Global Platform®

Securing the digital future

Why SESIP?
A Pragmatic Assurance Framework for Automotive Cybersecurity

GPS_WPR_032

Prepared by the GlobalPlatform
Automotive Task Force | May 2026

Public Release

Version 1.0

Table of Contents

Figures

1 EXECUTIVE SUMMARY

UN Regulation 155 and ISO/SAE 21434 create non-negotiable obligations: OEMs must demonstrate **structured, auditable, and repeatable** cybersecurity assurance across the full vehicle supply chain, and they must do so in a form that type-approval authorities can verify. In software-defined vehicle architectures assembled from dozens of components supplied by different organizations, generating that evidence without duplicating evaluation work across every vehicle development program is a significant cost and schedule risk.

SESIP is a GlobalPlatform methodology and European standard (EN 17927) that directly addresses this cost and risk. A SESIP certificate is a standardized, independently verified statement of what that component does and does not claim, and under what conditions those claims hold. OEMs, Tier-1 suppliers, and type-approval authorities can rely on the certificate across organizational and geographic boundaries, without repeating the evaluation.

The cost of the status quo

Without a shared evaluation framework, every OEM and Tier-1 must independently assess every supplier component. Across a full vehicle development program that means duplicated cost, inconsistent depth, and audit evidence that is **difficult to defend** under UNECE R155/R156 and ISO/SAE 21434/24089 scrutiny.

SESIP eliminates that duplication: A component evaluated once produces a certificate that is reusable across vehicle development programs, model years, and supply chain tiers.

What the certificate gives you

The certificate is the difference between a supplier's assertion and auditable evidence. Five assurance levels map directly to the attack potential assigned in the Threat Analysis and Risk Assessment process, so the depth of evaluation is proportionate to actual risk rather than set by procurement convention.

How assurance flows up the supply chain

SESIP is designed for supply chain composition. Each evaluated component's certificate defines not only what it provides but the conditions the layer above must satisfy for those properties to remain valid. Those conditions chain upward through the stack.

A Tier-1 integrating an evaluated Hardware Protected Security Environment inherits its conditions and must either satisfy them through its own evaluated platform or pass them to the OEM, to resolve the conditions in the vehicle deployment context.

Two rules govern the chain: the overall assurance level cannot exceed the lowest-assurance component in the stack unless a specific argument addresses the gap; and a platform may claim different assurance levels for different security functions within the same product, provided the Security Target makes the boundary unambiguous. The latter is relevant for mixed-criticality platforms where safety-critical and general-purpose workloads share hardware.

Evidence Supporting OEM Liability

Under UNECE R155/R156 and ISO/SAE 21434/24089, the OEM carries non-delegable responsibility for vehicle-level cybersecurity. A supplier's SESIP certificate contributes to the OEM evidence chain.

SESIP improves the quality and consistency of the evidence the OEM receives from suppliers and reduces the cost for the OEM during the verification and validation phase. The role of the OEM focuses on verifying the composition chain and that the aggregate argument holds at vehicle level. SESIP makes that verification tractable according to international metrics on security assurance.

Vehicle Development Program Returns

The vehicle development program returns are concrete:

- **Evaluated once, reused across vehicle development programs:** A certificate for a component carries forward across ECU variants and model years.
    - Evaluation cost is incurred once: The return scales with the scale of the deployment.
- **Supplier assurance becomes auditable:** Required SESIP Profiles and assurance levels can be specified in RFQ documents and verified through the certificate. Self-declarations and bespoke audit processes are replaced by an already recognized standard.
- Working with certified components reduces the likelihood of having to resolve problems in the later development phases **(saving non-recurring engineering costs and reducing delays to market**).
- **Type-approval evidence, ready to use:** SESIP evaluation reports provide normalized technical evidence, including vulnerability analysis and test results, that feed directly into CSMS documentation and type-approval submissions, reducing preparation time and rework.
- **Scalable for software-defined vehicles:** The efficiency gain from reusing evaluated building blocks scales with the scale and complexity of the SDV architecture. The larger the SDV platform and the more hardware and software variants it spans, the greater the return on each evaluation investment.

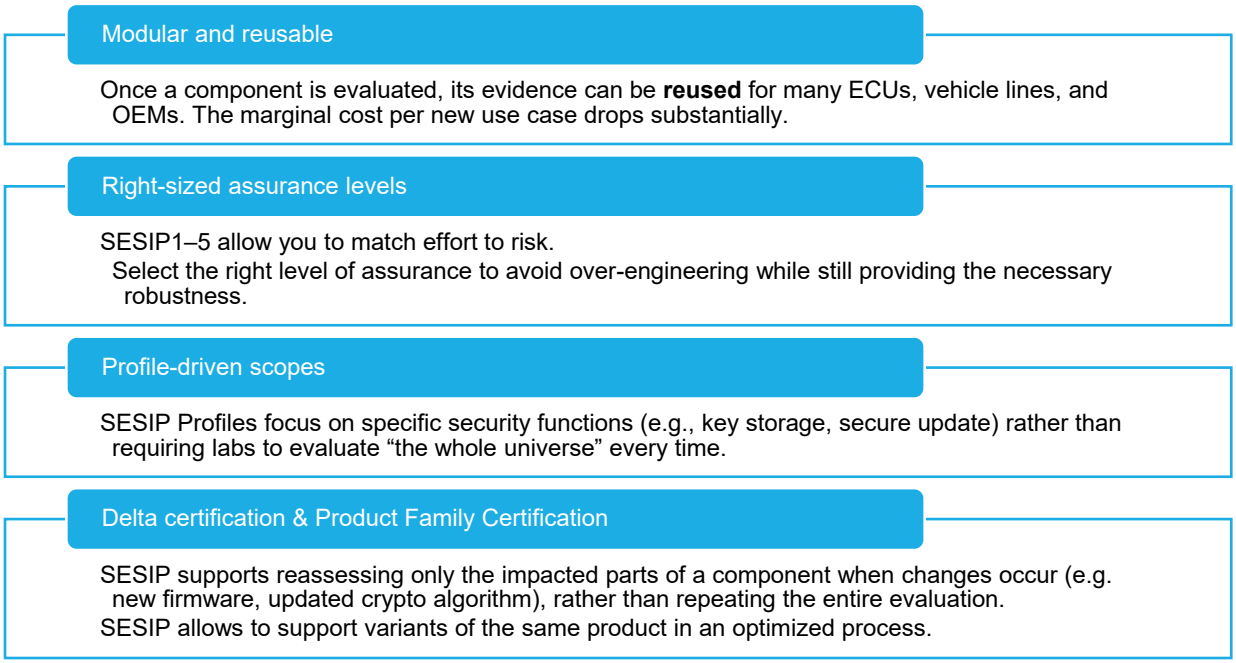

Figure 1: SESIP’s Foundational Origins

The ecosystem is operational

SESIP is an operational standard, not a proposal. Published as CENELEC EN 17927, governed by GlobalPlatform, with accredited laboratories and certification bodies issuing certificates today. Automotive-specific profiles are in active development, including certification paths for Hardware Protected Security Environments per SAE J3101, a CMOS image sensor profile, and an ECU profile.

Automotive adoption is at an early stage and this is a strategic opportunity: Organizations that define supplier requirements now will set the baseline that others will follow. The question is not whether to require SESIP, but where to start. GlobalPlatform continues to develop its cooperation with SAE International to align SESIP with automotive cybersecurity engineering practice.

Three decisions

Three decisions determine how much SESIP delivers to automotive:

Decision	Consideration
Which components to require evaluation for	Not every component warrants formal evaluation. The decision should be driven by the cybersecurity assurance levels assigned in the TARA. Hardware Protected Security Environments and components with direct V2X or OTA responsibilities are natural starting points.
At what assurance level	The assurance level must match the attack potential assigned to the relevant threats. The CAL allocated to the item in the TARA is the primary input: It defines the attack potential the component must be demonstrated to resist. Requiring a higher level (than the threat model justifies) wastes vehicle development program budget; requiring a lower level than necessary creates a gap the composition chain will not be able to bridge.
Supply chain governance	Certificate requirements, applicable profiles, and certificate maintenance obligations simplify the contractual relationship with suppliers while providing discrete robustness assurances for the cybersecurity management system objectives. A supplier whose certificate lapses breaks the composition chain of every vehicle development program that depends on it. The OEM strategically drives the coherence of the full chain.

2 CONTEXT: REGULATORY PRESSURE AND SDV ARCHITECTURES

2.1 Regulatory Landscape

Automotive OEMs and their supply chains must provide structured evidence of cybersecurity and software-update management throughout the vehicle life cycle (development, production, operation, decommissioning). UNECE R155/R156, ISO/SAE 21434/24089, ISO/SAE TR 8477 and ISO/SAE TR 8475, and horizontal regulations require **traceable risk analysis, defined assurance levels, and verifiable implementation of requirements.**

These regulations do not mandate a specific certification scheme for components, but they do require that evidence be **auditable, repeatable, and proportionate to risk**. This creates a need for evaluation methodologies that can be applied consistently across different platform parts and reused in higher-level components.

2.2 Market Evolution Towards Software Defined Vehicles (SDVs)

Modern vehicles increasingly use **shared hardware and software platforms** (e.g. domain/zone controllers, HPC ECUs) with over-the-air updates and multi-tenant workloads. Security-relevant platform parts (Secure SoCs, TEEs, secure storage, update mechanisms) recur across ECUs, platforms, and model years.

In this context, repeatedly evaluating the same building blocks in isolation for each vehicle line is inefficient and may lead to inconsistencies. SESIP's explicit support for **evaluation of platforms and platform parts** with composition and reuse is therefore directly applicable to SDV architectures.

3 WHY SESIP?

3.1 Origins

SESIP was developed by GlobalPlatform to address the scale and heterogeneity of security evaluation for devices and platforms. SESIP:

- Focuses on **platforms and parts thereof**, rather than entire end products
- Defines **Security Functional Requirements (SFRs)** and **Security Assurance Requirements (SARs) packages** grouped into five assurance levels (SESIP1–SESIP5), based on and refining concepts from ISO/IEC 15408
- Supports **composition and reuse**, so that an evaluated platform part can be incorporated into multiple composite products with its assurance preserved, under stated assumptions

SESIP is published as European Standard CEN-CENELEC EN 17927 and provides a basis for aligning with external schemes and regulations.

3.2 SESIP Ecosystem

SESIP is a live ecosystem with operational governance, active laboratories, and certificates being issued today. GlobalPlatform maintains the methodology, profiles, and supporting documents, and coordinates:

- **Certification bodies and laboratories** licensed to operate SESIP-based schemes, typically under ISO/IEC 17065 and 17025 accreditations
- **SESIP Profiles** for several platform classes
- **Mappings** that relate SESIP SFRs and SARs to external standards or schemes

GlobalPlatform's SESIP Committee and associated working groups oversee the evolution of the methodology, the standard profiles, mappings, and interpretation. Importantly, the Governance Working Group of the SESIP Committee provides a forum to address cross-scheme alignment topics.

Ecosystem at a Glance	**Certification Bodies:** Currently TrustCB and BrightsightCB. For the latest updates: https://globalplatform.org/sesip-cb/ **SESIP Laboratories**: Applus+, BCTC, Brightsight, CAICT, DEKRA, DPS Labs, ECSEC, Institute for Information Industry, Keysight, SERMA, Thales ITSEF. For the latest updates: https://globalplatform.org/sesip-lab/ **Published SESIP Profiles:** Code Update Mechanism; DTSec Connected Diabetes Device Platforms; Edge Compute Node (ECN); Secure External Memories; Secure MCUs and MPUs; WPC Qi Chargers. For the latest updates: https://globalplatform.org/specs-library/ **Published SESIP mappings:** EN 18031/RED hEN; IEC 62443-4-2; NIST 8425; SSIPS **Certificate validity:** 5 years from Evaluation Technical Report issuance date

3.2.1 PSA Certified: SESIP at Scale

PSA Certified is a security evaluation scheme created by an ecosystem promoting two key fundamental security measures that should be embedded into every connected device: the PSA Certified Security Goals and the PSA Root of Trust.

