NEED SECURE SUPPLY CHAINS? START WITH THEIR DNA_

By now there's little debate about whether the challenge of securing digital supply chains is one of the most critical jobs facing today's cybersecurity teams. We can debate whether it's "the" most important or just in the Top 5. We can debate how much we should spend on it. We can debate whether the task belongs to traditional InfoSec teams or whether it belongs in the emerging office and budget of the "Chief Supply Chain Officer," the CSCO. We just can't argue the importance and priority of securing our supply chains any longer.

We know this because of two real and present realities:

1. Demonstrated, damaging history of recent supply chain attacks
2. New mandates coming from governments and regulatory bodies

The real question is "How?" If by "secure" we mean our supply chains must be "free from danger" or "safe" or "free from risk of loss," then how do we do this? How can we hope to secure something so sprawling and vast as the supply chains that deliver our information and communications technology (ICT), whether as code or as devices?

RECENT SUPPLY CHAIN ATTACKS_

The first of the "two realities" cited earlier – *the crippling string of recent supply chain attacks* – speaks for itself:

- Pulse Secure network devices: as the world turned remote in 2021, employers and knowledge workers began to rely heavily on VPNs and networking equipment. By honing exploits against known firmware-level vulnerabilities in VPNs, routers and switches, attackers were effectively able to create their own "supply chain effect" by identifying end users who had employed this gear.

- ASUS ShadowHammer: Attackers were able to modify the ASUS Live Update Utility to deliver properly signed, yet malicious updates to more than 57,000 downstream users. The LiveUpdate Utility is particularly significant as it provides BIOS and UEFI firmware updates in addition to software updates, allowing attackers to potentially deliver malicious system firmware.

- Fortinet network devices: Just as the exploit of Pulse Secure devices described above is a de facto example

of supply chain attacks focused on downstream users, Fortinet devices were targeted heavily in 2022 as an amplifying force. Used extensively throughout on-prem, cloud-based, and virtual networks around the world, these Fortinet devices .

- SolarWinds SUNBURST: Sophisticated attackers were able to infiltrate the SolarWinds supply chain and implant malicious code within the source code of their IT Monitoring and Management platform known as Orion. The compromised code was subsequently published and delivered to more than 18,000 federal and private sector customers including a variety of leading technology companies. By infecting IT management tools, attackers had the opportunity to compromise a wide range of servers and assets including the ability to update device firmware.

- Log4j - LOG4SHELL: A critical security flaw in the Log4j framework allows cybercriminals to compromise vulnerable systems with just a single malicious code injection. The vulnerability is associated with the user activity logger known as Log4J, a logging library freely distributed by the Apache Software Foundation. Java is implemented across a wide range of digital products beyond just application, including cloud solutions, web servers and embedded firmware, making each of these products vulnerable to exploitation through the Log4Shell vulnerability. Because this security flaw is so widespread and most organizations are unaware that they're impacted, an exploitation frenzy is currently underway in the cybercriminal world: security researchers have identified approximately 10 million Log4Shell exploitation attempts every hour with retail, technology, financial services and manufacturing segments receiving the highest number of attacks.

It's clear from these attacks that both nation-state and criminal adversaries see supply chain attacks as having a promising ROI, with the ability to increase a given attack's impact and deliver many more targets into the campaign. But if our news feeds point to the urgency of the problem, the second of the "two realities" cited in the opening paragraph – mandates pouring from governments and regulatory bodies – may actually point out a useful path forward.

ASSESSMENT OF THE CRITICAL SUPPLY CHAINS SUPPORTING THE U.S. INFORMATION AND COMMUNICATIONS TECHNOLOGY INDUSTRY

Report from U.S. Dept of Commerce and U.S. Dept of Homeland Security dated February 2022

MANDATES FROM GOVERNMENTS AND REGULATORY BODIES_

A February, 2022 joint report from the United States Departments of Commerce and Homeland Security – "Assessment of the Critical Supply Chains Supporting the U.S. Information and Communications Technology Industry" – called out two major areas of focus in their prescription for addressing these problems:

1. Better management of open source software used throughout the supply chain
2. Lifecycle management for the firmware embedded in the supply chain's devices

The first requirement was not a surprise to most, as the challenges of addressing open source software (OSS) have become the focus of entire industries. But the second took many by surprise: no other federal report had explicitly called out the need for firmware lifecycle management as a way to secure our technology supply chains.

