SLS37CSAEU V2X HSM

Security Target

About this document

Scope and purpose

This document is the Security Target for the Common Criteria EAL4+ Certification of the Infineon V2X HSM

Intended audience

Customers of the V2X HSM, Common Criteria Evaluation Labs, Common Criteria Certification Bodies

Table of contents

Table of contents

Table of contents

1 Introduction

1.1 ST reference

Title: SLS37CSAEU V2X HSM Security Target

Version: rev. 1.3

Publication date: 2023-11-13

Sponsor: Infineon Technologies AG, 81726 Munich, Germany

Editor: Infineon Technologies AG, 81726 Munich, Germany

1.2 TOE reference

TOE Name: SLS37CSAEU 01.03.4091

CC certificate number: NSCIB-CC-0238862

1.2.1 Underlying hardware platform

CC certificate number: NSCIB-CC-2200060-01-MA1

CC Identifier: IFX_CCI_00003Dh

Security Target: [ST_ICC]

The underlying hardware is uniquely defined by the CC indentifier. The flashloader is permanently disabled. The TOE uses the optional libraries as described in ch. 1.4.1.

1.2.2 TOE identification

The TOE can be identified by reading out the firmware versions of the V2X and VFUL application with the GET_INFO command.

GET_INFO with SELECT parameter 0x00 returns the version of the V2X application.

GET_INFO with SELECT parameter 0x01 returns the version of the VFUL application.

GET_INFO with SELECT parameter 0x08 returns the TOE name . The TOE name shall be “Infineon SLS37 V2X HSM”

The version information consists in each case of 8 bytes

- 2 bytes major firmware version
- 2 bytes minor firmware version
- 2 bytes build number
- 2 bytes RFU

The correct firmware version associated with this certificate is stated in Table 2.

1.3 TOE overview

The TOE is a V2X HSM (Vehicle-to-anything Hardware Security Module) which is used for secure cryptographic operations and key management.

The TOE serves a communication device (VCS) in Cooperative Intelligent Transport System (C-ITS).

The TOE is intended to be used in vehicle or in stationary deployments.

The TOE has an interface towards the VCS.

1.3.1 TOE type

The TOE is a discret chip security controller and realises an external V2X HSM according to the V2X HSM Protection Profile [V2X HSM PP].

1.3.2 TOE major security features

The TOE supports the VCS with cryptographic operations and key management functionality.

The TOE major security features are:

- Global Platform SCP03 protected communication
- Random number generation
- Digital signature generation
- User data ECIES encryption/decryption
- Protection of user data during transport and storages
- Supporting functions for Butterfly key derivation and implicit certificates
- Role based private key access control
- Private key import
- V2X Key Management
- Genuiness proof
- In field software update

1.3.3 Non TOE HW/SW/FW available to the TOE

There is no non-TOE HW/SW/FW available to the TOE

1.4 TOE description

Figure 1 displays a schematic overview of the TOE. The physical TOE is an integrated security chip in a VFQN-32 package. The TOE communicates with a SPI interface with the VCS. The physical boundary is the whole chip, i.e the content of the container “composite TOE of this ST”.

The blue parts show the certified parts (controller and libraries) of the underlying platform. The green parts show the certified parts of the composite TOE. The file slots and the associated functions provide a secure storage to the user. But this feature is not part of the TSF of this product.

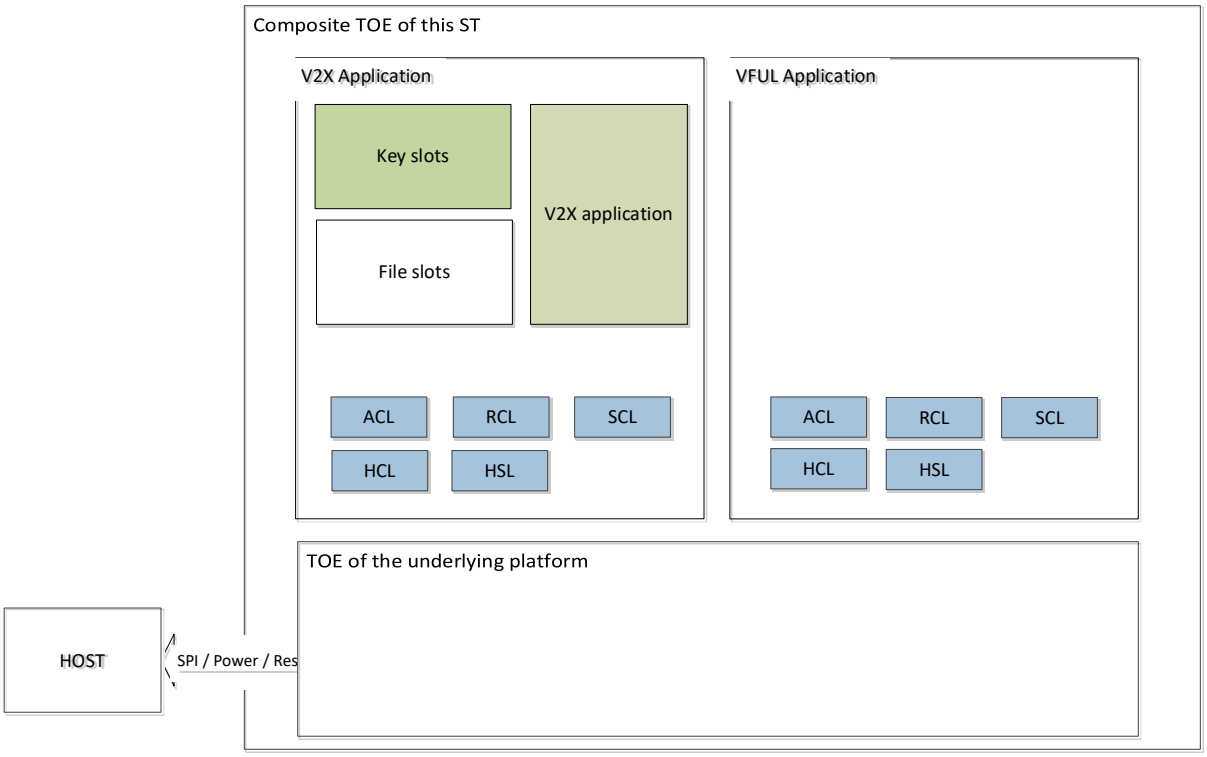


Figure 1 TOE schematic

1.4.1 Physical scope of the TOE

In Figure 1 , the certified part of the composite TOE is marked green. The scope of the certified underlying platform is marked in blue. The TOE provides file slots where the user can store arbitrary (non-key) data and associated commands to read and write to the file system. However this feature is not in the certification scope.

Table 1 Components of the underlying platform

Component	Version	Comment
CC Identifier	IFX_CCI_00003Dh	
Firmware	80.203.00.3	
Flash-loader	v8.06.001	Flash-loader is permanently disabled
Software libs		
ACL	v3.03.003	Asymmetric cryptographic library
SCL	v2.13.001	Symmetric cryptographic library
HCL	v1.13.001	Secure Hash library
RCL	v1.10.006	Secure RNG library
HSL	v2.01.6198	Hardware Support Layer services

Table 2 Embedded software

Component	Version	Comment
V2X application	01.03.4091	
VFUL application	01.03.3526	

Table 3 Guidance Document List

Guidance	Version
SLS37CSAEU V2X Databook	Rev 1.3
SLS37CSAEU V2X Errata and Update	Rev 1.3

1.4.2 Logical scope of the TOE

The logical TOE consists of an embedded software which contains the V2X application and the VFUL application. Both applications reside in separate NVM memory addresses and use seperate instances of the libraries provided by the underlying platform. The normal operating mode is provided by the V2X application. The VFUL application is used for downloading a new V2X application. A switch form the V2X application to the FVUL application can be triggered by a SET_LIFECYCLE command.

1.4.2.1 V2X mode:

This is the normal operation mode where the following TSF are available

- Random number generation:

The TOE uses a NIST SP800-90A conformant random number generator which is used for key generation and as an external service for the VCS.

- V2X Key Management:

The TOE handles key generation and secure internal storage of private keys.
The TOE generates ECC asymmetric key pairs for use in ECDSA digital signature generation. When generated inside the TOE, the generated public keys are exported to the VCS.
The TOE also generates ephemeral ECC asymmetric key pair for the need of ECIES encryption scheme.
Generated private keys are stored and protected by the TOE.
Keys and related cryptographic material can be destroyed when no longer needed.
The TOE can also import private keys used for decryption and signature generation.

- Digital Signature Generation

The TOE generates digital signatures according to the ECDSA (Elliptic Curve Digital Signature Algorithm) scheme.

- ECIES encryption/decryption

The TOE supports ECIES encryption and decryption according to [IEEE 1609.2].

- Self-protection

The TOE provides a resistance to Moderate attack potential based on hardware and software security measures allowing failure and physical attack resistance with preservation of a secure state.

- VCS Communication

Introduction

The TOE is a discrete security controller and communicates with the VCS over an SPI interface. This interface will be protected in access, integrity and confidentiality by a trusted channel protocol.

- Genuiness proof

The TOE provides an ECDSA signature to proof genuinity to a user

- Key Derivation
- The TOE supports Butterfly Key Generation according to [IEEE 1609.2.1] In-Field SW Update

##The TOE supports authenticated and protected in field update of the V2X application

1.4.2.2 V2X Field Update Mode (VFUL Mode)

The TOE embedded software can be updated in field by switching in the VFUL mode and then issuing Update commands. Only signed images can be loaded.

1.4.3 TOE life cycle

The TOE life cycle may be described in five phases: Development, manufacturing, platform integration, operational usage, and end-of-life. Because the TOE may support software update functionality, the TOE life cycle distinguishes two cases:

- Case 1: Initial provisioning of the TOE hardware and software
- Case 2: Software update of the TOE

The following figures shows the TOE lifecycle phases. The left side are the phases as defined in [V2X HSM PP] and the right side shows the corresponding phases from [PP0084].

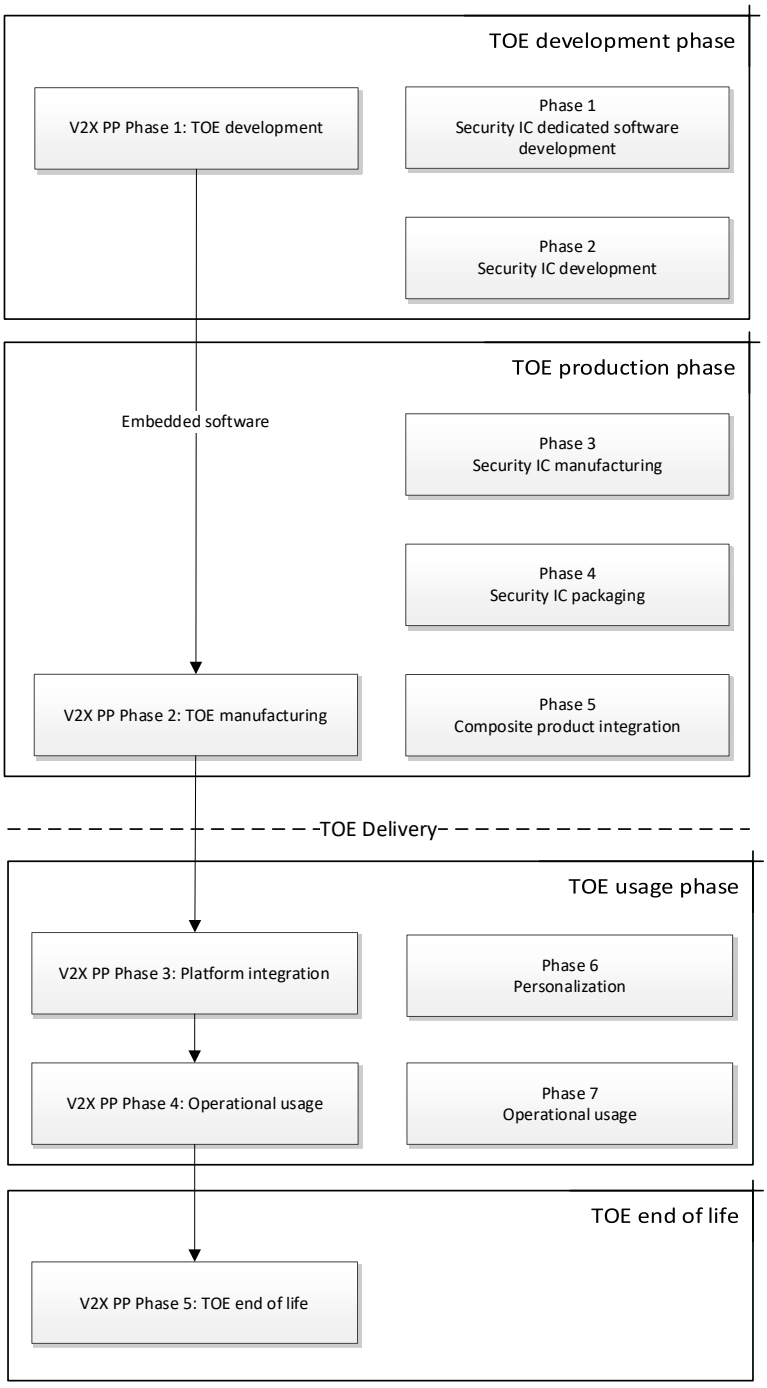
This TOE has an additional “Factory Reset” command which puts the the TOE back into lifecycle phase 3. This life cycle transition is possible in both lifecycle cases and in phases 3-5.

