SOLUTIONS_

ZERO TRUST FOR DEVICES

Extending Zero Trust to Physical Devices and Supply Chains in Federal Agencies.

WHO SHOULD READ THIS_

Federal security leaders, including authorizing officials (AOs), CISOs and CIOs; security and network architects; teams responsible for data center security, infrastructure security or network security who are planning and executing Zero Trust projects.

WHAT THEY WILL LEARN_

How and why Zero Trust principles apply to devices and their supply chains, and best practices for extendiing these principles down to physical hardware including chips, processors, and system components.

FURTHER READING_

- TAG Cyber white paper, “Making the Case for Firmware in the Context of Zero Trust Security”
- Eclypsium white paper on Executive Order 14028
- Teledyne Lecroy & Eclypsium joint white paper, “Applying Zero Trust in the Supply Chain to Prevent DMA Attacks”

Recent mandates and regulations have declared the necessity for federal networks to establish and maintain Zero Trust security postures. These include but are not limited to:

- **Executive Order (EO) 14028**, which called out the need for federal networks to "adopt security best practices" and "advance toward Zero Trust architectures".
- **M-22-18** Memorandum For the Heads of Executive Departments and Agencies, which aligned Zero Trust goals with FISMA mandates and set forth timelines for completion.
- **DoD Zero Trust Capability Execution Roadmap (COA 1)**, which defines milestones and timelines for implementing Zero Trust tactics in Department of Defense networks.
- **NIST SP 1800-35 (Draft)** Implementing a Zero Trust Architecture (Preliminary Draft) which will lay out a comprehensive strategy to balance access and productivity with Zero Trust concepts like least privilege, default deny, and risk context.

But how will these guides and requirements impact our information and communication technology (ICT) devices? How will they stand up to the multitude of embedded components that exist within these devices? This solution brief will answer those questions and give strategists and practitioners guidance in achieving Zero Trust through hardware.

UNDERSTANDING ZERO TRUST_

The O'Reilly textbook, *Zero Trust Networks: Building Secure Systems in Untrusted Networks* by Evan Gilman and Doug Barth, sets out five basic principles at work in a Zero Trust network design:

1. The network is always assumed to be hostile.
2. External and internal threats exist on the network at all times.
3. Network locality is not sufficient for deciding trust in a network.
4. **Every device, user, and network flow is authenticated and authorized.**
5. **Policies must be dynamic and calculated from as many sources of data as possible.**

This brief focuses on Principles 4 and 5 from the Wiley book, where the Zero Trust "rubber" hits the real-world "road." It's here we start thinking about the natural extension of Zero Trust principles down into the devices and hardware they use, while also calculating the unique, highly dynamic risks not only of our devices, but of the millions of internal components that make them whole.

ZERO TRUST FOR EVERY DEVICE_

Agency networks consist of two major entities: people and devices. "People" are generally covered by identity and access management (IAM) programs. Devices come in a wide array of forms and uses: endpoints, servers, networked and connected devices, security devices, and IoT/OT devices. There should be little argument that each device needs to be uniquely authenticated, either through user interaction, embedded X.509 certificates, SSH keys, or other methods. Inherited trust or legacy permissions aren't allowed in a Zero Trust network.

As the Riley books says in Principle 4, "Every device, user, and network flow is authenticated and authorized."

> **"Every device, user, and network flow is authenticated and authorized"**

However, Zero Trust requires more than simply checking the identity of devices. Agencies also need to know that their assets are free from threats and vulnerabilities down to the underlying code, hardware, and components. For example, a device could pass identity checks by having the appropriate certificate yet be compromised by a backdoor or UEFI/BIOS implant.

In order to verify the integrity of physical devices, agencies will need new types of visibility that understand the inner workings of devices and the supply chains that produce them. We need to recognize that each asset is really an amalgamation of components and code, often from

dozens of disparate supply chain suppliers including motherboards, central processing units (CPUs), memory, PCI cards, solid state drives, system management modules (SMM), baseboard management controllers (BMCs) in servers. Each component is a potential target, and each supplier is a potential point of compromise.

This potential risk has rapidly turned into a documented reality as threat actors have aggressively pivoted to exploiting technology supply chains. While agency networks are often well-defended, attackers can target dozens of supply chain suppliers to compromise devices before they are ever delivered, or can similarly steal keys and source code in order to deliver low-level implants within product updates. Within the past several months, threat actors have compromised major supply chain vendors affecting chipsets, motherboards, laptops, and server baseboard management controllers. Thus, in the same that agencies must assume their networks are hostile, they must also assume their devices and supply chains are hostile.

