RUNSAFE SECURITY

Driving Security:

Safeguarding the Future of Automotive Software

RunSafeSecurity.com

Table of Contents

Executive Summary

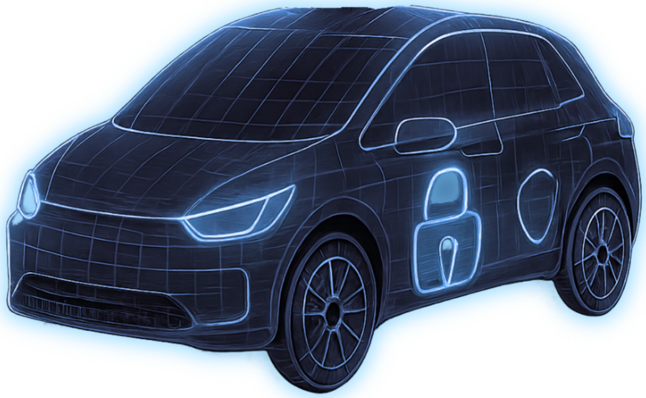
Automotive vehicles are highly complex, software-defined platforms, relying on intricate networks of embedded systems and digital interfaces. With over 100 million lines of code now common in advanced vehicles, the attack surface has grown exponentially. Software innovation has enabled increased automation, enhanced safety, and new features, but it has also elevated the risk—and complexity—of cybersecurity threats.

The automotive industry faces distinct challenges: connected vehicles operate in complex, dynamic environments, involve a vast array of third-party and open-source components, and must adhere to stringent regulatory requirements.

Automotive engineers, security professionals, product teams, compliance officers, and decision-makers who understand the risks will be able to advocate for integrating security as a tenet of both engineering practice and organizational culture, leveraging standards compliance, proactive risk management, and industry collaboration to drive continuous improvement in automotive cybersecurity.

In this white paper, we will cover:

- An overview of the automotive software landscape
- Security challenges in securing modern vehicles and real-world incidents
- Tactical approaches to strengthening software supply chain security and software integrity
- Mitigation techniques and collaborative models to improve vehicle security and safety



Introduction

The Evolution of Automotive Software

The last two decades have marked a transition from primarily mechanical vehicles to those increasingly defined and powered by software. Modern automotive architectures integrate hundreds of electronic control units (ECUs), advanced driver assistance systems (ADAS), infotainment consoles, and cloud-enabled telematics. High-end vehicles routinely feature OTA (over-the-air) update mechanisms and always-on network connectivity.

According to a 2023 McKinsey report, by 2030, the global automotive software and electronics market is expected to reach $462 billion, representing a 5.5 percent CAGR from 2019 to 2030[1]. Additionally, the complexity of in-vehicle software has doubled since 2006 and is expected to further increase fourfold by 2030[2].

This transformation delivers unprecedented customer value—features can be enhanced post-purchase, vehicles can adapt to driver preferences, and predictive diagnostics elevate safety and efficiency. However, this digitalization also introduces vulnerabilities endemic to all software-centric systems: code defects, insufficient patch management, third-party component risks, increased remote attack surfaces, and new regulatory exposures.

The Stakes

Compromised automotive software poses risks that extend beyond data theft to immediate safety concerns, including loss of vehicle control or operational sabotage. Incidents such as the 2015 Jeep Cherokee hack and more recent infotainment system exploits have underscored the potential ramifications, including large-scale recalls, loss of brand reputation, and legal action. As regulators and consumers demand greater accountability, the imperative to embed security across all stages of development and the supply chain has never been more urgent.

[1]McKinsey & Company, 2023: "Outlook on the automotive software and electronics market through 2030" https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/mapping-the-automotive-software-and-electronics-landscape-through-2030

[2]McKinsey & Company, 2017: "The Automotive Revolution Is Speeding Up" https://www.mckinsey.de/~/media/McKinsey/Industries/Automotive%20and%20Assembly/Our%20Insights/The%20automotive%20revolution%20is%20speeding%20up/The-automotive-revolution-is-speeding-up.pdf

The Current Landscape of Automotive Software

Getting Software-Defined Vehicles (SDVs) Right

SDVs represent a fundamental shift in vehicle architecture, as software orchestrates everything from vehicle dynamics to user experience. Research from Capgemini studies show that the average luxury vehicle contains over 150 ECUs, and most vehicles today surpass 100 million lines of code spanning powertrain, braking, steering, and environment-sensing modules.[3,4] The complexity is compounded by dependencies among components, interactions with external devices (such as smartphones and roadside infrastructure), and the modular adoption of advanced AI-based systems.

