Google LLC

Look-aside Cryptography and Compression Engine (LCE)

FIPS 140-3 Non-Proprietary Security Policy

Public Material – May be reproduced only in its original entirety (without revision).

Table of Contents

1 General 4
1.1 Overview 4
1.2 Security Levels 5
1.3 Additional Information 5
2 Cryptographic Module Specification 6
2.1 Description 6
2.2 Tested and Vendor Affirmed Module Version and Identification 8
2.3 Excluded Components 9
2.4 Modes of Operation 9
2.5 Algorithms 9
2.6 Security Function Implementations 12
2.7 Algorithm Specific Information 14
2.8 RBG and Entropy 15
2.9 Key Generation 15
2.10 Key Establishment 15
2.11 Industry Protocols 15
3 Cryptographic Module Interfaces 15
3.1 Ports and Interfaces 16
4 Roles, Services, and Authentication 16
4.1 Authentication Methods 16
4.2 Roles 16
4.3 Approved Services 16
4.4 Non-Approved Services 22
4.5 External Software/Firmware Loaded 23
4.6 Bypass Actions and Status 23
4.7 Cryptographic Output Actions and Status 23
5 Software/Firmware Security 23
5.1 Integrity Techniques 23
5.2 Initiate on Demand 23
6 Operational Environment 23
6.1 Operational Environment Type and Requirements 23
6.2 Configuration Settings and Restrictions 24
7 Physical Security 24
7.1 Mechanisms and Actions Required 24
8 Non-Invasive Security 24

Public Material – May be reproduced only in its original entirety (without revision).

8.1 Mitigation Techniques....24
9 Sensitive Security Parameters Management....24
9.1 Storage Areas....24
9.2 SSP Input-Output Methods....25
9.3 SSP Zeroization Methods....25
9.4 SSPs....25
9.5 Transitions....27
10 Self-Tests....27
10.1 Pre-Operational Self-Tests....27
10.2 Conditional Self-Tests....28
10.3 Periodic Self-Test Information....29
10.4 Error States....30
10.5 Operator Initiation of Self-Tests....31
11 Life-Cycle Assurance....32
11.1 Installation, Initialization, and Startup Procedures....32
11.2 Administrator Guidance....32
11.3 Non-Administrator Guidance....32
11.4 Maintenance Requirements....32
11.5 End of Life....32
12 Mitigation of Other Attacks....32

Public Material – May be reproduced only in its original entirety (without revision).

List of Tables

Table 1: Security Levels 5
Table 2: Tested Module Identification – Hardware 9
Table 3: Modes List and Description 9
Table 4: Approved Algorithms 11
Table 5: Non-Approved, Allowed Algorithms with No Security Claimed 12
Table 6: Non-Approved, Not Allowed Algorithms 12
Table 7: Security Function Implementations 14
Table 8: Ports and Interfaces 16
Table 9: Roles 16
Table 10: Approved Services 22
Table 11: Non-Approved Services 23
Table 12: Mechanisms and Actions Required 24
Table 13: Storage Areas 24
Table 14: SSP Input-Output Methods 25
Table 15: SSP Zeroization Methods 25
Table 16: SSP Table 1 26
Table 17: SSP Table 2 27
Table 18: Pre-Operational Self-Tests 28
Table 19: Conditional Self-Tests 29
Table 20: Pre-Operational Periodic Information 29
Table 21: Conditional Periodic Information 30
Table 22: Error States 31

List of Figures

Figure 1: Block Diagram 8
Figure 2: Single-chip picture 8

1 General

1.1 Overview

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 program. NVLAP accredits independent testing labs to perform FIPS 140-3 testing; the CMVP validates modules meeting FIPS 140-3 validation requirements. 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:

Public Material – May be reproduced only in its original entirety (without revision).

http://csrc.nist.gov/groups/STM/cmvp/index.html

About this Document

This non-proprietary Cryptographic Module Security Policy for the Google LLC Look-aside Cryptography and Compression Engine (LCE) (Hardware Version: 3.1) provides an overview of the product and a high-level description of how it meets the overall Security Level 1 requirements per FIPS 140-3.

The Look-aside Cryptography and Compression Engine (LCE) may also be referred to as "LCE" or 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. Google LLC 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.

1.2 Security Levels

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
	Overall Level	1

Table 1: Security Levels

1.3 Additional Information

The module claims an overall Security Level of 1 with all individual sections at a Security Level 1. The module does not implement any non-invasive security mitigations or mitigations of other attacks and thus the requirements per these sections are inapplicable.

Public Material – May be reproduced only in its original entirety (without revision).

