S400 on i.MX8ULP

SESIP Security Target

Rev. 1.0 — 29 June 2023

Document Information

Information	Content
Keywords	SESIP, Security Target, Secure MCU, S400 on i.MX8ULP
Abstract	Evaluation of the S400 on i.MX8ULP developed and provided by NXP Semiconductors, according to SESIP Assurance Level 2 (SESIP2), based on SESIP methodology, version 1.1, and PSA L2.

Revision History

Rev.	Date	Description
1.0	29 June 2023	Initial release

1 Introduction

This Security Target describes the core security features provided by the S400 secure subsystem to an SoC integrating it. The S400 subsystem can indeed be integrated in different System-on-Chips (SoCs), in which it will be the Root-of-Trust (RoT) and will act as an Hardware Security Module (HSM), implementing trust-based services for the SoC modules. In particular, OEM applications will be able to use the S400 to ensure their own security.

The target of evaluation is the S400 and is referred as “S400” or “platform” into the rest of this document. The term “application” refers to the rest of the SoC modules, which can invoke the S400 features.

For the current evaluation, the S400 subsystem has been integrated into the i.MX8ULP System-on-Chip (SoC) on which the applications are the A35 and CM33 domains.

The current Security Target is covering the SESIP Secure MCU/MPU profile for the SESIP scheme and the SESIP Profile for PSA Certified™ for the PSA Verified scheme.

1.1 ST Reference

S400 on i.MX8ULP, SESIP Security Target, Revision 1.0, NXP Semiconductors, 29 June 2023.

1.2 SESIP Profile Reference and Conformance Claims

This Security Target claims conformance to the two following SESIP Profiles:

Table 1. SESIP Profile for Secure MCUs and MPUs Conformance Claims

Reference	Value
SP Name	SESIP Profile for Secure MCUs and MPUs [2]
SP Version	Version 1.0
Assurance Claim	SESIP Assurance Level 2 (SESIP2)
Package Claim	Base SP, Package Secure Services, Package Software Isolation of Platform

Table 2. SESIP Profile for PSA Certified Level 2 Conformance Claims

Reference	Value
SP Name	SESIP Profile for PSA Certified Level 2 [3]
SP Version	V1.0
Assurance Claim	SESIP Assurance Level 2 (SESIP2)
Optional and Additional SFRs	Base profile with optional Secure Debugging SFR

1.3 Platform Reference

Table 3. Platform Reference

Reference	Value
Platform Name	S400
Platform Version	S400 ROM: A2 - See components details in Section 3.2.1.1 S400 FW: 0.0.10 - See hash value in Section 3.2.1.1

Table 3. Platform Reference...*continued*

Reference	Value
Platform Identification	S400 on i.MX8ULP
Platform Type	Secure Subsystem on SoC

1.4 Included Guidance Documents

The following documents are included with the platform:

Table 4. Guidance Documents

Document	Reference
SESIP Security Target	S400 on i.MX8ULP, SESIP Security Target, Revision 1.0, NXP Semiconductors, 29 June 2023.
Reference Manual	i.MX 8ULP Processor Reference Manual [5]
Security Reference Manual	i.MX 8ULP Security Reference Manual [6]
Core API Reference Manual	MX8ULPELEAPI EdgeLock Enclave (ELE) API Reference Guide [7]
HSM API Reference Manual	EdgeLock Enclave API Bridge detailed implementation [8]

1.5 Platform Overview and Description

1.5.1 Platform Security Features and scope

The S400 services in the scope of the evaluation are the following:

- Secure unique identification
- Secure initialization of the S400 and SoC components
- Secure Updates of S400 and SoC firmware's
- Signature based Authenticated Debug for SoC domain access
- HSM Crypto Key storage and Operations (AES, RSA, ECC, SHA2, HMAC/CMAC, RNG)
- Isolation of S400 towards rest of the SoC
- Remote Attestation of S400 and SoC components

The S400 subsystem consists of hardware and firmware: the physical scope includes the S400 processing unit, and the logical scope includes the S400 firmware in ROM and the loadable firmware stored in external NVM.

