The Ultimate Guide to Connected Device Security

Six steps to secure products and software supply chains

FINITE STATE

Nearly 70% of organizations surveyed by the Linux Foundation report being very or extremely concerned about the security of the software they use. When that software powers critical infrastructure systems in sectors such as energy, telecom, or health care, the stakes to society rise high.

Cyber attackers are constantly seeking new entry points when they target their victims.

Even if you have designed and implemented seemingly impenetrable product security and risk management controls, what about your suppliers? What happens when you purchase products with connected devices, embedded systems, and open-source code manufactured by other organizations or even individuals with smaller or non-existent risk management budgets? In your own product security control processes, do you have to take binaries at face value?

When you cannot see inside all the components of the products in your asset inventory, your software supply chain risk (and that of your customers too) grows. And so does your risk of a cybersecurity attack.

"Supply chain risks are the next big thing," says PwC in a recent industrial products publication on manufacturer cybersecurity and supply chain. Indeed, PwC research indicates that 60% of leaders in the manufacturing sector anticipate third-party threats growing this year. Fifty-eight percent expect a growing number of reportable incidents at the software supply chain level.

What is product security? Supply chain security?

Product security includes the totality of the efforts developers and manufacturers use when they build secure products. Product security represents a critical process when products are created. In other words, it's not a bolt-on verification added at the end of the process before shipping containers are sealed shut.

The product security measures taken by each of your upstream manufacturers represent your supply chain security—whether it's good or bad. When your suppliers (and their suppliers) embrace a strong product security strategy, you get a more secure product, whether you sell it as a final product or a component to the final product of a downstream supply chain partner. Regardless, whether you manufacture connected products or buy, deploy, and manage them–you own this risk. Even when third parties introduce product security risk, you cannot outsource your responsibility to find and address it.

Product security cannot exist without supply chain security. Whether you are the first (or only) link in your supply chain, even one vulnerable embedded software stack can make your software highly vulnerable to threat actors. Often, that vulnerability—wherever it lies along that the software supply chain—represents a world of unseen risk, unseen to product security teams, chief product security officers, risk management professionals, and the people who rely on the resulting connected devices to function safely and fulfill their intended purpose.

Among proactive product security tools, most organizations have looked to software composition analysis (SCA) as their cybersecurity focal point for 2022, followed by static application security testing (SAST), and dynamic application security testing (DAST). To build to an optimal product security program, however, organizations must also adopt more comprehensive tools such as:

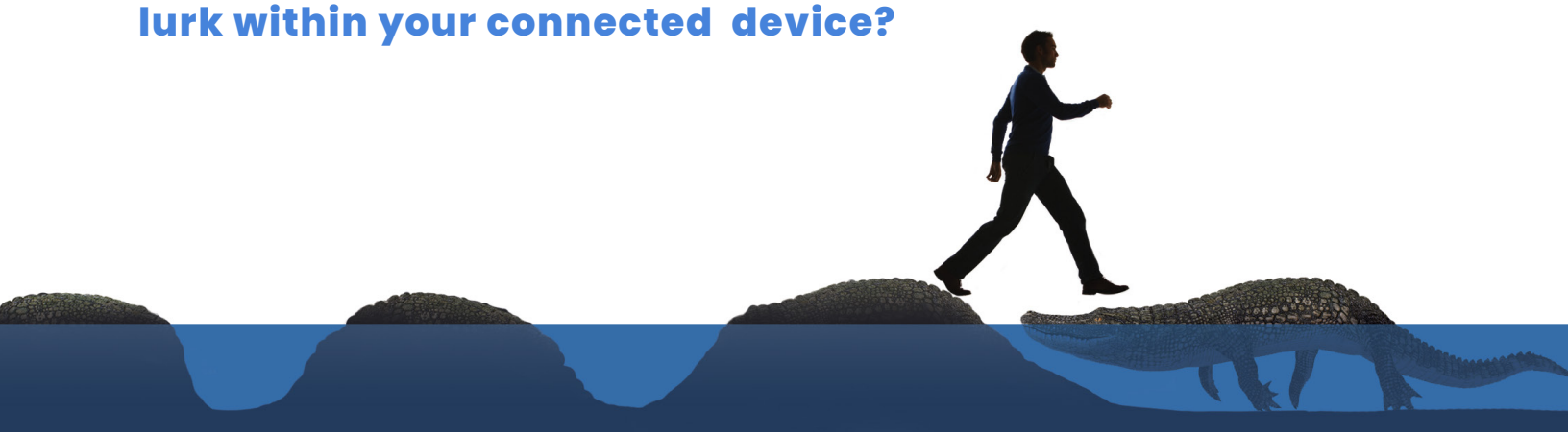
- **Binary Software Composition Analysis (Binary SCA),** a technique that helps organizations see inside the unknown binary files lurking within their connected devices and embedded firmware.
- **Device Composition Analysis (DCA)**, a holistic, all-encompassing product security process that identifies all components within a device, including both hardware and software components.

