Firmware Integrity: The Greatest Cybersecurity Threat to National Security That No One is Addressing

TRAPEZOID®
Firmware Integrity Management

Foreword

Imagine for one moment that you have purchased every single security tool on the market today. You have integrated them all to perfection. You have hired highly trained cleared staff, locked down SCIFs, and put all the traditional forwardleaning security policies and procedures in place. You have trained your security team extensively on security protocol. You even have a robust cybersecurity insurance policy and an assigned Incident Response team 24/7. As far as you and your Board are concerned, you are meeting all the requirements of a sound cyber-security professional and adhering to the requirements of your chosen security framework.

But what if you are missing the biggest threat to your organization? What if you are completely wrong because the game has changed? What if you do not even know the battlefield you are fighting on? What if the potential cost is billions of dollars and millions of lives, and grave damage to our nation?

The intent of this white paper is to inform you, educate you, and enlighten you to the greatest vulnerability in cybersecurity today. It is to share with you critical information about a threat vector you are likely ill-prepared to fight and have zero chance of detecting or remediating. To make it perfectly clear, you do not know the battle you are fighting and you are losing every single day.

Firmware and hardware roots-of-trust is the new battlefield. Firmware is at the heart of every server, storage device, router, workstation, printer, SCADA device and IoT device on your network. And firmware is currently vulnerable to attack without detection. In fact, 50% of the code on your network is firmware – and none of it is being monitored by a single established security tool. Your organization is also succeptable to firmware attacks from your supply chain, which is predominantly foreign. Threat actors are compromising hardware at the factory and sending it through to you, and you will never know you have been compromised it until it is too late.

To make matters worse, compromised firmware cannot be removed by rebooting, swapping the hard drives, replacing the memory, or even reinstalling the Operating System (“OS”). Compromised firmware is persistent and sits below the operating system and can remain undetected for months. Stealing data is the low hanging fruit of such an exploit; when someone controls the firmware, they can control the entire system itself.

What does that mean in real world terms? It means a hacker can can take down the Ukrainian power grid and leave 225,000 customers without heat in the dead of winter, just prior to a military attack. It means someone with malicious intent can cause a nuclear centrifuge in Iran to report that it is running at a constant speed when it is actually violently revving up then suddenly stopping to destroy its bearings. It is difficult to understate how devastating and far-reaching a firmware attack can be. It can drive your computer-controlled car off a cliff. It can make your heart monitor readings inaccurate, resulting in your death. At its most benign, someone with firmware access can steal your data. But when someone has access at the firmware level, they control the entire system and can make it do whatever they want. That means all critical infrastructure, power grids, airways, water treatment facilities, telecommunications infrastructure, hospitals, dams, railways, financial systems, and every single military weapons system in our national defense is vulnerable.

This threat is not academic. Over the years, various leaks have revealed that spy agencies have been weaponizing firmware since at least 2006. Recently, the world has caught up, and it is not just the bad

actors looking to extort a few dollars. What is at risk now is our very way of life. Firmware attacks have occurred around the globe and are growing in number as major technology companies like Intel, AMD, and Cisco continue to acknowledge that hardware and firmware is vulnerable. Every single cybersecurity framework from NIST CSF to GDPR, and even the President's Cybersecurity Executive Order, now requires the monitoring of firmware in one way or another, and for good reason.

Meanwhile, insurance companies are now beginning to deny cybersecurity claims using the novel theory that cyber-attacks fall under a "act of war" exemption.[1] Viewed against this backdrop, it is impossible for any organization, be it government, commercial, finance, healthcare, telecommunications, critical infrastructure or military, to claim that it is meeting its obligations to protect citizens, its constituents and the nation if it is not monitoring the greatest cybersecurity threat for which literally no organization is currently prepared.

Trapezoid has spent the last 10 years working on this national security problem, developing the only leading-edge solution to continuously monitor firmware for unauthorized changes and remediation. We hold 3 patents in this specific area of cybersecurity, and are among the world's leading experts on how to combat this threat.

The intent of this white paper is to educate readers on the nature of the firmware and hardware attacks and how pervasive this problem is, what compliance requirements exist for mandating action, and what is available today to help you in your defensive posture to combat this national security crisis.

Bryan K. Mossey
Cyber Security Thought Leader
Chief Growth Officer
Trapezoid Inc.

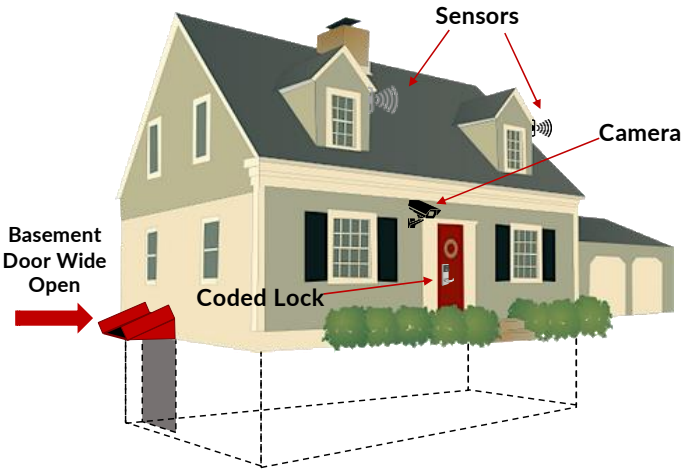
[1] https://www.zdnet.com/article/notpetya-an-act-of-war-cyber-insurance-firm-taken-to-task-for-refusing-to-pay-out/

TRAPEZOID
Firmware Integrity Management

Why Protecting Firmware Matters

To Prevent an Attack at the Foundation you Need to Close the Basement Door

If you think of a typical IT device as a house, most people consider the Operating System (Windows/Linux/MacOS) as the 1st floor and the applications (Word/Outlook/Firefox) as the 2nd floor. Similarly, today's cybersecurity tools are the equivalent of a coded front door lock and security camera guarding the front door, with motion sensors on the 2nd floor to protect that area. The problem, however, is that the controls for the house are in the basement and the basement door is wide open!