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

LCE provides look-aside cryptography and compression services. Cryptography services provided by LCE include bulk encryption, cryptographic hashing, Integrity Checksum Value (ICV) verification, and compression services. LCE supports operation chaining including cryptographic and pipeline compressing/decompressing and verification as well as standalone or combined DMA data (payload) transfers between nodes.

The module provides lookaside hardware acceleration for bulk encryption and compression operations, including AES (XTS, GCM, CTR, GMAC), HMAC, SHA-1, SHA-2, SHA-3, RSA and ECDSA cryptographic algorithms as well as zStandard and Snappy compression algorithms and CRC for ICV checksum calculation.

Within the LCE IP block are four Command Push/Pull Clusters (CPP) which implement an internal interface to a Shared Slice & Memory Module (SSM) component. Each “slice” represents a compression/ cryptography engine. Additionally, the SSM implements the shared RAM for the slices. Additionally, the module contains a DMA engine (acting as a “master”), and a LCE Interface block providing an AXI4 endpoint, accelerator RAM (aRAM), and a Work Queue Manager (WQM). The WQM within the LCE Interface block provides Queue Pairs implementing a Request and Response Queue. Request descriptors are placed into Request queues by module functionality consumes (ARM Compute Complex (ACC) internal to the SoC or external Host) and Response descriptors are placed in the Response queues. The DMA engine uses data in the request descriptors to copy data from the on-chip or external DRAM to the internal Shared RAM to be processed and ultimately returned via response descriptor.

The module also includes firmware which is loaded into aRAM and subsequently loaded onto the LCE engines for execution. The firmware implements the following functions:

- queue management,
- state management,
- general-purpose functions including some cryptographic algorithm implementations, compression and DMA services,
- known-answer test (CAST) logic for cryptographic algorithms,
- error handling, and
- diagnostic functions.

The module is connected to other subsystems via an AMBA on-chip system bus. Through the AXI4 interfaces the module provides access to Control/Status Registers (CSR), a Push Interface, Request Interface, and Compression Interface. The CSR is used for device initialization, heartbeat information, and firmware loading via the Network Management Fabric (NMF). The Push Interface is used to doorbell the device when a job is placed in a ring buffer (via HIF), and provides pointers to a request descriptor. The Request Interface is used by LCE to read data (request descriptors, key/IVs, configuration data, and payload data) and write data (response descriptors, processed payloads) to external RAM (either on the Host or attached to the ACC via HIF). The Compression Interface is an interface used to monitor state information and reads/writes data to be processed (compressed) via the Coherency Mesh Network (CMN).

Public Material – May be reproduced only in its original entirety (without revision).

DMA reads/writes to external RAM occur via the HIF. As necessary to satisfy IG 2.3.B, intervening circuitry must not pose any risk to modification or interception of CSPs or other types of attacks. The HIF is a standard PCI Express (PCIe) device. Queues within LCE are associated with a specific bus device function. Each queue has a specific function ID. By design, the requests must use a defined PCIe channel. This separation is maintained by the Address Translation Engine within the host CPU's Input-Output Memory Management Unit (IOMMU), or the System Memory Management Unit within the ACC. If a request is made for the wrong function ID, it will be treated as an unsupported request and will fail.

The module is validated for use with the Integrated Compute Complex (ICC) module which employs firmware running inside the ICC. The ICC Secure Firmware is required to perform the firmware integrity test/firmware load test on the LCE firmware image prior to initialization.

Module Type: Hardware

Module Embodiment: Single Chip

Module Characteristics: SubChip

Cryptographic Boundary:

The physical perimeter of the module is the single chip i.e., the IN763 SoC. The module's cryptographic boundary is the firmware and hardware functionality contained within the LCE IP Block at the sub-chip level. i.e. a sub-chip cryptographic subsystem within the IN763 SoC/single-chip. The module supports an Approved mode and a non-Approved mode of operation.

Tested Operational Environment's Physical Perimeter (TOEPP):

The cryptographic boundary of the module and the relationship among the various internal components of the module have been depicted in Figure 1 below.

Public Material – May be reproduced only in its original entirety (without revision).