Note: *The default configuration does not allow to perform additional backward lifecycle transitions, i.e. it is not possible to switch from phase 5 to phase 4 or from phase 4 to phase 3. However during phase 3 the TOE can be configured such that those backward transitions are allowed.*

Note: *The TOE has an additional temporary lifecycle called “failure mode”. The TOE transitions from the current lifecycle into this lifecycle when one of the self-tests failed. After a power-on reset and successful execution of the self-tests the TOE is in the previous lifycycle again.*

1.4.3.1 Case 1

Figure 2 TOE lifecycle case 1

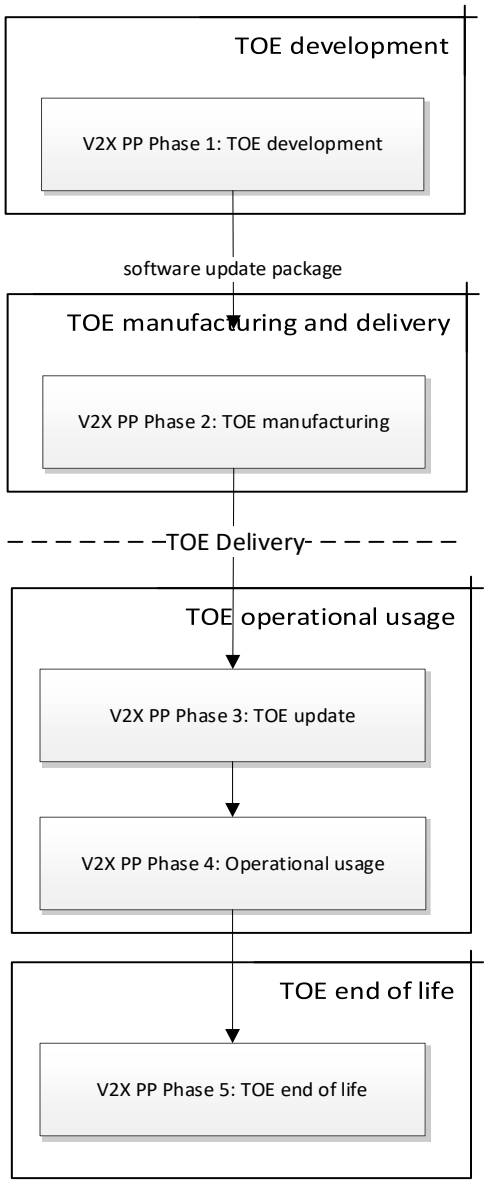


- V2X PP Phase 1: The V2X embedded software development takes place. This is done at Infineon Technologies AG.
- V2X PP Phase 2: Static configuration of the embedded software hex image is done. This is done at Infineon Technologies AG
  - Keys for SCP03 key derivation are written in the hex image.

  - The public ECDSA key for verification of the VFUL upload image is written in the hex image
  - The private ECDSA key for genuiness proof is written to the hex image
  - The hex image will be put on the chip by the standard Infineon process
- The TOE will be delivered to a platform integrator.
- V2X PP Phase 3: The TOE will be connected to a VCS by a third party platform integrator.
- V2X PP Phase 4: The V2X HSM is in operational phase.
- V2X PP Phase 5: TOE end of life
  In this phase, private and secret keys are not accessible anymore. The TOE will still provide its status.

1.4.3.2 Case 2

Figure 3 TOE lifecycle case 2



- V2X PP Phase 1: This phase comprises the development and testing of the TOE software updates to be installed on hardware of a previous TOE. This is done at Infineon Technologies AG.

- V2X PP Phase 2: Infineon creates the protected software update image.
- The TOE will be delivered to a platform integrator.
- V2X PP Phase 3: TOE Update
  The platform integrator or the end-user uses the update functionality to install the new TOE software on the hardware of the previous TOE.
- V2X PP Phase 4: The V2X HSM is in operational phase.
- V2X PP Phase 5: TOE end of life
  In this phase, private and secret keys are not accessible anymore. The TOE will still provide its status.This is the TOE End-of-Life. All assets will be destroyed.

1.5 TOE Delivery

The following table lists the parts of the TOE and the delivery formats

Table 4 TOE delivery

Part	Format	Delivery form
Hardware with embedded software	VQFN-32	Postal delivery to distribution centers in cages, locked
Guidance documentation	Personalized PDF	Secure download
Keys and certificates	Electronic file	Secure email

2 Conformance Claims

2.1 CC Conformance Claim

The ST is conformant to Common Criteria 3.1 revision 5:

- Part 2: extended [CCp2]
- Part 3: conformant [CCp3]

This assurance package conformance is EAL4 augmented by ALC_FLR.1 and AVA_VAN.4.

2.2 PP Conformance Claims

This ST claims strict conformance to the V2X HSM Protection Profile [V2X HSM PP].

The following optional pacakges from this PP have been included:

- Additional Communication Protections Package augmented
- Private Key Import (online) Package conformant
- Software Update Package conformant
- Key derivation package conformant

The packages have been directly merged together in this ST and will therefore not explicitly occur in this ST.

2.3 Conformance Claim Rationale

The Additional Communication Protections Package has been included, because the TOE is a discrete security controller which is connected to the host by an SPI interface. This package is augmented by three additional SFRs:

- FMT_MSA.1 and FMT_MSA.3, because the access authorization to private keys via this communication channel can be changed

The Private Key Import package has been included because the TOE provides a TSF to import private keys via the secure channel defined by the Additional Communication Protections Package

The Software Update Package has been included because the TOE provides the TSF to update the V2X software in operational phase.

The Key derivation package has been included because the TOE provides the optional Butterfly key derivation mechanism.

3 Security Problem Definition

3.1 Introduction

The security problem definition described below includes threats, organisational security policies and security usage assumptions.

3.2 Assets

Table 5 Assets to be protected by the TOE

Asset	Description
ECC private keys	Cryptographic keys used exclusively by the TOE. Several types of ECC private keys are handled: • ECC private keys used to perform digital signature operations; • ECC private keys used in ECIES. In V2X context, ECDSA private keys are: • Canonical Key: used to sign EC requests; • Enrolment Credential Keys: used to sign AT requests; • Authorization Ticket Keys: used to sign ITS messages. These assets must be protected in confidentiality and integrity for private ECC.
VCS data	User data exchanged between TOE and the VCS. In V2X context, VCS data can be: • Representation of parts of EC/AT requests or ITS information provided to the V2X HSM to be signed; • Data encryption key (symmetric) provided to the V2X HSM to be encrypted/decrypted (ECIES); • Recipient public key and parameters provided to the V2X HSM for ECIES encryption; • Sender (ephemeral) public key and parameters provided to the V2X HSM for ECIES decryption; • Public keys returned by TOE corresponding to ECC private keys generated by the TOE; • Random number generated by the TOE provided to VCS. User data must be protected at minimum in integrity. Furthermore, confidentiality protection is required for data to be ECIES encrypted/decrypted and for random numbers. The protections are needed during communication and while in operation by the TOE.
Secure Services	Secure services provided by the TSF to users, comprising all security functionality as defined in terms of “Major Security Features of the TOE” in section 1.3.2 and additionally all security functionality defined by functional packages claimed. Secure services must be protected in runtime integrity.
HSM Software	Encoded instructions that regulate the behaviour of the TOE. HSM software must be protected in integrity.
Added from Additional Communication Protections Package & Private Key Import (online) Package	

Security Problem Definition

Trusted channel keys	Cryptographic keys used for establishment and maintenance of trusted channel allowing entity authentication, and data authentication and/or confidentiality. This asset must be protected in confidentiality and integrity.
Added from Software Update Package	
Software Update keys	Cryptographic keys used for verification of authenticity and integrity of the software update image. This asset must be protected in integrity for public key and in integrity and confidentiality for private/secret key. Please note that the software update keys will not be changed.
Software Update Image	HSM Software image loaded onto the TOE to replace whole or part of the current one. Software images must be protected in integrity and authenticity.

3.3 Users

Table 6 gives a generic basic description of V2X HSM users.

Table 6 TOE users

User	Description
VCS (IT Entity)	User invoking Secure Services of the TOE.

3.4 Threat Agents

Two main types of attackers have been identified, both attacker types have moderate attack potential.

Table 7 Threat agents

Threat Agent	Description
Local attacker	Attacker with physical access to the TOE; such attacker does not have an authorized access to the TOE services (other than through the VCS during operation of the vehicle). Local attacker can run hardware or software attacks through physical or logical TOE interfaces.
Remote attacker	Attacker with access (authorized or not) through the VCS; such attacker has an authorized access to the TOE services by means of VCS. Remote attacker can run hardware or software attacks through logical TOE interfaces only.

3.5 Threats

Threats are described by an adverse action performed by defined threat agents on the assets that the TOE has to protect.

Attackers in V2X networks will have two objectives in the final V2X context:

- Be able to track a vehicle.
- Cause safety hazardous situation.

The V2X HSM provides supporting functionalities to prevent such risks.

Security Problem Definition

The threats against the TOE according to Table 8 are identified. In this table, the generic term “attacker” is used to cover both local and remote type of attacker (see previous section). Attacks on data can be “direct” or using existing services.

