CRATON2/SECTON embedded V2X HSM –

FIPS 140-2 Cryptographic Module

Non-Proprietary Security Policy

Revision 1.8

Revision History

Rev.	Date	Description
1.0	November 2017	First release
1.1	January 2018	Second release
1.2	July 2018	Third release
1.3	February 2019	Fourth release
1.4	July 2019	Fifth release
1.5	September 2019	Tables 4, 5 and 11 updated in response to CMVP comments; Table 16 relocated
1.6	October 2021	Updated to Cut3.0 additions
1.7	October 2022	Updated Sections 1.3 & 3.4; updated Tables 4, 5, 10 & 13
1.8	November 2022	Corrected KTS algorithm certificate # note in Table 4

Table of Contents

List of Figures

List of Tables

1 Introduction

The Security Policy for the Autotalks CRATON2/SECTON embedded V2X HSM (herein designated as "the Module") is described in this document. The Module provides a cost-effective and performance-optimized solution for signature generation and secure data storage in V2X systems. The V2X system is required to support vehicle communication with other vehicles, as well as with road infrastructure and other elements, potentially processing thousands of such messages per second. The Module will support highly secure and timely necessary cryptographic operations, so that all messages may be securely signed and authenticated.

The Module is implemented with the configurations listed in the table below:

Table 1: Cryptographic Module Configurations

Module	HW P/N and Version	FW Version	OE
CRATON2/SECTON embedded V2X HSM	ATK66610, Version 2.1.2	3.0	SECTON v3.0 and CRATON2 v3.0

The Module is intended for use in markets, including US Federal agencies, which require FIPS 140-2 validated cryptographic modules. The Module is embedded within a single-chip embodiment. The logical cryptographic boundary is defined as a sub-chip cryptographic subsystem per IG1.20, consisting of only the embedded HSM, and the physical boundary is the larger chip that contains it (see Figure 3 and Figure 4 below).

The eHSM is designated as a non-modifiable environment as per FIPS 140-2 definitions. The FIPS 140-2 security levels for the Module are as follows:

Table 2: Security Level of Security Requirements

Security Requirement	Security Level
Cryptographic Module Specification	3
Cryptographic Module Ports and Interfaces	3
Roles, Services, and Authentication	3
Finite State Model	3
Physical Security	3
Operational Environment	N/A
Cryptographic Key Management	3
EMI/EMC	3
Self-Tests	3
Design Assurance	3
Mitigation of Other Attacks	N/A
Overall FIPS 140-2 Security Level	**3**

The Module implementation is compliant with the following standards:

- FIPS 140-2, FIPS 140-2 IG, FIPS 140-2 DTR
- FIPS 180-4
- FIPS 186-4
- FIPS 197
- FIPS 198-1
- SP 800-38A, SP 800-38B, SP 800-38C
- SP 800-90A
- SP 800-38F
- SP 800-133

1.1 Hardware and Physical Cryptographic Boundary

The physical form of chip in which the Module is embedded, is depicted in Figure 1. The eHSM shall be validated as a sub-chip cryptographic subsystem, within the CRATON2/SECTON V2X system. Components on the chip outside the eHSM are supported by cryptographic services which are implemented internally to eHSM.

Figure 1: CRATON2 Chip Containing eHSM Module

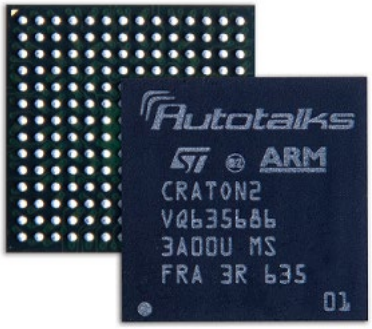
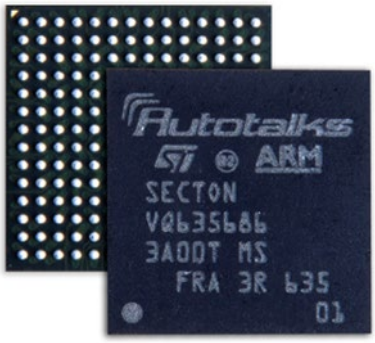


Figure 2: SECTON Chip Containing eHSM Module



The eHSM module is connected to the IC using the following ports and interfaces:

Table 3: Ports and Interfaces

Port	Description	Logical Interface Type
AHBLitemaster	Hardware bus connection.	Control in, Data in, Data out, Status out
JTAG	Permanently disabled via metal disconnection	N/A
Mailbox	Hardware mailbox used to send/ receive interrupts.	Control in, Status out
Anti-Tamper	1-bit Anti-Tamper fuse	Control in
Power	Power input on CRATON2/SECTON boundary	Power in

As indicated by the above table, all status output ports and control and data ports are directed through the interface of the module's logical boundary.

After the Module finishes initialization and all self-tests are completed successfully, all of the cryptographic functionality will be made available for use. If any of the module's Known Answer Tests (KATs) fail, the module will enter an error state and output an error indicator via the status output interface.

1.2 Firmware and Logical Cryptographic Boundary

Figure 3 and Figure 4 below depict the single-chip physical boundary (solid blue line) and sub-chip logical boundary (dotted red line).