In the current evaluation, the S400 has been integrated into the i.MX8ULP SoC, as represented in the figure below (the S400 subsystem, evaluation scope, is in red square):

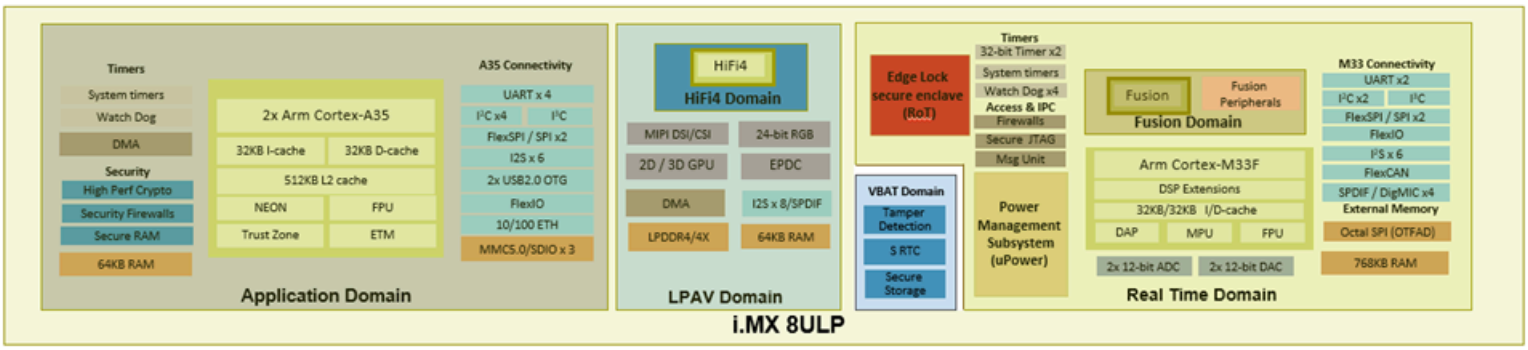

Figure 1. S400 on i.MX8ULP scope

In this SoC, the S400 is integrated into the Real Time domain and is accessible by both the CM33 and A35 (Application domain) cores (LPAV domain is a slave domain only). The platform boundaries include the following hardware components and interfaces:

Table 5. Hardware components and interfaces

Component/interface	Description
CPU	32 bit RISC-V
Communication ports	Message Units, Trust Bus, SoC Bus
Debug port	JTAG
Fuse Status Block	Read of public fuses
Memories	DMEM, IMEM RAMs, PKC RAM, SRAM-PUF, ROM
GPIO Signals/Interrupts	See [6]
Crypto module	Public Key Coprocessor (PKC) Symmetric Crypto Accelerator (SGI)

The platform boundaries include the following software components and interfaces:

Table 6. Software components and interfaces

Component/interface	Description
ROM Firmware	Includes secure boot, secure update, life cycle management, attestation features for S400
Loadable Firmware image (in external NVM)	Includes key management, cryptographic operations, RNG, attestation features for OEM firmware
Firmware APIs	Interfaces to the S400 services from ROM and loadable firmware

1.5.2 Required Non-Platform Hardware/Software/Firmware

The platform is meant to be integrated into an SoC with a Flash memory in which the platform will be able to store its own data.

1.5.3 Life Cycle

The S400 life cycle steps are as follows:

- **NXP Design**: hardware and firmware design of S400, integration into the SoC design; preparation for manufacturing.
- **NXP Manufacturing**: manufacturing of the SoC integrating the S400; the unique identification information is injected.
- **NXP Packaging**: final testing and packaging of the SoC integrating the S400.
  - NXP secrets and root-of-trust related keys are injected; debug access to NXP area is closed, debug access to OEM area remains opened.
- **OEM Manufacturing**: integration of the SoC integrating the S400 into the OEM product.
  - Customer secrets are provisioned and debug access to OEM secure part must be closed.

