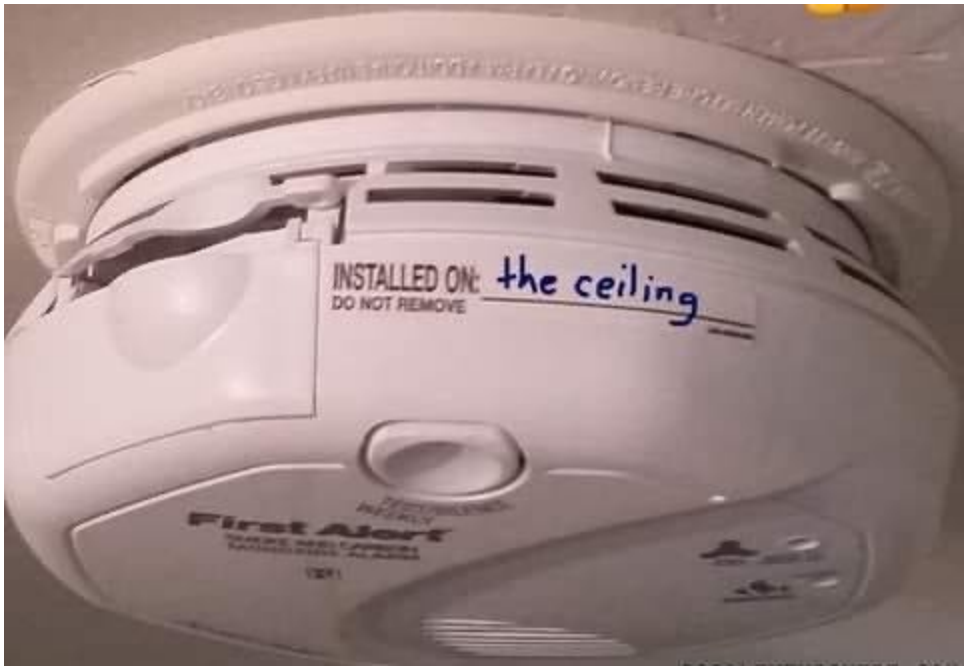
Security Now! #912 - 02-28-23

The NSA @ Home

This week on Security Now!

What mistake did Windows Update make last week? What if you don't want to paste with formatting? What browser is building-in a limited bandwidth VPN? What more did we just learn about LastPass's second breach? What did Signal say to the UK about scanning its user's messages? What was just discovered hiding inside the Python package Index repository? What proactive move has QNAP finally taken? What disastrous bug did SpinRite's testers uncover last weekend in motherboard BIOSes? And what amazingly useful "Best Practices" advice has the NSA just published for home users? Answers to all those questions and some additional thoughts will be yours — before you know it — on this week's 912th episode of Security Now!, titled: "The NSA @ Home."

I don't think that's what they had in mind…



Security News

Windows 11? … anyone?

Yesterday morning I received the following eMail from a podcast listener and SpinRite 6.1 tester. Jeremy wrote from Texas:

> *Hi. FYI...About Wednesday of last week, Windows 10 Update offered to "update" me to Windows 11 even though my HP AiO (all-in-one) does NOT qualify because it does not have the latest boot security level.*
>
> *I immediately switched over to GRC.com and downloaded InControl and set my PC to stay on: Windows 10, 22H2 security updates. When I reloaded Windows Update, Windows 11 was no longer Offered.*
>
> *I thought I was "safe" from Windows 11 since I didn't qualify, but....I guess not. Just wanted to let you know to combine my report with others if any. It's a reminder to be proactive rather than rely on assumptions. I have invested a lot of muscle memory in Windows 10 and don't think Win11 will offer me much.*
>
> *I don't know if it was a momentary glitch in Win Update or a real offer since I got InControl immediately.*
>
> *Regular listener and 6.1 tester..*
> *../Jeremy in Texas*

And also yesterday came the news that Microsoft had fixed a bug that was responsible for causing upgrade offers to unsupported PCs. Apparently, the issue came to light last Thursday and was quickly resolved and the fix was then pushed out to affected devices over the weekend. And this isn't the first time this has happened. Windows 11 22H2 was previously offered to Windows 11 Insiders in the Release Preview channel even when they were using ineligible devices. So it was that after Microsoft was aware of this but before they had pushed the update, Windows users were reporting via Reddit and Twitter that their unsupported devices were, to their surprise, being offered 11 upgrades.

