Google, LLC

Inline Crypto Engine (ICE)

FIPS 140-3 Non-Proprietary Security Policy

Google, LLC 2026

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 8
2.4 Modes of Operation 8
2.5 Algorithms 9
2.6 Security Function Implementations 10
2.7 Algorithm Specific Information 11
2.8 RBG and Entropy 13
2.9 Key Generation 13
2.10 Key Establishment 13
2.11 Industry Protocols 14
3 Cryptographic Module Interfaces 14
3.1 Ports and Interfaces 14
4 Roles, Services, and Authentication 14
4.1 Authentication Methods 14
4.2 Roles 15
4.3 Approved Services 15
4.4 Non-Approved Services 18
4.5 External Software/Firmware Loaded 18
4.6 Bypass Actions and Status 19
4.7 Cryptographic Output Actions and Status 19
5 Software/Firmware Security 19
5.1 Integrity Techniques 19
5.2 Initiate on Demand 19
6 Operational Environment 19
6.1 Operational Environment Type and Requirements 19
6.2 Configuration Settings and Restrictions 20
7 Physical Security 20
7.1 Mechanisms and Actions Required 20

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

8 Non-Invasive Security 20
8.1 Mitigation Techniques 20
9 Sensitive Security Parameters Management 20
9.1 Storage Areas 20
9.2 SSP Input-Output Methods 21
9.3 SSP Zeroization Methods 21
9.4 SSPs 22
10 Self-Tests 25
10.1 Pre-Operational Self-Tests 25
10.2 Conditional Self-Tests 26
10.3 Periodic Self-Test Information 26
10.4 Error States 27
10.5 Operator Initiation of Self-Tests 27
11 Life-Cycle Assurance 28
11.1 Installation, Initialization, and Startup Procedures 28
11.2 Administrator Guidance 28
11.3 Non-Administrator Guidance 28
11.4 Maintenance Requirements 28
11.5 End of Life 28
12 Mitigation of Other Attacks 28

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 8
Table 3: Modes List and Description 9
Table 4: Approved Algorithms 9
Table 5: Security Function Implementations 11
Table 6: Ports and Interfaces 14
Table 7: Roles 15
Table 8: Approved Services 18
Table 9: Mechanisms and Actions Required 20
Table 10: Storage Areas 20
Table 11: SSP Input-Output Methods 21
Table 12: SSP Zeroization Methods 21
Table 13: SSP Table 1 23
Table 14: SSP Table 2 25
Table 15: Pre-Operational Self-Tests 26
Table 16: Conditional Self-Tests 26
Table 17: Pre-Operational Periodic Information 27
Table 18: Conditional Periodic Information 27
Table 19: Error States 27

List of Figures

Figure 1: Block Diagram 7
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:
http://csrc.nist.gov/groups/STM/cmvp/index.html

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

About this Document

This non-proprietary Cryptographic Module Security Policy for the Google, LLC “Inline Crypto Engine (ICE)” (Hardware version: 2.0) 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 “Inline Crypto Engine (ICE)” may also be referred to as “ICE” 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	N/A
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. Also, the module does not contain any software/firmware and thus the requirements per the Software/Firmware Security section are inapplicable as well.

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

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The Inline Crypto Engine (ICE) module is a sub-chip cryptographic subsystem in the Google IN763 System-on-a-chip (SoC). The module provides a cryptographic engine supporting cryptographic offload for Internet Protocol Security (IPsec) and PSP Security Protocol (PSP) protocols. The module processes network infrastructure packets in both the ingress and egress directions. In addition to providing cryptographic primitives for security protocols, it also provides packet integrity authentication and anti-replay protection.

The module includes the Ingress and Egress Pipelines and Egress Arbiter and Class of Service (CoS) Queues (ECQ). Security protocol processing and cryptography services performed by the module include full data-plane protocol termination for IPsec ESP[1], and PSP. ESP is implemented using AES-GCM and AES-GMAC. PSP supports AES-GCM alone. The module also includes KBKDF used for PSP key derivation.