Why this focus? Because all technologies ultimately depend on devices: physical equipment, whether in the form of an end user laptop, networking devices, or servers located locally or in the cloud. And within all of these devices, firmware is pervasive. It's the persistent logic that governs how a device or physical component actually behaves. If devices are compromised at this level all higher-layer data, applications, and services on the device are put at risk.

The government report was explicit and clear:

> *Firmware presents a large and ever-expanding attack surface, as the population of electronic devices grows.*
>
> *Securing the firmware layer is often overlooked, but it is a single point of failure in devices and is one of the stealthiest methods in which an attacker can compromise devices at scale.*
>
> *Over the past few years, hackers have increasingly targeted firmware to launch devastating attacks."*

A Security Week article summarized these findings under a headline that ran, "U.S. Government Issues Stark Warning, Calling Firmware Security a 'Single Point of Failure.'" But

SECURITYWEEK
CYBERSECURITY NEWS, INSIGHTS & ANALYSIS

U.S. Government Issues Stark Warning, Calling Firmware Security a 'Single Point of Failure.'

while this language points out the severity and impact of firmware-based attacks, it might also point towards a surprising solution for supply chain problems in the enterprise: start with securing the firmware.

A FIRMWARE-FIRST APPROACH TO SECURING SUPPLY CHAINS_

In order to manage the real-world risks of modern supply chains, organizations must have an understanding of the hardware and firmware that underlies all technology. Firmware is the embedded code that tells every device and every internal component how to behave, who to trust, how to act, and where to turn for further instructions.

The hardware and firmware layer is also where supply chains become the most complex. Every device – from laptops, to servers, to network and security devices – can contain dozens of components that rely on integrated firmware, supplied by any number of suppliers and sub-suppliers spanning many countries of origin. This creates a near-perfect storm of risk with countless, often unseen opportunities for mistakes or compromises by external adversaries or malicious insiders.

Not only is firmware critical code embedded in devices of all kinds, but there's also far more of it than we might expect:

- Every endpoint relies on 15-20 types of firmware components including UEFI/BIOS, network controllers, storage drives, chipsets, GPUs, PCIe controllers, and more.

- Likewise, every server relies on 20 or more firmware components including highly privileged baseboard management controllers (BMCs) that provide critical out-of-band management capabilities.
- Every network and security device now ships with embedded firmware: In fact many of the devices that have come under serious attacks in recent years – like Fortigate and Pulse Secure – deliver their "operating system" via embedded firmware

When it comes to supply chain risk, the tight integration and dependencies of firmware and hardware create a near-perfect storm. Physical devices have by far the deepest supply chains spread across many countries of origin, creating many opportunities for mistakes or compromises by external adversaries or malicious insiders. Firmware packages are often reused by many suppliers, allowing any vulnerabilities to be passed on to countless products.

Like software, firmware often requires regular updates to prevent the supply chain from being compromised even after a device is deployed. But unlike application and operating system software, many practitioners are reticent to update firmware in their critical systems, for fear of unintended downstream consequences like outages and system failures.

All of these firmware-based risks are multiplied by the number of vendors and device types which will all have their own unique supply chains and risks. To mitigate this risk, organizations must be able to independently verify the authenticity, integrity, and posture of their devices, including independent verification of internal components down to the firmware level. Only by getting to this deep firmware layer can an organization truly verify that their technology is what it claims to be.

Despite the criticality of firmware and the clear advantages adversaries gain in attacking it, real time, proactive security of firmware throughout the supply chain is often overlooked. Firmware- and driver-level code is "below the radar" of security solutions like XDR and vulnerability management, and so goes uninspected, enjoying a special "trust by association".