Originally built by experts from seven founding companies, PSA Certified was established as an independent, collaborative initiative designed to adapt to industry and geographic demands. Today, the scheme is managed by GlobalPlatform, ensuring its continued growth and alignment with global security needs.

This security certification scheme has resulted in strong traction in different industries with close to 300 certifications having been issued for over 100 companies.

PSA Certified evaluation scheme was donated to **GlobalPlatform in September 2025**, where it is now maintained and advanced. Currently, the PSA Certified scheme is being aligned with SESIP evaluation methodology. The alignment between PSA Certified Levels 2 and 3 with SESIP Levels 2 and 3 respectively has been completed.

PSA Certified demonstrates that the SESIP evaluation methodology, laboratory network, and certification infrastructure are operational at scale today. For additional information: https://globalplatform.org/what-is-psa-certified/.

3.3 SESIP as a Component-centric Framework

SESIP is scoped to platform parts. This matters for automotive supply chains because it means each supplier evaluates what they own:

- Silicon vendors evaluate silicon
- OS vendors evaluate firmware

and the results compose upward without re-evaluation.

SESIP applies to any platform part, from secure microcontrollers and cryptographic libraries up to higher-level integration stacks.

An evaluation is scoped to a specific platform or platform part, and results in:

- A **Security Target** describing the security problem, objectives, and claimed requirements at a chosen SESIP assurance level.
- A set of evaluation activities and findings, leading to a certificate at that level, issued under a SESIP-based scheme.

Silicon vendors, Tier-1s, and OEMs can each contribute evaluated platform parts to the overall security argument.

3.3.1 SESIP General Resources

- Methodology: https://globalplatform.org/specs-library/sesip-methodology/
- SESIP FAQ: https://globalplatform.org/specs-library/security-evaluation-standard-for-iot-platforms-sesip-faq/
- Cryptographic Algorithm Recommendations: https://globalplatform.org/specs-library/globalplatform-technology-cryptographic-algorithm-recommendations/
- GlobalPlatform SESIP Profiles: https://globalplatform.org/specs-library/?filter-committee=sesip

3.4 SESIP Elements

SESIP is developed and maintained by GlobalPlatform and is recognized internationally as a relevant methodology for supporting regulatory compliance. ENISA recognizes it as a relevant cybersecurity methodology for assessing ICT solutions; in the US context, SESIP aligns with NIST IR 8425 device capability baselines.

SESIP solves a specific inter-organizational problem: How do you express a platform component's security properties in a form that organizations further up the supply chain can rely on, without each of them independently re-evaluating the components they did not build?

The answer is a structured set of claims, called Security Functional Requirements, scoped to a defined platform with a documented boundary, backed by evidence appropriate to the claimed assurance level, and verified by an accredited evaluation laboratory. The certificate is the transferable artefact; composition is the mechanism by which it is consumed.

SESIP is deliberately scoped to platform components, allowing each supplier to evaluate what it owns (i.e. independently, on its own schedule, against its own threats) and to present that evidence in a structured and composable form. Complex components like ECUs can then be evaluated as compositions of their constituent evaluated parts.

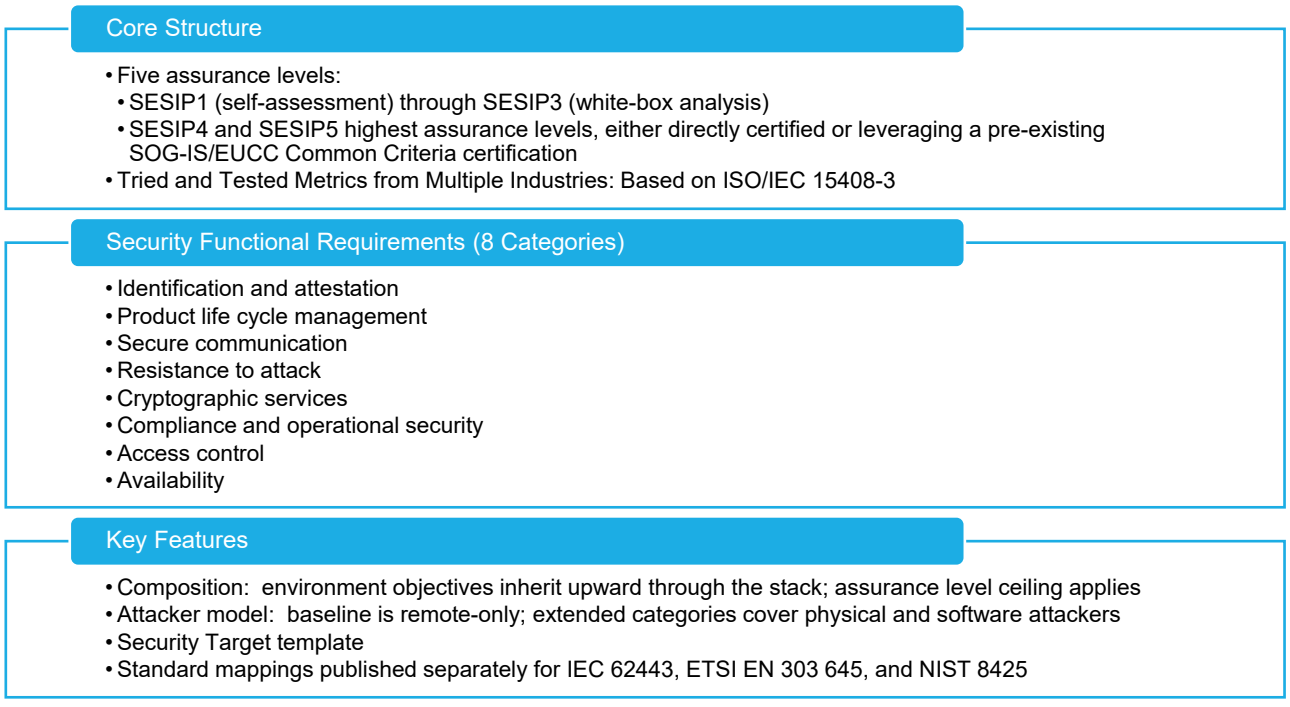

Figure 2: SESIP Methodology Overview

3.5 SESIP Levels: What They Actually Mean

SESIP defines five assurance levels (SESIP1-5). SESIP levels are explicitly tied to attack resistance – the level of effort a realistic adversary must expend to defeat the claimed security properties – not just to documentation rigor.

In automotive, level selection is driven by the attack potential assigned to threats in the TARA process and the security goals derived from ISO/SAE 21434.

Each level defines a minimum set of SARs and associated **attack potential** to be considered. Selecting a level therefore specifies the minimum evaluation effort required and the attack difficulty against which the TOE is claimed to be robust.

- **SESIP1**:
  - Self-assessment with a simplified Security Target and basic checks (e.g. security flaw handling, public vulnerability review).
  - Claims rest entirely on vendor-supplied evidence.
  - Rarely sufficient for automotive platform components where independent verification is expected by Tier-1s under UNECE R155/R156 supply chain obligations.
- **SESIP2**:
  - Independent black-box penetration testing based on a functional specification; no source code required.
  - In automotive, most evaluations are performed on the right side of the V development cycle (the last phase), close to going to production. SESIP Level 2 includes functional testing of the SFRs, evaluation of SARs, and AVA.VAN activities up to Level 2.
- **SESIP3**:
  - White-box vulnerability analysis including design and/or source review plus AVA.VAN activities (penetration testing); associated SARs support vulnerability-driven evaluation.
  - Lab performs functional testing and vulnerability analysis.
  - Resistance to Enhanced-Basic attack potential.
  - A practical baseline for automotive platform components (e.g. bootloaders, OS kernels, hypervisors) where physical access is assumed possible but sophisticated lab-grade attacks are outside the expected threats.

- **SESIP4**:
  - Structured AVA.VAN activities (penetration testing); associated SARs support vulnerability-driven evaluation by an accredited lab.
  - Resistance to Moderate attack potential. Appropriate for Hardware Protected Security Environment firmware and for secure communication stacks on ECUs with network exposure.
  - Increasingly expected for components in safety-adjacent domains or with V2X/OTA responsibilities.
  - Complementary evaluation with AVA_VAN4 methodology.
  - Enables reuse of evaluations already certified under SOG-IS/EUCC at AVA_VAN.4 or higher, or direct SESIP4 certification for specialized components. [1]
- **SESIP5**:
  - AVA.VAN activities (penetration testing); associated SARs support vulnerability-driven evaluation with resistance to High attack potential.
  - Reserved for hardware secure elements, Hardware Protected Security Environment silicon, and cryptographic modules at the root of the trust chain: the components whose compromise would undermine the entire composition chain.
  - Aligned with Common Criteria EAL 5 attack potential.
  - Enables reuse of evaluations already certified under SOG-IS/EUCC at AVA_VAN.5 or higher, or direct SESIP5 certification for specialized components.

Key Point	*Level selection is a threat-model decision tied to your TARA and CAL objectives, not a procurement decision.* *An over-evaluated component wastes vehicle development program budget; an under-evaluated one creates a gap the composition chain cannot bridge.*

[1] Note that the SESIP Level 4 & 5 evaluation results were first used for Common Criteria products and hence leverage these certificates. The decision on SESIP Level 4 & 5 for products without previous certificates is currently being refined.

4 SESIP STRUCTURAL COMPONENTS

A SESIP evaluation is built from six structural elements. Understanding each element, and how they relate to one another, is essential before attempting to scope or architect a security evaluation.

4.1 Platform Scope

The platform scope is the precise, bounded object being evaluated. In SESIP, the evaluated object is always a platform or platform part; e.g. a silicon vendor's software development kit, a ROM bootloader, or an isolated security execution environment implemented in hardware.

Defining the platform scope is arguably the most critical architectural decision in a SESIP engagement, and the decision has long-range consequences for every layer of the stack built on top of it.

A well-defined platform scope:

- includes all hardware and software components that implement or enforce the claimed Security Functional Requirements
- excludes components whose security properties are not relevant to those claims, but documents their assumed behavior as security objectives for the operational environment in the Security Target, where they become explicit conditions that integrators above must satisfy
- is expressed in the *Platform Functional Overview and Description* section of the Security Target, which defines the platform boundary, its interfaces, and its external dependencies. This is the precise surface that upper layers will depend on and that the evaluator uses to bound the assessment

4.1.1 Examples of Composition Applied to Automotive

For each interface between evaluated layers, the lower platform part must define its platform scope and environment objectives precisely enough that the integrating layer can resolve them unambiguously.

The table below sets out examples of what each party must do at each boundary.

