How the NSA's Firmware Hacking Works and Why It's So Unsettling

Kim Zetter ⋮ 11-14 minutes ⋮ 2/22/2015

One of the most shocking parts of the recently discovered spying network Equation Group is its mysterious module designed to reprogram or reflash a computer hard drive's firmware with malicious code. The Kaspersky researchers who uncovered this said its ability to subvert hard drive firmware---the guts of any computer---"surpasses anything else" they had ever seen.

The hacking tool, believed to be a product of the NSA, is significant because subverting the firmware gives the attackers God-like control of the system in a way that is stealthy and persistent even through software updates. The module, named "nls_933w.dll", is the first of its kind found in the wild and is used with both the EquationDrug and GrayFish spy platforms Kaspersky uncovered.

It also has another capability: to create invisible storage space on the hard drive to hide data stolen from the system so the attackers can retrieve it later. This lets spies like the Equation Group bypass disk encryption by secreting documents they want to seize in areas that don't get encrypted.

Kaspersky has so far uncovered 500 victims of the Equation Group, but only five of these had the firmware-flashing module on their systems. The flasher module is likely reserved for significant systems that present special surveillance challenges. Costin Raiu, director of Kaspersky's Global Research and Analysis Team, believes these are high-value computers that are not connected to the internet and are protected with disk encryption.

Here's what we know about the firmware-flashing module.

How It Works

Hard drive disks have a controller, essentially a mini-computer, that includes a memory chip or flash ROM where the firmware code for operating the hard drive resides.

When a machine is infected with EquationDrug or GrayFish, the firmware flasher module gets deposited onto the system and reaches out to a command server to obtain payload code that it then flashes to the firmware, replacing the existing firmware with a malicious one. The researchers uncovered two versions of the flasher module: one that appears to have been compiled in 2010 and is used with EquatinoDrug and one with a 2013 compilation date that is used with GrayFish.

The Trojanized firmware lets attackers stay on the system even through software updates. If a victim, thinking his or her computer is infected, wipes the computer's operating system and reinstalls it to eliminate any malicious code, the malicious firmware code remains untouched. It can then reach out to the command server to restore all of the other malicious components that got wiped from the system.

Even if the firmware itself is updated with a new vendor release, the malicious firmware code may still persist because some firmware updates replace only parts of the firmware, meaning the malicious portions may not get overwritten with the update. The only solution for victims is to trash their hard drive and start over with a new one.

The attack works because firmware was never designed with security in mind. Hard disk makers don't cryptographically sign the firmware they install on drives the way software vendors do. Nor do hard drive disk designs have authentication built in to check for signed firmware. This makes it possible for someone to change the firmware. And firmware is the perfect place to conceal malware because

antivirus scanners don't examine it. There's also no easy way for users to read the firmware and manually check if it's been altered.

The firmware flasher module can reprogram the firmware of more than a dozen different hard drive brands, including IBM, Seagate, Western Digital, and Toshiba.

"You know how much effort it takes to land just one firmware for a hard drive? You need to know specifications, the CPU, the architecture of the firmware, how it works," Raiu says. The Kaspersky researchers have called it "an astonishing technical accomplishment and is testament to the group's abilities."

Once the firmware is replaced with the Trojanized version, the flasher module creates an API that can communicate with other malicious modules on the system and also access hidden sectors of the disk where the attackers want to conceal data they intend to steal. They hide this data in the so-called service area of the hard drive disk where the hard disk stores data needed for its internal operation.

Hidden Storage Is the Holy Grail

The revelation that the firmware hack helps store data the attackers want to steal didn't get much play when the story broke last week, but it's the most significant part of the hack. It also raises a number of questions about how exactly the attackers are pulling this off. Without an actual copy of the firmware payload that gets flashed to infected systems, there's still a lot that's unknown about the attack, but some of it can be surmised.

The ROM chip that contains the firmware includes a small amount of storage that goes unused. If the ROM chip is 2 megabytes, the firmware might take up just 1.5 megabytes, leaving half a megabyte of unused space that can be employed for hiding data the attackers want to steal.

This is particularly useful if the the computer has disk encryption enabled. Because the EquationDrug and GrayFish malware run in Windows, they can grab a copy of documents while they're unencrypted and save them to this hidden area on the machine that doesn't get encrypted. There isn't much space on the chip for a lot of data or documents, however, so the attackers can also just store something equally as valuable to bypass encryption.