Within the ICE IP block are the Ingress and Egress Pipeline. The Ingress and Egress Pipelines each contain a GCM/GMAC encryption/decryption and authentication engine (identified as ICE Ingress and ICE Egress, respectively). ICE Ingress also contains KBKDF. In addition, the Ingress and Egress pipelines provide an Ingress and Egress Security Association Database (SADB) for decrypting keys imported from or encrypting keys exported to the SADB in the external on-chip SRAM or host DRAM using GCM.

TX and RX network packet data and metadata (PMD) are received on either the Ingress or Egress data path to be processed by the module.

The module is connected to other subsystems via an AMBA[2] on-chip system bus, though as described below, the specific protocol may vary between Advanced eXtensible Interface (AXI) and AXI Coherency Extensions (ACE/ACE-Lite). AXI is used for CSR writes (via NMF) when programming PSP master keys and SADB keys. AXI4 streams are used for config packets. ACE/ACE-Lite is used when the module reads/writes data to and from SADB memory (external DRAM or on-chip SRAM in the ACC).

Through the Security Association Database (SADB), ICE supports the ability to add, delete and read SA entries. When a Security Association (SA) is created, a config packet is read from FIFO buffers in the Ingress and Egress data paths. SA keys, data and statistics are fetched from and stored to external DRAM or ACC SRAM using a single ACE-Lite interface in each direction. The SADB is managed in two SA tables, one each for the Ingress and Egress directions. A separate AXI Control Status Register (CSR) configuration interface is provided and connected to the NIC Management Fabric (NMF).

[1] Note that data plane functionality represents the second-phase IPSec SA payload encryption. The IKE protocol is not implemented by the module.
[2] ARM Microcontroller Bus Architecture

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

Access to the CSRs and external DRAM is controlled by the underlying SoC configuration. Access requests are passed through the NIC Management Fabric which provides fixed routes between all AMBA-connected endpoints.

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 ICE IP Block i.e., the Inline Crypto Engine (ICE) (Hardware version: 2.0), a sub-chip within the IN763 SoC/single-chip. The module only supports a single mode of operation, the Approved mode, where only approved cryptographic functions and services are available.

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.