"It's so difficult for car companies to get software right. We have about 150 modules across the car, developed by 150 different companies, written in more than 100 different languages, that don't talk to one another… And we can't even understand it all."

—Jim Farley, CEO of Ford

Telematics accounted for 66% of total cyber incidents in 2024, and API-related attacks rose to 17% of incidents, followed by infotainment systems and ECUs.

Connected Services Are a Primary Target for Attackers

The proliferation of wireless interfaces—Bluetooth, Wi-Fi, cellular LTE/5G, V2X (vehicle-to-everything)—enables enhanced functionality such as real-time navigation, remote lock/unlock, and OTA updates. However, each integration serves as a potential ingress point for adversaries. Vulnerabilities in insufficiently isolated infotainment or telematics platforms can cascade to critical systems. Vehicles delivered with weak authentication, inadequate encryption, or outdated firmware have been successfully targeted by attackers employing remote exploits.

Upstream's 2025 Automotive Cybersecurity Report found that "cyber threats are rapidly shifting toward software-defined components, telematics systems, and cloud-based infrastructure." Telematics accounted for 66% of total cyber incidents in 2024, and API-related attacks rose to 17% of incidents, followed by infotainment systems and ECUs.[5] This underscores that connected services are now a primary target for attackers.

[3]Capgemini, 2024: "Software-driven Mobility" https://www.capgemini.com/wp-content/uploads/2023/02/Software-Driven-Mobility-POV_July-2024.pdf

[4]Capgemini, 2022: "Next Destination: Software How automotive OEMs can harness the potential of software-driven transformation" https://www.capgemini.com/wp-content/uploads/2022/02/Next-Destination-Software-_WEB.pdf

[5]Upstream, 2025: "Mind the Cyber Gap: Key Insights from Upstream's 2025 Automotive Cybersecurity Report" https://upstream.auto/blog/insights-from-upstreams-2025-automotive-cybersecurity-report/

Regulatory Challenges

Global regulations continue to evolve in response to these risks. ISO/SAE 21434 and UNECE WP.29 (R155 and R156) mandate the implementation of comprehensive cybersecurity management systems (CSMS) for vehicle development and lifecycle management, including incident response and software patching processes. Adhering to these guidelines requires cross-team coordination, continuous vulnerability assessment, and formalized processes for documenting software lineage and change management.

OEMs are now required to submit evidence such as threat analysis, risk assessment (TARA), and penetration testing outcomes as part of the homologation process—a significant shift from historical industry practice.

Automotive Cybersecurity Incidents

- **Jeep Cherokee (2015):** Vulnerability in Uconnect infotainment module exploited via cellular network, leading to remote control of steering and brakes.
- **Tesla Bluetooth Exploit (2022):** Exposure of cryptographic implementation flaws in Bluetooth led to the possibility of unauthorized vehicle access.
- **Subaru Starlink & Nissan Leaf (2025):** A flaw in Subaru's Starlink system allowed for remote access to sensitive customer data and vehicle functions. A similar flaw in the Nissan Leaf infotainment platform allowed unauthorized access and privilege escalation, providing attackers with a path to compromise other critical vehicle systems.
- **PerfektBlue Attack (2025):** Researchers discovered critical vulnerabilities in the BlueSDK Bluetooth stack that could have allowed remote code execution on car systems of millions of vehicles.

Cybersecurity Challenges in the Automotive Industry

Unlike traditional IT systems, automotive networks (CAN, LIN, FlexRay) were originally designed for reliability and low latency, not security. Attackers exploit this by pivoting from less critical subsystems (e.g., through infotainment or OBD-II ports) to safety-critical ECUs. Additionally, the dramatic increase in software has led to a need for software supply chain security and secure development practices to find and address vulnerabilities.