“Taking into account the fact that their GrayFish implant is active from the very boot of the system, they have the ability to capture the encryption password and save it into this hidden area,” Raiu says.

Authorities could later grab the computer, perhaps through border interdiction or something the NSA calls "customs opportunities," and extract the password from this hidden area to unlock the encrypted disk.

Raiu thinks the intended targets of such a scheme are limited to machines that are not connected to the internet and have encrypted hard drives. One of the five machines they found hit with the firmware flasher module had no internet connection and was used for special secure communications.

"[The owners] only use it in some very specific cases where there is no other way around it," Raiu says. "Think about Bin Laden who lived in the desert in an isolated compound---doesn't have internet and no electronic footprint. So if you want information from his computer how do you get it? You get documents into the hidden area and you wait, and then after one or two years you come back and steal it. The benefits [of using this] are very specific."

Raiu thinks, however, that the attackers have a grander scheme in mind. "In the future probably they want to take it to the next level where they just copy all the documents [into the hidden area] instead of the password. [Then] at some point, when they have an opportunity to have physical access to the system, they can then access that hidden area and get the unencrypted docs."

They wouldn't need the password if they could copy an entire directory from the operating system to the hidden sector for accessing later. But the flash chip where the firmware resides is too small for

large amounts of data. So the attackers would need a bigger hidden space for storage. Luckily for them, it exists. There are large sectors in the service area of the hard drive disk that are also unused and could be commandeered to store a large cache of documents, even ones that might have been deleted from other parts of the computer. This service area, also called the reserved are or system area, stores the firmware and other data needed to operate drives, but it also contains large portions of unused space.

An interesting paper (.pdf) published in February 2013 by Ariel Berkman, a data recovery specialist at the Israeli firm Recover, noted "not only that these areas can't be sanitized (via standard tools), they cannot be accessed via anti-virus software [or] computer forensics tools."

Berkman points out that one particular model of Western Digital drives has 141 MB reserved for the service area, but only uses 12 MB of this, leaving the rest free for stealth storage.

To write or copy data to service area requires special commands that are specific to each vendor and are not publicly documented, so an attacker would need to uncover what these are. But once they do, "[b]y sending Vendor Specific Commands (VSCs) directly to the hard-drive, one can manipulate these [service] areas to read and write data that are otherwise inaccessible," Berkman writes. It is also possible, though not trivial, to write a program to automatically copy documents to this area. Berkman himself wrote a proof-of-concept program to read and write a file of up to 94 MB to the service area, but the program was a bit unstable and he noted that it could cause some data loss or cause the hard drive to fail.

One problem with hiding large amounts of data like this, however, is that its presence might be detected by examining the size of the used space in the service area. If there should be 129 MB of unused space in this sector but there's only 80 MB, it's a dead giveaway that something is there that shouldn't be. But a leaked NSA document that was written in 2006 but was published by *Der Spiegel* last month suggests the spy agency might have resolved this particular problem.

NSA Interns to the Rescue

The document (.pdf) is essentially a wish list of future spy capabilities the NSA hoped to develop for its so-called Persistence Division, a division that has an attack team within it that focuses on establishing and maintaining persistence on compromised machines by subverting their firmware, BIOS, BUS or drivers. The document lists a number of projects the NSA put together for interns to tackle on behalf of this attack team. Among them is the "Covert Storage" project for developing a hard drive firmware implant that can prevent covert storage on disks from being detected. To do this, the implant prevents the system from disclosing the true amount of free space available on the disk.

"The idea would be to modify the firmware of a particular hard drive so that it normally only recognizes, say, half of its available space," the document reads. "It would report this size back to the operating system and not provide any way to access the additional space." Only one partition of the drive would be visible on the partition table, leaving the other partitions---where the hidden data was stored---invisible and inaccessible.

The modified firmware would have a special hook embedded in it that would unlock this hidden storage space only after a custom command was sent to the drive and the computer was rebooted. The hidden partition would then be available on the partition table and accessible until the secret storage was locked again with another custom command.

How exactly the spy agency planned to retrieve the hidden data was unclear from the eight-year-old document. Also unclear is whether the interns ever produced a firmware implant that accomplished what the NSA sought. But given that the document includes a note that interns would be expected to produce a solution for their project within six months after assignment, and considering the proven ingenuity of the NSA in other matters, they no doubt figured it out.