How to Mitigate Firmware Security Risks in Data Centers, and Public and Private Clouds

Published: 3 July 2019 **ID:** G00387620

Analyst(s): Tony Harvey

Firmware vulnerability gives attackers entry into systems that is invisible and persistent with total control of the server, storage or network device. I&O leaders must deliver an infrastructure, whether on-site, outsourced or in the public cloud, that is protected from firmware-based attacks.

Key Challenges

- Firmware-based vulnerabilities are not widely understood, and existing vulnerability scanners and malware identification tools cannot detect firmware-based malware, leaving organizations vulnerable, despite significant security investments.
- I&O teams have tested procedures for OS and application patching, but firmware is rarely included in these policies, even though firmware may contain known vulnerabilities from open-source libraries such as OpenSSL. This leads to inefficient manual upgrades being performed on an ad hoc basis when an issue occurs.
- Features for firmware protection are new and are not enabled by default, even on systems that support them, leaving systems at risk.
- Despite hardware in the cloud or hardware managed by third-party vendors not being under the control of the I&O team, it is still vulnerable, and the I&O team is still responsible for the delivery of a secure infrastructure.

Recommendations

I&O leaders responsible for data center infrastructure should:

- Partner with the chief information security officer to develop the skills in the team to understand firmware and hardware threats by investing in emerging firmware threat detection and scanning tools and engaging with industry consortia and specialists in this area.
- Integrate a firmware upgrade policy into standard data center procedures, so that updates are made on a regular basis and emergency planning is in place.

- Secure access to firmware updates by implementing network isolation and user access controls, logging, and using the vendor's secure firmware features.
- Ensure that both cloud and on-premises vendors have secure firmware update programs by working with the vendor management team.

