presented by

Best Practices for Secure Firmware Patching

UEFI 2020 Virtual Plugfest

August 19, 2020

Presented by Alex Bazhaniuk, Eclypsium & Tim Lewis, Insyde Software

Meet the Presenters

Alex Bazhaniuk
Co-Founder and CTO
Member Company:
Eclypsium

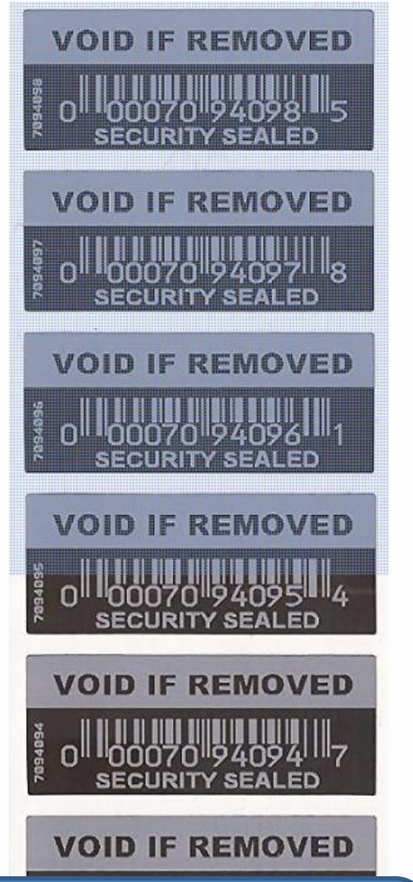
Tim Lewis
CTO
Member Company:
Insyde Software

The Threat Is Real

- Firmware holds a unique, valuable security position
  - Computer systems are only as secure as their firmware
  - Value to a hacker is not access and control to the system's hardware, but the system's data
- Firmware is under increasing numbers of attacks
  - Not just from researchers and hackers, but from professionals
  - No "if" but "when" a security vulnerability is found in code
- Firmware threats often appear years after 1st shipment
  - Support for shipping platforms will extend longer than ever before

Supply Chain to Server Room to Office…

- “Bad actors compromise hardware by inserting physical implants into a product component or by modifying firmware. Often these manipulations create a “back door” connection between the device and external computers that the attacker controls. Once the device reaches its final destination, adversaries use the back door to gain further access or exfiltrate data.”*

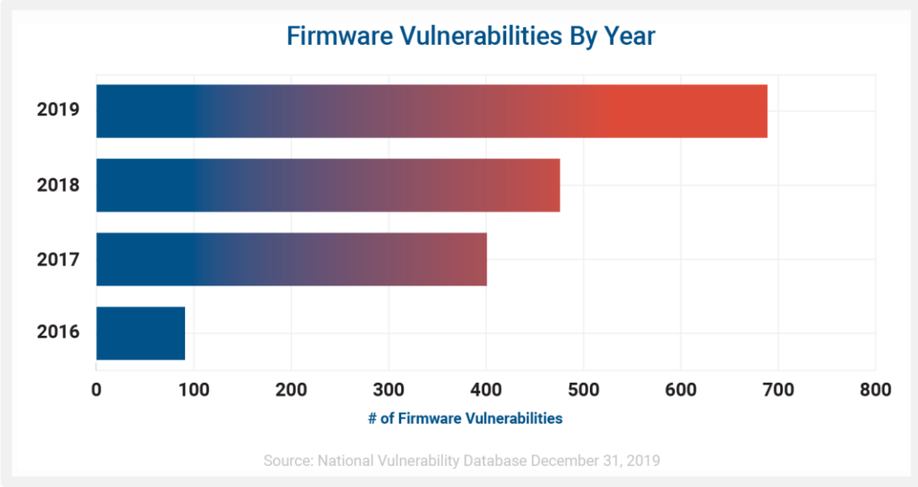


Modifying firmware has predictable deployment time, requires no soldering, and can be deployed against multiple targets.

https://www.microsoft.com/security/blog/2020/02/03/guarding-against-supply-chain-attacks-part-2-hardware-risks/

Firmware Attacks Are High Impact

- The Highest Levels of Privilege
- Bypass of Traditional Security
- Persistence
- Stealth
- Damage



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

Source: Gartner Research

FBI warns that high-impact ransomware attacks threaten US businesses, organizations. Advises patching operating system, software, and firmware on devices as part of cyber defense best practices.