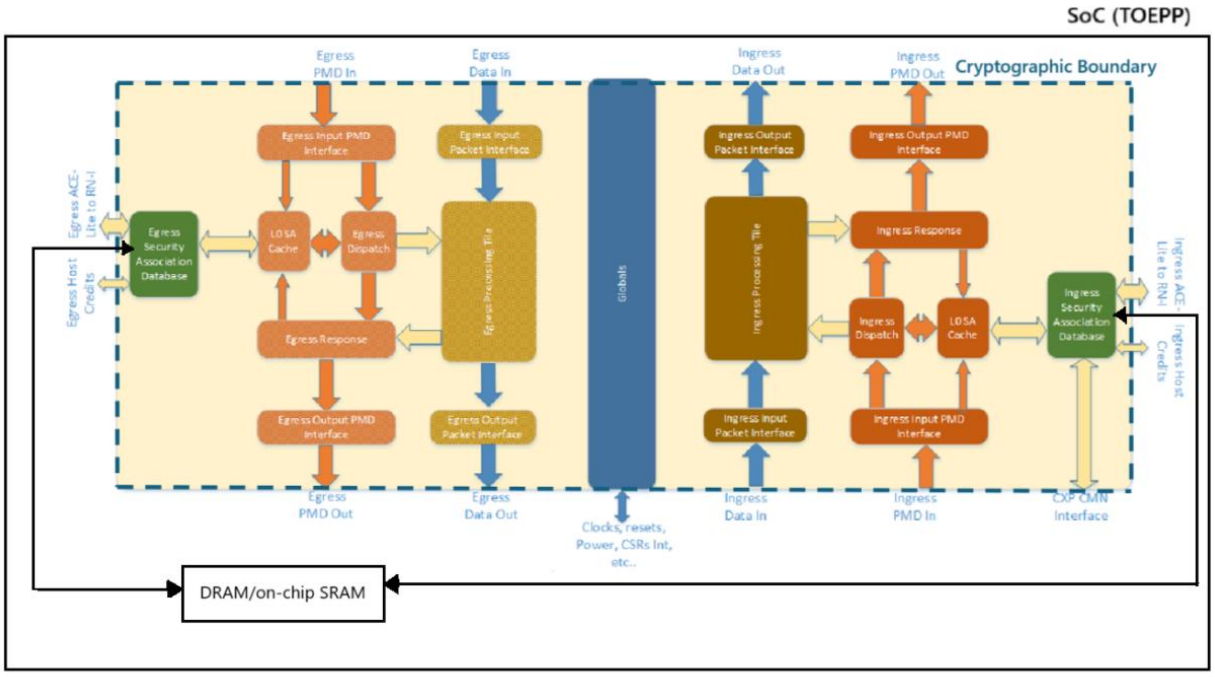


Figure 1: Block Diagram

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



Figure 2: Single-chip picture

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
IN763 A0 SoC	2.0	N/A	N/A	None

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 an 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 cryptographic algorithm 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	Approved	Successful completion of the self-tests (register ICM_FATAL_ECR0.kat_fail set to 0 implies that the module is operational in the Approved mode)

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

Mode Name	Description	Types of Services	Status Indicator
	any of the self-tests fail, the module will transition to an error state		

Table 3: Modes List and Description

Degraded Mode Description:

The module does not support a degraded mode of operation.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CMAC	A6038	Direction - Generation Key Length - 256	SP 800-38B
AES-GCM	A6038	Direction - Decrypt IV Generation - External IV Generation Mode - 8.2.1 Key Length - 128, 256	SP 800-38D
AES-GCM	A6039	Direction - Encrypt IV Generation - External IV Generation Mode - 8.2.1 Key Length - 128, 256	SP 800-38D
AES-GCM	A6040	Direction - Decrypt, Encrypt IV Generation - External IV Generation Mode - 8.2.1 Key Length - 256	SP 800-38D
AES-GMAC	A6038	Direction - Decrypt IV Generation - External IV Generation Mode - 8.2.1 Key Length - 128, 256	SP 800-38D
AES-GMAC	A6039	Direction - Encrypt IV Generation - External IV Generation Mode - 8.2.1 Key Length - 128, 256	SP 800-38D
KDF SP800-108	A6038	KDF Mode - Counter Supported Lengths - Supported Lengths: 128-256 Increment 128	SP 800-108 Rev. 1

Table 4: Approved Algorithms

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

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.

Non-Approved, Allowed Algorithms with No Security Claimed:

Non-Approved, Allowed Algorithms with No Security Claimed do not apply to the module.

Non-Approved, Not Allowed Algorithms:

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

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Ingress Authenticated Block Cipher	BC-Auth	Used for packet data decryption		AES-GCM: (A6038)
Ingress Message Authentication	MAC	Used for data authentication		AES-GMAC: (A6038)
Key Derivation	KBKDF	Used to derive PSP encryption keys		AES-CMAC: (A6038) KDF SP800-108: (A6038)
Egress Authenticated Block Ciper	BC-Auth	Used for packet data encryption		AES-GCM: (A6039)
Egress Mesage Authentication	MAC	Used for data authentication		AES-GMAC: (A6039)
KTS-1	KTS-Unwrap KTS-Wrap	SADB Block Cipher: Used for key encryption and decryption (i.e. wrapping and unwrapping respectively)	Standard:SP 800-38F IG D.G:approved method from IG D.G Caveat:Key establishment methodology provides 256 bits	AES-GCM: (A6040)

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

Name	Type	Description	Properties	Algorithms
			of security strength	
AES GCM IV Construction	UNK	Constructed deterministically per Section 8.2.1 of NIST SP 800-38D and IG C.H Scenario 4 or 5 (based on the IV construction method) details of the construction for the three different use cases are in Section 2.7 (IG C.H) of this document		AES-GCM: (A6038, A6039, A6040)

Table 5: Security Function Implementations

2.7 Algorithm Specific Information

IG D.M
In accordance with NIST SP 800-133r2 Section 6.2.2 (Symmetric Keys Derived from a Pre-Existing Key) and IG D.M, the module uses the NIST SP 800-108r1 KDF for derivation of symmetric keys.