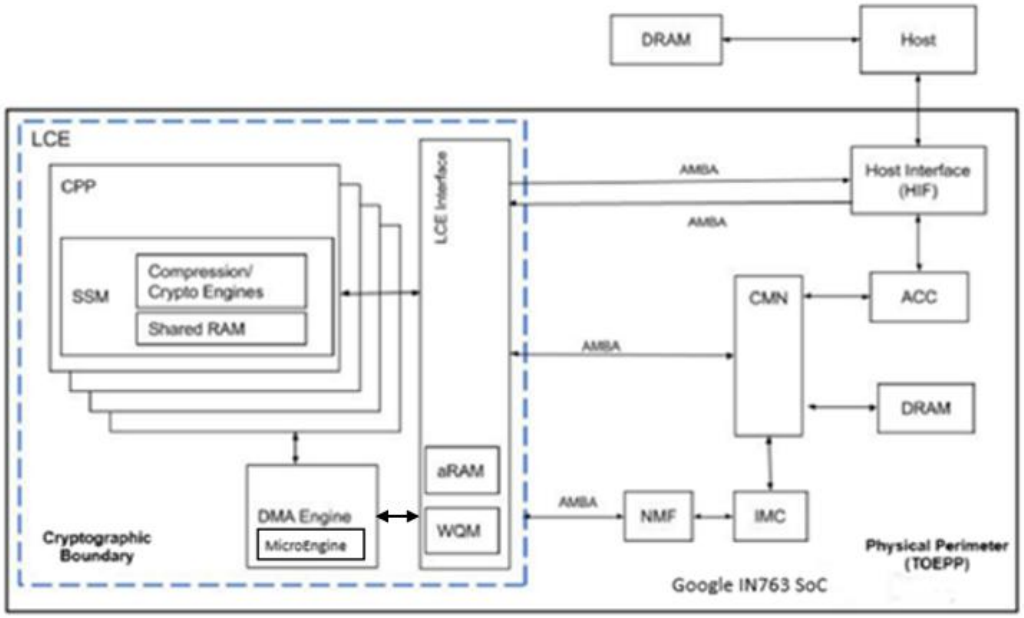


Figure 1: Block Diagram



Figure 2: Single-chip picture

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

Public Material – May be reproduced only in its original entirety (without revision).

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
IN763 SoC	3.1	0.1.a0.144E	Intel LCE 3.1	N/A

Table 2: Tested Module Identification – Hardware

2.3 Excluded Components

No components have been excluded from the cryptographic boundary of the module.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Types of Services	Status Indicator
Approved Mode	The module only supports one Approved mode of operation (and a non-Approved mode of operation); No configuration of the module or installation steps are required from the operator as stated in Section 11.1.; When the module is powered on its pre-operational self-tests are executed without any operator intervention; The module's cryptographic functions will only be available after all self-tests have passed successfully; If any of the self-tests fail, the module will transition to an error state	Approved	Successful completion of the approved services and return of the indicator 0x80
Non-Approved Mode	Invocation of a non-Approved service causes the module to enter the non-Approved mode of operation	Non-Approved	Successful completion of the non-approved services and return of the indicator 0x84

Table 3: Modes List and Description

Degraded Mode Description:

The module does not support a degraded mode of operation.

2.5 Algorithms

Approved Algorithms:

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm	CAVP Cert	Properties	Reference
AES-CMAC	A7381	Direction - Generation Key Length - 128, 192, 256 MAC Length - MAC Length: 128 Message Length - Message Length: 8-1024 Increment 8	SP 800-38B
AES-CTR	A6137	Direction - Decrypt, Encrypt Key Length - 128, 256 Payload Length - Payload Length: 128 Supports Counter larger than maximum value - No Incremental Counter - Yes Counter Tests Performed - Yes	SP 800-38A
AES-ECB	A6137	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38A
AES-GCM	A6137	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 256 Tag Length - 128 IV Length - IV Length: 96, 128 Payload Length - Payload Length: 128, 192, 256, 384, 512, 640, 1024, 2048, 4096, 65536 AAD Length - AAD Length: 128, 256, 512, 1024, 65536	SP 800-38D
AES-GMAC	A6137	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 256 Tag Length - 128 IV Length - IV Length: 96, 128 AAD Length - AAD Length: 128, 192, 256, 384, 512, 640, 768, 896, 1024, 1152, 1280, 1408, 1536, 1664, 2048, 4096, 8192, 16384, 32768, 65536	SP 800-38D
AES-XTS Testing Revision 2.0	A6137	Direction - Decrypt, Encrypt Key Length - 128, 256 Payload Length - Payload Length: 128-32832 Increment 8 Tweak Mode - Hex Data Unit Length Matches Payload Length - Yes	SP 800-38E
ECDSA SigGen (FIPS186-5)	A8016	Curve - P-384 Hash Algorithm - SHA2-384 Component - Yes	FIPS 186-5
HMAC-SHA-1	A6137	MAC - MAC: 160 Key Length - Key Length: 512	FIPS 198-1
HMAC-SHA2-224	A6137	MAC - MAC: 224 Key Length - Key Length: 512	FIPS 198-1