- **In-field**: usage of the device integrating the SoC and its S400 subsystem. SoC Debug access is protected by ECC authentication, for S400 Debug is closed.
- **Field-return**:
  - The device integrating the SoC and its S400 subsystem is sent back to the OEM; the OEM changes the life-cycle state of the SoC (handled by S400, after OEM authentication) to “OEM Field-Return” and erase its own secrets.
  - The OEM sends back the SoC to NXP who change the life-cycle state of the SoC (handle by S400, after NXP authentication).
- **Destruction**: the destruction of the SoC can be done from any state; this is handled by S400. In this state, all secrets become unavailable by the zeroization of related encryption keys or related fuses.

Each step corresponds to one or several life-cycle states of the SoC, each of these states being associated to restricted to specific security restrictions.

The life-cycle state machine is handled by the S400 subsystem (see LMDA descriptions in section 7.1 of [6]).

1.5.4 Use Case Environments

The S400 is to be part of SoCs which will be integrated into devices requiring a Hardware Root-of-Trust to ensure the security of the final device use. In particular, the i.MX8ULP integrating it is expected to be used in home and general embedded control, wearables, portable healthcare or printing, IoT edge, SOM board solutions, etc.

[Any code] The S400 can be integrated into different SoCs and/or with different cores firmware and software implementation, thanks to the implementation of the secure isolation of the S400 security subsystem against the rest of the SoC modules (see Software Attacker Resistance: Isolation of Platform).

[Trusted users] The S400 does not provide protection for its security features against physical attacks. Therefore, in case of use in which an attacker could have physical access to the i.MX8ULP, the customer shall consider this in their risk analysis and may need to add mitigations at higher layers of the product or the system.

2 Security Objectives for the Operational Environment

2.1 Platform Objectives for the Operational Environment

For the platform to fulfill its security requirements, the operational environment (technical or procedural) must meet the following objectives:

Table 7. Platform Objectives for the Operational Environment

Title	Description	Reference
Platform Acceptance	hen receiving the platform, the user is expected to verify the correct version of all platform components that it depends on, as described in Section 3.2.1.1 of this document.	This document
Key Management	Cryptographic keys and certificates outside of the platform are subject to secure key management procedures.	This document
Trust Provisioning	Any secret to be provisioned into the platform is generated securely (e.g., via a standard compliant HSM) and subject to secure key management procedures. The provisioning process is done in secure sites with physical, logical security and organizational policies in place.	This document
Trusted Users	Actors in charge of TOE management, for instance for signature of firmware update, are trusted.	This document
Secure Boot	The operating system or application code is expected to make use of the AHAB feature as described in guidance manuals (see Section 1.4).	[6] chapter 5
Secure Update	Actors in charge of executing update of the platform firmware or applications are expected to securely initiate the update process. The update image is expected to be properly signed and distributed in secure manner to ensure its confidentiality and authenticity.	This document
Secure use	Users shall ensure secure and correct use of the platform according to guidance listed in Section 1.4. Note that there is only one user role allowing to access all the interfaces of the platform with the same privilege and in a unique mode of operation; also all features are accessible in only one configuration of the platform.	This document
Physical protection	The operational environment must protect the TOE against physical access of attackers as described in Section 1.5.4.	This document

3 Security Requirements and Implementation

3.1 Security Assurance Requirements

The claimed assurance requirements package is: **SESIP Assurance Level 2 (SESIP2)** as defined in Chapter 4 of Security Evaluation Standard for IoT Platforms (SESIP) [1].