To address this risk, agencies must actively validate all their critical devices and the components within those devices. How do we verify device components? Some use signed certificates and keys, but for the majority, we can verify them through the firmware and microcode embedded in them by their manufacturer.

According to research from analyst firm Gartner:

- Every endpoint is delivered, on average, with 15-20 firmware components
- Every server is delivered with around 30 components, and sometimes more than 50
- Every network device is now shipped with dedicated firmware

This firmware represents potential risk, but it also helps us uniquely establish – through version numbers and hash comparisons – the "integrity" of the devices in which it's contained.

EXTENDING ZERO TRUST TO THE SUPPLY CHAIN AND FIRMWARE_

Firmware is simply software code that's been embedded directly onto hardware components by that hardware's manufacturer, or one of their suppliers. It doesn't reside in common storage locations for files and data but in specialized chips.

Firmware is increasingly used as an initial attack vector. This table lists common hardware or firmware-based vulnerabilities and the recent exploits that leverage them.

Component	Role	Vuln?	Exploited?
Central Processing Unit (CPU)	Often called microcode, CPU-level firmware is powerful and privileged. Microcode firmware typically resides in special high-speed memory and translates machine instructions, state machine data, or other input into sequences of detailed circuit-level operations.	Yes	Dirty Cow Spectre Meltdown
Unified Extensible Firmware Interface (UEFI)	A specification that defines a software interface between an operating system and platform firmware. UEFI replaces the legacy Basic Input/Output System (BIOS) boot firmware.	Yes	Moon Bounce Cosmic Strand TrickBoot
Trusted Platform Module (TPM)	An international standard for a dedicated microcontroller designed to secure hardware through integrated cryptographic keys. The term can also refer to a chip conforming to the standard.	Yes	Various probing, side- channel, interposer attacks
Management Engines (ME)	Intel's autonomous subsystem that has been incorporated in virtually all of Intel's processor chipsets since 2008.	Unk	Conti is focused here
Baseboard Management Controller (BMC)	Provides the intelligence in an Intelligent Platform Management Interface (IPMI). It is a specialized microcontroller embedded on the motherboard of a server computer. The BMC manages the interface between system- management software and platform hardware through dedicated firmware and RAM.	Yes	iLOBleed USBAnywhere
Network Card (NIC)	A network interface controller (NIC, also known as a network interface card, network adapter, LAN adapter or physical network interface, and by similar terms is a computer hardware component that connects a computer to a computer network.		EtherLED NetSpectre
Direct Memory Access (DMA)	A feature of computers that allows certain hardware subsystems to access main system memory independently of the central processing unit (CPU) and run commands through on-board RAM.	Yes	Various DMA and side channel attacks
Embedded Controller	A microcontroller in computers that handles various system tasks. Usually merged with Super I/O, especially on mobile platforms.	Unk	Multitude • Firmware-based • Network-based • Side-based
System Management Module (SMM)	Chips that enable System Management Mode, which when active provides an alternate firmware-based software system with higher privileges.	Yes	HPE devices AMD chips
Solid State Drive (SSD)	A solid-state storage device that uses integrated circuits to store data persistently, typically using flash memory, and functioning as secondary storage.	Soon	SSD attacks Micron Flex

In addition to these common components, most computers have additional chips for video processing, sound processing, digital signal processing, and other application-specific integrated circuits (ASICs). Each of these components represents a source for compromise or vulnerability. Each must be included in the universe of data we use in the decision-making processes of Zero Trust systems. By assessing the attributes of this low-level code – its version, source date, binary signature and provenance – practitioners can build trust in these underlying (and invisible) components.

How do we make those critical Zero Trust decisions? We assess asset risk down to the chip level.

ASSESSING CHIP-LEVEL RISK_

The 5th and final point from the Zero Trust Networks book cited earlier is both specific and almost impossibly broad: "Policies must be dynamic and calculated from as many sources of data as possible."

> **"Policies must be dynamic and calculated from as many sources of data as possible"**

"Policy" refers to the output from a trust engine. A trust engine, in turn, calculates risk based on system inputs. Given the data on hand, do we trust this component or not? Can we allow this device or this user on the network, or not?

We can generate a Zero Trust test case using an example from the "BMC" row in the table above:

- A subject system on the network is an HPE Gen9 server using iLO4, HP's "integrated lights out BMC" module.
- A process (or person) wants to store critical or sensitive data on this server.
- But as this post explains, this BMC module has been actively exploited in ransomware attacks known as iLOBLeed, and it is difficult to tell whether an implant is at work on this HPE server without doing a firmware-level scan.
- Do we trust it? Or do we explicitly distrust it?

