FIPS 140-3 Non-Proprietary Security Policy

Motorola Solutions Cryptographic Software Module

Software Version: R01.15.00

Running on Intel Core i7-8700 or Intel i5-12600

Document Version: 1.0

Date: October 16, 2024

Prepared by:

intertek acumen security

www.acumensecurity.net

Introduction

Federal Information Processing Standards Publication 140-3 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic modules to be deployed in a Sensitive but Unclassified environment. The National Institute of Standards and Technology (NIST) and Canadian Centre for Cyber Security (CCCS) Cryptographic Module Validation Program (CMVP) run the FIPS 140-3 program. The NVLAP accredits independent testing labs to perform FIPS 140-3 testing; the CMVP validates modules meeting FIPS 140-3 validation. Validated is the term given to a module that is documented and tested against the FIPS 140-3 criteria.

More information is available on the CMVP website at:
https://csrc.nist.gov/projects/cryptographic-module-validation-program

About this Document

This non-proprietary Cryptographic Module Security Policy for the Motorola Solutions Cryptographic Software Module provides an overview of the product and a high-level description of how it meets the overall Level 1 security requirements of FIPS 140-3.

The Motorola Solutions Cryptographic Software Module may also be referred to as the “module” in this document.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Motorola Solutions, Inc. shall have no liability for any error or damages of any kind resulting from the use of this document.

Notices

This document may be freely reproduced and distributed in its entirety without modification.

Table of Contents

List of Tables

List of Figures

1. General

This document defines the cryptographic module security policy for the Motorola Solutions Cryptographic Software Module (Software version: R01.15.00 running on Intel Core i7-8700 or Intel i5-12600), also referred to as the "module" hereafter. It contains specification of the security rules, under which the cryptographic module operates, including the security rules derived from the requirements of the FIPS 140-3 standard.

The module is a software-hybrid based cryptographic module that runs on a general-purpose computing platform that supports processor algorithm acceleration. The module is classified as a multi-chip standalone module embodiment. The module provides approved cryptographic functionalities via an Application Programming Interface (API) to the application layer.

The following table lists the level of validation for each area in FIPS 140-3:

ISO/IEC 24759 Section 6.	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	1
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	N/A

Table 1 – Security Levels

2. Cryptographic Module Specification

The module is intended for use by the markets that require FIPS 140-3 validated overall Security Level 1.

#	Operating System	Hardware Platform	Processor	PAA/Acceleration
1	Windows 10 Enterprise	Motorola Command Central HUB	Intel Core i7-8700 (6 core)	Yes
2	Windows 10 Enterprise	HP Z2 Mini G9 Workstation	Intel i5-12600	Yes

Table 2 – Tested Operational Environments

The module has also been confirmed by Motorola Solutions to be operational on the following OEs shown in Table *3*. However, no target testing was performed on this platform for FIPS 140-3 validation with the specific software versions listed in this document. Note: The CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys on the operational environments for which operational testing was not performed.

#	Operating System	Hardware Platform
1	*Microsoft Windows 7 and 10 Professional*	HP ZBook 15 G3 Mobile Workstation, Intel Core i7 with AES-NI
2	*Red Hat OpenShift 3 on Red Hat UBI 7*	HP DL20 Gen10 server, Intel(R) Xeon(R) E-2236 CPU with AES-NI
3	*Red Hat OpenShift 4 on Red Hat UBI 8*	HP DL160 Gen 10 Server, Intel(R) Xeon(R)-S 4215R CPU with AES-NI
4	*Red Hat OpenShift 4 on Red Hat UBI 8*	HPE ProLiant DL20 Gen10 server, Intel(R) Xeon(R) E-2236 CPU with AES-NI
5	*Microsoft Windows 10 IoT Enterprise LTSB 2016 64bit*	HP ZBook 15u G4 Mobile Workstation, Intel® Core i7 CPU with AES-NI
6	*Microsoft Windows 10 IoT Enterprise LTSC 2019 64bit*	HP ZBook 15u G5 Mobile Workstation, Intel® Xeon® E-2186M CPU with AES-NI
7	*Microsoft Windows 10 IoT Enterprise LTSC 2019 64bit*	HP ZBook 15u G6 Mobile Workstation, Intel® Xeon® E-2286M CPU with AES-NI
8	*Microsoft Windows 10 IoT Enterprise LTSC 2019 64bit*	HP ZBook Fury 15 G7 Mobile Workstation, Intel® Xeon® W-10885M CPU with AES-NI
9	*Microsoft Windows 10 IoT Enterprise LTSB 2016 64bit*	HP Z440 Workstation, Intel Xeon E5-1603v3 CPU with AES-NI

