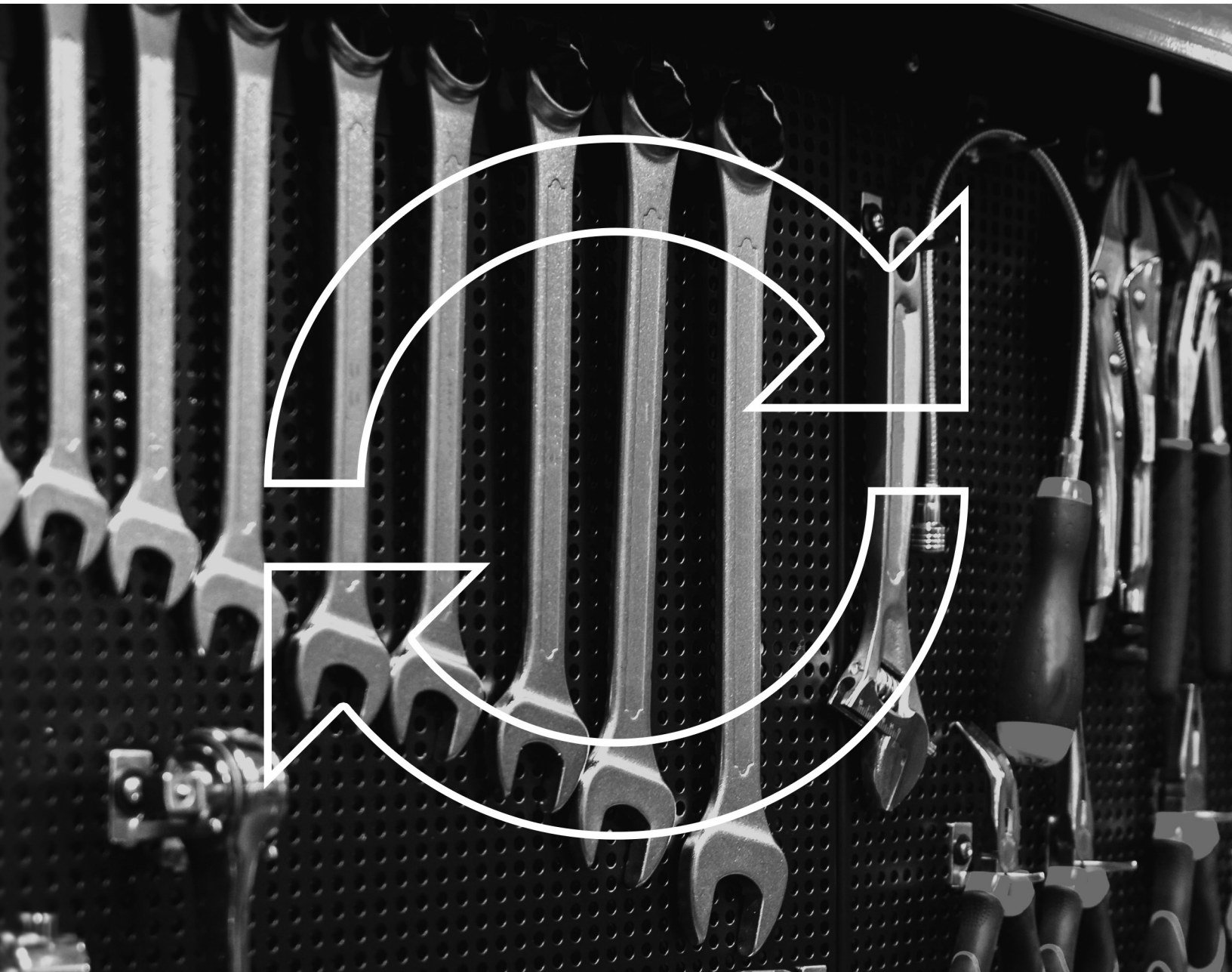
eclypsium

# Tools and Techniques for Updating Enterprise Firmware



# INTRODUCTION

*Part two of Eclypsium's series on best practices for firmware updates focuses on the tools and techniques used by the enterprise IT teams tasked with implementing update processes.*

Enterprise IT teams must maintain firmware on a wide variety of devices, ranging from end-user laptops to fleets of servers and networking gear. Each individual device can contain dozens of individual firmware components that need to be kept up to date — the system UEFI firmware, drives, network adapters, server BMCs, and much more.

The tools and techniques used to update firmware on a device can vary considerably from one vendor to the next, and often even between different products from a single vendor. Different operating systems have different capabilities and play differing roles in the update of system and component firmware. This can create a potentially confusing landscape for IT teams as they try to maintain firmware across their many devices and components.