Example composition boundary	What the lower platform part must define in its Security Target	What the integrating layer must resolve
Hardware Protected Security Environment to ROM bootloader	**Platform scope:** The precise hardware boundary of the isolated execution and storage environment, including all interfaces through which security services are exposed upward. **Mandatory SFRs to claim:** Cryptographic services, key store, physical attacker resistance, verification of platform identity. **Environment objectives to define:** The conditions the bootloader must satisfy, for example that debug interfaces are disabled before handover, that key provisioning follows a specified procedure, and that the boot sequence invokes the hardware verification mechanism.	**Must satisfy or inherit:** Every environment objective from the Hardware Protected Security Environment Security Target. Each must either map to a claimed Security Functional Requirement in the bootloader's Security Target, or be carried forward as an environment objective for the layer above. **Assurance ceiling:** The bootloader platform part cannot be certified above the assurance level of the Hardware Protected Security Environment it depends on, unless a specific argument is made.
ROM bootloader and secure boot chain to hypervisor or trusted execution environment	**Platform scope:** The verification chain from the hardware root of trust to the first mutable software layer, with explicit interface description of what the hypervisor or trusted execution environment may assume has been verified. **Mandatory SFRs to claim:** Secure initialization of platform, attestation of platform genuineness, and cryptographic	**Must satisfy or inherit:** All inherited environment objectives from the Hardware Protected Security Environment layer, plus those defined by the bootloader. The hypervisor or trusted execution environment Security Target must contain a dedicated inherited objectives section accounting for each one. **Key risk:** If the bootloader's platform scope is defined too

Example composition boundary	What the lower platform part must define in its Security Target	What the integrating layer must resolve
	services supporting signature verification. **Environment objectives to define:** For example, that the hypervisor or trusted execution environment does not modify the boot measurement chain, and that the platform is only deployed in environments consistent with the Security Target's operational assumptions.	narrowly and does not cover the full verification chain, the upper layer inherits an unverified gap rather than an evaluated property.
Hypervisor or trusted execution environment to regular operating system or AUTOSAR	**Platform scope:** The isolation boundary between partitions must be explicitly described, including which interfaces cross partition boundaries and under what conditions. The security services interface exposed to the regular operating system must be fully specified. **Mandatory SFRs to claim:** Software attacker resistance through isolation of platform parts, access control, and any cryptographic services or key store functions exposed as services. **Environment objectives to define:** For example, that the regular operating system invokes cryptographic operations only through defined interfaces, does not attempt to access secure partition memory directly, and follows the guidance for secure service invocation.	**Must satisfy or inherit:** All accumulated environment objectives from lower layers. By this point in the stack the list may be substantial; the regular operating system Security Target must address each one explicitly. **Additive composition opportunity:** The operating system may claim a lower SESIP level for general execution environment functions and a higher level for the isolated security partition, provided the Security Target makes the boundary between the two claims unambiguous.

Example composition boundary	What the lower platform part must define in its Security Target	What the integrating layer must resolve
Regular operating system or AUTOSAR to vehicle application and OEM software layer	**Platform scope:** The security services interface consumed by the application layer must be defined as part of the platform scope. Security Functional Requirements claimed at this layer, such as secure communication enforcement or access control, must be described with sufficient precision that the application developer can verify their deployment satisfies the operational environment assumptions. **Environment objectives to define:** For example, that the application does not bypass secure communication channels, invokes update mechanisms in accordance with the guidance, and does not install or execute code from untrusted sources.	**OEM closure obligation:** This is the point at which the composition chain closes. It must be demonstrated that all accumulated environment objectives across the full stack are either satisfied by a Security Functional Requirement claimed at this layer or explicitly owned as a condition of the vehicle deployment context and addressed in the vehicle cybersecurity case. **UN R155 alignment:** The OEM's non-delegable responsibility for vehicle cybersecurity means that unresolved environment objectives at this boundary are outstanding.

4.1.2 Examples of Composition Failures As Applied to Automotive

The following failure mode examples are the most common causes of composition gaps as applied to the multi-supplier nature of automotive.

Failure mode	How it arises	Consequence for the composition chain
Platform scope defined too narrowly	The lower layer's platform scope excludes a component that implements a claimed Security Functional Requirement; for example, a cryptographic accelerator that is outside the defined hardware boundary.	The upper layer inherits an assumption about a property that was never actually evaluated. The composition gap is invisible in the certificate but present in the deployment.
Environment objective not resolved at any layer	An environment objective defined by a lower layer Security Target is carried upward without being mapped to a Security Functional Requirement at any intermediate layer, and is not explicitly owned in the vehicle cybersecurity case.	The certificate's claimed properties are conditional on an assumption that is never verified in the deployment context. Under UNECE R155/156 this is a CSMS gap.
Interface defined outside the Security Target	The security-relevant interface between layers is documented only in an API reference or design specification that the evaluator does not treat as normative evidence.	The evaluated boundary does not match the deployed boundary. Upper layer assumptions about interface behavior are unverified, breaking the traceability on which composition depends.
Assurance level mismatch	A higher-assurance layer is composed on top of a lower-assurance platform part without a specific argument addressing the gap.	By default the composition can only be certified to the lowest assurance level in the chain. The higher-assurance claims of the upper layer are not valid for the composition as a whole.
Certificate maintenance failure	A supplier's certificate for a lower layer expires, is withdrawn, or covers a component version that has since been superseded by a security update.	The evaluated basis for the composition is broken. This is not addressed by the SESIP methodology itself and must be managed contractually under the vehicle development program's cybersecurity management obligations.

4.2 Security Functional Requirements (SFRs)

Security Functional Requirements are the generic security features that a platform implements and claims in its Security Target. SESIP defines a set of security functional features from which developers select the requirements that reflect what their platform actually provides.

Each Security Functional Requirement covers a complete security purpose by itself, written in plain language so that product vendors and integrators can understand and rely on the claims without specialist evaluation expertise.

Selection is not prescriptive: The developer chooses the requirements that reflect what their platform does and the threats it addresses, with two exceptions:

- Verification of Platform Identity is mandatory for every evaluation.
- Secure Update of Platform must either be claimed or formally argued as not applicable in the flaw reporting procedure.

For each claimed requirement, the Security Target must include a conformance rationale describing how the platform implements it and how that implementation has been assessed.

The full set of SESIP SFR categories is summarized below:

Security Functional Requirements Category	What it requires and why it matters
Identification and attestation	The platform must be able to prove what it is, which specific unit it is, and that it has not been cloned or tampered with. This includes verifying a unique identity for the platform model and for each individual device instance, cryptographically confirming that the platform is genuine rather than a counterfeit, and checking its own integrity and authenticity every time it starts up. The platform must also be able to confirm that any application running on it is genuine and in a known, trusted state.
Product life cycle management	Security must be maintained throughout every phase of the product's life, not just during normal operation. This covers the secure installation of software on a device in the field, delivering updates over the air in a way that prevents rollback to older vulnerable versions, removing an application while guaranteeing that all associated data is destroyed, and permanently decommissioning a device at end of life so that no sensitive data can be recovered. It also covers returning a unit for repair or investigation without exposing user data to the vendor.

Security Functional Requirements Category	What it requires and why it matters
Secure communication	Any communication between the device and external systems, such as backend servers, cloud services, or other connected devices, must take place over authenticated and encrypted channels. The requirements cover both the provision of such channels (specifying which protocols and algorithms are used and what attacks they protect against) and their enforcement (ensuring the platform actively prevents unprotected communication, so an application cannot accidentally bypass the security layer and transmit data in the clear).
Resistance to attack	Beyond protecting against remote network threats, these requirements address scenarios where an attacker has physical access to a device or is able to run malicious code on part of the system. Physical resistance requirements are relevant wherever a device may be accessible outside a controlled environment, such as a vehicle in a workshop, at a fuel station, or recovered after an incident. Software isolation requirements ensure that a compromise of one component, such as a general-purpose application, cannot spread into security-critical functions.
Cryptographic services	The platform must provide standard-compliant cryptographic functions for use by applications running on it. This includes encryption and decryption operations, the generation of cryptographic keys, the secure storage of keys and credentials in a protected vault from which raw key material cannot be extracted by the application, and the generation of cryptographically strong random numbers needed for authentication and key material. Each requirement specifies the algorithms, key lengths, and referenced standards that must be followed, giving integrators verifiable confidence in the quality of the cryptographic foundations.

Security Functional Requirements Category	What it requires and why it matters
Compliance and operational security	This is the broadest category, covering a range of properties commonly required by product regulations and sector-specific standards. It includes several distinct levels of storage protection: • verifying that stored data has not been modified (integrity and authenticity), • ensuring data is inaccessible to unauthorized parties (confidentiality), and • encrypting all stored data at rest with a device-unique key. It also covers the secure handling of data written outside the platform's direct memory (serialization), the active erasure of sensitive data from memory when it is no longer needed (to prevent recovery after power loss or cold-boot attacks), the generation and protection of audit logs recording security-relevant events, a monotonically increasing counter useful for replay protection and versioning, and the restriction of debug access to authenticated parties only. Recovery and backup requirements ensure the platform can return to a known safe state after a fault, while a generic catchall requirement allows platform-specific security features not covered by the catalogue to be formally evaluated.
Access control	The platform must enforce restrictions on which entities are permitted to access which resources or trigger which operations. Two distinct mechanisms are required. The first covers access based on context rather than credentials; for example, restricting an operation to requests originating from a trusted hardware zone, a specific operating mode, or a physical push-button action. The second adds formal authentication, requiring any party requesting a privileged operation to be identified, authenticated, and explicitly authorized before access is granted.
Availability	Two complementary requirements address the continued operation of the device under adverse conditions. The first ensures the platform does not itself become the source of a denial-of-service problem; for example, by placing unreasonable demands on shared infrastructure when many devices operate concurrently. The second ensures the platform can withstand denial-of-service attacks directed at it, remaining operational and continuing to fulfil its security functions even when subjected to deliberate traffic flooding or resource exhaustion attempts.

4.2.1 Choosing SFRs

Not all security features are claimed in every evaluation. Selection is driven by the platform's function, the threats it faces, and the security properties that product vendors and integrators above it need to rely on. SESIP does not prescribe which features must be selected beyond the two mandatory requirements; the developer's threat context and the intended deployment determine the rest.

In automotive, the relevance of each Security Functional Requirements category can be summarized in the following manner:

- Identification and attestation confirms that a platform is what it claims to be and has not been modified or substituted. For an automotive cybersecurity architect this is the formal equivalent of secure boot combined with device identity, directly addressing the UN R155 threat of ECU substitution and unauthorized firmware modification.
- Product life cycle management maps directly to the OTA update threat surface, anti-rollback protection, and the ISO 21434 obligation to maintain cybersecurity throughout vehicle operational life.
- Secure communication requires that communication channels are authenticated and encrypted, and that the platform cannot be made to bypass them. For an automotive architect this covers the same ground as secure onboard communication for internal bus traffic and TLS for backend connectivity and is directly relevant to the UN R155 threat categories around unauthenticated and unencrypted interfaces.
- Resistance to attack extends the threat model beyond remote attackers to include physical access and hostile code execution. This is relevant wherever a component could be accessed outside a controlled environment, covering resistance against physical attacks such as side-channel analysis and fault injection.
- Cryptographic services defines what cryptographic services the platform provides and to what standard. For an automotive architect this is the assurance basis for trusting that key generation, storage, and algorithm implementation meet the quality required by the vehicle cybersecurity case, particularly for components handling certificate management or secure onboard communication keys.
- Compliance and operational security covers data erasure, audit logging, secure debug access, and recovery. This category is most directly relevant to diagnostic security, incident response obligations under ISO 21434 (operational phase), and data privacy when a vehicle changes ownership.
- Access control defines which entities can access which functions and under what conditions. This is the formal underpinning of diagnostic access restrictions, privilege separation between execution domains, and the principle that safety-critical functions should not be reachable from general-purpose or externally-facing software layers.
- Availability addresses platform resilience against resource exhaustion and denial-of-service attacks, including bus flooding on ECUs and telematics units operating under load.

Depending on the assurance level, claimed security features must be substantiated by evidence appropriate to that level:

- At the lowest level this is a developer self-assessment rationale.
- At higher levels it extends to functional specifications, test results, and source code analysis.

The assurance level determines the depth of evidence required, not the set of features claimed.

Security feature selection at one layer creates obligations for the layers above. Whatever a lower platform part claims, the layer above can rely on. Whatever a lower platform part does not claim, the layer above must either claim itself or accept as a gap in its security argument.