Software Supply Chain Security

The automotive industry's reliance on complex, multi-tiered supply chains increases exposure to software risks that third-party vendors, open-source libraries, or off-the-shelf commercial software may introduce. Recent high-profile incidents—ranging from compromised development dependencies to malicious code insertion—underscore the importance of rigorous supply chain oversight as a board-level concern for OEMs and suppliers.

Supply chain security challenges include:

- Component Origin Traceability: Difficulty identifying the origin and authenticity of all software modules within embedded automotive systems.
- Vendor Assurance: Inconsistent security practices across vendors result in uneven risk levels.
- Timely Vulnerability Response: Delays in patching or updating components from external providers extend exposure windows.

Memory Safety Vulnerabilities

Memory safety defects (buffer overflows, use-after-free, integer underflows, heap corruptions) remain the leading source of vulnerabilities in embedded software, including vehicle ECUs, ADAS, and other systems. Languages commonly used in embedded automotive development, such as C and C++, lack built-in memory protection and require rigorous manual validation to ensure security.

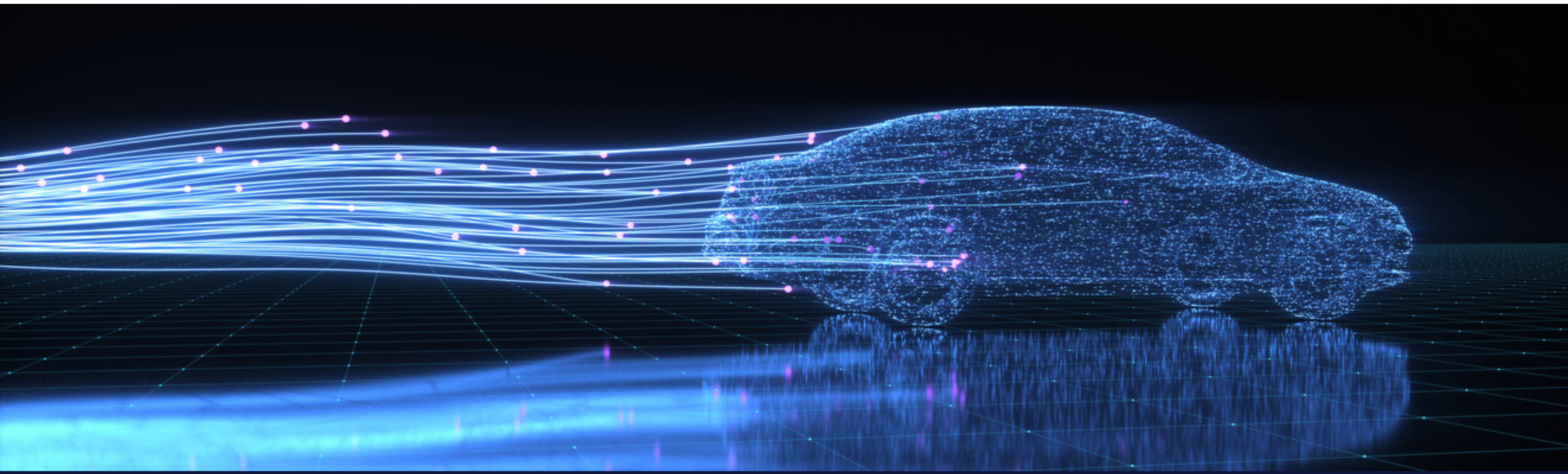
Actionable Measures:

- Transition critical modules to memory-safe languages (e.g., Rust, Ada) where feasible
- Employ compiler-level mitigations (stack canaries, control-flow integrity, load-time function randomization)
- Perform mandatory static and dynamic analysis on all codebases, especially before integration or field updates

Regulatory Pressures

Compliance with standards like ISO 26262 (functional safety for road vehicles) and ISO/SAE 21434 (cybersecurity risk management for road vehicles) is now a prerequisite for market access, particularly in the European Union and Japan. These standards require:

- Security risk evaluation at the system, hardware, and software levels
- Traceable and auditable security processes
- Regular penetration testing, vulnerability disclosure, and incident management



The Impact of Cybersecurity Breaches

Safety Risks

Unlike other software industries, failure in automotive systems can result in direct harm to users or bystanders. Attack scenarios include:

- Remote disabling of brakes or engine shutdown while in motion
- Manipulation of dashboard or ADAS outputs, compromising situational awareness
- Unauthorized deployment of airbags or emergency safety features

A single safety-critical vulnerability affecting a widely deployed model could impact millions of vehicles, potentially leading to accidents, injuries, or fatalities. In recognition of the need to focus on cybersecurity and the concurrent safety risks, the National Highway Traffic Safety Administration (NHTSA) and other federal agencies regularly research and offer guidance on vehicle cybersecurity and safety.