IG C.H

Usage of AES-GCM in the Module

The Initialization Vector (IV) used by ICE is considered an SSP. In all the supported cases the AES-GCM IV for encryption is constructed deterministically per Section 8.2.1 of NIST SP 800-38D. AES-GCM IV construction is performed within the ICE boundary in accordance with FIPS 140-3 IG C.H Scenario 4 (for Secure SADB) and Scenario 5 (for PSP and IPSec protocols). 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 ever 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}.

In accordance with the IG C.H Scenario 5, per the module and SoC design, the IV is only used in the context of the AES-GCM encryption executing the provisions of the two cases (IPsec v2.0 per RFCs 4106, 5282, and 7296 and OCP PSP Architecture Specification v1.0 (June 23, 2026) — Section 4.3 (IV Generation), Section 6 (NIC Transmit Processing)) below within which the IV was generated.

The IV construction method for each implementation is described in the following sections.

IV Construction for Secure SADB

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

The IG C.H Scenario 4 applies to this IV Construction method. The 96-bit IV is internally constructed from a 32-bit salt entered/input into the module via a CSR write and an internally managed 64-bit IV counter maintained by the module. The counter is reset to zero and increments every time an SA (security association) add operation request is submitted to the SADB. The IV increments for every SA add request.

The 64-bit counter may roll over. The module does not implement any 64-bit rollover detection on the SADB add counter. ICE supports 10,000 SADB additions per second which would require 56 million years of operation to roll over the add counter. Therefore, the vendor asserts that it is not possible for the secure SADB keys to be used with the same IV more than once for the lifetime of the Secure SADB.

IV construction: {IV counter, CSR.salt} this is in Big Endian/Network Order

IV Construction for PSP

The IG C.H Scenario 5 applies to this IV Construction method. The 96-bit IV is internally constructed using the Security Association's 32-bit SPI allowing for 2^32 possible values and 64-bit value to form a 96 bit IV. A timestamp is periodically passed into the module, which then modifies the value to ensure it is unique for every clock cycle. The module also includes protection mechanisms to ensure that the timestamp always increases and is greater than the local value. Due to an increasing counter and the fact that consecutive packets cannot be sent with the same IV, the module design ensures that the IV for PSP is guaranteed to be unique.

The module implements a 64-bit rollover (once every 213 days). Security Associations in the module are limited in lifetime to no more than ~2 days. Therefore, the vendor asserts that it is not possible for a Security Association (and its key) to be used with the same IV more than once for the PSP protocol AES-GCM encryption requests.

IV construction: {Pkt.IV, Pkt.SPI} this is in Big Endian/Network Order

IV Construction for IPSec

The IG C.H Scenario 5 applies to this IV Construction method. The 96-bit IV is internally constructed using the Security Association's 32-bit salt allowing for 2^32 possible values and the 64-bit value (sequence number) to form a 96 bit IV. When a new flow (i.e. traffic flow/sequence of related network packets traveling from a source host to a destination host) is created, a new cryptographic key and salt are generated, and the sequence number (SN) is initialized to 0. The SN is incremented for every packet processed, ensuring that each packet has a unique IV. ICE includes protection mechanisms that prevent the SN from wrapping around to 0, which guarantees that an IV can never be reused within a given flow.

The module implements a 64-bit rollover detection on the packet sequence number. The IV rollover protection is achieved by detecting SN (or extended SN i.e. ESN, when enabled) rollover and subsequently dropping all egress packets. Therefore, it is not possible for a Security Association (and its key) to be used with the same IV more than once for the IPsec ESP protocol AES-GCM encryption requests.

