of the dynamic link library file `slcryptokernel.dll` accompanied by the file `slcryptokernel.dll.sha256` containing a HMAC-SHA-256 reference value for the software/firmware integrity test. On Linux and UNIX based platforms it consists of the shared library file `libslcryptokernel.<extension>,` accompanied by the file `libslcryptokernel. <extension>.sha256` containing the HMAC-SHA-256 reference value for the software/firmware integrity test (where "<extension>" stands for the OS-specific extension for shared libraries; extensions are "sl" for HP-UX and "so" or "o" for the other Linux/Unix operating systems used). The cryptographic module provides an API in terms of C++ methods for key management and operation of the following cryptographic functions:

Approved cryptographic functions

- AES symmetric cipher encryption and decryption according to [FIPS 197], supporting key lengths 128 bit, 192 bit and 256 bit and block cipher modes CBC, CTR, ECB, OFB, CFB8 and CFB128 according to [SP 800-38A] and GCM according to [SP 800-38D] (Certs. #3665 and #3666)
- Triple-DES (3TDES, i.e. using key length 168 bit) symmetric cipher encryption and decryption according to [SP 800-67] and block cipher modes CBC, CTR, ECB, OFB, CFB8 and CFB64 according to [SP 800-38A] (Certs. #2047 and #2048)
- DSA signature generation and verification according to [FIPS 186-4] (Certs. #1035 and #1036)
- RSA signature generation and verification according to [FIPS 186-4] (supporting signature schemes RSASSA-PSS and RSASSA-PKCS1-V1_5 from [PKCS#1 V2.1]) (Certs. #1898 and #1899)
- ECDSA signature generation component (signature Generation of hash sized messages) with curves P-521, P-384, P-256 or P-224 according to [FIPS 186-4] (CVL Certs. #672 and #675)
- ECDSA signature verification with curves P-521, P-384, P-256, P-224 or P-192 according to [FIPS 186-4] (Certs. #772 and #773)
- HMAC generation according to [FIPS 198-1] (Certs. #2415 and #2416)
- CTR_DRBG (using AES-256) deterministic random number generation according to [SP 800-90A] and key generation for all of the approved cryptographic functions listed above (Certs. #986 and #987)
- SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 hash functions according to [FIPS 180-4] (Certs. #3083 and #3084)
- ECC CDH primitive (KASECC DLC primitive) according to [FIPS186-4] section 5.7.1.2 (CVL Certs. #670 and #673)
- RSADP primitive according to [FIPS 186-4] (CVL Certs. #671 and #674)

Non-Approved cryptographic functions, which are allowed in the Approved mode of operation

- Diffie-Hellman (key agreement; key establishment methodology provides between 112 and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength[1])
- RSA (key wrapping; key establishment methodology provides between 112 and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength[1])
- Entropy collection (used for generation of seeds for the Approved random number generator (CTR_DRBG) listed above)

[1] The module supports arbitrary encryption strengths for Diffie-Hellman key agreement and RSA key wrapping.

Non-Approved cryptographic functions, which are not allowed in the Approved mode of operation

- DES, 2TDES (2-key Triple-DES, non-compliant), IDEA, RC2, RC5-32 symmetric cipher encryption and decryption
- RC4 stream cipher encryption and decryption
- ElGamal key wrapping
- MD2, MD4, MD5, RIPEMD-128 and RIPEMD-160 hash functions

1.2 Architecture

To be able to provide its cryptographic services, the cryptographic module is subdivided as follows:

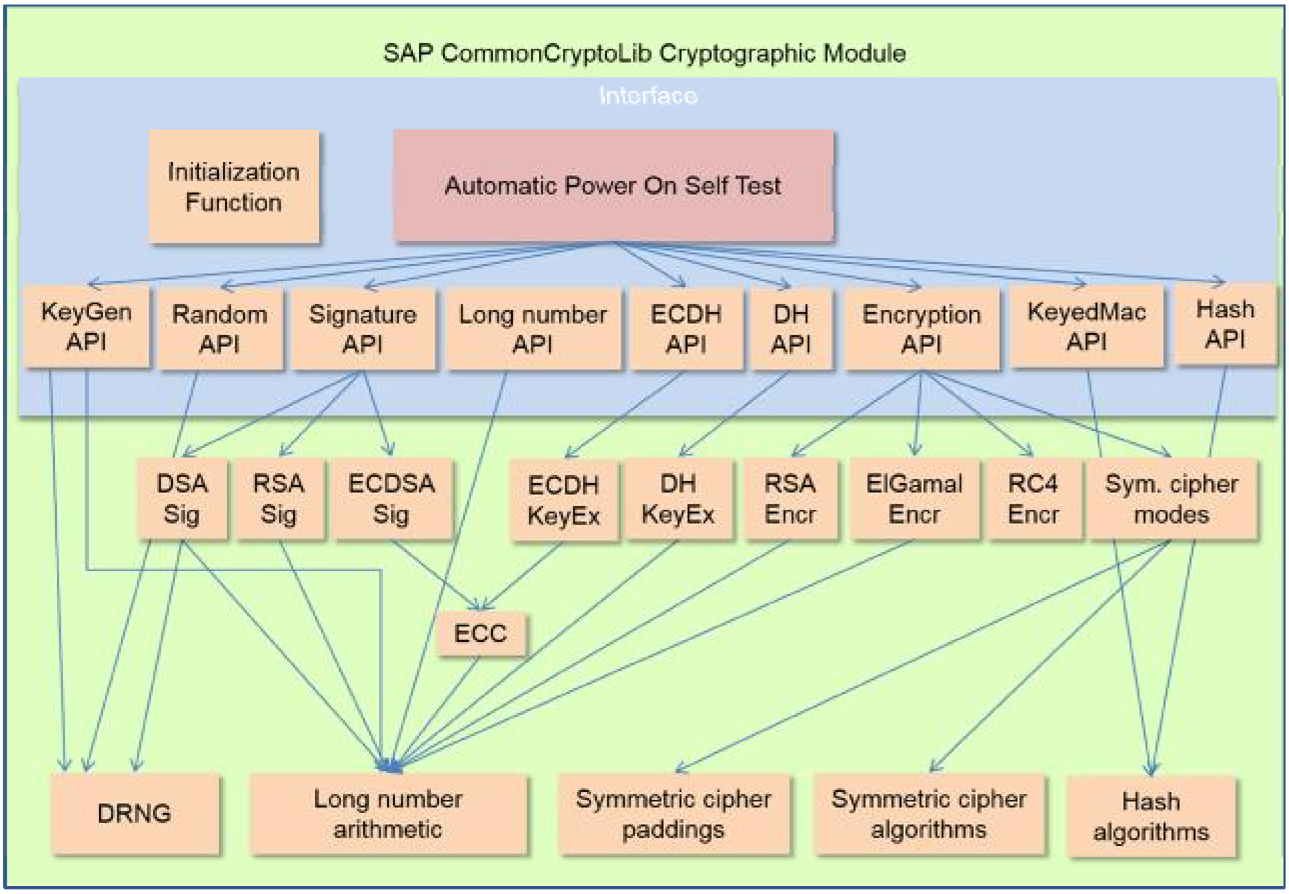

Figure 1: Block diagram of the cryptographic module

1.3 Ports and interfaces of the module

As the cryptographic module is a software library, its logical interfaces are realized in terms of a set of APIs, which are shown in Figure 1 hereinabove. All functionality of the cryptographic module is made available to the calling application (i.e. the user of the cryptographic module) in terms of exported API functions. Some of the API functions are also used internally, e.g. the self-test service makes use of some of those API functions when performing cryptographic algorithm self-tests. For a full reference of all API functions please see guidance document “FIPS PUB 140-2 Cryptographic Ports and Interfaces of SAP CommonCryptoLib Crypto Kernel” as provided by SAP AG together with the cryptographic module.

In the meaning of FIPS 140-2 the cryptographic module is of type multiple-chip standalone, therefore the physical ports of the module are identical to the ports (e.g. power port, data input/output ports) of the system the cryptographic module is used on.

1.4 Usage

The cryptographic module is used by a single operator, which is the application using the library by calling its methods. The module does not employ functions for identification and authentication of its operator. It supports a crypto officer role and a user role, whereas the crypto officer role is authorized to load and initialize the cryptographic module, and the user role is authorized to perform cryptographic operations. The roles are implicitly selected by the calling application in terms of methods being called. Typically, the cryptographic module is used by the “SAP CommonCryptoLib”, which itself is a library providing cryptographic protocols and services to an application using it. The cryptographic module in context of the other parts of the “SAP CommonCryptoLib” is shown in the following figure. Please take note that the cryptographic module can also be used without the other parts of the “SAP CommonCryptoLib”, i.e. it can also be used directly by an application.

