SUSE Linux Enterprise Libgcrypt Cryptographic Module

version 3.2

FIPS 140-3 Non-Proprietary Security Policy

Version 1.1

Last update: 2024-06-19

Prepared by:

atsec information security corporation

4516 Seton Center Parkway, Suite 250

Austin, TX 78759

www.atsec.com

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Table of Contents

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

1 General

This document is the non-proprietary FIPS 140-3 Security Policy for version 3.2 of the SUSE Linux Enterprise Libgcrypt Cryptographic Module. It has a one-to-one mapping to the [SP 800-140B] starting with section B.2.1 named “General” that maps to section 1 in this document and ending with section B.2.12 named “Mitigation of other attacks” that maps to section 12 in this document.

ISO/IEC 24759 Section 6. [Number Below]	FIPS 140-3 Section Title	Security Level
1	General	1
2	Cryptographic Module Specification	1
3	Cryptographic Module Interfaces	1
4	Roles, Services, and Authentication	1
5	Software/Firmware Security	1
6	Operational Environment	1
7	Physical Security	N/A
8	Non-invasive Security	N/A
9	Sensitive Security Parameter Management	1
10	Self-tests	1
11	Life-cycle Assurance	1
12	Mitigation of Other Attacks	1

Table 1 - Security Levels

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

2 Cryptographic Module Specification

2.1 Module Embodiment

The SUSE Linux Enterprise Libgcrypt Cryptographic Module (hereafter referred to as “the module”) is a Software multi-chip standalone cryptographic module.

2.2 Module Design, Components, Versions

The software block diagram below shows the cryptographic boundary of the module, and its interfaces with the operational environment.

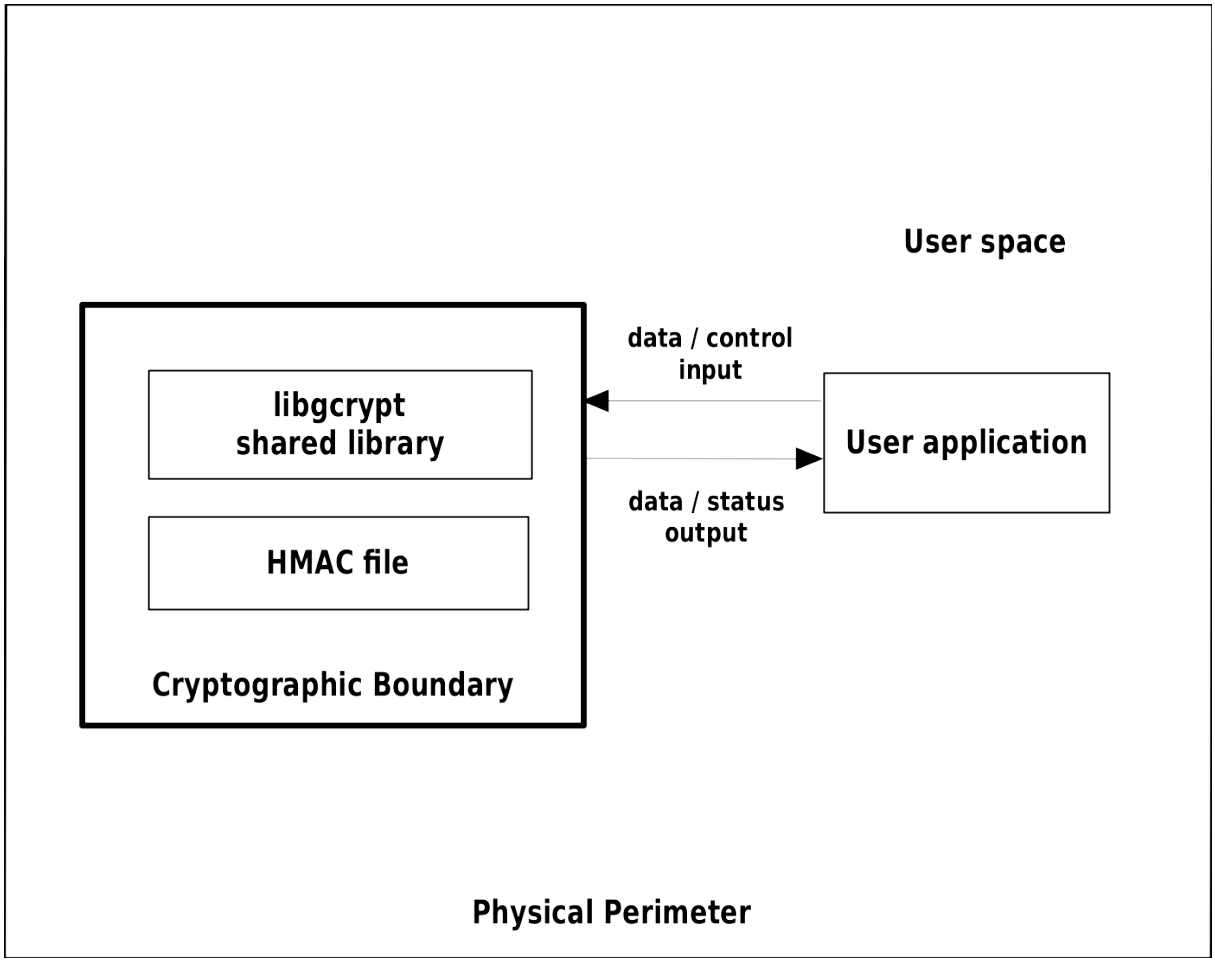

Figure 1 – Cryptographic Boundary

Table 2 lists the software components of the cryptographic module, which defines its cryptographic boundary.

Components	Description
libgcrypt.so.20.3.4	Shared library for cryptographic algorithms.
.libgcrypt.so.20.hmac	Integrity check HMAC value for the libgcrypt shared library.

Table 2 – Cryptographic Module Components

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

2.3 Modes of Operation

Only if the integrity test passed successfully, the module transitions to the operational state. No operator intervention is required to reach this point. The module operates in the approved mode of operation by default and can only transition into the non-approved mode by calling one of the non-approved services listed in Table 10. Please see section 4 for the details on service indicator provided by the module that identifies when an approved service is called.