IV construction: {Pkt.IV, SA.salt}, in big endian.

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

In each case above, the IV is constructed within the module, specifically, the salt values of SADB, PSP, and IPSec can be retrieved from on-chip memory (outside the cryptographic boundary) and presented to the module which then constructs the IV. The salt is an external component, and the counters are internal to the module's cryptographic boundary. The counter, i.e. a 64-bit invocation field, is established and controlled by the module itself. The salt is generated together with the cryptographic key for a given traffic flow and remains fixed for the duration of that flow. After a reboot/power-cycle, a new flow is established, and a new cryptographic key and salt are generated as part of the flow setup i.e. there is no flow maintained across power-cycles.

All SSPs are stored ephemerally and input/derived in the module afresh after a reboot/power-cycle and the following restoration condition per FIPS 140-3 IG C.H Scenario 3) applies:

- In case the module's power is lost and then restored, a new key for use with the AES-GCM encryption/decryption shall be established: This is ensured since for every new flow i.e. after every reboot, a new GCM key is imported into the module.

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 caller provides the keys for encryption and/or decryption. Keys are stored in hardware registers (write-only) by the Crypto Officer. Once the keys are written to the hardware registers, they are not readable from outside the module. The module only derives the symmetric PSP encryption keys using a NIST SP 800-108r1 KDF (per NIST SP 800-133r2 Section 6.2.2 Symmetric Keys Derived from a Pre-existing Key) where the key derivation key (PSP Master Key) is not generated/established within the module but is input via a CSR write to the module).

The cryptographic module does not provide any asymmetrical algorithms.

2.10 Key Establishment

The module supports the following approved key establishment/derivation techniques:

- NIST SP 800-108 KDF for deriving PSP Encryption Keys from the PSP Master Key

- KTS-1: NIST SP 800-38D for key-transport of PSP Encryption, ESP Encryption, and ESP Authentication Keys per IG D.G (second bullet point under the Approved methods: Employing a key wrapping key, shared by the sender and receiver, together with an approved symmetric key-wrapping algorithm to wrap the keying material to be transported)

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

KTS (AES CAVP Cert. #A6040; Key establishment methodology provides 256 bits of security strength)

2.11 Industry Protocols

The module does not implement an industry protocol (such as IKE) itself but provides cryptography required in the context of the IPsec (specifically, it provides IPsec security association payload encryption). No parts of this protocol have been reviewed or tested by the CAVP and CMVP. This is in accordance with FIPS 140-3 IG D.C Resolution 3.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
AMBA (On-chip system bus)	Data Input	Inbound AXI4-Stream Packet data and config packets; Inbound SA entries from DRAM; AXI CSR writes from NMF containing PSP Master Key and Secure SADB Key
AMBA (On-chip system bus)	Data Output	Outbound SA entries to DRAM
AMBA (On-chip system bus)	Control Input	AXI CSR writes from the NMF; Control bits in Ingress packet metadata (PMD) from CXP (crypto packet processor); Control bits in Egress packet metadata (PMD) from FXP (flexible packet processor)
AMBA (On-chip system bus)	Status Output	AXI CSR status output to NMF; Status bits in packet metadata (PMD)
Power	Power	Provides power to the module

Table 6: 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.

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

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	Crypto Officer	None

Table 7: Roles

The module implements the Crypto Officer (CO) role alone. The module does not allow concurrent operators. The Crypto Officer role is implicitly assumed by the entity accessing services implemented by the module. The CO role has access to all services provided by the module. No further authentication is required. The Crypto Officer is responsible for setting up i.e. providing the SSPs needed in the context of the services supported by the module.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Payload encryption	Egress packet payload encryption for ESP and PSP	Approved mode indicator and successful completion of service	Plaintext payload	Ciphertext	Egress Authenticated Block Ciper Egress Mesage Authentication AES GCM IV Construction	Crypto Officer - ESP Encryption Key: W,E - PSP Encryption Key: W,E - ESP Authentication Key: W,E - AES GCM IV: G,E
Payload decryption	Ingress packet payload decryption for ESP and PSP	Approved mode indicator and successful completion of service	Ciphertext payload	Plaintext	Ingress Authenticated Block Cipher Ingress Message Authentication AES GCM IV	Crypto Officer - ESP Decryption Key: W,E - PSP Decryption Key: W,E - ESP