Figure 3: Module Block Diagram – CRATON2

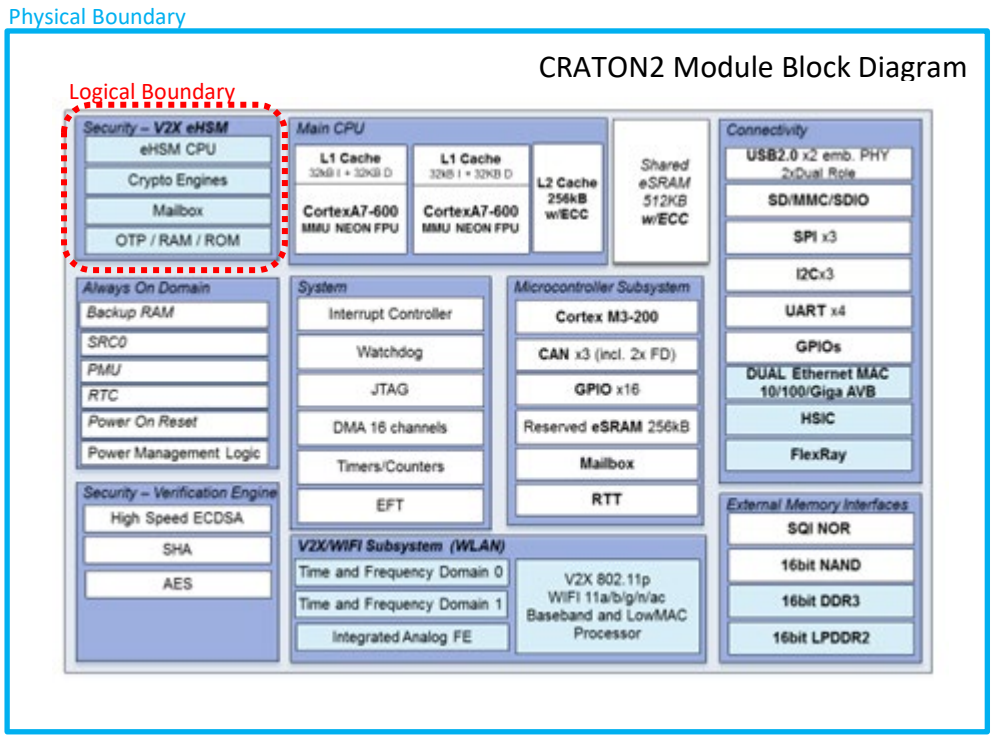
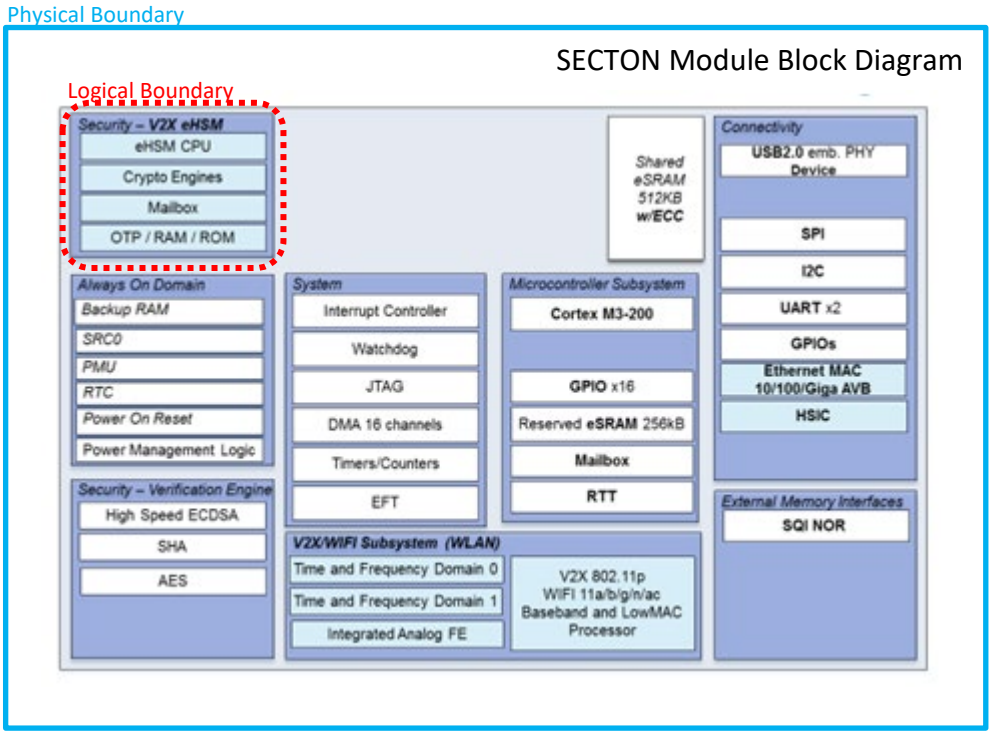


Figure 4: Module Block Diagram – SECTON



The block in the Figures above notated “Security – Verification Engine” is a disjoint sub-chip cryptographic subsystem not included as part of this validation.

All other services and functionality outside of the sub-chip boundary defined above are excluded from this validation.

The Module includes the following hardware components:

1. ARM Cortex-M0 CPU
2. 32KB RAM
3. 128KB ROM
4. Cryptographic accelerator
5. One-time programmable (OTP) memory

The Module can access the following external memories only for data input or output:

1. SRAM (inside the chip boundary, but outside the eHSM).
2. DRAM (outside the chip boundary).

Firmware components are stored permanently in internal ROM, and may not be updated. The eHSM firmware runs exclusively from eHSM internal ROM. Internal ROM and RAM are not accessible (hardware protected) from components external to the eHSM.