Financial and Reputational Costs

Cyber incidents compel automakers to conduct large-scale recalls, sometimes at the cost of hundreds of millions of dollars per incident. Research from VicOne found that the estimated cost of automotive cyberattacks rose from about $1 billion in 2022 to $22.5 billion in 2024[6].

Implications for Innovation

Major security events slow product development cycles, stall the introduction of new features, and consume engineering resources with unplanned remediation. With the increasing automation and autonomy of future vehicles, the stakes and potential damages will only escalate.

[6] VicOne, 2025 : "Shifting Gears: VicOne 2025 Automotive Cybersecurity Report" https://documents.vicone.com/reports/shifting-gears-2025-automotive-cybersecurity-report.pdf

Best Practices for Securing Automotive Software

1. Implement Software Development Lifecycle Practices

Over the past decade, the industry has seen dramatic improvements in the software development lifecycle. Automotive manufacturers should continue to enhance these practices by:

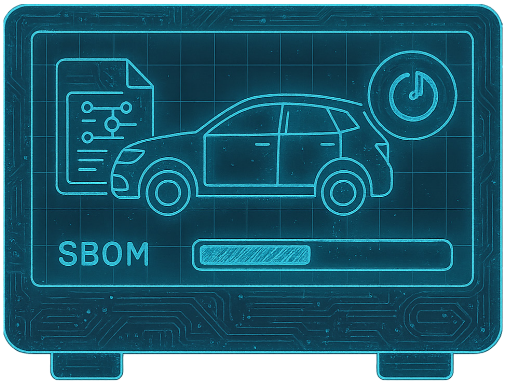
- Implementing secure coding standards
- Conducting thorough threat modeling
- Utilizing automated vulnerability detection tools
- Integrating security protections into the development process

Software Supply Chain Security & SBOMs

With the complexity of automotive software supply chains, addressing and mitigating third-party software risks and risks from open source software is crucial. An SBOM is a detailed inventory of all software components, libraries, modules, and their dependencies integrated into a vehicle's system. Its use is rapidly becoming a regulatory and industry best practice, driven by requirements from bodies such as the US Cybersecurity and Infrastructure Security Agency (CISA) and the EU's Cyber Resilience Act.

Benefits of SBOMs:

- **Component Visibility:** Provides a structured, transparent view of every software element, including open-source and third-party modules.
- **Automated Vulnerability Tracking:** Facilitates the correlation of components with public vulnerability databases, enabling the rapid identification and remediation of vulnerabilities.
- **Regulatory Compliance:** Supports adherence to ISO/SAE 21434, UNECE WP.29, and other standards mandating software provenance, risk reporting, and auditability.
- **Incident Response:** Streamlines investigation and impact analysis by rapidly pinpointing affected systems when new vulnerabilities are discovered.
- **Supply Chain Transparency:** Empowers OEMs to demand more rigorous security controls and disclosure from their vendors.



Best Practices for SBOM Adoption:

- Require suppliers to provide Software Bills of Materials (SBOMs) as part of all software procurement and integration contracts.
- Conduct security assessments of third-party components.
- Establish transparent communication channels with suppliers.
- Regularly update SBOMs throughout the software lifecycle, not just at product release.
- Leverage SBOM management tools that integrate directly with CI/CD pipelines and vulnerability scanning solutions.

For automotive manufacturers and OEMs, a software supply chain security program anchored by SBOMs is essential for operational integrity, long-term brand protection, and regulatory compliance.

