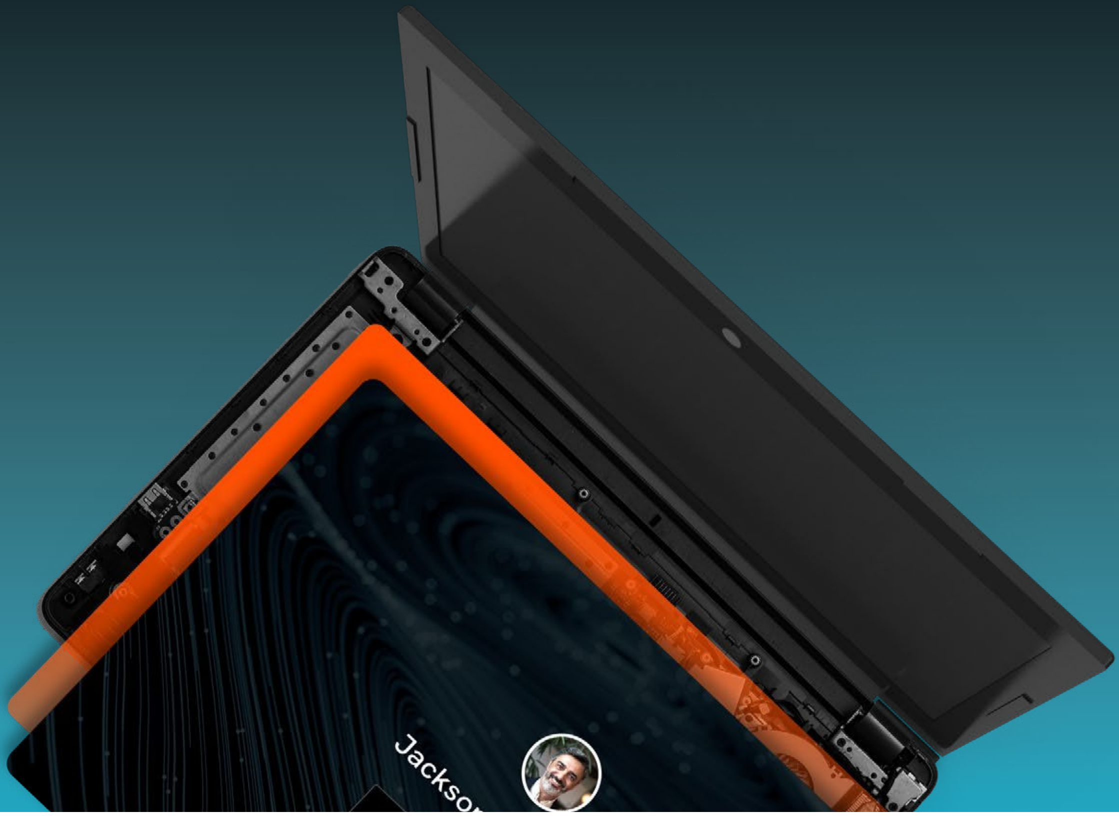
eclypsium

THE ULTIMATE GUIDE TO SUPPLY CHAIN SECURITY_



A NEW FRONT IN CYBERSECURITY_

The industry is facing arguably its most fundamental challenge in IT and security today—how to easily and independently audit the security of their supply chains and verify the integrity of the products and services that they rely on. While this may sound straightforward on its face, this is a challenge unlike anything faced before, and one that requires a very unique set of solutions.

This is because supply chain risks and threats fundamentally undermine many of the core assumptions about how the cybersecurity game is played. They change when and where the fight happens. Instead of attacking an enterprise directly in real-time, adversaries can avoid enterprise defenses altogether by rewinding the clock and instead targeting any of the dozens of suppliers, subsuppliers, or developers that built the product or service in the first place.

.

But most importantly, supply chain threats take advantage of the complex webs of trust that underlie all the technology we use today. Every piece of equipment, every application, and every cloud service is a collection of highly specialized, interdependent pieces made by different entities. Applications and services consist of dozens of components, APIs, and open-source projects, each often with hundreds of contributors. Those apps and services run on servers, switches, and laptops that are built from hundreds of physical components such as SSDs, network adapters, GPUs, I/O devices, RAM, PCIe controllers, etc. These components run on their own internal firmware, and likewise depend on the core system firmware and chipset, the system boot process, and so on. Every time an organization buys or uses a “thing” they are actually dealing with dozens of things.