Engineering Implication	*Security feature selection is an architectural commitment.* • *Claiming the Cryptographic Key Store requirement means that the hardware and software must demonstrably protect the confidentiality and integrity of stored keys such that not even the application can access the underlying key material.* • *Claiming the Residual Information Purging requirement means that the platform must actively erase data using a method appropriate to the underlying memory technology before that memory can be accessed again.* • *Claiming Software Attacker Resistance isolation requirements means that the platform must demonstrably prevent code running in one domain from compromising the security functions or data of another.* *These are design constraints that must be resolved before the Security Target is finalized, not after evaluation commences.*

4.3 Security Assurance Requirements (SARs)

SARs define how rigorously the evaluation itself must be conducted: the activities, evidence, and depth of examination the lab must perform at each level. This definition is based upon a combination of different assurance requirements which are grouped into six families to cover the five different assurance levels (SESIP1-5):

Assurance family	What it requires	How depth changes: SESIP1 to SESIP5
ASE: Security Target Evaluation (all five levels)		
Introduction, environment objectives, requirements list, platform summary	Identifies the platform, states the SESIP version, lists all claimed requirements from the SESIP security feature catalogue with a compliance rationale for each. Verification of Platform Identity is always mandatory. No augmentation of the declared assurance level is permitted.	**SESIP1:** Developer self-assessment only. Evaluator checks clarity and consistency but does not independently verify the implementation. **SESIP2:** Add complete functional specification. **SESIP3:** Add full source code with requirement mapping. **SESIP4–5:** Additionally reference the prerequisite Common Criteria certificate and cross-reference the original Common Criteria requirements.
ADV: Development (absent at SESIP1; source code and architecture added progressively from SESIP2)		
Functional specification, implementation mapping, security architecture, modular design	Descriptions of all platform interfaces mapped to claimed requirements, supplemented at higher levels by source code, security architecture, and subsystem design.	**SESIP1:** Not required. **SESIP2:** Complete functional specification required. No source code. **SESIP3:** Add full source code with mapping from source to each claimed requirement. **SESIP4:** Add explicit security architecture description. **SESIP5:** Additionally requires a basic modular design decomposing security function into subsystems.

Assurance family	What it requires	How depth changes: SESIP1 to SESIP5
AGD: Guidance Documents (present and unchanged at all five levels)		
Operational user guidance and preparative procedures	Publicly accessible documentation on secure platform operation and preparation, addressing every environment objective in the Security Target.	**SESIP1–5:** Identical requirement at all levels. Typically satisfied by existing data sheets and user manuals.
ALC: Life Cycle Support (one component at SESIP1–2; expands to five components at SESIP5)		
Flaw reporting, configuration management, delivery, development security, development tools	Procedures for receiving, tracking, and communicating flaw reports. Expands at higher levels to cover controlled delivery, version management, development environment security, and tooling documentation.	**SESIP1–2:** Flaw reporting only. Only externally visible steps need be described. **SESIP3:** Full flaw reporting procedure including internal steps. Platform must be held in a version control system with unique version identification. **SESIP4:** Add delivery procedures, identification of development security measures, and documented development tools. **SESIP5:** Configuration management must include automation and problem-tracking coverage. Security measures must be demonstrated as sufficient, not merely identified.
ATE: Tests (absent at SESIP1; independent testing from SESIP2; developer testing and coverage evidence at SESIP4–5)		
Independent testing, coverage evidence, functional testing, subsystem testing	An independent evaluator tests the platform against the functional specification. At higher levels the developer's testing documentation is reviewed and coverage must be demonstrably linked to claimed security functions.	**SESIP1:** No independent testing. **SESIP2–3:** Independent conformance testing introduced. Industry-standard test results may be reused where accepted by the certification body. **SESIP4:** Add developer functional testing with coverage evidence. **SESIP5:** Add subsystem-level testing to match the modular design requirement.

Assurance family	What it requires	How depth changes: SESIP1 to SESIP5
AVA: Vulnerability Assessment (present at all five levels; attacker capability escalates at each step)		
Vulnerability survey to advanced methodical analysis	The evaluator identifies vulnerabilities and conducts penetration testing to demonstrate resistance to a defined attacker capability level, which increases at each assurance step.	**SESIP1:** Vulnerability survey only: publicly known vulnerabilities identified and confirmed as addressed. No independent penetration testing. **SESIP2:** Independent penetration testing. Resistance to Basic attack potential required (public tools, limited specialist knowledge). **SESIP3:** Source-code-informed focused analysis. Resistance to Enhanced-Basic attack potential required (moderate specialist knowledge, code-level analysis). **SESIP4:** Methodical analysis. Resistance to Moderate attack potential required (significant specialist knowledge, structured multi-step campaigns). **SESIP5:** Advanced methodical analysis. Resistance to High attack potential required: the level applied to smartcards, secure elements, and electronic passports.

These requirements enable evaluations to leverage existing secure-development artefacts, facilitating integration with established cybersecurity engineering and CSMS processes.

4.4 Security Target

The Security Target is the primary document produced by the platform developer and forms the basis of the entire evaluation. It defines what security properties are claimed for the platform and is required at all five assurance levels. If a SESIP Profile for the relevant product type is being used, the Security Target needs to cite it, integrating at minimum all the security objectives and requirements the profile specifies.

The Security Target template is provided in Annex D of the SESIP Methodology (https://globalplatform.org/specs-library/?filter-committee=sesip), and focuses on the definitions from SESIP1 to SESIP3. It defines both mandatory and optional sections. At SESIP4 and SESIP5, the template sections remain applicable but are supplemented with additional requirements.

Security Target section	What must be included	Key requirements and notes from the template
D.1 Security Target title page		
Title page	Mandatory cover information identifying the platform, version, developer, and the SESIP methodology version used.	Must include platform name, document version and date, development organization, and a statement of the SESIP methodology version on which the evaluation is based. No other sections are required on the title page.
D.2 Introduction		
D.2.1 Security Target reference	Reference back to the title page. No separate content required.	Satisfied by the title page. No additional content is needed in the body of the document.
D.2.2 Platform reference	A structured table uniquely identifying the platform by name, version, identifier, and type.	Must provide a globally unique identifier that distinguishes this evaluated version from all others. The identification must be consistent with the *Verification of Platform Identity* requirement claimed in section D.4.2.
D.2.3 Included guidance documents	A table listing all documents provided with the platform, with title, reference, and version.	Must include all documents that will be provided to the evaluator for the documentation review. These documents must be publicly accessible to customers without restriction.

Security Target section	What must be included	Key requirements and notes from the template
D.2.4 Other certification (optional)	Details of any prior evaluation or certification of the same platform under a different scheme.	Optional section. If included, must state the scheme, certification body, certificate number, and issuance date. A link to the certificate or a copy may also be provided.
D.2.5 Platform functional overview and description	A short description of the platform covering its components, intended use, main security features, and the boundary of the evaluation scope.	**Scope:** The platform scope must be depicted in a figure distinguishing in-scope parts from out-of-scope parts. The physical scope must be explicitly described. **Length:** Typically one to two pages covering the datasheet-level description, an overview figure, and the feature set. **Content:** Must describe the platform parts, the intended integrator or user, and the main security features. Must state what is excluded from the evaluated scope.
D.3 Security objectives for the operational environment		
D.3.1 Platform objectives for the operational environment	A list of all conditions the surrounding environment must satisfy for the platform's security claims to hold.	**Format:** Each objective must be described with enough detail for a product vendor or integrator to understand and implement it, and must include a precise reference to the guidance document section where it is addressed. **Examples from template:** The application must verify the correct version of all platform components it depends on; the platform must only be deployed where no physical attacker is possible; the application must not allow execution of hostile code.
D.3.2 Inherited objectives for the operational environment (composite platforms only)	For composite platforms, an explanation of how every environment objective defined by an incorporated platform part has been handled.	**Required when:** The platform incorporates one or more platform parts that have themselves been evaluated under a SESIP scheme. **Content:** For each environment objective of each platform part, the

Security Target section	What must be included	Key requirements and notes from the template
		Security Target must explain whether the objective is fully covered by the platform's own security requirements, partially covered, or passed forward as an environment objective for the composite platform.
D.4 Security requirements and implementation		
D.4.1 Security Assurance Requirements	A declaration of the claimed SESIP assurance level together with the evidence required at that level that is included directly in the Security Target.	**Declaration:** Must state exactly one of SESIP1, SESIP2, SESIP3, SESIP4, or SESIP5. No augmentation is permitted. **Embedded evidence:** At SESIP1 and SESIP2 the flaw reporting procedure must be described within this section. At SESIP1 the vulnerability survey and its test results must also be included here, as these cannot be verified from external deliverables at that level.
D.4.2 Security Functional Requirements	The full list of claimed security requirements, each with a conformance rationale describing how the platform implements it and how that implementation has been assessed.	**Mandatory SFRs:** *Verification of Platform Identity* must always be included. *Secure Update of Platform* must be included or, if not, the flaw reporting section must contain a formal argument for why updates are not applicable. **Conformance rationale:** For each requirement the developer must describe: (1) how the platform implements the requirement, and (2) how that implementation has been assessed, for example by testing, conformance to another standard, or reliance on a third-party statement. **Additional requirements:** Any requirement not from the SESIP security feature catalogue must go in section D.4.3 with a clear statement that it falls outside SESIP.

Security Target section	What must be included	Key requirements and notes from the template
D.4.3 Additional security functional requirements (optional)	Platform-specific security requirements that are not part of the SESIP security feature catalogue.	Must be clearly separated from the SESIP requirements and must begin with a statement that these requirements are not part of SESIP and may not be recognized by other stakeholders. Each requirement must be written in plain language with a conformance rationale.
D.5 Mapping and sufficiency rationales		
D.5.1 to D.5.5 Sufficiency rationale for the claimed assurance level	A table mapping each assurance family to the section of the Security Target or external deliverable that satisfies it, together with a brief rationale.	Demonstrates to the evaluator that all assurance requirements are addressed. Only the rationale for the selected level should appear in the actual Security Target.

The Security Target serves three distinct purposes within a SESIP evaluation.

- For the evaluation laboratory, it is the normative scope document that bounds the assessment.
- For a product vendor or integrator building on the evaluated platform, it is both a specification of the security properties they can rely on and a statement of the environment objectives they must satisfy for those properties to hold in their deployment.
- For any party seeking to reuse the evaluation result, including in a composition with other evaluated components, it is the publicly accessible document that makes the security claims and their conditions transparent and verifiable.

The most common source of evaluation delays is a Security Target that is internally inconsistent: where a threat is listed but no security objective addresses it, or an SFR is claimed but the rationale does not trace it to an objective.

4.5 SESIP Profiles

Each SESIP Profile defines the minimum security requirements and associated evaluation activities for a specific type of platform or product category, tailored to the relevant threats and use cases identified by the stakeholders of that product type. It is not a product-specific document but rather a generic security template that any platform of the given type must conform to in order to achieve credible certification. The profile explicitly states the assurance level or levels being claimed, and each security function defined within it must be directly traceable to a specific threat it addresses. In this manner, SESIP Profiles provide market transparency on security levels as long as the levels are compared with the “same” target of evaluation.

Profiles can be developed either within GlobalPlatform through the SESIP Technical Working Group, or externally by industry stakeholders and submitted to a Certification Body for review and approval.

SESIP Profile Section	Content
Introduction	A high-level introduction to the concept of a SESIP Profile, including the profile name, version, and the type of platform it covers. Also contains a list of references, a glossary of key terms and abbreviations, and a revision history.
Platform Functional Overview and Description	A comprehensive description of the platform type, including its architecture, components, security features, intended use cases, and security boundaries. May include a block diagram illustrating the platform’s components and interactions. Optionally highlights any pre-existing certifications relevant to the platform type.
Security Objectives for the Operational Environment	The security responsibilities and expectations placed on the environment in which the platform is deployed. Where the platform is composed from separately evaluated components, inherited objectives from those components must also be listed.
Security Requirements	The Security Assurance Requirements, aligned with the claimed assurance level, defining the evaluation activities that must be performed. The Security Functional Requirements, listing the specific security functions the platform must implement. Any security process packages the platform must follow to mitigate identified threats.
Mapping and Sufficiency Rationales	A justification for the selected assurance level based on the threat landscape and risk assessment, with a structured demonstration of how each assurance requirement is met. Also includes mappings to relevant external standards to support broader compliance recognition.

