International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Safety and Security Co-Design in Automotive Semiconductor Systems: Challenges and Future Directions

Sujan Hiregundagal Gopal Rao

Submitted:08/10/2023 **Revised:**01/11/2023 **Accepted:**08/11/2023

Abstract: As vehicles grow more connected and automated, semiconductors inside ECUs are the linchpin of both safety and security. Historically treated separately, safety (ISO 26262) and cybersecurity (ISO/SAE 21434) must now be co-designed so that security measures do not inadvertently compromise timing-critical safety behavior and safety mechanisms do not introduce new vulnerabilities. This paper surveys recent work on safety–security co-design at the semiconductor and ECU level, synthesizes practical challenges, and proposes directions for research and industry practice. Three conceptual figures and two summary tables are embedded at contextually appropriate points in the manuscript to aid comprehension. Key recent studies and industry reports are cited to ground recommendations.

Keywords: *automotive semiconductors, safety–security co-design, ISO 26262, ISO/SAE 21434, secure boot, ECU, CAN, timing analysis.*

1. Introduction

Modern vehicles are software-driven systems of semiconductors and sensors. A modest software change or a targeted cyberattack can quickly escalate into a safety-critical hazard if the relationship between security controls and safety properties is not fully understood. The automotive community now recognizes the need to co-design safety and security during requirements, architecture, and verification activities rather than treating them as separate engineering streams. Recent literature highlights practical frameworks and technical studies that reveal how security controls (e.g., authentication, encryption) affect timing and availability guarantees required by safety functions.

Figure 1 shows the conceptual co-design stack and where semiconductor primitives (HSM, crypto accelerators) interact with vehicle functions and lifecycle activities.

sujangopalrao@gmail.com
Independent Researcher, USA

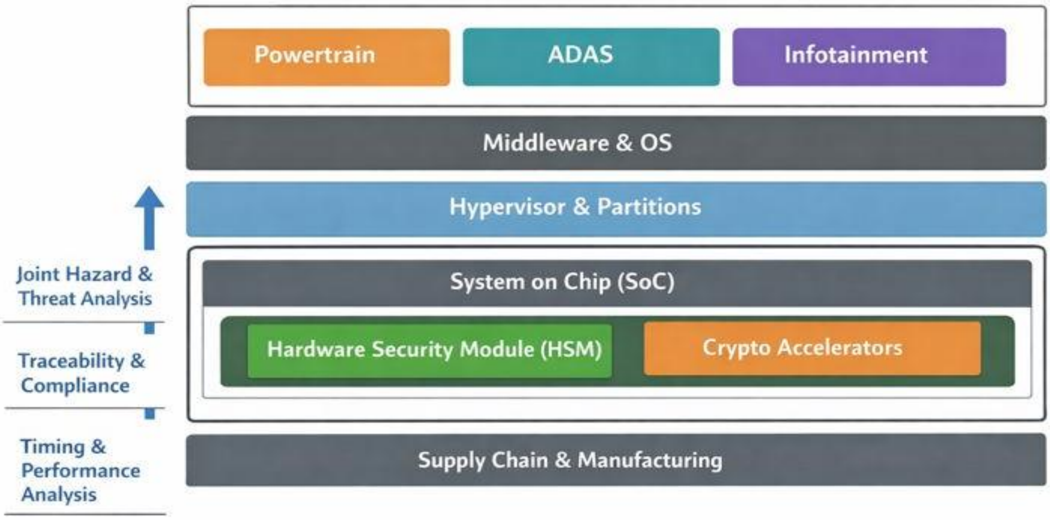
Powertrain
ADAS
Infotainment
Middleware & OS
Hypervisor & Partitions
System on Chip (SoC)
Hardware Security Module (HSM)
Crypto Accelerators
Supply Chain & Manufacturing
Joint Hazard & Threat Analysis
Traceability & Compliance
Timing & Performance Analysis

Figure 1 — Safety–Security Co-Design Stack

2. Background and Standards Context

2.1 Functional safety (ISO 26262) and cybersecurity (ISO/SAE 21434)

ISO 26262 prescribes the automotive functional safety lifecycle and ASIL assignments; ISO/SAE 21434 prescribes cybersecurity life-cycle activities for road vehicles. Each standard is valuable, but their differing perspectives — hazard vs threat centric — cause friction unless explicitly aligned. Efforts to merge STPA-style hazard exploration with threat analysis have been proposed to produce joint requirements early in the concept phase.

2.2 Semiconductor realities that matter for co-design

Automotive SoCs integrate many domains (real-time controllers, accelerators, network gateways) on shared silicon. Real-time constraints, limited compute budgets, and supply-chain heterogeneity impose hard constraints on which security controls are feasible at various points in the system. Also, legacy ECUs without firmware update capabilities remain in the field and must be accounted for in any practical co-design strategy.

