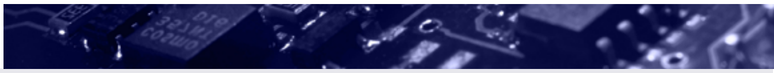
Managing Open Source Risk with Device Composition Analysis for Connected Devices



Open source risks and rewards

Development teams work hard to quickly and efficiently produce the components used in connected devices. Increasingly, this means using open source software, libraries, and operating systems to aid in that efficiency.

While there are many benefits to using open source components, they also present a fair amount of risk. Not only could these components be insecure, but they may be accompanied by licensing rules that may affect your organization from a legal standpoint. The risk is especially high if you're sourcing your device components from a vendor, because you often don't know which components are being used or how secure their development processes are.

What is Device Composition Analysis?

This is where **Device Composition Analysis (DCA)** comes into play. You can think of DCA kind of like Software Composition Analysis (SCA) used in app development, except that DCA is compatible with the embedded systems and architectures found in connected devices. DCA enables you to uncover and track all third party components in your devices as well as their software licenses and vulnerabilities.

Why don't existing SCA tools work on devices?

SCA tooling is built for use on Apps. Because devices are fundamentally different from apps device security approaches and tooling must be fundamentally different from those used in application security.

While an app is a singular program, a device is an entire system that may contain hundreds of programs along with hundreds or thousands of configuration files and settings. It relies on a technology stack (including hardware, bootloaders, OS components, and drivers). Traditional Appsec tools perform source code SCA, but they lack the ability to perform binary analysis which is imperative to evaluating the security of connected devices and the complex array of components within them. Naturally, this more complex ecosystem requires a more complex solution.

The Finite State Platform uses our robust DCA process to allow our users to:

Identify open source vulnerabilities

Gain access to a robust Software Bill of Materials (SBOM) and a list of known vulnerabilities, all automatically generated when you upload your device firmware.

Analyze supply chain risk & dependencies

Any given device component may rely on other components, which also have licenses and potential security issues. With the Finite State platform, you can view mismatches in license types, discover vulnerabilities in upstream dependencies, and see where your device components originate.

Manage open source licensing

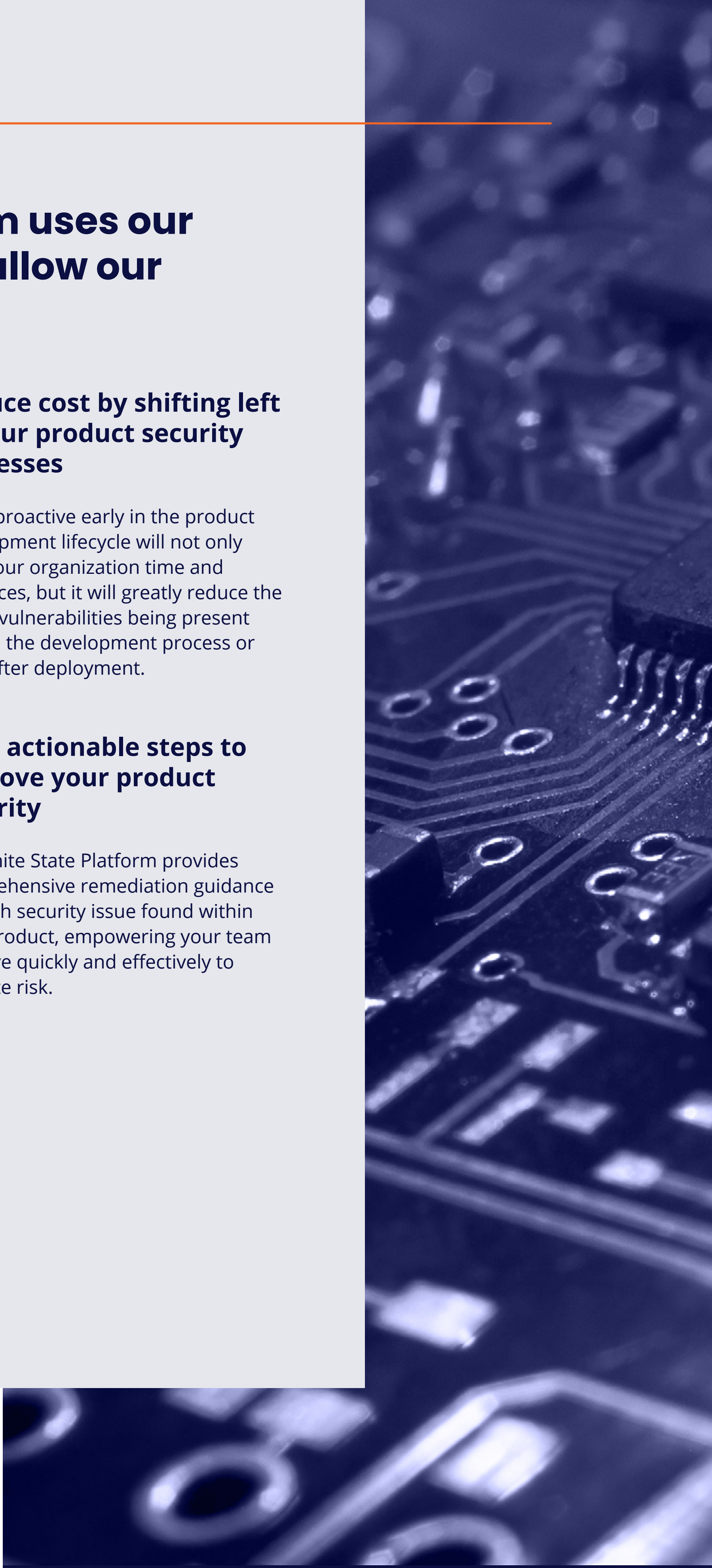
Make it easy on yourself and your legal team. The Finite State Platform displays and exports licensing information for each open source component used in your product.

Reduce cost by shifting left in your product security processes

Being proactive early in the product development lifecycle will not only save your organization time and resources, but it will greatly reduce the risk of vulnerabilities being present later in the development process or even after deployment.

Take actionable steps to improve your product security

The Finite State Platform provides comprehensive remediation guidance for each security issue found within your product, empowering your team to move quickly and effectively to mitigate risk.



@FiniteStateInc
/company/FiniteState
www.finitestate.io

FINITE STATE