3.1.1 Flaw Reporting Procedures (ALC_FLR.2)

In accordance with the requirement for flaw reporting procedures (ALC_FLR.2), the developer has defined the following procedure:

NXP has defined a Product Security Incident Response Process (PSIRP), implemented by a dedicated team (PSIRT). This process provides a publicly available interface (https://nxp.com/psirt), and includes four major steps:

- **Reporting**. The process begins when the PSIRT becomes aware of a potential security vulnerability in an NXP product. The reporter receives an acknowledgment and updates throughout the handling process.
- **Evaluation**. The PSIRT confirms the potential vulnerability, assesses the risk, determines the impact and assigns a processing priority. If the vulnerability is confirmed, the priority determines how the issue is handled throughout the remaining steps in the process.
- **Solution**. Working with PSIRT, the product team develops a solution that mitigates the reported security vulnerability. Solutions will take different forms based on the vulnerability. Because of the nature of NXP products – mostly silicon products where the firmware is in ROM -, very often the solution can only be provided in a next version of the chips and the short-term solution will consist of recommending security measures to be applied in systems using the NXP product.
- **Communication**. As said above, because of the nature of the NXP products, the solution to systems using the affected products often needs to be found in additional countermeasures in those systems. The communication on the vulnerability and solutions will in most cases be done directly towards the affected customers. For previously unknown or unreported issues, NXP will acknowledge the reporter of the issues (unless the reporter requests otherwise).

The platform's secure boot feature is able to verify the authenticity of its loadable firmware part and of the customer code; it also provides an appropriate mechanism for the update of its own loadable firmware and support for the update of the customer code, the update mechanism itself being to be provided by the customer, most likely at the operating system level (not in scope of this evaluation).

3.2 Security Functional Requirements

The platform fulfills the following security functional requirements:

3.2.1 Base SP Security Functional Requirements

3.2.1.1 Verification of Platform Identity

The S400 unique identification information is injected into the S400 subsystem during the SoC manufacturing in NXP sites. This information can be retrieved through the following APIs described in detail in chapters 3.10 and 3.31 of [7]:

- Message *Get Info*: the response fields *Soc_rev*, *Soc_id*, *Fw_hash* and *Sha256 ROM patch* allow identifying the version of the silicon and the loadable firmware. In the current integration into the i.MX8ULP SoC, the expected values are:

Table 8. *Get Info* expected values

Field	Meaning	Expected value
Soc_rev	SoC revision number	0xA200
Soc_id	SoC identity	0x084D
Fw_hash	Firmware hash	ad78132027eb8d47330c7c3c7cb98916bbe81bd81d189 f87b60e13f24faa99e5
Sha 256 ROM patch	ROM patch sha	568d42789315dc20f854ff2fdf96dc4010f3dd151a7922a7c2912612f3fe310b

- Message *Get FW version*: the response fields *FW version* and *Commit SHA1* are identifying the firmware. In the current integration into the i.MX8ULP SoC, the expected values are:

Table 9. *Get FW version* expected values

Field	Expected value
FW version	0x0800000a
Commit SHA1	0x187e4177

3.2.1.2 Secure Initialization of Platform

The platform ensures its authenticity and integrity during the platform initialization. If the platform authenticity or integrity cannot be ensured, the platform will go to *abort mode or failure state*.

Conformance rationale:

Secure initialization (authenticity and integrity checks) of the S400, A35 and CM33 domains is ensured by the Advance High Assurance Boot (AHAB) feature implemented in the S400.

As part of this process, the authenticity and integrity of firmware to be run on S400 is checked by the S400 ROM based on a signature verification with NXP dedicated ECDSA P256 bits key and SHA-256. The other domains firmware are also checked by the S400 ROM or FW based on an OEM dedicated asymmetric key that can be ECDSA P256/384/521 bits or RSA 2048/3072/4096 bits.

Hashes SHA 256bits of asymmetric public keys are securely handled in S400 fuses (public keys are in firmware image container headers), initially generated in NXP HSMs.

Note that S400 firmware are always encrypted while this is optional for other domains firmware. Encryption is done with an AES-256 bits key, and decryption is handled withing the S400. Decryption keys are securely handled by the S400, derived by secrets in the S400 ROM and fuses.

For secure initialization purpose, the S400 is also in charge of all initial secure configuration of other SoC domains according to the life-cycle state; in particular:

- the domain controllers (RDCs) which set access policies for their domain including access to data and resources;
- the debug and test interfaces access.