Table 8 Threats against the TOE

Threat	Description	Asset / protection
T.KEY_REPLACE	An attacker is able to replace a key stored in internal ~~or external~~ NVM by one he knows (e.g. generated by him or taking a weak value) without being detected by the TOE. In V2X context, the attacker will be able to: • track the victim vehicle (key known); • request a certificate for the public key and then sign himself (out of TOE) wrong information (on behalf of the victim or of himself). Note: The integrity only covers changes controlled by an attacker leading to knowledge of private keys, or modification of public key to value chosen by the attacker. Compromise of the integrity of keys leading to unavailability of the device is not in the scope of this ST.	ECC private keys / integrity
T.KEY_DISCLOSE	An attacker is able to disclose the private key (stored in internal or external NVM). In V2X context, the attacker will be able to: • track the victim vehicle (key known); • sign himself (out of TOE) wrong information (on behalf of the victim or himself).	ECC private keys / confidentiality
T.SW_TAMPER	An attacker is able to modify the HSM software; he then has a partial control of the TOE behaviour and potentially on assets. In V2X context, various exploitations will be possible depending on the modifications (see impacts in other threats as examples).	HSM Software / integrity
T.SRV_MALFUNCTION	An attacker may take advantage of a malfunction of the Secure Services. This may affect any asset and could result in any of the other threats.	Secure Services / integrity
T.SW_REPLACE	An attacker is able to directly replace the HSM software; he then has the full control on TOE behaviour and then on assets. In V2X context, all exploitation will be possible (see impacts in other threats as examples).	HSM Software / integrity
T.VCS_DATA_MODIF	An attacker is able to modify VCS data during communication and when processed by the TOE. In V2X context, the attacker will then be able to make sign wrong information; if modifications are controlled so the message can be interpreted by receivers, it can provoke an undesired reaction of the vehicle; if modifications are not controlled and cannot be interpreted, this could at least make receivers consume resources unduly or provoke	VCS data / integrity

Security Problem Definition

Threat	Description	Asset / protection
	unexpected reactions of receiver devices (e.g. crash).	
T.VCS_DATA_DISCLOSE	An attacker is able to disclose VCS data during communication and when processed by the TOE when confidentiality has been requested by User. In V2X context, when data is the data encryption key the attacker will then be able to decrypt data exchanged between VCS and PKI. The exchanged data comprises certificate signing requests, including long term identity of the vehicle, as well as authorization tickets. If this information is disclosed the privacy of the vehicle it compromised. When data is random number used for key generation by the VCS, the attacker will then be able to disclose the data encryption key.	VCS data / confidentiality
Added from Software Update Package		
T.SW_UPDATE	An attacker is able to replace the HSM software through the software update mechanism; if an older image is installed, the attacker could target unpatched vulnerabilities; if a forged image is installed, he then has control on TOE behaviour. In V2X context, various exploitations will be possible depending on the modifications (see impacts in other threats as examples).	Software Update Image / integrity and authenticity
Added in this ST		
T.MASQUERADE	An attacker may threaten the property being a genuine TOE by producing an IC which is not a genuine TOE but wrongly identifying itself as genuine TOE.	HSM Software / integrity

3.6 Organisational Security Policies

Table 9 Organisational Security Policies, OSPs

Organisational Security Policy	Description
P.SIGNATURE_GENERATION	The TOE shall be able to generate ECDSA digital signatures.
P.KEY_GENERATION	The TOE shall be able to generate ECC asymmetric key pairs for ECDSA and ECIES operations.
P.ECIES	The TOE shall be able to encrypt and decrypt VCS data according to ECIES.
P.RNG	The TOE is required to generate random numbers that meet specified quality metric, for use by the TOE itself and the VCS. These random numbers shall be suitable for use as keys, authentication/authorisation data or seed data for another random number generator.
P.SECURE_COMMUNICATION	The TOE environment must implement protection for integrity, and confidentiality if required, of VCS data when exchanged between the TOE and the VCS.

Security Problem Definition

Organisational Security Policy	Description
P.SRV_ACCESS	Access to the V2X HSM services shall be restricted to the VCS only.
Added from Additional Communication Protections Package	
P.TRUSTED_CHANNEL	The TOE shall be able to establish the trusted channel.
Added from Private Key Import (online) Package	
P.PRIVKEY_IMPORT_TC	The TOE shall be able to import ECC private keys generated externally through trusted channel.
Added from Key Derivation Package	
P.KEY_DERIVE	The TOE and the operational environment together shall implement or support key derivation following [IEEE 1609.2.1] chapter “9.3 Butterfly key mechanism”.
Added from Software Update Package	
P.SW_UPDATE	The TOE shall be update-able following related TOE security guidance.

3.7 Assumptions

Table 10 Assumptions on the TOE environment

Assumption	Description
A.INTEGRATION	It is assumed that appropriate technical and/or organisational security measures in the Platform Integration (Phase 3) in order to guarantee for the confidentiality, integrity and authenticity of the assets of the TOE
Added from Private Key Import (online) Package	
A.KEY_EXT_MANAGEMENT	It is assumed that in case a key pair is generated outside the TOE to be then imported, this one is securely managed: • Key generation service shall be provided to authorized users only; • Key generation shall be performed in accordance with [FIPS 186-4], [RFC 5639]; • Confidentiality of private key shall be ensured while outside the TOE.

4 Security Objectives

4.1 Introduction

The statement of security objectives defines the security objectives for the TOE and its environment. The security objectives intend to address all security environment aspects identified. The security objectives reflect the stated intent and are suitable to counter all identified threats and cover all identified organisational security policies and assumptions. The following categories of objectives are identified:

- The security objectives for the TOE shall be clearly stated and traced back to aspects of identified threats to be countered by the TOE and/or organisational security policies to be met by the TOE.
- The security objectives for the environment shall be clearly stated and traced back to aspects of identified threats countered by the TOE environment, organisational security policies or assumptions.

4.2 Security Objectives for the TOE

The following security objectives for the TOE (OT) are defined.

Table 11 Security objectives for the TOE

Security Objective	Description
OT.SIGNATURE_GENERATION	The TOE shall be able to generate ECDSA digital signatures on VCS data.
OT.KEY_MANAGEMENT	The TOE shall be able to generate, store, and protect ECC asymmetric keys for ECDSA and ECIES operations.
OT.ECIES	The TOE shall be able to encrypt and decrypt VCS data according to ECIES (as described in [V2X HSM PP] ch. 2.1.4).
OT.TOE_SELF-PROTECTION	The TOE shall be able to protect itself and its assets from manipulation including physical and software tampering.
OT.PRIVKEY_ACCESS	The TOE shall ensure that private keys can only be used through V2X services to which access is restricted to User and cannot be retrieved out of the TOE in a form allowing usage outside of the TOE.
OT.RNG	Random numbers generated shall meet a defined quality metric in order to ensure that random numbers are not predictable and have sufficient entropy. For security operations, e.g. key generation, high quality random numbers are required.
OT.VCS_DATA	The TOE shall implement security measures to prevent alterations, and disclosure when confidentiality is requested, of received user data stored and processed in the TOE.
Added from Additional Communication Protections Package	
OT.ACCESS_CONTROL	The TOE shall implement protections to restrict the access to the Secure Services to authorized user only.
OT.AUTHENTICATION	The TOE shall verify that communication links are established with the expected VCS.
OT.TRUSTED_CHANNEL	The TOE shall enforce the establishment of a trusted channel with the VCS and usage for secure communication.
Added from Private Key Import (online) Package	
OT.PRIVKEY_IMPORT_TC	The TOE shall be able to import ECC private keys generated externally.
Added from Key Derivation Package	

Security Objectives

OT.KEY_DERIVE	The TOE shall support private key derivation following [IEEE 1609.2.1] chapter "9.3 Butterfly key mechanism". See note below for an explanation what "support" mean.
Added from Software Update Package	
OT.SW_UPDATE	The TOE shall be able to update whole or part of its software with an authorized image i.e. authenticity and integrity verifications are performed on loaded image before installation process. The TOE shall protect against loading of an older image version.
Defined in this ST	
OT.GENUINITY	The TOE shall provide cryptographic means to proof genuinity.

4.3 Security Objectives for the Operational Environment

The following security objectives for the Environment (OE) are defined.

Table 12 Security objectives for the TOE operational environment

Security Objective	Description
~~OE.SECURE_COMMUNICATION~~	~~The TOE operational environment must implement protections for integrity and confidentiality of VCS data when exchanged between the TOE and the VCS in accordance with protections specified in chapter 3.2 (asset definition). This protection can be limited to physical protection.~~
~~OE.SRV_ACCESS~~	~~The TOE environment must implement security measures to restrict V2X HSM services access to the VCS only. This protection can be limited to physical protection.~~
OE.INTEGRATION	Appropriate technical and/or organisational security measures shall be in place in the Platform Integration (Phase 3) in order to guarantee the confidentiality, integrity and authenticity of the assets of the TOE in accordance with protections specified in chapter 3.2 (asset definition).
Added from Additional Communication Protections Package	
OE.TRUSTED_CHANNEL	The VCS shall be able to handle the establishment of a trusted channel with the TOE and use it for secure communication.
Added from Private Key Import (online) Package	
OE.KEY_MANAGEMENT	In case a key pair is generated outside the TOE to be then imported, the environment shall ensure that this one is securely managed: • Key generation service shall be provided to authorized users only; • Key generation shall be performed in accordance with [FIPS 186-4], [RFC 5639]; • Confidentiality of private key shall be ensured while outside the TOE.
Added from Key Derivation Package	
OE.KEY_DERIVE	The operational environment shall provide inputs for key derivation following [IEEE 1609.2.1] chapter "9.3 Butterfly key mechanism". These inputs shall be protected in integrity and confidentiality by the environment (for privacy reasons).

Note: Due to inclusion of the Additional Communication Protection Package, the following modifications are performed:
- OE.SECURE_COMMUNICATION is replaced by OT.TRUSTED_CHANNEL and OE.TRUSTED_CHANNEL - OE.SRV_ACCESS is replaced by OT.ACCESS_CONTROL

Security Objectives

Note: *The TSF supports the Butterfly key Derivation and the Implict Certificate generation by providing the mathematical functions*
$prk_{der} = (a * prk) + b \ (mod\ p)$ *or* $prk_{der} = (a + prk) * b \ (mod\ p)$,
where prk is some private key, p is the prime of the elliptic curve field and a and b are some arbitrary parameters provided by the VCS
OT.KEY_DERIVE takes care of prk and prk_{der}
OE.KEY_DERIVE takes care of the parameters a and b

4.4 Security Objectives Rationale

4.4.1 Security Objectives Coverage

This section provides tracings of the security objectives for the TOE to threats, OSPs, and assumptions.

Table 13 Security objectives coverage

	OT.PRIVKEY_ACCES	OT.SIGNATURE_GENERATION	OT.KEY_MANAGEMENT	OT.ECIES	OT.TOE_SELF-PROTECTION	OT.RNG	OT.VCS_DATA	OT.ACCESS_CONTROL	OT.AUTHENTICATION	OT.TRUSTED_CHANNEL	OT.PRIVKEY_IMPORT_TC	OT.SW_UPDATE	OT.GENUINITY	OT.KEY_DERIVE	OE.TRUSTED_CHANNEL	OE.INTEGRATION	OE.KEY_MANAGEMENT	OE.KEY_DERIVE
T.KEY_REPLACE	X		X		X													
T.KEY_DISCLOSE	X		X		X													
T.SW_TAMPER					X													
T.SRV_MALFUNCTION					X													
T.SW_REPLACE					X													
T.VCS_DATA_MODIF					X		X			X					X			
T.VCS_DATA_DISCLOSE					X		X			X					X			
T.SW_UPDATE												X						
T.MASQUERADE													X					
P.SIGNATURE_GENERATION		X				X												
P.KEY_GENERATION			X			X												
P.ECIES				X		X												
P.RNG						X												
P.SRV_ACCESS								X	X									
P.TRUSTED_CHANNEL										X					X			
P.PRIVKEY_IMPORT_TC										X	X				X			
P.SW_UPDATE												X						
P.KEY_DERIVE														X				X
A.INTEGRATION																X		

Security Objectives

A.KEY_EXT_MANAGEMENT																X	

4.4.2 Security Objectives Sufficiency

The following rationale provides justification that:

- the security objectives for the environment are suitable to cover each individual assumption or threat to the environment;
- each security objective for the environment that traces back to a threat or an assumption about the environment of use.