#	Operating System	Hardware Platform
10	*Microsoft Windows 10 IoT Enterprise LTSC 2019 64bit*	HP Z440 Workstation, Intel Xeon E5-1603v3 CPU with AES-NI
11	*Microsoft Windows 10 IoT Enterprise LTSB 2016 64bit*	HP Z2 Mini G3 Workstation, Intel Xeon E3-1225v5 CPU with AES-NI
12	*Microsoft Windows 10 IoT Enterprise LTSC 2019 64bit*	HP Z2 Mini G4 Workstation, Intel Xeon E-2144G CPU with AES-NI
13	*Microsoft Windows 10 IoT Enterprise LTSC 2019 64bit*	HP Z2 Mini G5 Workstation, Intel Xeon W-1250 CPU with AES-NI
14	*Microsoft Windows 10 IoT Enterprise LTSC 2021*	HP ZBook Fury G10, Intel Core i9 CPU with AES-NI

Table 3 – Vendor Affirmed Operational Environments

2.1 Modes of Operation

The module operates in two different modes of operation.

- **Approved mode**: All services listed in *Table 9* are available when the module is operating in Approved mode.
- **Non-Approved mode**: All services listed in *Table 9* and *Table 10* are available when the module is operating in non-Approved mode.

The module defaults to Approved mode at initial power-up and will transition between Approved Mode and non-Approved mode by using the "Configure Approved Mode" service. The operator can configure the mode of the module by using the "Configure Approved Mode" Service listed in *Table 9*. The "Configure Approved Mode" services sets an Approved mode flag via the API. The Approved mode flag is "fips_mode = 1" in the Approved mode and the non-Approved mode flag is "fips_mode = 0". The operator shall zeroize all CSPs by power cycling the module when transitioning between Approved and non-Approved modes. The operator must retain control of the module while zeroization is in process.

2.2 Cryptographic Functionality

The module's supported cryptographic functions are listed in the following tables.

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A3497	AES [FIPS 197]	ECB [SP 800-38A]	Key Size: 256	Encrypt, Decrypt
		CBC [SP 800-38A]	Key Size: 256	Encrypt, Decrypt
		OFB [SP 800-38A]	Key Size: 256	Encrypt, Decrypt
		GCM [SP 800-38D][1]	Key Size: 256	Encrypt, Decrypt
		KW [SP 800-38F]	Key Size: 256	Encrypt, Decrypt

[1] Per IG C.H option 2, the module generates 96-bit GCM IVs randomly as specified in SP800-38D section 8.2.2 using an approved DRBG (Cert. #A3497), that is internal to the module's boundary.

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
	DRBG [SP800-90Ar1]	CTR	AES-256	Deterministic Random Bit Generation
	ECDSA [FIPS 186-4]		P-384	Key Generation/Signature Generation/Signature Verification
	HMAC [FIPS 198-1]	HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512	128-1024 bits 192-1024 bits 256-1024 bits	Message authentication, Code Integrity tests
	KAS-ECC [SP 800-56Ar3]	ECC (Initiator, Responder), Key pair generation, Partial public key validation, One-step key derivation	P-384 with SHA2-256	Key Establishment provides 192 bits of encryption strength per IG D.F Scenario 2 path (2)
			P-384 with SHA2-384	
			P-384 with SHA2-512	
	KTS [IG D.G]	AES-KW	Key Sizes: 256	Key Wrap provides 256 bits of encryption strength
	KTS [IG D.G]	GCM	Key Sizes: 256	Key Wrap provides 256 bits of encryption strength
	PBKDF [SP 800-132]	Option 1a Option 2a (using HMAC)	sLen = 16 – 512 bytes C = 1 – 50000 SHA2-256 SHA2-384 SHA2-512	Password-Based Key Derivation
	SHS [FIPS 180-4]	SHA2-256 SHA2-384 SHA2-512	N/A	Message Digest Generation
Vendor Affirmed	CKG	CTR_DRBG	N/A	Asymmetric key seed and symmetric key generation in accordance with SP 800-133rev2 sections