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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					Construction	Authentication Key: W,E - AES GCM IV: G,E
PSP and ESP Key encryption	Encrypt PSP and ESP keys	Approved mode indicator and successful completion of service	Keys to be encrypted	Encrypted keys	KTS-1 AES GCM IV Construction	Crypto Officer - Secure SADB Key: W,E - ESP Encryption Key: W,E - PSP Encryption Key: W,E - AES GCM IV: G,E
PSP Key Derivation	Derive PSP encryption key	Approved mode indicator and successful completion of service	PSP Master Key	PSP Encryption Key	Key Derivation	Crypto Officer - PSP Master Key: W,E - PSP Encryption Key: G
Show Status	Show the current status of the module	Approved mode indicator and successful completion of service; register ICM_FATAL_ECR0.kat_fail set to 0 = Approved mode and register ICM_FATAL_ECR0.kat_fail set to	N/A	Status (register ICM_FATAL_ECR0.kat_fail set to 0 or register ICM_FATAL_ECR0.kat_fail set to 1 to signify that the module is in the Approved mode and in the Hard Error state respectively)	None	Crypto Officer

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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		1 = Hard Error state				
Show module's versioning information	Show module ID and version (identifier to be correlated to name)	Approved mode indicator and successful completion of service	N/A	Hardware version: 0x0200 (decimal: 2.0); Module ID: 0x494345 (ASCII: ICE)	None	Crypto Officer
Perform Self-Tests	Perform the pre-operational and conditional self-tests	Approved mode indicator and successful completion of service; register ICM_FATAL_ECR0.kat_fail set to 0 = success and register ICM_FATAL_ECR0.kat_fail set to 1 = failure	N/A	KAT indicators	None	Crypto Officer
Perform zeroisation	Zeroise; Clear all SSPs via module reset	Approved mode indicator and successful completion of service; KAT flags not set to 0 = operational; KAT bits set to 0 = Hard Error	Reset signal	Status	None	Crypto Officer - Secure SADB Key: Z - ESP Encryption Key: Z - ESP Authentication Key: Z - PSP Encryption Key: Z - PSP Master Key: Z - ESP Decryption Key:

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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Z - PSP Decryption Key: Z - AES GCM IV: Z
Initialization	Module initialization	Approved mode indicator and successful completion of service	Reset signal	Status of self-test execution at boot, indicator	None	Crypto Officer
Diagnostic functions	Diagostic functions performed by the module	Approved mode indicator and successful completion of service	-	Diagnosis/results	None	Crypto Officer
Transmit/receive plaintext data plane packet	Plaintext transmission of packets through the module	Approved mode indicator and successful completion of service	Packets	Packets	None	Crypto Officer

Table 8: 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

The module does not support any Non-Approved Services.

4.5 External Software/Firmware Loaded

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

The module does not support loading of firmware.

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 module is implemented entirely in hardware and is non-modifiable. Therefore, the integrity test requirements do not apply in accordance with AS10.19. Per the allowance in AS10.19, instead, the module implements Known Answer Tests (KAT) i.e., Cryptographic Algorithm Self-Tests (CAST) and is designed to execute the same on every boot.

5.2 Initiate on Demand

The 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. No firmware is implemented in the module.

The module resides within the following single chip:

- IN763 SoC

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

6.2 Configuration Settings and Restrictions

The module is implemented entirely in hardware and is thus non-modifiable, i.e., configuration of the module is infeasible.

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 9: Mechanisms and Actions Required

The module is a sub-chip cryptographic subsystem 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 a standard passivation applied and hence conforms to the Security Level 1 requirements for this section.

