RESEARCH MEMO

The Missing Middle: Addressing the Absence of Firmware Security

By Michael Sugden

January 25, 2024

INTRODUCTION

Coinciding with Russia's invasion of Ukraine on February 24, 2022, Russian cyber operatives targeted thousands of satellite modems and routers throughout Ukraine and other parts of Europe. This attack, dubbed AcidRain by threat intelligence group SentinelLabs, targeted the firmware in Viasat KA-SAT modems, wiping the devices' file system and storage device files.[1] Firmware is the code embedded into a device's hardware and functions as a bridge between software and hardware. The attack resulted in the abrupt loss of network connection for thousands of customers and a "huge loss in communications in the very beginning of the war." The hack rendered 30,000 modems completely inoperable, all of which needed to be replaced.[2]

AcidRain demonstrated how disruptive and destructive firmware attacks can be. In this case, tens of thousands of consumers had to throw out their devices, as an update or patch at the firmware level would have been impossible for reasons particular to the Viasat modems. Nevertheless, U.S. federal lawmakers and executive branch officials have not paid sufficient attention to the risk of firmware attacks. Even the CHIPS and Science Act of 2022, a benchmark, bipartisan law promoting the development of advanced technology in the United States, mentions the word "firmware" only once in its 394 pages.[3] By comparison, the same law mentions "software" 45 times and "hardware" 15.

In 2018, the National Institute of Standards and Technology (NIST) issued guidelines emphasizing the dangers of firmware attacks and providing minimum best practices for securing firmware. However, there is no enforcement mechanism or regulating body to make sure technology companies comply with these guidelines.[4] Compliance is entirely discretionary. Companies that do not make the responsible choice to produce secure-by-design products are offloading the risks to their consumers, including critical infrastructure owners and operators and government entities.

1. Juan Andres Guerrero-Saade and Max van Amerongen, "AcidRain | A Modem Wiper Rains Down on Europe," *SentinelLabs*, March 31, 2021. (https://www.sentinelone.com/labs/acidrain-a-modem-wiper-rains-down-on-europe)
2. Sam Cohen, "AcidRain Malware and Viasat Network Downtime in Ukraine: Assessing the Cyber War Threat," *Just Security*, September 12, 2022. (https://www.justsecurity.org/83021/acidrain-malware-and-viasat-network-downtime-in-ukraine-assessing-the-cyber-war-threat)
3. The CHIPS and Science Act of 2022, Pub. L. 117-167, 136 Stat. 1366. (https://www.congress.gov/117/plaws/publ167/PLAW-117publ167.pdf)
4. Andrew Regenscheid, U.S. National Institute of Standards and Technology, Computer Security Resource Center, "NIST SP 800-193: Platform Firmware Resiliency Guidelines," May 2018. (https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-193.pdf)

As private and public consumers purchase ever more electronic devices, the rate of cyberattacks only grows.[5] While guidelines, standards, and regulation have tackled some cybersecurity issues, firmware vulnerabilities remain largely untouched either by government regulation or public-private collaboration.

This memo begins by explaining what firmware is and then describes the risks associated with it, including how attackers target firmware. The memo concludes with recommendations for addressing firmware security as part of the design process, updating federal firmware security guidelines, and incentivizing compliance with those guidelines, as well as investing in key technologies. Together, these recommendations provide a roadmap for how policymakers should begin to secure the code that lies at the foundation of billions of devices in the United States alone.

BACKGROUND: WHAT IS FIRMWARE?