CAVP Cert	Algorithm and Standard	Mode/Method	Description / Key Size(s) / Key Strength(s)	Use / Function
				4, 5.2, 6.1 and 6.2 and IG D.H with B=U[2]

Table 4 – Approved Algorithms

Algorithm	Caveat	Use / Function
AES MAC[3]		[IG D.C] AES MAC for Project 25 APCO OTAR (Cert. #A3497)

Table 5 – Non-Approved Algorithms Allowed in the Approved Mode of Operation

Algorithm/Function	Use/Function
ADP	ADP Encryption/Decryption – Motorola Solutions proprietary algorithm
DES	DES Encryption/Decryption – ECB, OFB and CBC Mode

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

The module does not support Non-Approved Allowed Algorithms with No Security Claimed.

2.3 Module Description and Cryptographic Boundary

The module is classified by FIPS 140-3 as a software-hybrid module and multi-chip standalone module embodiment running on a modifiable operating environment. The physical perimeter is the hardware platform (a.k.a. general-purpose computer) on which the module is installed. The TOEPP is listed in the Hardware Platform column in Table 2. The logical perimeter which is also the cryptographic boundary of the module is a shared object file, libALG.dll, that is linked into the application running on a general-purpose computing platform containing an Intel processor with algorithm acceleration.

[2] PBKDF salt, GCM IV, and ECDH private key

[3] AES-CBC-MAC is allowed for use within OTAR until November 1, 2023.

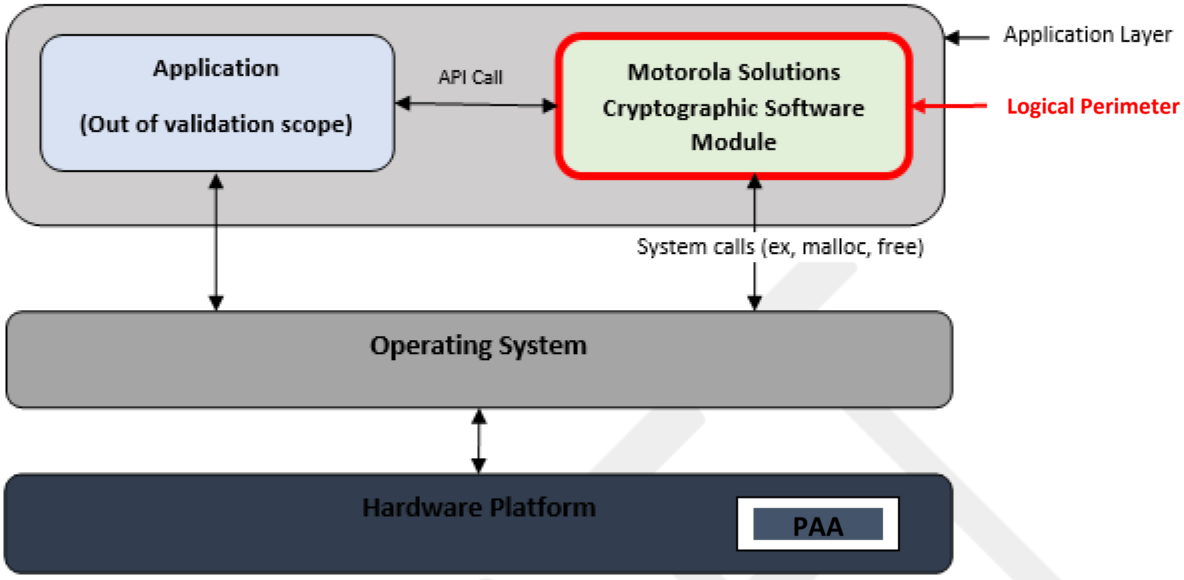
Application
(Out of validation scope)
API Call
Motorola Solutions Cryptographic Software Module
Application Layer
Logical Perimeter
System calls (ex, malloc, free)
Operating System
Hardware Platform
PAA