1.3 Modes of Operation

The module supports multiple standards and cryptographic APIs, some of which are FIPS Approved, some are non-Approved but allowed, and others that are non-Approved services. The cryptographic APIs clearly identify which algorithm is being called.

All callable services will indicate the mode of operation in bits 24:25 of the return code. If these bits are 00, the API ran in an Approved mode and used only Approved cryptographic algorithms. If these bits are 01, then the API ran in an Approved mode, but used Brainpool curves. Any other value indicates the API used a non-Approved algorithm, and therefore ran in the non-Approved mode.

The module’s firmware version can be checked by calling the Read HSM info service.

The module does not share keys/CSPs between the Approved and non-Approved modes.

The services which are available in each mode are specified in the various tables in Section 3.4.

2 Cryptographic Functionality

The Module implements the FIPS Approved and Non-Approved but Allowed cryptographic functions as summarized in the tables below.

Table 4: Approved and CAVP Validated Cryptographic Functions

Algorithm	Description	Cert #
AES	[FIPS 197, SP 800-38A] Functions: Encryption, Decryption Modes: ECB, CBC Key sizes: 128, 192, 256 bits	A837
	[SP 800-38C] Functions: Authenticated Encrypt, Authenticated Decrypt Mode: CCM Key sizes: 128, 192, 256 bits	
	[SP 800-38B] Functions: Generation, Verification Mode: CMAC Key sizes: 128, 192, 256 bits	
AES	Used in TRNG [FIPS 197, SP 800-38A] Functions: Encryption, Decryption Modes: ECB, CBC Key size: 128 bits *Note: decryption was tested, but is not utilized	5797
	[SP 800-38B] Function: Generation Mode: CMAC Key size: 128 bits *Note: this implementation is only utilized for SP 800-90B entropy conditioning and during power up self-tests	
CKG	[SP 800-133] - Section 5.1: Asymmetric signature key generation using unmodified DRBG output. - Section 6.1: Direct symmetric key generation using unmodified DRBG output	Vendor Affirmed
CVL	[FIPS 186-4] Functions: Signature Generation Curves/Key sizes: P-224 w/ SHA-224, P-256 w/ SHA-256, P-384 w/ SHA-384	2091
DRBG	[SP 800-90A] Functions: HMAC SHA-256, with or without prediction resistance (as per NIST SP.800-90Ar1) Security Strengths: 256 bits	2390

Algorithm	Description	Cert #
ECDSA	[FIPS 186-4] Functions: Key Pair Generation, Signature Generation and Verification Curves/Key sizes: P-224 w/ SHA-224, P-256 w/ SHA-256, P-384 w/ SHA-384 Per IG A.2, the following Brainpool 256-bit (providing security strength of 128 bits) and 384-bit (providing security strength of 192 bits) elliptic curves are supported: P256t1 w/ SHA-256, P384t1 w/ SHA-384, P256r1 w/ SHA-256 and P384r1 w/ SHA-384.	A838
HMAC	[FIPS 198-1] Functions: Generation SHA sizes: SHA-224, SHA-256, SHA-384, SHA-512	3832
KTS	[SP 800-38F] Functions: Authenticated Encryption, Authenticated Decryption Mode: CCM Key sizes: 128, 256 bits	A837
	[SP 800-38F] Functions: Authenticated Encryption, Authenticated Decryption Modes: CBC & CMAC Key sizes: 128, 256 bits *Note: Cert. #A837 applies both to AES-CBC encryption and CMAC authentication	
SHA	[FIPS 180-4] Functions: Digital Signature Generation SHA sizes: SHA-224, SHA-256, SHA-384, SHA-512 *Note: SHA-512 has been tested but is not utilized.	4607

Table 5: Non-Approved but Allowed Cryptographic Functions

Algorithm	Description
AES-CMAC Conditioning	SP 800-90B vetted CMAC (AES Cert. #5797) conditioning component.
NDRNG	TRNG Based on Tracking Jitter of a Phase-Locked Loop. Provides 99.9% bits of min entropy per bit sampled for a security strength of 256-bits.
Proprietary Post-Processing (no security claimed)	Proprietary algorithm used to perform IG 7.8 post-processing; allowed per IG 1.23.

The module implements the non-Approved cryptographic functions as shown in Table 6.

Table 6: Non-Approved Cryptographic Functions

Algorithm	Description
ECIES	Elliptic Curve Integrated Encryption Scheme
SM2	[GM/T 0003-2012, Part1] Key pair generation
	[GM/T 0003-2012, Part2] Functions: Digital signature algorithm Curves/Key size: P-256 w/ SM3 SHA-256
	[GM/T 0003-2012, Part3] Key exchange protocol
	[GM/T 0003-2012, Part4] Public key encryption algorithm
SM3	[GM/T 0004-2012] SHA size: SHA-256
SM4	[GM/T 0002-2012] Functions: Encryption, Decryption Modes: ECB, CBC Key size: 128 bits
	[GM/T 0002-2012, SP 800-38C] Functions: Authenticated Encrypt, Authenticated Decrypt Mode: CCM Key size: 128 bits
	[GM/T 0002-2012, SP 800-38B] Functions: Generation, Verification Mode: CMAC Key size: 128 bits

2.1 Critical Security Parameters

All CSPs used by the Module are summarized in the table below. Usage of these CSPs by the Module (including all CSP lifecycle states) is described in the services detailed in Section 3.