And every component and layer is trusted to do its job or everything falls apart. But more importantly from a security perspective, every layer is trusted. And when any given strand of that web is compromised, it can be incredibly hard to notice it. And this is where supply chain security becomes very different from all other forms of security in the market. Yes, we still care about vulnerabilities and threats. But we also have to do the incredibly hard work of unraveling the web to verify that every component is what it should be, is working the way it should, and that all the pieces are working together. In addition to knowing how adversaries attack, we have to know how very complex systems actually work. While other security tools can focus on a specific area such as analyzing malicious binaries, we care about the whole product—that all the components are authentic, free from vulnerabilities and threats, and that all the components and protections are working together as intended.

This ability to unwind and audit complex systems is an enormous job. It is a job that existing tools don’t do. But it is a job that must be done. And just as importantly, it is a job that must be done simply. Every business has a business to run, and every agency has its mission. Deconstructing the minutia of every piece of technology they use is not their core business. They need to be able to actively test their technology, know if there is a problem, and if so, how to fix it.

This guide will lay out the key considerations when implementing a supply chain security program, with a focus on securing the infrastructure supply chain. We will focus on the operational environment and infrastructure that supports the organization, not on the application development environment. By the end, you’ll be equipped with a solid understanding of the criteria that your supply chain security program should meet with recommendations on how to achieve those goals.

WHAT’S AT STAKE_

Supply chain risks pose a critical problem today for virtually any organization that either produces or consumes technology (i.e. everyone). To understand why, we have to understand the nature of supply chains themselves, why they are being attacked now, and how this is playing out in the real world.

The World Runs On Supply Chains

Supply chains are at the heart of what makes modern technology and even the global economy work. They make things far easier, more efficient, and cost-effective. Instead of building everything from scratch, highly complex systems are broken into component parts or tasks and handled by specialists. An enterprise doesn’t need to know how the sausage is made for their laptops; they just need to use them to do their work. A laptop vendor doesn’t need to personally engineer and build every component from scratch (e.g. SSD, network controller, etc). Developers don’t have to recode the same functionality for every job but can pull from an open-source project or just use an API. Every layer is designed so that someone or something can consume the value of a technology without having to worry about the details upstream. This is 100% by design and a feature, not a bug.

However, the other side of the coin is that supply chains are also chains of trust. Each time technology is consumed in the supply chain it introduces the potential for problems both intentional and unintentional such as:

Introduction of Vulnerabilities - Newly written or modified code could introduce vulnerabilities in the component. Depending on the component, the vuln could be easily missed and buried at a level not seen by traditional scans. Developers could inadvertently pull from an outdated, vulnerable branch of an open-source project, or call vulnerable dependencies.

Examples: Vulnerabilities in hardware components are everywhere. The recent BMC&C vulnerabilities in a leading manufacturer affected dozens of server manufacturers and the cloud infrastructures they support. Such problems can persist and resurface years after their discovery as seen in the recent resurfacing of the PantsDown vulnerability. In the software world, the recent Log4j vulnerability had industry-wide impacts and was widely attacked in the wild. Vulnerabilities in OpenSSL have repeatedly caused problems and as the example of HeartBleed illustrated, these vulnerabilities can persist undetected for years.

Introduction of Malicious Code - Attackers can introduce malicious code in a variety of ways. Many components are built in countries where nation-state adversaries can easily implant within the firmware of a component or within an application. Likewise, attackers can compromise a vendor and unknowingly add malicious code into otherwise “valid” products or software updates. The accessible nature of open-source code means attackers can add malicious code into projects that are passed downstream.

Examples: The SolarWinds attack known as Sunburst allowed attackers to implant malicious code within valid source code that was delivered to thousands of organizations. Ransomware operators implanted malicious code known as iLOBleed within server BMCs in order to take over entire data centers. More than 200 malicious cryptomining packages were recently found in npm and PyPI open-source libraries.

Introduction of Counterfeit Products or Components - The supply chain can also introduce fake or unauthorized products or components. Vendors and resellers have repeatedly been caught selling fake Cisco devices. Likewise, when facing supply chain shortages, suppliers or manufacturers may substitute lower-quality components in order to meet the demand in the market. These sub-standard components can contain vulnerabilities, malicious code, or may lack capabilities and protections that are essential for the safety of the final product.