Source: FBI Alert I-100219-PSA

Analysis of ransomware distribution methods **implicated compromised firmware as the 3rd most common infection vector** in 1H 2019, accounting for 12% of attacks disrupting companies, public entities and other organizations.

Source: F-Secure Attack Landscape H1 2019

Firmware Attack Delivery

- Firmware-as-vehicle attacks modify existing firmware to create a vulnerability
  - Vulnerability added after development engineer finished
  - Need to give assurance that the current firmware is what was delivered from the OEM

- Firmware-as-agent uses existing firmware weaknesses to create vulnerabilities
  - Vulnerability present in firmware as shipped
  - Need to give assurance that the current firmware isn't being used for an attack

How To Know An Attack Has Landed

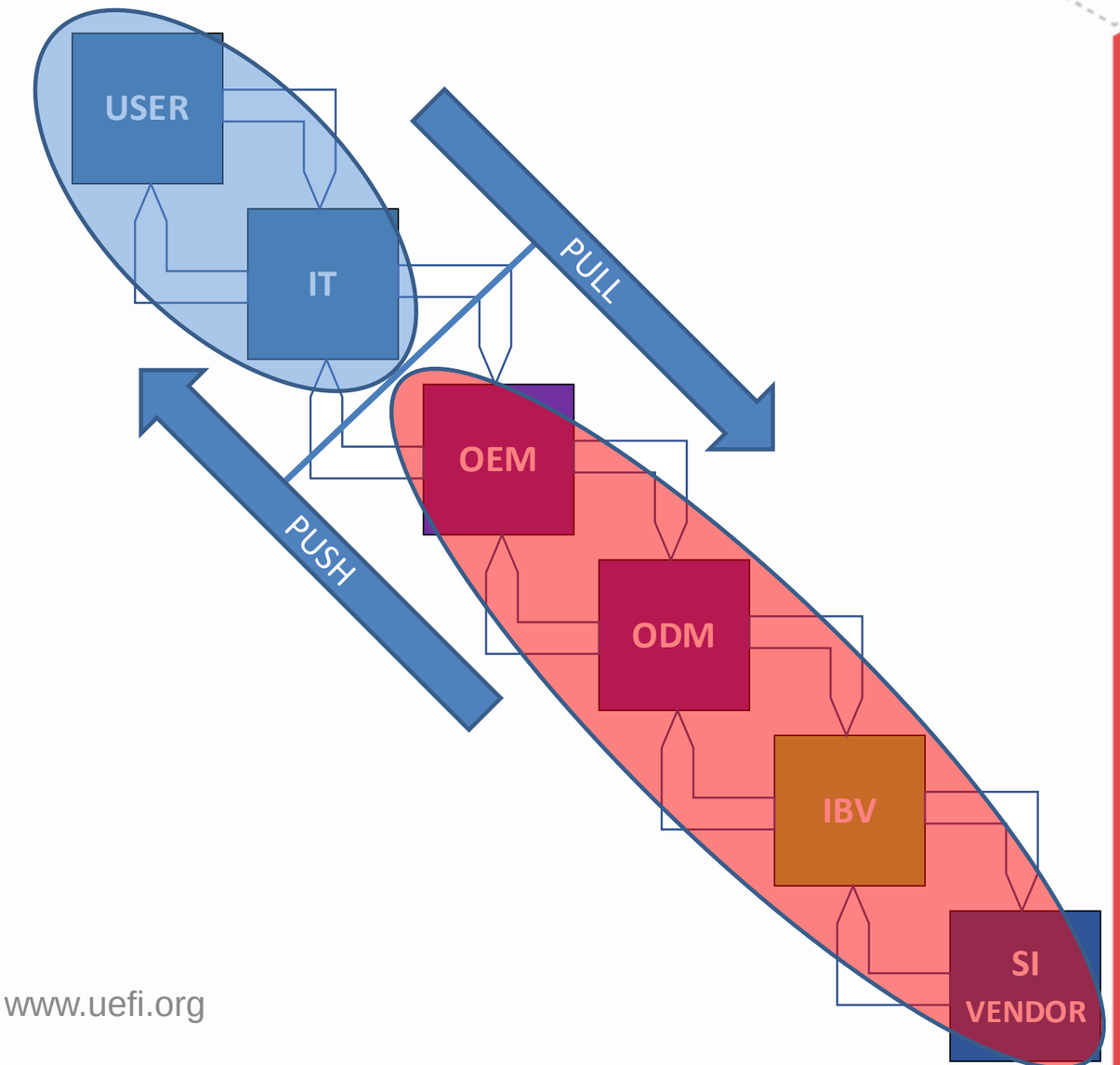
- From cybersecurity employee awareness training material:
  - “Please call the Help desk immediately if you have reason to believe your computer has been infected with a virus.”
    - Reacts slower than usual
    - Stops running for no apparent reason
    - Fails to boot
    - Seems to be missing important files
    - Prevents you from saving your work

