BOOTHOLE: HOW IT STARTED, HOW IT'S GOING

Understanding the latest Secure Boot vulnerabilities, patches, protections, and what you need to do today.

It has been a little over nine months since Eclypsium researchers first published their findings on the BootHole vulnerability in the GRUB2 bootloader. BootHole garnered plenty of attention due to its potential to undermine the boot security of a device along with its incredibly wide reach, affecting virtually every Linux distribution as well as any device, including Windows devices, that used the industry-standard Microsoft Third Party UEFI Certificate Authority.

However, the BootHole research was far from the end of the story. The renewed interest in GRUB2 and boot security, in general, helped to kick off a new wave of industry research and collaboration. This led to the discovery of several new important CVEs and scores of patches to address them. Additionally, the industry began to look at ways to improve the revocation process itself to hopefully make it easier to respond to widespread vulnerabilities like BootHole in the future.

This is important because Secure Boot is the fundamental security control that allows the primary operating system to trust components that are loaded during the boot-up process. Threat actors use bootkits to disrupt this process and execute their malicious code prior to the OS running. Secure Boot is designed to forestall that possibility by preventing attackers from running unsigned code during the boot process, such as APT 28's Drovorub kernel-level bootkit, which, previously, was only able to target devices that did not have Secure Boot enabled. BootHole and related vulnerabilities afford present-day actors like these (and others) the ability to bypass Secure Boot, and thus greatly expand the number of attackable devices in the enterprise. Without Secure Boot, device and operating system integrity cannot be trusted. The magnitude of the BootHole class of vulnerabilities cannot be overstated

In this post, we aim to highlight some of the key developments in this area to help promote and inform on the progress being made by other researchers in the industry. Hopefully, these updates will help security practitioners to make well-informed, risk-based patching decisions by understanding the vulnerabilities that could affect their systems and the updates that are available to address them. This post leverages information published in the GNU grub-devel list and we encourage readers to review SECURITY PATCH 000/117 for more information on the topic.

NEW GRUB VULNERABILITIES

The publication of BootHole caught the attention of researchers across the industry and brought a fresh focus on the GRUB2 bootloader. Subsequent work from independent researchers as well as teams at

Canonical, IBM, and IOActive uncovered a variety of new flaws within GRUB. This included 8 new GRUB CVEs including 5 High and 3 Medium vulnerabilities based on CVSS v3 scores.

Fixes for these and other flaws resulted in a total of 117 patches including an initial implementation of stack protection for UEFI. This is a particularly important addition in that it aims to defend against an entire class of buffer overflows as seen in BootHole and other related vulnerabilities. This type of anti-exploitation protection has been incorporated into operating systems for years but largely was not applied to UEFI.

The vulnerabilities are summarized below, sorted by CVSS score

CVE	CVSS v3 Score	Summary
CVE-2020-25632	8.2 HIGH	The rmmod implementation allows the unloading of a module used as a dependency without checking if any other dependent module is still loaded leading to a use-after-free scenario. This could allow arbitrary code to be executed or a bypass of Secure Boot protections.
CVE-2021-20233	8.2 HIGH	Setparam_prefix() in the menu rendering code performs a length calculation on the assumption that expressing a quoted single quote will require 3 characters, while it actually requires 4 characters which allows an attacker to corrupt memory by one byte for each quote in the input.
CVE-2020-25647	7.6 HIGH	During USB device initialization, descriptors are read with very little bounds checking and assumes the USB device is providing sane values. If properly exploited, an attacker could trigger memory corruption leading to arbitrary code execution allowing a bypass of the Secure Boot mechanism.
CVE-2020-14372	7.5 HIGH	A flaw in GRUB2 incorrectly enables the ACPI command when Secure Boot is enabled. A privileged attacker could create an SSDT and overwrite the Linux kernel lockdown variable. The SSDT is further loaded and executed by the kernel, defeating its Secure Boot lockdown and allowing the attacker to load unsigned code.
CVE-2020-27779	7.5 HIGH	The cutmem command does not honor Secure Boot locking allowing a privileged attacker to remove address ranges from memory creating an opportunity to circumvent Secure Boot protections after proper triage about grub's memory layout.
CVE-2020-27749	6.7 MEDIUM	Variable names are expanded without sufficient bounds checking. If the function is called with a command line that references a variable with a sufficiently large payload, it is possible to overflow the stack buffer, corrupt the stack frame, and control execution which could also circumvent Secure Boot protections.
CVE-2021-20225	6.7 MEDIUM	The option parser allows an attacker to write past the end of a heap-allocated buffer by calling certain commands with a large number of specific short forms of options.
CVE-2021-3418	6.4 MEDIUM	If certificates that signed GRUB are installed into db, then GRUB can be booted directly and will then boot any kernel without signature validation. The booted kernel will think it was booted in Secure Boot mode and will implement lockdown, yet could have been tampered with.