In all of these cases, supply chain risks are especially pernicious when it comes to infrastructure, such as endpoints, servers, network devices, and cloud assets.

Stealth and Privilege - Vulnerabilities or implants in the supply chain arrive in the guise of code that is presumed to be valid and safe. Such problems can be hard to detect since deconstructing every component and line of code would defeat the value of outsourcing the job in the first place. Problems are often buried in low-level code that isn’t visible to traditional vulnerability scans.

This code sits lower and runs before the host operating system, allowing its privileges to supersede that of the OS or root users. This means that malicious code in the supply chain can easily lie to or subvert the OS or applications that ride on top of it, and evade many of the security controls that organizations rely on, such as EDR.

Persistence - Attackers can also use the supply chain to persist within a product or environment. If an organization suspects a problem, it may update or reinstall software. However, if the problem is in the supply chain, they will likely just reinfect themselves. Likewise, implants in low-level code and firmware can hide within components that are not updated even after a complete reinstallation of the OS. This happened in the iLOBleed attack referenced above.

Theft - Naturally, if attackers have access to a system, they will have many opportunities to steal or alter sensitive data. And with direct access to low-level systems such as drives, network controls, or out-of-band management components, attackers can steal data in ways that are not likely to be detected by common security controls.

Disruption - Attackers can also use the supply chain in order to disable or permanently break an asset. With access to the code that runs physical components, attackers can "brick" devices in ways that can be difficult or impossible to recover from. Ransomware operators in particular can use this technique to disrupt organizations and extort higher payments from their victims.

Reputation - All cyberattacks carry the potential to cause extensive reputational damage. However, this risk is particularly high when it comes to manufacturers and vendors. If a vendor or supplier is found to have unknowingly delivered malware to their customers, it can cause irreparable damage and loss of trust.

EVERYTHING HAS A SUPPLY CHAIN_

Lastly, we have to recognize that this problem applies to every level and type of technology.

Software -All software has supply chain risk. Even third-party compiled software made by a single vendor has risk as seen in the case of the SolarWinds attack. It likewise affects open-source software packages that are continuously reused by countless organizations and developers. Organizations will need control to ensure the integrity of the software they buy as well as tools to ensure the safety and integrity of open-source resources. This can include the ability to assess open source code and projects directly as well as ensuring the integrity of an organization's in-house CI/CD pipeline.

Firmware - Firmware deserves additional focus because it is some of the most privileged and powerful code, and also often the code that is the least audited. This includes system firmware such as BIOS, UEFI, Intel ME, and Mac EFI. It also includes firmware embedded within hardware components such as CPU microcode, SSD firmware, and NIC firmware. Network devices heavily rely on firmware within their operating systems (e.g. Cisco IOS, F5 TMOS, Juniper JuneOS, Fortinet FortiOS, Huawei, etc). IoT devices often exclusively run on firmware such as Linux-based embedded firmware, router OpenWRT, RouterOS, and others.

Cloud - Cloud infrastructure and services are crucial to enterprises. However, moving to the cloud doesn't mean that hardware or supply chain risk magically disappears. The hardware within a third-party private cloud will have all the same supply chain risks organizations would have if they bought the hardware themselves (see the CloudBorne vulnerability as an example). Likewise, public cloud infrastructure such as AWS, Microsoft Azure, and Google Cloud can have their own vulnerabilities and misconfigurations. The same is true for cloud infrastructure software as well as third-party SaaS software and APIs hosted in the cloud.