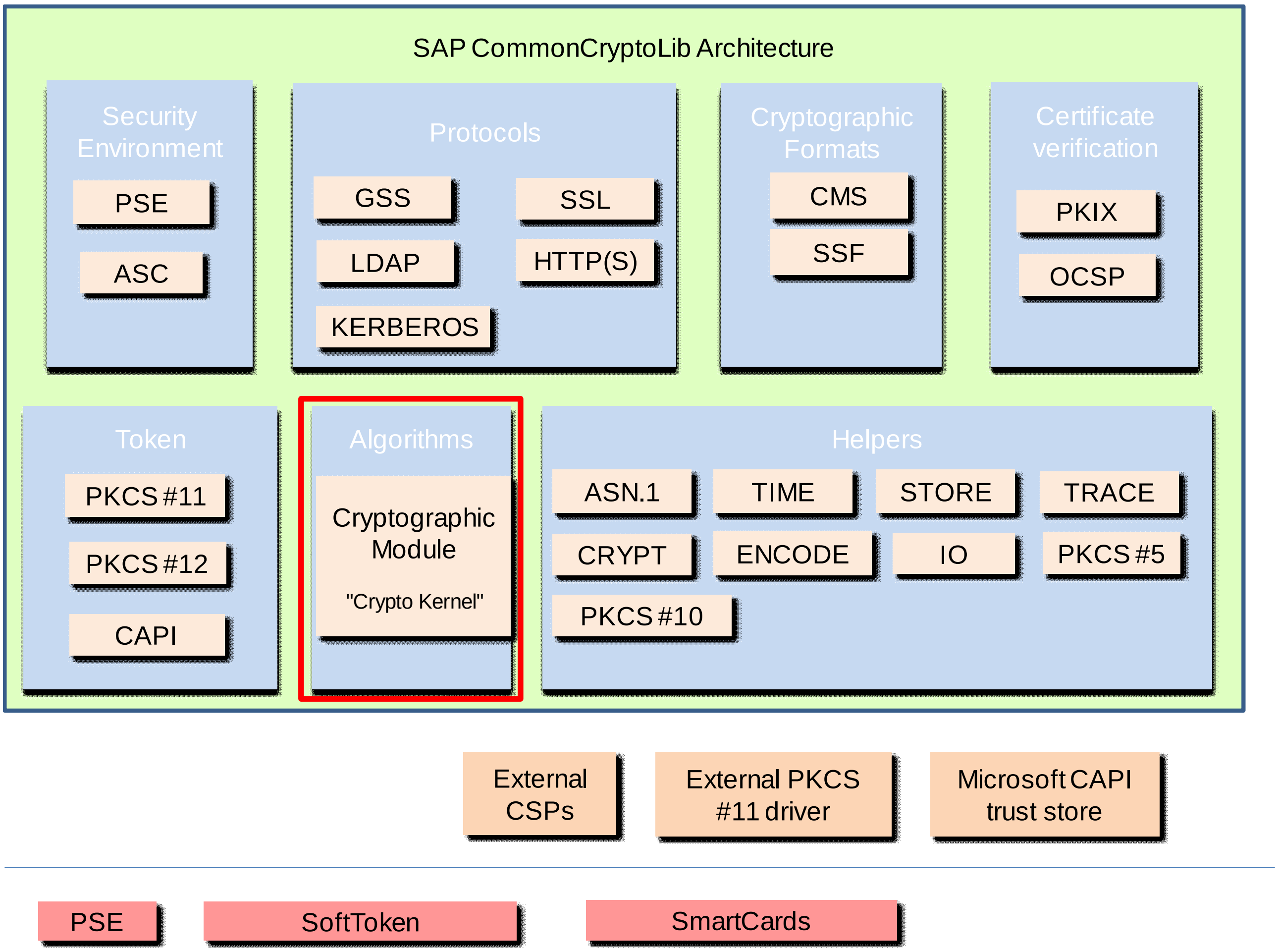

Figure 2: The cryptographic module (indicated by red frame) in the context of SAP CommonCryptoLib

1.5 Supported platforms

The cryptographic module can be used on a great variety of different platforms (i.e. in different operational environments in the meaning of FIPS 140-2). It has been tested and validated on the following platforms:

Operating system and version	Hypervisor ("-" means no hypervisor used)	Processor with or without PAA (if applicable)	Module Executable
AIX 5.2 64-bit	-	PowerPC without VCIPHER	64-bit
AIX 6.1 64-bit	IBM PowerVM 2.2	PowerPC without VCIPHER	32-bit and 64-bit
HPUX 11.11 PA-RISC 64-bit	-	PA-RISC 2.0	64-bit
HPUX 11.23 ia64 64-bit	-	IA64	64-bit
HPUX 11.31 ia64 64-bit	-	IA64	32-bit and 64-bit
Linux 2.6.5 x86_64 64-bit	-	x86_64 without AES-NI	32-bit and 64-bit
Linux 2.6.32 x86 32-bit	-	x86 without AES-NI	32-bit
Linux 2.6.32 ia64 64-bit	-	ia64	64-bit
Linux 2.6.32 Power 64-bit	IBM PowerVM 2.2	PowerPC without VCIPHER	32-bit and 64-bit
Linux 3.0.101 Power 64-bit	IBM PowerVM 2.2	PowerPC with VCIPHER	64-bit
Linux 3.0.101 x86_64 64-bit	Vmware ESXi 5.1.0	x86_64 with AES-NI	32-bit and 64-bit
Linux 3.0.101 zSeries 64-bit	IBM z/VM 6.2.0	z/Architecture (s390x)[2]	64-bit
SunOS 5.9 SPARC 64-bit	-	SPARC	64-bit
SunOS 5.10 SPARC 64-bit	-	SPARC	32-bit and 64-bit
SunOS 5.10 x64 64-bit	-	x86_64 without AES-NI	32-bit and 64-bit
Windows Server 2008 SP2 64-bit	-	x86_64 without AES-NI	32-bit and 64-bit
Windows Server 2008 R2 SP1 64-bit	Vmware ESXi 5.1.0	x86_64 with AES-NI	32-bit and 64-bit

For further details, please see the list of test configurations for the SAP CommonCryptoLib Crypto Kernel 8.4.47.0 at http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm.

1.6 Logical and cryptographic boundary

The logical boundary of the module encloses the shared library (including its executable code and data).

The physical boundary, i.e. the "cryptographic boundary" of the module as defined by FIPS PUB 140-2 is made up by the housing of the workstation or server hardware the cryptographic module is running on, i.e. in the meaning of FIPS PUB 140-2 the cryptographic module has got the type "multiple chip standalone".

[2] z/Architecture (s390x) processor supports "CP Assist for Cryptographic Function" (CPACF), but this PAA is not used by the module regardless whether present or not (the module supports the cryptographic algorithms natively in both cases).

2 SECURITY LEVEL

The cryptographic module fulfills overall Security Level 1 according to FIPS PUB 140-2.

The ratings of the individual requirement sections are as follows:

Requirement section of FIPS PUB 140-2	Rating
Cryptographic Module Specification	Security Level 1
Cryptographic Module Ports and Interfaces	Security Level 1
Roles, Services and Authentication	Security Level 1
Finite State Model	Security Level 1
Physical Security	Not applicable
Operational Environment	Security Level 1
Cryptographic Key Management	Security Level 1
EMI/EMC	Security Level 1
Self-Tests	Security Level 1
Design Assurance	Security Level 1
Mitigation of Other Attacks	Not applicable

3 APPROVED MODE OF OPERATION

The cryptographic module implements several non-Approved cryptographic functions as listed in section 1.1 hereinabove, but it does not technically enforce an Approved mode of operation (also known as FIPS mode), which would disable these non-Approved functions. Instead, the operator has to obey the following instructions to operate the cryptographic module in its Approved mode (i.e. the developer of the application using the cryptographic module has to obey these):