Critical home controls are in the basement.

What is Firmware?

Firmware is programmable code that sitst below the Operating System, in the "basement" of every device. This critical code is totally unprotected, and yet it controls all the interaction between the Operating System and Applications with the underlying hardware.

Firmware can be very simple or very complex, varying from device to device. In fact, depending on the complexity of the device, you could have 18 million lines of code executing before you get to the Operating System. And like any other code, firmware can have vulnerabilities that need to be patched. It can be compromised in the supply chain or someone can flash it in as little as 30 seconds. Just one single compromised device can serve as a powerful platform to attack the rest of your network. And all devices have firmware – it's in everything.

Continuing with the house analogy, think of a network as a neighborhood of devices. In that case, imagine how secure your neighborhood would be if EVERY basement door was open and no one was paying attention. Bottom line: because firmware sits below the Operating System, if you control the firmware, you can do anything from stealing data to rendering the devices inoperable.

Where are Firmware Attacks Happening?

Firmware attacks are not new. Some of the first attacks were catalogued in the late 1990s and they continue today. These attacks either infect a device with firmware malware, or, more commonly, exploit vulnerabilities in the firmware provided by the device manufacturer. Firmware is an attractive target because it is persistent, stealthy and allows the attacker complete system control once inside. Firmware incidents generally fall into 3 categories.

1. **Attacks in the Wild**

   These include several successful attacks that were subsequently discovered by the security industry. The following are few examples of firmware related attacks:

- **Chernobyl Virus** - In 1998, the Chernobyl Virus[2] infected more than 1 million Windows 95 and 98 computers causing an estimated $250 Million in damages[3]. The most dangerous functionality about Chernobyl was that it would attempt to write data to the Flash BIOS Chip[4] on the system, so that the system would not boot unless the computer was manually opened up and the chip was reprogrammed.

- **Dell Supplier** - In 2010 a supplier shipped replacement system boards[5] for Dell PowerEdge series Servers that were infected with the w32 Spybot information stealing worm. Dell later identified that the infection was on flash storage on the system board, which, while technically not firmware, was deeper than the hard drive like most infections. Installing a new hard drive would not make the infection go away.

- **Lojax Firmware Rootkit** - In 2018, the security researchers at ESET published details of a UEFI firmware-based rootkit in the wild.[6] The attack group known as APT28, Sofacy, Strontium, and Fancy Bear used Logax to infiltrate the motherboard firmware of PCs and reset machines so they could spy below where anti-virus tools could detect it. Lojax was capable of writing a complete rootkit on the UEFI firmware, installing malware and ensuring the OS would continue to operate at startup, making it impossible to remove the malware or enable a secure boot.

- **VPNFilter** – In May of 2018 the FBI issued a warning telling people that they should reboot their routers because they could be subject to a Russian attack. The idea of rebooting the routers was to wipe the malware from memory. Unfortunately, part of the attack altered firmware so that it could persist across reboots and call out to again download the malware modules that the reboot wiped.[7] As a result, all of the infected devices became permanently unsafe unless the firmware itself was updated. With many of the infected devices no longer supported, the only solution was to throw them away.

2. Unauthorized Disclosures of Government Exploits

These are primarily tools, techniques and exploits used by nation states and include:

- **NSA ANT Catalog** - This is a 50-page classified document listing technology available to the United States National Security Agency (NSA) Tailored Access Operations (TAO) by the Advanced Network Technology (ANT) Division to aid in cyber surveillance. It was published by the German magazine *Der Spiegel* in December 2013. Most devices and techniques had been available since 2006 to the NSA and members of the Five Eyes alliance. Among the tools were a number of firmware related implants. For example, "DEITYBOUNCE" installs a "backdoor" software implant on Dell PowerEdge servers via the motherboard BIOS and RAID controller(s).[8]

---

[2] https://www.f-secure.com/v-descs/cih.shtml
[3] https://www.symantec.com/security-center/writeup/2000-122010-2655-99
[4] https://en.wikipedia.org/wiki/BIOS
[5] http://www.homelandsecuritynewswire.com/dell-replace-server-parts-infected-virus
[6] https://www.welivesecurity.com/2018/09/27/lojax-first-uefi-rootkit-found-wild-courtesy-sednit-group/
[7] https://blog.talosintelligence.com/2018/05/VPNFilter.html
[8] https://en.wikipedia.org/wiki/NSA_ANT_catalog

- **Vault 7** – In March of 2017, WikiLeaks began to publish is a series of documents that detailed Central Intelligence Agency activities and capabilities to perform electronic surveillance and cyber warfare. The files, dated from 2013–2016, include details on the agency's software capabilities, such as the ability to compromise cars, firmware of smart TVs, firmware of most smartphones (including Apple's iOS and Google's Android) and Cisco switches.[9]

- **Hacking Team Leak** – Hacking Team was a Milan-based information technology company that sold offensive intrusion and surveillance capabilities to governments, law enforcement agencies and corporations. Its "Remote Control Systems" enabled governments and corporations to monitor the communications of internet users, decipher their encrypted files and emails, record Skype and other Voice over IP communications, and remotely activate microphones and camera on target computers. In July of 2015 Hacking Team suffered a major data breach of customer data, software code, internal documents and e-mails via a vulnerability in the firmware of an embedded device.[10] Ironically, one of the leaked tools was designed to infect target computer's UEFI BIOS firmware with a rootkit.[11] This disclosure contained tools and data used in the UEFI firmware attack mentioned in the previous section under “Lojax Firmware Rootkit”.