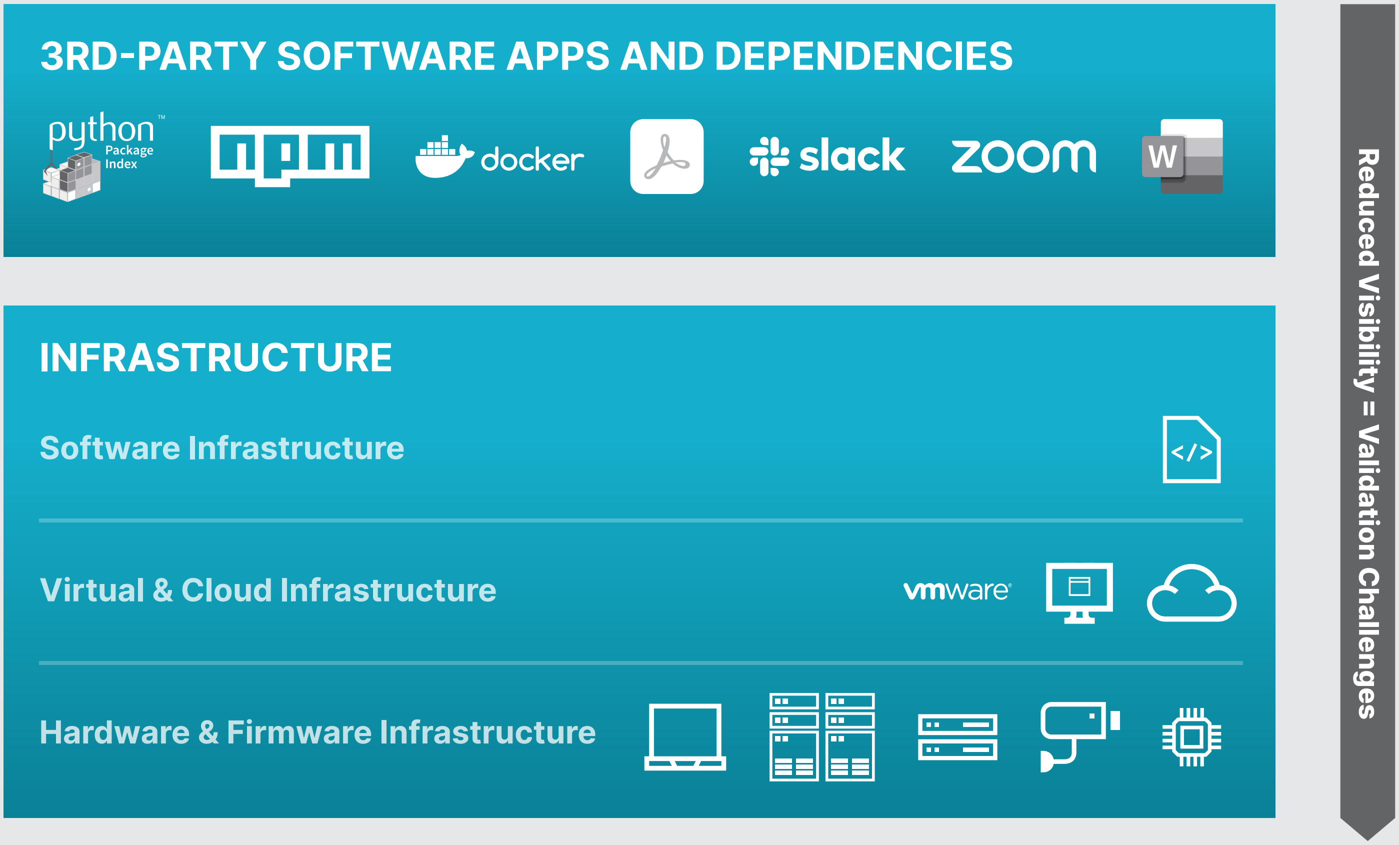
Hardware - Virtually every piece of equipment is a complex amalgamation of physical components and code sourced from different providers, which often have their own upstream supply chains. This includes standard IT devices such as laptops and desktops. It includes networking equipment such as switches, routers, VPNs, firewalls, and telco equipment. It includes servers, data centers, and cloud hardware. It includes IoT devices, medical devices, and OT and industrial control systems. If it has a power button, it most likely has the potential for supply chain risk.

Collectively, this is a massive problem for any organization that uses technology. It's a problem that is quite literally everywhere, carries massive impact, and is untenably complicated. Unraveling the inner workings and history of every piece of technology they use is not their job. But the risks that those supply chains pose to their business very much is their problem. And this is precisely where a supply chain security program and tooling comes into play.

KEYS TO AN INFRASTRUCTURE SUPPLY CHAIN SECURITY PROGRAM_

Organizations need security controls and processes that mitigate the unique risks of their infrastructure supply chains. A supply chain security strategy should cover all types of critical assets (devices, software, and cloud). Supply chain security should also extend across the lifecycle of a technology such that an organization can validate the security and integrity of their products and services from initial evaluation through end-of-life. Ideally, these same processes should extend into the supply chain itself where vendors and suppliers evaluate the components they acquire, validate their own products, and build auditable documentation that can be passed on to end customers.