4.5.1 SESIP Profiles vs. Security Target

A SESIP Profile and a Security Target are related but distinct documents

The relationship is best understood through a concrete analogy between Building Codes and Structural Drawings.

A SESIP Profile:

- Is like a building code for a category of structure.
- Includes regulations governing, e.g. all multi-story reinforced concrete buildings in a given jurisdiction.
- Written once, by an industry body or standards committee, and applies to every building of that type.
- Specifies the minimum requirements any such building must meet, the hazards it must withstand, and the evidence an inspector will need to verify compliance.
- Says nothing about any specific building.

The Security Target:

- Structural drawings for a specific building.
- Authored by its architect for that specific building, on that site.
- Must demonstrate conformance to the applicable building code
- Captures everything particular to this structure: exact materials, specific foundation design, actual load calculations, and the precise scope of what is and is not part of the building.

How they work together:

- The evaluation lab (the inspector) works from the structural drawings, not the building code, but ...
- uses the building code as the benchmark against which the drawings are assessed.

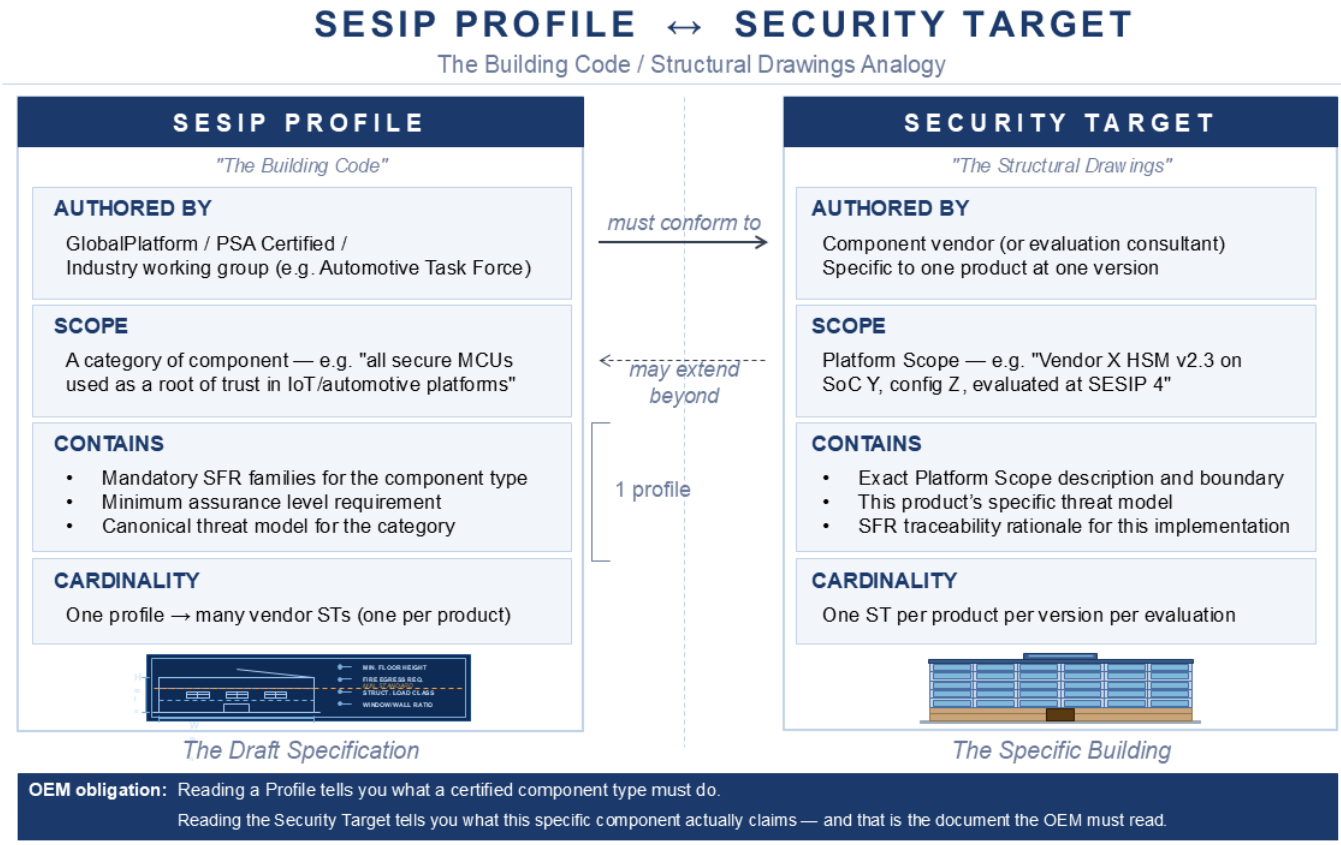

Figure 3: SESIP Profile and Security Target: Building Code and Structural Drawings Relationship

In SESIP terms this means:

- A SESIP Profile defines a generic Security Target for a category of platform or platform part. It specifies the security features required for that category, any mandatory requirements, the minimum assurance level, and the expected threats.
    - For example, a profile for a microcontroller used as a root of trust would capture the baseline security expectations for all platforms of that type.
    - A profile is authored by an entity or group with a stake in that market, such as GlobalPlatform, and is intended to be consumed by multiple platform developers building products of that type.
- A Security Target instantiates those requirements for a specific platform or platform part. The developer selects security features from the SESIP security feature catalogue, maps them to their specific implementation and the threats their platform addresses, and provides the conformance rationale specific to their product.
    - It is authored by the platform developer and is specific to one version of one platform.
- A Security Target that claims conformance to a Profile must satisfy every security feature the Profile requires, but may extend beyond it. A Security Target may claim

additional security features the Profile does not mandate, provided those claims are substantiated in the conformance rationale.

- Where no Profile exists for a component type, the Security Target stands alone.
  - The platform developer selects security features directly from the SESIP security feature catalogue without a pre-agreed baseline.
  - The evaluation laboratory still assesses the Security Target, but without the benchmark that a Profile provides, cross-vendor comparability is more difficult and procurement decisions require more direct technical scrutiny of each individual Security Target.

4.5.2 Published SESIP Profiles Relevant to Automotive

GlobalPlatform has a number of SESIP Profiles related to a class of products and two are particularly relevant to automotive.

SESIP Profile	Scope and Automotive Relevance
MCU/MPU Profile (GPT_SPE_150 v1.1)	Covers microcontrollers and microprocessors used as a root of trust or security anchor. Defines base security properties: secure initialization, key storage, cryptographic services, secure update, and optional feature packages. Applicable to automotive MCUs providing foundational security services below the TEE layer. Updated to v1.1 in May 2025.
Secure External Memory Profile (GPT_SPE_148 v1.1)	Covers external flash and storage components with security properties. Relevant where automotive ECUs rely on external secure storage for firmware or credential storage outside the main SoC.

4.5.2.1 Common Criteria Protection Profiles (GlobalPlatform)

The GlobalPlatform SE and TEE Protection Profiles serve as a composable foundation for a SESIP evaluation of higher-level components, as described in SAE International's Hardware Protected Security Environments (SAE J3101-5).

GlobalPlatform Protection Profile	Scope and Automotive Relevance
Secure Element Protection Profile (GPC_SPE_174)	Covers tamper-resistant hardware components providing cryptographic services, key storage, and secure life cycle management. Directly applicable to Hardware Protected Security Environments and embedded SEs used as the root of trust in automotive ECUs.

GlobalPlatform Protection Profile	Scope and Automotive Relevance
TEE Protection Profile (GPD_SPE_021) Secure Media Path PP-Module (GPD_SPE_090)	Covers Trusted Execution Environments implementing isolation between secure and non-secure worlds. Applicable to TEE implementations on automotive SoCs. The profile includes PP-Modules for extensions such as secure media path; additional modules for automotive-specific TEE use cases are under development.

4.5.3 SESIP Automotive Working Group

The SESIP Automotive Working Group focuses on adapting SESIP to automotive contexts by developing automotive-relevant SESIP Profiles (e.g. for ECUs, sensors, HPC platforms). The first work has focused on automotive CMOS Image Sensors and certification paths for Hardware Protected Security Environments according to SAE J3101.

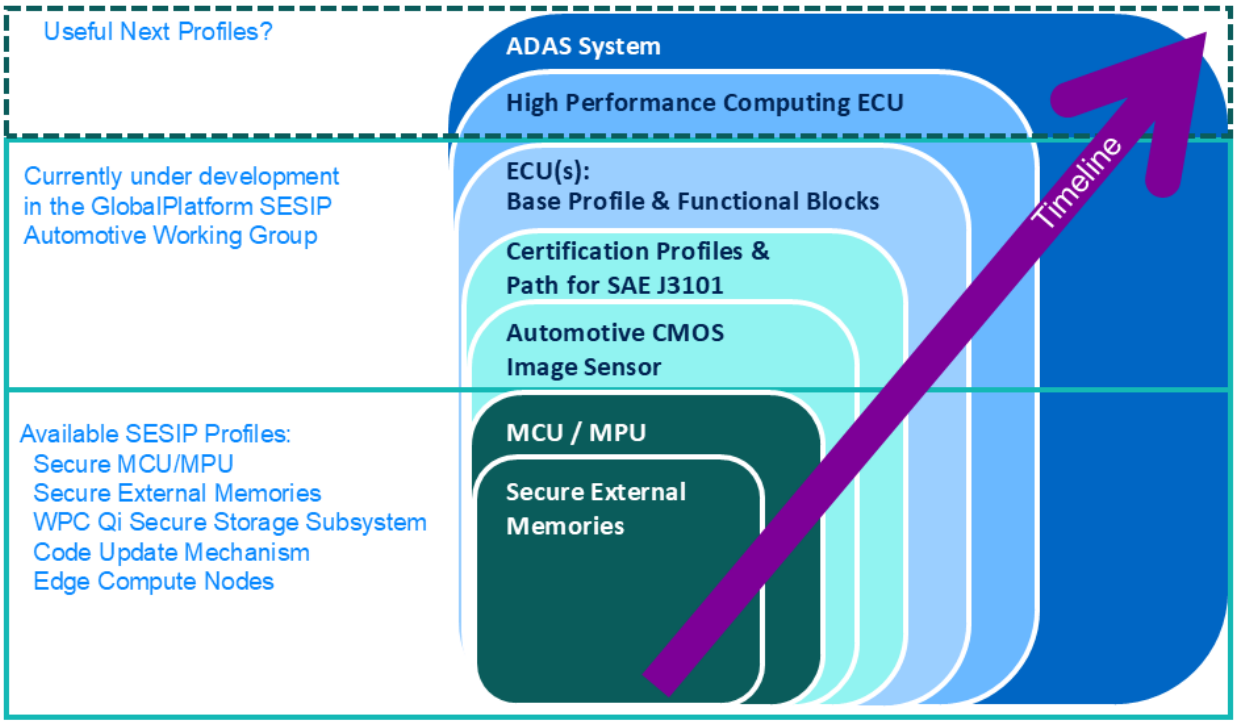

Figure 4: Automotive SESIP Roadmap

4.6 Composition in SESIP

SESIP supports the incremental construction of evaluated platforms from previously evaluated components: referred to as composition. The rationale is practical: Evaluating a complete product from scratch for every deployment is impractical given the pace at which products are assembled from reusable building blocks. Composition allows evaluation work performed once on a platform part to be carried forward into higher-level evaluations without being repeated. SESIP distinguishes between two forms of reuse. The first is the reuse of evaluations performed under external schemes, such as Common Criteria certifications, where SESIP provides guidance on how an existing certificate can be translated into a SESIP Security Target and credited within the SESIP ecosystem. This is the basis for SESIP4 and SESIP5. The second is the reuse of platform parts that have themselves been evaluated under SESIP, which is the subject of the SESIP composition rules.