- None of the “Non-Approved cryptographic functions, which are not allowed in the Approved mode of operation” listed in section 1.1 hereinabove shall be used in the Approved mode of operation (unless it is part of an allowed cryptographic function, like MD5 as part of SSL v3.1 or TLS).
- Triple-DES encryption is restricted to full key length of 168 bit (i.e. 3TDES).
- RSA signature generation shall only be used with modulus size supporting between 112 and 256 bit of encryption strength (see NIST SP 800-57 part 1, table 2 for corresponding key lengths) in the Approved mode of operation.
- DSA signature generation shall only be used with private key size supporting between 112 and 256 bit of encryption strength (see NIST SP 800-57 part 1, table 2 for corresponding key lengths) in the Approved mode of operation.
- ECDSA signature generation shall be used with curves P-521, P-384, P-256 or P-224 in the Approved mode of operation. Curve P-192 shall not be used for signature generation in the Approved mode of operation. Curve P-192 may be used by applications in the Approved mode of operation for digital signature verification.
- ECC CDH primitive (KASECC DLC primitive) shall be used with curves P-521, P-384, P-256 or P-224 in the Approved mode of operation. Curve P-192 shall not be used for ECC CDH primitive in the Approved mode of operation.
- SHA-1 shall not be used for digital signature generation or hash-only applications in the Approved mode of operation (SHA-1 may be used in the Approved mode of operation for digital signature verification and keyed hashing, i.e. HMAC).
- Diffie-Hellman key agreement and RSA key wrapping are allowed functions in the Approved mode of operation as long as these are used with key lengths supporting between 112 bit and 256 bit of encryption strength (see NIST SP 800-57 part 1, table 2 for corresponding key lengths).

4 CRYPTOGRAPHIC MODULE SECURITY POLICY

4.1 Roles

The cryptographic module supports the minimum two roles as required by FIPS PUB 140-2, this is a

- crypto officer role (allowed to load, initialize and power-up self-test the module), and a
- user role (allowed to perform cryptographic operations including related key entry and output).

As the cryptographic module does not allow the operator to perform maintenance services, it does not support a maintenance role.

4.2 Services

The services employed by the cryptographic module are as follows:

- Loading
- Initialization
- Power-up and conditional self-tests
- Block cipher encryption and decryption using AES or Triple-DES
- Signature generation and verification using RSA, DSA or ECDSA
- HMAC generation
- Random number generation and key generation
- Cryptographic hashing
- Zeroization (no corresponding API call; implicitly performed when key objects are deleted)
- Show status
- Asymmetric encryption and decryption
- Key transport using RSA key wrapping
- Key agreement using Diffie-Hellman (DH) or ECC CDH primitive

All services above, which use cryptographic functions, may be Approved or non-Approved, depending on whether an Approved or non-Approved function is used (only the asymmetric encryption/decryption and the key exchange are always non-Approved). For some of the services there are restrictions concerning the key length or the underlying elliptic curves to be used in Approved mode, see the corresponding keys in following section 4.3 with references to section 3 explaining the restrictions for Approved or allowed usage.

4.3 Cryptographic keys and other CSPs

According to the cryptographic functions supported by the module the following keys are supported by the cryptographic module in its Approved mode of operation:

- AES-128, AES-192- and AES-256 keys
- Triple-DES (3TDES) keys
- RSA private and public keys (see section 3 for key length restrictions)
- DSA private and public keys (see section 3 for key length restrictions)
- ECDSA private and public keys (see section 3 for elliptic curve restrictions)
- ECDH private and public keys (for use with ECC CDH primitive; see section 3 for elliptic curve restrictions)
- DH private and public keys (see section 3 for key length restrictions)
- HMAC keys

There are no other CSPs (e.g., authentication secrets or similar) supported by the cryptographic module in the Approved mode of operation.

4.4 Identification and authentication (I&A) policy

The cryptographic module does not implement operator identification and authentication mechanisms. Crypto officer role and user role are implicitly selected by the operator in terms of the API call used to perform the corresponding service.

4.5 Access control policy

The following tables list the services, which the roles are authorized to perform in the Approved mode of operation, as well as the cryptographic keys and other CSPs, which are accessed while the services are performed in the Approved mode of operation.

Role	Authorized Services
Crypto officer	· Loading · Initialization · Power-up self-tests
User	· Block cipher encryption/decryption · Signature generation/verification · Key transport · Key agreement · HMAC generation · Key generation · Export and import of cryptographic objects · Hashing · Conditional self-tests · Zeroization (no corresponding API call; implicitly performed when key objects are deleted)

Service	Accessed cryptographic keys	Type(s) of access
Loading	None	
Initialization	None	
Self-test	None	
Block cipher encryption/decryption	AES-128/192/256 key, or Triple-DES (3TDES) key	Write, Execute
Signature generation	RSA private key, ECDSA private key, or DSA private key	Write, Execute
Signature verification	RSA public key, ECDSA public key or DSA public key	Write, Execute
Key transport	RSA private key and RSA public key	Write, Execute
Key agreement	DH private key and DH public key, ECDH private key and ECDH public key	Write, Execute
HMAC generation	HMAC key	Write, Execute
Key generation	Any of the keys listed above	Write
Key export	Any of the keys listed above	Read
Hashing	None	
Zeroization	Any of the keys listed above	Write (overwrite)

The self-test service includes the following set of self-tests (in the following “KAT” denotes to “known-answer test”).