INFRASTRUCTURE SUPPLY CHAIN ATTACK SURFACE



3RD-PARTY SOFTWARE APPS AND DEPENDENCIES

INFRASTRUCTURE

Software Infrastructure

Virtual & Cloud Infrastructure

Hardware & Firmware Infrastructure

Reduced Visibility = Validation Challenges

Core Capabilities for Supply Chain Security

This requires a range of new capabilities aligned to the unique risks of the supply chain. Organizations will need visibility into the deepest levels of an asset, insights into hidden risks, and an understanding of how all the elements of a supply chain work together. However, supply chain security tools can easily automate these functions. Core capabilities include:

- **Expertise in supply chain-specific threats and vulnerabilities** - There are many tools in the market that look for threats and vulnerabilities. However, the supply chain introduces threats and vulnerabilities that are often buried deep within products (or early in the development process) where they are beyond the view of traditional scanners. Reliably detecting these risks often requires the development of specialized drivers, research experience, and detection mechanisms that are independent of the operating system.

- **Integrity validation** - Most security tools look for what is “bad”. A supply chain solution must do this as well, but even more importantly, it must be able to verify what is “good” and this requires a very different set of skills. It requires us to build a massive and constantly changing database of all the critical code and components that products rely on, down to the level of firmware packages, open-source branches, and dependencies. The only way you can tell if something has been altered is if you have an incredibly deep and precise view of how things are supposed to be. This ability to deeply catalog the known good across many different types of assets (laptops, servers, apps, cloud), many different vendors, many different suppliers, many different software stacks, and open-source projects is something that no one else in the industry does.

- **Product-level validation** - Products aren’t just a collection of parts - everything has to fit together and dozens of low-level systems and settings all have to work together. For example, a modern “Secured-core” PC requires OEMs to bring together a variety of protections from different vendors and configure them in a very specific way. If a single bit gets flipped or the right setting isn’t enabled, then the collective protections can fall apart. There is a very long history of this happening. Again, traditional tools have limited scope in that they focus on specific exploits, malicious binaries, or vulnerabilities. A supply chain security tool should cover this all-important step of validating the product or service as a whole.

Key Steps of a Supply Chain Security Program

Organizations will need to apply these security capabilities as part of a larger security program. Supply chain risks exist across the full lifecycle of an asset. The risks begin before the asset is ever acquired, and through updates, new vulnerabilities, or threats, will extend until the asset is decommissioned. This introduces yet another unique aspect of supply chain security. Responsibility must often be shared across multiple functional teams. Many or most of these teams will not be security specialists, yet still need to be able to assess the products that they are working with. This makes it all the more important that supply chain tools are as simple and automated as possible.

Ideally, supply chain security should include supply chain vendors as well as their enterprise customers. However, an enterprise may not have much visibility or control over how their vendors operate. So we will look at these two cases (enterprises and vendors) separately, starting with the enterprise.

Key Steps For Enterprises

There are several cases where an organization will likely want to assess the integrity and risk of its technology. As with all security it is best to address issues as early as possible in order to reduce exposure and mitigate any unnecessary risks. However, security is also never static. New updates, changes, threats, and vulnerabilities will always alter the risk profile of an asset, and organizations will need to have processes in place to address these risks.

- **Initial Product Evaluation** - IT and procurement teams will need deep insight into prospective solutions to understand not just the cost, but what is actually inside, and if the product has any vulnerabilities or components with suspicious origins. Organizations should naturally prefer vendors that have secure products and components including those sourced from other suppliers, and detailed and up-to-date Software Bill of Materials (SBOM) for their products.

- **Pre-Deployment Validation** - IT and/or Security teams will need to validate the products that they receive before they are put into operation. Supply chain security tools can generate an SBOM of an actual asset that can be compared to a vendor-supplier SBOM to ensure that all artifacts match. Scans should likewise verify that critical software, firmware, and hardware components are authentic, unaltered, free of vulnerabilities, and have not been tampered with in the supply chain.

- **Continuous Monitoring After Deployment** - Security teams will need to perform ongoing automated scanning to identify any changes in integrity, identify new vulnerabilities, or detect signs of compromise. Security or IT staff should also monitor new software and firmware updates including behavioral analysis to identify new threats that may be hidden within signed vendor code.