4.6.1 Environment Objective Inheritance

A composed platform is one that incorporates one or more evaluated platform parts. Each platform part has its own Security Target, which includes a set of security objectives for the operational environment. These environment objectives represent the conditions that the platform part developer assumed would be satisfied by whatever system integrated the part. When a platform is built on top of such a part, the integrating developer must address every one of those environment objectives. For each objective the developer must either:

- demonstrate that it is satisfied by one of the platform's own security requirements, referring to the relevant Security Functional Requirement in the platform's Security Target; or
- carry it forward as an environment objective for the platform's own operational environment, where it becomes a condition that the next integrator in the chain must satisfy.

SESIP is explicit that this chain of environment objectives must be fully resolved at every level. No objective may simply be dropped. The Security Target of the composed platform must contain a dedicated section for inherited objectives, identifying each platform part and accounting for every environment objective it defined. This section is mandatory in the Security Target template and is required for any platform that incorporates previously evaluated parts.

4.6.2 Assurance Level Ceiling

The assurance level of a composed platform is governed by a ceiling rule. By default the composed platform can claim at most the lowest assurance level of any of its constituent parts. This means that if a hardware platform part has been evaluated at a higher assurance level than a software component integrated on top of it, the composition can only be certified to the lower level unless a specific argument is made.

SESIP acknowledges that components evaluated at higher assurance levels, such as Secure Elements and Hardware Protected Security Environments, are often the parts most in need of higher assurance. For those components, existing Common Criteria results can be reused through the SESIP4 and SESIP5 pathways. The ceiling rule therefore creates a direct incentive

to evaluate all significant components at the assurance level required by the composed product.

This has a direct implication for CAL-driven supplier specifications. A component with a CAL3 assignment cannot be supported by a composition whose assurance ceiling sits at SESIP2: The gap is not bridged by the composition mechanism and must be addressed by re-evaluating the weaker component at the required level. The CAL target for an item therefore sets a practical floor for the lowest SESIP assurance level acceptable in any composition that underpins it.

4.6.3 Additive Composition

A single platform may claim different SESIP assurance levels for different subsets of its Security Functional Requirements, provided the Security Target makes the mapping unambiguous. This allows a developer to claim a higher level for the most sensitive functions (such as cryptographic key management) while claiming a lower level for less critical functions, without producing two separate Security Targets.

4.6.3.1 Composition Applied to an Automotive Platform Stack

A modern vehicle electronic architecture presents a natural composition chain that maps closely to the SESIP composition model. Consider a representative stack in a domain controller or central compute platform. The following describes how the composition rules would apply if each layer were evaluated under SESIP and notes the significant caveat that most automotive components have not yet undergone such evaluation.

4.6.3.1.1 The Hardware Foundation Layer

At the base of the stack sits a microcontroller or system-on-chip integrating a Hardware Protected Security Environment. Where such a component has been evaluated, it would naturally serve as the lowest-level platform part in the composition, anchoring the root of trust for the platform above it.

In practice, most automotive microcontrollers have not yet been evaluated under SESIP or any equivalent certification scheme. For those that have, the evaluation has most commonly been at SESIP2 or SESIP3. Dedicated secure elements used for credential storage, vehicle identity management, or certificate handling represent the category most likely to carry a higher-assurance evaluation SESIP5. As regulatory pressure from UN R155 and national type-approval schemes increases, evaluation coverage of automotive microcontrollers is expected to grow, but it remains limited in current production.

Where a hardware component has been evaluated, its Security Target will define environment objectives that the software integrating it must satisfy. Typical examples relevant to an automotive context would include:

- that the software does not re-enable the hardware debug interface after production lock-down
- that key provisioning follows the procedure defined in the hardware guidance documentation

- that the secure boot sequence invokes the hardware root of trust verification before any application code executes

4.6.3.1.2 The Software Platform Layer

Above the hardware sits a software platform layer, which in an AUTOSAR-based architecture would include the real-time operating system, the hypervisor or partition manager if present, and the relevant basic software modules including the Crypto Service Manager, the Key Manager, and the Secure Onboard Communication module. This layer would be evaluated as a platform part in its own right.

If the underlying hardware has been evaluated under SESIP, the software platform Security Target must address every environment objective inherited from the hardware platform part.

For each inherited objective, the developer must either map it to a Security Functional Requirement in the software platform (for example, showing that the secure boot sequence demonstrably invokes the hardware verification mechanism) or pass it upward as an environment objective for the integrating OEM or Tier-1 to satisfy through their production and delivery processes.

The software platform's own Security Target will in turn define new environment objectives for the application layer above it. These might include:

- that the application invokes cryptographic operations only through the Crypto Service Manager interface and does not attempt to access key material directly
- that the software update client follows the update procedure described in the guidance documentation
- that the application does not disable or bypass the Secure Onboard Communication authentication layer

4.6.3.1.3 The Application And Connectivity Layer

At the top of the stack sits a connected application layer, which may include the telematics application, a vehicle diagnostics client, and a V2X communication stack. If this layer is itself subject to evaluation (for example, as part of a product-level assessment under a national automotive cybersecurity scheme or an OEM supplier security requirement), then its Security Target must incorporate the environment objectives cascaded from both the hardware and software platform parts beneath it.

This is where the practical consequence of the inheritance rule becomes most visible to an automotive cybersecurity architect. If the Hardware Protected Security Environment's Security Target specifies that the platform must not enable JTAG access after delivery to the customer, that obligation does not disappear when the software layer is evaluated on top of it. It must either be demonstrably enforced by the software platform as a security requirement (for example, by locking the debug interface as part of the secure boot sequence) or it must appear as a named environment objective in the application layer Security Target, where it becomes an explicit obligation on the OEM or Tier-1 integrator to satisfy through their production, delivery, and vehicle configuration processes.

4.6.3.1.4 Assurance Ceiling In An Automotive Context

The assurance level ceiling rule has direct consequences for automotive certification strategies. A vehicle platform that integrates a SESIP3 evaluated operating system on top of a SESIP2 evaluated hardware component cannot claim SESIP3 for the composition as a whole without the hardware being re-evaluated at SESIP3. This creates a practical alignment with the ISO 21434 approach of assigning cybersecurity assurance levels to components based on their risk analysis outcomes: A component with a high cybersecurity assurance level requirement drives the minimum evaluation assurance level of any composition that depends on it.

4.6.3.2 Additive Composition And Mixed-Criticality Partitioning

The additive composition mechanism is particularly relevant in automotive platforms that host both safety-critical and non-safety-critical functions on the same physical hardware.

A central compute platform might claim SESIP3 for its general-purpose execution environment while claiming SESIP4 for the isolated security partition managing vehicle identity credentials and over-the-air update authentication. SESIP permits this provided the Security Target makes the boundary between the two claims unambiguous. This aligns directly with the hypervisor-based and AUTOSAR-based partitioning approaches commonly used to separate mixed-criticality workloads in modern vehicle architectures, and with the ISO 21434 principle that different cybersecurity assurance levels may be assigned to different items within the same physical component depending on their respective risk profiles.

5 GLOBALPLATFORM'S SESIP CERTIFICATION SYSTEM

5.1 Governance

GlobalPlatform's role in the governance of SESIP includes:

- Publishing and maintaining the SESIP methodology, profiles, mappings, and related guidance
- Defining qualification criteria for SESIP laboratories and certification bodies and maintaining the lists of organizations licensed to operate SESIP-based schemes
- Providing coordination forums (e.g. Attack Working Group, SESIP Committee) to maintain attack-method knowledge, interpret methodological questions, and evolve criteria

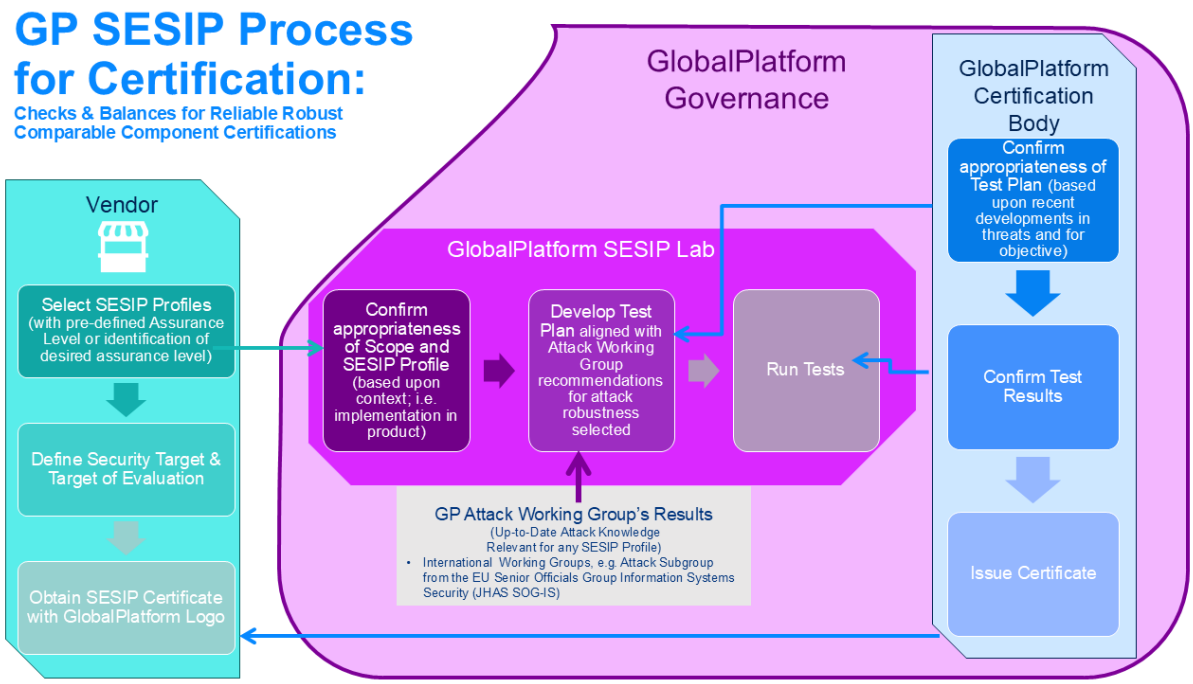

Figure 5: SESIP Process for Certification Governed by GlobalPlatform

This governance is intended to ensure that evaluations performed under SESIP-based schemes are **comparable and technically coherent**, and that updates to attack methods or regulatory expectations can be propagated consistently.

5.2 SESIP Certification Process (high-level)

In a typical certification flow:

1. **Vendor and Scheme Selection**
    - The vendor selects a SESIP-based certification scheme, certification body, and laboratory licensed by GlobalPlatform.
2. **Technical Definition**
    - The vendor and lab agree on the applicable SESIP Profile(s), if available; targeted assurance level; and platform scope, and the vendor prepares a Security Target accordingly.
3. **Evaluation**
    - The lab derives an evaluation plan from the Security Target, performs the required SAR activities (documentation review, testing, vulnerability analysis, etc.), and produces evaluation reports.
4. **Certification Decision**
    - The certification body reviews the evaluation results, checks consistency with scheme rules and SESIP methodology, and issues a certificate with the claimed SESIP level and profile if criteria are met.

Mutual-recognition agreements allow reuse of certification across GlobalPlatform certification bodies or specific markets where such agreements exist.

5.3 Automotive GlobalPlatform SESIP Profiles

Developing a new GlobalPlatform SESIP Profile requires a champion within the GlobalPlatform membership to initiate the process. The dedicated SESIP Automotive Working Group brings forward agreed requirements, scope, and assurance levels. The SESIP Committee reviews the draft profile and, once contents are agreed and revisions addressed, the final draft goes to the GlobalPlatform membership for approval. Upon approval it is added to the official repository; ongoing maintenance is managed by the working group and SESIP Committee.