In case of a general failure or firmware authentication during secure boot process, e.g. driver failure, secure disabling and cleaning of security settings and memory are performed to reach an abort mode, a failure state entering an endless loop in which accessible services are restricted and resetting after a timeout. After the warm reset, the access to the firmware authentication is restricted by a growing time delay, and to many attempt will block the boot; only a hard reset it possible.

3.2.1.3 Secure Update of Platform

The platform can be updated to a newer version in the field such that the integrity, authenticity and confidentiality of the platform is maintained.

Conformance rationale:

The authenticity and integrity of the updated firmware image is check during the secure boot process as described in Section 3.2.1.2. To protect against rollback of firmware, those are associated to a version number and the S400 manage a “minimum allowed firmware version” in fuses.

Regarding ROM update, the S400 ROM can be patched, and the handling of those patches is fully handled by the S400. ROM patches can be part of OTP (called early patches) or be part of the S400 loadable firmware (called late patches).

In the first case, OTP patches can only be done during NXP manufacturing (through ECDSA P256 signed messages).

In the second case, the patches are part of the S400 loadable firmware and their authenticity and integrity is checked along with the overall firmware verification. Note that late patches loading feature is disabled by default and can only be enable by an early patch.

3.2.1.4 Secure Debugging

The platform only provides *JTAG* authenticated as specified in *[10]* with debug functionality.

The platform ensures that all data stored by the application, with the exception of *debugging information depending of the configured access level*, is made unavailable.

Conformance rationale:

The S400 debug access is disabled by default in all states; there is no service to re-enable it.

3.2.1.5 Residual Information Purging

The platform ensures that *user data*, with the exception of *none*, is erased using the method specified in *the rational below* before the memory is (re)used by the platform or application again and before an attacker can access it.

Conformance rationale:

User data handled by the S400 are the OEM keys or data in secure storage which are stored in SoC memory, external to S400, but encrypted by the S400. Those data need to be erased in case of field return or decommissioning processes which involve a life cycle change; encrypted containers in which those user data are stored are life cycle dependent i.e. the encryption key will be automatically changed during the life cycle change process. The encrypted containers then cannot be accessed anymore after this life cycle change, making the user data unavailable.

3.2.2 Package ‘Security Services’ Security Functional Requirements

3.2.2.1 Cryptographic Operation

The platform provides the application with *list of cryptographic operations specified in Table 10* functionality with *list of algorithms in Table 10* as specified in *specifications in Table 10* for key lengths *defined in Table 10* and modes *defined in Table 10*.

Table 10. Cryptographic Operations by S400

Algorithm	Operations	Specification	[Key] Lengths	Modes
AES	Encryption Decryption	FIPS 197 (AES) NIST SP 800-38A	128, 192, 256 bits	ECB, CBC
AEAD	Authentication Encryption Authenticated decryption	NIST SP 800-38C	128, 192, 256 bits	CCM
SHA2	Hash	FIPS-180-4	224, 256, 384, 512 bits	
HMAC	Mac	FIPS PUB 198-1	224, 256, 384, 512 bits	With SHA-256, SHA-384
CMAC	Mac	FIPS 197 (AES) NIST SP 800-38B	128, 192, 256 bits	CMAC
ECDSA	Signature generation Signature verification	FIPS 186-4	Brainpool R1 and T1 256, 320, 384 bits SECP R1 224, 256, 384, 521 bits	

Conformance rationale:

The S400 implements symmetric (SGI) and asymmetric (PKC) cryptographic accelerator to provide cryptographic operations services to the application. Those resources are dedicated and only accessible by the S400 domain

3.2.2.2 Cryptographic Key Generation

The platform provides the application with a way to generate cryptographic keys for use in *list of cryptographic algorithms in Table 11* as specified in *specifications in Table 11* for key lengths *described in Table 11*

Table 11. Cryptographic Key Generation

Algorithm	Specification	Key Lengths
ECC	ANSI X9.63 NIST FIPS 186-4	From 128 to 640 bits
AES	NIST FIPS 197 SP800-38C	128, 192, 256 bits
HMAC	NIST FIPS 198-1	224, 256, 384, 512 bits

Conformance rationale:

S400 implements cryptographic key generation services for the application based on cryptographic resources dedicated and accessible only by the S400 domain.