[9] https://en.wikipedia.org/wiki/Vault_7
[10] http://pastebin.com/raw/0SNSvyjJ
[11] https://en.wikipedia.org/wiki/Hacking_Team

3. Disclosures by Security Researchers

A growing group of researchers have been focused on the problem of firmware and hardware security. They are continuously analyzing firmware and hardware for vulnerabilities, developing different proof of concepts and tools to bypass some firmware level code on servers, desktops, network devices, and IoT devices. At security conferences such as BlackHat, DefCon, and others, there have been proofs of concept shared and vulnerabilities disclosed.

Meltdown/Spectre: Since January of 2018, researchers have disclosed a number of chip level vulnerabilities that require firmware/OS mitigation.[12] In order for the vulnerability to be addressed, BOTH the firmware and the OS have to use the correct mitigation technique. Because the mitigations impact performance, the need to understand what firmware/OS combination a system is running has added an operational impact to a serious security issue.

All Modern Compliance Frameworks Require Firmware Monitoring

As a result of the cybersecurity implications of firmware attacks, multiple compliance frameworks are now requiring firmware monitoring as part of every organization's risk-based cybersecurity program. Failure to implement firmware controls can result in costly audit failures. Prior to 2020, auditors often handed out waivers to organizations without firmware controls because compliance was so difficult to achieve. However, beginning in 2021, these waivers began being phased out as a commercially available product for firmware protection became available. This section discusses the various framework requirements and how they impact different sectors.

National Institute of Standards and Technologies (NIST) Special Publications (SP) 800-53r4 and NIST SP 800-53Ar4

These two publications offer foundational guidance and controls that are well respected and often referenced by other frameworks and standards. NIST SP 800-53 is a set of standards and guidelines to help federal agencies and federal contractors meet the requirements set by the Federal Information Security Management Act (FISMA). Today, all federal agencies, and all contractors and subcontractors working within the federal supply chain must comply with these controls.

The most relevant section for Firmware is SI-7 (SI stands for "System Integrity"). There are 16 controls in this section, all of them are related to firmware. The key concept of SI-7 is whether the organization "employs integrity verification tools to detect unauthorized changes to organization-defined software, firmware, and information."

It is worth taking the time to read through these 16 controls to gain a better understanding of these requirements. Does your organization have the controls in place already to meet them? Other controls and frameworks point back to SI-7.[13]

[12] https://meltdownattack.com/
[13] https://nvd.nist.gov/800-53/Rev4/control/SI-7

Centers for Medicare & Medicaid Services (CMS) Information Security and Privacy Acceptable Risk Safeguards (ARS)

CMS provides direction and technical guidance for the administration of the Federal healthcare financing programs and policies. To this end, CMS contracts with 3rd parties to procure a wide range of supplies and services to support Medicare and Medicaid programs. The CMS ARS[14] provides guidance to CMS and its contractors as to the minimum acceptable level of required security controls that CMS and CMS contractors must implement to protect CMS' information and information systems.

The CMS Acceptable Risk Safeguards match NIST 800-53 including important firmware-related controls in sections *SI-2* for *Flaw Remediation*, *SI-3 Malicious Code Protection*, and *SI-7 System Integrity*. In addition, the definition of ***malicious code*** expressly includes firmware.[15]

NIST Framework for Improving Critical Infrastructure Cybersecurity (CSF)

The NIST CSF[16] has five categories: Identify, Protect, Detect, Respond, and Recover. Each of those has various subcategories that offer guidance and point to references for additional guidance. In the Protect Data Security category, PR.DS-6, DS references NIST 800-53 SI-7 and requies that "integrity checking mechanisms are used to verify software, firmware, and information integrity."

The NIST CSF has been widely adopted outside of the Critical Infrastructure Sector. It has evolved into a common framework across multiple industries, both in the US and many countries around the world.[17] It is also being adopted by many state governments in the US, some by reference and other by actual codification. For example, the State of Florida implemented the "Florida Cybersecurity Standards" incorporating the NIST CSF almost word for word. This rule is now part of the Florida Administrative Code, which all Florida state agencies must follow and includes the following language:

> *"In protecting data security, agencies shall: (f) Employ integrity checking mechanisms to verify software, firmware, and information integrity (PR.DS-6). … 1. Application controls shall be established to ensure the accuracy and completeness of data, including validation and integrity checks, to detect data corruption that may occur through processing errors or deliberate actions."* [18]

US Executive Order on Strengthening Cybersecurity

Facing intensifying cybersecurity threats from around the world, President Trump issued Executive Order 13800[19] in May 2017 to improve the nation's cyber posture and capabilities. The order included the following terminology:

> *"Agency heads will be held accountable by the President for implementing risk management measures commensurate with the risk and magnitude of the harm that would result from unauthorized access, use, disclosure, disruption, modification, or destruction of IT and data."*

---

[14] https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/InformationSecurity/Information-Security-Library.html
[15] https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/InformationSecurity/Downloads/RMH_VI_10_Terms_Defs_Acronyms.pdf
[16] https://www.nist.gov/cyberframework
[17] https://www.nist.gov/news-events/news/2019/02/nist-marks-fifth-anniversary-popular-cybersecurity-framework
[18] https://www.flrules.org/gateway/RuleNo.asp?id=74-2.003
[19] https://www.whitehouse.gov/presidential-actions/presidential-executive-order-strengthening-cybersecurity-federal-networks-critical-infrastructure/

It then directs the Agencies to use the CSF to conduct risk assessments and gives departments 90 days to provide a risk assessment report to the Director of Homeland Security. As a result, Agencies need to apply all relevant NIST SP 800-53 controls.

Defense Federal Acquisition Regulations Supplement (DFARS) Clause 252.204-7012 (Safeguarding Covered Defense Information and Cyber Incident Reporting)

DFARS 252.204.7012[20] mandates what a defense contractor that holds *controlled unclassified government information* (CUI) needs to do with respect to "malicious software." The definition of malicious software includes both software and firmware and is expressly called out in the definition. As a result, all of the NIST SP 800-53 controls that deal with firmware apply to any DFARS 252.204.7012 compliance program. Without those controls you cannot detect, report and provide samples of a "malicious software incident" involving firmware as required by the provision. This contractual provision directly applies to any cloud operating at Impact Levels 4 & 5.

It is important to understand that, while many people think that DFARS 252.204.7012 and NIST SP 800-171 are synonymous, they are not. NIST SP 171 only addresses the ***confidentiality*** of the Government's data. It does not address data ***integrity*** or ***availability***. Under DFARS 252.204.7012 contractors must provide the "adequate security" necessary if possessing or handling "covered defense information" (CDI) in connection with the performance of a federal contract. This means that in order to meet the requirements of DFARS 252.204.7012, contractors must also implement the relevant NIST SP 800-53 integrity and availability controls.

Federal Risk and Authorization Management Program (FedRAMP)

FedRAMP[21] is a government-wide program that provides a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services. In order to qualify, Cloud Service Providers (CSPs) must receive an independent security assessment, conducted by a third-party assessment organization (3PAO). These assessments are based on NIST SP 800-53 Rev 4, thus FedRAMP includes the NIST baseline of controls, tailored for cloud computing projects. Both FedRAMP and FISMA categorize information and information systems based on the objectives of providing appropriate levels of information security and on risk impact system levels (Low, Moderate, and High).

New York State Dept. of Financial Services (NY DFS) Cybersecurity Rule

The "Cybersecurity Requirements for Financial Services Companies" adopted by the NY State Dept. of Financial Services (DFS) went into effect in March 2017 (the "NY Cyber Rule").[22] The NY Cyber Rule requires that an institution's board of directors or a senior officer personally annually certify that the Cybersecurity Program meets all the requirement of the NY Cyber Rule.

In performing the NY Cyber Rule risk assessment, a financial organizaiton looking to be compliant should analyze the potential adverse impacts of firmware exploits: e.g. data exfiltration, intellectual property theft, eavesdropping on video/telephony systems, destruction of core operational systems and persistent attack vectors. The organizational risk from any of these potential events could include reportable breaches under

[20] https://www.acq.osd.mil/dpap/dars/dfars/html/current/252204.htm#252.204-7012
[21] https://www.fedramp.gov/
[22] https://www.dfs.ny.gov/docs/legal/regulations/adoptions/dfsrf500txt.pdf

the NY Cyber Rule, service disruption or complete outage, reputational damage and ultimately financial liability.

Given that the NIST firmware controls are specifically designed to mitigate those risks, is it worth the risk to not look at 50% of the code in your organization? Should a financial institution decide not to include controls for firmware exploits in its Cybersecurity Program, it should clearly document why a lack of firmware monitoring is an acceptable risk, or what other controls sufficiently mitigate against the risk of firmware exploits.

Payment Card Industry Data Security Standard (PCI DSS)

Many organizations think that PCI-DSS provisions dealing with "malicious software" and "malware" are limited to employing anti-virus tools. However the PCI DSS documentation clearly states in multiple locations that the definition of "malicious software/malware" applies to both software and firmware.[23]

Financial Services Sector Cybersecurity Profile (PR.DS-6)

The Profile is a scalable and extensible assessment that financial institutions of all types can use for internal and third-party cybersecurity risk management assessment. It is modeled on the NIST CSF but adds a "Governance" category to evidence compliance with various regulatory frameworks both within the United States and globally. The Profile includes the same PR.DS-6 language regarding firmware as does the NIST CSF:[24]

> *"Integrity checking mechanisms are used to verify software, firmware, and information integrity."*

Health Insurance Portability and Accountability Act of 1996 (HIPAA)

HIPAA (45 CFR Part 164.306(a)(2) – In its publication "FACT SHEET: Ransomware and HIPAA,"[25] HHS calls out firmware integrity management as an example of the type of potential risks that should be part of a covered entity's risk analysis and risk management process, even if firmware is not expressly called out in the statute.