To avoid such confusion, it is important to know the many tools, features, and capabilities provided by the various vendors and operating systems. While IT teams should always refer to the official vendor documentation for a given tool, this paper provides a high-level comparison across multiple vendors and technologies to highlight how they are similar, how they are different, and how differences can affect the firmware update process.

# Vendor Tools for Updating Firmware

Virtually all enterprise-grade vendors include tools to help teams check the current version of firmware and update firmware on a device. However, each vendor's approach is somewhat distinct and can vary between product lines. This section highlights some of the core capabilities and differences between the tools of leading vendors.

## HP BIOS Configuration Utility (BCU)

The HP BIOS Configuration Utility is one of the tools included under the umbrella of HP Client Management Solutions. As the name implies, it is an update utility for Windows and the BIOS configuration utility (BCU), a free utility that enables you to do the following:

- Read the BIOS settings.
- Reset the setup password.
- Replicate BIOS settings across multiple client computers.

The update's executable utility generally works for a very limited amount of systems (systems that use similar UEFI firmware). And like almost all firmware updating tools, updates require a reboot.

## HPE SPP and OneView

HPE Service Pack for ProLiant (SPP) is a software and firmware update tool for HP Enterprise (HPE) servers and also provides an ISO for manual updating. The tool works in conjunction with HP Smart Update Manager (HP SUM).

SPP contains a variety of firmware for the system; however, applying new firmware is not easy. The process requires a Media Server to provide updates, mount the SPP files, and perform several steps. Read a summary of the steps involved.

HPE OneView provides an overall approach to IT infrastructure management. This overall management platform includes the ability to update firmware on HPE platforms such as HPE ProLiant DL or ML, HPE BladeSystem, HPE Apollo or Mission Critical Servers, HPE 3PAR, and HPE Synergy.

HPE also provides the CONREP tool, which allows teams to read and write BIOS configurations from the command line.

## HPE Intelligent Provisioning

HPE Intelligent Provisioning is a tool built into HPE servers which can be accessed by pressing a hotkey while the server boots and provides automated provisioning and firmware update capabilities. In addition to performing BIOS and BMC firmware updates, this tool can also update firmware on RAID controllers, Network controllers, and other add-in cards.

## Lenovo System Update

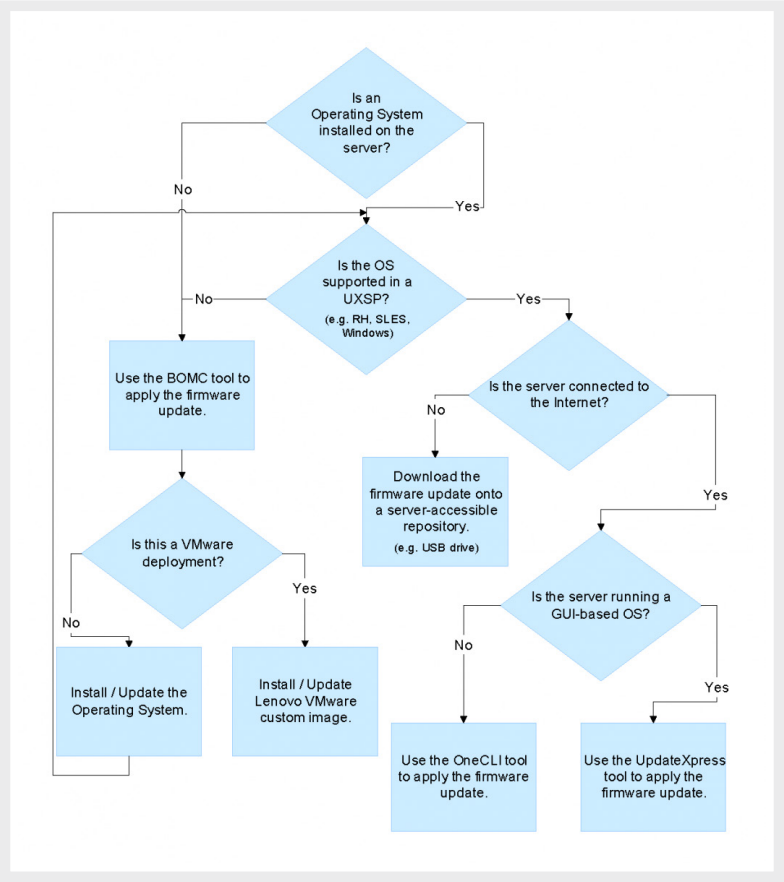
Lenovo System Update is a tool that automates the process of finding and installing the latest drivers, BIOS, and other applications. This tool can be installed in the host OS or run on demand. Notably, Lenovo System Update works across virtually any Lenovo device and thus provides a consistent way to check the firmware of Lenovo systems.