The “blob” is a data structure used to store CSPs in encrypted and authenticated form. The data will be encrypted in the blob for secure storage outside the HSM boundary. The data within the blob, as well as keys used to secure the blob, will be accessible in plaintext only within the HSM boundary.

Table 7: Critical Security Parameters (CSPs)

CSP	Description / Usage
Chip-Specific Key (CSK)	256-bits random value used to derive the AES-256 bits KEK (Key Encryption Key) via IG 7.8 post-processing
Key Encryption Key (KEK)	AES-256 key used to encrypt the BEK (Blob Encryption Key)
Blob Encryption Key (BEK)	AES-128 or 256 bit key used to encrypt Blobs
ECC Private Key	NIST P224, P-256, P-384, Brainpool (P256t1, P384t1, P256r1 or P384r1) key used to sign V2X messages. Each key is bound to a specific algorithm/ service.
AES Key	AES-128, AES-192 or AES-256 key used to encrypt/ decrypt/ authenticate application data
HMAC Key	Key used for HMAC. Key size range [digest_size/2:128] bytes
Master Authentication Key	AES-128 or AES-256 bit key used to authenticate Application Session Keys
Master Encryption Key	AES-128 or AES-256 bit key used to encrypt Application Session Keys
Application Authentication Session Key	AES-128 or AES-256 bit key used to authenticate Application sessions
Application Encryption Session Key	AES-128 or AES-256 bit key used to encypt Application sessions
Importing Key	AES-128 or AES-256 bit key used to import encrypted keys
DRBG State	Internal to the module (V and key). Necessary entropy is introduced at fixed intervals. Size of state data is 512 bits.
DRBG Seed	Provided by the TRNG; provides at least 256 bits of entropy
TRNG AES-CMAC key	AES-128 bit key used internally by the TRNG

2.2 Public Keys

Table 8: Public Keys

Key	Description / Usage
Root Public Key	NIST ECDSA (P-256,or P-384) key.
ECC Public key	NIST P-224, P-256, P-384 or BrainPool (P256t1, P384t1, P256r1 or P384r1) key used to verify V2X signatures or for ECIES.

3 Roles, Authentication and Services

3.1 Assumption of Roles

The Module supports two operational roles, User and Cryptographic Officer. Services supported by each role are secured by identity-based authentication.

The Module does not support a maintenance role. The JTAG debug interface is disabled by underlying hardware.

The eHSM Module also does not support any bypass capability, nor does it support concurrent operators.

Separation of roles is enforced by authentication method (see Section 3.2).

Table 9: Roles Description

Role ID	Role Description	Authentication Type	Authentication Data
CO	Chip bootloader	Identity-based	Signature
User	Any authenticated software applications loaded on the chip, other than the bootloader	Identity-based	Signature

3.2 Authentication Methods

The User and CO roles are authenticated by verifying an ECDSA signature over the application firmware image inside the module using a public key stored in the module's OTP. The supported elliptic curves are NIST P-256 w/ SHA-256 and P-384 w/ SHA-384.

During manufacturing, the OEM will generate an EC key pair to be used for ECDSA. The public key will be permanently burned in the module's OTP and the private key will be used to sign the application firmware image.

256-bit ECDSA has an authentication strength of 128 bits. The probability that a random signature verification attempt will succeed or a false acceptance will occur, is 1/ 2 ^ 128, which is significantly less than the required 1/1,000,000.

A very conservative estimate of the maximum authentication rate is 100 microseconds per failed authentication, which would allow 6 billion consecutive attempts per minute. Under this assumption the probability that random authentication attempts will succeed within a one-minute interval is 6,000,000,000 * 1/2 ^ 128, which is significantly less than the required 1/100,000.

3.3 Authentication Data Protection

The root ECC public key is stored in the read-only OTP memory of the module and used as the means to verify the application firmware image. Since this memory is non-volatile read-only memory, it cannot be modified. The public key is used to verify the OEM's signature of the signed bootloader or software application images. Only the images that are signed by the OEM can be authenticated to the module. Any image with an altered signature won't be authenticated, and hence won't be loaded and get to use the module.

3.4 Services

All services implemented by the Module are listed in the tables below. Non-Approved mode relates only to use of non-Approved algorithms.

Table 10: Pre-Deployment Services

Service Class	Service(s)	Description
Lifecycle	Generate and program CSK	CSK is generated and programmed internal to HSM. One-time only (enforced by hardware).
	Disable importing key creation	Set the importing key flag in OTP. After importing key flag is set in OTP, the “Generate importing key” service will no longer be supported.
	Disable master pairing key creation	Set the master key flag in OTP. After master key flag is set in OTP, the “Generate master key pair” service will no longer be supported.
	Disable plaintext AES key import	Once called, the import of a plaintext AES key will be permanently disabled (OTP flag).
	Disable plaintext HMAC key import	Once called, the import of a plaintext HMAC key will be permanently disabled (OTP flag).
	Disable plaintext ECC key pair import	Once called, the import of a plaintext ECC key pair will be permanently disabled (OTP flag).
	Disable raw random data generation	Once called, the Generate raw random data will be permanently disabled (OTP flag).
Pairing	Create master key pair	Master key pair (Master Authentication Key and Master Encryption Key) is generated externally and imported into the module and output encrypted.
AES	Import plaintext AES key	The AES key will be imported.
HMAC	Import plaintext HMAC key	Import plaintext HMAC key and export in blob.
V2X	Import plaintext ECC key pair	Import plaintext ECC key pair (ECC private key is defined as a CSP).
Importing	Create importing key	Importing key is generated externally and imported into the module and output encrypted.
RNG	Generate raw random data	Random data is output from the TRNG engine.

