DerScanner

Full-cycle application security testing solution to secure automotive software – from embedded systems to connected services.

A modern vehicle runs on over 100 million lines of code spread across multiple electronic control units (ECUs), each sourced from different suppliers. Software now governs many safety-critical points – braking, steering, adaptive cruise control – as well as the infotainment systems, telematics platforms, mobile apps and cloud APIs that keep vehicles connected. Attacks on automotive software are accelerating.

530+

CVEs in automotive software, 2024

49

0-days at Pwn2Own Automotive 2025

100M+

Lines of code in a modern vehicle

49.5%

of auto cyberattacks exploit software

DerScanner is a full-cycle application security testing platform (SAST, DAST, SCA, Supply Chain Security) that enables automotive OEMs and their suppliers to analyze proprietary code in 43 programming languages, audit open-source dependencies with automatic SBOM generation, and inspect third-party binary components without source code access – all from a fully on-premise, air-gapped deployment.

Core Challenges Facing Automotive Software Security

Challenge 1

Regulatory Pressure With No Room for Delay

UN Regulation 155 became mandatory for all new vehicle types in July 2024 – without a certified Cybersecurity Management System, vehicles cannot be sold in 64 countries.

ISO/SAE 21434 references static code analysis as part of software verification (Section 10.4). Japan's METI SBOM guidance and the EU Cyber Resilience Act add further requirements for software transparency across the entire supply chain.

Challenge 2

Vulnerabilities Across the Software Stack

ECU firmware in C/C++ suffers from memory-safety flaws (buffer overflows, null pointer dereferences).

Infotainment systems built on Java, Kotlin and JavaScript expose injection and authentication weaknesses.

Cloud backends and telematics APIs – written in Go, Python, PHP – leak data and enable remote access (Kia 2024, Subaru 2025, Volkswagen Cariad). Mobile apps for vehicle owners introduce yet another attack surface.

Challenge 3

Hidden Vulnerabilities in Supply Chain

OEMs integrate firmware from dozens of Tier-1 and Tier-2 suppliers – typically as compiled binaries without source code.

Researchers have found vulnerabilities as old as 10 years (Heartbleed, Dirty COW) still present in production automotive components.

Without visibility into what runs inside each ECU, every supplier handoff is a blind spot.

DerScanner

How DerScanner Addresses Each Challenge

Secure In-House Code

Use DerScanner as the main engine for application security assessments across all systems:

- **SAST for C/C++** ECU firmware with inter-module analysis and MISRA-relevant CWE detection
- **SAST for Java/Kotlin/JS** in infotainment and Android Automotive
- **SAST for Go/Python/PHP** backend and telematics services
- **Incremental scans in CI/CD (GitHub, GitLab, Bitbucket)** for shift-left security

Verify Components

Assess the security of third-party deliverables from suppliers – even without access to their source code:

- **Binary Analysis for Java, Android and Scala** applications from Tier-1/ Tier-2 suppliers
- **MAST for security assessment** of companion mobile applications (iOS/ Android)
- **Unified vulnerability reports** across own code and supplier deliverables
- **Results mapped to CWE/ OWASP** for standardized supplier benchmarking

Control Open Source Risk

Gain full visibility into embedded open-source components and track risks before they reach production:

- **SCA with SBOM generation (CycloneDX)** for JS/TS, Python, Java, Go, Rust, C/C++ and 6 more
- **Hybrid SAST+SCA** for call-graph analysis showing reachability of vulnerable functions
- **Supply chain scoring** across 8 risk metrics (MavenGate, Typosquatting, Starjacking)
- **License risk analysis** for compliance with OEM procurement policies
- **Continuous monitoring** of deployed dependencies against new CVE disclosures

UN R155/ R156

SBOM generation and vulnerability tracking support CSMS documentation

ISO/ SAE 21434

Helps fulfill static analysis requirements referenced in Sec.10.4

CWE/ SANS Top 25

All findings mapped to standard weakness enumerations

PCI DSS/ HIPAA

Compliance reports for connected-car payment and health data

OWASP Top 10 / MASVS

Web and mobile findings aligned to OWASP classifications

EU Cyber Resilience Act

Software composition data supports CE marking documentation

MITRE CWE-Compatible

Officially recognized by MITRE; listed in NIST recommendations

METI SBOM Guidance (Japan)

CycloneDX SBOM output aligned with software transparency requirements

Why DerScanner

Wide language coverage

Analyze code across the entire software stack without switching tools – from embedded C/C++ and infotainment Java/Kotlin to backend Go/Python and mobile Swift/Dart, including legacy languages most AST vendors do not support.

On-Premise & air-gapped deployment

Run DerScanner in your own infrastructure or deploy it on client premises when policies require on-site scanning and data residency control. Cloud-free, air-gapped scenarios are supported.

Binary analysis without source code

Scan compiled binaries from Tier-1 and Tier-2 suppliers to detect vulnerabilities when source code is not available. Supports OEM acceptance testing and third-party component verification.

Correlated Static + Dynamic Analysis

Automatically correlate SAST and DAST findings in a single interface. Trace each vulnerability from code-level root cause to runtime behaviour without manual cross-referencing.

AI-Powered Triage (DerTriage)

Automatically prioritize findings by exploitability and business impact. Security teams can focus on actionable risks first when scanning large codebases.

Hybrid SAST + SCA With Reachability

Build a call graph from application code to transitive dependencies, confirming which vulnerabilities are actually reachable – reducing false positives and triage workload.

Supply Chain Security Scoring

Evaluate open-source libraries on 8 health metrics to detect dependency confusion, typosquatting, and repository hijacking before risky packages enter your build pipeline.

White-Label Reporting & CI/CD Integration

Generate branded PDF/HTML/CSV reports under your organisation's logo. Integrate with GitHub, GitLab, and Jira to automate scanning on every commit.

DerScanner is a MITRE certified and NIST recommended solution.

With over a decade of expertise in application security, DerScanner provides powerful analysis capabilities in an easy-to-start setup.

NIST MITRE

Contact us at company@derscanner.com