Table 14 Security objectives sufficiency

Threat/OSP/Assumption	Objective	Rationale
T.KEY_REPLACE	OT.KEY_MANAGEMENT	Once generated, private keys are securely stored.
	OT.PRIVKEY_ACCESS	Logical write access to private keys stored in the TOE is only possible through the Secure Services to which access is restricted to User.
	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering to prevent – among others – replacement of keys by circumventing access control.
T.KEY_DISCLOSE	OT.KEY_MANAGEMENT	Once generated, private keys are securely stored.
	OT.PRIVKEY_ACCESS	Logical read access to private keys is only possible through the Secure Services to which access is restricted to User.
	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering to prevent – among others – disclosure of keys by circumventing access control.
T.SW_TAMPER	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering, thus parts of the TOE cannot be modified.
T.SRV_MALFUNCTION	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering protecting against any malfunction.
T.SW_REPLACE	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering, thus the HSM

Security Objectives

Threat/OSP/Assumption	Objective	Rationale
		software cannot be illegally replaced.
T.VCS_DATA_MODIF	OT.VCS_DATA	The VCS data have integrity protections when stored or processed by the TOE.
	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering protecting against illegal modification of – among others – VCS data processed inside the TOE.
	OE.TRUSTED_CHANNEL OT.TRUSTED_CHANNEL	The integrity of VCS data during communication is protected by a trusted channel between V2X HSM and VCS
T.VCS_DATA_DISCLOSE	OT.VCS_DATA	The VCS data have confidentiality protections when stored or processed by the TOE.
	OT.TOE_SELF-PROTECTION	The TOE is protected from physical and software tampering protecting against any illegal disclosure of – among others – VCS data processed inside the TOE.
	OE.TRUSTED_CHANNEL OT.TRUSTED_CHANNEL	The confidentiality of VCS data during communication is protected by a trusted channel between V2X HSM and VCS.
T.MASQUERADE	OT.GENUINITY	The TOE provides a proof that it is a genuine one.
P.SIGNATURE_GENERATION	OT.SIGNATURE_GENERATION	OT.SIGNATURE_GENERATION is rephrasing the OSP.
	OT.RNG	The quality of the random numbers required for signatures is ensured by the TOE.
P.KEY_GENERATION	OT.KEY_MANAGEMENT	OT.KEY_MANAGEMENT is rephrasing the OSP.
	OT.RNG	Key generation inside the TOE is based on a random number generation ensuring randomness quality.
P.ECIES	OT.ECIES	OT.ECIES is rephrasing the OSP.

Security Objectives

Threat/OSP/Assumption	Objective	Rationale
	OT.RNG	The quality of the random numbers required for encryption based on ECIES is ensured by the TOE.
P.RNG	OT.RNG	OT.RNG is rephrasing the OSP.
P.SECURE_COMMUNICATION	OE.TRUSTED_CHANNEL	Instead of OE.SECURE_COMMUNICATION, the operational environment uses a trusted channel for VCS data protection.
P.SRV_ACCESS	OT.ACCESS_CONTROL OE.AUTHENTICATION	The access control feature is directly addressed by the TOE through OT.ACCESS_CONTROL and based on OT.AUTHENTICATION.
P.TRUSTED_CHANNEL	OT.TRUSTED_CHANNEL OE.TRUSTED_CHANNEL	The trusted channel feature is addressed by the TOE through the OT.TRUSTED_CHANNEL; the other channel end-point is handled through the objective on the environment OE.TRUSTED_CHANNEL.
P.PRIVKEY_IMPORT_TC	OT.PRIVKEY_IMPORT_TC OT.TRUSTED_CHANNEL OE.TRUSTED_CHANNEL	The private key import feature is addressed by the TOE through the OT.PRIVKEY_IMPORT_TC, OT.TRUSTED_CHANNEL and the OE.TRUSTED_CHANNEL.
P.SW_UPDATE	OT.SW_UPDATE	OT.SW_UPDATE counters the threat that the TOE can be updated with a modified, illegal software update image or with a software update image containing an older HSM software version than currently installed
P.KEY_DERIVE	OT.KEY_DERIVE OE.KEY_DERIVE	OT.KEY_DERIVE mirrors policy for the TOE. The generation of input parameters generation while outside of the TOE must also be securely handled which is covered by OE. KEY_DERIVE
A.INTEGRATION	OE.INTEGRATION	Objective mirrors assumption
A.KEY_EXT_MANAGEMENT	OE.KEY_MANAGEMENT	Objective mirrors assumption

5 Extended Components Definition

5.1 FCS_RNG (Random Number Generation)

This extended component is defined in the V2X HSM Protection Profile [V2X HSM PP]

5.2 FCS_CKM.5 (Cryptographic Key Derivation)

This extended component is defined in the V2X HSM Protection Profile [V2X HSM PP]

5.3 FIA_API Authentication Proof of Identity

This extended component is defined in [PP0084], ch. 7.2.2.

6 Security Requirements

6.1 Definitions

6.1.1 Formatting Conventions

Operations on the SFRs are identified as follows:

- Fixed assignments and selections by the [V2X HSM PP] are printed in underline text.
- Open assignments and selections are printed in **[**underline text**]** surrounded by square brackets and a footnote. The footnote describes the original content from [CC part2] , [V2X HSM PP] or the extended component definition from chapter 5.
- Refinements are marked with text literal “(refined)” and annotated by an application note;
- Iterations are denoted by appending a slash and a descriptive identifier.
- In case of multiple iterations of a single SFR, tables are used to define SFRs in a condensed form (each table line stating the iteration identifier and all operations for that iteration).

6.1.2 Subjects, objects and security attributes

The following table defines subjects, objects and information which will be used in security functional requirements.

Table 15 Definition of subjects, objects and security attributes

Subject/Object	Security Attribute	Value	comment
S.User	Role	R.VCS	Refined from ACP package of [V2X HSM PP]
		R.ADMIN, R.USER	See note below
S.SWU	Current Version	Var	Added from SWU package of [V2X HSM PP]
S.ImportComponent	-	-	Added from private key import (online) package of [V2X HSM PP]
O.PrivateKey	Type	Sign,encrypt,generic	Added in this ST
	read, delete, change_access	R.VCS, R.ADMIN, R.USER	Added in this ST
O.ImageUpdate	New Version	Var	Added from SWU package of [V2X HSM PP]
	Software Update Signature	Var	Added from SWU package of [V2X HSM PP]

Note: The roles R.ADMIN and R.USER are defined in this ST. R.VCS can be either R.ADMIN or R.USER. Therefore those roles are refinements of R.VCS. The role R.VCS stands for either one of R.ADMIN or R.USER

Note: S.ImportComponent does not use a dedicated security attribute associated to the imported key. S.ImportComponent uses the role of S.User to check authorization for key import. The

role which is authorized to import the key is defined by the key slot where this key is supposed to be stored. See also TSS in ch. 7.6

6.1.3 Operations

The following table defines operations which will be used in security functional requirements.

Table 16 Definition of operations

Operations	Comment
OP.KeyPair_Gen	Copied from base PP of [V2X HSM PP]
OP.RNG	Copied from base PP of [V2X HSM PP]
OP.Signature	Copied from base PP of [V2X HSM PP]
OP.EncDec	Copied from base PP of [V2X HSM PP]
OP.SWU	Copied from SWU package of [V2X HSM PP]
OP.Import	Copied from private key import (online) package of [V2X HSM PP]
OP.Key_derive	Copied from key derivation package of [V2X HSM PP]
OP.Delete	Delete a private key. This operation has been added in this ST
OP.Change_access	Change access conditions for a private key. This operation has been added in this ST

6.1.4 Security Functional Policies

The following section defines security functional policies which will be used in security functional requirements.

6.1.4.1 Private Key Access Control SFP

The TOE shall enforce this SFP to forbid the direct access to ECC private keys. The access to ECC private keys is allowed only via the Secure Services. No user authentication, nor role management is required to be performed by the TOE, as this is handled by operational environment, see OE.SRV_ACCESS.

6.2 Security Functional Requirements

6.2.1 SFRs from base PP of [V2X HSM PP]

6.2.1.1 FCS_CKM.1

FCS_CKM.1.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm ECC Key Pair Generation and specified cryptographic key sizes 256 bits [and 384 bits][1] that meet the following: [FIPS 186-4] Appendix B.

[1] [assignment: additional larger cryptographic key size]

Security Requirements

Note: This certification covers the following curves:
NIST standard curves from [FIPS 186-4] : P256, P384
Brainpool curves from [RFC 5639]: BrainpoolP256r1, BrainpoolP384r1

6.2.1.2 FCS_CKM.4

FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [key zeroization][1] that meets the following: [none][2].

6.2.1.3 FCS_RNG.1

FCS_RNG.1.1	The TSF shall provide a [hybrid deterministic][3] random number generator that implements: [NIST [SP 800-90A] CTR_DRBG][4].
FCS_RNG.1.2	The TSF shall provide random numbers that meet [NIST [SP 800-90A] validation testing][5].

6.2.1.4 FCS_COP.1

FCS_COP.1.1/<iter>	The TSF shall perform the operations according to Table 17 in accordance with a specified cryptographic algorithm according to Table 17 and cryptographic key sizes according to Table 17 that meet the following: standards according to Table 17.

Table 17 FCS_COP.1 operations, algorithms and key sizes

<iter>	Operation	Algorithm	Key length	Standard
ECDSA	Digital signature generation	ECDSA with NIST and Brainpool prime curves	256 bits and 384 bits and 521 bits[6]	algorithms: [SEC-1] and [FIPS 186-4] curves: [FIPS 186-4] [RFC 5639].
ECIES_ENC	ECIES Encryption	ECIES with NIST and Brainpool prime curves	256 bits and 384 bits[7]	algorithm: [IEEE 1363a] curves:

[1] [assignment: cryptographic key destruction method]
[2] [assignment: list of standards]
[3] [selection: physical, deterministic, hybrid physical, hybrid deterministic]
[4] [assignment: list of security capabilities]
[5] [assignment: a defined quality metric]
[6] [assignment: additional larger key size]
[7] [assignment: additional larger key size]

				[FIPS 186-4] [RFC 5639]
ECIES_DEC	ECIES Decryption	ECIES with NIST and Brainpool prime curves	256 bits and 384 bits[1]	algorithm: [IEEE 1363a] curves: [FIPS 186-4] [RFC 5639]

Note: The hashing part of ECDSA algorithm will be performed outside of the TOE.

Note: For ECIES encryption/decryption, the choices described in [IEEE 1609.2] Section 5.3.5 apply

Note: The certification covers the following curves for ECDSA used in V2X message signing and ECIES_ENC and ECIES_DEC operations:
- NIST standard curves from [FIPS 186-4] : P256, P384
- Brainpool curves from [RFC 5639]: BrainpoolP256r1, Brainpool384r1

Note: The NIST P521 curve is used with ECDSA for Infineon genuiness proof and for checking the image in the software update package.

6.2.1.5 FDP_RIP.1

FDP_RIP.1.1	The TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from the following objects: O.PrivateKey and any working copies of ECC private key values.

6.2.1.6 FDP_SDI.2

FDP_SDI.2.1	The TSF shall monitor user data stored in containers controlled by the TSF for [integrity errors detected by a checksum][2] on all objects, based on the following attributes: [none][3].
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF shall: • prevent use of the altered data; • [perform a security reset][4].

[1] [assignment: additional larger key size]
[2] [assignment: integrity errors]
[3] [assignment: user data attributes]
[4] [assignment: other action to be taken]

6.2.1.7 FDP_ACC.1

FDP_ACC.1.1	The TSF shall enforce the Private Key Access Control SFP on • Subjects: S.User • Objects: O.PrivateKey • Operations:OP.KeyPair_Gen, OP.Signature, OP.EncDec, Op.Delete, Op.Change_access [1]

6.2.1.8 FDP_ACF.1

FDP_ACF.1.1	The TSF shall enforce the Private Key access control SFP to objects based on the following: • Subjects: S.User • Objects: O.PrivateKey • Security attributes: – S.User with security attribute: Role. – [O.PrivateKey with security attributes: read, delete, change_access, type][2].
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: • O.PrivateKey can only be accessed by S.User through operations involving private keys (OP.KeyPair_Gen, OP.Signature, OP.EncDec) when S.User has Role "R.VCS"; • [Additional rules defined in this ST: – O.PrivateKey_can be used for OP.Signature_only if attribute "read"_matches role of S.User and attribute type has "sign" or "generic". – O.PrivateKey_can be used for OP.EncDec only if attribute_"read"_matches role of S.User and attribute type has "encdec" or "generic" – Security attributes of_O.PrivateKey can be canged, if attribute "change_access" matches role of S.User. – O.PrivateKey_can be used for OP.Delete only if attribute "delete" matches role of S.User][3].
FDP_ACF.1.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: none.
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: • No one shall be able to retrieve O.PrivateKey in a form allowing usage outside of the TOE; • [None][4].