Microsoft confessed: *"Some hardware ineligible Windows 10 and Windows 11, version 21H2 devices were offered an inaccurate upgrade to Windows 11. These ineligible devices did not meet the minimum requirements to run Windows 11. Devices that experienced this issue were not able to complete the upgrade installation process."*

Apparently some users with Windows 10 and even some who were using Win11 21H2 were surprised and delighted by this news (whereas I would have been horrified by it). The impacted devices included those running Windows 11 21H2, Windows 10 21H2, and Windows 10 20H2. And, as Microsoft indicated, some later portion of the upgrade process recognized the mistake and aborted the upgrade.

Remember that last month Microsoft announced that it had started a forced rollout of Windows 11 22H2 (also know as the Windows 11 2022 Update) to systems running Windows 11 21H2,

approaching their end-of-support (EOS) date which will be later this year on October 10th, 2023. And this automated feature update rollout phase came after the Windows 11 2022 update also became available for broad deployment the same day to users with eligible devices via Windows Update.

As it happens, the little Win10 machine I use weekly for zooming this podcast with you, Leo, has recently been bugging me to upgrade it to Windows 11. Since I only turn it on before the podcast and then off immediately afterward, I hadn't gotten around to running GRC's *"InControl"* utility to tweak the registry to get Microsoft to leave me alone. Recall that *"InControl"* is the spiritual successor to *"Never10"* — I still remember you laughing, Leo, when you first heard that name. The expectation was that *"Never10"* would be all that was ever needed. But then Microsoft decided that Windows 10 would **not** be the last Windows after all. So rather than creating *"Never11"* then *"Never12"* and so on, and needing to keep changing the name, I switched to *"InControl"*. Anyway, this story finally prompted me to take action — and InControl worked beautifully:

Initially, Windows Update was showing the upgrade offer at the top. So I ran InControl and told it that I wanted to stay put at Win10 22H2. I clicked the button, it tweaked the registry, and that was that. Then I reran Windows Update and the offer for Windows 11 was still there since it hadn't refreshed. So I asked Windows Update to re-check for any updates and that refreshed the screen. The offer for Windows 11 disappeared and I received the red asterisked notice at the top reading *"*Some settings are managed by your organization"*. Perfect. Leave... me... alone.

So, just a reminder on this occasion, that there's a little, free, 88K digitally-signed GRC-style Windows apps that will induce Windows Update to leave you alone. It doesn't remain resident. It needs no installation. You just run it once and it tweaks the registry to tell Windows Update that your system is being managed by the "Higher Ups" ... so not to bother you with any upgrade opportunities. Meanwhile, all security updates continue to be received as normal, just no more pushing to move.

As Plain as Ever

This next item is not security related but I wanted to make sure everyone was at least aware of it. I also have no sense for how large the audience for this might be, but the facility is one of my most-used keystroke combinations. It is: "Pasting from the clipboard without formatting." I'm a big user of copy and paste for moving things around. But I almost never want to also copy and paste any of the text formatting of the source text when I paste it. It's quite annoying when I paste something and it jumps into the appearance that it had, when all I want is the textual content itself – lose the text metadata.

Finally, a couple of years ago, after years of annoyance and doing things like using Windows Notepad as an intermediate to drop formatting since Notepad cannot retain any formatting then copying that bare text before pasting to the final destination, I thought "somebody else must have had this problem and fixed it." So I went Googling for something and I've been using a slick and clean little utility known as PureText ever since – and that's all it does. It sits in my system tray and, for me, CTRL-ALT-V performs a non-formatted textual clipboard paste. It's beautiful. And you can assign any combination of shift states and an action key that you like.

I'm mentioning this today because Windows PowerTools will soon be getting a new module called "Paste as Plain Text." I'm currently using PureText on both my Win7 and Win10 machines. But PowerTools won't run under Windows 7. So I'll likely stick with PureText. But for those who are Win10 or 11 only, you might find that "Paste as Plain Text" comes in handy. As I noted, I use this functionality constantly. https://SteveMiller.net

Edge's new built-in VPN?