UEFI SECURE BOOT ADVANCED TARGETING (SBAT)

BootHole has required an enormous amount of coordinated response across the industry, which is still ongoing today. Updating the dbx UEFI revocation database is an essential mitigation step to prevent attackers from using a vulnerable shim to gain control over a system's boot process. This naturally has required extensive testing at every step along the way. However, the widespread nature of BootHole meant that 3 certificates and 150 image hashes needed to be added to the dbx. This single update accounted for almost ⅓ of the storage allotted to the dbx.

UEFI Secure Boot Advanced Targeting (SBAT) aims to simplify the revocation process for shims, particularly for widespread events such as BootHole. This is important given that such vulnerabilities are likely to become even more common as both attackers and researchers increasingly focus on boot security.

SBAT works by taking a generational approach to revocation. Instead of requiring every individual image to be added to the dbx, SBAT would define global and product-specific generation numbers for key boot components such as the shim, GRUB, kernel, etc. Thus if there were multiple versions of the same shim with a vulnerability, they could all be addressed by version number instead of requiring a hash for each version and certificate. The same logic could apply to vulnerabilities that affect multiple versions of a particular component. Revocation could be targeted to the affected versions, again, without individually adding hashes for each version.

SBAT is a promising step that we will be watching closely. SBAT would itself require its own update to the dbx in order to take effect, but would then simplify subsequent shim updates. The process would depend on vendors adding a .sbat section to their PE files that would define component metadata such as the vendor name, component name, package version, etc. The industry would naturally need to make sure that vendors supply the necessary information and that it is accurate.

PROTECTIONS AND NEXT STEPS

Security teams need to do several things in order to address devices with vulnerable bootloaders in their environments. These efforts should not be limited to devices that use GRUB2, but will also apply to any device that uses the industry-standard Microsoft Third Party UEFI Certificate Authority in the UEFI db and dbx databases.

For devices that have a vulnerable version of GRUB2, teams need to deploy new signed bootloaders and shims as they are available. Note that the availability of a specific update package may vary from project to project or vendor to vendor such as Red Hat, Canonical, etc. As a result, security teams should closely monitor their relevant projects for updates. Once new versions are available, the vulnerable versions will need to be revoked to prevent them from being used in an attack. This means that teams will need to make sure that they are using the latest stable OS and that the dbx revocation database is up to date. Organizations will need to first update their operating system and bootloaders before the dbx database is updated.

Like the updates themselves, the revocation updates are also a work in progress, and teams will need to watch for them to be available at https://uefi.org/revocationlistfile.

The Eclypsium platform can greatly simplify these steps and automate much of the work. It automatically identifies devices that have vulnerabilities related to BootHole and other Secure Boot issues, and locates available updates. Eclypsium can also detect devices that have been compromised by threats such as bootkits. Organizations can gain visibility across their entire fleet of Windows and Linux devices to quickly identify their risk. This includes visibility into system bootloaders, EFI system partition, UEFI firmware, and the dbx revocation database. Likewise, Eclypsium scans can automatically identify devices with vulnerable bootloaders, missing updates, and detect the presence of malicious bootloaders and exploit code on compromised devices.

CONCLUSIONS

Boot security continues to be a highly active area of cybersecurity. The industry is ramping up the rate at which it discovers problems while seeking out better ways to respond to those problems once they are found.

However, real-world responses are often quite slow, and in our experience, it remains exceedingly common to see organizations and their devices that are vulnerable to BootHole. As the industry evolves, it is all the more important that organizations have the visibility and tools to find and address their risk in their devices' firmware and boot process. The Eclypsium team continues to be available to help organizations deal with the risks of BootHole and the wide variety of related vulnerabilities and risks. To learn more please contact Eclypsium at info@eclypsium.com.