> *"For example, although there is a not a Security Rule standard or implementation specification that specifically and expressly requires entities to update the firmware (3) of network devices, entities, as part of their risk analysis and risk management process, should, as appropriate, identify and address the risks to ePHI of using networks devices running on obsolete firmware, especially when firmware updates are available to remediate known security vulnerabilities."*

The footnote to the term firmware then points back to NIST SP 800-53 Rev 4:

> *"(3) Firmware refers to "computer programs and data stored in hardware… such that the programs and data cannot be dynamically written or modified during execution of the programs." NIST SP 800-53 Rev. 4, Security and Privacy Controls for Federal Information Systems and Organizations. (April 2013). Available at http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf."*

[23] https://www.pcisecuritystandards.org/documents/PCI_DSS_Glossary_v3-2.pdf
[24] https://www.fsscc.org/Financial-Sector-Cybersecurity-Profile
[25] https://www.hhs.gov/sites/default/files/RansomwareFactSheet.pdf

HITRUST Alliance Cybersecurity Framework

HITRUST develops, maintains and provides broad access to its widely-adopted common risk and compliance management frameworks; related assessment and assurance methodologies. The HITRUST CSF has numerous references to firmware, is available to HITRUST Qualified Organizations or Qualified Individuals here: https://hitrustalliance.net/csf-license-agreement/ In addition, the HITRUST definition of ***malicious code*** expressly includes firmware.[26]

GDPR

GDPR does not specifically call out firmware. The statement from Nigel Houlden, Information Commissioners Office Head of Technology in the UK about the chip level vulnerabilities "Spectre" and "Meltdown" is clear signaling that breaches at the firmware layer that expose personal information could be penalized.[27]

> *"We are aware of reports detailing potentially significant flaws in a wide range of computer processors, which could affect various operating systems. We strongly recommend that organizations with affected hardware test and apply patches from suppliers as soon as they are released."*
>
> *"All organizations have a duty to keep personal information in their care secure and that involves having layered security defenses in place, including procedures for applying patches and updates, to help to mitigate the risk of exploitation."*

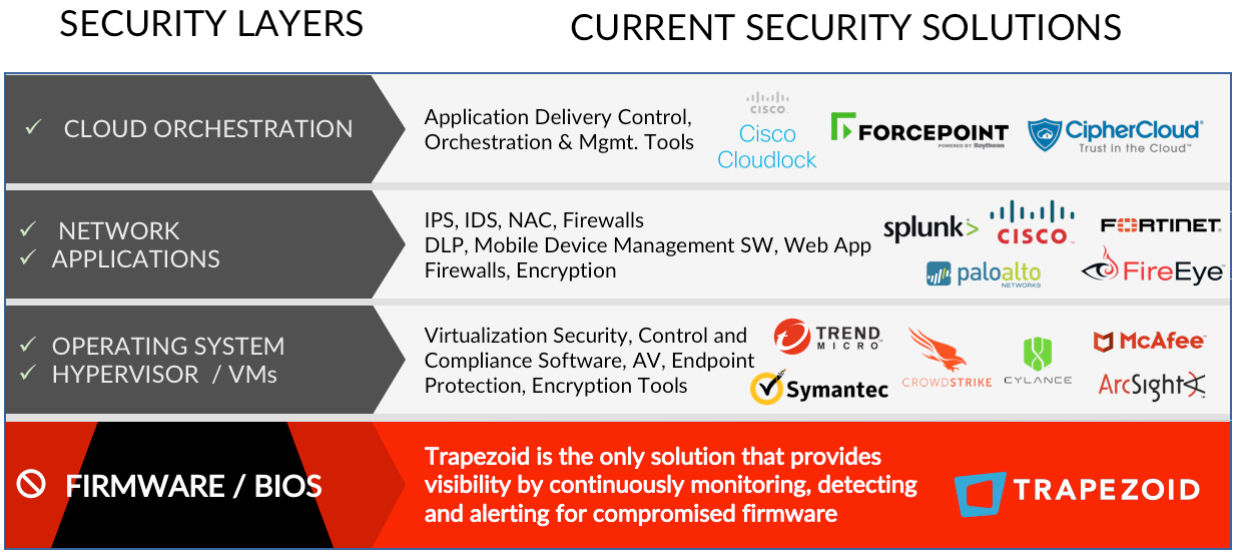
Making Devices Secure with Trapezoid® FIVE

The Issue with Traditional Security Tools

Existing security tools have been focused solving for threats on the other "floors in the house;" on the other layers of the IT stack.

Trapezoid initially focused on cybersecurity integration and operations building out solutions using available commercial off-the-shelf (COTS) products. As we examined the threats, we realized that there were no COTS tools providing any type of visibility at the firmware layer. Seeing this critical gap, Trapezoid took the opportunity to build the first of its kind vendor and device agnostic tool to enable this type of visibility.and developed our Trapezoid® Firmware Integrity Verification Engine (FIVE) to be the "integrity verification tool to detect unauthorized changes in firmware" described by NIST.

[26] https://hitrustalliance.net/content/uploads/HITRUST_Glossary_of_Terms_and_Acronyms.pdf
[27] https://digitalguardian.com/blog/gdpr-meltdown-eu-regulator-sends-warning-chip-flaws



How does Trapezoid address Firmware and Device Integrity?