All of the services listed above are called in a production environment and thereafter permanently disabled.

Table 11: Approved Cryptographic Officer Services

Service Class	Service(s)	Role	Description
Management	Initialize system	CO	Initialize system: initialize crypto accelerators and OTP memory, configure interrupts,OTP access permissions and allowed memory regions, DRBG instantiation, and initialize security-related data. Note: calling the initialization service triggers the power on self test and the service will only succeed if all power on self-tests are successful; this sequence is considered to be part of the IG 9.5 initialization period. Only non-security relevant status information can be output from the module during the initialization period.
	Disable management	CO	Once the "Disable management" API is called, the HSM will enable a relevant internal flag; the HSM will use this flag to ensure that no application may subsequently invoke any CO service, until chip reset (power cycle).
Tamper	Tamper response standby	CO	Enable normal tamper response, including CSP zeroization. Cannot subsequently be disabled.
	Tamper response normal	CO	Enable standby tamper response, including CSP zeroization. Cannot subsequently be disabled.

CO services initialize the module and relevant keys. These services may be invoked only by the bootloader.

After system and key setup, and on completion of bootloader functionality, a relevant internal flag will be set, after which CO service requests will no longer be accepted by the HSM from any application (until reset).

Table 12: Approved Services

Service Class	Service	Role	Description
Diagnostic	Read HSM info	CO User	Output informative data: HSM firmware version, lifecycle/runtime status (not security related).
OTP	Manage application OTP data	CO User	Read, write, lock, and check OTP. Related to application area (not security related)
DRBG	Generate random data	CO User	Random data is output from a FIPS 140-2 compliant DRBG.
Storage (AES based)	Generate MAC	CO User	Generate MAC with BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
	Verify MAC	CO User	Verify MAC with BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
	Authenticate and encrypt	CO User	Authenticate and encrypt with BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).

Service Class	Service	Role	Description
	Authenticate and decrypt	CO User	Authenticate and decrypt with BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
V2X enrollment	Generate ECC key pair	CO User	Generate ECC key pair internally to module with NIST curves.
	Import encrypted ECC key pair	CO User	Import encrypted ECC key pair (ECC private key is defined as a CSP) with NIST curves.
V2X	ECDSA sign	CO User	Generate ECDSA signature on data and/or hash with NIST curves.
	ECDSA verify	CO User	Verify ECDSA data and/or hash with NIST curves.
	Get ECC public key	CO User	Derive public key from private key stored inside the blob with NIST curves.
Pairing (AES based)	Create application session key	CO User	Application session (authentication, encryption) key is generate internal to HSM, and encrypted for secure storage external to HSM.
	Import master key pair	CO User	Master key pair is protected by authenticated encrypted blob for secure storage external to HSM.
	Generate MAC with session key	CO User	Generate MAC with application session key.
	Verify MAC with session key	CO User	Verify MAC with application session key.
	Authenticate and encrypt with session key	CO User	Authenticate and encrypt with application session key.
	Authenticate and decrypt with session Key	CO User	Authenticate and decrypt with application session key.
AES	Create AES key	CO User	Generate AES key and export in blob.
	Import encrypted AES key	CO User	Import encrypted AES key and export in blob.
	Encrypt with AES-ECB	CO User	AES-ECB encryption using an encapsulated AES blob.
	Decrypt with AES-ECB	CO User	AES-ECB decryption using an encapsulated AES blob.
	Encrypt with AES-CBC	CO User	AES-CBC encryption using an encapsulated AES blob.
	Decrypt with AES-CBC	CO User	AES-CBC decryption using an encapsulated AES blob.
	Authenticate and encrypt with AES-CCM	CO User	AES-CCM authenticated encryption using an encapsulated AES blob.
	Authenticate and decrypt with AES-CCM	CO User	AES-CCM authenticated decryption using an encapsulated AES blob.

Service Class	Service	Role	Description
	Generate MAC with AES-CMAC	CO User	AES-CMAC generation using an encapsulated AES blob.
	Verify MAC with AES-CMAC	CO User	AES-CMAC verification using an encapsulated AES blob.
HMAC	Create HMAC key	CO User	Generate HMAC key and export in blob.
	Import encrypted HMAC key	CO User	Import encrypted HMAC key and export in blob.
	Generate HMAC	CO User	MAC generation using an encapsulated HMAC blob.
	Verify HMAC	CO User	MAC verification using an encapsulated HMAC blob.
Hash	Compute hash	CO User	Compute SHA2 hash
Blob I/O	Encrypt BEK	CO User	Encrypt a Blob Encryption Key (BEK).
	Decrypt BEK	CO User	Decrypt a Blob Encryption Key (BEK).
Self-test	System and cryptographic self-tests	CO User	Perform ROM integrity test; as well as AES, CCM, CMAC, SHA, HMAC, ECDSA and DRBG tests.
ECC	ECDSA authentication	CO User	ECDSA Authentication using a public key stored in OTP.
Importing	Insert importing key	CO User	Importing key is protected by authenticated encrypted blob for secure storage external to HSM.