2.4 Tested Operational Environments

The module has been tested on the following platforms with the corresponding module variants and configuration options:

#	Operating System	Hardware Platform	Processor	PAA/Acceleration
1	SUSE Linux Enterprise Server 15 SP4	Supermicro Super Server SYS-6019P-WTR	Intel® Xeon® Silver 4215R	With and without AES-NI (PAA)
2	SUSE Linux Enterprise Server 15 SP4	GIGABYTE R181-Z90-00	AMD EPYC™ 7371	With and without AES-NI (PAA)
3	SUSE Linux Enterprise Server 15 SP4	GIGABYTE G242-P32-QZ	ARM Ampere® Altra® Q80-30	With and without Cryptography Extensions (PAA)
4	SUSE Linux Enterprise Server 15 SP4	IBM z/15	z15	With and without CPACF (PAI)
5	SUSE Linux Enterprise Server 15 SP4 on PowerVM (VIOS 3.1.4.00)	IBM Power E1080 (9080-HEX)	Power10	With and without ISA (PAA)

Table 3 - Tested Operational Environments

2.5 Vendor-Affirmed Operational Environments

In addition to the platforms listed in Table 3, SUSE has also tested the module on the platforms in Table 4, and claims vendor affirmation on them.
Note: the CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

#	Operating System	Hardware platform	Processor	PAA/Acceleration
1	SUSE Linux Enterprise Server 15SP4	IBM LinuxONE III LT1	z15	With and without CPACF (PAI)
2	SUSE Linux Enterprise Micro 5.3	Supermicro Super Server SYS-6019P-WTR	Intel® Xeon® Silver 4215R	With and without AES-NI (PAA)

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

#	Operating System	Hardware platform	Processor	PAA/Acceleration
3	SUSE Linux Enterprise Micro 5.3	GIGABYTE R181-Z90-00	AMD EPYC™ 7371	With and without AES-NI (PAA)
4	SUSE Linux Enterprise Micro 5.3	GIGABYTE G242-P32-QZ	ARM Ampere® Altra® Q80-30	With and without Cryptography Extensions (PAA)
5	SUSE Linux Enterprise Micro 5.3	IBM z/15	z15	With and without CPACF (PAI)
6	SUSE Linux Enterprise Micro 5.3	IBM LinuxONE III LT1	z15	With and without CPACF (PAI)
7	SUSE Linux Enterprise Server for SAP 15SP4	Supermicro Super Server SYS-6019P-WTR	Intel® Xeon® Silver 4215R	With and without AES-NI (PAA)
8	SUSE Linux Enterprise Server for SAP 15SP4	GIGABYTE R181-Z90-00	AMD EPYC™ 7371	With and without AES-NI (PAA)
9	SUSE Linux Enterprise Server for SAP 15SP4	IBM Power E1080 (9080-HEX)	Power10	With and without ISA (PAA)
10	SUSE Linux Enterprise Base Container Image 15SP4	Supermicro Super Server SYS-6019P-WTR	Intel® Xeon® Silver 4215R	With and without AES-NI (PAA)
11	SUSE Linux Enterprise Base Container Image 15SP4	GIGABYTE R181-Z90-00	AMD EPYC™ 7371	With and without AES-NI (PAA)
12	SUSE Linux Enterprise Base Container Image 15SP4	GIGABYTE G242-P32-QZ	ARM Ampere® Altra® Q80-30	With and without Cryptography Extensions (PAA)
13	SUSE Linux Enterprise Base Container Image 15SP4	IBM z/15	z15	With and without CPACF (PAI)
14	SUSE Linux Enterprise Base Container Image 15SP4	IBM LinuxONE III LT1	z15	With and without CPACF (PAI)
15	SUSE Linux Enterprise Base Container Image 15SP4	IBM Power E1080 (9080-HEX)	Power10	With and without ISA (PAA)
16	SUSE Linux Enterprise Desktop 15SP4	Supermicro Super Server SYS-6019P-WTR	Intel® Xeon® Silver 4215R	With and without AES-NI (PAA)
17	SUSE Linux Enterprise Desktop 15SP4	GIGABYTE R181-Z90-00	AMD EPYC™ 7371	With and without AES-NI (PAA)
18	SUSE Linux Enterprise Real Time 15SP4	Supermicro Super Server SYS-6019P-WTR	Intel® Xeon® Silver 4215R	With and without AES-NI (PAA)

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

#	Operating System	Hardware platform	Processor	PAA/Acceleration
19	SUSE Linux Enterprise Real Time 15SP4	GIGABYTE R181-Z90-00	AMD EPYC™ 7371	With and without AES-NI (PAA)

Table 4 - Vendor-Affirmed Operational Environments

2.6 Approved Algorithms

Table 5 below lists all security functions of the module, including specific key strengths employed for approved services, and implemented modes of operation. The following are allowed for legacy use only: Digital signature verification using ECDSA with a SHA-1.

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A3022 A3023 A3025 A3026	AES FIPS197, SP800-38A	CBC	128, 192, 256-bit keys with 128-256 bits key strength	Symmetric encryption; Symmetric decryption
	AES FIPS197, SP800-38A	CFB8, CFB128	128, 192, 256-bit keys with 128-256 bits key strength	Symmetric encryption; Symmetric decryption
	AES SP800-38C	CCM	128, 192, 256-bit keys with 128-256 bits key strength	Key wrapping; Key unwrapping
	AES SP800-38B	CMAC	128, 192, 256-bit keys with 128-256 bits key strength	Message authentication code (MAC)
	AES FIPS197, SP800-38A	CTR	128, 192, 256-bit keys with 128-256 bits key strength	Symmetric encryption; Symmetric decryption
	AES FIPS197, SP800-38A	ECB	128, 192, 256-bit keys with 128-256 bits key strength	Symmetric encryption; Symmetric decryption
	AES FIPS197, SP800-38F	KW	128, 192, 256-bit keys with 128-256 bits key strength	Key wrapping; Key unwrapping
	AES FIPS197, SP800-38A	OFB	128, 192, 256-bit keys with 128-256 bits key strength	Symmetric encryption; Symmetric decryption
	AES SP800-38E	XTS	128, 256-bit keys with 128 and 256 bits key strength	Symmetric encryption and symmetric decryption (for data storage)

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
Vendor Affirmed	CKG SP 800-133Rev2 section 4 IG D.H	RSA FIPS 186-4	2048, 3072, 4096-bit keys with 112-149 bits key strength	Key generation
		ECDSA FIPS 186-4	P-224, P-256, P-384, P-521 with 112-256 bits key strength	
A3022 A3023 A3025 A3026	CTR_DRBG SP800-90Arev1	AES-128, AES-192, AES-256 with DF, with/without PR	128, 192, 256-bit AES keys with 128-256 bits key strength	Random number generation
A3022 A3023 A3024 A3025 A3026	Hash_DRBG SP800-90Arev1	SHA-1, SHA-256, SHA-512 with/without PR	N/A	
A3022 A3023 A3024 A3025 A3026	HMAC_DRBG SP800-90Arev1	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512 with/without PR	112 or more-bit HMAC keys with 112 bits key strength or greater	
A3022 A3023 A3024 A3025 A3026	ECDSA FIPS186-4	FIPS 186-4 Appendix B.4.2 Testing Candidates	P-224, P-256, P-384, P-521 with 112-256 bits key strength	Key generation
		N/A	P-224, P-256, P-384, P-521 with 112-256 bits key strength	Public key verification
		SHA-224, SHA-256, SHA-384, SHA-512	P-224, P-256, P-384, P-521 with 112-256 bits key strength	Digital signature generation
		SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	P-224, P-256, P-384, P-521 with 112-256 bits key strength	Digital signature verification
E31	Non-Physical Entropy Source SP 800-90B	N/A	Entropy input with 256-bit strength	Random number generation
A3021 A3022 A3023 A3024 A3025 A3026 A3027	HMAC FIPS198-1	SHA-1	112 or more-bit keys with 112 bits key strength or greater	Message authentication code (MAC)
A3022 A3023 A3024 A3025 A3026		SHA-224, SHA-256, SHA-384, SHA-512	112 or more-bit keys with 112 bits key strength or greater	Message authentication code (MAC)