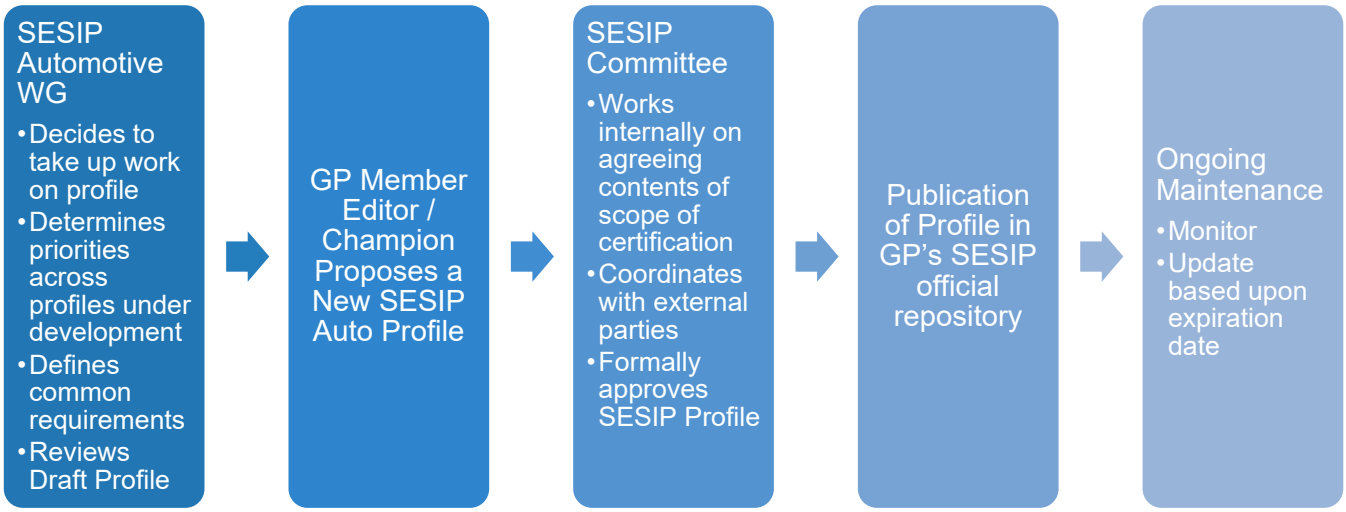

Figure 6: SESIP Profile Development Process

6 SESIP AND AUTOMOTIVE CYBERSECURITY IMPLEMENTATION

6.1 Support ISO/SAE 21434 Process Objectives

ISO/SAE 21434 assigns Cybersecurity Assurance Levels (CALs) to items and allocates those CALs to components. It does not prescribe how a component supplier should prove that their product meets the implied attack resistance. That gap is where SESIP operates: A SESIP certificate is the structured, independently verified answer to the question “does this component actually resist the attack potential its CAL requires?”

In practice, SESIP supports ISO/SAE 21434 objectives at three levels:

- SESIP assurance levels can be used to **parameterize the depth of evaluation** expected for platform parts, associated with items at different CALs; e.g. SESIP2 for components underpinning CAL2 items; SESIP3 for most CAL3 platform components; SESIP4 or SESIP5 for high-assurance CAL3 components such as Secure Elements.
- SESIP Profiles provide **pre-structured threat and requirement sets** for particular component classes, which can be reused when performing item-level TARA and cybersecurity concept activities.
- SESIP evaluation reports contribute **normalized technical evidence** (including vulnerability analysis and test results) that can be referenced in CSMS documentation and type-approval submissions.

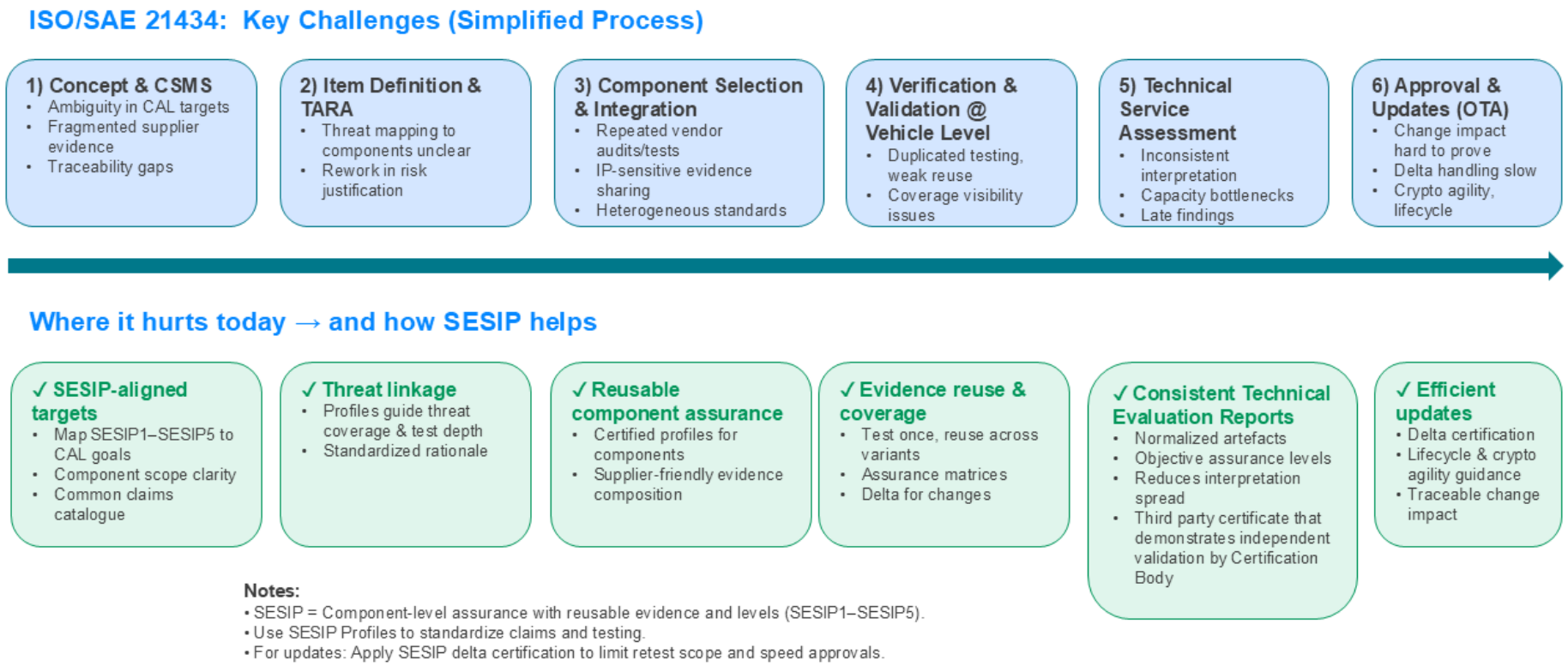

Figure 7: Examples of How SESIP Supports ISO/SAE 21434 Process

6.2 Relationship between ISO/SAE Objectives of Cybersecurity Assurance Levels and SESIP

CAL (ISO/SAE 21434) is an item-level assurance target: It tells the engineering program how rigorous its cybersecurity process and verification must be. It does not tell a component supplier how to prove their product meets the implied attack potential, and it does not produce certificates that cross organizational boundaries.

SESIP fills those gaps. It is a component-level evaluation methodology with defined assurance levels and accredited certificates: the mechanism by which a supplier demonstrates, to any integrator up the chain, that their component resists the attack potential its CAL assignment requires.

CAL: Process Rigor During Engineering Development	SESIP: Evaluation Methodology (also for Certification) for Components
Assurance rigor target (CAL1–CAL3) for an item/system. Dictates development process & test depth.	Component certification scheme with standardized SESIP Profiles and defined Assurance Levels (SESIP1–SESIP5) according to ISO 15408 levels.
Set for an item and allocated to components. No vehicle-wide CAL.	Evaluates a component or stack.
No certificate; produces required evidence to demonstrate rigor.	Accredited lab **certificate** at SESIP Assurance Level against a Security Target (which may be based on a SESIP Profile).
From TARA and impact/feasibility; defined early.	Choose relevant profile(s) and target Assurance Level for the component.
Evidence re-use within program; not a composable certification.	Designed for **composition**; integrators inherit lower-level certifications.
Says **how rigorous** engineering & verification must be	Suppliers **prove** rigor via evaluated components.
Not currently defined what testing procedures and associated attack methods need to be performed.	Includes specific tests for each assurance level with predefined and updated attack methods.

Figure 8: CAL & SESIP Compatibility

The practical consequence for procurement:

- OEMs and Tier-1 suppliers can translate TARA outcomes directly into supplier requirements, specifying the SESIP assurance level in RFQs rather than relying on self-declaration or bespoke audit.
- GlobalPlatform is developing structured guidance to support this translation, in ongoing cooperation with SAE International.

The table below provides guidance on the anticipated relationship between SESIP assurance levels and CAL targets.

SESIP Level	Anticipated CAL	Description
SESIP1	CAL1	Basic functional testing and validation of security requirements.
SESIP2	CAL2	Independent penetration testing (black-box). Minimum level for independent laboratory verification. Appropriate for platform components underpinning lower-risk items.
SESIP3	CAL3	White-box evaluation with increased testing rigor. Covers most automotive platform components today, including bootloaders, OS kernels, and hypervisors.
SESIP4	CAL3	Stronger evidence and life cycle controls: full functional specification and implementation representation. Appropriate for Secure Element firmware and components with V2X/OTA responsibilities. The ISO/SAE JWG TR 8475 requirement for Hardware Formal Methods may also drive SESIP4 selection.
SESIP5	CAL3	Highest assurance intent: aligned to high-assurance hardware components at the root of the trust chain (e.g. Secure Elements, Hardware Protected Security Environment silicon) where compromise would undermine the entire composition.

Note: This mapping is indicative general guidance only. SESIP level selection must always be driven by the specific TARA outcomes and CAL allocation for the item in question. This mapping is subject to further verification through ongoing work.

6.3 Use in Automotive

SESIP can be adopted incrementally, starting where regulatory pressure or supply chain risk is highest. In practice, cybersecurity architects use it in four ways:

- As a **requirements reference** when specifying platform parts in RFQs (e.g. “conformant to SESIP Profile for Secure MCUs and MPUs at SESIP3 or higher”).
- As a **structuring method** for internal security documentation, using Security Targets and SFR catalogues to document security properties of in-house components.
- As a source of structured threat and requirement material:
    - SESIP Profiles for automotive component classes provide pre-built TARA inputs (e.g. threat categories, attack potential assignments, and SFR mappings) that directly support CAL allocation at item level and accelerate the cybersecurity concept phase, not just satisfy it downstream.
- As a **certification path** where formal assurance claims are required for specific components, while still allowing the SESIP methodology to be used informally elsewhere.

6.4 Addressing Common Objections

“Certification will slow my program down.”

With SESIP, the first evaluations require planning, but subsequent vehicle development programs can reuse results. For long-lived platforms and SDV architectures, the amortized cost and delay are typically much lower than bespoke audits for every program.

“Labs don’t understand automotive.”

SESIP Profiles for automotive are being defined jointly by GlobalPlatform members and automotive experts, with specific focus on components such as secure MCUs, memories, and update mechanisms. The GlobalPlatform Attack Working Group keeps threat models aligned with state-of-the-art automotive threats. Some accredited labs are already approved automotive technical services for type approval.

“I just want something that helps me engineer better systems.”

You can adopt SESIP as a design and documentation discipline only. Certificates can remain optional and focused on high-impact blocks or regulatory hotspots.

“We already struggle with ISO/SAE 21434; adding SESIP will increase complexity.”

SESIP is designed to map into ISO/SAE 21434’s concepts, especially CAL, TARA outputs, and CSMS evidence. Rather than adding a new layer of complexity, it can provide better-structured evidence that you already need to produce.