[1] [assignment: list of additional operations]

[2] [assignment: list of SFP-relevant security attributes or named groups of SFR-relevant security attributes]

[3] [assignment: additional rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

[4] [assignment: other rules, based on security attributes, that explicitly deny access of subjects to objects]

Note: This SFR is from the Additional Communication Protection Package and replaces the SFR from the base PP. See [V2X HSM PP] for a justification of this transformation.

6.2.1.9 FPT_FLS.1

FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: • Failing self-test according to FPT_TST.1; • [Detecting checksum errors in keys]1.

6.2.1.10 FPT_PHP.3

FPT_PHP.3.1	The TSF shall resist physical tampering to the all TOE components implementing the TSF by responding automatically such that the SFRs are always enforced.

Note: The TOE is not always powered and therefore not able to detect, react or notify that it has been subject to tampering. Nevertheless, its design characteristics make reverse-engineering and manipulations etc. more difficult. This is regarded as being an "automatic response" to tampering. Therefore, the security functional component resistance to physical attack (FPT_PHP.3) has been selected. The TOE also provide features to actively respond to a possible tampering attack which is also covered by FPT_PHP.3.

6.2.1.11 FPT_TST.1

FPT_TST.1.1	The TSF shall run a suite of self tests [during initial start-up, at the request of the authorised user][2] to demonstrate the correct operation of [the TSF][3].
FPT_TST.1.2	The TSF shall provide authorised users with the capability to verify the integrity of [TSF data][4].
FPT_TST.1.3	The TSF shall provide authorised users with the capability to verify the integrity of [TSF][5].

[1] [assignment: list of other types of failures in the TSF]

[2] [selection: during initial start-up, periodically during normal operation, at the request of the authorised user, at the conditions[assignment: conditions under which self test should occur]]

[3] [selection: [assignment: parts of TSF], the TSF]

[4] [selection: [assignment: parts of TSF data], TSF data]

[5] [selection: [assignment: parts of TSF], TSF]

6.2.2 SFRs from Additional Communication Protections Package (ACP)

This package has been included because the TOE is a discrete security chip, which communicates with the host over a SPI wire bus. The TOE implements the GlobalPlatform [SCP03] protocol for the trusted channel.

6.2.2.1 FDP_UCT.1/ACP

FDP_UCT.1.1/ACP	The TSF shall enforce the Private Key access control SFP to transmit and receive confidential VCS Data ~~user data~~ in a manner protected from unauthorized disclosure.

Note: Confidential VCS Data covers only the VCS Data defined in the assets list as confidential.

6.2.2.2 FDP_UIT.1/ACP

FDP_UIT.1.1/ACP	The TSF shall enforce the Private Key access control SFP to receive VCS Data ~~user data~~ in a manner protected from modification and insertion errors.
FDP_UIT.1.2/ACP	The TSF shall be able to determine on receipt of VCS data ~~user data~~, whether modification or insertion has occurred.

Note: The ECDSA signatures are protected by their nature, as such protection for transmit is not needed for OP.Signature operation.

6.2.2.3 FMT_SMR.1

FMT_SMR.1.1	The TSF shall maintain the roles R.VCS [R.ADMIN, R.USER][1].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

6.2.2.4 FMT_SMF.1

FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: modification of Trusted channel keys [and modification of security attributes for private keys][2].

6.2.2.5 FMT_MTD.1

FMT_MTD.1.1	The TSF shall restrict the ability to modify the trusted channel keys to [R.VCS][3].

[1] [assignment: other authorised identified roles]

[2] [assignment: list of other management functions to be provided by the TSF]

[3] [assignment: the authorised identified roles]

Note: *The TOE contains separate Trusted channel (TC) keys for R.ADMIN and R.USER. Role R.VCS can be either R.ADMIN or R.USER. R.ADMIN can modify the TC keys of R.ADMIN and R.USER, but R.USER can only modify the TC keys of R.USER.*

Note: *The TOE derives card individual trusted channel keys by first usage of the trusted channel. This will normally be done in phase 3 of the TOE life cycle. (see ch. 6.2.2.11)*

6.2.2.6 FIA_UID.1

FIA_UID.1.1	The TSF shall allow: • Self-test according to FPT_TST.1; • Establishment of a trusted channel up to all steps needed for identification and authentication of its end points; • [Perform GET_INFO, RUN_SELFTEST, GET_SELFTEST_STATUS][1] on behalf of the user to be performed before the user is identified.
FIA_UID.1.2	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

6.2.2.7 FIA_UAU.1

FIA_UAU.1.1	The TSF shall allow: • Self-test according to FPT_TST.1; • Establishment of a trusted channel up to all steps needed for identification and authentication of its end points; • [Perform GET_INFO, RUN_SELTEST, GET_SELFTEST_STATUS][2] on behalf of the user to be performed before the user is authenticated.
FIA_UAU.1.2	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

[1] [assignment: other TSF-mediated actions]

[2] [assignment: other TSF-mediated actions]

6.2.2.8 FTP_ITC.1/ACP

FTP_ITC.1.1/ACP	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2/ACP	The TSF shall permit another trusted IT product to initiate communication via the trusted channel.
FTP_ITC.1.3/ACP	The TSF shall ~~initiate~~ enforce communication via the trusted channel for: Transfer of VCS data, [none][1].

Note: *"Another trusted IT product" is in the V2X context the VCS.*

Note: *The following management SFRs have been added in this ST, because the private keys in this TOE have security attributes to control access of the keys by different roles.*

6.2.2.9 FMT_MSA.3

FMT_MSA.3.1	The TSF shall enforce the [Private Key access control SFP][2] to provide [restrictive][3] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the [R.ADMIN][4] to specify alternative initial values to override the default values when an object or information is created.

Note: *The default values for "read" and "delete" security attributes are "R.VCS", which means that any authenticate role can use or delete the private key. The default value for security attribute "change_access" is R.ADMIN, which mean s that only the admin can redefined all security attributes. The role R.ADMIN can therefore restrict the usage and deletion of a key to a specific role.*

6.2.2.10 FMT_MSA.1

FMT_MSA.1.1	The TSF shall enforce the [Private Key access control SFP][5] to restrict the ability to [modify][6] the security attributes [read, delete, change_access][7] to [the role defined by the attribute "change_access"][8].

[1] [assignment: list of additional functions for which a trusted channel is required]
[2] [assignment: access control SFP, information flow control SFP]
[3] [selection, choose one of: restrictive, permissive, [assignment: other property]]
[4] [assignment: the authorised identified roles]
[5] [assignment: access control SFP(s), information flow control SFP(s)]
[6] [selection: change_default, query, modify, delete, [assignment: other operations]]
[7] [assignment: list of security attributes]
[8] [assignment: the authorised identified roles]

6.2.2.11 FCS_CKM.5/AES

FCS_CKM.5.1/AES	The TSF shall support derivation of cryptographic keys [AES keys][1] from [AES keys][2] in accordance with a specified cryptographic key derivation algorithm [KDF-AES256-CMAC-CTR][3] and specified cryptographic key sizes [256 bit][4] that meet the following: [SP800-38B] and [SP800-108][5].

Note: This SFR has been added in this ST, because on initial usage of the tusted channel, the device individual trusted channel keys will be derived from global pairing keys, which are injected during personalization.

Note: The verb "support" here actually means that the TOE implements the whole key derivation.

6.2.3 SFRs from Private Key Import (online) Package

This package will be used in conjunction with the "Additional communications protection package" defined in chapter 6.2.2. The "Additional communications protection package" is used to verify the authenticity of the endpoints of the secure channel.

6.2.3.1 FTP_ITC.1/Import_TC

FTP_ITC.1.1/Import_TC	The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2/Import_TC	The TSF shall permit another trusted IT product to initiate communication via the trusted channel.
FTP_ITC.1.3/Import_TC	The TSF shall initiate communication via the trusted channel for: OP.Import.

6.2.3.2 FDP_ACC.1/Import_TC

FDP_ACC.1.1/Import_TC	The TSF shall enforce the PrivateKey Import TC SFP on • Subject: S.ImportComponent • Object: O.PrivateKey • Operation: OP.Import

6.2.3.3 FDP_ACF.1/Import_TC

FDP_ACF.1.1/Import_TC	The TSF shall enforce the PrivateKey Import TC SFP to objects based on the following: • Subject: S.ImportComponent • Object: O.PrivateKey • Security attributes: [none][1].
FDP_ACF.1.2/Import_TC	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: • S.ImportComponent is allowed to perform operation OP.Import to import O.PrivateKey only after establishment of trusted channel according to FTP_ITC.1/Import_TC with FDP_ITC.1/Import_TC, FDP_UIT.1/Import_TC and FDP_UCT.1/Import_TC; • [none][2].
FDP_ACF.1.3/Import_TC	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: none.
FDP_ACF.1.4/Import_TC	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [none][3]

Note: Refined text of FDP_ACF.1.2/Import_TC, because operation OP.Import was missing.

6.2.3.4 FDP_ITC.1/Import_TC

FDP_ITC.1.1/Import_TC	The TSF shall enforce the PrivateKey Import TC SFP when importing O.PrivateKey ~~user data~~, controlled under the SFP, from outside of the TOE.
FDP_ITC.1.2/Import_TC	The TSF shall ignore any security attributes associated with the O.PrivateKey ~~user data~~ when imported from outside the TOE.
FDP_ITC.1.3/Import_TC	The TSF shall enforce the following rules when importing private key ~~user data~~ controlled under the SFP from outside the TOE*:* [none][4].

6.2.3.5 FDP_UCT.1/Import_TC

FDP_UCT.1.1/Import_TC	The TSF shall enforce the PrivateKey Import TC SFP to receive O.PrivateKey ~~user data~~ in a manner protected from unauthorized disclosure.

[1] [assignment: list of SFP-relevant security attributes or named groups of SFR-relevant security attributes]

[2] [assignment: additional rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

[3] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]

[4] [assignment: additional importation control rules]

6.2.3.6 FDP_UIT/Import_TC

FDP_UIT.1.1/Import_TC	The TSF shall enforce the PrivateKey Import TC SFP to receive private key ~~user data~~ in a manner protected from modification and insertion errors.
FDP_UIT.1.2/Import_TC	The TSF shall be able to determine on receipt of O.PrivateKey ~~user data~~, whether modification or insertion has occurred.

6.2.4 SFRs from Key Derivation Package

6.2.4.1 FCS_CKM.5

FCS_CKM.5.1	The TSF shall support derivation of cryptographic keys ECC private key from an initial ECC private key in accordance with a specified cryptographic key derivation algorithm private key derivation mechanisms and specified cryptographic key sizes size of the initial ECC private key that meet the following: [IEEE 1609.2.1] chapter "9.3 Butterfly key mechanism".

Note: *The TSF supports the Butterfly key Derivation and the Implict Certificate generation by providing the mathematical functions* $prk_{der} = (a * prk) + b\ (mod\ p)$ *or* $prk_{der} = (a + prk) * b\ (mod\ p)$, *where prk is some private key, p is the prime of the elliptic curve field and a and b are some arbitrary parameters provided by the VCS .* This SFR supports key sizes of 256 and 384 bits.