Persistent keys generated are then securely stored as described in Section 3.2.2.3.

3.2.2.3 Cryptographic KeyStore

The platform provides the application with a way to store *cryptographic keys* such that not even the application can compromise the *authenticity, integrity, confidentiality* of this data. This data can be used for the cryptographic operations *encryption, decryption, key derivation, signature generation, key exchange, signature verification (see complete list in [8])*.

Conformance rationale:

The S400 provides the application level an API for AES, ECC and RSA key storage. The application keys are sent via the API encrypted by an AES 256 bits pre-shared key.

Persistent keys can be generated or imported in S400 and are stored in SoC memory, encrypted by S400. Blobs are associated to a version (monotonic counter) stored in S400 fuses used for anti-rollback protection and blobs are die unique.

The S400 ensures the secure storage of the persistent keys (versus transient keys not stored) generated for the application.

More details are provided in sections 3.3 and 3.4 of [8].

3.2.2.4 Cryptographic Random Number Generation

The platform provides the application with a way based on *physical noise and DRBG* to generate random numbers to as specified in *[9] and NIST.SP.800-90A CTR-DRBG with AES-128*.

Conformance rationale:

The S400 provides random number to the application level by implementing in software a DRBG as defined in NIST SP 800-90A using a physical TRNG (on-chip entropy source) for the initialization and reseeding with fresh entropy (see more in [9]).

The S400 has a physical true random number generator and internal DRBG module as defined in NIST SP 800-90A. See more in [9].

3.2.3 Package 'Software Isolation' Security Functional Requirements

3.2.3.1 Software Attacker Resistance: Isolation of Platform

The platform provides isolation between the application and itself, such that an attacker able to run code as an application on the platform cannot compromise any other claimed security functional requirements.

Application Note:

PSA profile specific requirements:

- The PSA-RoT is isolated from the NSPE.
- The PSA-RoT is isolated from the Application RoT Services.

Conformance rationale:

All SFRs are fully implemented by the S400 domain with hardware dedicated resources.

Access to those SFRs, and to any S400 service can only be done through a set of interfaces called Message Unit (MU) receiving the SoC requests to be transmitted to S400 domain. There is one MU per domain, A35 and CM33, and a third MU is used for SoC general requests e.g., life-cycle states handling. Messages parsing is fully handled by the S400, and their format is carefully checked.

From S400, out of internal S400 memory and dedicated S400 RAM regions, access to other SoC memory areas is done through DMA or CPU; in such case, format of retrieved data from those memories is carefully checked.

From PSA requirements perspective, this covers the isolation between S400 platform (SPE) and other SoC domains CA35 and CM33 (NSPE). S400 is not handling Application RoT Services.

3.2.4 PSA specific Security Functional Requirements

3.2.4.1 Verification of Platform Instance Identity

The platform provides a unique identification of that specific instantiation of the platform, including all its parts and their versions.

Application Note:

The unique identification of platform must meet the attestation requirements of [4].

Conformance rationale:

The unique identification of a S400 instance is based on the UUID generated and fused into S400 during NXP manufacturing.

This information can be retrieved through the *UUID* field of the *Get Info* message (see chapter 3.31 of [7]).

3.2.4.2 Attestation of Platform Genuineness

The platform provides an attestation of the “Verification of Platform Identity” and “Verification of Platform Instance Identity”, in a way that cannot be cloned or changed without detection.

Application Note:

See attestation mechanism of [4].