Deploy Runtime Protection and Memory Hardening Techniques

One of the most effective security strategies is implementing runtime memory protection at the software build stage. This approach eliminates the entire class of memory-based vulnerabilities, which account for 40-70% of all vulnerabilities in embedded systems, without requiring code rewrites or adding system overhead.

Hardening memory can block attempts at privilege escalation, code injection, or lateral movement between vehicle components. Advanced solutions, such as Load-time Function Randomization (LFR), eliminate the ability for attackers to exploit memory safety vulnerabilities, dramatically reducing the attack surface of embedded systems.

Leverage OTA Updates

OTA update capabilities provide a mechanism for addressing vulnerabilities and deploying security patches throughout a vehicle's lifecycle, allowing manufacturers to continuously improve security posture even after vehicles have been deployed to customers.

40-70%

of all vulnerabilities in embedded systems are memory-based.

Case Studies

Tesla Model 3 Exploit

CVE-2022-42431 is a critical vulnerability discovered in Tesla Model 3 vehicles. The vulnerability is classified as a classic buffer overflow, specifically affecting the bcmdhd driver. The vulnerability allows local attackers to escalate privileges on affected Tesla vehicles and potentially execute arbitrary code on the target system.

Lessons Learned:

- Memory safety vulnerabilities pose a significant risk in embedded software, including vehicle systems written in C or C++.
- Proactive memory hardening would have prevented the feasibility of the attack.

Bluetooth Exploit as an On-Ramp to Critical Systems

At the Pwn2Own Vancouver 2023 hacking contest, researchers from the French cybersecurity firm Synacktiv successfully compromised a Tesla Model 3 in under two minutes using two separate zero-day exploits.

- First exploit: The team targeted the vehicle's Gateway energy management system with a time-of-check-to-time-of-use (TOCTTOU) vulnerability, allowing them to transmit arbitrary CAN bus messages. This enabled actions such as opening the front trunk or doors while the car was in motion.
- Second exploit: Synacktiv exploited a heap overflow and out-of-bounds write vulnerability in the infotainment system's Bluetooth chipset, gaining root access to the system and demonstrating the ability to overwrite the IVI display with custom images.

Lessons Learned:

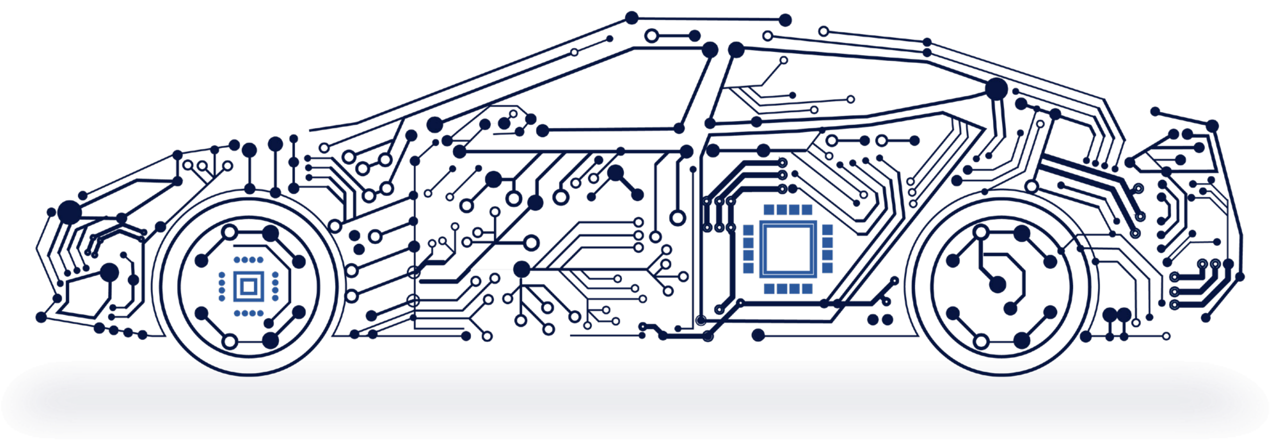
- By chaining these vulnerabilities, the researchers achieved deep subsystem access and remote code execution, illustrating the risks posed by complex, interconnected vehicle systems.
- Vehicle cybersecurity and safety go hand-in-hand. A vulnerability can quickly become a serious safety risk.
- Automotive manufacturers should collaborate with and coordinate with vendors and software suppliers to gain a clear understanding of risks in the software supply chain and take steps to address them.