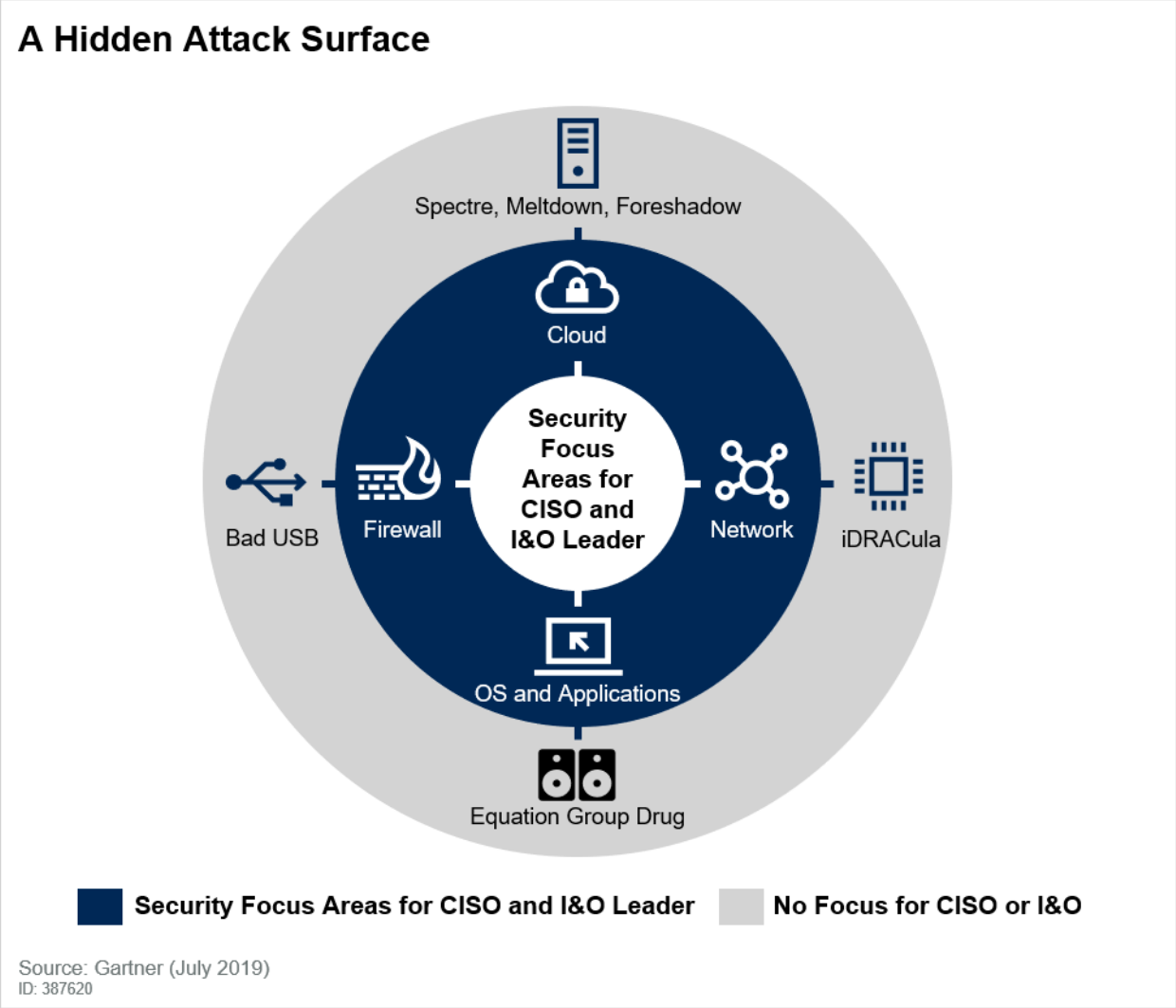
Strategic Planning Assumption

By 2022, 70% of organizations that do not have a firmware upgrade plan in place will be breached due to a firmware vulnerability.

Introduction

Firmware-based malware has largely been ignored by both the security and I&O teams (see Figure 1), making firmware in servers, storage systems and networking devices the "soft underbelly" of enterprise security. In general, neither the I&O or security teams will be fully aware of the amount of code embedded as firmware and of just how many components in their infrastructure contain firmware (see Note 1). Once installed, firmware-level attacks are invisible to most traditional security controls, able to persist indefinitely (even through a system reimage) and can even disable systems. To date, there have been both ransomware[1] and rootkits[2] installed through firmware-based vulnerabilities; and, as the techniques for infecting firmware become more widespread, the issue is likely to grow.

Figure 1. A Hidden Attack Surface



Such threats are typically regarded as a CISO problem; however, once a potential threat or vulnerability is identified, the I&O team will be tasked with patching or upgrading systems and maintaining them to ensure that they are secure.

Hypervisor, OS and application patching is well-understood. I&O leaders will have procedures and tools in place to ensure that, when an update is required due to a critical vulnerability, it can be rolled out to systems that require in it a controlled fashion. No such tools and processes currently exist for firmware in most enterprises. I&O leaders who do not take a proactive stance on developing these processes and tools to deal with hardware and firmware security threats will inevitably be operating in panic mode when new threats requiring a firmware-based response, such as Spectre and Meltdown, appear. (See “Microprocessor Security Vulnerabilities Require Measured Mitigation Response.”)

Given the ever-increasing number and type of firmware threats, how can I&O leaders become more proactive and develop a strategy to minimize the risks associated with firmware-based attacks? This research will describe the key elements needed to analyze the threats, secure the existing infrastructure and develop a risk-based response strategy with the CISO for mitigating firmware security incidents.

Analysis

Develop the Skills to Understand Firmware Threats

I&O leaders can use external specialists to address temporary shortfalls in internal skills, but they should be working with the CISO to ensure that internal staff are building the necessary skill sets to protect the infrastructure against firmware threats.

The I&O leader should:

- Ensure that the I&O team understands key concepts for platform security, such as root of trust, measurement and attestation. Without an understanding of these concepts, any attempt to enable security features that protect firmware, such as secure boot or measured boot, is likely to fail.
- Use scripting and automation tools to scale delivery of firmware updates in a rapid and repeatable manner. (See "Choose the Right I&O Automation Tool Categories to Maximize Success" and "How to Automate Server Provisioning and Configuration Management.")
- Work with the security team to evaluate and assess firmware vulnerabilities and threats by identifying which systems are most at risk of exploitation and patching or updating appropriately.
- Invest in tools that can identify known security issues in firmware systems and identify and track when firmware is changed.