6.2.5 SFRs from Software Update Package

6.2.5.1 FCS_COP.1/SWU

FCS_COP.1.1/SWU	The TSF shall perform software update signature verification in accordance with a specified cryptographic algorithm [ECDSA verify][1]and cryptographic key sizes [521 bit] that meet the following: [FIPS 186-4][2].

[1] [assignment: additional importation control rules]

[2] [assignment: list of standards]

6.2.5.2 FDP_ITC.2/SWU

FDP_ITC.2.1/SWU	The TSF shall enforce the HSM SW Update SFP when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2/SWU	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3/SWU	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.
FDP_ITC.2.4/SWU	The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.
FDP_ITC.2.5/SWU	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: • O.ImageUpdate shall be imported with proof of authenticity and version number.

Note: In this SFR, the security attributes associated with the imported data are software update signature and software update version, see FDP_ACF.1/SWU below.

6.2.5.3 FDP_ACC.1/SWU

FDP_ACC.1.1/SWU	The TSF shall enforce the HSM SW Update SFP on • Subject: S.SWU • Object: O.ImageUpdate • Operation: OP.SWU

6.2.5.4 FDP_ACF.1/SWU

sFDP_ACF.1.1/SWU	The TSF shall enforce the HSM SW Update SFP to objects based on the following: • Subject: S.SWU • Object: O.ImageUpdate • Security attributes: New Version, Software Update Signature, Current Version.
FDP_ACF.1.2/SWU	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: • S.SWU is allowed to perform OP.SWU, i.e. to import O.ImageUpdate according to FDP_ITC.2/SWU, if – the Software Update Signature over O.ImageUpdate and New Version is successfully verified according to FCS_COP.1.1/SWU, and – New Version is greater than Current Version[1].
FDP_ACF.1.3/SWU	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: none.
FDP_ACF.1.4/SWU	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: • [none].

6.2.5.5 FPT_TDC.1/SWU

FPT_TDC.1.1/SWU	The TSF shall provide the capability to consistently interpret security attribute New Version when shared between the TSF and another trusted IT product.
FPT_TDC.1.2/SWU	The TSF shall use the following rules: the New Version must be identified when interpreting the TSF data from another trusted IT product.

6.2.6 SFRs for Genuiness Proof

The following SFR has been added to this ST, because the TOE provides a genuiness proof.

6.2.6.1 FIA_API.1

FIA_API.1.1	The TSF shall provide a [ECDSA Challenge-Response protocol with NIST P521][2] to prove the genuity of the TOE[3] to an external entity.

[1] refinement of V2X HSM PP: “New Version is greater than or equal than Current Version”

[2] [assignment: authentication mechanism]

[3] [selection: TOE, [assignment: object, authorized user or role]]

6.3 Security Assurance Requirements

The security assurance requirements according to Table 18 have been chosen. They comprise EAL4 augmented by AVA_VAN.4 and ALC_FLR.1 (marked as bold text in Table 18).

Table 18 Security Assurance Requirements

Assurance Class	Assurance Component Name	Component
ADV: Development	Security architecture description	ADV_ARC.1
	Complete functional specification	ADV_FSP.4
	Implementation representation of the TSF	ADV_IMP.1
	Basic modular design	ADV_TDS.3
AGD: Guidance documents	Operational user guidance	AGD_OPE.1
	Preparative procedures	AGD_PRE.1 (refined)
ALC: Life-cycle support	Production support, acceptance procedures and automation	ALC_CMC.4
	Problem tracking CM coverage	ALC_CMS.4
	Delivery procedures	ALC_DEL.1
	Identification of security measures	ALC_DVS.1
	Basic flaw remediation	**ALC_FLR.1**
	Developer defined life-cycle model	ALC_LCD.1
	Well-defined development tools	ALC_TAT.1
ASE: Security Target evaluation	Conformance claims	ASE_CCL.1
	Extended components definition	ASE_ECD.1
	ST introduction	ASE_INT.1
	Security objectives	ASE_OBJ.2
	Derived security requirements	ASE_REQ.2
	Security problem definition	ASE_SPD.1
	TOE summary specification	ASE_TSS.1
ATE: Tests	Analysis of coverage	ATE_COV.2
	Testing: basic design	ATE_DPT.1
	Functional testing	ATE_FUN.1
	Independent testing - sample	ATE_IND.2
AVA: Vulnerability assessment	**Methodical vulnerability analysis**	**AVA_VAN.4**

6.3.1 Refinements of the TOE Assurance Requirements

The following refinements shall support the comparability of evaluations according to this Protection Profile.

6.3.1.1 Refinements Regarding Preparative Procedures, AGD_PRE.1

The following text states the requirements of the selected component AGD_PRE.1:

Security Requirements

Developer action elements:

AGD_PRE.1.1D	The developer shall provide the TOE including its preparative procedures.

Content and presentation elements:

AGD_PRE.1.1C	The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
AGD_PRE.1.2C	The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST. Refinement: The preparative procedures shall describe all necessary measures for integration with the VCS to guarantee the confidentiality, integrity and authenticity of the TOE assets according to OE.INTEGRATION.

Evaluator action elements:

AGD_PRE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AGD_PRE.1.2E	The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

6.4 Security Requirements Rationale

6.4.1 Security Functional Requirements Dependencies

Table 19 SFR dependencies

Requirement	Direct explicit dependencies	Dependencies met by	Comment
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1] and FCS_CKM.4	FCS_COP.1/ECDSA FCS_COP.1/ECIES_ENC FCS_COP.1/ECIES_DEC FCS_CKM.4	
FCS_CKM.4	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1]	FCS_CKM.1	
FCS_CKM.5	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_COP.1 FCS_CKM.4	
FCS_RNG.1	None	---	
FCS_COP.1 /ECDSA	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1, FCS_CKM.4, FCS_CKM.5	FCS_CKM.1 is only used for 256 and 384 bit keys. The 521 bit key for FIA_API is personalized during production.

Security Requirements

Requirement	Direct explicit dependencies	Dependencies met by	Comment
			FCS_CKM.5 is used because keys can also be generated by the Butterfly key derivation.
FCS_COP.1 /ECIES_ENC	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1, FCS_CKM.4, FCS_CKM.5	FCS_CKM.5 is used because keys can also be generated by the Butterfly key derivation.
FCS_COP.1 /ECIES_DEC	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1, FCS_CKM.4 FCS_CKM.5	FCS_CKM.5 is used because keys can also be generated by the Butterfly key derivation.
FDP_RIP.1	None	---	
FDP_SDI.2	None	---	
FPT_FLS.1	None	---	
FPT_PHP.3	None	---	
FPT_TST.1	None	---	
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1	
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1 FMT_MSA.3	
FDP_UIT.1/ACP	[FDP_ACC.1 or FDP_IFC.1] [FTP_ITC.1 or FTP_TRP.1]	FDP_ACC.1 FTP_ITC.1/ACP	
FDP_UCT.1/ACP	[FDP_ACC.1 or FDP_IFC.1] [FTP_ITC.1 or FTP_TRP.1]	FDP_ACC.1 FTP_ITC.1/ACP	
FMT_SMR.1	FIA_UID.1	FIA_UID.1	
FMT_SMF.1	None	None	
FMT_MTD.1	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1	
FIA_UID.1	---	None	
FIA_UAU.1	FIA_UID.1	FIA_UID.1	
FTP_ITC.1/ACP	---		
FTP_ITC.1 /Import_TC	---		None
FDP_ACC.1 /Import_TC	FDP_ACF.1	FDP_ACF.1/Import_TC	
FDP_ACF.1 /Import_TC	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Import_TC Not applicable	There is no security attributes related to import defined in this ST, hence FMT_MSA.3 is not needed.
FDP_ITC.1 /Import_TC	[FDP_ACC.1 or FDP_IFC.1] FMT_MSA.3	FDP_ACC.1/Import_TC Not applicable	There is no security attributes related to import defined in this ST, hence FMT_MSA.3 is not needed.

Security Requirements

Requirement	Direct explicit dependencies	Dependencies met by	Comment
FDP_UCT.1 /Import_TC	[FDP_ACC.1 or FDP_IFC.1] [FTP_ITC.1 or FTP_TRP.1]	FDP_ACC.1/Import_TC FTP_ITC.1/Import_TC	
FDP_UIT.1 /Import_TC	[FDP_ACC.1 or FDP_IFC.1] [FTP_ITC.1 or FTP_TRP.1]	FDP_ACC.1/Import_TC FTP_ITC.1/Import_TC	
FCS_COP.1/SWU	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FDP_ITC.2/SWU FCS_CKM.4	The 521 bit ECC key for ECDSA verification is personalized during production.
FDP_ITC.2/SWU	[FDP_ACC.1 or FDP_IFC.1] [FTP_ITC.1 or FTP_TRP.1] FPT_TDC.1	FDP_ACC.1/SWU -- FPT_TDC.1/SWU	FTP_ITC.1 or FTP_TRP is not necessary, as the integrity and authenticity of the update code is protected by the signature per FDP_ACF.1/SWU.
FDP_ACC.1/SWU	FDP_ACF.1	FDP_ACF.1/SWU	
FDP_ACF.1/SWU	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/SWU Not Applicable	For the software update there is no creation of objects.
FPT_TDC.1/SWU	-	-	
FIA_API	-	FCS_COP.1/ECDSA	ECDSA used as challenge-response protocol
FCS_CKM.5/AES	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_CKM.4	Derived keys will be used to derive the static SCP03 trusted channel keys (FTP_ITC.1/ACP). Listing cryptographic operations for a trusted channel is not formally required by CC and is therefore not explicitly mentioned in this ST. Therefore FCS_CKM.2 or FCS_COP.1 are not required.
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 FMT_SMR.1	
FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1 FMT_SMR.1 FMT_SMF.1	

6.4.2 Security Functional Requirements Coverage

Table 20 Security Functional Requirements Coverage

Security Requirements

	OT.PRIVKEY_ACCESS	OT.SIGNATURE_GENERATION	OT.KEY_MANAGEMENT	OT.ECIES	OT.TOE_SELF-PROTECTION	OT.RNG	OT.VCS_DATA	OT.ACCESS_CONTROL	OT.AUTHENICATION	OT.TRUSTED_CHANNEL	OT.PRIVKEY_IMPORT_TC	OT.SW_UPDATE	OT.KEY_DERIVE	OT.GENUINITY
FCS_CKM.1			X											
FCS_CKM.4			X											
FCS_CKM.5													X	
FCS_RNG.1		X	X			X								
FCS_COP.1/ECDSA		X												
FCS_COP.1/ECIES_ENC				X										
FCS_COP.1/ECIES_DEC				X										
FDP_RIP.1			X											
FDP_SDI.2			X				X							
FDP_ACC.1	X							X						
FDP_ACF.1	X							X						
FPT_FLS.1					X									
FPT_PHP.3					X		X							
FPT_TST.1					X									
FDP_UIT.1/ACP										X				
FDP_UCT.1/ACP										X				
FMT_SMR.1								X						
FMT_SMF.1									X	X				
FMT_MTD.1									X	X				
FIA_UID.1									X					
FIA_UAU.1									X					
FTP_ITC.1/ACP										X				
FTP_ITC.1/Import_TC									X					
FDP_ACC.1/Import_TC									X					
FDP_ACF.1/Import_TC									X					
FDP_ITC.1/Import_TC											X			
FDP_UCT.1/Import_TC									X					
FDP_UIT.1/Import_TC									X					
FCS_COP.1/SWU												X		
FDP_ITC.2/SWU												X		
FPT_TDC.1/SWU												X		
FDP_ACC.1/SWU												X		
FDP_ACF.1/SWU												X		