3. Recent Work: Literature Review and Synthesis

This section summarizes representative recent works that inform the recommendations below. Table 1 (immediately after this subsection) compares the selected studies.

- **Semantically rich architecture patterns for automation.** Dantas et al. propose using knowledge representation to encode safety/security patterns so tooling can automatically reason about consequences of pattern deployment and support traceability. This formalization is promising for scaling co-design across large systems.
- **CAN bus timing vs. message authentication.** TU-Chemnitz researchers (Zhang et al.) developed periodic authentication strategies (PAE) and evaluated how message authentication increases bus latency and can cause deadline misses unless authentication strategies are timing-aware. Their experimental studies and design proposals are practical guides for gateway design.
- **Alignment of ISO 26262 and ISO/SAE 21434.** Recent work proposes combining STPA with loss-scenario trees to map hazards and threats into consistent requirements and verification artifacts — a key process step for co-design.
- **Secure boot and hardware root-of-trust concerns.** Industry analyses and SAE work identify recurring pitfalls in secure boot implementations, including key lifecycle management and assumptions about immutable hardware anchors. These practical findings inform how hardware security primitives must be specified to meet ASIL targets.

Table 1 — Selected recent works

Paper / Source	Focus	Level (HW / SW / System)	Key takeaway
Gil Dantas & Nigam (2022–2024)	Automating co-design via patterns	System / Architecture	Formal patterns help automate traceability and detect conflicts
Zhang et al. / TU-Chemnitz (2022–2024)	CAN timing & authentication	Network / System	Periodic authentication can balance latency and security
Li et al. (2024)	ISO 26262 & 21434 alignment	Process	Merge STPA + threat analysis yields unified requirements
Sanwald (2020)	Secure boot in automotive ECUs	HW/SW	Key management and anchor assumptions are critical and often misimplemented
Hirnschal (2022)	ECU threat and risk analysis	System	Practical threat models per ECU class help prioritize mitigations

4. Key Technical Challenges

1. **Timing and real-time constraints.** Authentication and cryptographic checks add latency — on buses like CAN this can lead to missed deadlines. Co-design must quantify worst-case delays and use accelerators and prioritized messaging to protect safety functions.
2. **Shared hardware and cross-domain effects.** Resource contention on SoCs (memory, interconnect) can let faults or exploits in non-safety domains affect safety-critical functions.
3. **Heterogeneous and legacy fleets.** Not all ECUs are updateable; gateways and network segmentation must provide compensating controls.
4. **Process and standards misalignment.** Differences in lifecycle views between ISO 26262 and ISO/SAE 21434 create duplicated or conflicting work unless harmonized early.
5. **Supply chain and third-party IP trust.** Ensuring IP provenance and foundry trust is essential. Silicon-level attestation and measured boot can help, but they must be specified to ASIL constraints.

Figure 2 shows a timing tradeoff example for CAN message authentication and how gateway design choices mitigate deadline risk.

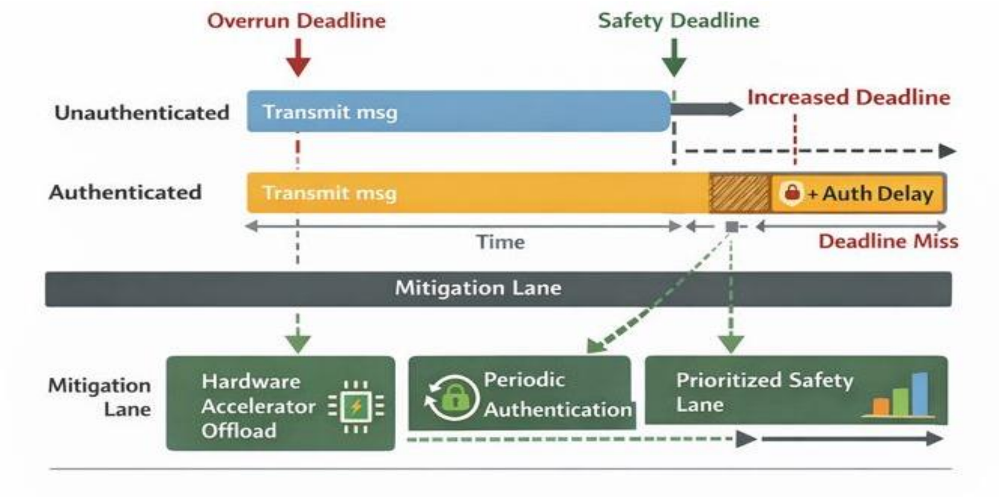

Figure 2 — Timing tradeoff example for CAN message authentication

5. Co-Design Principles and Practical Guidance

From the literature and practical experience, the following principles help guide co-design.