The Trapezoid® Firmware Integrity Verification Engine (FIVE) collects and analyzes data from four areas of a device or group of devices, which we call ***Integrity Quadrants***:

Hardware – Any change in hardware means that someone had physical access to and control of the system. This could easily lead to changes in firmware. Trapezoid enumerates the hardware of the system and continuously monitors it over time. Any changes in hardware, Trapezoid will detect and alert internally and to any other security reporting and monitoring tools used by the organization.

Firmware – Trapezoid continuously monitors system firmware and provides a *Remote Attestation* capability to organizations. Trapezoid can leverage integrity verification capabilities that manufacturers make available like Intel Trusted Execution Technology[28], Cisco IOS Software Integrity Assurance[29], and Linux Integrity Measurement Architecture[30]. If the manufacturer supplies known good values, Trapezoid will leverage or build a whitelist to compare what is on system to what the manufacturer provides.

Configuration – Trapezoid provides configuration monitoring for changes that could lead to a firmware-based compromise. Changes in certain configuration settings may lead to escalated privileges or may be indicators that an attacker is exploiting a known vulnerability. Simple configuration settings include removal of default passwords or enabling password encryption. More sophisticated configuration checks include looking at the Translation Lookaside Buffer settings on Cisco devices to determine if these have been changed to read/write. The latter is an indicator that someone is using the SYNFul Knock attack to inject unauthorized code into the firmware.[31]

[28] https://www.intel.com/content/www/us/en/support/articles/000025873/technologies.html
[29] https://tools.cisco.com/security/center/resources/integrity_assurance.html
[30] https://sourceforge.net/p/linux-ima/wiki/Home/
[31] https://www.cisco.com/c/en/us/about/security-center/event-response/synful-knock.html

Operational Metrics – When a system is under attack, its operational profile could be impacted. Power or bandwidth consumption may spike unexpectedly due to some change of firmware. Trapezoid's continuous monitoring develops a baseline for each device, that can be analyzed over time allowing it to detect anomalous behavior.

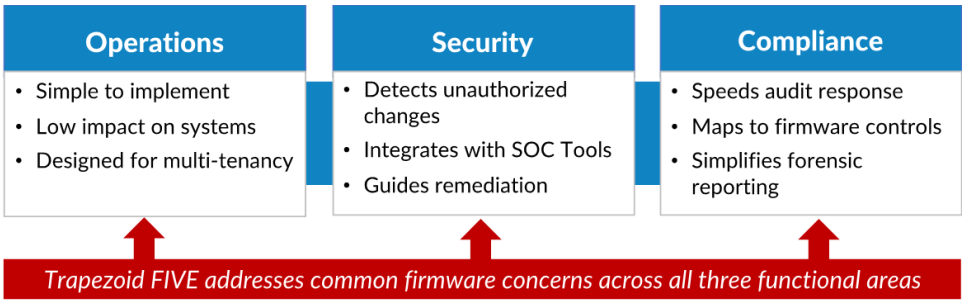
Trapezoid FIVE is device and vendor agnostic. The solution really does not care what type of device it is monitoring. From a large enterprise or cloud server to the smallest IoT device, if data is available for any of these quadrants, Trapezoid will automatically start keeping track of changes. Trapezoid FIVE can continuously monitor an organization's devices for unauthorized changes in firmware and device integrity. Because of its modular nature, Trapezoid works with legacy, current and future deployments.

Trapezoid® FIVE across multiple domains

We designed Trapezoid FIVE to meet the needs of threedifferent constituencies inside organizations: Operations; Security; and Compliance.

Traditionally, firmware has been the responsibility of the Operations team, and they only looked at patching a system if a firmware issue was impacting performance. As exploits at the firmware level become more prevalent, the security team finds that it needs to add this code to all the other code they are monitoring on the organization's system. At the same time, compliance frameworks require that firmware be part of every risk-based cybersecurity program and the compliance team needs to ensure that appropriate controls are in place to address firmware-related risks.

As security professionals coming from an operational environment, Trapezoid's founders understood that trying to address firmware using three different tools would not only be inefficient, it would be a recipe for disaster. With that in mind, we built a single tool that supports all three functions.



- **Operations** has to deploy it within the organization's infrastructure, to it must be easily installed, have minimum impact on the systems that run the organization, and support multi-tenancy to be able to segregate the data of separate organizational units.

- **Security** needs a continues monitoring tool that can detect authorized changes, alert the security tools already deploying within the organization, and help quickly address remediation due to vulnerable firmware.

- **Compliance** needs to quickly understand how the state of systems in the infrastructure map to relevant firmware controls without having to spend weeks trying to understand how specific devices work.

Targeting these three functions with one tool provides important efficiencies for all three groups. From an operational perspective, the ability to quickly provide the compliance team a real-time, up-to-the-minute report reduces the time operational resources have to spend on compliance audits.