Firmware is the code embedded into a device's hardware. It functions as the bridge between the device's software and hardware, directing the hardware how to operate.[6] For example, when a user presses the power button on their personal computer, the firmware communicates this stimulus from the physical button to the code operating the computer, resulting in the device turning on. A personal computer has firmware for every piece of hardware that operates as part of the device, including its many different chips, monitors, microphones, cameras, and cooling systems.[7]

Firmware code is most commonly found in two types of embedded memory: read-only memory (ROM) and flash memory. ROM is a permanent storage medium that cannot be updated. ROM was more common in older devices but can still be found in simpler ones that manufacturers do not expect to need updates over the product's lifetime, such as a smart lightbulb. Different types of ROM have different characteristics, such as programmable ROM that allows for limited updates.[8] Flash memory, on the other hand, is where the majority of firmware is stored today. This can be easily updated in large patches over the air, meaning companies can issue and implement firmware patches without on-site operators and without a wired connection. This is how smartphones receive firmware updates, for example.[9] Over-the-air patching makes firmware more versatile and efficient, serving a much wider range of products. However, attackers can use the same over-the-air method to insert malware into the firmware, as discussed below.

While firmware is found embedded in every piece of hardware, users do not directly interact with it during regular usage of a device, although they may access firmware settings in specific interfaces. Typically, users interact with a device's Operating System (OS). An OS manages all the software, hardware, and firmware on a device and is designed to allow users to interact with their device through a graphical user interface or command-line interface.

5. Chuck Brooks, "Alarming Cyber Statistics For Mid-Year 2022 That You Need To Know," *Forbes*, June 3, 2022. (https://www.forbes.com/sites/chuckbrooks/2022/06/03/alarming-cyber-statistics-for-mid-year-2022-that-you-need-to-know/?sh=5ac092697864)
6. Ben Lutkevich, "Firmware," *TechTarget*, accessed December 21, 2023. (https://www.techtarget.com/whatis/definition/firmware)
7. "What is Firmware," *Techslang*, accessed November 27, 2023. (https://www.techslang.com/definition/what-is-firmware-in-computer)
8. "What is an Operating System?" *GCFGlobal*, accessed December 21, 2023. (https://edu.gcfglobal.org/en/computerbasics/understanding-operating-systems/1)
9. "What is the Difference Between Flash Memory and EEPROM?" *electronicsforu.com*, August 19, 2023. (https://www.electronicsforu.com/technology-trends/learn-electronics/eeprom-difference-flash-memory)

Simply put, these interfaces are what the user sees on the screen when operating any device. The three most common examples of OSs are Microsoft Windows, macOS, and Linux.[10]

Even though an OS manages firmware, the OS has a limited ability to secure the firmware. An OS's antivirus scan diagnoses the status of software but not firmware, as the latter operates on a separate and lower level.[11] Malware at this lower level is not only hard to detect but also difficult to remove. Even a full hard drive wipe, which removes all data previously stored on the device, likely would not rid a device of malware at the firmware level.[12]

This is especially problematic because modern firmware has a wide range of functions. Should attackers successfully infiltrate firmware, they can gain operational control of a device's hardware, such as microphones, webcams, keyboards, screens, and individual chips vital to the functioning of the device.[13] This allows attackers to spy on activity, exfiltrate data, remotely control a device, or render a device inoperable.

Most firms, including tech giants such as Google and Amazon, outsource the production of their firmware to original design manufacturers (ODMs) or systems manufacturers.[14] For example, the firmware on a Dell laptop is usually not written by Dell employees. Instead, companies ask ODMs to design their firmware based on their exact specifications. The ODM industry is highly concentrated, with the top eight ODMs producing roughly 94 percent of outsourced firmware.[15] All eight of these ODMs are Taiwanese-based companies. Non-Taiwanese ODMs, such as the Chinese company Inspur, make up a very small fraction of the market. Smaller, independent contractors also exist but often cannot complete large-scale projects demanded by major companies.