Can We Do Better?

Complex Ecosystem & Updates

- How to simplify pulling and pushing updates based on discovered issues?

- Biggest problem is crossing the IT/User and OEM barrier
  - How does an IT/User audit their own firmware security status?
  - How does an OEM reliably deliver firmware updates?



How Does The User Know They Need An Update?

- UEFI BIOS virus scanners
- TPM/TCM – Known execution patterns
- UEFI's ESRT – Reports individual firmware components and their version
- Eclypsium Platform – Enterprise-wide firmware inventory/monitoring

CHIPSEC

- An open-source framework for analyzing the security of PC platforms including hardware, system firmware (BIOS/UEFI), and platform components
  - Verifies hardware register settings
  - Calculates “fingerprint” for firmware volumes and hashes for firmware drivers
  - Runs under Windows, Linux, macOS or UEFI Shell

- Generates detailed report but needs interpretation to know what is pass and what is fail
- Must run with secure boot “off”
- Limited chipset support
- Insyde’s InsydeSST leverages CHIPSEC to prompt end-users for downloads

https://github.com/chipsec/chipsec

What More Could We Do?

- Identify other sections that are immutable (code or data)
- Audit while the system is not running
- Audit firmware configuration settings (PCDs, UEFI variables)
- Finer granularity updates (like individual drivers and FVs)
- Guarantee access to measure other firmware on the platform

Enterprise Challenges

Lack of Firmware and Hardware Inventory

Difficult Update Processes

Fears of Potential Negative Effects

Device Downtime

And now … patching in the new remote work environment

Enterprise Best Practices for Firmware Updates

- Treat firmware updates with the same discipline that you apply to software updates
- Get visibility into the hardware and firmware that are in your fleet
- Invest in the tooling and skills you need to manage testing, rollout and rollback
- Make firmware support a priority in your hardware purchasing decisions

Industry Challenges

Fractured Firmware Ecosystem

Firmware problems with Long Time Horizons

Lack of Basic Firmware Security

Compromises in Technology Supply Chain

Fractured Firmware Ecosystem

- Different OEM/ODM
- Different firmware vendors
- Different environment: UEFI, coreboot, etc.
- **Different Threat Models!!!!!**

Example: Summary of ThinkPad X1 Carbon 6th

- 8 components have firmware into persistent storage of the device (UEFI, ME, EC, SSD, Thunderbolt, USB-C Dock, Synaptics Touchpad & TrackPoint). Firmware of all components outdated
- 5 runtime firmware blobs correspond to 3 peripheral devices (Graphic, Wifi, Bluetooth)
- 1 external peripheral device (Docking Station) has firmware supported by vendor (potentially capable for DMA)
- System firmware has 510 executables
- System firmware configuration has 143 runtime variables
- Device has 3rd party components: Synaptics, Acer, Samsung, Realtek, etc.
- System Firmware has 3rd party components: Computrace from Absolute Software

Fractured Firmware Ecosystem

- Provide visibility and inventory methods to cover:
  - BOM, including Vendor, Model, Version of each component
  - Details for all components of the device (including interface used to communicate to it)
  - Details on what components have firmware and interface for communicate/update it
  - Details on how to read/verify firmware