Conformance rationale:

The S400 implements an attestation of its genuineness building a payload including the S400 identity (as described in Section 3.2.1.1) and instance identity (as described in Section 3.2.4.1), as shown in section 3.32 of [7].

The payload is signed with ECDSA P256 key.

The nonce has been sent with the attestation request.

The payload preparation and signature are fully performed by the S400 in RROT mode.

3.2.4.3 Attestation of Platform State

The platform provides an attestation of the state of the platform, such that it can be determined that the platform is in a known state.

Conformance rationale:

The S400 implements the attestation of its state by including this state (FW&patch hashes, life-cycle) as part of the attestation payload described in Section 3.2.4.2.

3.2.4.4 Secure External Storage

The platform ensures that all data stored outside the direct control of the platform, except for *none* is protected such that the *authenticity, integrity, confidentiality and binding to platform instance* is ensured.

Conformance rationale:

The S400 implements the secure encrypted storage uses same encryption mechanism as for the persistent key storage (see Section 3.2.2.3). It is encrypted as specified in FIPS 197 and NIST SP 800-38C (AEAD CCM) with a platform instance unique key of key length 256 bits. See section 7.8 of [8].

4 Mapping and Sufficiency Rationales

4.1 SESIP2 Sufficiency

Table 12. SESIP2 Sufficiency

Assurance Class	Assurance Family	Covered By	Rationale
ASE: Security target evaluation	ASE_INT.1 ST Introduction	Section 1	The ST reference is in Section 1.1, the platform reference in Section 1.3 and the platform overview and description in Section 1.5.
	ASE_OBJ.1 Security requirements for the operational environment	Section 2	The objectives for the operational environment in Section 2 refer to the guidance documents.
	ASE_REQ.3 Listed security requirements	Security Requirements and Implementation	All SFRs in this ST are taken from [2] and [3].
	ASE_TSS.1 TOE Summary Specification	Security Requirements and Implementation	All SFRs are listed per definition, and for each SFR the implementation and rational are provided in the SFR.
ADV: Development	ADV_FSP.4 Complete functional specifications	Material provided to the evaluator	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
AGD: Guidance documents	AGD_OPE.1 Operational user guidance	Section 1.4	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
	AGD_PRE.1 Preparative procedures	Section 1.4	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
ALC: Life-cycle support	ALC_FLR.2 Flaw reporting procedures	Section 3.1.1	The flaw reporting and remediation procedure is described.
ATE: Test	ATE_IND.1 Independent testing: conformance	Material provided to evaluator.	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis	N.A. A vulnerability analysis is performed by the evaluator to ascertain the presence of potential vulnerabilities.	The evaluator performs penetration testing, to confirm that the potential vulnerabilities cannot be exploited in the operational environment for the TOE. Penetration testing is performed by the evaluator assuming an attack potential of Basic.

5 Bibliography

5.1 Evaluation Documents

[1] Security Evaluation Standard for IoT Platforms (SESIP), GlobalPlatform GP_FST_070, version 1.1.
[2] SESIP Profile for Secure MCUs and MPUs, GlobalPlatform Technology GPT_SPE_150.
[3] SESIP Profile for PSA Certified Level 2, PSA JSA JSADEN012, v1.0.
[4] Platform Security Model, Arm, ARM DEN 0079.

5.2 Developer Documents

[5] i.MX 8ULP Processor Reference Manual, NXP Semiconductors, rev. C.
[6] i.MX 8ULP Security Reference Manual, NXP Semiconductors, Rev. A.
[7] MX8ULPELEAPI EdgeLock Enclave (ELE) API Reference Guide, NXP Semiconductors, Rev. 1.
[8] EdgeLock Enclave API Bridge detailed implementation, NXP Semiconductors, v0.3.
[9] Design of the Entropy Source in the NXP RNG4 Random Number Generator, NXP Semiconductors, Rev. 1.23.
[10] 8ULP S400 debug authentication detailed specification, NXP Semiconductors.

6 Legal information

6.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

6.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

6.3 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Figures

Contents

Date of release: 29 June 2023