Lenovo also provides Windows executables for updating each firmware type individually. However, these executables only work for some specific systems or a small family of systems. Lenovo systems that run Linux can be managed by the Linux Vendor Firmware Service (LVFS) described later in this document.

## Lenovo Enterprise Tools

Lenovo enterprise systems, which primarily includes servers as well as some workstations, rely on a variety of different methods and tools to update firmware, depending on the deployment type. These tools include XClarity Essentials Bootable Media Creator (BoMC), XClarity Essentials UpdateXpress, and XClarity Essentials OneCLI.

The following Lenovo image provides an overview of the different methods that are used in different situations.



## Dell Client Configuration Toolkit, Command | Configure

Dell has provided tools under several names that allow staff to modify Dell BIOS settings on workstations. The Dell Client Configuration Utility (DCCU), Client Configuration Toolkit (CCTK), and the Command | Configure tools provide similar capabilities tied to specific generations and types of Dell hardware. The tools work by creating a stand-alone package that is manually run on a Dell client computer to configure a BIOS, update a BIOS, or capture BIOS settings inventory data.

Dell also provides individual executable packages for Windows for updating firmware in Windows. However, these executable wrappers can change considerably between versions and only work with specific systems or families of systems. Additionally, Dell publishes SCUP catalogs for SCCM that support automated firmware updates.

## Dell Enterprise Tools

For Dell enterprise systems, the update process almost always includes the BMC. Dell's BMC utility helps users manage the system remotely, including flashing BIOS updates. One can also access the server and run the executable provided to update it manually.

Dell also provides a Dell EMC Server Update Utility which allows for updating the firmware of the server with physical or remote access to the system. Additionally, Linux-based Dell systems can use the LVFS service.

## Cisco Firmware Updates

Cisco updates are uploaded as a ZIP package and delivered via the baseboard management controllers (BMC). Cisco also provides the option to update using Auto-Install via the Cisco UCS Manager.

For Cisco IOS  devices, updates are delivered via a tFTP server of the organization's choosing. The team will then need to run commands on the target networking device IOS  console to fetch, install, and set up the new OS as described here.

## Apple

Apple firmware updates are included in MacOS updates automatically as part of the regular update process. Find more information from Apple.

## IPMI Standard

The Intelligent Platform Management Interface (IPMI) is a specification that provides for out-of-band management and monitoring of servers. The initial version of the specification was published in 1998 and is supported by the majority of server vendors. IPMI provides a standardized message-based interface and set of protocols with a core set of mandatory commands, specification-defined optional commands, and the ability to support OEM-specific commands. Some vendors provide the ability to perform firmware updates over IPMI, but others require that the user login to the BMC web management UI and upload the firmware through this mechanism. Vendors may also publish their own management tools by adding their own capabilities on top of the IPMI standard through the use of OEM-specific commands.

### RedFish Standard

Developed by the Distributed Management Task Force (DMTF), the Redfish standard provides a RESTful API for the management of servers and IT infrastructure and attempts to address some of the issues present in IPMI. This includes the ability to inventory and update firmware on devices. As an industry standard, Redfish is supported by a variety of vendors including Dell, HPE, Lenovo, and many others. However, Redfish is a relatively new standard and vendors are still adding support for it in devices. For example, a vendor may support both Redfish and IPMI on the latest generation of a device, while previous generations may only support IPMI or a more limited set of Redfish capabilities.

## COMMONALITIES ACROSS VENDORS AND TOOLS

There are a variety of tools related to firmware and a variety of approaches to firmware updates. However, there are some areas that are relatively similar across vendor approaches.

In terms of dependencies, most of the tools are portable. Windows tools typically don't need anything other than the OS runtime, and the bootable ISOs are also self-contained (in this case, one can consider the physical dependency of having a bootable storage device in which to write the ISO).

Most vendors support a "silent mode." The parameters for each vendor's silent mode are as follows:

- **HP:** Run the application with the /Silent parameter.
- **Dell:** Run the application with /silent or /s.
- **Lenovo:** Lenovo provides two parameters, /silent and /verysilent. There are reported cases where the silent mode doesn't work. It depends on how the application is packaged and its version.

Additionally, all tools that update the system firmware will require a reboot.

## OPEN SOURCE FIRMWARE OPTIONS

As an alternative to vendor-supplied tools, some organizations may want to consider an open-source approach to firmware updates and even development. While this option can require considerable development resources, organizations may have specific needs that make it desirable. With direct control over the firmware update process, organizations can:

- Create a more standardized way of managing their firmware that is not limited by the capabilities of a given vendor's tools.
- Establish their own hardware root of trust based on their own certificates.
- Develop their own firmware update and release cadence.