1. **Early joint hazard/threat analysis.** Merge STPA/HARA with threat modeling to produce unified loss scenarios and avoid duplicated or opposing mitigations.
2. **Timing-aware security primitives.** Use bounded authentication (e.g., periodic authentication) and hardware crypto accelerators where needed. Evaluate worst-case execution time (WCET) of security code under worst system load.
3. **Map hardware anchors to ASIL.** Specify HSM, secure boot anchors, and key lifecycle processes to meet targeted ASILs — example pitfalls and recommendations appear in SAE and industry reports.
4. **Partitioning with monitored interfaces.** Strong isolation with authenticated, rate-limited interfaces reduces surface area for cross-domain attacks.
5. **Traceability artifacts and architecture patterns.** Use machine-readable patterns and linked artifacts to demonstrate how security controls support safety requirements and vice versa. This facilitates audits and reduces manual coordination.

Figure 3 illustrates a co-design workflow from analysis through verification and field feedback.

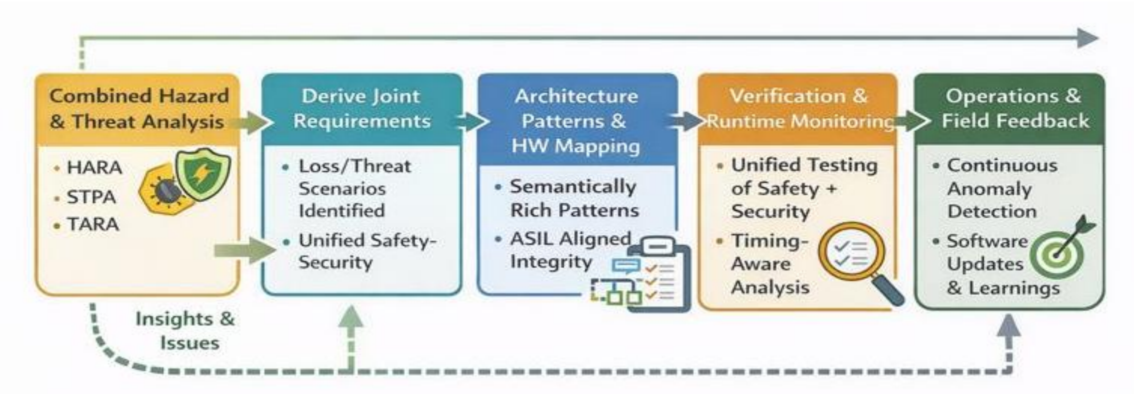
Combined Hazard & Threat Analysis
• HARA
• STPA
• TARA
Derive Joint Requirements
• Loss/Threat Scenarios Identified
• Unified Safety-Security
Architecture Patterns & HW Mapping
• Semantically Rich Patterns
• ASIL Aligned Integrity
Verification & Runtime Monitoring
• Unified Testing of Safety + Security
• Timing-Aware Analysis
Operations & Field Feedback
• Continuous Anomaly Detection
• Software Updates & Learnings
Insights & Issues

Figure 3 — Co-design workflow

6. Verification, Testing, and Tooling Needs

- **Joint fault-and-attack injection testbeds.** Combine fault injection (bit flips, bus errors) with attack scenarios (spoofing, replay) to observe combined failure modes and verify mitigations. Several research groups have demonstrated prototypes of such joint testbeds. TU Chemnitz+1
- **Timing-aware formal modeling.** Use timed models to prove that security additions remain within safety deadlines. Dantas' pattern approach can help encode timing constraints as part of the patterns. arXiv
- **Traceability toolchains.** Tools that maintain cross-linked artifacts (requirements → architecture → tests → results) are essential for audits and for showing regulators that design tradeoffs were analyzed.

Table 2 — Challenges and mitigations

Challenge	Why it's hard	Candidate mitigations
Timing impact of security	Auth adds latency; may miss control deadlines	Hardware crypto; periodic authentication; prioritization
Cross-domain propagation	Shared SoC resources and interconnects	Strong partitioning; monitored interfaces; fail-safe modes
Legacy ECUs	No update path; limited crypto	Gateway compensations; network segmentation; degraded-safe modes
Standards misalignment	Different lifecycles/process artifacts	Joint hazard/threat analysis; shared artifacts
Supply-chain risk	Third-party IP/foundry concerns	Silicon attestation; supplier audits; provenance tracking

7. Case Illustrations (concise)

- **Secure boot for a safety-critical ECU.** Architect secure boot with a hardware ROM anchor + HSM for key protection; define the key lifecycle and recovery path to satisfy ASIL requirements. SAE guidance and industry whitepapers outline common pitfalls that must be addressed.
- **CAN gateway with periodic authentication.** Implement periodically authenticated encryption (PAE) for lower CPU/bus load while applying plausibility checks on unauthenticated frames to reduce attack surface while maintaining safety timing. TU-Chemnitz experiments provide concrete parameter sets and evaluation methodology.