Product security represents a critical process when products are created... not a bolt-on verification added at the end before shipping containers are sealed shut.

Coupling together SCA and DCA creates a comprehensive approach to the continuous assessment of device vulnerabilities, and also provides the business with peace of mind that risks introduced by third-party software and components are minimized.

Binary SCA and DCA create more comprehensive SBOMs and reconcile embedded components within your product against known risks and scan for open source license exposures. When security teams put these tools in place, they get actionable information that can be shared within a manufacturer's product team or with a supplier who can then remediate the vulnerability.

What's the risk? How do you know what vulnerabilities lurk within your connected device?



The best defense against product security risk is a strong product security strategy that spans your connected products, whether you make them or buy them. Vulnerabilities can come from many sources within your software supply chain or even within your own product development processes. These vulnerabilities often remain unseen, hidden within the binaries skipped over by the static and dynamic software testing we have long relied on to keep our products—and the processes and people who depend on them—safe.

Whether the product security risk comes from your own development process or has slid toward you along your software supply chain, the risk of vulnerabilities in the code of your embedded systems and connected devices represents a universe of unseen risk overlooked by many risk management programs—even those well-versed in cybersecurity risk. And when vulnerabilities are not continuously surfaced, they can become threats that expose the assets of an organization to cyberattacks.

Until recently, many cybersecurity experts downplayed, or even ignored, the risks associated with the firmware that powers all connected devices and embedded systems—even if those risks have raised concerns for years. Today, as cyber attackers seek new ways to find and exploit vulnerabilities, firmware offers a greenfield of opportunities, as threat actors like Russian ransomware group Conti have shown this year.

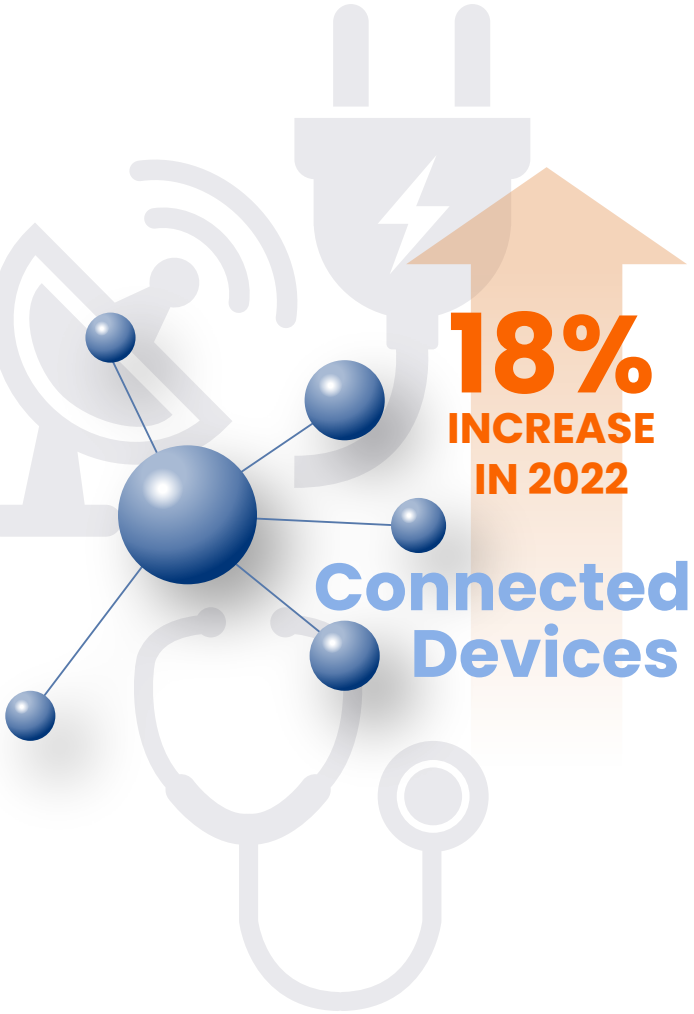
In the past, the easiest fix was to do nothing at all—and hope for the best. After all, product security takes time, resources, and people to do right, and, like all controls-based initiatives, it's a cost center that hits bottom lines. On the asset owner side, when you buy an asset and the product security process isn't yours, you may be facing an uphill battle to convince a manufacturer that a vulnerability even exists—and that they need to work with you to fix it.