Often, when a manufacturer purchases firmware from the ODM, the responsibility for managing problems down the line is not contractually addressed. For software, by contrast, purchase agreements and management agreements usually map which party will be responsible for the code if something goes wrong or if the software needs to be updated. The lack of contractual specificity creates confusion when firmware needs to be fixed. As a result, end users sometimes must replace their hardware because nobody can patch the firmware vulnerabilities. The end user and the device manufacturer may not have the technical knowledge to patch the device because they did not write the firmware, and the ODM may have no contractual obligation or relationship with the consumer to issue a patch. This is not always the case, however, and in many cases device manufacturers will take responsibility to prevent device disaster, even if not contractually obligated.

10. "What is an Operating System?" *GCFGlobal*, accessed December 21, 2023. (https://edu.gcfglobal.org/en/computerbasics/understanding-operating-systems/1)
11. Samuel Gush, "What Is Firmware Malware and How Can You Prevent Infections?" *Make Use Of*, May 5, 2021. (https://www.makeuseof.com/what-is-firmware-malware-and-how-can-you-prevent-infections)
12. Jeff Phungglan, "Does factory reset remove virus and malware?" *MacPaw*, accessed December 21, 2023. (https://macpaw.com/how-to/does-factory-reset-remove-virus)
13. Samuel Gush, "What Is Firmware Malware and How Can You Prevent Infections?" *Make Use Of*, May 5, 2021. (https://www.makeuseof.com/what-is-firmware-malware-and-how-can-you-prevent-infections)
14. "ODM (original design manufacturer)" *TechTarget*, accessed December 21, 2023. (https://www.techtarget.com/whatis/definition/ODM-original-design-manufacturer)
15. Jim Hsiao, "Servers Report Database – 2Q 2023," *DIGITIMES Research*, July 2023. (https://www.digitimes.com/news/a20230804RS400.html)

FINDINGS: HOW AND WHY IS FIRMWARE VULNERABLE?

Firmware attacks have been rapidly increasing since 2019.[16] A 2021 Microsoft report found that 83 percent of surveyed companies had suffered a firmware-related cyberattack, an alarming percentage given the lack of media and government policy attention to firmware vulnerability.[17] Experts agree that while firmware is significantly harder to attack than software, a successful firmware attack provides hackers more "bang for the buck" than some software attacks. Due to the complexity of these attacks, experts usually assess nation-states or state-backed groups to be the culprits, as they can more easily obtain the resources required. And as more firmware is installed in devices and performs a greater range of tasks, the opportunities and incentives for attackers to infiltrate it increase.

Firmware Vulnerabilities

Firmware is more difficult to attack because it operates at a separate level from software and does not directly interface with the internet. Nevertheless, there are four main ways that attackers can insert malware into firmware. In each of the following instances, the malicious actor could be an outside attacker or an insider threat.

First, whether due to thoughtless design or a lack of rigorous security testing, some firmware has weak code that leaves holes in the system for malicious actors to exploit. Once discovered, these holes become known vulnerabilities, which ideally can be patched and monitored. Yet before discovery, these holes are known as "zero-days," which attackers can exploit undetected. This makes knowledge of zero-days especially valuable.

Second, a malicious actor can also gain access somewhere in the firmware's supply chain. The first point of concern is at the ODM facility, where a rogue employee or other unauthorized party could contaminate the firmware during the initial build process. A malicious actor could also infiltrate the build process during the delivery of the firmware to the device manufacturer. In some cases, a third party may inspect the firmware before shipment to the device manufacturer, creating another point of access. Lastly, an intrusion can occur after the firmware reaches the device manufacturer. Such attacks are more likely if there is not a proper security system in place to verify that only authorized employees can alter the code.

Third, if the firmware requires an update or patch, attackers could corrupt that process. If the firmware allows for over-the-air updates and the updating process does not contain the proper safeguards, malicious actors can insert malware into the distribution process. If the manufacturer's network is hacked, an attacker can gain access to the build process over the air.

Even if the firmware does not need an upgrade, devices themselves often need repairs. If the repair service does not have strong security measures, a malicious actor could infiltrate the supply chain.