7 CONCLUSION

SESIP is a GlobalPlatform methodology and European standard that specifies **component-level security evaluation requirements for platforms and parts thereof**, with explicit support for **assurance levels, profiles, composition, and reuse.**

For automotive, SESIP provides:

- A direct mechanism for CAL-driven procurement: SESIP certificates translate TARA outcomes into verifiable supplier requirements, giving OEMs and Tier-1s a standard they can specify in RFQs and verify at the organizational boundary.
- A technically defined way to evaluate and reuse platform part assurance across component variants, model years, and supply chain tiers in SDV architectures.
- A governed ecosystem of profiles, mappings, labs, and certification bodies maintained by GlobalPlatform.

SESIP avoids the need to create new automotive-specific evaluation schemes: The methodology exists, the laboratories are licensed, and the automotive profiles are in active development. Structured component-level evaluation will play an increasing role as automotive cybersecurity standards mature. SESIP is built for that role.

8 ANNEX: PRACTICAL GUIDANCE FOR ARCHITECTS

This annex provides some practical tips on how to apply SESIP in automotive.

8.1 Define Platform Scope: Top-Down, Not Bottom-Up

The instinct when scoping a SESIP evaluation is to start from the implementation and draw a boundary around what already exists. In automotive, this approach produces platform scopes that are convenient to evaluate but poorly suited for composition. Instead, work top-down:

- Start at the OEM or Tier-1 level and identify what security properties the vehicle platform must claim, typically driven by UN R155, ISO/SAE 21434, and any applicable type-approval requirements.
- Decompose those claims into the stack layers responsible for each property. Identify where each property originates: which silicon, which firmware, which operating system component.
- Define each layer's platform scope to include exactly the components needed to substantiate its security claims and to expose at its interface exactly the properties the layer above will need to assume.

This produces a set of coordinated platform scopes that compose cleanly under the SESIP inherited objectives mechanism, rather than a set of independently optimized scopes that leave gaps in the composition chain.

Platform Scope

The platform scope is defined in the Security Target and must be depicted in a diagram distinguishing in-scope components from out-of-scope components.

8.2 Map Security Functional Requirements to Architectural Components Early

Each Security Functional Requirement maps to a specific implementation artefact. Performing this mapping during architecture review, before code freeze, allows the development to:

- identify gaps in the design that would prevent a requirement from being substantiated in the conformance rationale
- generate the evidence trail naturally during development rather than constructing it retroactively under evaluation time pressure
- avoid expensive redesigns triggered by evaluation findings that could have been anticipated at the design stage.

In automotive, this mapping should align with the cybersecurity concept phase defined in ISO/SAE 21434: The CAL assigned to an item sets the floor for the required SESIP assurance level, the threat analysis identifies what must be protected, and the SFR selection specifies how the component claims to provide that protection.

8.3 Stabilize Composition Interfaces Before Evaluation

The interfaces between platform layers are the composition boundary. In multi-supplier automotive chains, these interfaces are also organizational boundaries: between the system-on-chip vendor, the Trusted Execution Environment vendor, the operating system vendor, and the Tier-1 integrator. Instability at these interfaces is the primary cause of composition failures in evaluation.

- Define and freeze the security interface at each layer boundary before any evaluation commences. Changes after evaluation starts may trigger scope reassessment and potentially re-evaluation of the affected layer.
- Document all security-relevant interface behaviors in the Security Target as part of the platform scope description, not only in application programming interface reference documentation that evaluators may not treat as normative evidence.

Cascading Effects on Composed Products

A vulnerability in a base certified platform can have a cascading impact. Products that rely on the composition of that platform's certificate may also need to be re-evaluated or have their own certificates reviewed.

Certificate maintenance commitments and certificate life cycle obligations should be addressed contractually with suppliers as part of the cybersecurity management system obligations under ISO/SAE 21434.

8.4 Understand Cybersecurity Evidence

ISO/SAE 21434 and UN R155 place cybersecurity responsibility for the vehicle squarely and non-delegably with the OEM. A supplier's SESIP certificate is evidence of what that supplier's component does and under what conditions those properties hold.

Architects and managers should reflect on these important distinctions:

- The security properties of every component in the stack so as to verify that the composition chain is complete and coherent, including what each layer explicitly does not claim.
- The alignment of security objectives for the operational environment at each layer in the vehicle deployment context. A SESIP certificate is issued against a defined operational environment described in the Security Target. If the vehicle deployment differs from that environment in a security-relevant way, the certificate's claims may not hold for that deployment.
- Gaps between layers. If a lower-layer certificate's platform scope ends at a point that leaves an upper layer's environment objective unaddressed, that gap will not be resolved by the composition mechanism itself.
- Coherent vehicle-level cybersecurity arguments along the full composition chain, which demonstrates traceability from vehicle-level threats to platform-level controls, and satisfies the cybersecurity management system intended functionality. SESIP makes this work more transparent and explicitly linked by structuring the evidence that suppliers provide in a consistent, publicly accessible, and platform-neutral way. The OEM has the opportunity to review the supplier Security Targets, stress-testing the environment objective assumptions, and verifying the coherence of the chain from silicon to vehicle.

Supply Chain Security

SESIP changes how supplier security properties are expressed and verified across organizational boundaries. SESIP provides a means to obtain measurable, structured supply chain certificates that can support the cybersecurity assurance case.

8.5 Understand Threats First, Select Security Functional Requirement Second

Security Functional Requirement selection requires understanding threats (although SESIP does not mandate a documented threat model as an evaluation deliverable at any level).

In automotive, relevant threat models are identified from the UNECE WP.29 threat categories and the ISO/SAE 21434 TARA process. Common platform-level threats and their correspondence to SESIP requirement categories include:

- **Physical access:** Exploitation of debug interfaces such as JTAG and SWD, flash memory extraction, and fault injection against the secure boot chain or the isolated security execution environment. These threats are addressed by the *Resistance to Attack* SFRs described in section 4.2.
- **Firmware tampering:** Loading of unsigned firmware images, rollback to vulnerable software versions, and injection of malicious content during the supply chain. These threats are addressed by the *Identification and Attestation* SFRs and *Product Life Cycle Management* SFRs described in section 4.2.
- **Key extraction:** Side-channel attacks against hardware cryptographic engines, cold-boot attacks targeting volatile key storage, and inter-partition memory leakage. These threats are addressed by the *Cryptographic Services* SFRs and *Compliance and Operational Security* SFRs described in section 4.2.
- **Diagnostic interface abuse:** Escalation of UDS diagnostic session privilege and impersonation of authorized diagnostic tools. These threats are addressed by the *Access Control* SFRs and secure debugging SFRs described in section 4.2.
- **Vehicle-to-everything credential compromise:** Extraction of long-term pseudonym certificates and linkability attacks against pseudonym identities. These threats are addressed by the *Cryptographic Services* SFRs and *Secure Communication* SFRs described in section 4.2.

9 ABOUT GLOBALPLATFORM

GlobalPlatform is a leading industry association that develops open and globally adopted standards for the secure deployment and management of digital services and devices. For over 25 years, GlobalPlatform technologies have enabled trusted interactions across billions of devices in markets including mobile, payments, identity, IoT, and automotive.

Through its SESIP framework, GlobalPlatform simplifies and scales cybersecurity certification by defining reusable, composable evaluation methods and domain-specific profiles. The organization works closely with SAE International, ISO, ENISA, and others to promote harmonized, risk-based security assurance across industries.

The Automotive Task Force (ATF) within GlobalPlatform unites OEMs, Tier-1 suppliers, semiconductor vendors, laboratories, and regulators to identify requirements for strategic security standardization in automotive. One of the key areas includes assessing SESIP for automotive use cases: for a practical, interoperable certification ecosystem that accelerates compliance with UNECE R155/R156 and ISO/SAE 21434, while fostering global trust in connected vehicles. For more information: **www.globalplatform.org**

10 REFERENCES

10.1 GlobalPlatform SESIP Resources

GlobalPlatform SESIP overview page

GlobalPlatform Technology: SESIP Methodology

GlobalPlatform Technology: SESIP FAQ

GlobalPlatform Technology: Cryptographic Algorithm Recommendations

GlobalPlatform Specifications Library: all SESIP technical documents

10.2 Standards and Regulations Referenced / Mentioned

EN 17927:2023: Security Evaluation Standard for IoT Platforms (SESIP) (info page)

UNECE UN Regulation No. 155: Cyber security and CSMS (UNECE page)

UNECE UN Regulation No. 156: Software update and SUMS (UNECE page)

Regulation (EU) 2024/2847: Cyber Resilience Act (EUR-Lex ELI)

ISO/SAE 21434:2021: Road vehicles, Cybersecurity engineering (ISO)

ISO/SAE PAS 8475: CAL/TAF (SAE)

SAE J3101-5: Hardware Protected Security Environment (SAE)

AUTOSAR: Explanation of Security Overview (PDF)

ISO/IEC 15408-1:2022: Common Criteria, Part 1 (ISO)

ISO/IEC 18045:2022: Common Criteria Evaluation Methodology (ISO)

ISO/IEC 17025:2017: Testing and calibration laboratories (ISO)

ISO/IEC 17065:2012: Requirements for product certification bodies (ISO)

IEC 62443-4-2:2019: Technical security requirements for IACS components (IEC Webstore)

NIST Consumer IoT Cybersecurity program (NIST)

NIST IR 8425: Profile of the IoT Core Baseline for Consumer IoT Products (NIST CSRC)

11 GLOSSARY

Attack potential: A qualitative/quantitative notion used in evaluation to describe the effort and capability needed by an attacker to exploit a vulnerability.

CAL (Cybersecurity Assurance Level): An assurance-rigor target for an item/system used in automotive cybersecurity engineering to set expectations for process/verification depth.

Certificate: A formal attestation issued by a certification body stating that a target has met a specified evaluation level/profile under a given scheme.

CSMS (Cybersecurity Management System): Organizational processes and governance used to manage cybersecurity risk across a vehicle life cycle and supply chain.

ECU (Electronic Control Unit): An embedded computing unit in a vehicle (e.g. domain controller, zone controller, gateway, sensor ECU).

Hardware Protected Security Environment (HPSE): A hardware-enforced isolated execution and storage environment providing security services to the platform, as defined in SAE J3101.

OEM: Original Equipment Manufacturer (vehicle manufacturer).

OTA (Over-the-Air): Remote update or provisioning of software/firmware and related configuration via network connectivity.

Profile (SESIP Profile): A reusable, generic Security Target template for a platform class defining model/configuration, required SFRs, and targeted SAR packages/levels.

SAR (Security Assurance Requirement): Assurance activities and evidence requirements defining how confidence is established (e.g. documentation review, testing, vulnerability analysis).

SDV (Software-Defined Vehicle): A vehicle architecture emphasizing software platforms, workload consolidation, and frequent updates across a long life cycle.

Secure Element (SE): A tamper-resistant secure component (often discrete) providing protected execution/storage for sensitive assets such as keys and credentials.

Security Target (ST): The primary technical document describing the platform, its security problem definition, objectives, and claimed requirements at a chosen assurance level.

SESIP: A methodology for evaluating platforms and platform parts with defined assurance levels (SESIP1–SESIP5) and reusable profiles.

SFR (Security Functional Requirement): A structured functional security requirement describing a capability or behavior a target provides (e.g. secure storage, identification, update).

TARA (Threat Analysis and Risk Assessment): A structured activity used to identify threats, assess risk, and select controls/requirements.

Target of Evaluation (TOE): Common Criteria term. In SESIP the equivalent concept is platform scope, defined in the *Platform Functional Overview and Description* section of the Security Target.

TEE (Trusted Execution Environment): An isolated execution environment providing hardware-backed separation between trusted and untrusted software layers.

Tier-1: A supplier that delivers systems or modules directly to an OEM (often integrating semiconductor and software components).

Vulnerability analysis: Systematic analysis to identify, assess, and prioritize weaknesses, typically combined with testing and flaw-handling processes.