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A3024 A3025 A3026		SHA3-224, SHA3-256, SHA3-384, SHA3-512	112 or more-bit keys with 112 bits key strength or greater	
A3022 A3023 A3025 A3026	KTS SP800-38F SP800-38C FIPS IG D.G	AES in KW mode	128, 192, 256-bit keys with 128-256 bits key strength	Key wrapping; Key unwrapping
		AES in CCM mode	128, 192, 256-bit keys with 128-256 bits key strength	Key wrapping; Key unwrapping
A3022 A3023 A3024 A3025 A3026	PBKDF2 SP800-132	Option 1a with SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512	Password: N/A; derived key with 112-256 bits key strength	Key derivation
A3022 A3023 A3024 A3025 A3026	RSA FIPS186-4	B.3.3 Random Probable Primes	2048, 3072, 4096 with 112-149 bits key strength	Key generation
		PKCS#1v1.5: SHA-224, SHA-256, SHA-384, SHA-512	2048, 3072, 4096 with 112-149 bits key strength	Digital signature generation
		PSS: SHA-224, SHA-256, SHA-384, SHA-512	2048, 3072, 4096 with 112-149 bits key strength	
		PKCS#1v1.5: SHA-224, SHA-256, SHA-384, SHA-512	2048, 3072, 4096 with 112-149 bits key strength	Digital signature verification
		PSS: SHA-224, SHA-256, SHA-384, SHA-512	2048, 3072, 4096 with 112-149 bits key strength	
A3024 A3025 A3026	SHA-3 FIPS202	SHA3-224, SHA3-256, SHA3-384, SHA3-512, SHAKE-128, SHAKE-256	N/A	Message digest

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A3021 A3022 A3023 A3024 A3025 A3026 A3027	SHS FIPS180-4	SHA-1	N/A	Message digest
A3022 A3023 A3024 A3025 A3026		SHA-224, SHA-256, SHA-384, SHA-512	N/A	Message digest

Table 5 - Approved Algorithms

2.7 Non-Approved Algorithms Allowed in the Approved Mode of Operation

The module does not implement non-approved algorithms that are allowed in the approved mode of operation.

2.8 Non-Approved Algorithms Allowed in the Approved Mode of Operation with No Security Claimed

The module does not implement non-approved algorithms that are allowed in the approved mode of operation.

2.9 Non-Approved Algorithms Not Allowed in the Approved Mode of Operation

Table 6 lists non-approved algorithms that are not allowed in the approved mode of operation. These algorithms are used by the non-approved services listed in Table 10.

Algorithm/Functions	Use/Function
AES EAX	Symmetric encryption; Symmetric decryption
AES GCM	Symmetric encryption; Symmetric decryption
AES GMAC	Message authentication code (MAC)
AES OCB	Symmetric encryption; Symmetric decryption
BLAKE2B-160, BLAKE2B-256, BLAKE2B-384, BLAKE2B-512, BLAKE2S-128, BLAKE2S-160, BLAKE2S-224, BLAKE2S-256	Message digest
CRC32	Error detection code
ElGamal	Key generation
GOST R 34.11	Message digest
MD4, MD5	Message digest
PBKDF2 with non-approved message digest algorithms or using input parameters not	Key derivation

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

meeting requirements stated in section 11.2.3	
RIPEMD-160	Message digest
RSA OAEP	Key encapsulation
Tiger	Message digest
SM3, STRIBOG-256, STRIBOG-512	Message digest
Whirlpool	Message digest

Table 6 - Non-Approved Not Allowed in the Approved Mode of Operation

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

3 Cryptographic Module Ports and Interfaces

As a software-only module, the module does not have physical ports. The operator can only interact with the module through the API provided by the module. Thus, the physical ports are interpreted to be the physical ports of the hardware platform on which the module runs.
All data output via data output interface is inhibited when the module is performing pre-operational test or zeroization or when the module enters error state.

Logical Interface[1]	Data that passes over port/interface
Data Input	API input parameters for data.
Data Output	API output parameters for data.
Control Input	API function calls, API input parameters for control input, /proc/sys/crypto/fips_enabled control file.
Status Output	API return codes, API output parameters for status output.

Table 7 - Ports and Interfaces

[1] The control output interface is omitted on purpose because the module does not implement it.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

4 Roles, services and authentication

4.1 Roles

The module supports the Crypto Officer role only. This sole role is implicitly assumed by the operator of the module when performing a service. The module does not support authentication.

Role	Service	Input	Output
Crypto Officer (CO)	Digital signature generation	Private key, message, hash algorithm	Signature
	Digital signature verification	Signature, message, hash algorithm, public key	Signature verification result
	Error detection code	None	Code
	Key generation	Key size	Key pair
	Key derivation	Password or passphrase	Derived key
	Key encapsulation	Key encapsulating key, key to be encapsulated	Encapsulated key
	Key unwrapping	Wrapped key, key unwrapping key	Unwrapped key
	Key wrapping	Key wrapping key, key to be wrapped	Wrapped key
	Message authentication code (MAC)	Message, key	Message authentication code
	Message digest	Message	Message digest
	On-demand integrity test	None	Return codes/log messages
	Public key verification	Key	Return codes/log messages
	Random number generation	Size	Random number
	Symmetric decryption	Ciphertext, key	Plaintext
	Symmetric encryption	Plaintext, key	Ciphertext
	Show version	N/A	Name and version information
	Show status	N/A	Module status
	Self-test	N/A	Pass/fail results of self-tests
	Zeroization	Any SSP	N/A

Table 8 - Roles, Service Commands, Input and Output

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

The module provides services to the users that assume one of the available roles. All services are shown in Table 9 and Table 10.