Power-up self-tests:

- Software/firmware integrity, tested by verification of a HMAC-SHA-256;
- AES-128, AES-192 and AES-256, each tested by a KAT for encryption and a KAT for decryption;
- Triple-DES (3TDES), tested by a KAT for encryption and a KAT for decryption;
- RSA signature generation and verification, tested by a KAT for signature generation and a subsequent verification;
- DSA signature generation and verification, tested by a pairwise consistency test;
- ECDH/ECDSA key pair generation
- ECDSA signature generation and verification, tested by a pairwise consistency test;
- ECC CDH primitive, tested by a pairwise consistency test;
- DH key agreement, tested by a pairwise consistency test;
- CTR_DRBG, tested by a KAT;
- SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 (and all non-approved hash functions as well), each tested by a KAT;
- HMAC generation, tested by a KAT.

Conditional self-tests:

- CTR_DRBG, tested by a continuous random number generator test;
- Entropy source provided by the platform, tested by a continuous random number generator test (only applicable while seeding the CTR_DRBG);
- RSA key pair generation, tested by two pairwise consistency tests;
- DSA key pair generation, tested by a pairwise consistency test;
- ECDSA key pair generation, tested by a pairwise consistency test;
- ECDH key pair generation, tested by a pairwise consistency test;
- ElGamal key pair generation, tested by a pairwise consistency test.

In case of a failure in at least one of the power-up self-tests the cryptographic module returns an error code to the application that tried to load and initialize the module, and none of the services of the module are available to the application in this case.

In case of a failure during a conditional self-test the cryptographic module returns an error code to the application that tried to perform a corresponding function. Access of the application to generated random numbers or key pairs that caused the conditional self-test error is inhibited in this case.

4.6 Physical security policy

When run on one of the test platforms as indicated in section 1.5 hereinabove, the cryptographic module meets the physical security requirements of FIPS PUB 140-2 security level 1 (i.e. the hardware has to be production grade).
No physical security mechanisms concerning tamper evidence or tamper detection and response are employed by the module, and therefore no actions are required by the operator(s) to ensure that physical security is maintained.

4.7 Security policy for mitigation of other attacks

The cryptographic module does not implement measures to mitigate attacks other than those already addressed by functionality required by FIPS PUB 140-2 Security Level 1.

5 REFERENCES

[FIPS 140-2] NIST PUB 140-2, Security Requirements for Cryptographic Modules, May 25, 2001

[FIPS 180-4] Federal Information Processing Standards Publication 180-4, Secure Hash Standard (SHS), August 2015

[FIPS 186-4] Federal Information Processing Standards Publication 186-4, Digital Signature Standard (DSS), July 2013

[FIPS 186-4] Federal Information Processing Standards Publication 186-4, Elliptic Curve Digital Signature Algorithm Validation System (ECDSA2VS), March 2014

[FIPS 197] Federal Information Processing Standards Publication 197, Advanced Encryption Standard (AES), November 26, 2001

[FIPS 198-1] Federal Information Processing Standards Publication 198-1, The Keyed-Hash Message Authentication Code (HMAC), July 2008

[PKCS#1 V2.1] PKCS #1 v2.1: RSA Cryptography Standard, RSA Laboratories, June 14, 2002

[SP 800-38A] NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation, Methods and Techniques, December 2001

[SP 800-38A Add] Addendum to NIST Special Publication 800 -38A, Recommendation for Block Cipher Modes of Operation: Three Variants of Ciphertext Stealing for CBC Mode, October 2010

[SP 800-38D] NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007

[SP 800-56A] NIST Special Publication 800-56A Revision 2, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, May 2013

[SP 800-67r1] NIST Special Publication 800-67 Revision 1, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Version 1.2, revised January 2012

[SP 800-90Ar1] NIST Special Publication 800-90A Revision 1, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015

www.sap.com

© 2015 SAP SE or an SAP affiliate company. All rights reserved.
No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.
SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. Please see http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark for additional trademark information and notices. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors.
National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.
In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.