8 Non-Invasive Security

8.1 Mitigation Techniques

The module does not implement any non-invasive security mitigations and thus the requirements per this section do not apply to the module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
Internal memory (registers, register files, FIFO buffers, SRAM)	Ephemeral storage in the module's internal memory	Dynamic

Table 10: 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
Output to external DRAM	Internal memory (registers, register files, FIFO buffers, SRAM)	SADB in external DRAM	Encrypted	Automated	Electronic	KTS-1
Input from SADB in external DRAM	SADB in external DRAM during data plane operations	Internal memory (registers, register files, FIFO buffers, SRAM)	Encrypted	Automated	Electronic	KTS-1
Input from on-chip SRAM	On-chip SRAM	Internal memory (registers, register files, FIFO buffers, SRAM)	Encrypted	Automated	Electronic	KTS-1
Output to on-chip SRAM	Internal memory (registers, register files, FIFO buffers, SRAM)	On-chip SRAM	Encrypted	Automated	Electronic	KTS-1
Input via CSR write	Within the SoC, outside the module via a direct internal CSR write	Internal memory (registers, register files, FIFO buffers, SRAM)	Plaintext	Manual	Electronic	

Table 11: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Reset	All storage points (SRAMs, RFs, Flop) are reset on core reset	Memory is cleared on reset	Operator Initiation

Table 12: SSP Zeroization Methods

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

When the operator performs a reset of the IN763 SoC, it will zeroise all SSPs contained within the module. SSPs stored within the module's memory and FIFOs are zeroised as part of the reset process. While this action will clear all secrets internal to module, it will also leave the module in a non-usable state. For normal data path operations, when a session is terminated, the SA is deleted via a command initiated by the external software and zeroes are written to the entire SA entry.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Secure SADB Key	Used to encrypt/decrypt PSP and ESP encryption keys in DRAM	256 - 256	Symmetric - CSP			KTS-1
ESP Encryption Key	Used to encrypt packet payload (Egress)	128, 256 - 128, 256	Symmetric - CSP			Egress Authenticated Block Ciper
ESP Authentication Key	Used to authenticate packet payload	128, 256 - 128, 256	Symmetric - CSP			Egress Mesage Authentication
PSP Encryption Key	Used to encrypt packet payload (Egress)	128, 256 - 128, 256	Symmetric - CSP	Key Derivation		Egress Authenticated Block Ciper
PSP Master Key	Key derivation key for PSP keys	256 - 256	Symmetric - CSP			Key Derivation
ESP Decryption Key	Used to decrypt packet payload (Ingress)	128, 256 - 128, 256	Symmetric - CSP			Ingress Authenticated Block Cipher
PSP Decryption Key	Used to decrypt packet	128, 256 -	Symmetric - CSP	Key Derivation		Ingress Authenticate

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

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	payload (Ingress)	128, 256				d Block Cipher
AES GCM IV	AES GCM IV	96 bits - N/A	Initialization Vector - PSP	AES GCM IV Construction		Ingress Authenticated Block Cipher Egress Authenticated Block Ciper KTS-1

Table 13: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Secure SADB Key	Input via CSR write	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	ESP Encryption Key:Encrypts PSP Encryption Key:Encrypts ESP Encryption Key:Decrypts PSP Encryption Key:Decrypts
ESP Encryption Key	Output to external DRAM Input from SADB in external DRAM Input from on-chip SRAM Output to on-chip SRAM	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	
ESP Authentication Key	Output to external DRAM Input	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	

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

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	from SADB in external DRAM Input from on-chip SRAM Output to on-chip SRAM				
PSP Encryption Key	Output to external DRAM Input from SADB in external DRAM	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	PSP Master Key:Derived From
PSP Master Key	Input via CSR write	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	PSP Encryption Key:Used to derive PSP Decryption Key:Used to derive
ESP Decryption Key	Output to external DRAM Input from SADB in external DRAM Input from on-chip SRAM Output to on-chip SRAM	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	
PSP Decryption Key	Output to external DRAM Input from	Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	PSP Master Key:Derived From