- **Vulnerability Management and Updates** - IT and vulnerability management teams will need the ability to discover devices and scan for vulnerabilities that are often missed by traditional scans. This can include finding and scanning networking infrastructure, server BMCs, and digging down into the firmware that is hidden beneath the OS. Updating these devices can require strategic planning, so it will also be important for teams to know the real-world risks of a vulnerability and how to apply needed updates safely.

- **Incident Response and Threat Hunting** - IR teams will need to be able to verify the integrity of assets that are known to have been involved in a security incident and to validate systems are safe before being returned to service. Threat hunters will likewise need to search for supply chain threats. These threats can be deeply hidden within an asset and able to evade traditional threat detection tools. In other cases, malicious code could be delivered within 'valid' vendor code. In these cases, staff will need deep visibility not only into supply chain components but their behavior as well.

Key Steps for Technology Vendors

Technology vendors have an even greater responsibility when it comes to supply chain security. Like their customers, technology vendors are also enterprises and should be addressing all the same requirements covered in the previous section. However, they are also intrinsically a part of the supply chain and the integrity and security of the products they deliver is a central aspect of their core business. This makes it all the more important that technology vendors have the tools and processes to validate the security and integrity of their upstream supply chains and to build the detailed documentation they need to provide their customers. Since vendors will also need to address the previous steps, we will continue the numbering to cover the following additional vendor-specific steps.

1. *Analysis and Tracking of Upstream Components* - Vendors will need to validate and regularly assess the software, firmware, and hardware they consume from suppliers and sub-suppliers. This can include checking the integrity of components, detecting vulnerabilities, checking patch levels, and building SBOMs for third-party components.

2. *Collecting and Distributing Product Evidence for Customers* - Vendors also play a critical role in ensuring the supply chain is not opaque and that customers can easily verify the security of the products they receive.

Creating and maintaining detailed SBOMs and documentation of product attributes and artifacts will ensure customers have a definitive reference point when validating their products.

3. *Infrastructure and Operational Analysis* - In addition to analyzing their products, vendors should regularly evaluate the methods, processes, and infrastructure used to deliver and deploy their products and updates. Weaknesses in update mechanisms can allow attackers to intercept and compromise products and updates as they are delivered to the customer.

Integrating Supply Chain Security Into the Overall Security Strategy

Supply chain security cuts across virtually every part of an organization. It can involve virtually any type of asset and can impact a wide range of job roles including technical staff as well as business-line managers and executive leaders. This makes it particularly important that the insights and contexts from a supply chain security platform don't exist in a silo, but are integrated and shared with other enterprise systems. Organizations should ensure that their supply chain security tools share context with and ingest context from other security tools. Some examples can include but are not limited to:

- *Integration With Developer Tools* - Many organizations have invested in a variety of tools to better secure their internal development processes such as open source code analysis and CI/CD security tools. These tools can provide important insights into the open-source supply chain, and supply chain security tools should ingest and integrate this data to provide a complete view of supply chain security across all of an organization's assets.

- *Zero-Trust Initiatives* - Knowing the posture and integrity of an asset is critical when it comes to making Zero-Trust access decisions. However, many organizations are unable to go beyond basic software checks to actually verify the integrity of the asset itself. A supply chain security program will have critical insight into the posture and risk of an organization's many classes of assets whether laptops, servers, or cloud, allowing Zero Trust strategies to extend to the most fundamental code and even to extend upstream into the supply chain itself.

- *Asset and Vulnerability Management Tools* - Most organizations will have tools and processes for managing their assets and for finding and patching important vulnerabilities. However, these systems often lack visibility into critical device-level code and vulnerabilities or misconfigurations in the supply chain. Supply chain tools can share this information in both directions so that IT and vulnerability management can always have a full-context view of what vulnerabilities they have, how they should be prioritized, and how to safely apply updates.

CONCLUSIONS AND NEXT STEPS_

This document has introduced many of the fundamental requirements and capabilities of an infrastructure-focused supply chain security program. However, every organization is also unique with its own internal structure, processes, assets, and tolerance for risk. Supply chain security is also not an all-or-nothing proposition. Most organizations will initially focus in areas where their risk is greatest and where they can align supply chain security as part of the larger security strategy. Eclypsium specializes in supply chain security and in guiding organizations through this journey. If you have questions or would like to assess your initial or ongoing supply chain security strategy, please reach out to the Eclypsium team at info@eclypsium.com.