Firmware Problems with Long Time Horizons

- Bugs have different severity, priority, exploitability
- Some bugs hard to fix (TLBleed, MDS, etc.)
- For some bugs hard to deliver update (BootHole, Meltdown, etc.)
- Some bugs hard to reproduce, identify (HPE SAS Solid State Drives Failure at 32,768 Hours of Operation)

Firmware Problems with Long Time Horizons

- Build modern CI/CD pipeline for firmware development:
  - Run automated build, collect artifacts
  - Use emulator for non hardware specific functionality for testing
  - Run automated tests:
    - Unit Testing
    - Integration Testing
    - Regression Testing

Lack of Basic Firmware Security

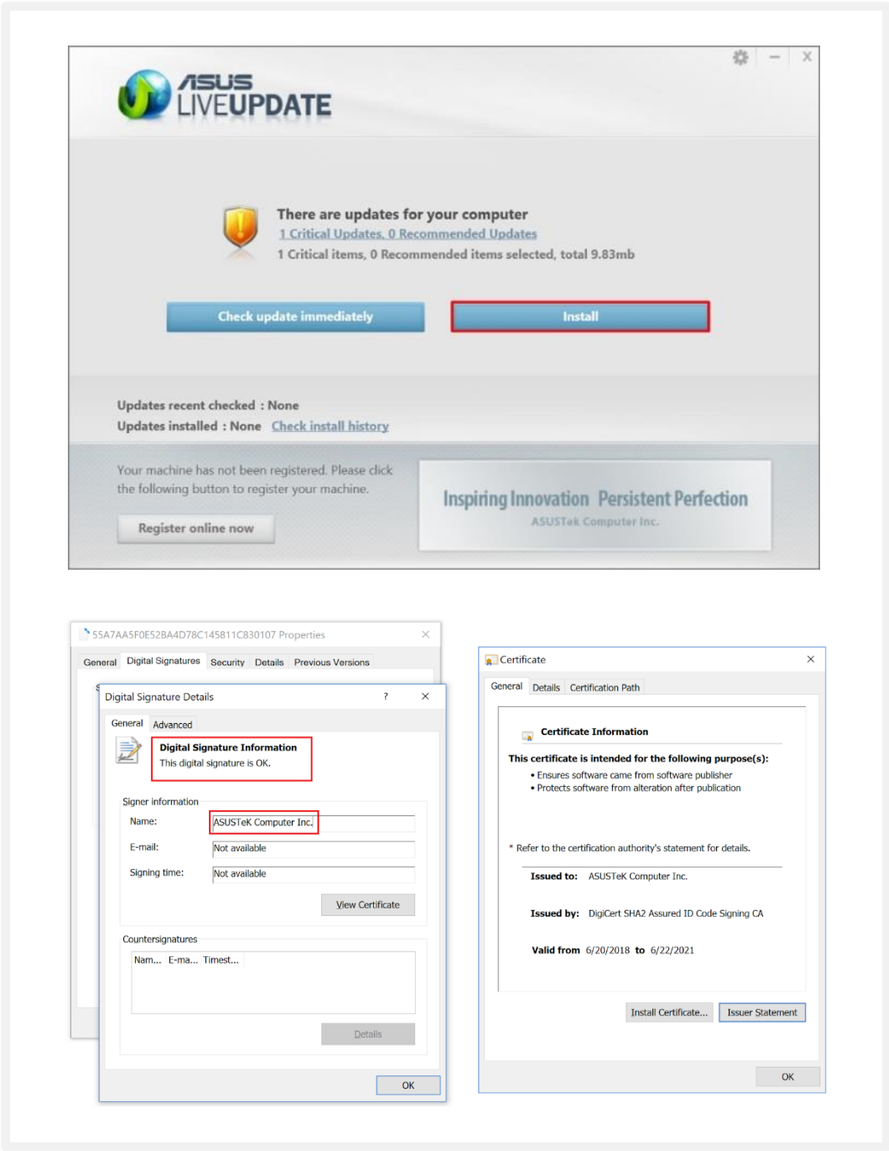
- SecureBoot
- Signed Update
- Anti-rollback protection
- Enable and enforce security boundaries (boot vs runtime)
- Lockdown or disable interfaces
- Disable debug interface
- Secure recovery mode