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-256	A6137	MAC - MAC: 256 Key Length - Key Length: 512	FIPS 198-1
HMAC-SHA2-384	A6137	MAC - MAC: 384 Key Length - Key Length: 1024	FIPS 198-1
HMAC-SHA2-512	A6137	MAC - MAC: 512 Key Length - Key Length: 1024	FIPS 198-1
HMAC-SHA3-224	A6137	MAC - MAC: 224 Key Length - Key Length: 1152	FIPS 198-1
HMAC-SHA3-256	A6137	MAC - MAC: 256 Key Length - Key Length: 1088	FIPS 198-1
HMAC-SHA3-384	A6137	MAC - MAC: 384 Key Length - Key Length: 832	FIPS 198-1
HMAC-SHA3-512	A6137	MAC - MAC: 512 Key Length - Key Length: 576	FIPS 198-1
RSA Decryption Primitive Sp800-56Br2 (CVL)	A8016	Modulo - 3072 Key Format - Standard Public Exponent Mode - random	SP 800-56B Rev. 2
SHA-1	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 180-4
SHA2-224	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 180-4
SHA2-256	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 180-4
SHA2-384	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 180-4
SHA2-512	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 180-4
SHA3-224	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 202
SHA3-256	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 202
SHA3-384	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 202
SHA3-512	A6137	Message Length - Message Length: 128-65408 Increment 32	FIPS 202

Table 4: Approved Algorithms

Vendor-Affirmed Algorithms:

Vendor-Affirmed Algorithms do not apply to the module.

Non-Approved, Allowed Algorithms:

Non-Approved, Allowed Algorithms do not apply to the module.

Public Material – May be reproduced only in its original entirety (without revision).

Non-Approved, Allowed Algorithms with No Security Claimed:

Name	Caveat	Use and Function
Snappy	No Security Claimed	Compression Algorithm
zStandard	No Security Claimed	Compression Algorithm
CRC	No Security Claimed	Cyclic Redundancy Check

Table 5: Non-Approved, Allowed Algorithms with No Security Claimed

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
RSA primitives	RSA signature generation/verification/encrypt/decrypt (DPA-resistant implementation)
ECDSA primitives	ECDSA verification (signature verification) and ECDSA signature generation/verification (DPA-resistant implementation)
EdDSA primitives	EdDSA signature generation/verification

Table 6: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Encryption/Decryption	BC-Auth BC-AuthDecrypt BC-AuthEncrypt BC-UnAuth BC-UnAuthDecrypt BC-UnAuthEncrypt	Encryption/decryption of payloads		AES-ECB: (A6137) AES-CTR: (A6137) AES-GCM: (A6137) AES-XTS Testing Revision 2.0: (A6137)
Hash/message digest	SHA	Hashing of payloads		SHA-1: (A6137) SHA2-224: (A6137) SHA2-256: (A6137) SHA2-384: (A6137) SHA2-512: (A6137) SHA3-224:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				(A6137) SHA3-256: (A6137) SHA3-384: (A6137) SHA3-512: (A6137)
Authentication	BC-Auth MAC	Payload Authentication		AES-GMAC: (A6137) HMAC-SHA-1: (A6137) HMAC-SHA2-224: (A6137) HMAC-SHA2-256: (A6137) HMAC-SHA2-384: (A6137) HMAC-SHA2-512: (A6137) HMAC-SHA3-256: (A6137) HMAC-SHA3-384: (A6137) HMAC-SHA3-512: (A6137) SHA-1: (A6137) SHA2-224: (A6137) SHA2-256: (A6137) SHA2-384: (A6137) SHA2-512: (A6137) SHA3-224: (A6137) SHA3-256: (A6137) SHA3-384: (A6137)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				SHA3-512: (A6137) HMAC-SHA3-224: (A6137)
AES GCM IV Construction	BC-Auth	Deterministic construction per section 8.2.1 of NIST SP 800-38D		AES-GCM: (A6137)
[EVM] AES-CMAC Verification	BC-Auth	AES-CMAC Verification by the bound Integrated Compute Complex (ICC Cert. #5558) for the Pre-operational Integrity Test performed on the firmware residing within the LCE module		AES-CMAC: (A7381)
Signature Generation	DigSig-SigGen	ECDSA Signature Generation		ECDSA SigGen (FIPS186-5): (A8016)
RSA Decryption	KEM-Decap	RSA Decryption Primitive		RSA Decryption Primitive Sp800-56Br2: (A8016)

Table 7: Security Function Implementations

2.7 Algorithm Specific Information

IG C.I

The module implements a check to verify that Key1!=Key2 for the AES-XTS implementation. This verification is performed prior to using the keys in encryption/decryption per the test listed in the Section 10.2 Conditional Self-Tests table of this document. AES-XTS keys (i.e., Key_1 and Key_2) entered into the module shall be generated and/or established independently according to NIST SP 800-133rev2, Section 6.3. for an approved use of AES-XTS.

IG C.C

The module meets the FIPS 140-3 IG C.C requirements for its SHA-3 implementations in accordance with Resolution 2 a. and b. in the IG.