Figure 1 – Logical cryptographic boundary and physical boundary

2.4 Security Rules and Guidance

The module enforces the following security rules:

1. The module does not support any operator authentication.
2. The module is available to perform services only after successfully completing the pre-operational self-tests.
3. Data output is inhibited during pre-operational self-tests, zeroization, and while in an error state.
4. The module shall not support concurrent operators.
5. The module enters the uninitialized state if any pre-operational self-test fail. The Uninitialized state can be exited by restarting the module allowing the module to attempt to re-initialize itself.
6. The module can perform periodic self-tests. An operator can perform periodic self-tests on demand by using the Operator Initiated Self-Test API to run conditional self-tests or module restart to run pre-operational tests.
7. The module does not perform any cryptographic functions while in the uninitialized state.
8. The module returns the results of the pre-operational and integrity self-tests to the operator.
9. The module may be power cycled to zeroize all CSPs.
10. The operator may choose whether the module will run in the Approved mode or non-approved mode using the “Configure Approved Mode” Service.
11. The only operating environment restrictions is that processor algorithm acceleration must be enabled.

3. Cryptographic Module Interfaces

The Module’s logical interfaces are described in Table 7; the Module’s physical ports are outside the module boundary.

Physical Port	Logical interface	Data that passes over port/interface
N/A: Internal (call stack)	Control input	API entry point and corresponding stack parameters
	Data input	API entry point data input stack parameters
	Status output	API entry point return values and status stack parameters
	Data output	API entry point data output stack parameters

Table 7 – Ports and Interfaces

The module does not support a control output interface.

4. Roles, Services, and Authentication

The module supports the Cryptographic Officer (CO) role and does not support authentication. An operator is considered the owner of the thread that instantiates the module and, therefore, only one operator is allowed.

Role	Service	Input	Output
CO	Self-Test	Power-up/Run Self-Test command	Status: Success/Error
CO	Load Entropy	Entropy Input String	N/A
CO	Get Module Status	Get module status command	Module initialization status, Approved mode status, AES acceleration status
CO	Get Module Version	Get module version command	"libALG Library R01.15.00 – Copyright 2022 Motorola Solutions, Inc."
CO	Configure Approved Mode	Approved mode enabled/Approved mode disabled	Enable/Disable
CO	Utility	Module query for algorithm/key status	Algorithm/key status information
CO	Encrypt	Encryption key, plaintext	Ciphertext or error status
CO	Decrypt	Decryption key, ciphertext	Plaintext of error status
CO	AES Key Wrapping	Encryption key, input data	Wrapped key
CO	AES Key Unwrapping	Decryption key, input data	Unwrapped data
CO	Generate OTAR MAC	Input data	MAC Key
CO	DRBG	Entropy input data	Pseudo-random number
CO	Hashing	Hash algorithm, input data	Hashed output
CO	HMAC-SHA	Hash Key, input data	digest
CO	Zeroize	N/A	N/A

Role	Service	Input	Output
CO	PBKDF	Password, iteration count, salt, hash algorithm	Derived key
CO	ECDSA Key Generation	Private key or SP 800-90Ar1 Seed	Private key/Public key
CO	ECDSA Sig Generation	Private key, digest	signature
CO	ECDSA Sig Verification	Signature, digest, Public Key	Status: Success/Error
CO	KAS-ECC	Private key, Public Key of Remote Party (Host B)	ECDH Shared Secret/KDF Derived Key

Table 8 – Roles, Service Commands, Input and Output

The SSPs modes of access shown in Table 9, are defined as:

- **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** = Zeroize: The Module zeroizes the SSP