Finally, a malicious actor can gain physical access to the device and rewrite the firmware. For firmware that cannot receive over-the-air updates, this is the only way to tamper with firmware while the device is in the consumer's possession.

Notable firmware-based attacks have used one of these four main avenues of access. A suspected Chinese campaign dating back to 2021 targeted an appliance made by cybersecurity firm SonicWall that provides secure remote

16. James Moutsos, "5 Reasons Firmware Attacks Have Skyrocketed (How to Protect Your Hardware)," *Dynamix Solutions*, July 29, 2021. (https://dynamixsolutions.com/5-reasons-firmware-attacks-have-skyrocketed)
17. "Security Signals," *Microsoft*, March 2021. (https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RWPStZ)

network connections. The malware, which targeted unpatched appliances, stole user credentials and enabled the attacker to remotely control the device. This attack survived routine firmware updates by scanning for incoming updates every 10 seconds then backing up the malware file and copying itself onto the new files.[18] This allowed the malware to remain undetected at the firmware level for months or possibly years.

In a similar attack attributed to Chinese government-linked actors, the attackers surreptitiously changed the firmware on targeted routers by exploiting the administration and communication links between assets centrally controlled by the router.[19] The attackers are believed to have used the firmware vulnerabilities in subsidiary companies to pivot to corporate networks in the United States and Japan to steal intellectual property and sensitive data.

Another recent attack involved compromised Wi-Fi routers from the Chinese company TP-Link. In this instance, a Chinese advanced persistent threat group implanted malicious firmware that provided the hackers with remote control of the affected devices, file-extraction capabilities, and access to communications between targets.[20] The attackers apparently sought to target European government officials' information.

A third notable example was the Russian Industroyer attack on Ukraine's power grid in 2016, which attacked a firmware vulnerability in the system.[21] The first known use of malware to target an electrical grid, Industroyer caused thousands of local blackouts.[22]

While these examples are not exhaustive, they demonstrate the damage that firmware attacks can cause.

Firmware Is Often Not Secure-By-Design or Secure-By-Default

Secure-by-design is the idea that a device should be built with a reasonable ability to protect itself against malicious activity and attempts by unauthorized users to gain access to or control of the device. Similarly, secure-by-default is the concept that a device should be secure by the time a consumer receives it, with no additional updates or features needed. In the past few years, there has been a significant increase in the emphasis on secure-by-design frameworks, culminating in an April 2023 report by the U.S. Cybersecurity and Infrastructure Security Agency (CISA), titled "Shifting the Balance of Cybersecurity Risk: Principles and Approaches for Security-by-Design and -Default."[23] This report, published in cooperation with cybersecurity agencies from six foreign partner nations, urges manufacturers to adopt these standards.

18. Daniel Lee, Stephen Eckels, and Ben Read, "Suspected Chinese Campaign to Persist on SonicWall Devices, Highlights Importance of Monitoring Edge Devices," *Mandiant*, March 8, 2023. (https://www.mandiant.com/resources/blog/suspected-chinese-persist-sonicwall)
19. U.S. Cyber and Infrastructure Security Agency, Press Release, "CISA, NSA, FBI and Japan Release Advisory Warning of BlackTech, PRC-Linked Cyber Activity," September 27, 2023. (https://www.cisa.gov/news-events/news/cisa-nsa-fbi-and-japan-release-advisory-warning-blacktech-prc-linked-cyber-activity)
20. Itay Cohen, Radoslaw Madej, and the Threat Intelligence Team, "The Dragon Who Sold His Camaro: Analyzing Custom Router Implant," *Check Point Research*, May 16, 2023. (https://research.checkpoint.com/2023/the-dragon-who-sold-his-camaro-analyzing-custom-router-implant)
21. Saed Alrabaee, Mourad Debbabi, and Lingyu Wang, "A Survey of Binary Code Fingerprinting Approaches: Taxonomy, Methodologies, and Features," *ACM Computing Surveys*, January 2022, page 5. (https://dl.acm.org/doi/10.1145/3486860)
22. André Lameiras, "Industroyer: A cyber-weapon that brought down a power grid," *WeLiveSecurity*, June 13, 2022. (https://www.welivesecurity.com/2022/06/13/industroyer-cyber-weapon-brought-down-power-grid)
23. U.S. Cybersecurity and Infrastructure Security Agency, "Shifting the Balance of Cybersecurity Risk: Principles and Approaches for Security-byDesign and -Default," April 13, 2023. (https://www.cisa.gov/sites/default/files/2023-06/principles_approaches_for_security-by-design-default_508c.pdf)