The Road Ahead

Securing the automotive software ecosystem must remain an ongoing, adaptive process, particularly as new technologies like AI, V2X communication, and other technologies are added to the equation.

- Prioritize security in organizational culture, from executive planning to daily engineering.
- Foster industry-wide cyber threat intelligence sharing.
- Advocate for continuous regulatory and standards improvement in response to emerging threats that prioritize both safety and security.
- Increase the budget and focus on software security to enable innovation alongside vehicle resilience.

Leadership at every level—OEM, supplier, regulator, researcher—must collaborate to ensure safe, resilient automotive systems.



Software Supply Chain Security and Software Protection with RunSafe

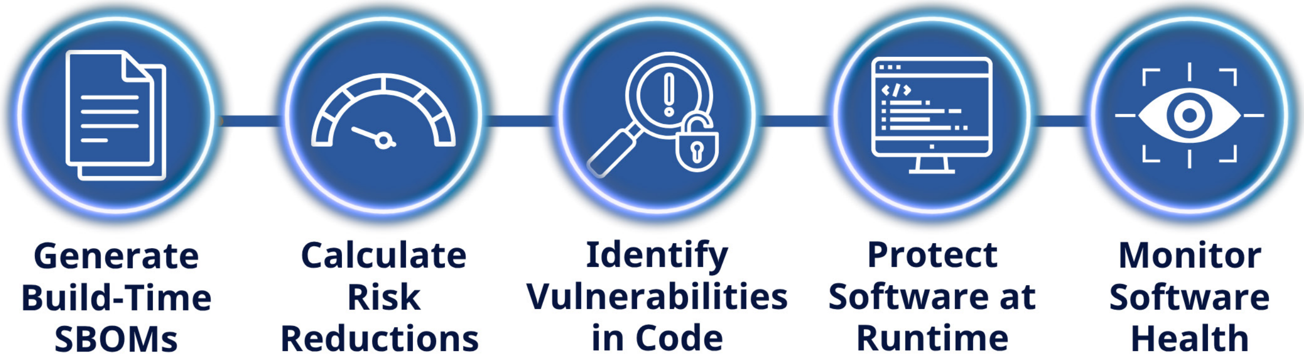
RunSafe Security automatically mitigates known and unknown vulnerabilities in connected and autonomous vehicles and generates comprehensive build-time SBOMs. Our technology seamlessly integrates with automotive standards and safeguards critical vehicle functions to protect products and customers from cyber-attacks.

C/C++ SBOMs

RunSafe's SBOM generation capabilities provide deep visibility into embedded software written in C/C++ without requiring a package manager. By generating SBOMs at build time, RunSafe captures only the components and libraries in a build, providing an accurate picture of your embedded software and reducing the likelihood of false positives and negatives. Accurate SBOMs are critical for regulatory compliance and rapid vulnerability identification.

Code Protection

RunSafe deploys our patented code protection, called Load-time Function Randomization (LFR), to defend automotive software against attacks. By relocating software functions in memory every time the software is run, RunSafe creates a unique memory layout that prevents attackers from exploiting memory-based vulnerabilities. As a result, automotive software can defend itself against one of the largest classes of vulnerabilities in embedded systems without affecting performance, even before a patch is available.



ABOUT RUNSAFE SECURITY, INC.

RunSafe Security protects embedded software across critical infrastructure, delivering automated vulnerability identification and software hardening from build-time to runtime to defend the software supply chain and critical systems without compromising performance or requiring code rewrites. The RunSafe Security Platform includes the authoritative build-time SBOM generator for embedded systems and C/C++ projects, automated vulnerability identification and risk quantification, patented memory relocation techniques to mitigate memory-based vulnerabilities, and pre-hardened open-source packages and containers for immediate protection.

Headquartered in McLean, Virginia, with an office in Huntsville, Alabama, RunSafe Security's customers span the aerospace and defense, energy, operational technology, industrial automation, transportation and automotive, medical device, and high-tech manufacturing verticals.

RunSafeSecurity.com

571.441.5076

Sales@RunSafeSecurity.com