4.2 Services

Table 9 lists the approved services. For each service, the table lists the associated cryptographic algorithm(s), the role to perform the service, the cryptographic keys or CSPs involved, and their access type(s). The following convention is used to specify access rights to a CSP:

- **G** = **Generate**: The module generates or derives the SSP.
- **R** = **Read**: The SSP is read from the module (e.g., the SSP is output).
- **W** = **Write**: The SSP is updated, imported, or written to the module.
- **E** = **Execute**: The module uses the SSP in performing a cryptographic operation.
- **Z** = **Zeroise**: The module zeroises the SSP.
- **N/A**: the calling application does not access any CSP or key during its operation.

The details of the approved cryptographic algorithms including the CAVP certificate numbers can be found in Table 5. The module implements dedicated functions based on the type of requested service to indicate whether it utilizes an approved security function or not. The specific functions are as listed below

1. _gcry_fips_indicator_cipher - For symmetric algorithms
2. _gcry_fips_indicator_kdf - For KDF functions
3. _gcry_fips_indicator_pk - For asymmetric functions.
4. _gcry_fips_indicator_hash - For digest functions
5. _gcry_fips_indicator_mac - For MAC functions
6. drbg_generate – For DRBG functions

For all approved services, the output of the function will be "0” indicating the service is approved.

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Cryptographic Services						
Symmetric encryption	Encrypt a plaintext	AES	AES key	CO	W, E	_gcry_fips_indicator_cipher function returns “0”
Symmetric decryption	Decrypt a ciphertext	AES	AES key		W, E	_gcry_fips_indicator_cipher function returns “0”
Key generation	Generate a key pair	RSA, DRBG, CKG	RSA public key, RSA private key, Intermediate key generation value		G, R	rsa_generate and _gcry_fips_indicator_pk functions returns “0”
		ECDSA, DRBG, CKG	ECDSA public key, ECDSA private key, Intermediate key generation value		G, R	_gcry_fips_indicator_pk function returns “0”

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Digital signature generation	Generate a signature	RSA, SHS, DRBG	RSA private key		W, E	rsa_sign and _gcry_fips_indicator_pk functions returns “0”
		ECDSA, DRBG, SHS	ECDSA private key		W, E	_gcry_fips_indicator_pk function returns “0”
Digital signature verification	Verify a signature	RSA, SHS	RSA public key		W, E	rsa_verify and _gcry_fips_indicator_pk functions returns “0”
		ECDSA, SHS	ECDSA public key		W, E	_gcry_fips_indicator_pk function returns “0”
Public key verification	Verify a public key	ECDSA	ECDSA public key		W, E	_gcry_fips_indicator_pk function returns “0”
Random number generation	Generate random bitstrings	CTR_DRBG	Entropy input		W, E	drbg_generate function returns “0”
			DRBG seed		E, G	
			DRBG Internal state (V, Key)		W, E, G	
		Hash_DRBG	Entropy input		W, E	
			DRBG seed		E, G	
			DRBG Internal state (V, C)		W, E, G	
		HMAC_DRBG	Entropy input		W, E	
			DRBG seed		E, G	
			DRBG Internal state (V, Key)		W, E, G	
Message digest	Compute a message digest	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512	N/A		N/A	_gcry_fips_indicator_hash function returns “0”
XOF	Compute the output of an XOF	SHAKE-128, SHAKE-256	N/A		N/A	_gcry_fips_indicator_hash function returns “0”

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Message authentication code (MAC)	Compute a MAC tag	HMAC	HMAC key		W, E	_gcry_fips_indicator_mac function returns “0”
		AES CMAC	AES key		W, E	
Key wrapping	Perform AES-based key wrapping	AES-KW, AES-CCM	AES key		W, E	_gcry_fips_indicator_cipher function returns “0”
Key unwrapping	Perform AES-based key unwrapping	AES-KW, AES-CCM	AES key		W, E	_gcry_fips_indicator_cipher function returns “0”
Key derivation	Derive a key from a password	PBKDF2	Password or passphrase		W, E	_gcry_fips_indicator_kdf function returns “0”
			Derived key		G, R	
On demand integrity test	Perform the integrity test on demand	HMAC-SHA-256	N/A		N/A	None
Other FIPS-related Services						
Show status	Return the module status	N/A	N/A	CO	N/A	None
Zeroization	Zeroize all SSPs	N/A	Any SSP		Z	
Self-tests	Perform selft-tests	AES, CMAC, DRBG, Non-Physical Entropy Source, ECDSA, HMAC, RSA, SHS, PBKDF2 See Table 13 for additional details	AES keys HMAC keys RSA public key RSA private key ECDSA public key ECDSA private key Intermediate key generation value Password or passphrase Entropy input		E	
			Derived key DRBG seed DRBG Internal state (V, Key) DRBG Internal state (V, C)		E, G	
Show version	Return the name and version information	N/A	N/A		N/A	

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Table 9 - Approved Services

Table 10 lists the non-approved services. The details of the non-approved cryptographic algorithms available in non-approved mode can be found in Table 6.

Service	Description	Algorithms Accessed	Role
Cryptographic Services			
Symmetric encryption	AES encryption using non-approved AES modes	AES (GCM, EAX and OCB)	CO
Symmetric decryption	AES decryption using non-approved AES modes		
Message authentication code (MAC)	Message authentication	AES GMAC	
Key derivation	PBKDF2 Key derivation	PBKDF2 with non-approved message digest algorithms or using input parameters not meeting requirements stated in section 11.2.3	
Key encapsulation	Encapsulate a key	RSA OAEP	
Key generation	Generate a key pair	ElGamal	
Message digest	Message digest using non-approved algorithms	MD5, MD4, GOST R 34.11, RIPEMD-160, Tiger, Whirlpool, SM3, STRIBOG-256, STRIBOG-512, BLAKE2B-160, BLAKE2B-256, BLAKE2B-384, BLAKE2B-512, BLAKE2S-128, BLAKE2S-160, BLAKE2S-224 and BLAKE2S-256	
Error detection code	Error detection code	CRC32	

Table 10 - Non-Approved Services

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

5 Software/Firmware security

5.1 Integrity Techniques

The integrity of the module is verified by comparing an HMAC-SHA-256 value calculated at run time with the HMAC value stored in the .hmac file that was computed at build time for each software component of the module. If the HMAC values do not match, the test fails and the module enters the error state.

5.2 On-Demand Integrity Test

Integrity tests are performed as part of the Pre-Operational Self-Tests.
The module provides the Self-Test service to perform self-tests on demand which includes the pre-operational tests (i.e., integrity test) and cryptographic algorithm self-tests (CASTs). This service can be invoked by using the gcry_control(GCRYCTL_SELFTEST) API function call or by powering-off and reloading the module. During the execution of the on-demand self-tests, services are not available, and neither data input nor output is possible.
In order to verify whether the self-tests have succeeded and the module is in the Operational state, the calling application may invoke the gcry_control(GCRYCTL_OPERATIONAL_P). The function will return TRUE if the module is in the operational state, FALSE if the module is in the Error state.