However, despite the relevance of those standards for firmware, the word "firmware" is not mentioned once in the entire publication.

In another step toward ensuring security-by-design, the Biden administration announced the "U.S. Cyber Trust Mark" in July 2023, a cybersecurity labeling program for smart devices, expected to enter into force in 2024.[24] Intended to mimic the effects of the U.S. government's "Energy Star" program by enabling customers to proactively choose labeled products, the U.S. Cyber Trust Mark would inform consumers that particular smart devices adhere to cybersecurity guidelines that NIST will publish. Ideally, this program will drive consumer demand for secure-by-design and secure-by-default products, incentivizing more businesses to adhere to cybersecurity best practices. While the program is still in the development phase, there is no indication at this time that it will include NIST guidelines on firmware protection.

Another tool for software supply chain security that has gained prominence is the "software bill of materials" (SBOM). An SBOM lists the components of a piece of software, identifying the origin of each line of code along the supply chain. SBOMs identify the version of each component, its dependencies on other components, the known vulnerabilities associated with each component, and other indicators of risk.[25] SBOMs are a step in the right direction, but existing secure-by-design frameworks do not require a firmware SBOM, and the SBOM development community is generally distinct from the firmware community. Unless this is changed, a key opportunity to improve security will be missed.

The general absence of firmware from leading security frameworks raises the question: How secure can the design of a device be if firmware security is not taken into consideration? If a device's software is built securely but the firmware supply chains are compromised, devices can still be riddled with vulnerabilities.

The Importance of Securing Updates

Even if firmware is designed securely, it is still possible, albeit more difficult, for a determined hacker to corrupt the firmware. If this happens, timely firmware updates can mitigate the problem. However, as noted above, over-the-air updates can present their own security issues. If firms do not secure their over-the-air update methods, malicious firmware can be introduced by threat actors. Under normal circumstances, secure cryptographic signatures would tell a device that an incoming update originates from the authorized company. However, hackers can trick the cryptographic signatures often required in firmware updates.

To prevent this deception, the party responsible for firmware updates uses — or should use — a root of trust (RoT) system. RoT systems prevent tampering with cryptographic signatures by employing hardware that is inaccessible outside its own ecosystem, ensuring only authorized parties can access it.[26] Firms can also use attestation servers to ensure tamper-free firmware. Rather than sending firmware updates over the air, an attestation server provides a secure repository of the validated firmware versions for download by consumers.[27]

24. The White House, Press Release, "Biden-Harris Administration Announces Cybersecurity Labeling Program for Smart Devices to Protect American Consumers," July 18, 2023. (https://www.whitehouse.gov/briefing-room/statements-releases/2023/07/18/biden-harris-administration-announces-cybersecurity-labeling-program-for-smart-devices-to-protect-american-consumers)
25. Georgianna Shea, "A Software Bill of Materials Is Critical for Comprehensive Risk Management," *Foundation for Defense of Democracies*, September 29, 2021. (https://www.fdd.org/analysis/2021/09/29/a-software-bill-of-materials-is-critical-for-comprehensive-risk-management)
26. "What is Root of Trust," *Thales*, accessed December 21, 2023. (https://cpl.thalesgroup.com/faq/hardware-security-modules/what-root-trust)
27. "Server Attestation," *PrivateCore*, accessed December 21, 2023. (https://privatecore.com/resources-overview/server-attestation/index.html)

These measures are not new. RoT architecture is already commonplace for securing software updates and is used by major cloud service providers for both software and firmware. Some smaller companies use RoT systems for software, but they rarely do so for firmware. The insufficiently broad use of secure firmware verification measures leaves devices vulnerable.