Nearly a year ago, last April, The Verge carried the news *"Microsoft is adding a free built-in VPN to its Edge browser"* with the subhead that *"Edge Secure Network* [as it's being called] *will roll out as a part of a security upgrade."* And nearly a year later, this now appears to finally be coming to fruition since on Sunday BleepingComputer posted: *"Microsoft Edge's built-in VPN functionality could soon begin rolling out to all users in the stable channel, with some users already getting access to the feature."*

https://support.microsoft.com/en-us/topic/use-the-microsoft-edge-secure-network-to-protect-your-browsing-885472e2-7847-4d89-befb-c80d3dda6318

Okay, so what do we know? Microsoft explains their "Microsoft Edge Secure Network" with a few expected bullet points:

- ***Encrypts your connection:*** *Encrypts your internet connection to help protect your data from online threats like hackers. When using Microsoft Edge Secure network, your data is routed from Edge through an encrypted tunnel to create a secure connection, even when using a non-secure URL that starts with HTTP. This makes it harder for hackers to access your browsing data on a shared public Wi-Fi network.*

- ***Helps prevent online tracking:*** *By encrypting your web traffic directly from Microsoft Edge, we help prevent your internet service provider from collecting your browsing data like details about which websites you visit.*

- ***Keeps your location private:*** *Online entities can use your location and IP address for profiling and sending you targeted ads. Microsoft Edge Secure Network lets you browse with a virtual IP address that masks your IP and replaces your geolocation with a similar regional address to make it more difficult for online trackers to follow you as you browse.*

- ***Is free to use:*** *Get 1 gigabyte of free data every month when you sign into Microsoft Edge with your Microsoft Account. A few early adopters will be in a data upgrade trial, at the end of their 30-day trial period the experience will reflect the normal VPN gigabyte limits.*

Not surprisingly, my Edge doesn't have it yet. Under Edge's main menu near the menu's bottom, you'll find entries for "Read aloud" and "More tools." Assuming that Edge's UI hasn't changed since this posting was last updated, a new "Secure network" entry will appear in between "Read aloud" and "More tools."

The other piece of interesting news is that this is being done in affiliation with Cloudflare.

Microsoft wrote:

> *Microsoft Edge Secure Network is a service provided in partnership with Cloudflare. Cloudflare is committed to privacy and collects a limited amount of diagnostic and support data acting as Microsoft's data subprocessor in order to provide the services. Cloudflare permanently deletes the diagnostic and support data collected every 25 hours. To provide access, we store minimal support data and access tokens which are only retained for the duration of the required service window.*
>
> *A Microsoft account is required to access Microsoft Edge Secure Network and is retained to keep track of the amount of Microsoft Edge Secure Network data that is used each month. This data retention is necessary to provide 1GB of free Microsoft Edge Secure Network service and to indicate when the data limit has been reached.*

I don't really have any calibration on how quickly one gigabyte will be consumed, but that doesn't sound like much data for a month. But I checked my phone and I have the "small" Verizon plan since I'm not doing a lot with my phone. And I've used less than 0.3 gigabytes per month for the past three months.

So I expect that this might be something that's used sparingly and only when necessary. Edge's user interface has a "bytes used so far this month" meter, so it'll be possible to track one's usage.

Overall, this seems like a useful and welcome feature. It's limited, but it'll be there in a pinch.

LastPass Incident Update

Yesterday, LastPass provided far more detail about the second more devastating attack. I have to admit that the forensics presented are impressive. This doesn't forgive them from screwing up in the several other ways that we know they did, but as far as forensics examinations go, it's impressive. It's easily to tell a story in retrospect. But as i'm describing that they have determined imagine actually figuring this out:

https://support.lastpass.com/help/incident-2-additional-details-of-the-attack

LastPass has now learned and explained:

> *To access the cloud-based storage resources – notably S3 buckets which are protected with encryption – the threat actor needed to obtain AWS Access Keys and the LastPass-generated decryption keys. The encrypted cloud-based storage services house backups of LastPass customer and encrypted vault data.*
>
> *As mentioned in the first incident summary, certain LastPass credentials stolen during the first attack were encrypted and the threat actor did not have access to the decryption keys, which could only be retrieved from two locations:*
> - *A segregated and secured implementation of an orchestration platform and key-value store*

used to coordinate backups of LastPass development and production environments with various cloud-based storage resources, or

- *A highly restricted set of shared folders in a LastPass password manager vault that are used by DevOps engineers to perform administrative duties in these environments.*

Due to the security controls protecting and securing the on-premises data center installations of LastPass production, the threat actor targeted one of the four DevOps engineers who had access to the decryption keys needed to access the cloud storage service.