Security Requirements

	OT.PRIVKEY_ACCESS	OT.SIGNATURE_GENERATION	OT.KEY_MANAGEMENT	OT.ECIES	OT.TOE_SELF-PROTECTION	OT.RNG	OT.VCS_DATA	OT.ACCESS_CONTROL	OT.AUTHENICATION	OT.TRUSTED_CHANNEL	OT.PRIVKEY_IMPORT_TC	OT.SW_UPDATE	OT.KEY_DERIVE	OT.GENUINITY
FIA_API.1														X
FCS_CKM.5/AES									X					
FMT_MSA.3								X						
FMT_MSA.1								X						

6.4.3 Security Functional Requirements Sufficiency

Table 21 Security Functional Requirements Sufficiency

Objective	SFR	Rationale
OT.PRIVKEY_ACCESS	FDP_ACC.1, FDP_ACF.1	The TOE shall protect private key assets (FDP_ACC.1, FDP_ACF.1).
OT.SIGNATURE_GENERATION	FCS_RNG.1, FCS_COP.1/ECDSA	Signature generation is performed using ECDSA (FCS_RNG, and FCS_COP.1/ECDSA).
OT.KEY_MANAGEMENT	FCS_CKM.1, FCS_CKM.4, FCS_RNG.1, FDP_RIP.1, FDP_SDI.2	The TOE shall be able to generate ECC asymmetric key pairs (FCS_CKM.1) using RNG (FCS_RNG.1). The TOE shall be able to destroy key and key material (FCS_CKM.4, FDP_RIP.1). The TOE shall protect the integrity of these keys during the storage (FDP_SDI.2). Note: Confidentiality is covered by OT.PRIVKEY_ACCESS.
OT.ECIES	FCS_COP.1/ECIES_ENC, FCS_COP.1/ECIES_DEC	The TOE shall be able to manage the ECIES operations (FCS_COP.1/ECIES_ENC and FCS_COP.1/ECIES_DEC) Note: Internal ECC key creation is covered by OT.KEY_MANAGEMENT.
OT.TOE_SELF-PROTECTION	FPT_FLS.1, FPT_PHP.3, FPT_TST.1	The TOE for its self-protection shall detect and react failures (FPT_TST.1) and preserve the secure state (FPT_FLS.1), as well as the resistance against tampering (FPT_PHP.3).

Security Requirements

Objective	SFR	Rationale
OT.RNG	FCS_RNG.1	The TOE shall implement secure RNG.
OT.VCS_DATA	FDP_SDI.1, FPT_PHP.3	The TOE shall guarantee the integrity of the stored data (FDP_SDI.1) and their confidentiality through resistance to tampering attacks (FPT_PHP.3)
OT.ACCESS_CONTROL	FDP_ACC.1, FDP_ACF.1, FMT_SMR.1 FMT_MSA.1, FMT_MSA.3	OT.ACCESS_CONTROL is addressed by the implementation of FDP_ACC.1 and FDP_ACF.1; related roles on which OT.ACCESS_CONTROL is based are handled by FMT_SMR.1. Management of security is handled by FMT_MSA.1 and FMT_MSA.3.
OT.AUTHENTICATION	FIA_UID.1, FIA_UAU.1, FMT_SMF.1, FMT_MTD.1	OT.AUTHENTICATION is addressed by the implementation of FIA_UID.1 and FIA_UAU.1; controlled management of Trusted channel keys including authentication data is addressed by FMT_SMF.1 and FMT_MTD.1.
OT.TRUSTED_CHANNEL	FDP_ITC.1, FDP_UIT.1/ACP, FDP_UCT.1/ACP, FMT_SMF.1, FMT_MTD.1, FTP_ITC.1/Import_TC, FDP_UIT.1/Import_TC, FDP_UCT.1/Import_TC, FDP_ACC.1/Import_TC, FDP_ACF.1/Import_TC FCS_CKM.5/AES	OT.TRUSTED_CHANNEL is addressed by the implementation of FDP_ITC.1; the details of transfer protections are defined in FDP_UIT.1/ACP and FDP_UCT.1/ACP; handling of received information is defined in FDP_ITC.1; controlled management of Trusted channel keys including authentication data is addressed by FMT_SMF.1 and FMT_MTD.1. For private key (online) import OT.TRUSTED_CHANNEL is addressed by the implementation of FTP_ITC.1/Import_TC; the details of transfer protections are defined in FDP_UIT.1/Import_TC (integrity protection), FDP_UCT.1/Import_TC (confidentiality protection), FDP_ACC.1/Import_TC and FDP_ACF.1/Import_TC (authenticity protection by requiring trusted channel establishment before key import is allowed). FCS_CKM.5/AES perform initial key derivation of all keys necessary to establish a trusted channel.
OT.PRIVKEY_IMPORT_TC	FDP_ITC.1/Import_TC	OT.PRIVKEY_IMPORT_TC is addressed by the implementation of FDP_ITC.1/Import_TC.
OT.KEY_DERIVE	FCS_CKM.5	OT.KEY_DERIVE is addressed by FCS_CKM.5, which requires the implementation of a derivation algorithm.

Security Requirements

Objective	SFR	Rationale
OT. SW_UPDATE	FCS_COP.1/SWU, FDP_ITC.2/SWU, FPT_TDC.1/SWU, FDP_ACC.1/SWU, FDP_ACF.1/SWU	OT. SW_UPDATE is addressed by the implementation of FCS_COP.1/SWU for authenticity verification; FDP_ITC.2/SWU, FPT_TDC.1/SWU, FDP_ACC.1/SWU, FDP_ACF.1/SWU, for handling of image reception.
OT.GENUINITY	FIA_API.1	OT.GENUINITY directly maps to FIA_API.1, which provides ECDSA as a genuinity proof.

6.4.4 Security Assurance Dependencies Analysis

The chosen evaluation assurance level EAL4 augmented by ALC_FLR.1 and AVA_VAN.4. Since all dependencies are met internally by the EAL package only the augmented assurance components dependencies are analysed.

Table 22 Security Assurance Dependencies Analysis

Assurance Component	Dependencies	Met
ALC_FLR.1	None	Yes
AVA_VAN.4	ADV_ARC.1 Security architecture description	Yes
	ADV_FSP.4 Complete functional specification	Yes
	ADV_TDS.3 Basic modular design	Yes
	ADV_IMP.1 Implementation representation of the TSF	Yes
	AGD_OPE.1 Operational user guidance	Yes
	AGD_PRE.1 Preparative procedures	Yes
	ATE_DPT.1 Testing: basic design	Yes

According to Table 22 all dependencies are met.

6.4.5 Justification of the Chosen Evaluation Assurance Level

The assurance level EAL4 augmented with AVA_VAN.4 and ALC_FLR.1 has been chosen as appropriate for a Secure Hardware Module resisting threat agents possessing a Moderate attack potential.

7 TOE Summary Specification

7.1 Cryptographic Services

The TSF provides the following cryptographic functions as APDU commands

- FCS_CKM.1: The TSF uses the ECC_KEYGEN function to generate private ECDSA keys for NIST and Brainpool curves with 256 and 384 bits according to [FIPS 186-4]. The private key will be stored in the key store (7.2) in a key slot number which is given as a parameter to the ECC_KEYGEN function. The public key will not be stored in the key slot. Instead it will be generated from this private key on demand by the ECC_GETPUBLIC function.
- FCS_CKM.4: The TSF ECC_DELETEPRIVATE deletes a specific ECDSA or ECIES private key. All keys can be deleted in one batch by the ZEROIZE function.
- FCS_CKM.5: The ECC_MODMULADD function is used to support Butterfly key generation and implicit certificate generation according to [IEEE 1609.2.1]. This function gets a private key from a source key slot, derives a new private key by performing the mathematical function as described in the Note in ch. 6.2.4.1 and stores the derived key in a destination key slot. The key type can be either a signature for an encryption key.
- FCS_CKM.5/AES: When first used, the TOE derives SCP03 trusted channel keys (key diversification data). This is typically done during integration phase. Afterwards, this specific functionality is no longer available.
- FCS_RNG.1: The GET_RANDOM function provides high quality random numbers to the VCS. The physical entropy source is from the underlying platform according to PTG.2. The entropy source will be used as seed for an NIST DRBG according to [SP800-90A]. This DRBG is certified together with the underlying platform.
- FCS_COP.1/ECDSA, FCS_COP.1/ECIES_ENC, FCS_COP.1/ECIES_DEC: The TSF provides commands to perform the ECDSA and ECIES_DEC cryptographic services. The commands get an index to a private key stored in the Key store. ECDSA can only be used with keys which have the signature type and ECIES_DEC can only be used with keys which have the decryption flag.
  The ECIES_ENC is an encryption function and does not use a long term private key from the key store.

7.2 Key Store

The TSF uses a secure key store which stores the generated or imported ECC keys. Each key slot has pre-assigned roles for the security attributes “write”, “read”, “delete” and “change_access”. By storing a key in a key slot, the security attribute values “read”, “delete” and “change_access” will be assigned to that key.

- FDP_RIP.1: When a private key is deleted by the ECC_DELETEPRIVATE or ZEROIZE service then the keyslot is marked as empty and the key value will be overwritten with zeros.
- FDP_SDI.2: Each key in a key slot is stored in NVM storage of the underlying platform. This NVM is protected by an EDC (see [ST_ICC]). Additionally, the TSF uses a 64 bit checksum to protect the keys in the store.
- FDP_ACC.1/FDP_ACF.1: The TSF enforces that only a user authenticated with the matching “write” role can store a key in that slot. Only a user with the matching role “read” can use the private kes for a TSF function.
- FMT_MSA.1/FMT_MSA.3: The default access rights for the “change_access” attribute in each slot is set to R.ADMIN. That means that only R.ADMIN can change the security attributes.

7.3 Physical Protection

- FPT_PHP.3: The TSF uses the physical security mechanism from the underlying platform (see [ST_ICC]). In particular, it uses the Memory Protection Unit form the underlying platform to catch misaligned memoryaccesses due to fault attacks.
- FPT_FLS.1: The TSF perform a security reset when the self tests are failing or when the checksum calculation of the key is wrong.
- FPT_TST.3: The TSF performs self tests after each reset. The self tests are also accessible as a security service and can be triggered manually by the VCS. In addition it calls the hardware test functions of the underlying platform (see [ST_ICC]).

7.4 In Field Update

The TSF provides a field upgrade function to load a new V2X firmware to the chip. The role R.ADMIN can switch from the normal V2X operational mode to a V2X firmware upgrade loader (VFUL) mode by using the SET_LIFECYCLE function. After successful or abandoned firmware upgrade the TOE switches back to V2X operational mode.

FCS_COP.1/SWU: The firmware image is integrity protected by an ECDSA checksum, which will be checked by the TSF before the load process starts.

FDP_ITC.2/SWU: The TSF checks that the version number of the new image is strictly greater than the current version number.

FDP_ACC.1/SWU and FDP_ACF.1/SWU: The TSF ensures that only the role R.ADMIN can call the SET_LIFECYCLE function to switch to VFUL mode.

7.5 Trusted Channel

The TOE implements the GlobalPlatform Secure Channel Protocol 03 (SCP03) specified by GlobalPlatform Amendment D version 1.1.2 [SCP03].

The TOE uses a separate set (MAC/ENC/DEK keys) of symmetric SCP03 keys for role R.ADIM and R.USER. Opening a trusted channel with the respective MAC keys defines the current role. Only one role can be present at one time.

- FDP_UCT.1/ACP: Confidentiality protection of the trusted channel is enforced for all commands which send or receive confidential VCS data.
- FDP_UIT.1/ACP: Integrity protection of the trusted channel is enforced for all commands which send or receive authenticated VCS data.
- FTP_ITC.1/ACP: The TOE uses the GlobalPlatform SCP03 protocol
- FMT_SMR.1: The TOE maintains two sets of SCP03 keys. One set for role R.ADMIN and the other set for role R.USER. The User of the TOE selects the corresponding role by starting the SCP03 trusted channel with the respective key set.
- FMT_SMF.1: The values of the SCP03 keys can be changed by the GlobalPlatform PUTKEY command. The security attributes of a private ECC key can be modified by any role defined by the security attribute “change_access” of the private key.
- FMT_MTD.1: The GlobalPlatform keys can only be changed by the PUTKEY command which requires authentication by the resepective role which is assocoiated to the key.
- FIA_UID.1/FIA_UAU.1: The TOE allows services related to self tests to be performed without any authentication.