In addition, I&O leaders should be working with the CISO to deliver a risk-based response strategy for firmware-based vulnerabilities (see "Implement a Risk-Based Approach to Vulnerability Management"). The I&O leader must be involved in the creation of this strategy as, frequently, there are competing deliverables between the I&O and security teams. A firmware update may prevent applications from running, cause performance impacts or lead an OS vendor to refuse to support a system, creating a clear conflict between the priorities of the I&O team and security team. Without I&O involvement, these issues will not be surfaced and dealt with.

This strategy should include at a minimum:

- A risk analysis of the actual vulnerability. This must be a joint exercise between the I&O team and the security team to define how vulnerable systems are to the attack. Spectre and Meltdown, for example, required that an attacker be able to run arbitrary code on the system. For servers in a data center, if an attacker can run arbitrary code, then the system is already compromised.

- An impact analysis of the fix to ensure that applications, performance and support are not affected
- A risk-based order of priority in which systems must be updated. Internet-facing systems are clearly more at risk to any attack than systems that can only be accessed from internal networks. Systems that contain personally identifiable information that is subject to regulation (e.g., Health Insurance Portability and Accountability Act [HIPAA] or General Data Protection Regulation [GDPR]) are higher priority than test and development systems.
- Identification of compensating controls that could be used to mitigate the risk if the fix cannot be installed immediately.

Integrate a Firmware Update Policy Into Standard Procedures

OS patching, while still painful, is a known process for the I&O team. When a major vulnerability is published, I&O has a plan and process for rolling out patches. In 2018, the publication of the Spectre and Meltdown vulnerabilities, which could only be mitigated by firmware patches, exposed the sorry state of firmware updating in the modern enterprise. Calls to Gartner analysts revealed that most server firmware was not being updated on a regular basis. I&O leaders were having to send engineers into the data center to use USB keys to deliver firmware updates, rather using scriptable and automatable remote update tools that are available from most server vendors.[3]

To guarantee that I&O is ready when the next major firmware security vulnerability is published, I&O leaders must ensure that the I&O team regards updating firmware as routine as doing OS updates. The I&O leader must have the team create a firmware upgrade process that will:

- Update all the firmware on all systems on a regular cadence. Typically, three to six months for noncritical updates.
- Define an emergency update process for critical firmware vulnerabilities in critical systems.
- Identify, document and use scriptable and automatable methods for firmware updates wherever possible.
- Document firmware update methods for systems where manual intervention is required.
- Identify systems for which firmware updates are no longer being released, flag them as known risks, and apply compensating controls to restrict access and reduce the risk of infection.
- Collect firmware versions through infrastructure monitoring tools, store them in IT asset management (ITAM) databases, and automate comparison to vendor releases to identify out-of-date versions

Secure Access to Firmware Updates

Virtually all servers available today come with a baseboard management controller (BMC) of some kind. This “computer within a computer” is powered independently of the host server and can monitor and manage everything about the host server. This includes being able to update the

firmware, install an operating system and reboot the host. These functions are all available over the network to enable remote management.

Although it is convenient for the I&O team to deliver firmware updates in an automated and scriptable way, these capabilities represent a severe security risk. A remotely accessible system that can completely control a server, even while powered off, is a clear attack point for any hacker. Unfortunately, in many cases, this area is overlooked, and there are known issues with default passwords and remotely exploitable[4] vulnerabilities. Ransomware attacks through the BMC are now in the wild.[1]

I&O leaders must ensure that these remote management features are secured by:

- Updating BMC firmware to the latest release to ensure that known issues are not present. This is typically nondisruptive to server operation.
- Using a dedicated management network for all BMC traffic and isolating that network behind a firewall, so that it cannot be accessed directly from the internet. (For example, a recent check by the author listed more than 28,000 BMC devices that were directly accessible from the internet.)
- Requiring a VPN for external access to the BMC interface or preventing any external access completely for more secure systems.
- Disabling services such as Intelligent Platform Management Interface (IPMI), SNMP or Desktop Management Interface (DMI), unless they are specifically required.
- Logging access to the BMC interface.
- Including BMC passwords in a privileged access management solution (see “Best Practices for Privileged Access Management Through the Four Pillars of PAM”).