As a result, this paper proposes a new model for securing device supply chains that would:

1. Flip the current paradigm on its head and treat firmware lifecycle management as a first-class concern throughout device supply chains, with security operations teams appropriately equipped to manage these lifecycles
2. Provide a centralized, authoritative repository of known-good profiles against which firmware and drivers can be compared and that can prove their integrity at any point in their lifecycle or in the supply chain
3. Establish a programmatic and simplified people / process / technology model to patch, update and manage device-level firmware and driver life cycles throughout the supply chain

Most practitioners and administrators in networking and cyber security come from a world where updating firmware is the last, rather than the first resort. According to the UEFI Forum, just 13 percent of enterprises and their security professionals have implemented comprehensive security controls for their firmware. As mentioned earlier, we have been educated and trained – and then have practiced our professions – in a world that is trepidatious if not outright fearful of the power, complexity and "otherness" of embedded firmware and drivers. So much so that up to 85% of endpoints and servers, and over 95% of network devices, reach end-of-life without ever having their critical firmware updated or patched.

The rest of this paper is dedicated to proving the claim that managing firmware and driver lifecycles could be the most immediate way to secure complex supply chains, and to providing a roadmap for those efforts.

FIRMWARE IN THE CONTEXT OF DEVICE SUPPLY CHAIN RISK_

Technology supply chains come in many forms, but from an acquiring organization's point of view, we can break supply chain risk into three fundamental sources: Equipment, Applications, and Services. All three of these categories have been used as real-world threat vectors.

Likewise, all three have the potential to introduce supply chain risk when the technology is initially acquired as well as its ongoing operational lifecycle. When we look at an organization's equipment and devices, virtually all of the long-term risk is tied to firmware: the chips, processors and components are essentially "neutral," without intent. As the"DNA of the device" it's the firmware that provides intent, whether beneficial or malicious.

Additionally, many of the underlying factors that lead to supply chain risk are magnified when we look at the firmware in physical equipment and devices. For example, physical devices have the deepest supply chains spread across the most countries of origin. Code reuse is common at the firmware level, which can lead to extremely widespread issues when new vulnerabilities are found. And naturally, equipment and their firmware are some of the most broadly deployed and high-value assets in an environment, making them also one of the highest impact classes of technology.

When we discuss the varying risk factors and their drivers, it's tempting to see it all in the light of software: "It's all the same issue – it's all code of one kind or another. How is firmware any different?" But for the purposes of this paper we'll take the stance that it is different, especially in the sense that while defenders have gained skill in remediating and hardening applications and OSes, the situation is flipped with firmware: attackers are disproportionately better at weaponizing firmware than security teams are at defending it.

With that perspective in mind, Let's take a closer look at how the three major categories of the technology supply chain – Equipment, Applications, and Services – compare with regard to risk.

A Comparison Of Severity Across DIfferent Classes of Supply Chain Risk

Risk Factors	Managed Services	Applications & Software	Device Firmware & Drivers
"Supply Chain Depth" Risk	Low	Medium	Very High
"Country of Origin" Risk	Low/Medium	Low	High
"Number of Vendors" Risk	Low	High	Medium/High
"Code Reuse" Risk	Low	High	High/Very High
"Impact on Target" Risk	Very High	High	Very High

Eclypsium's risk level alignment across supply chain categories, with risk increasing as device firmware and drivers become targets.

Managed Service

Managed service providers regularly need privileged access into a customer environment in order to provide their services. If the service provider is compromised, attackers can abuse this position of trust to then compromise the provider's downstream customers. This strategy has been particularly popular with ransomware gangs as seen in the well-known Kesaya attacks in which attackers used the services firm to infect up to 1,500 businesses with ransomware. These attacks can have a very broad scope and impact to an organization since a service provider may have access to highly sensitive systems and infrastructure with the customer organization.

The only good news is that most organizations will have a very limited number of providers with this level of trusted access to their technology and infrastructure. Service providers are also fairly self-contained and directly provide services themselves instead of farming them out to other vendors. These traits can limit the opportunities for an attacker to gain access to a service provider.

Applications and Software Supply Chain Risk

3rd-party applications and software naturally account for a significant portion of an organization's technology and likewise its supply chain risk. One of the most notable issues with application supply chain risks is that most organizations will do business with many application vendors. Any one of these vendors could be compromised by external adversaries, malicious insiders, or can introduce known or unknown vulnerabilities in their code. The other most common source of application supply chain risks stems from the reuse of open source software and projects. Any vulnerabilities in these open source libraries can be passed on to the vendor's applications and lead to widespread exposures.