Software applications executing on the device assume the User role when requesting any services provided by the module. The user role has access to all of the module's services, except those listed only as CO in the tables.

User authentication is via image authentication (Secure Boot). If the signature is verified, then the image is authenticated and hence can be loaded and executed.

Table 13: Services related to Brainpool Curves

Service Class	Service	Role	Description
V2X enrollment	Generate ECC key pair	CO User	Generate ECC key pair internally to module with brainpool curves.
	Import encrypted ECC key pair	CO User	Import encrypted ECC key pair (ECC private key is defined as a CSP) for brainpool curves.
	Generate one time ECC key pair	CO User	Generate one time ECC key pair used for implicit certificate generation
	Generate contribution data	CO User	Generate contribution data used for implicit certificate generation
V2X	ECDSA sign	CO User	Generate ECDSA signature on data and/or hash with brainpool curves.

Service Class	Service	Role	Description
	ECDSA verify	CO User	Verify ECDSA data and/or hash with brainpool curves.
	Get ECC public key	CO User	Derive public key from private key stored inside the blob with brainpool curves.
	Multiply-add ECC private key	CO User	Multiply-add ECC private key

Table 14: Non-Approved Services

Service Class	Service	Description
DRBG (SM3 based)	Generate random data	Random data is output from a SM3-HMAC based DRBG.
Storage (SM4 based)	Generate MAC	Generate MAC with SM4 BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
	Verify MAC	Verify MAC with SM4 BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
	Authenticate and encrypt	Authenticate and encrypt with SM4 BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
	Authenticate and decrypt	Authenticate and decrypt with SM4 BEK in order to support secure storage. CSPs are secured by eHSM (accessed in clear only within eHSM).
V2X	Generate ECIES decryption key (non-compliant)	Generate ECIES decryption key
	Encrypt with ECIES (non-compliant)	ECIES encryption
	Decrypt with ECIES (non-compliant)	ECIES decryption
SM2 V2X enrollment	Generate SM2 ECC key pair	Generate SM2 ECC key pair internally to module.
	Import encrypted SM2 ECC key pair	Import encrypted SM2 ECC key pair (ECC private key is defined as a CSP).
	Generate one time SM2 ECC key pair	Generate one time SM2 ECC key pair used for implicit certificate generation
	Generate contribution data	Generate contribution data used for implicit certificate generation with SM2 curves.
SM2 V2X	SM2 ECDSA sign data	Generate ECDSA signature on data and/or hash with SM2 curves.
	Get ECC public key	Derive public key from private key stored inside the blob with SM2 curves.
	Multiply-add ECC private key	Multiply-add ECC private key with SM2 curves
	SM2 Key Exchange	SM2 key exchange protocol for generating a shared secret key between two users
Pairing (SM4 based)	Create application session key	SM4 application session (authentication, encryption) key is generated internal to HSM,

		and encrypted for secure storage external to HSM.
	Import master key pair	SM4 master key pair is protected by authenticated encrypted blob for secure storage external to HSM.
	Generate MAC with session key	Generate MAC with SM4 application session key.
	Verify MAC with session key	Verify MAC with SM4 application session key.
	Authenticate and encrypt with session key	Authenticate and encrypt with SM4 application session key.
	Authenticate and decrypt with session Key	Authenticate and decrypt with SM4 application session key.
SM4	Create SM4 key	Generate SM4 key and export in blob.
	Import encrypted SM4 key	Import encrypted SM4 key and export in blob.
	Encrypt with SM4-ECB	SM4-ECB encryption using an encapsulated SM4 blob.
	Decrypt with SM4-ECB	SM4-ECB decryption using an encapsulated SM4 blob.
	Encrypt with SM4-CBC	SM4-CBC encryption using an encapsulated SM4 blob.
	Decrypt with SM4-CBC	SM4-CBC decryption using an encapsulated SM4 blob.
	Authenticate and encrypt with SM4-CCM	SM4-CCM authenticated encryption using an encapsulated SM4 blob.
	Authenticate and decrypt with SM4-CCM	SM4-CCM authenticated decryption using an encapsulated SM4 blob.
	Generate MAC with SM4-CMAC	SM4-CMAC generation using an encapsulated SM4 blob.
	Verify MAC with SM4-CMAC	SM4-CMAC verification using an encapsulated SM4 blob.
SM3-HMAC	Create HMAC key	Generate HMAC key and export in blob.
	Import encrypted HMAC key	Import encrypted HMAC key and export in blob.
	Generate HMAC	MAC generation using an encapsulated HMAC blob.
	Verify HMAC	MAC verification using an encapsulated HMAC blob.
SM3	Compute hash	Compute SM3 hash
Blob I/O	Encrypt SM4 BEK	Encrypt a Blob Encryption Key (BEK).
Blob I/O	Decrypt SM4 BEK	Decrypt a Blob Encryption Key (BEK).
Importing	Insert SM4 importing key	Importing key is protected by authenticated encrypted blob for secure storage external to HSM.

Notes:

- The approved and non-approved services described in the tables 12 and 14 are completely isolated and independent from each other. CSPs cannot be shared between two services variants.
- Invocation of the non-aproved services returns non-approved indication.

Table 15: Unauthenticated Services

Service Class	Service	Description
Tamper	Zeroize	Zeroize all keys/ CSPs stored in the Module by triggering a tamper response. The CO must call the Tamper response standby or Tamper response normal service before calling this service.