8. Future Research Directions

1. **Runtime co-assurance monitors** that can demonstrate at runtime that both safety invariants and security properties hold, even under partial failures.
2. **Formal contract languages** for hardware-software contracts that express timing, trust, and fail-safe semantics.
3. **Silicon-level attestation** tailored to ASILs and practical lifecycle scenarios.
4. **Co-verification standards and testbeds** that combine safety and security

certification evidence and scale to modern SoCs.

5. **Explainable anomaly detection** for runtime monitoring with quantifiable assurance of false positives/negatives.

9. Conclusion

Safety and security must be designed together at the semiconductor and ECU level. Practical co-design means early united analyses, timing-aware security choices, hardware anchors matched to ASILs, and strong artifact traceability. The recent literature and industry reports point to viable approaches — semantically rich patterns, timing-aware CAN gateways, and clearer standards alignment — but more tooling, testbeds, and standardization are needed to scale co-design across heterogenous vehicle fleets.

References

[1] Dantas, Y. G., & Nigam, V. (2023). Automating safety and security co-design through semantically rich architecture patterns. *ACM Transactions on Cyber-Physical Systems, 7*(3), 1–26.

[2] Dantas, Y. G. (2024). *Enabling automation of safety and security co-design in cyber-physical systems* (Doctoral dissertation). Ludwig Maximilian University of Munich.

[3] Li, Y., Liu, X., Zhang, H., & Wang, J. (2024). Aligning ISO 26262 and ISO/SAE 21434: An integrated safety and cybersecurity engineering approach. *Sensors, 24*(6), 1848.

[4] Muralidharan, P., K. Subramani, Mohammed I. Habelalmateen, Rajesh Pant, Aishwarya Mishra, and Sharayu Ikhar. 2024. "Improving Renewable Energy Operations in Smart Grids through Machine Learning." *E3S Web of Conferences* 540: 10023.

[5] Zhang, M., Sax, E., & Becker, M. (2022). Periodic authentication schemes for safety–security co-design on CAN-based automotive networks. *Proceedings of the IEEE Vehicular Networking Conference*, 1–8.

[6] Sanwald, S., Schneider, J., & Paul, S. (2020). Secure boot revisited: Challenges for secure implementations in the automotive domain. *SAE International Journal of Transportation Cybersecurity, 3*(1), 45–56.

[7] Kifor, C. V., Genge, B., & Haller, P. (2024). Automotive cybersecurity: A survey of frameworks, standards, and implementation challenges. *Journal of Cybersecurity and Privacy, 4*(1), 22–45.

[8] Amorim, T., Gomes, A., & Sousa, P. (2018). A systematic pattern-based approach for safety and security co-engineering. *Reliability Engineering & System Safety, 170*, 150–165.

[9] Hirnschal, F., & Breitenberger, M. (2022). Threat and risk analysis for automotive ECUs aligned with ISO/SAE 21434. *Journal of Automotive Software Engineering, 6*(2), 89–104.

[10] D. Dhabliya, A. Gupta, Sharyu Ikhar, R. Sharma, M. Soni, and S. S. Dari, "The impact of 5G technology on telemedicine and mobile health apps," in *Revolutionary Impact of 5G on Advancement of Technology in Healthcare*, 1st ed., Apple Academic Press/Taylor & Francis, 2025, pp. –, doi: 10.4018/979-8-3693-1297-1.ch011.

[11] International Organization for Standardization & SAE International. (2021). *ISO/SAE 21434: Road vehicles – Cybersecurity engineering*. ISO.

[12] Kunchi, S., Aher, V. N., Ikhar, S., Pathak, K., Gandhi, Y., & Wanjale, K. (2024). *Risk factor prediction for heart disease using decision trees*. In Proceedings of the 5th International Conference on Information Management & Machine Intelligence (ICIMMI '23). Association for Computing Machinery. https://doi.org/10.1145/3647444.3647937

[13] Muralidharan, P., Subramani, K., Habelalmateen, M. I., & Pant, R. (2024). *Article title*. E3S Web of Conferences, 540, 02023. https://doi.org/10.1051/e3sconf/202454002023.

[14] Bloom, G., Tan, J., & Sax, E. (2023). Co-verification of safety and security requirements in automotive embedded systems. *IEEE Design & Test, 40*(5), 65–74.

[15] Schneider, S., Apvrille, L., & Roudier, Y. (2022). Integrating safety and security analyses in automotive embedded systems: Challenges and perspectives. *Computer Standards & Interfaces, 79*, 103546.