7.6 Private Key Import

Keys can be imported from outside the TOE and written into a specified key slot. Only authenticated roles (i.e. R.ADMIN or R.USER) are allowed to import keys. The authenticated role must match the role for “write” associated to that key slot.

- FDP_ACC.1/Import_TC) and FDP_ACF.1/Import_TC: The TOE allows private key import only after authentication of R.VCS
- FDP_ITC.1/Import_TC: The TSF performs the import independent of the security attributes associated to the key.
- FPT_ITC.1/Import_TC , FPT_UCT.1/Import_TC, FPT_UIT.1/Import_TC: The TSF uses the trusted channel from 7.5 to protect the integrity and confidentiality of the imported key.

7.7 Genuiness proof

- FIA_API.1: The TSF provides a service which performs an ECDSA signature of a challenge chosen by the verifier a pre-personalized and certified 521 bit ECC key. The signature can be used as a proof that the TOE is genuine.

8 Statement of Compatibility

The following tables fulfills the ASE_COMP.1.1D requirement form “Composite product evaluation Appendix 1.1: CC v3.1 for Smart Cards and similar devices” [CC_COMP]

Table 23 Platform Objectives for the TOE

Objective	relevant	Rationale for compatibility
O.Phys-Manipulation	yes	TOE software fulfills platform guidance
O.Phys-Probing	yes	TOE software fulfills platform guidance
O.Malfunction	yes	TOE software fulfills platform guidance
O.Leak-Inherent	yes	TOE software fulfills platform guidance
O.Leak-Forced	yes	TOE software fulfills platform guidance
O.Abuse-Func	no	Deactivated before TOE delivery to composite integrator
O.Identification	yes	Will be used by TOE Identification and cannot be changed. Part of the TOE identification data of this composite TOE will be taken form the TOE identification of the hardware platform.
O.RND	yes	Is only indirectly used by DRBG function of O.Add-Functions
O.Cap_Avail_Loader	no	Flash loader is deactivated
O.Authentication	no	Flash loader is deactivated
O.Ctrl_Auth_Loader	no	Flash loader is deactivated
O.TDES	no	Is not used in this TOE
O.AES	yes	TOE software fulfills guidance for symmetric crypto library
O.Add-Functions	yes	TOE software fulfills guidance for cryptographic library for DGRB, ECDSA and Hash functions
O.Mem Access	yes	TOE software fulfills platform guidance
O.Prot_TSF_Confidentiality	yes	TOE software fulfills platform guidance
O.Data_IntegrityService	yes	TOE software fulfills guidance for hash library

Table 24 Platform Objectives for the Operational Environment

Objective	classification	Rationale for classification
OE.Lim_Block_Loader	IrOE	Flashloader is deactivated
OE.TOE_Auth	IrOE	Flashloader is deactivated
OE.Loader_Usage	IrOE	Flashloader is deactivated
OE.Resp-Appl	CfPOE	Fulfilled by: OT.KEY_MANAGEMENT OT.TOE_SELF-PROTECTION OT.VCS_DATA
OE.Process-Sec-IC	CfPOE	Part of ALC class of this CC certification

Statement of Compatibility

Table 25 Platform SFR classification

SFR	class	rationale
FAU_SAS.1	IP_SFR	Testmode: Used in manufacturing phase only
FMT_LIM.1	IP_SFR	Testmode: Used in manufacturing phase only
FMT_LIM.2	IP_SFR	Testmode: Used in manufacturing phase only
FDP_ACC.1	RP_SFR-MECH	MPU: Used as countermeasure against fault attacks
FDP_ACF.1	RP_SFR-MECH	MPU: Used as countermeasure against fault attacks
FTP_ITC.1	IP_SFR	Flashloader: Used in manufacturing phase only
FDP_UCT.1	IP_SFR	Flashloader: Used in manufacturing phase only
FIA_API.1	IP_SFR	Flashloader: Used in manufacturing phase only
FMT_LIM.1/Loader	IP_SFR	Flashloader: Used in manufacturing phase only
FMT_LIM.2/Loader	IP_SFR	Flashloader: Used in manufacturing phase only
FDP_UIT.1	IP_SFR	Flashloader: Used in manufacturing phase only
FDP_ACC.1/Loader	IP_SFR	Flashloader: Used in manufacturing phase only
FDP_ACF.1/Loader	IP_SFR	Flashloader: Used in manufacturing phase only
FPT_PHP.3	RP_SFR-MECH	Used by FPT_PHP.3 in this TOE
FDP_ITT.1	RP_SFR-MECH	Implicitly used as countermeasure against side channel leakage
FPT_ITT.1	RP_SFR-MECH	Implicitly used as countermeasure against side channel leakage
FDP_SDC.1	RP_SFR-MECH	Implicitly used as countermeasure against side channel leakage
FDP_SDI.2	RP_SFR-MECH	Implicitly used as countermeasure against fault attacks
FDP_IFC.1	RP_SFR-MECH	Implicitly used as countermeasure against fault attacks and side channel leakage
FMT_MSA.1	RP_SFR-MECH	MPU: Used as countermeasure against fault attacks
FMT_MSA.3	RP_SFR-MECH	MPU: Used as countermeasure against fault attacks
FMT_SMF.1	RP_SFR-MECH	MPU: Used as countermeasure against fault attacks
FRU_FLT.2	RP_SFR-MECH	Implicitly used as countermeasure against fault attacks
FPT_TST.2	RP_SFR-SERV	UMSLC test: Used by FPT_TST.1 in this TOE
FPT_FLS.1	RP_SFR-MECH	Implicitly used as countermeasure against fault attacks
FCS_RNG.1/TRNG	RP_SFR-SERV	Used as entropy source for the FCS_RNG.1/DRBG
FCS_RNG.1/HPRG	IP_SFR	Not used in this TOE
FCS_RNG.1/DRNG	IP_SFR	Not used in this TOE
FCS_RNG.1/KSG	IP_SFR	Not used in this TOE
FCS_RNG.1/DRBG	RP_SFR-SERV	Random: Used by FCS_RNG.1 in this TOE
FCS_COP.1/TDES	IP_SFR	Not used in this TOE

Statement of Compatibility

SFR	class	rationale
FCS_COP.1/TDSCL	IP_SFR	Not used in this TOE
FCS_COP.1/AES	RP_SFR-SERV	Used by FCS_COP.1/AESCL in the SCL
FCS_COP.1/AESCL	RP_SFR-SERV	Used by the SCP03 trusted channel protocol and in this TOE
FCS_COP.1/RSA	IP_SFR	Not used in this TOE
FCS_COP.1/ECDSA	RP_SFR-SERV	Used by FCS_COP.1/ECDSA and FIA_API.1 in this TOE
FCS_COP.1/ECDH	RP_SFR-SERV	Used by FCS_COP.1/ECIES_ENC and FCS_COP.1/ECIES_DEC in this TOE
FCS_COP.1/HCL	RP_SFR-SERV	Used by FCS_COP.1/ECIES_ENC, FCS_COP.1/ECIES_DEC and SCP03 key derivation in this TOE
FCS_CKM.1/RSA	IP_SFR	Not used in this TOE
FCS_CKM.1/EC	RP_SFR-SERV	Used by FCS_CKM.1 and FCS_CKM.5 in this TOE
FCS_CKM.4/TDES	IP_SFR	Not used in this TOE
FCS_CKM.4/AES	RP_SFR-SERV	Used by SCP03 when the session is terminated

9 Glossary and Acronyms

Acronym or Abbreviation	Explanation
ACP	Additional Communication Protections
AT	Authorization Ticket, a.k.a. Pseudonym Certificate (PC)
C2C-CC	Car2Car Communications Consortium
CA	Certification Authority
EAL	Evaluation Assurance Level
EC	Enrolment Credentials, a.k.a. Long-Term Certificate (LTC)
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ECIES	Elliptic Curve Integrated Encryption Scheme
EDC	Eooro Detetction Code
FIPS	Federal Information Processing Standard
HSM	Hardware Security Module
ITS	Intelligent Transport System
ITS-S	Intelligent Transport System – Station
C-ITS	Cooperative Intelligent Transport System
IC	Integrated Circuit
Import_AE	Import with Authentication and Encryption used in Private Key Import (offline)
Import_TC	Import using Trusted Channel used in Private Key Import (online)
IVN	In Vehicle Network
NIST	National Institute of Standards and Technology
OSP	Organisational Security Policy
PP	Protection Profile
RFC	Request For Comments
SFR	Security Functional Requirement
ST	Security Target
TOE	Target Of Evaluation
TSF	TOE Security Functionality
V2X	Vehicle to anything
VCS	Vehicle C-ITS Station
VFUL	V2X Field Upgrade Loader

10 References

Symbol	Version	Title
[ST_ICC]	1.5 / 2021-04-14	Confidential Security Target IFX_CCI_00003Fh IFX_CCI_000059h IFX_CCI_00005Bh IFX_CCI_00003Ch IFX_CCI_00003Dh IFX_CCI_00005Ah G11
[V2X HSM PP]	1.0.1	Protection Profile V2X Hardware Security Module, v1.0.1, Release 1.6.0, 2021-11-30, CAR 2 CAR Communication Consortium
[TS 103 097]	1.3.1	Intelligent Transport Systems (ITS); Security; Security Header and Certificate Formats
[IEEE 1609.2]	2016 amended by 2017	IEEE Std 1609.2™ Standard for Wireless Access in Vehicular Environments -- Security Services for Applications and Management Messages
[IEEE 1609.2.1]	D3, August 2019	IEEE Std 1609.2™ Draft Standard for Wireless Access in Vehicular Environments (WAVE) -- Certificate Management Interfaces for End-Entities
[FIPS 186-4]	July 2013	FIPS publication Digital Signature Standard (DSS)
[SEC-1]	Version 2.0 May 21 2009	Standards for Efficient Cryptography Group, “SEC 1: Elliptic Curve Cryptography”
[IEEE 1363a]	2004	IEEE Standard Specifications for Public-Key Cryptography - Amendment 1: Additional Techniques
[RFC 5639]	March 2010	Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation
[CCp1]	3.1, rev 5	Common Criteria for Information Technology Security Systems, Part 1: Introduction and general model
[CCp2]	3.1, rev 5	Common Criteria for Information Technology Security Systems, Part 2: Security functional requirements
[CCp3]	3.1, rev 5	Common Criteria for Information Technology Security Systems, Part 3: Security assurance requirements
[CC_COMP]	v 1.5.1	Composite product evaluation for Smart Cards and similar devices
[FIPS 186-4]	July 2014	FIPS 186-4 Digital Signature Standard (DSS)
[SP800-90A]	June 2015	SP 800-90A Rev. 1 Recommendation for Random Number Generation Using Deterministic Random Bit Generators
[SCP03]	1.1.2	GlobalPlatform Technology Secure Channel Protocol '03' Card Specification v2.3 – Amendment D Version 1.1.2
[SP800-38B]	June 2016	SP 800-38B

		Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication
[SP800-108]	October 2009	SP 800-108 Recommendation for Key Derivation Using Pseudorandom Functions (Revised)
[PP0084]	1.0	Security IC Platform Protection Profile with Augmentation Packages

Revision history

Revision	Description
rev. 1.rev. 1.31, 2021-06-24	Initial version
rev. 1.3, 2023-11-13	corrected HSL version number
Rev 1.3, 2023-11-13	New version of Errata Sheet added and Protection Profile Reference updated

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

2023-11-13
Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about this document?
Email:
dsscustomerservice@infineon.com

IMPORTANT NOTICE
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.