To answer this question, we need to reliably verify the code within the BMC itself. However, this can be easier said than done. Not only does it require firmware and BMC expertise, the iLOBleed threat is designed to prevent patching of the firmware, yet will report false information in order to appear that the device has been updated. So in reality the device is vulnerable and compromised, yet traditional security and vulnerability scans see it as safe. To close this gap, agencies need independent, purpose-built visibility into low-level code, firmware, and components within their assets.

HOW ECLYPSIUM HELPS DOD AND CIVILIAN TEAMS

At its core, Zero Trust is about rooting out areas where trust is assumed, based on a perception of risk, and replacing that assumption with active verification of a trust state. For many federal agencies and organizations, the firmware, hardware, and supply chain code within devices has been a persistent blind spot that has been trusted "by default". This is not only a bad security strategy, but it's also now contrary to the federal mandates requiring Zero Trust.

Instead, practitioners should implement a program to actively and continually assess all laptops, servers, and networking infrastructure when joining the network to validate Zero Trust posture and integrity. Assets should be assessed before being allowed on the network or to access sensitive resources, and continuously monitored to detect any changes to the integrity or posture of the asset over time.

For civilian cybersecurity teams: this approach not only meets Executive Order (EO) 14028's requirements but also extends Zero Trust programs all the way down to the base hardware in their endpoints, servers, and critical network equipment.

For DoD teams: Eclypsium specifically addresses the new requirements published on November 15, 2022, in DoD Zero Trust Capability Execution Roadmap (COA 1). These include firmware-level controls for:

Devices

- 2.1 Device Inventory
- 2.2 Device Detection and Compliance
- 2.5 Partially & Fully Automated Asset, Vulnerability, and Patch Management

Software

- 3.3 Software Risk Management

Automation and Orchestration

- 6.7 Security Operation Center and Incident Response

Visibility and Analytics

- *7.2 Security Information and Event Management*
- *7.4 User and Entity Behavior Analytics*

DoD Zero Trust Capabilities

User

1.1 User Inventory
1.2 Conditional User Access
1.3 MultiMulti-Factor Authentication
1.4 Privileged Access Management
1.5 Identity Federation & User Credentialing
1.6 Behavioral, Contextual ID, and Biometrics
1.7 Least Privileged Access
1.8 Continuous Authentication
1.9 Integrated ICAM Platform

Device

2.1 Device Inventory
2.2 Device Detection and Compliance
2.3 Device Authorization with Real Time Inspection
2.4 Remote Access
2.5 Partially & Fully Automated Asset, Vulnerability and Patch Management
2.6 Unified Endpoint Management (UEM) & Mobile Device Management (MDM)
2.7 Endpoint & Extended Detection & Response (EDR & XDR)

Application & Workload

3.1 Application Inventory
3.2 Secure Software Development & Integration
3.3 Software Risk Management
3.4 Resource Authorization & Integration
3.5 Continuous Monitoring and Ongoing Authorizations

Data

4.1 Data Catalog Risk Assessment
4.3 Data Labeling and Tagging
4.2 DoD Enterprise Data Governance
4.5 Data Encryption & Rights Management
4.4 Data Monitoring and Sensing
4.6 Data Loss Prevention (DLP)
4.7 Data Access Control

Network & Environment

5.1 Data Flow Mapping
5.3 Macro Segmentation
5.4 Micro Segmentation
5.2 Software Defined Networking (SDN)

Automation & Orchestration

6.1 Policy Decision Point (PDP) & Policy Orchestration
6.2 Critical Process Automation
6.3 Machine Learning
6.4 Artificial Intelligence
6.5 Security Orchestration, Automation & Response (SOAR)
6.6 API Standardization
6.7 Security Operations Center (SOC) & Incident Response (IR)

Visibility & Analytics

7.1 Log All Traffic (Network, Data, Apps, Users)
7.2 Security Information and Event Management (SIEM)
7.3 Common Security and Risk Analytics
7.4 User and Entity Behavior Analytics
7.5 Threat Intelligence Integration
7.6 Automated Dynamic Policies

Eclypsium addresses hardware- and firmware-level risks in seven core areas called out by DoD that have specific deadlines and milestones for Zero Trust practices

To learn more about Eclypsium's end-to-end approach to achieving a Zero Trust security posture for enterprise devices, visit the Eclypsium website or schedule a demo through email.

info@eclypsium.com
eclypsium.com