This was accomplished by targeting the DevOps engineer's home computer and exploiting a vulnerable third-party media software package, which enabled remote code execution capability and allowed the threat actor to implant keylogger malware. The threat actor was able to capture the employee's master password as it was entered, after the employee authenticated with multi-factor authentication, and gain access to the DevOps engineer's LastPass corporate vault.

The threat actor then exported the native corporate vault entries and content of shared folders, which contained encrypted secure notes with access and decryption keys needed to access the AWS S3 LastPass production backups, other cloud-based storage resources, and some related critical database backups.

Wow. No one wants to have those guys on their tail.

So listen to the steps they have since taken in an effort to recover from this attack. Remember how last week I was talking about how difficult it would be to ever be able to trust anything again. They wrote:

As we progress through incident response and as part of our on-going containment, eradication, and recovery activities related to the second incident, we have performed the following actions, with additional work currently being accomplished in scoping and planning:

- *With the assistance of Mandiant, we forensically imaged devices to investigate corporate and personal resources and gather evidence detailing potential threat actor activity.*
- *We assisted the DevOps Engineer with hardening the security of their home network and personal resources.*
- *We enabled Microsoft's conditional access PIN-matching multifactor authentication using an upgrade to the Microsoft Authenticator application which became generally available during the incident.*
- *We rotated critical and high privilege credentials that were known to be available to the threat actor; we continue to rotate the remaining lower priority items that pose no risk to LastPass or our customers.*
- *We began revoking and re-issuing certificates obtained by the threat actor.*
- *We analyzed LastPass AWS S3 cloud-based storage resources and applied or started to*

apply additional S3 hardening measures:

- *We put in place additional logging and alerting across the Cloud Storage environment with tighter IAM (Identity and Access Management) policies enforced.*
- *We deactivated prior development IAM users.*
- *We enabled a policy that prevents the creation and use of long-lived development IAM users in the new development environment.*
- *We rotated existing production service IAM user keys, applied tighter IP restrictions, and reconfigured policies to adhere to least privilege.*
- *We deleted obsolete service IAM users from the development and production environments.*
- *We are enabling IAM resource tagging enforcement on accounts for both users and roles with periodic reporting on non-compliant resources.*
- *We rotated critical SAML certificates used for internal and external services.*
- *We deleted obsolete/unused SAML certificates used for development, services, or third parties.*
- *We revised our 24x7 threat detection and response coverage, with additional managed and automated services enabled to facilitate appropriate escalation.*
- *We developed and enabled custom analytics that can detect ongoing abuse of AWS resources.*

One of the bugaboos of evolution is that things tend to become more complicated over time. This is usually driven by inevitably changing requirements. New systems are added to improve or enable some job. But the new system doesn't completely take over for the old one. So that older system needs to also remain around to do those few things that the newer system doesn't quite do the same. Then the requirements change again and some customizations are required and some custom glue code is created by someone who later quits and takes his notes and knowledge with him. Now, no one wants to touch that weird box since no one is quite sure how it works. And so on. Anyone who has been working within a complex environment with many players and constant time pressure, where needs are dynamically changing, will probably be able to relate to the sort of mess that winds up evolving from a once-simple solution. My point is, in the context of security, this sort of creeping evolving complexity makes both keeping things truly secure and recovering rapidly from a security incident much more difficult.

There really needs to be someone assigned the task of stepping back from the day to day fray to take a holistic view of an enterprise's systems and be constantly working to reintegrate the inherently disintegrating systems that naturally form. Keeping things as simple as possible has tremendous benefits for an organization, and in a sufficiently large organization, it really ought to be a job title.

Signal says NO to the UK

Exactly as we predicted, three days ago, BBC News headlined their coverage: *"Signal would 'walk' from UK if Online Safety Bill undermined encryption"* with the subhead: *"The encrypted-messaging app Signal has said it would stop providing services in the UK if a new law undermined encryption."*

Signal's president Meredith Whittaker told the BBC that if were forced to weaken the privacy of their messaging system under the Online Safety Bill, the organization "would absolutely, 100% walk." Of course, the government said that its proposal was not "a ban on end-to-end encryption." But the bill which was introduced by Boris Johnson is currently going through Parliament. And as we recently covered in detail, under this revisions proposed by this new legislation, companies would be required to scan messages on encrypted apps for child sexual abuse material, language suggestive of "grooming" or terrorism content.