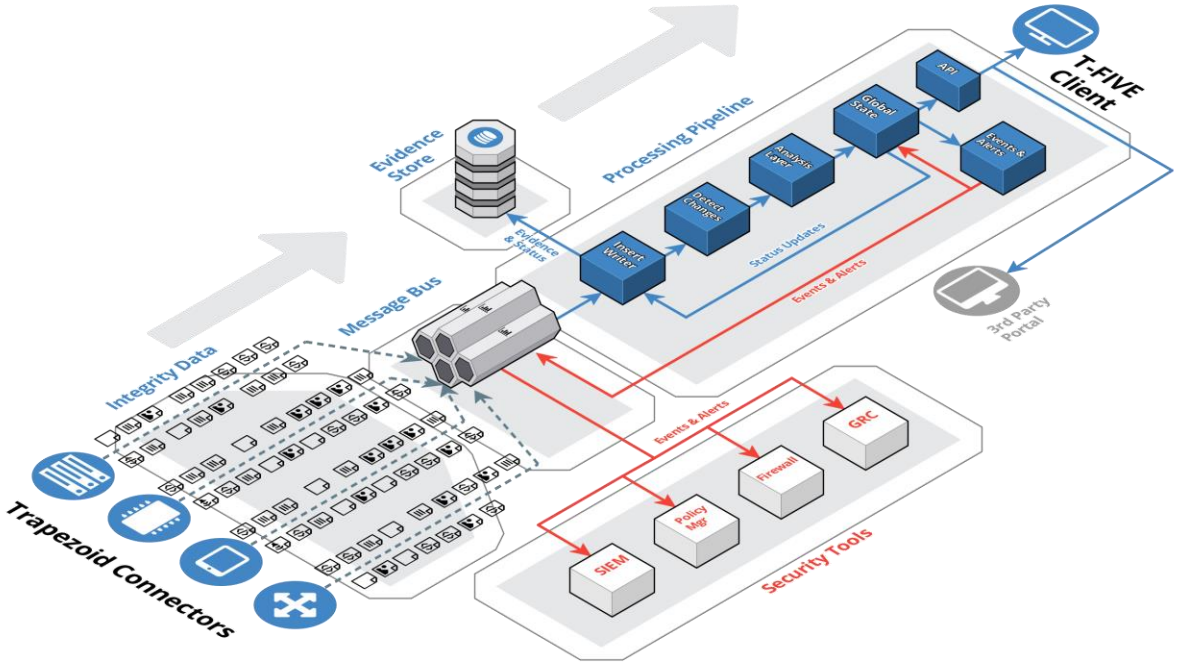
Continuous monitoring means that security can more quickly respond to an issue at the firmware space. At the same time, the organization saves on travel costs associated with reviewing the state of devices at remote sites.

Coordination among Operations, Security and Compliance enhances the procurement process by aligning the interest of all three functions around device integrity. Buying based solely on price lead introduces unacceptable supply chain risk, and lack of integrity verification methods increases the likelihood that exploitable vulnerabilities go undetected.

Trapezoid FIVE Architecture

Designed by Operators and Security Engineers for Operators and Security Engineers

Trapezoid FIVE gives previously unavailable visibility and the ability to continuously monitor all things firmware related on the system. This allows the user to quickly validate the integrity of an organization's infrastructure.



General Architecture

Trapezoid FIVE can be deployed as a standalone tool in one system or distributed across an organization's infrastructure. It fully supports any multitenant environment either as a dedicated instance for each tenant or as a centralized service with multitenant schemas within a single shared database.

Trapezoid FIVE uses **"Connectors"** to access devices via the best available communications method; from REST APIs to basic SSH CLI sessions. The goal is always to use the most secure and out-of-band methodology possible to have the best shot at getting good firmware data. The device type can have an impact on what is available. For example, a brand-new server with the latest and greatest API's and secure boot capabilities will provide far different and better information than a legacy IoT device that can only be accessed by telnet or web interface.

The Connectors deliver device data, which we call **"Evidence"** via secure message bus architecture to an **insert only** forensic database. This is important because once Trapezoid FIVE connects to a device and

begins to continuously monitor it, there is a starting point in time and the data cannot be modified in the database. This supports both forensics and compliance of the devices being monitored.

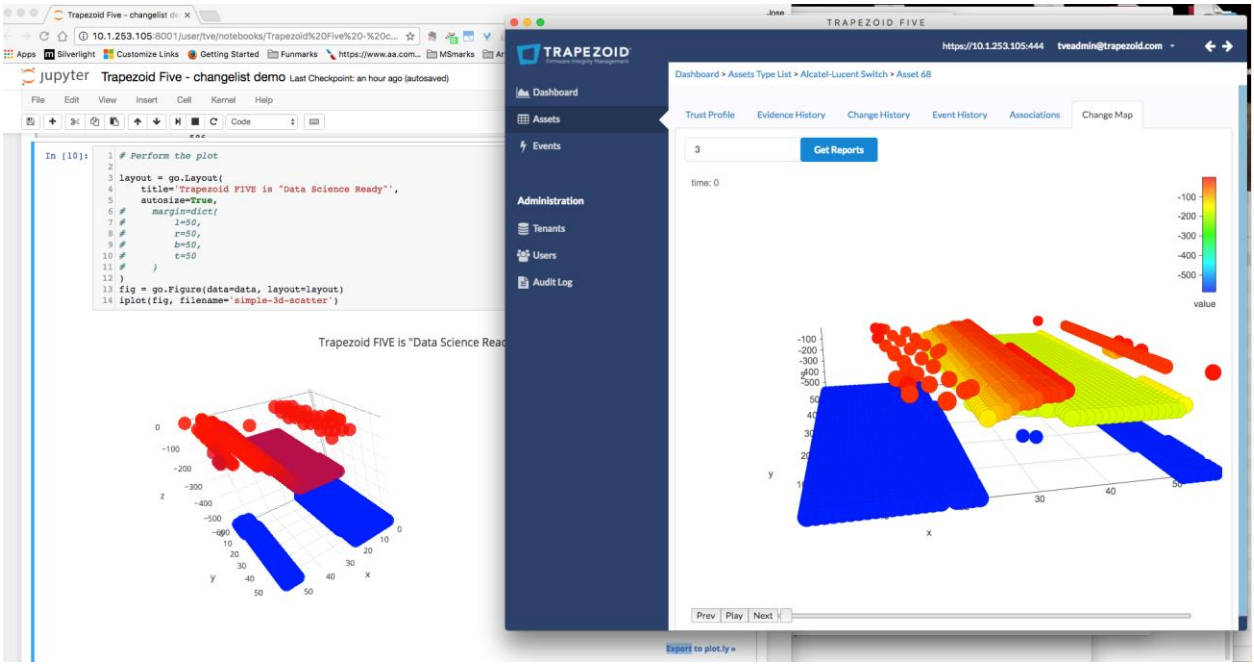
At its core the application is a change detection engine that detects changes in device state. All data and changes run through an Analysis layer to identify anomalies, generate Events, and create Alerts when something should be looked at further.