IG C.H

Public Material – May be reproduced only in its original entirety (without revision).

The Initialization Vector (IV) used by LCE is considered a PSP. The IV is constructed deterministically per section 8.2.1 of SP 800-38D. GCM IV construction is performed within the LCE boundary in accordance with FIPS 140-3 IG C.H scenario 3 using a deterministic method. The IV is constructed using a 32-bit salt allowing for 2^32 possible values and 64-bit counter value to form a 96 bit IV. Per the requirements specified in Section 8 in NIST SP 800-38D, the probability that the authenticated encryption function will be invoked with the same IV and the same key on two (or more) distinct sets of input data is no greater than 2^-32.

The maximum operating rate of the LCE encryption engine is 25MPPS. Operating at full capacity, it would take 23,300 years before a counter reset, thus, the module would cease operation prior to exhausting the 2^64 counter values. If power is removed from the module, a new key and IV are generated as there is no persistent memory of the last IV used.

SHA-1
Usage of SHA-1 is non-approved for digital signature generation and deprecated through December 31, 2030, for non-digital signature applications.

2.8 RBG and Entropy

The module does not perform any SSP generation and thus an entropy source is not contained in/used by the module.

2.9 Key Generation

The module does not provide any SSP generation service or perform SSP generation for any of its Approved algorithms. The external host or ACC software writes requests to a ring buffer and doorbells the device when a key is to be loaded onto the device. The request descriptors (originating from Host CPU or ACC) provide the location for keys and payload data and the module in turn fetches the keys and payloads from external DRAM via DMA. For encryption operations, the IV is generated and output to DRAM (via DMA) to the address specified in the request descriptor. For decryption, the IV is retrieved from DRAM (via DMA).

The cryptographic module does not provide any asymmetrical algorithms or SSP establishment methods.

2.10 Key Establishment

The module does not support SSP establishment.

2.11 Industry Protocols

The module does not support any industry protocols.

3 Cryptographic Module Interfaces

Public Material – May be reproduced only in its original entirety (without revision).

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
AMBA (On-chip system bus)	Data Input	Request interface DMA reads from on-chip; DRAM or external DRAM via HIF (key/IV, payloads); Compression interface reads (via CMN); CSR writes from NMF (firmware load)
AMBA (On-chip system bus)	Data Output	Request interface DMA writes to on-chip; DRAM or external DRAM via HIF (processed data); Compression interface reads (via CMN)
AMBA (On-chip system bus)	Control Input	CSR writes from the NMF (initialization); Request interface DMA reads via HIF (request descriptors); Push Interface (doorbell)
AMBA (On-chip system bus)	Status Output	CSR status output to the NMF (self-heartbeat information, interrupts, self-test status); Request interface DMA writes via HIF (response descriptors)
Physical power connector	Power	Power provided to the module

Table 8: Ports and Interfaces

The module does not support a control output interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

The module does not support authentication.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
CO	Role	Crypto Officer	None

Table 9: Roles

The module implements the Crypto Officer (CO) role alone. The CO role has access to all the module's services. The Crypto Officer role is implicitly assumed by the entity accessing services implemented by the module. The module also supports concurrent operators.

4.3 Approved Services

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Payload Encryption using AES-GCM, CTR, XTS	Payload Encryption	Global Approved Mode indicator (0x80) combined with the successful completion of the service	Payload to be encrypted	Encrypted payload	Encryption/ Decryption AES GCM IV Construction	CO - Cipher -Auth Key: W,E - GCM Initialization Vector : G,E
Payload Decryption using AES-GCM, CTR, XTS	Payload Decryption	Global Approved Mode indicator (0x80) combined with the successful completion of the service	Payload to be decrypted	Decrypted payload	Encryption/ Decryption AES GCM IV Construction	CO - Cipher -Auth Key: W,E - GCM Initialization Vector : G,E
Perform self-tests	On demand self-test execution on rebooting the module	Global Approved Mode indicator (0x80	Power cycle	Status	None	CO

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		) combined with the successful completion of the service and 1				
Initialization	Module initialization	Global Approved Mode indicator (0x80) combined with the successful completion of the service and 1	Power	Status	[EVM] AES-CMAC Verification	CO
Perform zeroisation	Reboot/power-cycle of the module zeroises all SSPs as they are stored ephemerally in the SRAM	Global Approved Mode indicator (0x80) combined with the succe	Power cycle/reboot	Status	None	CO - Cipher-Auth Key: Z - HMAC-Auth Key: Z - GCM Initialization Vector: Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		ssful completion of the service and 1				
Show Status	Approved mode status provided by the module	Global Approved Mode indicator (0x80) combined with the successful completion of the service	N/A	Status	None	CO
Show module's versioning information	Retrieval of the module ID, firmware version and hardware version	Global Approved Mode indicator (0x80) combined with the successful completion of the service	sv.mmg.nss.lceapf0.pf.mmio.cmvp_fw_version_lo and sv.mmg.nss.lceapf0.pf.mmio.cmvp_fw_version_hi	Firmware version, Hardware version and Module ID	None	CO