Enable Firmware Security Features

Modern systems include several features that provide improved firmware security. These features include:

- **Hardware-based root of trust:** A secure hardware device on the board (commonly referred to as a trusted platform module [TPM]) that can identify and authenticate the system to ensure that the firmware components or signed firmware updates are valid. It also provides secure storage for cryptographic keys and measurement checksums.
- **Secure boot:** Part of the Unified Extensible Firmware Interface (UEFI) specification, secure boot is designed to protect a system against malicious code being loaded and executed early in the boot process, before the operating system has been loaded. This will prevent a rootkit from being installed and compromising the security of the OS.
- **Measured boot:** Creates checksums for the system and components, which are then securely stored. These checksums can then be used to attest whether the system has changed since

last boot. This can be used in conjunction with secure boot to protect the system against changes in firmware not protected by the secure boot process.

- **Signed firmware updates:** This feature allows only approved firmware cryptographically signed by the vendor to be installed. This reduces the risk of a malware infection being delivered through the firmware.

Enabling these features on any system that supports them greatly reduces the risk of firmware-based malware being installed.

Include Secure Firmware Requirements in RFPs

With firmware-based attacks now in the wild, I&O leaders must drive their traditional on-premises hardware vendors to provide secure systems and easily scriptable and automatable firmware update processes. Vendor documentation must be provided that includes:

- A list of all components with updatable firmware
- A list of open-source libraries used; many vendors use existing libraries such OpenSSL in their firmware and will inherit any security vulnerabilities in these libraries
- The process for updating all firmware
- Where to obtain updates
- A downloadable database of all firmware checksums
- A management API to check the firmware checksum
- Any other relevant info (e.g., are updates available only if you have a current maintenance contract)

In addition, with more critical business system being placed on cloud services, I&O leaders must use their expertise to ensure that the underlying cloud infrastructure services are secured at the hardware and firmware layers. Hardware- and firmware-based security questions must be included in all future RFPs and the responses used as part of the decision process.

Traditional Vendor Supply Contracts

The I&O leader should work with the CISO and vendor management team to create a full list of questions that should cover the following key areas:

- How does the vendor ensure security at the firmware and hardware level?
- Does the vendor conform with relevant guidelines, such as NIST SP 800-193 or the European Union (EU) “ENISA Hardware Threat Landscape and Good Practice Guide”?
- How does the vendor deliver and secure firmware updates, and is the process scriptable and automatable?

- Does the vendor provide a mechanism to check the installed firmware against a database of known good firmware, and is the process automatable?

Cloud and Third-Party-Managed Systems

When using cloud systems or managed service offerings where the vendor is responsible for maintaining the systems, the underlying hardware is no longer under the control of the I&O team; but it is still vulnerable to firmware-based attacks.[5] However, the I&O and security teams will still be held responsible for the security of the infrastructure.

When contracting with a cloud vendor or using hardware as a managed service, I&O leaders must ensure that the vendor has clearly defined procedures and processes for dealing with firmware security vulnerabilities. The areas to cover differ slightly from those of a traditional vendor, but should include:

- How does the vendor ensure that firmware on a system is valid and has not been changed?
- Can the vendor provide details on how firmware updates are managed, and who is responsible for updating the firmware and any impacts to production systems?
- How does the cloud vendor ensure that hardware that can be transferred between end users is reset to a known-good firmware basis prior to the transfer?