5.3 Executable Code

The module consists of executable code in the form of libgcrypt file as stated in Table 2.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

6 Operational Environment

6.1 Applicability

This module operates in a modifiable operational environment per the FIPS 140-3 level 1 specifications. The SUSE Linux Enterprise Server operating system is used as the basis of other products. Compliance is maintained for SUSE products whenever the binary is found unchanged per the vendor affirmation from SUSE based on the allowance FIPS 140-3 management manual section 7.9.1 bullet 1 a i).
Note: The CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when supported if the specific operational environment is not listed on the validation certificate.

6.2 Policy

Instrumentation tools like the ptrace system call, gdb and strace utilities, as well as other tracing mechanisms offered by the Linux environment such as ftrace or systemtap, shall not be used in the operational environment. The use of any of these tools implies that the cryptographic module is running in a non-tested operational environment.

6.3 Requirements

The module shall be installed as stated in section 11. The operating system provides process isolation and memory protection mechanisms that ensure appropriate separation for memory access among the processes on the system. Each process has control over its own data and uncontrolled access to the data of other processes is prevented.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

7 Physical Security

The module is comprised of software only, and therefore this section is not applicable.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

8 Non-invasive Security

This module does not implement any non-invasive security mechanism, and therefore this section is not applicable.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

9 Sensitive Security Parameter Management

Table 11 summarizes the Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module.

Key/SSP Name/ Type	Strength	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroization	Use & related keys
AES keys (CSP)	AES-XTS: 128, 256 bits Rest of the modes: 128, 192, 256 bits	AES-CBC, AES-CCM, AES-CFB128, AES-CFB8, AES-CMAC, AES-CTR, AES-KW, AES-OFB, AES-XTS A3022 A3023 A3025 A3026	N/A	MD/EE **Import:** API input parameters From: Operator calling application (TOEPP) To: Cryptographic module **Export:** None	N/A	RAM	gcry_cipher_close(), gcry_free()	**Use:** Symmetric encryption; Symmetric decryption; Message authentication code (MAC); Key wrapping; Key unwrapping **Related SSPs:** N/A
HMAC keys (CSP)	112 or greater bits	HMAC A3021 A3022 A3023 A3024 A3025 A3026 A3027			N/A	RAM	gcry_mac_close(), gcry_free()	**Use:** Message Authentication Code (MAC) **Related SSPs:** N/A
RSA public key (PSP)	112, 128, 149 bits	RSA A3022 A3023 A3024 A3025 A3026	Generated using method B.3.3 specified in FIPS 186-4; random values are obtained from the SP800-90Arev1 DRBG.	MD/EE **Import:** API input parameters From: Operator calling application (TOEPP) To: Cryptographic module **Export:** API output parameters From: Cryptographic module To: Operator calling application (TOEPP)	N/A	RAM	gcry_sexp_release(), gcry_mpi_release(), gcry_free()	**Use:** Digital signature verification; Key generation **Related SSPs:** RSA private key, Intermediate key generation value
RSA private key (CSP)	112, 128, 149 bits				N/A	RAM	gcry_sexp_release(), gcry_mpi_release(), gcry_free()	**Use:** Digital signature generation; Key generation **Related SSPs:** RSA public key, Intermediate key generation value

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Key/SSP Name/ Type	Strength	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroization	Use & related keys
ECDSA public key (PSP)	112, 128, 192, 256 bits	ECDSA A3022 A3023 A3024 A3025 A3026	Generated using method B.4.2 specified in FIPS 186-4; random values are obtained from the SP800-90Arev1 DRBG.	MD/EE **Import:** API input parameters From: Operator calling application (TOEPP) To: Cryptographic module **Export:** API output parameters From: Cryptographic module To: Operator calling application (TOEPP)	N/A	RAM	gcry_sexp_release(), gcry_mpi_release(), gcry_ctx_release(), gcry_mpi_point_release(), gcry_free()	**Use:** Digital signature verification; Key generation; Public key verification **Related SSPs:** ECDSA private key, Intermediate key generation value
ECDSA private key (CSP)	112, 128, 192, 256 bits				N/A	RAM	gcry_sexp_release(), gcry_mpi_release(), gcry_ctx_release(), gcry_mpi_point_release(), gcry_free()	**Use:** Digital signature generation; Key generation; Public key verification **Related SSPs:** ECDSA public key, Intermediate key generation value
Intermediate key generation value (CSP)	112-256 bits	CKG vendor affirmed	SP 800-133r2 Section 4, 5.1, and 5.2	**Import:** None **Export:** None	N/A	RAM	Automatic	**Use:** Key generation **Related SSPs:** ECDSA public key, ECDSA private key, RSA public key, RSA private key
Password or passphrase (CSP)	N/A	PBKDF2 A3022 A3023 A3024 A3025 A3026	N/A	MD/EE **Import:** API input parameters From: Operator calling application (TOEPP) To: Cryptographic module **Export:** None	N/A	RAM	gcry_free()	**Use:** Key derivation **Related SSPs:** Derived key
Derived key (CSP)	N/A	PBKDF2 A3022 A3023 A3024 A3025 A3026	SP 800-133r2, Section 6.2	MD/EE **Import:** None **Export:** API output parameters From: Cryptographic module	N/A	RAM	gcry_free()	**Use:** Key derivation **Related SSPs:** Password or passphrase

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Key/SSP Name/ Type	Strength	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroization	Use & related keys
				To: Operator calling application (TOEPP)				
Entropy input (CSP) IG D.L Compliant	256 bits	Non-Physical Entropy Source E31	Obtained from Non-Physical Entropy Source	**Import:** None **Export:** None	N/A	RAM	gcry_ctrl(GCRYCTL_TERM_SECMEM)	**Use:** Random number generation **Related SSPs:** DRBG seed
DRBG seed (CSP) IG D.L Compliant	CTR_DRBG: 128, 192, 256 bits Hash_DBRG: 128, 256 bits HMAC_DRBG: 128, 256 bits	CTR_DRBG A3022 A3023 A3025 A3026 Hash_DRBG, HMAC_DRBG A3022 A3023 A3024 A3025 A3026	CTR_DRBG Hash_DRBG HMAC_DRBG	**Import:** None **Export:** None	N/A	RAM	gcry_ctrl(GCRYCTL_TERM_SECMEM)	**Use:** Random number generation **Related SSPs:** Entropy input, DRBG Internal state (V, Key), DRBG Internal state (V, C)
DRBG Internal state (V, Key) (CSP) IG D.L Compliant		CTR_DRBG A3022 A3023 A3025 A3026 HMAC_DRBG A3022 A3023 A3024 A3025 A3026	CTR_DRBG HMAC_DRBG	**Import:** None **Export:** None	N/A	RAM	gcry_ctrl(GCRYCTL_TERM_SECMEM)	**Use:** Random number generation **Related SSPs:** DRBG seed
DRBG Internal state (V, C) (CSP) IG D.L Compliant		Hash_DRBG A3022 A3023 A3024 A3025 A3026	Hash_DRBG	**Import:** None **Export:** None	N/A	RAM	gcry_ctrl(GCRYCTL_TERM_SECMEM)	**Use:** Random number generation **Related SSPs:** DRBG seed

