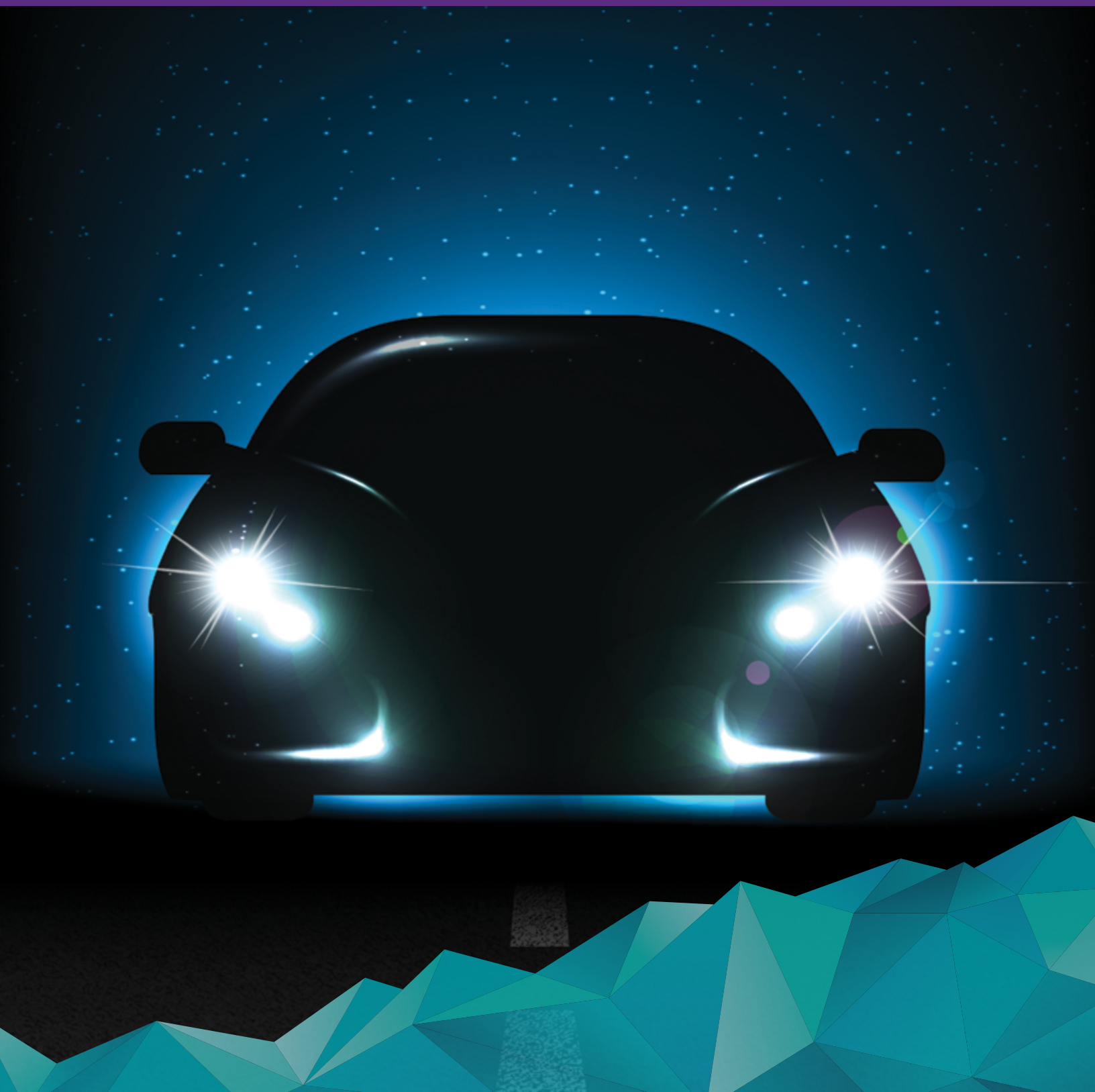
SYNOPSYS®

It's Not Your Father's Connected Car...



Since the automobile was first introduced to the public in the late 19th century, it has gone through a series of transformations. What was once merely a mechanical transport system built upon a single cylinder has evolved into a rolling computer designed by engineers and developers alike, running on over a million lines of code.

With every release, technological advancements have made vehicles much safer to drive; antilock braking systems, electronic stability controllers, backup sensors, and voice controls all have contributed to making our roads safer. Many of those systems now rely heavily on complex software, leading to the inevitable question, *What is this code composed of?* For most organizations, the answer is *a significant amount of open source software.*

In the following guide, we will discuss the security risks that are associated with the software within connected cars and introduce development and vulnerability management techniques that are being adopted by leaders in the automotive industry. By integrating these practices earlier and throughout the software development life cycle, you can maximize the benefits of open source software while effectively managing its risks.

What type of car is at risk?

Today, any car on the road can contain unpatched software vulnerabilities, putting it at exploitation risk. Because automakers have been focusing their attention on differentiating features, the disparity between innovation and security is growing at an accelerated speed. Open source software is at the heart of this race and enables developers to spend less time reinventing the wheel and more time on innovative, differentiating features.

BMW's iDrive, Tesla's Model X, Apple CarPlay, and Android Auto have turned a car's dashboard into a technological hub for entertainment, navigation, cellular communication, intelligent climate control, and more. Because these features are often built on a core of open source, software security vulnerabilities hidden within the code can put the entire automotive system at risk. To ensure the safety of the passengers and the performance of the vehicle, automakers must vet the code before, while, and after the vehicle goes to market.