Table 16 defines the relationship between access to Security Parameters and the different module services. The modes of access shown in the table are defined as:

- G = Generate: The service generates the key/ CSP.
- O = Plaintext Output: The service outputs the key/ CSP.
- E = Execute: The service uses the key/ CSP in an algorithm.
- I = Plaintext Input: The service inputs the key/CSP.
- Z = Zeroize: The service zeroizes the key/ CSP.
- i = Encrypted input: the service inputs the key/ CSP in encrypted form.
- o = Encrypted output: the service outputs the key/ CSP in encrypted form.

Table 16: Security Parameters Access by Service

Service Name	Service Class	CSK**	KEK	BEK	Importing key	ECC Private Key	ECC Public key	Root Public key	Master Encryption Key	Master Authentication Key	App Enc Session Key	App Auth Session Key	DRBG State	DRBG Seed	TRNG AES-CMAC Key	AES key	HMAC key
Generate and program CSK	Lifecycle	GE	G														
Disable importing key creation	Lifecycle																
Disable master pairing key creation	Lifecycle																
Disable plaintext AES key import	Lifecycle																
Disable plaintext HMAC key import	Lifecycle																
Disable plaintext ECC key pair import	Lifecycle																
Disable raw random data generation	Lifecycle																
Create master key pair	Pairing		E	GEo					Io	Io			E	GE	E		
Import plaintext AES key	AES		E	GE									E	GE	E	Io	
Import plaintext HMAC key	HMAC		E	GE									E	GE	E		Io
Import plaintext ECC key pair	V2X		E	GE		Io	O						E	GE	E		
Create importing key	Importing			GEo	Io								E	GE	E		
Generate raw random data	RNG														E		
Initialize system	Management												G	GE	GE		

Disable management	Management															
Tamper response standby	Tamper *															
Tamper response normal	Tamper *															
Read HSM info	Diagnostic															
Manage application OTP data	OTP															
Generate random data	DRBG											E	GE	E		
Generate MAC	Storage		E	GEo								E	GE	E		
Verify MAC	Storage		E	iE												
Authenticate and encrypt	Storage		E	GEo								E	GE	E		
Authenticate and decrypt	Storage		E	iE												
Generate ECC key pair	V2X enrollment		E	GEo		Go	GO					E	GE	E		
Import encrypted ECC key pair	V2X enrollment		E	GEo	E	io	GO					E	GE	E		
ECDSA sign	V2X		E	iE		iE						E	GE	E		
ECDSA verify	V2X						IE									
Get ECC public key	V2X		E	iE		iE	GO									
Create application session key	Pairing								E	E	Go	Go	E	GE	E	
Import master key pair	Pairing		E	iE					i	i						
Generate MAC with session key	Pairing											E				
Verify MAC with session key	Pairing											E				
Authenticate and encrypt with session key	Pairing										E	E				
Authenticate and decrypt with session key	Pairing										E	E				
Create AES key	AES		E	GEo									E	GE	E	Go
Import encrypted AES key	AES		E	GEo	E								E	GE	E	io
Encrypt with AES-ECB	AES		E	iE												iE
Decrypt with AES-ECB	AES		E	iE												iE
Encrypt with AES-CBC	AES		E	iE												iE
Decrypt with AES-CBC	AES		E	iE												iE
Authenticate and encrypt with AES-CCM	AES		E	iE												iE
Authenticate and decrypt with AES-CCM	AES		E	iE												iE
Generate MAC with AES-CMAC	AES		E	iE												iE
Verify MAC with AES-CMAC	AES		E	iE												iE
Create HMAC key	HMAC		E	GEo									E	GE	E		Go
Import encrypted HMAC key	HMAC		E	GEo	E								E	GE	E		io
Generate HMAC	HMAC		E	iE									E	GE	E		iE
Verify HMAC	HMAC		E	iE									E	GE	E		iE
Compute hash	SHA																
Encrypt BEK	Blob I/O		E	E													
Decrypt BEK	Blob I/O		E	E													
System and cryptographic self-tests	Self-test															
ECDSA authentication	ECC API							E								
Insert importing key	Importing		E	iE	i											
Generate one time ECC key pair	V2X enrollment					G	O					E	GE	E		
Generate contribution data	V2X enrollment		E	iE		iE										
Multiply-add ECC private key	V2X		E	iE		io	GO					E	GE	E		
Zeroize	Tamper *	Z	Z	Z	Z	Z			Z	Z	Z	Z	Z	Z	Z	Z	Z

* When Normal and Standby tamper mode are enabled, all CSPs in the eHSM will be zeroized upon trigger of tamper signal.

** Note that the CSK is not utilized by a service; it is only used during powerup to derive the KEK via IG 7.8 post-processing.

4 Self-Tests

Self-tests are provided to test correct operation of all cryptographic algorithms each time the module is powered up. In addition, self-tests are available on demand by API.

On power up or reset, the Module performs the self-tests described in the table below and in the order as listed. All KATs must be completed successfully prior to any other use of cryptography by the module. If one of the power-up self-test KATs fails, the module enters error state.

Note that error status is output as part of the error code. No valuable information is included in the error code, and therefore there is no need for a secure channel for error code output.

The conditional self-tests are listed in Table 18 below.