Table 11 - SSPs

9.1 Random Number Generation

The module employs a Deterministic Random Bit Generator (DRBG) based on [SP800-90Arev1] for the creation of RSA and ECDSA keys, RSA and ECDSA signature generation. In addition, the module provides a Random Number Generation service to calling applications.
The DRBG supports the Hash_DRBG, HMAC_DRBG and CTR_DRBG mechanisms. The DRBG is initialized during module initialization; the module loads by default the DRBG using the HMAC_DRBG mechanism with SHA-256 and without prediction resistance. A different DRBG mechanism can be chosen by invoking the gcry_control(GCRYCTL_DRBG_REINIT) function.
The module uses an [SP800-90B]-compliant entropy source specified in Table 12. This entropy source is located within the cryptographic boundary. The module obtains 384 bits to seed the DRBG, and 256 bits to reseed it, sufficient to provide a DRBG with 256 bits of security strength.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Entropy Source	Minimum number of bits of entropy	Details
SP 800-90B compliant Non-Physical Entropy Source (ESV cert. E31)	256 bits of entropy in 256-bit output	The Libgcrypt CPU Time Jitter RNG version 3.4.0 entropy source (with SHA-3 as the vetted conditioning component) is located within the module's cryptographic boundary.

Table 12 - Non-Deterministic Random Number Generation Specification

9.2 SSP Generation

The module provides an [SP800-90Arev1]-compliant Deterministic Random Bit Generator (DRBG) for the creation of key components of asymmetric keys, and random number generation.
The Cryptographic Key Generation (CKG) methods implemented in the module for Approved services in approved mode are compliant with section 5.1 of [SP800-133rev2] and with IG C.H.
For generating RSA and ECDSA keys the module implements asymmetric key generation services compliant with [FIPS186-4]. A seed (i.e., the random value) used in asymmetric key generation is directly obtained from the [SP800-90Arev1] DRBG.

9.3 SSP Transport

The module provides the following key transport mechanisms:

- AES key wrapping using AES-KW.
- AES key wrapping using AES-CCM.

According to Table 2: Comparable strengths in [SP 800-57rev5], the key sizes of AES provide the following security strength in approved mode of operation:

- AES key wrapping in KW mode provides 128, 192, 256-bit keys with key strength between 128-256 bits in compliance with IG D.G.
- AES key wrapping using AES-CCM provides 128, 192, 256-bit keys with key strength between 128-256 bits in compliance with IG D.G.

9.4 SSP Derivation

The module supports password-based key derivation (PBKDF2). The implementation is compliant with option 1a of [SP-800-132]. Keys derived from passwords or passphrases using this method can only be used in storage applications.

9.5 SSP Entry and Output

The module does not support direct manual SSP entry or intermediate SSP generation output. The SSPs are provided to the module via API input parameters in plaintext form and output via API output parameters in plaintext form within the physical perimeter of the operational environment. This is allowed by [FIPS140-3_IG] 9.5.A, according to the "CM Software to/from App via TOEPP Path" entry on the Key Establishment Table.

9.6 SSP Storage

The module does not perform persistent storage of SSPs. The SSPs are temporarily stored in the RAM in plaintext form. SSPs are provided to the module by the calling process and are destroyed when released by the appropriate zeroization function calls.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

9.7 SSP Zeroization

The memory occupied by SSPs is allocated by regular memory allocation operating system calls. The application that is acting as the CO is responsible for calling the appropriate zeroization functions provided in the module's API and listed in Table 11. Calling gcry_free() will zeroize the SSPs and also invoke the corresponding API functions listed in Table 11 to zeroize SSPs. The zeroization functions overwrite the memory occupied by SSPs with “zeros” and deallocate the memory with the regular memory deallocation operating system call. The completion of a zeroization routine(s) will indicate that a zeroization procedure succeeded.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

10 Self-tests

The module performs pre-operational tests automatically when the module is loaded into memory; pre-operational tests ensure that the module is not corrupted, and the CASTs ensure that the cryptographic algorithms work as expected. While the module is executing the pre-operational tests and CASTs, services are not available, and input and output are inhibited. The module is not available for use by the calling application until the pre-operational tests are completed successfully. After the pre-operational test and the CASTs succeed, the module becomes operational. If any of the pre-operational test or any of the CASTs fail an error message is returned, and the module transitions to the error state.

10.1 Pre-Operational Tests

The module performs the integrity test using HMAC-SHA-256. The details of integrity test are provided in section 5.1.

10.2 Conditional Tests

10.2.1 Cryptographic algorithm tests

Table 13 specifies the CASTs performed by the module. All CASTs performed are in the form of the Known Answer Tests (KATs) and are run prior to performing the integrity test. A KAT includes the comparison of a calculated output with an expected known answer, hard coded as part of the test vectors used in the test. If the values do not match, the KAT fails.

Algorithm	Condition	Test
AES	Power on	KAT AES ECB mode with 128, 192 and 256-bits keys, encryption, and decryption (separately tested).
CMAC	Power on	KAT AES CMAC with 128-bit key, MAC generation.
DRBG	Power on	KAT CTR_DRBG with AES with 128-bit key with DF, with and without PR. KAT Hash_DRBG with SHA-256 with and without PR. KAT Hash_DRBG with SHA-1 without PR. KAT HMAC_DRBG with HMAC-SHA-256 with and without PR. Health tests according to section 11.3 of [SP800-90Arev1]
Non-Physical Entropy Source	Power on	NIST SP800-90B Entropy source start-up test RCT and APT with 1024 samples
	Continuous	NIST SP800-90B Entropy source continuous test: RCT with Cutoff C = 31; APT with Cutoff C = 325; W = 512
ECDSA	Power on	KAT ECDSA signature generation and verification with P-256 and SHA-256 (separately tested).
HMAC	Power on	KAT HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512. KAT HMAC-SHA3-224, HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Algorithm	Condition	Test
RSA	Power on	KAT RSA PKCS#1 v1.5 signature generation and verification with 2048-bit key and SHA-256 (separately tested).
SHS	Power on	KAT SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512.
PBKDF2	Power on	KAT PBKDF2 with SHA-1 and SHA-256.