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Self-Test	Perform all pre-operational CASTs prior to module initialization via module restart/Perform all the conditional CASTs prior to first use of that service operation or on demand via Self-test API call	N/A	N/A	CO	N/A	”fips_mode = 1”
Load Entropy	Load external entropy to seed the DRBG	N/A	Entropy Input string	CO	W,E,Z	”fips_mode = 1”
Get Module Status	Show the module status	N/A	N/A	CO	N/A	”fips_mode = 1”
Get Module Version	Get module version number	N/A	N/A	CO	N/A	”fips_mode = 1”

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Configure Approved Mode	Set/Unset module to Approved mode	N/A	N/A	CO	N/A	”fips_mode = 1”
Utility	Key check and other services	N/A	N/A	CO	N/A	”fips_mode = 1”
Encrypt	Encryption of voice and data	AES (OFB, CBC, ECB, GCM) Cert #A3497	AES-256 Key	CO	W,E,Z	”fips_mode = 1”
Decrypt	Decryption of voice and data	AES (OFB, CBC, ECB, GCM) Cert #A3497	AES-256 Key	CO	W,E,Z	”fips_mode = 1”
AES Key Wrapping	Used for the encryption of keys	KTS (AES-KW or AES-GCM) Cert #A3497	AES-256 Key Wrap Key	CO	W,E,Z	”fips_mode = 1”
AES Key Unwrapping	Used for the decryption of keys	KTS (AES-KW or AES-GCM) Cert #A3497	AES-256 Key Wrap Key	CO	W,E,Z	”fips_mode = 1”
Generate OTAR MAC	Used to generate MAC (Message Authentication Code) as defined in [OTAR]	AES MAC (CBC) Cert #A3497	OTAR MAC Key	CO	W,E,Z	”fips_mode = 1”
DRBG	Used for random number, IV and key generation using DRBG [SP 800-90Ar1]	DRBG (output directly used for CKG) CKG Cert #A3497	Entropy Input string, SP 800-90Ar1 Seed, SP 800-90Ar1 Internal State (“V” and “Key”)	CO	G,W,R	”fips_mode = 1”
Hashing	Used to generate SHA2-	SHS	N/A	CO	N/A	”fips_mode = 1”

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
	256/384/512 message digest	Cert #A3497				
HMAC-SHA	Used to calculate data integrity codes with HMAC	HMAC Cert #A3497	Keyed Hash Key	CO	W,E	”fips_mode = 1”
Zeroize[4]	Zeroize all SSPs	N/A	All	CO	Z	”fips_mode = 1”
PBKDF[5]	Used to generate keys using PBKDF [SP 800-132]	PBKDF Cert #A3497	PBKDF Secret Value DPK	CO	W,E	”fips_mode = 1”
ECDSA Key Generation	Used for generating asymmetric key pair	ECDSA, CKG Cert #A3497	ECDSA Private Key, ECDSA Public Key	CO	R,G	”fips_mode = 1”
ECDSA Signature Generation	Used to generate a digital signature	ECDSA Cert #A3497	ECDSA Private Key	CO	R,E	”fips_mode = 1”
ECDSA Signature Verification	Used to verify a digital signature	ECDSA Cert #A3497	ECDSA Public Key	CO	R,E	”fips_mode = 1”
KAS-ECC	Used for key agreement process using ECDH	KAS-ECC, CKG Cert #A3497	ECDH Shared Secret, KDF Derived Key, ECDH Private Key, ECDH Public Key, ECDH Remote Party Public Key	CO	W,G,E,R	”fips_mode = 1”

Table 9 – Approved Services

[4] The Zeroize service zeroizes the key in the volatile memory by power cycling the module.

[5] As per NIST SP 800-132 and IG D.N, keys generated by the module shall be used as recommend in Section 5.4 of [132]. Any other use of the approved PBKDF is non-conformant. In approved mode the operator shall enter a password no less than 8 hexadecimal digits in length. The probability of guessing the password will be equal to 1:16^8. Due to the computational limitations of this embedded operational environment, the iteration count associated with the PBKDF should not exceed 50,000. The minimum iteration count is 1, however it shall be selected as large as possible. Keys derived from passwords may only be used in storage applications.