In order to take an open-source approach to firmware updating, organizations will need to use hardware that supports open source firmware such as coreboot, LinuxBoot or openBMC. Again, directly maintaining firmware can be risky and requires a very technically savvy team in order to support the project safely. Organizations should therefore carefully review their requirements and capabilities before adopting this approach. However, if needed, teams may actively select for vendors that support these standards or pressure vendors to support them via feature requests.

## A BIOS VENDOR PERSPECTIVE

Firmware updates are usually considered risky. However, everyone seems willing to do security-related updates. Users rely on OS or platform vendor notifications, but some users want to be more informed.

Current solutions rely on checking whether the firmware version is the latest firmware version. This works for functional issues but not for security issues. It does not help when the firmware image has been tampered with (but still reports the same version information) or is misconfigured.

One solution is to add fingerprints. Fingerprinting detects undesirable changes in the immutable portions of the firmware image. Tools can check the fingerprint against an online database of golden images and provide updates if it is not found. This requires an update to the build process to insert flash usage information into the flash binary, such as a hash of the logical regions and information about whether the region is expected to be modified. The fingerprint itself can be signed, allowing auditing by tools or BMCs purely through inspection of the binary to detect post-manufacturing tampering.

Another solution is checking whether the security-related firmware settings are correct. Bad settings can weaken the security even if the platform has the right firmware version. Tools like CHIPSEC and Eclypsium evaluate the security-related settings of the platform and generate a report.

Using this report, tools can provide a simple pass/fail indicator to users to let them know that something is wrong, along with instructions about which settings need to be changed. The tool could also download setting updates that can be automatically applied.

Alerting users that their platform might be compromised improves the adoption of firmware updates. Insyde uses these techniques in its InsydeH2O firmware and InsydeSST tools to alert users and help them resolve it.

insyde®

# FIRMWARE UPDATES VIA ENCAPSULATION

Recent industry efforts have focused on making firmware updates far more transparent and automated through a process called encapsulation. The UEFI specification provides a standardized mechanism for storing and processing updates as a "capsule" that is presented to firmware during the boot process.

The overall process works by allowing vendors to submit signed firmware updates to a central repository; the updates are then delivered to users and installed by the operating system. This provides a much more centralized approach to managing firmware updates and greatly streamlines the update process.

Adding this capability means vendors can support system firmware updates via a standardized format that could potentially be used by any OS. Most notably, Microsoft Update and the Linux Vendor Firmware Service (LVFS) have led the way with methods to enable the host operating system to automate installation via encapsulation. Likewise, Apple's macOS firmware updates are automatically included as part of regular OS updates via encapsulation.

While the major operating systems all support encapsulation in some form, there are gaps created both by each OS's approach as well as participation from the relevant OEMs and ODMs. For example, Microsoft-based updates are largely limited to UEFI firmware updates, so organizations may need to apply a more manual approach to updating peripheral components. The Linux-based LVFS, on the other hand, does provide encapsulation for both UEFI and peripheral firmware updates.

In any case, it is important to note that not all firmware vendors participate. This creates potential gaps where support will vary considerably based on the combination of operating system, OEM, and components within a device. Vendors such as Dell provide SCUP catalogs that include firmware for use with Microsoft SCCM, for example. However, not all vendors support such integration, and teams will need to be aware of the options on a vendor-by-vendor basis. Organizations must have visibility into their devices to understand which components can be updated via encapsulation and which will require a more manual approach.

# CONCLUSION

Maintaining firmware requires having the appropriate tools for the job, and these can come from the OEM vendors themselves, operating system vendors, open-source projects, or third-party vendors. The capabilities of these options can vary considerably, and it is important for IT and security teams to understand their differences and to remain up to date on the latest specifications. Likewise, these teams must actively consider the quality of tools provided by a vendor during the evaluation and acquisition of new technologies. This should allow them to identify any potential gaps in their existing tools and how that will affect their organization's overall firmware update strategy.

# ACKNOWLEDGMENTS / THANK YOU

Eclypsium would like to thank the following organizations that contributed to this report: Criteo, Insyde Software, Linux Foundation, Phoenix Technologies, TAG Cyber, and several contributors/reviewers who have asked to remain anonymous.

For additional information see Enterprise Best Practices for Firmware Updates.

# ABOUT ECLYPSIUM

Eclypsium is the most comprehensive cloud-based device security platform for modern distributed organizations. From corporate laptops to network equipment to servers in data centers, Eclypsium protects the devices that organizations rely on, all the way down to the firmware and hardware level. The Eclypsium platform provides security capabilities ranging from basic device health and patching at scale to protection from the most persistent and stealthiest threats. For more information, please visit eclypsium.com, follow us on Twitter @eclypsium, or request additional information at info@eclypsium.com.