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Hashing - SHA-1, SHA-2, and SHA-3 message digest	Payload hashing	Global Approved Mode indicator (0x80) combined with the successful completion of the service	Payload to be hashed	Hash of the payload	Hash/message digest	CO
Payload Authentication using AES-GMAC and HMAC	Payload Authentication	Global Approved Mode indicator (0x80) combined with the successful completion of the service	Payload to be authenticated, keyed hash key	Status	Authentication	CO - HMAC-Auth Key: W,E - Cipher-Auth Key: W,E
Compression	Snappy/z Standard	Global Approved Mode indica	Data to be compressed	Compressed form of the input data	None	CO

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		tor (0x80) combined with the successful completion of the service				
DMA	CRC function on DMA reads/writes	Global Approved Mode indicator (0x80) combined with the successful completion of the service	N/A	Status of the CRC	None	CO
Signature Generation	Signature generation using ECDSA	Global Approved Mode indicator (0x80) combined with the succe	Message to be signed	Signature	Signature Generation	CO - ECDSA Private Key: E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		ssful completion of the service				
RSA Decryption	RSA private-key operation used during RSA-based key establishment	Global Approved Mode indicator (0x80) combined with the successful completion of the service	Keying material to be decrypted	Decrypted keying material	RSA Decryption	CO - RSA Private Key: E

Table 10: Approved Services

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

4.4 Non-Approved Services

Name	Description	Algorithms	Role
Signature Generation/Verification	RSA, ECDSA and EdDSA primitives used in signature generation/verification (hash generated externally)	RSA primitives ECDSA primitives EdDSA primitives	CO
Encrypt/Decrypt	RSA primitives used for encryption/decryption	RSA primitives	CO

Public Material – May be reproduced only in its original entirety (without revision).

Table 11: Non-Approved Services

4.5 External Software/Firmware Loaded

In accordance with IG 10.3.F Resolution 3 and Additional Comments 2, 3 a. and 8, the module supports firmware loading (the entirety of the firmware is replaced) by the bound FIPS 140-3 validated Integrated Compute Complex (ICC) module (Cert. #5558).

4.6 Bypass Actions and Status

The module does not support a bypass capability.

4.7 Cryptographic Output Actions and Status

The module does not support self-initiated cryptographic output.

5 Software/Firmware Security

5.1 Integrity Techniques

The Pre-operational Firmware Integrity Test for the LCE module is performed by the bound FIPS 140-3 validated Integrated Compute Complex (ICC) module (Cert. #5558). The latter implements an AES-CMAC verification as the integrity technique. The AES-CMAC key used for the integrity technique is considered a non-SSP. The firmware loaded onto the module is a binary file.

5.2 Initiate on Demand

The Pre-operational Firmware Integrity Test and Cryptographic Algorithm Self-Tests (CAST) can be initiated on demand by rebooting the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Non-Modifiable

How Requirements are Satisfied :

The module is a sub-chip cryptographic subsystem implemented within a single chip embodiment. The procurement build and configuration procedures are controlled by the

Public Material – May be reproduced only in its original entirety (without revision).

manufacturer. Since firmware can be loaded onto the module, the operational environment is considered to be limited.

The module resides within the following single-chip:

- IN763 SoC

6.2 Configuration Settings and Restrictions

The configuration of the module is infeasible apart from loading of the module firmware onto it.

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Standard passivation applied on the IN763 SoC; The SOC/single chip is commercial grade in regard to power and voltage ranges, temperature, reliability, and shock and vibration	None	None

Table 12: Mechanisms and Actions Required

The module is implemented as part of the IN763 SoC, which is the physical perimeter of the module. The IN763 SoC is a production grade single-chip with standard passivation applied to it and hence conforms to the Security Level 1 requirements for this section.

8 Non-Invasive Security

8.1 Mitigation Techniques

Please see Section 12 for the non-invasive security mitigations implemented within the module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
Shared RAM	Temporary/ephemeral storage	Dynamic

Table 13: Storage Areas

Public Material – May be reproduced only in its original entirety (without revision).

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Retrieved from on-chip or external DRAM via DMA	on-chip or external DRAM	Shared RAM	Plaintext	Manual	Electronic	
Output to external DRAM via DMA	Shared RAM	external DRAM	Plaintext	Manual	Electronic	

Table 14: SSP Input-Output Methods

The module supports MD/EE per FIPS 140-3 IG 9.5.A.