Whether the comprised code is proprietary or open source, no software is immune to the risk of security vulnerabilities. Managing these risks is extremely challenging given the complexity of the automotive supply chain, and doesn't end once the vehicle goes to market. The OEM is responsible for the security of each vehicle it manufactures and for issuing patches to vulnerabilities over the course of the vehicle's lifetime.

Software defects and security vulnerabilities have led to costly recalls, as experienced in 2015 by Fiat Chrysler, when the company was forced to recall 1.4 million of their vehicles.

Hackers redefine the “car crash”

As the number of connected vehicles on the road increases, federal entities are placing pressure on manufacturers to focus on safety in addition to entertaining features. With government mandated technologies like tire-pressure sensors, electronic stability controls, dual front airbags, and passive restraint systems, they are doing their part to keep passengers safe. These systems, albeit safety oriented, have left out one important safety factor on the road: *Drivers and passengers aren’t the only people able to cause a crash.*

This is evidenced by the high-profile hacking of a Jeep SUV in 2015, during which hackers were able to connect through the Uconnect dashboard, gain access to mechanical functions, as well as safety features, and take complete control over the vehicle. The recall that followed didn’t require every Jeep that was affected by the vulnerability to be brought back to dealerships. Instead, Fiat Chrysler elected to send out individual USB flash drives to each of their customers and entrust that the end users would supply the update themselves. This approach solved some problems but created others.

Flash drives and over-the-air updates are not always possible, or practical, for automakers. Few automakers have been able to supply their customers with regular updates, and automakers in the past few years have only begun to address how to enact this technology in legacy vehicles. With companies like IHS Automotive projecting that automakers would save $35 billion from over-the-air updates by 2022, automakers need to push to start enacting safety practices surrounding these code updates. Additionally, vehicle manufacturers need to adopt a cyber security approach that addresses not only obvious exposures in their cars’ software but also the hidden vulnerabilities that can be introduced by open source components in that software.

To avoid a car “crash” and a costly recall, automotive suppliers should be continually vetting their code to ensure its safety and impenetrability. Governments around the world have begun to help the automotive industry in establishing these practices. In the United States, the NHTSA (National Highway Traffic Safety Administration) has been enabling and supporting the safe development of connected vehicles by focusing on cyber security to ensure that these systems work as intended. Their *Cybersecurity Best Practices for Modern Vehicles* eBook outlines voluntary best practices that would help prevent exploitation of vulnerabilities. Unless an organization is aware that a vulnerable open source component is in its software, it’s highly probable that the component will remain unpatched and open to exploit. Visibility into and control over open source are essential to maintaining the security of automotive software applications.

In 2015, hackers Chris Valasek and Charlie Miller redefined what a computer crash could entail. With a driver voluntarily behind the wheel of a Jeep, the duo hacked in through the Uconnect dashboard through a software vulnerability, adjusted the temperature, blared music from the infotainment center, and finally cut the transmission from more than 10 miles away.

As the Jeep came to a screeching halt, the roadmap of the automotive industry was changed forever. The exhibition of the seriously vulnerable code in this supposedly "safe" vehicle has since brought forth new protective measures within the automotive industry.

The new routine maintenance schedule

Software updates occur more frequently than updates to physical components. Think of the average cell phone; your cell phone has a life of 2 or 3 years. During this time, software updates are continuously pushed out to it by carriers, handset manufacturers, and application providers. This is the maintenance schedule typical in the software and telecommunications industries. Since cars have an average life-span of 10–15 years, ensuring automotive security through routine software maintenance is a challenge.

In the case of auto manufacturers, their products rely on hundreds of independent vendors supplying them with both hardware and software components. Software from each vendor is likely to be a mix of custom code and open source components.

Because the rate of software iteration and innovation can easily outpace traditional supply chain quality control practices, companies looking to stay ahead of these risks must adopt agile mechanisms for tracking and managing their software supply chain.

What can I do?

Over the last 100 years, innovations in manufacturing automation and integrated end-to-end quality management have been hallmarks of the most successful automotive manufacturers and suppliers. As software and internet connectivity revolutionize the makeup of the modern vehicle, companies must apply these same innovations to the management of automotive software supply chains. This is especially important for the management of open source components. As an auto manufacturer or supplier to the automotive industry, you must establish policies and procedures to manage the open source going into the vehicle, not only to ensure quality but most importantly to ensure the safety and security of drivers and passengers. These practices should:

- Integrate open source management into the software development process, and then automate it.
- Ensure that software suppliers track and report on open source components and vulnerabilities.
- Create a software-specific bill of materials (BoM) of all open source used in the applications within the vehicle.
- Proactively monitor for and remediate new open source security threats as they are reported.

For over 15 years, Synopsys has been helping organizations in all industries leverage open source to fuel innovation, while minimizing the security and IP risks that can come from its use. Our products and services are the industry standard for open source risk management and M&A due diligence. Let us help your organization bring new connected and autonomous vehicle solutions to market more quickly, reliably, and securely by taking control of your open source.

SYNOPSYS®

The Synopsys difference

Synopsys helps development teams build secure, high-quality software, minimizing risks while maximizing speed and productivity. Synopsys, a recognized leader in application security, provides static analysis, software composition analysis, and dynamic analysis solutions that enable teams to quickly find and fix vulnerabilities and defects in proprietary code, open source components, and application behavior. With a combination of industry-leading tools, services, and expertise, only Synopsys helps organizations optimize security and quality in DevSecOps and throughout the software development life cycle.

For more information, go to www.synopsys.com/software.

Synopsys, Inc.
185 Berry Street, Suite 6500
San Francisco, CA 94107 USA

Contact us:
U.S. Sales: 800.873.8193
International Sales: +1 415.321.5237
Email: sig-info@synopsys.com