But, with cyberattacks on the rise and bad actors emerging all over the world, we can no longer be complacent when it comes to product security across your software supply chain. In our internet-connected global ecosystem, we must be continuously vigilant against threats from our adversaries.

Who's most at risk?

Connected device use soared during the COVID-19 pandemic. The average US household now uses some 25 connected devices, including smartphones, wireless headphones, and smart home devices. That's more than double the average number of connected home devices in 2019, according to a report from Deloitte.

That same proliferation has come to Energy, Telecom, Healthcare and other critical infrastructure sectors of our society. Even with the pandemic-era chip shortage that's slowed the recovery of the IoT market, the number of connected devices is still slated to grow 18% in 2022 to reach 14.4 billion by the end of the year.



However, while the most common risks facing household IoT devices may materialize into theft of privacy, personal data, or sense of safety and well-being, in America's most critical sectors, the risks can grow far larger.

> **Consider:**

- **Embedded devices** control critical parts of national defense strategies. When embedded devices control missiles and aircraft, the implications of an attack on unseen or misunderstood vulnerabilities could rise to the level of catastrophic.
- **Medical devices** that regulate and control critical systems of the human body may be prone to cyberattacks as these devices become increasingly connected. The consequences of these attacks could be fatal.
- **US critical infrastructure—**such as utilities, waste treatment plants, and transportation systems increasingly rely on connected devices whose failure could lead to widespread and far-reaching destruction and devastation.

The risks supporting a strong product security and supply chain strategy may be clear and critical, but when it comes to securing your connected device and embedded systems, where do you start? What are the steps you can take to arrive at connected device security?

The steps to connected device security

1 Discover

You cannot assess, improve, respond to, and remediate threats and vulnerabilities if you do not know where they wait within your code, whether you've built the product or bought it. **When you set out to achieve connected device security, your most formidable challenge may be that you don't actually know what you have in your products.**

So, the first step becomes discovering what you have, which isn't as simple as it may first appear. When you bring your software to the discovery stage, it's not unlike approaching an automobile at a car dealership. On the surface, you can form assumptions about what you're observing, and what may lie inside. You may form these assumptions based on what you know about the manufacturer, the dealership, and the appearance and condition of the automobile.

How do you see into what you cannot see?

For automobiles, we can view reports that list a vehicle's accident and damage history, and all known vulnerabilities that have been significant enough to merit a recall. You need a similar process to protect your software from vulnerabilities and threats. That's why the discovery step becomes so important.

Similar to how a car history report can reveal potential hidden problems, you need a process to protect your software from vulnerabilities and threats.

To understand the threat environment that lurks within software, you must examine all the code that runs within it, and within all its components. That code could be your own, or it could have come from an upstream provider in your supply chain.

To get to connected device security, you need to inventory all the code you're using whether you wrote it or not.

Many reach for source code when device security discussions turn to discovery. After all, source code forms the basic building blocks of your product. If you are the manufacturer—and have developed complete vertical integration—you can analyze your source code because it's fully available and visible to your product security efforts.

But, **does anyone ever produce anything in a vacuum**—with no help from anyone else on a software supply chain or with no reliance on open-source code? In modern development, nearly every piece of software depends on components developed by supply chain partners or even unknown open-source developers, which may, in turn, depend on other layers of components developed by their partners.

Like a manufacturer in the middle of a software supply chain, asset owners don't have the source code for devices and systems they acquire from supply chain partners. Therefore, analyzing source code won't even be an option for asset owners. Unless you're a small shop that develops just one library and know that you have no dependencies whatsoever, you very likely cannot say that you have all the source code for everything at all times.

You find yourself in the situation of a car buyer, looking under the hood of a new car—perhaps with a familiar, well-trusted brand—and seeing components manufactured by many other companies, considerably lesser known.

But, even if you have ***all*** available source code, assessing connected device security comes with pitfalls.