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

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
	SADB in external DRAM Input from on-chip SRAM Output to on-chip SRAM				
AES GCM IV		Internal memory (registers, register files, FIFO buffers, SRAM):Plaintext	Until ICE reset	Reset	Secure SADB Key:Used With ESP Encryption Key:Used With PSP Encryption Key:Used With ESP Decryption Key:Used With PSP Decryption Key:Used With

Table 14: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
AES-GCM (CAVP Certs. #A6038, #A6039 and A6040) and KDF SP800-108 (CAVP Cert. A6038) CASTs	256-bits	CAST	Hardware-CAST	register ICM_FATAL_ECR0.kat_fail set to 0	AES-GCM and KDF SP800-108 CASTs performed in lieu of the pre-operational firmware integrity test in accordance

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

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
					with AS10.19

Table 15: Pre-Operational Self-Tests

The module is implemented entirely in hardware and is non-modifiable. Therefore, the pre-operational firmware integrity test requirements do not apply. In accordance with AS10.19, the module instead is designed to execute the Cryptographic Algorithm Self-Tests (CAST) on every boot. These tests do not require operator intervention to run. These self-tests must be passed before the Crypto Officer can perform services.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A6038)	256-bit	KAT	CAST	register ICM_FATAL_ECR0.kat_fail set to 0	Decrypt	On module boot
KDF SP800-108 (A6038)	CMAC 256-bit, Counter mode	KAT	CAST	register ICM_FATAL_ECR0.kat_fail set to 0	Generate/Verify	On module boot
AES-GCM (A6039)	256-bit	KAT	CAST	register ICM_FATAL_ECR0.kat_fail set to 0	Encrypt	On module boot
AES-GCM (A6040) - Encrypt	256-bit	KAT	CAST	register ICM_FATAL_ECR0.kat_fail set to 0	Encrypt	On module boot
AES-GCM (A6040) - Decrypt	256-bit	KAT	CAST	register ICM_FATAL_ECR0.kat_fail set to 0	Decrypt	On module boot

Table 16: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (CAVP Certs. #A6038, #A6039 and A6040)	CAST	Hardware-CAST	On every boot, On demand	Programmatically, Manually via reboot

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

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
and KDF SP800-108 (CAVP Cert. A6038) CASTs				

Table 17: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (A6038)	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
KDF SP800-108 (A6038)	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-GCM (A6039)	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-GCM (A6040) - Encrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot
AES-GCM (A6040) - Decrypt	KAT	CAST	On every boot, On demand	Programmatically, Manually via reboot

Table 18: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Hard Error	If any of the conditional self-tests fail, the module will return an error code and transition to an error state where no functions can be executed	Conditional Cryptographic Algorithm Self-Test (CAST) failure (covers the Pre-Operational Firmware Integrity Test failure)	An operator can attempt to reset the state by cycling the power; However, the failure of a self-test may require the module to be replaced	register ICM_FATAL_ECR0.kat_fail set to 1

Table 19: Error States

10.5 Operator Initiation of Self-Tests

The operator can initiate the Cryptographic Algorithm Self-Tests (CAST) by rebooting the module.

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

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

No configuration of the module or installation steps are required by the operator. When the module is powered on its Known Answer Tests i.e., Cryptographic Algorithm Self-Tests (CAST) are executed without any operator intervention. The module enters the Approved mode of operation automatically if the self-tests are completed successfully. If any of the self-tests fail during power-up, the module will transition to the Hard Error state. The status of the module can be determined by the availability of the module. If the module is available, it has passed all self-tests. If it is unavailable, it is in the error state or has not been properly initialized.

11.2 Administrator Guidance

Additional guidance does not apply for the operation of the module apart from that specified in Sections 2, 3 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, 3 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. In other words, the module can be zeroised to securely sanitize it.

12 Mitigation of Other Attacks

The module does not mitigate any other attacks and thus the requirements per this section do not apply.

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