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Zeroised on reset	SSPs stored in shared RAM	Temporarily/ephemerally stored SSPs zeroised on reset	Operator Initiated

Table 15: SSP Zeroization Methods

The module does not provide any persistent storage. All SSPs are stored in the internal shared RAM (i.e. static registers). Once the keys are written to shared RAM, they are not readable from outside the module. The shared RAM is zeroised upon reset, which may be performed via the System Control block (SYSCON) within the Integrated Compute Complex (ICC) module. While this action will clear all secrets internal to LCE, it will also leave the module in a non-usable state as standalone reset is not supported. This process will result in returning the module to its default state.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Cipher-Auth Key	Encryption and decryption (and optionally authentication) of payload	AES-GCM, GMAC, CTR, XTS 128, 256-bit value - 128, 256 bits	Symmetric key - CSP			Encryption/Decryption Authentication

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
HMAC-Auth Key	MAC key used for payload authentication	HMAC-SHA1, HMAC-SHA2-224, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, HMAC-SHA3-224, HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 - 160-512 bits	MAC key - CSP			Authentication
GCM Initialization Vector	IV used for GCM encryption/decryption	96 bits - N/A	Initialization vector - PSP	AES GCM IV Construction		Encryption/Decryption
ECDSA Private Key	Private key used for signature generation	P-384 - 192 bits	Private Key - CSP			Signature Generation
RSA Private Key	Private key used in RSA decryption	3072 - 128 bits	Private Key - CSP			RSA Decryption

Table 16: SSP Table 1

Public Material – May be reproduced only in its original entirety (without revision).

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Cipher-Auth Key	Retrieved from on-chip or external DRAM via DMA	Shared RAM:Plaintext	Until reset	Zeroised on reset	GCM Initialization Vector:Used With
HMAC-Auth Key	Retrieved from on-chip or external DRAM via DMA	Shared RAM:Plaintext	Until reset	Zeroised on reset	
GCM Initialization Vector	Retrieved from on-chip or external DRAM via DMA Output to external DRAM via DMA	Shared RAM:Plaintext	Until reset	Zeroised on reset	Cipher-Auth Key:Used With
ECDSA Private Key	Retrieved from on-chip or external DRAM via DMA	Shared RAM:Plaintext	Until reset	Zeroised on reset	
RSA Private Key	Retrieved from on-chip or external DRAM via DMA	Shared RAM:Plaintext	Until reset	Zeroised on reset	

Table 17: SSP Table 2

9.5 Transitions

- The usage of SHA-1 is deprecated through December 31, 2030, for non-digital signature applications and disallowed i.e., non-approved and not allowed thereafter.
- Usage of keys with a minimum-security strength of 112 bits i.e. RSA mod 2048 bits and ECDSA P-224 curve is deprecated through December 31, 2030, and disallowed thereafter. January 1, 2031, and beyond a minimum of 128 bits of security strength will be required.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
AES-CMAC performed by the bound ICC module (Cert. #5558)	256 bits	KAT	SW/FW Integrity	1	MAC Verify [EVM]

Table 18: Pre-Operational Self-Tests

The pre-operational firmware integrity test for the module is implemented by the FIPS 140-3 validated Integrated Compute Complex (ICC) module (Cert. #5558). The module is designed to execute the Cryptographic Algorithm Self-Tests (CAST) automatically when it is initialized on every boot. These tests do not require operator intervention to run.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-XTS Encrypt	256 bits; Encrypt	KAT	CAST	1	-	On every boot, Manually
AES-XTS Decrypt	256 bits; Decrypt	KAT	CAST	1	-	On every boot, Manually
AES-XTS Key1!=Key2 check	256 bits, check for Key1 != Key2	Comparision of Key1 and Key2	CAST	1	-	Prior to use of an XTS key
AES-GCM Encrypt	256 bits; Encrypt	KAT	CAST	1	-	On every boot, Manually
AES-GCM Decrypt	256 bits; Decrypt	KAT	CAST	1	-	On every boot, Manually
AES-CTR Encrypt	256 bits; Encrypt	KAT	CAST	1	-	On every boot, Manually
AES-CTR Decrypt	256 bits; Decrypt	KAT	CAST	1	-	On every boot, Manually
HMAC-SHA-1	PRF: SHA-1	KAT	CAST	1	-	On every boot, Manually
HMAC-SHA2-256	PRF: SHA2-256	KAT	CAST	1	-	On every boot, Manually