> **Consider:**

- **A product, device, or component made by someone else comes with source code already compiled into binaries that are often impenetrable to many SCA methods.**
- **Differences often exist between your official source report and what you are shipping, due to differences between what development planned to use during the build process and what eventually got pulled into the compiled binary.**

How do you break into the binaries that have become part of your connected device or embedded system? You can't just take them at face value. Find a solution that features binary analysis, which can reverse engineer binaries to give you continuous visibility into software risk, whether you wrote it, or it came from an upstream supply chain partner.

Additionally, a product is more than its binaries. Connected products may also include exposures such as credentials, certificates, configuration files, file permissions, SELinux and other exceptions, unsafe functions that survive code compilation, and exploit mitigations, to name a few.

During the discovery phase, to truly identify what you have, and what you need to fix, you need a tool that can see beyond source code–and into a product's binaries, credentials, certificates, and configuration files. Even if you manufactured the product with your resources, you need

to see into the product you've stacked in your shipping area and not rely purely on an analysis of the source code you planned to use.

Without the discovery phase, you cannot defend your product, or the services it will provide, against vulnerabilities and attacks. You need a report that shows you the complete picture of what is in your product so that you can determine what it means to your security posture.

2 Assess

Even after you know what you have, turning the corner to understand what it means can represent a big step on the road to device security. When you begin the assessment stage, you face two types of third-party code:

Even with proprietary code, known CVEs (Common Vulnerabilities and Exposures) may exist if researchers have found vulnerabilities and exposures within its lines and published their findings.

To begin to assess which vulnerabilities and weaknesses apply to your connected device, you need to reconcile every package, whether open-source or proprietary, against the different feeds of CVEs that list publicly disclosed cybersecurity vulnerabilities. In that process of matching vulnerability data to the components you have identified during the discovery stage, you generate a list of vulnerabilities for your connected device or embedded system.

In this assessment, cybersecurity practitioners must also remember to look for CWEs that, if exploited, could become new vectors for cyber attackers looking to exploit an attack surface.

When you have a solution that includes binary analysis, you get full visibility into all of the code in your product–and when you check that code against known vulnerabilities and weaknesses, you get a more complete list.

By the end of the CVE process, you get a list of components and code pieces that comprise your device and the CVEs and CWEs associated with each of them. You also have a list of concerns that may not be code-related, but still merit attention. These can include:

- Hard-coded passwords that can offer backdoor access to your device
- Expired or improperly signed certificates

A complete assessment gets you not only the security weaknesses for the code that you execute, but also the system-level weaknesses that enable attackers to reach that code.

3 Prioritize

What happens when you get a list with thousands of vulnerabilities and weaknesses... and a note that you have a handful of hard-coded passwords? After you get your list of everything that can be exploited within your connected device, what do you do? How do you approach improving your device's security posture?

The sheer number of vulnerabilities, weaknesses, and other concerns makes it impractical—and prohibitive from a time and resources standpoint–to individually research and prioritize each one. How can cybersecurity professionals analyze these initial results from a scan that includes your code and all its binaries?

First, product security and risk management personnel should create a baseline, a starting line, so they can measure the improvements they will realize from future remediation efforts. Look for a device security solution that provides a score that reflects the overall risk of your connected device. Save that score for future reference.

Equipped with a cumulative risk score, the prioritization effort next shifts to weighing effort versus impact. Which remediations will deliver the largest improvement to the cumulative risk score of your device? Which vulnerabilities are the most pressing to mitigate?

Product security and risk management experts can resolve vulnerabilities using one or a combination of the following two strategies:

The Grouping Method

Choose to upgrade firmware and software for the components with the highest number of vulnerabilities

What if you could resolve large numbers of your vulnerabilities by focusing on just a few of your components? If you have hundreds of components that have fifty vulnerabilities together and five components that have hundreds, where would you start?

By focusing on the components that present the most vulnerabilities, the largest numbers of vulnerabilities can be addressed.

> **Consider:**

- **If you knew that upgrading or patching three components would knock down 50% of your CVEs, wouldn't you do it?**

When looking for a connected device security solution, look for a vendor that works with you to find the components whose remediations present the biggest improvement for the effort they require to remediate.

The Seek and Secure Method

Prioritize vulnerabilities and weaknesses with the greatest likelihood of being exploited, or that present the most severe threats

The Grouping Method mitigates the most vulnerabilities with the least effort, but how do you know that the most concerning vulnerabilities have been prioritized? When reviewing the vulnerabilities and weaknesses that remain, consider the risk that each presents, from the perspectives of risk severity and risk likelihood. Consider too the CV database score and focus on CVEs with scores over 8.