Service	Description	Algorithms Accessed	Role	Indicator
Encrypt	Encryption of voice and data	ADP	CO	”fips_mode = 0”
Decrypt	Decryption of voice and data	ADP	CO	”fips_mode = 0”
Encrypt	Encryption of voice and data	DES	CO	”fips_mode = 0”
Decrypt	Decryption of voice and data	DES	CO	”fips_mode = 0”

Table 10 – Non-Approved Services

5. Software/Firmware Security

The module is software-hybrid and operates on a general-purpose computing platform that is built with production grade materials. The software components are protected and authenticated using an HMAC hash function using the keyed hash key referenced in *Table 11*. The operator can initiate the software integrity test on demand by restarting the module. If integrity test fails, the module will not initialize and no security functions will be provided by the module.

6. Operational Environment

The module operates and was tested on the following modifiable operational environment:

- General purpose computing platform as specified in Table 2.

As per ISO/IEC 19790:2012 7.6.3:

- The cryptographic module has control over its own SSPs.
- The operational environment provides the capability to separate individual application processes from each other to prevent uncontrolled access to CSPs and uncontrolled modifications of SSPs, regardless if this data is in the process memory or stored on persistent storage within the operational environment. This ensures that direct access to CSPs and SSPs is restricted to the cryptographic module and the trusted parts of the operational environment.

7. Physical Security

The module is a multi-chip standalone, software hybrid embodiment module with an Intel CPU that supports Processor Algorithm Acceleration installed within a GPC. The module utilizes a production grade hardware component with standard passivation applied to it.

8. Non-invasive Security

Not Applicable. The module does not implement non-invasive security measures.

9. Sensitive Security Parameter Management

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroisation	Use & related keys
Entropy Input string	Variable (384-bit minimum)	N/A	External	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power cycle/Reset	Used to derived SP 800-90Ar1 seed
SP 800-90Ar1 Seed	384-bit	DRBG (#A3497)	Internal	N/A	N/A	Volatile memory (plaintext)	Power Cycle/Reset	Derived from the Entropy Input string. Used in AES IV, ECDSA Private Key, ECDSA Public Key, ECDH Private Key generation, ECDH Public Key generation
SP 800-90Ar1 Internal State (“V” and “Key”)	N/A	DRBG (#A3497)	Internal	N/A	N/A	Volatile memory (plaintext)	Power Cycle/Reset	CTR_DRBG state
Keyed Hash Key	Variable (192-bit minimum)	HMAC, SHS(#A3497)	External	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset	Used in HMAC function
AES-256 Key	256-bit	AES ECB, CBC, OFB, GCM (#A3497)	External	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset /End of data processing	Used in data encryption / decryption
AES-256 Key	256-bit	AES KW, AES GCM (#A3497)	External	Import (electronic)	Input via API in plaintext	Volatile memory	Power Cycle/Reset	Used in key encryption /

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroisation	Use & related keys
Wrap Key						(plaintext)	/End of data processing	decryption
PBKDF Secret Value	Variable (64-bit minimum)	PBKDF (#A3497) CKG	External[6]	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset	Used in Key Derivation
DPK	128-bit minimum	PBKDF (#A3497)	Internal	Export (electronic)	Internally computed	Volatile memory (plaintext)	Power Cycle/Reset	Derived by the PBKDF using the PBKDF Secret Value
OTAR MAC Key	256-bit	AES MAC	External	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset	Used for AES MAC
ECDH Private Key	192-bit	KAS-ECC (#A3497) CKG	External or Internal	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset	Used to generate ECDH Public Key
ECDH Shared Secret	192-bit	KAS-ECC (#A3497) CKG	Internal	Export (electronic)	Internally computed	Volatile memory (plaintext)	Power Cycle/Reset	Used to generate KDF derived key
ECDSA Private Key	192-bit	ECDSA (#A3497) CKG	External or Internal	Import or Export (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset	Used to create digital signature
KDF Derived Key	Variable (128-bit minimum)	KAS-ECC (#A3497) CKG	Internal	Export (electronic)	Internally computed	Volatile memory (plaintext)	Power Cycle/Reset	Used in KAS-ECC
ECDH Public Key	192-bit	KAS-ECC (#A3497) CKG	Internal	Export (electronic)	Internally computed	Volatile memory (plaintext)	Power Cycle/Reset	Used in key exchange

[6] Password generated externally. Salt may be generated externally or internally according to SP 800-133.