Security Tool Integration

Events and Alerts can be sent to the organizations existing security tools like SIEMs, Policy Managers, and GRC out of the box via all of the standard protocols. In addition, the Trapezoid FIVE application makes its Application Programming Interface (API) available of direct integration with an organization custom dashboard or portal. Trapezoid uses GraphQL as its API server, which offers many benefits over REST APIs. One of the main benefits is that a client application can dictate exactly what it need from the Trapezoid FIVE server, and receive that data in a predictable way. The reduces the problem of "overfetching" data from the back end server when integrating with a custom front-end application such as a custom dashboard.

Data Science and AI- Friendly Architecture

Trapezoid's data storage model allows an organization's data scientists to run machine learning tools directly against the Trapezoid FIVE database without the need to extract, transform and load (ETL) Trapezoid data into another data store. Not only can the organization run security-focused analytics against the Trapezoid database, but it can also use the aggregation of the operational metrics to develop data-backed, real-world usage models rather than broad estimates. In the figure below you can see the Trapezoid FIVE client application on the right and a Jupyter Notebook application, a popular application among data scientists, examining the same data in the Trapezoid FIVE database.



Patented Trapezoid® Marker Technology

A patented[32] mechanism of creating and continuously validating a unique system identifier incorporating multiple "Seed Values" to remotely attest to the integrity of a device or pool of devices:

The only limit to Seed Values is how sensitive you want to make the Marker. Because Trapezoid FIVE recalculates the value every time it examines a system, the more values that go into creating a Trapezoid Marker, the more values that need to not change so that the Marker recalculates. All of this is done without having to deploy any certificate management or private key infrastructure to validate the Marker. In essence, the Trapezoid Marker is the digital equivalent of a tamper-evident seal.

Use cases for the Trapezoid Marker include:

- **Workload boundary enforcement (Hybrid Cloud)** - Require that applications live only on specific servers, communicate only via specific network gear, and allow access only to specific clients, all with Markers
- **Supply Chain Integrity Validation** - Ensure system meets OEM specifications upon delivery
- **Highly Secure Environments** - Bind whitelist values to Marker for intrinsic, hardware-based attestation
- **Forensic Contextual Marker** - Identifies when and at which step of the process a compromise occurred

Trapezoid Markers are not required to run Trapezoid FIVE but using them increases the overall security of your environment with little operational overhead.

Monitor Your Firmware. Close the Basement Door!

Trapezoid FIVE is a complete firmware lifecycle management system, designed to continuously monitor the integrity of any device. It proactively identifies and analyzes firmware changes across any network. Trapezoid FIVE delivers true firmware visibility by continuously monitoring, detecting and alerting for compromises, then delivering actionable data to security teams via an online dashboard and integration with existing security tools.

Trapezoid FIVE is:

- **Vendor and hardware agnostic** – monitors any infrastructure, network or virtual device.
- **Built for compliance** – meets NIST CSF, FISMA, HIPAA, HITRUST CSF, and GDPR
- **Capable of big data analysis** – lets you run data science tools directly against its database; no ETL required.
- **Supportive of flexible software models** – available as a stand-alone solution or as a managed service from select partners.

[32] US Patent Numbers: 9,258,331, 9,674,138 & 10,305,893

- **Designed for seamless integration** – integrates with existing security monitoring and reporting tools - SIEM, logging tools and more - for a "single pane of glass" monitoring environment.
- **Lightweight and agentless** – avoids invasive code distribution by polling systems on user defined frequency without degrading performance.
- **Deployable anywhere devices live** – across data centers, in the cloud, over the network or on storage, SDx, and IoT devices.

By giving visibility into this layer of the IT stack, Trapezoid helps you "close the basement door."

Contact us today to schedule a demonstration or to discuss a more detailed discussion of this critical threat.

info@trapezoid.com

1-786-621-8580

ABOUT TRAPEZOID - Trapezoid Inc. is a leading provider of Firmware protection and monitoring tools. Founded in 2012, Trapezoid's Firmware Integrity Verification Engine (FIVE) leverages patented technology to proactively identify and analyze integrity changes in firmware and help remediate firmware-related security and compliance breaches related to compromised firmware. Trapezoid's customers include government, education and commercial sectors who use Trapezoid's solutions to validate the integrity of critical hardware and firmware by monitoring for changes across the systems.

Trapezoid, Inc.

4931 SW 75th Ave., Miami FL 33155
1-786-621-8580

www.trapezoid.com