> **Consider:**

- **Is the vulnerability exploitable?**
- **Would the impact of an exploit of the vulnerability be material?**
- **Is the component connected to a network?**
- **Was the code compiled with a mitigation enabled?**

By prioritizing the most troubling vulnerabilities and the largest numbers of vulnerabilities with the smallest number of actions, you can arrive at a plan to improve the security posture of your connected device or embedded system and move on to remediation.

4 Remediate

The path to remediate vulnerabilities takes two different roads. Your path depends on whether you own the code in your connected device or you've purchased it from someone in your software supply chain.

Product Manufacturers

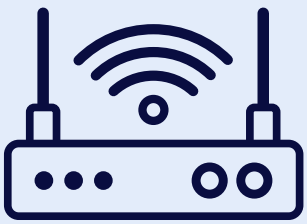
Within product manufacturers, product security teams can begin to fix the code with vulnerabilities and weaknesses upon discovery and assessment. While prioritization makes this process more efficient, when you own the code, fixing vulnerabilities and weaknesses within it depends upon your resources, time, and budget. To show progress, a manufacturer can run a post-fix scan and compare it to the scan completed before they did the work.

Asset Owners

Asset owners also rely on product manufacturers to fix code. But, what happens if they say 'no' or 'maybe later'? Convincing a product manufacturer to invest time and resources into improving a product that you have already purchased might become an uphill struggle.

Without advanced tools to generate SBOMs to understand all the subcomponents, we often cannot see product security. It's not a differentiator. Even with B2C devices like home routers, does anyone truly buy a router based on its security? Perhaps. But, more visible metrics like connection speed and potentially router design drive many more sales than a low number of vulnerabilities and weaknesses within a router's firmware.

Convincing a manufacturer to fix code in a product you've already purchased can be an uphill struggle.



When product security lies outside your direct control, how do you, as an asset owner, affect change? After you've progressed through discovery, and assessed and prioritized the findings, what power do you have to influence a manufacturer's willingness to improve the security of weakened or vulnerable products?

Of course, asset owners can wait for a new software or firmware update and hope that the vulnerability has been addressed. But, when the stakes are high—as is often the case in infrastructure, healthcare, or automotive technology—waiting is not a viable option, even when you have SBOMs to verify that these concerns have been mitigated.

But asset owners have the power of the purchase order. With the transparency and continuous visibility of vulnerability assessments and SBOMs, asset owners have the information they need to pressure vendors to make changes and improve their product security postures. Asset owners can go to their vendors and say that they will withhold future purchases if changes are not made.

Customer-vendor relationships do not have to go sour over product security concerns. Some product security and supply chain security vendors will work as intermediaries between asset owners and manufacturers to help address the vulnerabilities and weaknesses that their solutions have surfaced. The best solution is often collaboration, as it furthers the common good and a more secure environment less prone to cyberattacks.

5 Respond

According to NIST data from the National Vulnerability Database, over 20,000 vulnerabilities were discovered in 2021—more than fifty every day. The number of vulnerabilities discovered has been growing each year since 2016, when 6,447 were discovered.

How do you protect against product risk and software supply chain risk with a new vulnerability being discovered, on average, every 30 minutes? SBOMs and mitigation action plans often become point-in-time documents once the results are analyzed and assigned for remediation. However, the threatscape continues to evolve. Snapshot remediation just does not work.

New vulnerabilities get discovered every 30 minutes on average, putting product and software supply chains at constant risk.

When Log4j first surfaced in late 2021, product manufacturers and asset owners alike scrambled to determine if they were exposed, potentially over hundreds or even thousands of products. With a tool like Global Search, you can search all versions of firmware, software, components, and products in your asset inventory and determine if you have an exposure to newly discovered vulnerabilities and weaknesses. You can get an answer fast.

When you know whether you are exposed to new vulnerabilities and weaknesses, you can determine if you need to take devices off your network while you coordinate with your engineers (or the product manufacturer's engineers) to devise a mitigation plan.

Regardless, a Global Search tool identifies exposures in your connected products and embedded systems and allows you to take your device off the grid and away from harm until you have a resolution or a workaround.

6 Improve

In cybersecurity, a strong offense may well be a solid, strategic defense. But defenses don't always provide clear paths to the ROI numbers that Leadership needs to justify investments into product security and software supply chain security—no matter how big the risk may a ppear to cybersecurity practitioners.

It's hard to quantify a solution's value-add when it strengthens a cybersecurity control environment and helps prevent attacks from ever happening. But, it's still possible to show a return for the investment of time, money, and people that standing up a solution protecting product security across the software supply chain requires.