NIST Firmware Guidelines Have Critical Gaps and are not Enforceable

The document that provides best practices for firmware security is NIST 800-193, "Platform Firmware Resiliency Guidelines."[28] This document presents a framework for building mechanisms intended to protect firmware against malware, detect compromising attacks, and recover code and critical data in the event of an attack. NIST 800-193 is built specifically for personal computers, servers, and network devices but is also applicable to other systems.

While helpful, NIST 800-193 falls short in key areas. It does not provide standards for how system manufacturers should provide timely updates, nor does it address the common criticism that manufactures fail to support products for their full lifespan. Last updated in May 2018, the document predates more recent work on SBOMs (and other developments in the security field) and therefore makes no recommendations about them. NIST 800-193 also predates the aforementioned firmware-based attacks and therefore does not account for lessons learned from those attacks.

Compounding these shortfalls, adherence to existing NIST 800-193 guidelines is inconsistent across the industry. There is no regulatory body requiring their adoption, nor are there economic incentives for compliance. While some companies may choose to adhere to these guidelines for the safety of their devices, others may just as easily disregard them. And customers often have no way to tell if manufactures are adhering to the guidelines.

RECOMMENDATIONS

Because firmware is present in billions of digital devices across every sector of critical infrastructure, business, and personal enterprise, it would be very challenging and prohibitively expensive for a regulatory body to oversee the security of every line of firmware code created in the United States. Instead, firmware security issues should be tackled via extensive public-private collaboration, involving both Congress and the executive branch. The following five recommendations can help achieve this.

Include Firmware in Secure-By-Design and Secure-By-Default Frameworks

Secure-by-design frameworks help assure consumers that their products are built with security best practices in mind. Critical vulnerabilities are likely to persist if secure-by-design frameworks do not encompass firmware. Today, firmware attacks are less common than software attacks, but if software becomes relatively more secure than firmware, hackers and cybercriminals may switch their focus to firmware to find easier targets. If this pivot occurs, manufacturers and firmware vendors may be caught off guard and forced to rush to rework their security measures.

As such, firmware security should be included in secure-by-design frameworks as soon as possible. Firmware security should also be integrated into the U.S. Cyber Trust Mark plan. It is misleading to inform consumers that a device is secure when an entire category of code has not been checked or verified.

28. Andrew Regenscheid, U.S. National Institute of Standards and Technology, Computer Security Resource Center, "NIST SP 800-193: Platform Firmware Resiliency Guidelines," May 2018. (https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-193.pdf)

Update NIST 800-193

Alongside the implementation of the previous recommendation, NIST 800-193 should be updated to tackle the current gaps in firmware security. The document, which was initially released in 2018, should, at a minimum, add guidance advising firms to provide frequent and secure firmware patches to their systems for at least five years. The updated document should also recommend the creation and delivery of firmware SBOMs that catalog the code's supply chain and determine whether and where any vulnerabilities or bugs reside. Finally, the update should explicitly classify firmware security as a key component of CISA's secure-by-design framework.

The White House should direct NIST to undertake this 800-193 rewrite and should ensure this task receives sufficient priority despite the White House's competing demands. If the barrier to progress is a backlog of work in NIST's Computer Security Division of the Information Technology Laboratory — the department that would lead the rewrite — the division should be allocated additional resources to hire a more robust team.

Incentivize Compliance With NIST Standards and Security Best Practices

U.S. critical infrastructure continues to experience the rapid replacement of analog legacy systems with newer digital ones, so the protection of these digital systems is vital. While the private sector owns and operates most critical infrastructure in the United States, the federal government should begin purchasing only technologies that comply with NIST firmware security standards for critical infrastructure under federal control. Federal authorities should then encourage the private sector to follow suit.