Table 13 – Conditional Cryptographic Algorithms Self-Tests

10.2.2 Pairwise Consistency Test

The module performs the Pair-wise Consistency Tests (PCT) shown in Table 14. If at least one of the tests fails, the module returns an error code and enters the Error state. When the module is in the Error state, no data is output, and cryptographic operations are not allowed.

Algorithm	Test
ECDSA key generation	PCT using signature generation and verification with SHA-256.
RSA key generation	PCT using signature generation and verification with SHA-256.

Table 14 - Pairwise Consistency Test

10.2.3 Periodic/On-Demand Self-Test

On-Demand self-tests can be invoked by using gcry_control(GCRYCTL_SELFTEST) API function call or by powering-off and reloading the module which cause the module to run both the pre-operational self-tests and conditional self-tests again. During the execution of the on-demand self-tests, services are not available, and neither data input nor output is possible.
In order to verify whether the self-tests have succeeded and the module is in the Operational state, the calling application may invoke the gcry_control(GCRYCTL_OPERATIONAL_P). The function will return TRUE if the module is in the operational state, FALSE if the module is in the Error state.

10.3 Error States

When the module fails any pre-operational self-test or conditional test, the module will return an error code to indicate the error and will enter the Error state (i.e., Self-test error state). Additionally, when random numbers are requested in the error state or cipher operations are requested on a deallocated handle, the module is aborted, enters the Error state (i.e., Abort error state) and is not available for use. Any further cryptographic operation is inhibited. When the module is in Self-test Error state, the calling application can obtain the module state by calling the gcry_control(GCRYCTL_OPERATIONAL_P) API function. The function returns FALSE to indicate that the module is in the Error state. Otherwise, the function returns TRUE to indicate that the module is in the Operational state. In the Abort Error state the module is aborted and is not available for use, therefore, the module state cannot be obtained. In both Error states, all data output is inhibited, and no cryptographic operation is allowed. The Error states can be recovered by a restart (i.e., powering off and powering on) of the module. The following table shows the error codes and the corresponding condition:

Error State	Cause of Error	Status Indicator
Self-test error state	Failure of pre-operational tests or conditional tests.	An error message related to the cause of the failure.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Error State	Cause of Error	Status Indicator
Abort error state	Random numbers are requested in the error state or cipher operations are requested on a deallocated handle.	The module is aborted and is not available for use.

Table 15 - Error States

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

11 Life-cycle assurance

11.1 Delivery and Operation

11.1.1 Module Installation

The Crypto Officer can install the RPM packages containing the module as listed in Table 17 using the zypper tool as follows.

```
# zypper install libgcrypt20
# zypper install libgcrypt20-hmac
```

The integrity of the RPM package is automatically verified during the installation, and the Crypto Officer shall not install the RPM package if there is any integrity error.

11.1.2 Operating Environment Configuration

The operating environment needs to be configured to support FIPS, so the following steps shall be performed with the root privilege:

1. Install the dracut-fips RPM package:

```
# zypper install dracut-fips
```

2. Recreate the INITRAMFS image:

```
# dracut -f
```

3. After regenerating the initrd, the Crypto Officer has to append the following parameter in the /etc/default/grub configuration file in the GRUB_CMDLINE_LINUX_DEFAULT line:

```
fips=1
```

4. After editing the configuration file, please run the following command to change the setting in the boot loader:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

If /boot or /boot/efi resides on a separate partition, the kernel parameter boot=<partition of /boot or /boot/efi> must be supplied. The partition can be identified with the command "df /boot" or "df /boot/efi" respectively. For example:

```
# df /boot
Filesystem     1K-blocks    Used    Available    Use%    Mounted on
/dev/sda1      233191       30454   190296       14%                    /boot
```

The partition of /boot is located on /dev/sda1 in this example. Therefore, the following string needs to be appended in the aforementioned grub file:

```
"boot=/dev/sda1"
```

5. Reboot to apply these settings.

Now, the operating environment is configured to support FIPS operation. The Crypto Officer should check the existence of the file /proc/sys/crypto/fips_enabled, and verify it contains a numeric value “1”. If the file does not exist or does not contain “1”, the operating environment is not configured to support FIPS and the module will not operate as a FIPS validated module properly.

11.1.3 Module Installation for Vendor Affirmed Platforms

Table 16 includes the information on module installation process for the vendor affirmed platforms that are listed in Table 4.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Product	Link
SUSE Linux Enterprise Micro 5.3	https://documentation.suse.com/sle-micro/5.3/single-html/SLE-Micro-security/#sec-fips-slemicro-install
SUSE Linux Enterprise Server for SAP 15SP4	https://documentation.suse.com/sles/15-SP4/html/SLES-all/book-security.html
SUSE Linux Enterprise Base Container Image 15SP4	https://documentation.suse.com/smart/linux/html/concept-bci/index.html
SUSE Linux Enterprise Desktop 15SP4	https://documentation.suse.com/sled/15-SP4/html/SLED-all/book-security.html
SUSE Linux Enterprise Real Time 15SP4	https://documentation.suse.com/sle-rt/15-SP4/

Table 16 - Installation for Vendor Affirmed Platforms

Note: Per section 7.9 in the FIPS 140-3 Management Manual [FIPS140-3_MM], 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 this module is ported and executed in an operational environment not listed on the validation certificate.

11.1.4 End of Life Procedure

For secure sanitization of the cryptographic module, the module must first to be powered off, which will zeroize all keys and CSPs in volatile memory. Then, for actual deprecation, the module shall be upgraded to a newer version that is FIPS 140-3 validated.

The module does not possess persistent storage of SSPs, so further sanitization steps are not required.

11.2 Crypto Officer Guidance

The binaries of the module are contained in the RPM packages for delivery. The Crypto Officer shall follow sections 11.1.1 and 11.1.2 to configure the operational environment and install the module to be operated as a FIPS 140-3 validated module.
Table 17 lists the RPM packages that contain the FIPS validated module and the OE directory where the components are installed. The "Show version" service returns the value “Libgcrypt version 1.9.4-150400.6.8.1”, which matches the service output and the version information provided in the RPM packages where the module is distributed, and map to version 3.2 of the cryptographic module.