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-512	PRF: SHA2-512	KAT	CAST	1	-	On every boot, Manually
HMAC-SHA3-256	PRF: SHA3-256	KAT	CAST	1	-	On every boot, Manually
HMAC-SHA3-512	PRF: SHA3-512	KAT	CAST	1	-	On every boot, Manually
AES-CMAC (A7381)	Key Length: 256 bits	KAT	CAST	"Notice: GSP: crypto self-test passed"	[EVM]	On every boot, Manually
ECDSA SigGen (FIPS186-5) (A8016)	Curve: P-384, SHA2-384	KAT	CAST	1	-	On every boot, Manually
RSA Decryption Primitive Sp800-56Br2 (A8016)	Modulus: 3072	KAT	CAST	1-		On every boot, Manually

Table 19: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CMAC performed by the bound ICC module (Cert. #5558)	KAT	SW/FW Integrity	On boot, On demand	Programmatically, Manually via reboot

Table 20: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-XTS Encrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-XTS Decrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-XTS Key1!=Key2 check	Comparision of Key1 and Key2	CAST	Prior to use of an XTS	Programmatically, Manually via reboot followed by invoking the Payload Encryption using AES-

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
			key, On demand	GCM, CTR, XTS/Payload Decryption using AES-GCM, CTR, XTS service
AES-GCM Encrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-GCM Decrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-CTR Encrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-CTR Decrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
HMAC-SHA-1	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
HMAC-SHA2-256	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
HMAC-SHA2-512	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
HMAC-SHA3-256	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
HMAC-SHA3-512	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-CMAC (A7381)	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
ECDSA SigGen (FIPS186-5) (A8016)	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
RSA Decryption Primitive Sp800-56Br2 (A8016)	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot

Table 21: Conditional Periodic Information

10.4 Error States

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Conditions	Recovery Method	Indicator
CRYPTO_DISABLE state	Failure of the Cryptographic Algorithm Self-Tests causes the module to enter this state, where no functions can be executed; Each algorithm engine is tested, and the result of the test is placed in a register; If any status bit in the register indicates a failure, it will enter the CRYPTO_DISABLE state at which point the module halts all services with the exception of limited control input	Failure of Cryptographic Algorithm Self-Tests	An operator can attempt to reset the state by cycling the power; However, the repeated failure of a self-test may require the module to be replaced	Status bit in a register within the module
Soft Error state	Failure of the AES-XTS Key1!=Key2 check per IG C.I causes the module to enter this state, the individual "slice" where the error occurred will halt execution, an error is returned, and no payload data is written back from the device; The encryption/decryption request will fail as a result of failure of this test	Failure of AES-XTS check for Key1!=Key2	The module will continue to process payload encryption and decryption operations if the subsequent keys are valid and not equal	An error is returned via zeroing the "key valid" bit in the CSR
Hard Error state	If the Firmware Integrity Test fails, the module will not be initialized and will remain in an uninitialized state until the bound ICC cryptographic module (Cert. #5558) can successfully re-initialize the device	Failure of the Firmware Integrity Test	The module would have to be re-initialized by the bound ICC module (Cert. #5558)	ERROR: Firmware hash for image 99 DOES NOT MATCH, ERROR: Ice abort image load

Table 22: Error States

10.5 Operator Initiation of Self-Tests

The operator can initiate the Pre-operational Firmware Integrity Test and the Cryptographic Algorithm Self-Tests (CAST) by rebooting the module. The AES-XTS check for Key1!=Key2 per

Public Material – May be reproduced only in its original entirety (without revision).

the FIPS 140-3 IG C.I, can be done by requesting the AES-XTS service from the module given that this check is performed prior to using the XTS keys provided by the caller/operator.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

No configuration of the module or installation steps are required from the operator. When the module is powered on its Pre-operational Integrity Test and Cryptographic Algorithm Self-Tests (CAST) are executed without any operator intervention.

The module is not considered to be in the Approved mode of operation until the firmware image has been successfully verified by the bound Integrated Compute Complex (ICC) module (Cert. #5558). The module's cryptographic functions will only be available after all self-tests have passed successfully. If any of the self-tests fail the module will transition to an error state.

11.2 Administrator Guidance

Additional guidance does not apply for the operation of the module apart from that specified in Sections 2 and 3 of this document and other subsections under this section.

11.3 Non-Administrator Guidance

Additional guidance does not apply for the operation of the module apart from that specified in Sections 2 and 3 of this document and other subsections under this section.

11.4 Maintenance Requirements

Additional maintenance requirements do not apply to the module.

11.5 End of Life

A reset of the IN763 SoC can be used to zeroise all SSPs contained within the module i.e. to securely sanitize the module.

12 Mitigation of Other Attacks

The module implements mitigation against Differential Power Analysis (DPA) attacks for the RSA and ECDSA algorithms.

Public Material – May be reproduced only in its original entirety (without revision).