Compromises in Technology Supply Chain

- Examples of firmware supply chain issues:
  - OEM update tools and priv esc
  - MSI, Gigabyte, HP, Acer, ASRock, Dell, etc. not signing their images
  - LVFS legacy S3 bucket takeover, and GPG bypass

ShadowHammer

- ASUS Live Update Utility pushed malware-infected software updates for 6 months
- Had hardcoded target MAC addresses to act on, but was pushed out to hundreds of thousands
- It remained undetected for 8 months
- Signed with legitimate ASUS security certificates, hosted on legitimate ASUS update domains



Firmware Supply Chain Concerns

Created in good state (security focused development)

- Make sure that firmware is built with security in mind
- Collect, save, sign firmware blobs and other firmware data

Delivered in good state (delivery assurance)

- Validate that the initial firmware/hardware is untouched when it gets to the end customer

Maintained in good state (secure firmware update process)

- Ensure that the end customer can securely verify and update firmware to the latest

If You Are Developing Secure Firmware

- Firmware is a prime target for basic and sophisticated attacks. It will be compromised. Plan for it.

- Key steps for firmware are:
  a) Enabling every security protection
  b) Detect something has been attacked
  c) Reliable way to recover

- Work with industry groups (UEFI, TCG) and industry partners (like Insyde and Eclypsium) to improve the standard building blocks

If You Are Deploying Firmware Updates

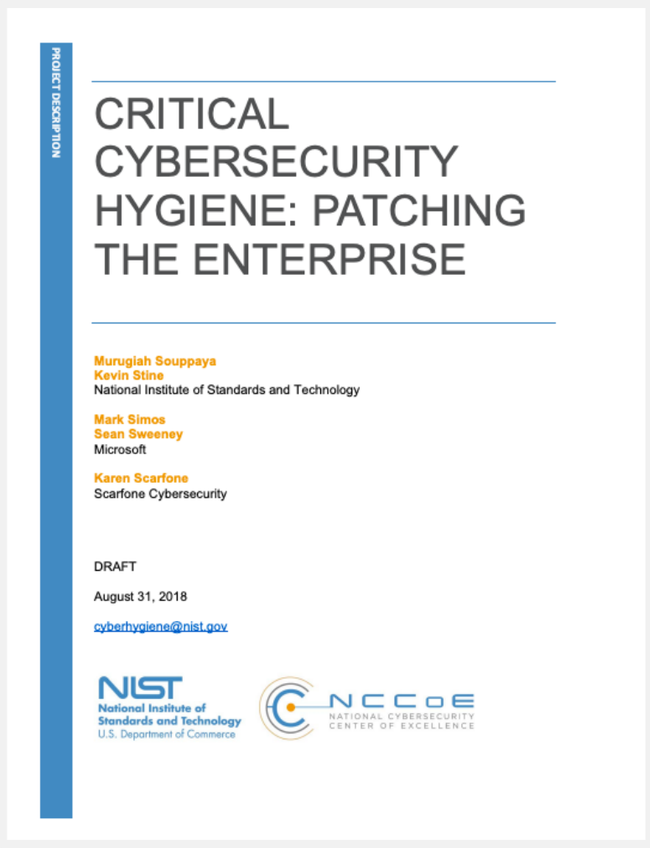
- Treat firmware updates with the same discipline that you apply to software updates
- Get visibility into the hardware and firmware in your fleet
- Invest in the tooling and skills you need to manage testing, rollout and rollback
- Make firmware support a priority in your hardware purchasing decisions

Building a Firmware Update Program

"Where firmware update management was once considered a nice-to-have component in an organization's plans for dealing with vulnerabilities, it has now evolved into one of the central elements of any successful security program."