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import/Export	Establishment	Storage	Zeroisation	Use & related keys
ECDH Remote Party Public Key	192-bit	KAS-ECC (#A3497)	External	Import (electronic)	Input via API in plaintext	Volatile memory (plaintext)	Power Cycle/Reset	Used in key exchange
ECDSA Public Key	192-bit	ECDSA (#A3497) CKG	Internal	Export(electronic)	Internally computed	Volatile memory (plaintext)	Power Cycle/Reset	Used for ECDSA

Table 11 – SSPs

Entropy sources	Minimum number of bits of entropy	Details
Entropy Input String	384 (minimum seed length for AES-256 CTR_DRBG)	The entropy for seeding the SP 800-90Ar1 DRBG is determined by the user operator of the module which is outside of the module's cryptographic boundary. To be compliant, the target application shall supply at least 384 bits of entropy in order to meet the security strength required for the random number generation mechanism as shown in [SP 800-90Ar1] Table 3 (CTR_DRBG) and set required bits into the module by calling module defined API function. Since entropy is loaded passively into the module, there is no assurance of the minimum strength of generated SSPs (e.g. keys).

Table 12 – Non-Deterministic Random Number Generator Specification

10. Self-Tests

10.1 Automatic Self-Test

The module automatically performs pre-operational self-tests and conditional cryptographic algorithm self-tests. Automatic pre-operational self-tests are initiated upon module power-up and must pass in order for the module to initialize and render any security services. A failure of any pre-operational self-test will prevent the module from initializing. Automatic conditional cryptographic algorithm self-tests (CAST) will run prior to the first use of a security service using an approved cryptographic algorithm after module initialization. Failure of a conditional CAST will cause the module to enter a critical error state whereby no cryptographic services will be rendered by the module.

A) Pre-Operational Self-Tests

- Software integrity test: HMAC-SHA2-384 (HMAC-SHA2-384 CAST done prior to integrity test)

B) Conditional Self-Tests

1) Conditional cryptographic algorithm test

- SHA2-256 CAST (run before Software Integrity test)
- SHA2-512 CAST (run before Software Integrity test)
- HMAC-SHA2-384 CAST (run before Software Integrity test)
- AES ECB Encrypt CAST (256-bit key)
- AES ECB Decrypt CAST (256-bit key)
- AES CBC Encrypt CAST (256-bit key)
- AES CBC Decrypt CAST (256-bit key)
- AES OFB Encrypt CAST (256-bit key)
- AES OFB Decrypt CAST (256-bit key)
- AES GCM Encrypt CAST (256-bit key)
- AES GCM Decrypt CAST (256-bit key)
- CTR_DRBG [SP 800-90Ar1] CAST (Instantiate, Generate, and Reseed)
- AES-KW [SP 800-38F] Wrap CAST
- AES-KW [SP 800-38F] Unwrap CAST
- KAS ECC [SP 800-56ar3] CAST
- KDF [SP 800-56Arev3] CAST (SHA2-256, SHA2-384, SHA2-512)
- PBKDF [SP 800-132] CAST (128-bit key, 128-bit salt, 2 iterations)
- ECDSA Signature Generation CAST
- ECDSA Signature Validation CAST

2) Conditional pair-wise consistency test

- ECDSA Key Gen PCT (384-bit private key, 384-bit public key)

10.2 User Initiated Self-test

Conditional self-tests can be user initiated by calling the “Self-Test” service via the LIBALG_API_Run_Self_Tests() API. User initiated self-tests via the API can only be invoked after the module has initialized. When initiating self-test via API call, the following tests are performed:

- SHA2-256 CAST
- SHA2-512 CAST

- HMAC-SHA2-384 CAST
- AES ECB Encrypt CAST (256-bit key)
- AES ECB Decrypt CAST (256-bit key)
- AES CBC Encrypt CAST (256-bit key)
- AES CBC Decrypt CAST (256-bit key)
- AES OFB Encrypt CAST (256-bit key)
- AES OFB Decrypt CAST (256-bit key)
- AES GCM Encrypt CAST (256-bit key)
- AES GCM Decrypt CAST (256-bit key)
- CTR_DRBG [SP 800-90Ar1] CAST (Instantiate, Generate, and Reseed)
- AES-KW [SP 800-38F] Wrap CAST
- AES-KW [SP 800-38F] Unwrap CAST
- KAS ECC [SP 800-56ar3] CAST
- KDF [SP 800-56Arev3] CAST (SHA2-256, SHA2-384, SHA2-512)
- PBKDF [SP 800-132] CAST (128-bit key, 128-bit salt, 2 iterations)
- ECDSA Key Gen PCT (384-bit private key, 384-bit public key)
- ECDSA Signature Generation CAST
- ECDSA Signature Validation CAST

Failure of any of the self-test initiated by calling the “Self-Test” service via the API will render the module inoperable.

11. Life-cycle Assurance

The cryptographic module is not installed, but it is dynamically linked to the application at run time. The operator shall configure the module to operate in Approved mode as specified in Section 2.1. After end-of-life for the module, the operator shall zeroize all SSPs used by the module by removing power to the device using the module.

12. Mitigation of Other Attacks

Not Applicable. The Module does not implement mitigations of other attacks outside the scope of FIPS 140-3.

References and Definitions

Abbreviation	Full Specification Name
[FIPS 140-3]	*Security Requirements for Cryptographic Modules*, March 2019
[IG]	*Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program, November 2021.*
[SP 800-132]	*NIST Special Publication 800-132, Recommendation for Password-Based Key Derivation, Part 1: Storage Applications, December 2010*
[FIPS 186-4]	*National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-4, July 2013.*
[FIPS 197]	*National Institute of Standards and Technology, Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, November 2001*
[FIPS 198-1]	*National Institute of Standards and Technology, The Keyed-Hash Message Authentication Code (HMAC), Federal Information Processing Standards Publication 198-1, July 2008*
[FIPS 180-4]	*National Institute of Standards and Technology, Secure Hash Standard, Federal Information Processing Standards Publication 180-4, August 2015*
[SP 800-38A]	*National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation, Methods and Techniques, Special Publication 800-38A, December 2001*
[SP 800-38D]	*National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, Special Publication 800-38D, November 2007*
[SP 800-38F]	*National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping, Special Publication 800-38F, December 2012*
[SP 800-56Ar3]	*NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, April 2018*
[SP 800-56Cr2]	*NIST Special Publication 800-56C Revision 2, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, August 2020*
[SP 800-90Ar1]	*National Institute of Standards and Technology, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Special Publication 800-90A, Revision 1, June 2015.*
[OTAR]	*Project 25 – Digital Radio Over-The-Air-Rekeying (OTAR) Messages and Procedures [TIA-102.AACA-A], September 2014*

Table 13 – References

Acronym	Definition
ADP	Advanced Digital Privacy
AES	Advanced Encryption Standard
APCO	Association of Public-Safety Communications Officials
CBC	Cipher Block Chaining
CKG	Cryptographic Key Generation
CSP	Critical Security Parameter
DRBG	Deterministic Random Bit Generator

Acronym	Definition
ECB	Electronic Code Book
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Diffie-Hellman
FIPS	Federal Information Processing Standards
GCM	Galois/Counter Mode
HMAC	Hash-based Message Authentication Code
IV	Initialization Vector
KAT	Known Answer Test
KDA	Key Derivation Algorithm
MAC	Message Authentication Code
OFB	Output Feedback
OTAR	Over The Air Rekeying
PBKDF	Password-Based Key Derivation Function
PCT	Pairwise Consistency Test
SSP	Sensitive Security Parameter

Table 14 -Definitions