To that end, the White House should develop an executive order laying out a plan for the transition to using only NIST-compliant technologies in all U.S. critical infrastructure. After laying out a general timeline for this goal, the White House should work with Congress to create incentives for critical infrastructure owners and operators to voluntarily adopt this approach and purchase products only from suppliers adhering to NIST standards as soon as possible. For example, the U.S. government could grant compliant firms tax breaks, allow them to write off transition-related expenses, and give them priority for government contracts. The technology market would hopefully react favorably to these incentives and begin adhering to NIST guidance without further government mandates. When the transition period comes to an end, the government should begin working with critical infrastructure regulators to require operators to purchase technology only from companies adhering to federal firmware standards.

Establish and Maintain a National Repository of Threats to Firmware

Existing certification organizations can help verify that firmware and device manufacturers are adhering to the updated NIST standard. With the right tools, such as a malware repository, these groups can take this a step further and certify that the embedded firmware has not been corrupted during the build process.

Congress or the executive branch should task CISA and provide the necessary resources for it to establish and maintain an open-source repository of known malware affecting firmware. To build this repository, CISA will need to work with private-sector partners, since the bulk of malware attacks target private firms. This repository could initially focus on frequently used malware.

Once the repository is active, companies developing firmware should be encouraged to check their code against the repository's contents to ensure their code had not been corrupted. Certification bodies could likewise compare firmware samples against the repository to provide third-party validation.

In order to benefit cybersecurity professionals the most, this open-source repository should be technical in nature. The repository should translate code into an international standard language, such as OASIS or Z2. This way, engineers can use the repository to directly compare the code errors and malware present in their systems.

The repository should also be able to grow seamlessly, since new malware is found constantly in both the public and private sectors.

Allocate Funding for Research and Development

As mentioned earlier, a current obstacle to securing firmware is that OS antivirus scans cannot detect malware in firmware once a device is in the field, since firmware operates on a lower level than software. There is potentially a technical solution to this issue: adding a component built specifically to read and diagnose firmware in real time. Theoretically, this component would broadcast the current firmware code to a control center, allowing engineers to determine if any changes have occurred in a device's code. This process would effectively act as an anti-virus scanner for firmware.

Although research is under way, this technology does not yet exist. Companies specializing in firmware are constantly under pressure to create more features, and only recently was firmware security put in the spotlight. Still, more features are being demanded, and this is a primary driver of profitability for these companies. Congress should secure funding to support a federally funded research and development center or national lab to expedite the creation of this security-minded technology.

CONCLUSION

The increasing threat to firmware and its demonstrated vulnerability to cyberattacks require urgent action, yet both the public and private sectors have neglected firmware in their pursuit of improved cybersecurity. Key security tools, such as secure-by-design frameworks, rarely account for firmware. If this situation persists, hackers will maintain the ability to stay one step ahead. Thus, the government and its relevant industry partners should tackle firmware security issues immediately, and the recommendations presented in this paper provide an appropriate place to start.

Foundation for Defense of Democracies (FDD)

FDD is a Washington, DC-based nonpartisan research institute focusing on national security and foreign policy.

FDD's Center on Cyber and Technology Innovation (CCTI)

The Center on Cyber and Technology Innovation (CCTI) seeks to advance U.S. prosperity and security through technology innovation while countering cyber threats that seek to diminish it. CCTI promotes a greater understanding within the U.S. government, private sector, and allied countries of the threats to and opportunities for national security presented by the rapidly expanding technological environment.

Michael Sugden is a research analyst and editorial associate at FDD's Center on Cyber and Technology Innovation, where he works on issues related to nation-state cyber threats, critical infrastructure protection, and U.S. cybersecurity policy.