Products are never static. Their security posture changes over time, as patches and remediations are released. With a scan of your product's software and firmware, you can track and identify the vulnerabilities and weaknesses in your product, assess their impact, and show the trendline of a product's security improvements over time.

With a solution that evaluates your product security posture over time, you get snapshots of your progress and can see the results of efforts to improve your product security and reduce software supply chain risk.

Conclusion

When first setting out to improve product security across the software supply chain, there are many solutions to consider and a universe of possible first (and second) steps to plan. Finite State recommends starting with an assessment of your organization's cybersecurity environment and its needs so that you can then compare that to the budget that your leadership can allocate to improving your product security posture.

Effective project management begins long before commitments to cybersecurity solutions that solve the problems of your organization. Successful product and supply chain security initiatives follow a comprehensive, disciplined approach with informed project plans that consider the due-diligence assessments and reviews completed by SMEs knowledgeable of your organization and industry.

Clear, actionable policies and organizational governance establish the guidelines that direct the planning and execution of your cybersecurity projects and initiatives and the formation and approval of project plans. Effective connected device security programs clearly establish expectations and requirements, reflect your business and its priorities, and set the goals, milestones, and schedules of your initiatives.

The right project champions–knowledgeable about your organization and its processes and priorities—can shepherd your project through its phases and report progress to executive management and other stakeholders. Transparent, truthful communication—from project start to finish—will result in a more efficient execution of the project and more effective outcomes. In the end, even after your project completes its final step and closes, progress must continue to be monitored and actioned upon, in order to ensure that your cybersecurity program continues to meet its objectives.

See automated product security in action

Learn more about automating your product security across the software supply chain. Connect with a Finite State expert today.

Learn more

Discover why SBOM is critical to connected device security.

Finite State's comprehensive SBOM guide

Hear from thought leaders in product and supply chain security

IoT: The Internet of Threats Podcast

Finite State for Manufacturers

Finite State Platform OEM Datasheet
Product Security Primer
Why Invest in Product Security?

Finite State for Asset Owners

Finite State Platform AO Datasheet
Asset Owner Solution Overview

Appendix

Even after you have committed to improving the security posture of your connected devices, you still need to assess where your organization sits today. When it comes to securing your connected devices and embedded systems, what progress have you already made and what still lies ahead?

Use this guide below to determine where your organization currently stands regarding improving product security across your software supply chain.

	Level 0	Level 1	Level 2	Level 3
	NO PROGRAM	**REACTIONARY PROGRAM**	**FUNCTIONING PROGRAM**	**OPTIMIZED PROGRAM**
CURRENT STATE	Ignores risks lurking in binaries, open-source code credentials, config files, etc. No security relationship with internal/ external software supply chain partners	Over-relies on SCA as a standalone solution Troubleshoots connected-device exposures **after** they become problems Minimizes connected device risks Communicates ineffectively or not at all with internal and external supply chain partners to resolve exposures	Integrates tools such as Device Composition Analysis and Binary SCA Regularly scans products for connected device risks Communicates effectively with internal and external supply chain partners to resolve exposures	Drives product and software supply chain strategy Nurtures development of Internal and external supply chain partnerships Fosters culture of continuous improvement in connected device security and beyond Creates a culture where connected device security inspires product design and controls
NEXT STEPS	Determine organizational needs Grow operational knowledge of Device Compositional Analysis and Binary SCA Establish requirements for solution Secure financial and operational budgets to establish a connected device security program	Ask the right questions in RFIs/RFPs and demos Create communication channels to keep leadership informed Create governance models to monitor project progress and costs	Empower product security and risk management teams to share findings with upstream supply chain partners Regularly meet with internal/ external supply chain partners to drive control environment improvements	Implement continuous monitoring of connected device security improvements Engrain importance of connected device security into organizational culture

About Finite State

Finite State empowers organizations to gain control of product security for their connected devices and supply chains. Backed by a team of seasoned experts, our automated, easy-to-use platform arms our customers with the actionable insights, critical vulnerability data, and remediation guidance necessary to mitigate product risk and protect the connected attack surface.

Finite State delivers the deep visibility required for organizations to find and address security issues at every stage of the development lifecycle. We handle connected devices and embedded systems across all industries, including those found in healthcare, utilities, connected vehicles, manufacturing facilities, critical infrastructure, and government entities.

FINITE STATE

finitestate.io