However, outside of open source projects, most application supply chains are not particularly deep. The vendor is responsible for their own code base and even when sold through intermediaries, code is increasingly delivered directly from the vendor's servers. This at least limits the number of entities involved that can lead to a supply chain problem.

Applications require regular updates, and compromising update code or processes has proven to be one of the most common application supply chain vectors. The SolarWinds attack is an example, in which adversaries were able to infiltrate SolarWinds and gain access to source code that was delivered via normal product updates. The scope of an organization's exposure will naturally vary widely based on the nature of the application. In many cases, the exposure may be limited to the theft or loss of the data that the application contains. However, applications that interact with many other systems may enable attackers to spread to other internal assets. Once again, the Solarwinds attack provides a good example, however it is not coincidental that one of the most significant software supply chain attacks involved software that gave attackers access to an organization's critical infrastructure and equipment.

Supply Chain Risk in Hardware, Firmware, and Drivers

The supply chains of physical devices and infrastructure are arguably the most complex, have the most risk, and virtually all of that risk is tied to firmware. Device firmware is the integrated logic that sits below the operating system and controls the actual function of the device and virtually every component within it. Many devices such as network security devices will only have firmware. Other types of equipment such as networking gear rely heavily on integrated firmware or highly customized operating systems that are tightly integrated with the hardware. And while organizations will typically install their own OS image on new servers and laptops, the firmware is often left unchanged. In short, it is the firmware that governs how the device actually works, and it is the firmware in a device that persists.

Unlike applications and services, physical devices have incredibly deep supply chains that often change. A device can easily have hundreds of specialized components and systems within a device, involving dozens of suppliers and sub-suppliers spanning many countries of origin. Suppliers are often selected based on cost and availability pressures as opposed to security concerns. OEMs may also need to quickly pivot to new suppliers to adapt to changing conditions such as recent supply shortages of important components. A vulnerability, external attacker compromise, or malicious insider in any of these many moving parts can put the integrity of the entire device at risk.

And much like applications, firmware is regularly put at risk due to code reuse. Proprietary and open-source firmware components are regularly used by a variety of vendors, meaning vulnerabilities can be passed on to dozens of vendors. As an example, several leading equipment vendors were affected by UEFI vulnerabilities tied to a common firmware SDK.

Likewise, the Ripple20 vulnerabilities in a commonly reused TCP/IP stack were passed on to dozens of vendors. But more importantly, these firmware-level vulnerabilities aren't just theoretical – they're rapidly and actively being exploited:

- HP recently rushed to patch 16 high-impact UEFI firmware vulnerabilities associated with active exploits in its Integrated Lights Out management interface
- CISA warned earlier this month about active exploits targeted at some 60 or so Cisco router vulnerabilities
- Ars Technica wrote about firmware exploits in

February of this year under the headline "Russia's most cutthroat hackers infect network devices with new botnet malware", referring to Cyclops Blink attacks on firewalls

Organizations usually buy equipment from a variety of vendors. Laptops, servers, and network infrastructure will all naturally have very unique supply chains. However, even products from the same vendor can vary from model to model or even between generations of the same model. And finally, an equipment supply chain compromise can be particularly damaging in terms of impact and scope. Any built-in vulnerabilities or threats would naturally be shared by similar devices, which could put an organization's entire fleet at risk. And with access to the most privileged code on a device, attackers would be free to pursue almost any malicious activity from stealing data to establishing stealthy persistence, to permanently disabling – bricking – devices and infrastructure. Lastly, it is worth noting that the physical equipment and its firmware is the logical foundation of other aspects of the supply chain. Any applications and services will ultimately run on hardware, and if the underlying infrastructure is compromised, then the security of higher layers is either suspect or, in high risk environments, completely lost.

THE ROLE OF FIRMWARE IN SUPPLY CHAIN ATTACKS_

Firmware is the first code to run on a device, and by sitting "below" the kernel it is also some of the most privileged code. These traits naturally make firmware very valuable to an adversary across many phases of a cyberattack. While a full analysis of the many ways attackers use and abuse firmware is beyond the scope of this document, let's review some of the most common ways firmware is used specifically in supply chain attacks.

Initial Access Vector

Firmware can be abused in multiple ways to provide adversaries with initial access into an organization.