Table 17: Power Up Self-Tests

Test Target	Description
Firmware Integrity	32-bit CRC performed over all Firmware (ROM)
AES (NDRNG)	Implicitly tested as part of the CMAC (NDRNG) KAT
AES	KATs: Encryption, Decryption Modes: ECB and CBC Key sizes: 128, 192 and 256 bits
CCM	KATs: Generation, Verification Key sizes: 128, 192 and 256 bits
CMAC (NDRNG)	KAT: Generation Key size: AES with 128
CMAC	KATs: Generation, Verification Key sizes: AES with 128, 192 and 256 bits
SHA	KATs: SHA-224, SHA-256, SHA-384 and SHA-512
HMAC	KATs: Generation SHA sizes: SHA-224, SHA-256, SHA-384, SHA-512
DRBG	KATs: HMAC SHA-256 DRBG (inclusive of the instantiate, generate and reseed functions) Security Strengths: 256 bits
ECDSA	KAT: Sign, Verify Curve:s P-256, P-384

Table 18: Conditional Self-Tests

Test Target	Description
Firmware Integrity	32-bit CRC performed over all Firmware (ROM)
TRNG	RCT and APT performed when a random value is requested from the NDRNG.
DRBG KAT	Performed DRBG conditional test for the generate function (as per section 11.3 of NIST SP 800-90A).
ECDSA	ECDSA Pairwise Consistency Test performed after every ECDSA key pair generation.

Table 19: Critical Function Tests

Test Target	Description
CSK Integrity Test	32-bit CRC performed on the CSK in OTP on each powerup
OTP Integrity Test	32-bit ECC for data that will never be changed (e.g., the CSK) or 32-bit (16 logical bits + 16 duplication bits) redundancy checks for monotonic counters, flags, etc. performed on each access to OTP.

5 Physical Security Policy

The Module is a sub-chip component intended to be installed as a component of a single chip.

Table 20: Physical Security Inspection Guidelines

Physical Security Mechanism	Recommended Frequency of Inspection/Test	Inspection/Test Guidance Details
Use plastic BGA package over the silicon die.	6 months	Check the Module packaging for evidence of tamper such as scratches, cracks or any other form of damage,

The Module is a standard, production-quality IC component, designed to meet automotive-grade grade 2 specifications for power, temperature, reliability, shock and vibration. The module uses standard passivation techniques for the entire chip.

The containing chip (i.e., CRATON2 or SECTON) has the following physical security mechanisms:

- Production-grade hard opaque tamper evident potting encapsulating material.

6 Operational Environment

The Module is designated as a non-modifiable operational environment. eHSM code may execute only from internal ROM, which is non-updateable. Internal eHSM RAM is configured on system initialization (by hardware) to be non-executable. Any attempt to execute code (of any type, from any source) in eHSM RAM, will generate an access error by hardware.

7 Security Rules and Guidance

The Module design corresponds to the Module security rules. This section documents the security rules enforced by the cryptographic module to implement the security requirements of this FIPS 140-2 Level 3 module.

1. Data output is inhibited during key generation, self-tests, zeroization, and error states.
2. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
3. There are no restrictions on which keys or CSPs are zeroized by the zeroization service.
4. The module does not support concurrent operators.
5. The module does not support a maintenance interface or role.
6. The module does not have any external input/output devices used for entry/output of data, during normal system operation (i.e., after secure production process).
7. The module does not output any intermediate key values.

8 Reference Documents

The following standards are referred to in this Security Policy.

Table 21: Reference Documents

Abbreviation	Document Name
[FIPS140-2]	Security Requirements for Cryptographic Modules, May 25, 2001
[FIPS140-2 IG]	Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program, August 28, 2020
[FIPS140-2 DTR]	Derived Test Requirements for FIPS PUB 140-2, Security Requirements for Cryptographic Modules, January 2011
[FIPS 180-4]	Secure Hash Standard (SHS), August 2015
[FIPS 186-4]	Digital Signature Standard (DSS), July 2013
[FIPS 197]	Advanced Encryption Standard (AES), November 2001
[FIPS 198-1]	The Keyed-Hash Message Authentication Code (HMAC), July 2008
[SP800-38A]	Recommendation for Block Cipher Modes of Operation: Methods and Techniques, December 2001
[SP800-38B]	Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005
[SP800-38C]	Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality, May 2004
[SP 800-38F]	Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping, Special Publication 800-38F, December 2012
[SP 800-90A]	Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015
[SP 800-133]	Recommendation for Cryptographic Key Generation, Revision 2, June 2020

9 Acronyms

Table 22: Acronyms

Term	Definition
API	Application Programming Interface
AES	Advanced Encryption Standard
BEK	Blob Encryption Key
CAVP	NIST Cryptographic Algorithm Validation Program
CSK	Chip Specific Key
CSP	Critical Security Parameter
eHSM	Embedded Hardware Security Module
FIPS	Federal Information Processing Standards
ECC	Elliptic Curve Cryptography
HSM	Hardware Security Module
KAT	Known Answer Test
KEK	Key Encryption Key
OEM	Original Equipment Manufacturer
OTP	One-Time Programmable
RAM	Random Access Memory
ROM	Read Only Memory
V2X	Vehicle-to-Everything

END OF DOCUMENT

Contact Information

http://www.auto-talks.com/

info@auto-talks.com

Headquarters

Grand Netter Building

P.O. Box 3846

Kfar Netter, Israel 40593

Phone: (+972) 9-886-5300

Fax: (+972) 9-886-5301