Processor Architecture	RPM Packages
Intel 64-bit	libgcrypt20-1.9.4-150400.6.8.1.x86_64.rpm libgcrypt20-hmac-1.9.4-150400.6.8.1.x86_64.rpm
AMD 64-bit	libgcrypt20-1.9.4-150400.6.8.1.x86_64.rpm libgcrypt20-hmac-1.9.4-150400.6.8.1.x86_64.rpm
IBM z15	libgcrypt20-1.9.4-150400.6.8.1.s390x.rpm libgcrypt20-hmac-1.9.4-150400.6.8.1.s390x.rpm

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Processor Architecture	RPM Packages
ARMv8 64-bit	libgcrypt20-1.9.4-150400.6.8.1.aarch64.rpm libgcrypt20-hmac-1.9.4-150400.6.8.1.aarch64.rpm
IBM Power10 64-bit	libgcrypt20-1.9.4-150400.6.8.1.ppc64le.rpm libgcrypt20-hmac-1.9.4-150400.6.8.1.ppc64le.rpm

Table 17 – RPM packages

11.2.1 Memory Management

The user shall only use the memory management functions provided by the libgcrypt API. Critical security parameters (e.g., keys) which are used as input or output parameters shall be managed using the gcry_malloc_secure(), gcry_calloc_secure() and gcry_free() functions.
The function gcry_set_allocation_handler() shall not be used; the user shall not change the libgcrypt memory handlers. The use of this API function implies that the cryptographic module is being executed in an invalid configuration.

11.2.2 AES XTS

The AES algorithm in XTS mode can be only used for the cryptographic protection of data on storage devices, as specified in [SP800-38E]. The length of a single data unit encrypted with the XTS-AES shall not exceed 2^{20} AES blocks, that is 16MB of data.
To meet the requirement stated in IG C.I, the module implements a check that ensures, before performing any cryptographic operation, that the two AES keys used in AES XTS mode are not identical.

11.2.3 Key derivation using SP800-132 PBKDF2

The module provides password-based key derivation (PBKDF2), compliant with SP800-132 and IG D.N. The module supports option 1a from section 5.4 of [SP800-132], in which the Master Key (MK) or a segment of it is used directly as the Data Protection Key (DPK).
In accordance with [SP800-132], the following requirements shall be met.

- Derived keys shall only be used in storage applications. The Master Key (MK) shall not be used for other purposes. The length of the MK or DPK shall be of 112 bits or more (this is verified by the module to determine the service is approved).
- A portion of the salt, with a length of at least 128 bits (this is verified by the module to determine the service is approved), shall be generated randomly using the [SP800-90Arev1] DRBG.
- The iteration count shall be selected as large as possible, as long as the time required to generate the key using the entered password is acceptable for the users. The minimum value shall be 1000 (this is verified by the module to determine the service is approved).
- Passwords or passphrases, used as an input for the PBKDF2, shall not be used as cryptographic keys.
- The length of the password or passphrase shall be of at least 20 characters (this is verified by the module to determine the service is approved). The probability of guessing the password, assuming the worst case scenario of all digits, is estimated to be at most 10^{-20}. Combined with the minimum iteration count as described above, this provides an acceptable trade-off between user experience and security against brute-force attacks.

The calling application shall also observe the rest of the requirements and recommendations specified in [SP800-132].

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

11.2.4 RSA Signatures

In compliance with IG C.F, the module implements only the approved modulus sizes of 2048, 3072, and 4096 bits for signature generation and verification. Each algorithm was tested, and corresponding certificates can be found detailed in Section 2.6 Approved Algorithms.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

12 Mitigation of other attacks

The module implements blinding Against RSA Timing Attacks.
RSA is vulnerable to timing attacks. In a setup where attackers can measure the time of RSA decryption or signature operations, blinding must be used to protect the RSA operation from that attack.
By default, the module uses the following blinding technique: instead of using the RSA decryption directly, a blinded value $y = x\ r^e\ mod\ n$ is decrypted and the unblinded value $x' = y'\ r^{-1}\ mod\ n$ returned.
The blinding value r is a random value with the size of the modulus n.

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard New Instructions
API	Application Programming Interface
CAST	Cryptographic Algorithm Self-Test
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CKG	Cryptographic Key Generation
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CPACF	Central Processor Assist for Cryptographic Function
CSP	Critical Security Parameter
CTR	Counter Mode
DF	Derivation Function
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
ISA	Instruction Set Architecture
KAT	Known Answer Test
KW	AES Key Wrap
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OAEP	Optimal Asymmetric Encryption Padding
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PAI	Processor Algorithm Implementation
PBKDF2	Password-based Key Derivation Function v2
PCT	Pair-wise Consistency Test
PKCS	Public-Key Cryptography Standards
PR	Prediction Resistance

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

PSS	Probabilistic Signature Scheme
RNG	Random Number Generator
RSA	Rivest, Shamir, Addleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SSP	Sensitive Security Parameter
XOF	Extendable Output Function
XTS	XEX-based Tweaked-codebook mode with cipher text Stealing

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

Appendix B. References

FIPS140-3 **FIPS PUB 140-3 - Security Requirements For Cryptographic Modules**
March 2019
https://doi.org/10.6028/NIST.FIPS.140-3

FIPS140-3_IG **Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program**
March 2024
https://csrc.nist.gov/csrc/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/FIPS%20140-3%20IG.pdf

FIPS140-3_MM **FIPS 140-3 Cryptographic Module Validation Program - Management Manual (Draft)**
December 2022
https://csrc.nist.gov/csrc/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/Draft%20FIPS-140-3-CMVP%20Management%20Manual%20v1.2%20%5BDec%2023%202022%5D.pdf

FIPS180-4 **Secure Hash Standard (SHS)**
August 2015
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

FIPS186-4 **Digital Signature Standard (DSS)**
July 2013
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

FIPS197 **Advanced Encryption Standard**
November 2001
https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

FIPS198-1 **The Keyed Hash Message Authentication Code (HMAC)**
July 2008
https://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

FIPS202 **SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions**
August 2015
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

PKCS#1 **Public Key Cryptography Standards (PKCS) #1: RSA Cryptography**
Specifications Version 2.1
February 2003
https://www.ietf.org/rfc/rfc3447.txt

SP800-38A **NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques**
December 2001
https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.

SP800-38B	**NIST Special Publication 800-38B - Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication** May 2005 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38b.pdf
SP800-38C	**NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality** May 2004 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf
SP800-38E	**NIST Special Publication 800-38E - Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices** January 2010 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38e.pdf
SP800-38F	**NIST Special Publication 800-38F - Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping** December 2012 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf
SP800-90Arev1	**NIST Special Publication 800-90A Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators** June 2015 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf
SP800-90B	**NIST Special Publication 800-90B - Recommendation for the Entropy Sources Used for Random Bit Generation** January 2018 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90B.pdf
SP800-132	**NIST Special Publication 800-132 - Recommendation for Password-Based Key Derivation - Part 1: Storage Applications** December 2010 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-132.pdf
SP800-133rev2	**NIST Special Publication 800-133 - Recommendation for Cryptographic** **Key Generation** June 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-133r2.pdf
SP800-140B	**NIST Special Publication 800-140B - CMVP Security Policy Requirements** March 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-140B.pdf

© 2024 SUSE, LLC / atsec information security corporation.
This document can be reproduced and distributed only whole and intact, including this copyright notice.