- Firmware implants or backdoors can be introduced to the product before it is ever delivered to the eventual customer
- It is important to note that an implant can be introduced into firmware in several ways
- A variety of widespread vulnerabilities could be exploited by virtually any entity in the supply chain in order to modify firmware from another supplier
- These same concepts apply to firmware updates as well, where attackers can directly compromise the source vendor or take advantage of insecure update processes.

Persistence

Firmware also provides adversaries a way to maintain their position on a device that is off of drives and beyond the realm of the operating system.

- Through firmware, attackers easily recover even if a threat is detected and the device is completely wiped, reinstalled, or has the system drives replaced
- The recent iLOBleed attacks provide an example from the wild
- Firmware gives access to code that is both highly privileged and long-lived

Defense Evasion

Attackers take advantage of firmware as a way to avoid the prying eyes of security tools.

- Most security tools simply lack the ongoing firmware expertise and focus needed to reliably detect problems
- Firmware sits in a uniquely powerful position capable of subverting the operating system and any security controls that rely on it
- Malicious firmware or bootloaders can allow an attacker to control how the system boots and even directly patch the operating system in order to disable features

Impact

Firmware can be used to cause extensive damage to an organization by stealing or destroying data or fully disabling the device itself.

- Unlike software-level attacks where data can easily be

restored, disabling devices at the firmware level can permanently damage the equipment

- The potential for supply chain attacks to affect an organization's entire fleet of devices can have a devastating impact on operations

A "FIRMWARE FIRST" APPROACH TO SUPPLY CHAIN SECURITY_

Organizations need to be able to verify the authenticity, posture, and integrity of all their critical equipment and devices. A strong supply chain security program will also extend throughout the lifecycle of the product from initial evaluation, acquisition, and ongoing operations of a device.

Given that many traditional security tools lack insight into firmware, organizations may need to develop new processes and tools in order to properly manage the lifecycles of firmware in their devices. A firmware lifecycle management solution can automate and simplify many of the most important tasks, whether or not they are expressly required in order to achieve a reliable supply chain security program.

The "Firmware First" approach is a blueprint to proactively manage the lifecycle of firmware and drivers. In essence, it guides practitioners in a way to Identify, Verify and Fortify firmware and device drivers throughout their complex supply chains.

1. **IDENTIFY: Establish Independent Firmware Visibility and Verification.** Supply chain security has an innate need for independent 3rd party visibility and verification, and this is particularly pronounced when it comes to firmware. While technology vendors naturally play a critical role in supply chain security, customers can't simply rely on the vendor to verify their own integrity when the vendor itself may be compromised.

   As a result, organizations must possess the technical tools to verify that the products a vendor delivers are authentic and have not been altered in the supply chain. All of the following technical steps and processes will rely on the ability to measure and analyze a device's firmware and drivers to identify vulnerabilities, threats, or other signs of tampering.

2. **VERIFY: Automate Verification In Three Distinct Phases.** Verification is one of the most critical steps, and can be broken down into three unique phases of activity that span from pre-purchase planning to roll-out and then to continuous monitoring.

   ### 2.1 Evaluate Prospective Equipment For Vulnerabilities Before Acquisition

   IT and Security Teams should analyze all devices and components for known vulnerabilities and misconfigurations as part of the selection process. A reputable vendor should ensure that their products and all underlying components are free of weaknesses. Also vendors should be assessed to see if they deliver software bill of materials (SBOMs) for the firmware included in the product. These SBOMs provide a clear declaration of the firmware in the device that staff can use as a reference later, after devices are ultimately delivered and deployed.

   Additionally, an OEM will need to ensure that a wide variety of low-level device configurations and security settings are properly enabled and working together. For example, for a Secured-core PC, an OEM will need to properly integrate a variety of technologies from multiple vendors such as System Guard, System Management Mode, and the Trusted Platform module. Small misconfigurations or mistakes could easily put the system at risk in unexpected ways. When possible, the evaluation should include assessing the vendor's firmware update process for weaknesses such as not requiring signed firmware or sending update traffic in the clear.

   A supply chain security platform for firmware and drivers can easily scan prospective devices for these and many other vulnerabilities and weaknesses. This can provide critical insight into each vendor and their dedication to delivering secure products, as well as verifying the security of their own supply chain.