Dr. Ed Amoroso, CEO of research and advisory firm TAG Cyber and former CISO for AT&T

Lessons from Published Resources



PROJECT DESCRIPTION

CRITICAL CYBERSECURITY HYGIENE: PATCHING THE ENTERPRISE

Murugiah Souppaya
Kevin Stine
National Institute of Standards and Technology

Mark Simos
Sean Sweeney
Microsoft

Karen Scarfone
Scarfone Cybersecurity

DRAFT

August 31, 2018

cyberhygiene@nist.gov

NIST National Institute of Standards and Technology U.S. Department of Commerce

NCCoE NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

intel

IT@INTEL

Developing a Gold Standard for Driver and Firmware Maintenance

Our Gold Standard configuration and process for drivers and firmware maintenance allow us to improve the health of our PCs, as well as the user experience and productivity.

Executive Overview

Advances in modern client computing have brought broader features and capabilities to enterprises, helping them develop new markets, innovate new products, and improve communication. But these advancements have also brought new challenges. Solution providers are releasing updates faster and more frequently, and not every update is required for every environment. Microsoft Windows® 10, as well as systems built on Intel® architecture, may require upgrades to drivers and firmware to provide all the capabilities and functionality users expect.

At Intel IT, we developed a Gold Standard configuration for our environment based on our experience and our lessons learned. We determine which drivers and firmware to upgrade based on specific criteria that balance the need to upgrade a platform against the disruption it might cause. We consider the security risk, whether the upgrade fixes known bugs, if the upgrade is required for new features or OS upgrades, and if the upgrade should be included in our standard build. Our process includes:

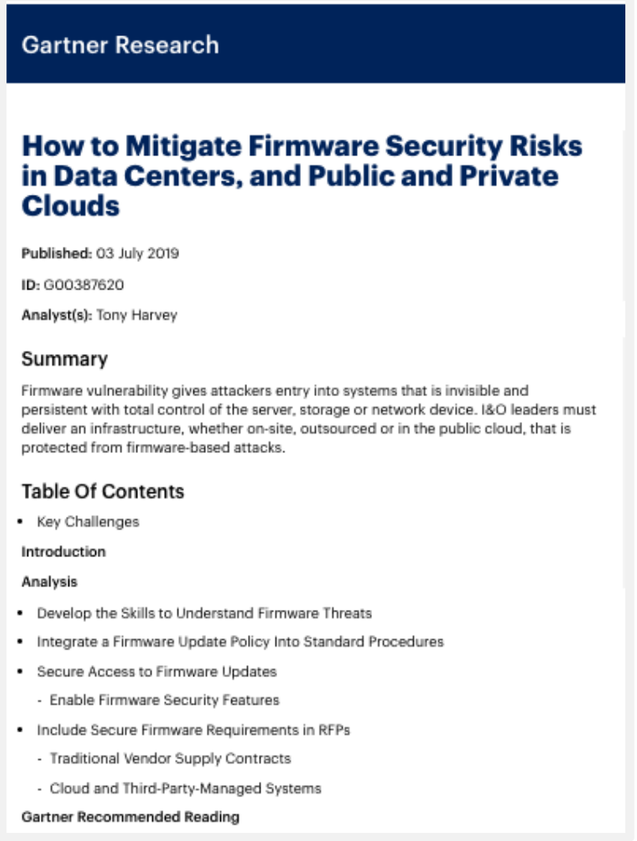
- Identifying prerequisites
- Testing and deployment
- Monitoring and communication
- Contingency planning

Our Gold Standard configuration and the process for driver and firmware maintenance has allowed us to mitigate many of the problems and vulnerabilities that older, non-optimized drivers and firmware can introduce into the environment, and has also helped Intel to remain focused on security. Overall, we have increased our success rate to more than 90 percent and improved the user experience.

Ricky Tejinder Singh
Client Platform Engineering Manager, Intel IT

Laura Mainer
Configuration Manager, Intel IT

Dan Costenaro
MLAM Product Engineer, Intel IT



Gartner Research

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

Published: 03 July 2019

ID: G00387620

Analyst(s): Tony Harvey

Summary

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.

Table Of Contents

Questions?

Thanks for attending the UEFI 2020 Virtual Plugfest

For more information on UEFI Forum and UEFI Specifications, visit http://www.uefi.org

presented by