The largest cloud vendors have provided information on how they deliver secure firmware as follows:

- Amazon Web Services (AWS) disables firmware updates via a custom chip called Nitro, which is accessible only to AWS. The Nitro chip performs the firmware updates and provides a root of trust by checking the firmware checksums prior to each boot and ensuring that no changes have been made.[5]
- Google Cloud Platform (GCP) uses a chip known as Titan to provide a secure root of trust and prevent malicious firmware from being deployed by checking firmware checksums prior to boot.[6]
- Microsoft Azure uses the Cerberus chip to provide a secure root of trust and to prevent malicious firmware from being deployed by checking firmware checksums prior to boot. The chip and the associated code have been released to the Open Compute Project.[7]

Gartner Recommended Reading

Some documents may not be available as part of your current Gartner subscription.

“Market Guide for Cloud Workload Protection Platforms”

“Seven Imperatives to Adopt a CARTA Strategic Approach”

“Reimagining Security and IT Resilience for a Cloud-Native DevSecOps World”

“Mitigating Spectre and Meltdown in the Enterprise”

“Microprocessor Security Vulnerabilities Require Measured Mitigation Response”

“What Networking Leaders Need to Know and Do About Spectre/Meltdown”

Evidence

The analysis and advice provided in this research are built from the aggregation of analyst experience and ongoing interactions with end users and technology and service providers.

[1] “Hacking iLO — Take a Moment to Secure Your Servers,” Avast Blog.

[2] “Russia’s Elite Hackers Have a Clever New Trick That’s Very Hard to Fix.” Wired.

[3] Inquiry call volume in 2018 on Spectre/Meltdown and firmware updates.

[4] Partial list of BMC-based Common Vulnerabilities and Exposures (CVE) scored more than 8 on the CVE scale can be found at CVE Details.

[5] “AWS Nitro System,” Perspectives.

[6] “Titan in Depth: Security in Plaintext,” Google Cloud Blog.

[7] “Microsoft Creates Industry Standards for Data Center Hardware Storage and Security,” Microsoft Azure Blog.

[8] “The Missing Security Primer for Bare Metal Cloud Services,” Eclypsium.

Note 1 How Much Code Is Embedded as Firmware?

Although many I&O leaders are aware that firmware is code embedded in the systems that they are responsible for, it comes as a surprise to many just how complex this code is. For example:

- The UEFI-based basic input/output system (BIOS) can include a full network stack to enable remote updates and remote boot.
- The BMC, such as the Dell Remote Access Card (DRAC) or HP Integrated Lights Out (iLO), runs an embedded Linux OS.
- The Intel Management Engine (ME), embedded in every Intel-based server, runs the MINIX OS.

Network interface cards (NICs), graphics processing units (GPUs), host bus adapters (HBAs), redundant array of independent disks (RAID) controllers and even power supplies all have embedded CPUs that run an OS. This represents millions of lines of code that are prey to the same security flaws prevalent in all software (e.g., buffer overruns, default passwords, password

backdoors, etc.). Much firmware also contains open-source libraries, such as OpenSSL, Apache Tomcat or the Apache HTTP Server (HTTPD). Therefore, the firmware will inherit the software vulnerabilities in these libraries, unless they and the associated firmware, are updated.

Note 2 Firmware-Based Security Tool Vendors

- Eclypsium[8]
- CrowdStrike (endpoint and client only currently)

GARTNER HEADQUARTERS

Corporate Headquarters
56 Top Gallant Road
Stamford, CT 06902-7700
USA
+1 203 964 0096

Regional Headquarters
AUSTRALIA
BRAZIL
JAPAN
UNITED KINGDOM

For a complete list of worldwide locations,
visit http://www.gartner.com/technology/about.jsp

© 2019 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates. This publication may not be reproduced or distributed in any form without Gartner's prior written permission. It consists of the opinions of Gartner's research organization, which should not be construed as statements of fact. While the information contained in this publication has been obtained from sources believed to be reliable, Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner research may address legal and financial issues, Gartner does not provide legal or investment advice and its research should not be construed or used as such. Your access and use of this publication are governed by Gartner Usage Policy. Gartner prides itself on its reputation for independence and objectivity. Its research is produced independently by its research organization without input or influence from any third party. For further information, see "Guiding Principles on Independence and Objectivity."