2.2 Scan All New Assets for Firmware Vulnerabilities and Threats

It is critical that organizations verify the integrity and security posture of all newly acquired devices down to the firmware level, even when dealing with a trusted vendor. Suppliers and components can regularly change, which could introduce new vulnerabilities or threats. Devices can also potentially be tampered with after leaving the manufacturer such as during warehousing, at a value-added reseller, or during the logistics and delivery process. In addition to scanning for vulnerabilities, an analysis should verify that all received firmware cryptographically matches the published firmware available from the vendor or component supplier. Likewise, scans should include checks for known firmware threats. Even as threats evolve, attackers will often reuse components of previous firmware threats as seen in the recent MosaicRegressor implant which reused components of the Hacking Team UEFI implant. This analysis should be applied to all devices and updates regardless of how they are acquired. For example, even if a new laptop is sent directly to a remote user, the device should be scanned or assessed remotely to ensure the integrity of the device. Likewise, teams should scan any equipment acquired as a result of any M&A activity.

2.3 Continuously Evaluate All Updates and Monitor Existing Device Behavior

If a vendor or supplier is compromised by an attacker, it is entirely possible that malicious code will be delivered as properly signed and otherwise seemingly "valid" code from the vendor. Simply comparing firmware to the vendor's approved whitelist would naturally not reveal the threat since the threat would be baked into the "approved" version. As a result, organizations should continue to monitor the behavior of firmware after it is acquired or after an update. Unlike many other forms of software, firmware tends to be remarkably predictable in terms of behavior, which will allow malicious behavior to stand out. Security teams may be able to detect some obvious malicious behaviors such as reaching out to known malicious domains, while more subtle methods may require a firmware lifecycle management platform. It is important to note however that firmware within components such UEFI, BMCs, or Intel's AMT can have their own network stacks which will not be seen by the host operating system, making it important to monitor the behavior of the firmware directly.

3. FORTIFY: Patch and Update Firmware Throughout the Supply Chain

One of the critical issues pointed out in the joint Department of Commerce and Department of Homeland Security report cited earlier was the lack of updates and patching for critical firmware:

> *Firmware updates present a major logistical challenge for many enterprises. In many instances, device firmware is never updated or may only be updated in an emergency.*

Why? As the government report goes on to say, the update process for firmware is never simple. "For example, it may not be clear what the latest firmware update is for a particular device. As a result, users may not be able to quickly determine whether the device's firmware is up to date. For devices with firmware that is not cryptographically signed and secure, devices may be updated with unsigned code, meaning firmware could be rewritten without needing any verification from the user. A once-trusted device may no longer be trusted as secure after an unencrypted update."

Organizations clearly need assistance in updating their firmware. That assistance needs to include locating the components that require updates and finding vetted and trusted update binaries with which to replace them. It also needs to include tight integration with standard IT automation tools like Microsoft SCCM or Intune, asset management solutions like Tanium, and endpoint visibility solutions like OSQuery.

Bearing in mind that "the cloud" is just "someone else's hardware," we need to constantly remind ourselves that the modern ICT supply chain also includes cloud platforms like AWS, Azure and Google Cloud. As such, organizations need their firmware lifecycle management

solutions to have rich APIs that integrate with the hardware beneath these leading cloud platforms. Only by assuring an update path for the firmware in those underlying systems can we fully assure our supply chains.

CONCLUSIONS AND NEXT STEPS_

Device and hardware supply chains are naturally complex and rely on highly interdependent webs of implied trust. Without the ability to independently verify the integrity and risks of the millions of lines of firmware that serve as the "DNA" for all this equipment, an organization's entire security program will be built on the shakiest of foundations.

To address these risks, organizations need to ensure that they can independently assess the security of their supply chain at all times. A lifecycle management solution for firmware – a tool that can identify, verify and fortify firmware code throughout technology ecosystems – is a key part of ensuring these efforts are done thoroughly, easily, and consistently. This ensures organizations never have to rely on implied trust, and instead can proactively verify their technology and mitigate risk before it ever becomes a problem.

To learn more about how Eclypsium can protect your organization, please contact us at info@eclypsium.com.