WhatsApp previously told the BBC that it would also refuse to lower its "security" for any government. In the case of WhatsApp the question might come down to the definition of the term "Security." But the folks behind Signal are likely to be far more clear about that.

The BBC's coverage reminds us that the government and prominent child protection charities have long argued that encryption hinders efforts to combat online child abuse - which they say is a growing problem. The UK's Home Office said in a statement: "It is important that technology companies make every effort to ensure that their platforms do not become a breeding ground for pedophiles." The Home Office added "The Online Safety Bill does not represent a ban on end-to-end encryption but makes clear that technological changes should not be implemented in a way that diminishes public safety - especially the safety of children online. It is not a choice between privacy or child safety - we can and we must have both."

Right. We can, because we say we can. We're willing to go as far as to change the definitions of words in order to have both the safety of our children and total privacy for everyone, even though we may need to change the meaning of "total" – but just a little bit. That pesky math is so annoyingly absolute. After all, we create laws. That's what we do. Unfortunately not the laws of nature or of mathematics. But, still, this is what we want and we're used to getting our way.

The UK's child protection charity the NSPCC said in reaction to Signal's announcement: "Tech companies should be **required** to disrupt the abuse that is occurring at record levels on their platforms, including in private messaging and end-to-end encrypted environments." But the digital rights campaigners the Open Rights Group said this highlighted how the bill threatened to "undermine our right to communicate securely and privately".

The Signal's Ms Whittaker said "back doors" to enable the scanning of private messages would be exploited by "malignant state actors" and "create a way for criminals to access these systems."

When asked if the Online Safety Bill could jeopardize Signal's ability to offer a service in the UK, she told the BBC: "It could, and we would absolutely 100% walk away rather than ever undermine the trust that people place in us to provide a truly private means of communication. We have never weakened our privacy promises, and we never will."
Matthew Hodgson, chief executive of Element, a British secure communications company, said

the threat of mandated scanning alone, would cost him clients. He argued that customers would assume any secure communication product that came out of the UK would "necessarily have to have backdoors in order to allow for illegal content to be scanned." Matthew added that it could also result in "a very surreal situation" where a government bill might undermine security guarantees given to customers at the Ministry of Defense and other sensitive areas of government." He said that his firm might have to cease offering some services.

And that raises a great point... would the most sensitive users within the government also be consenting to having all of their communications intercepted, scanned, and possibly forward to a central clearinghouse for human oversight? That seems unlikely.

As for child safety, Signal's Ms Whittaker said: "There's no-one who doesn't want to protect children," and she added "Some of the stories that are invoked are harrowing." When asked how she would respond to arguments that encryption protects abusers, Ms Whittaker pointed to a paper by Professor Ross Anderson, which argued for better funding of services working in child protection and warned that "the idea that complex social problems are amenable to cheap technical solutions is the siren song of the software salesman."

There's no question that the issue of child safety is real. But terrorism content was also mentioned, and doesn't everyone also appreciate that no government, no matter how respectful of its citizen's inherent and often constitutionally guaranteed privacy rights, is comfortable with **not** having some capability for oversight over its citizenry when it believes that such might be needed. As I've noted of our own constitutional government in the US, the constitution's guarantee for privacy is conditional. Courts are able to issue search warrants when presented with probable cause.

We've been watching the approach of this slow motion collision for years. It is going to be very interesting to see how this shakes out. We know that Signal and Telegram and Threema will not capitulate. There's just no way they would. But it's difficult to see how Meta's various services, Google with Android and Apple may not be forced to go along rather than lose access to those huge markets. Apple already demonstrated a willingness to find some compromise by performing local fuzzy hash scanning. The backlash was epic.

Ultimately, I think, governments are probably going to win this legal battle since they're the ones who write the laws and thus it's possible for them to delineate what behavior is legal and what is not. At that point, any use of fully secure end-to-end encrypted solutions will be outlawed, at which point only outlaws will be using them.

More PyPI troubles

Remember last week how I felt that I needed to at least mention the continuous background of ongoing attacks on open source repositories and registries? Well, last Friday the security firm Phylum posted some news regarding the PyPI Python Package Index.

Their posting's headline was: *"Phylum Discovers Aggressive Attack on PyPI Attempting to Deliver Rust Executable."*

On the morning of February 23, 2023, Phylum's automated risk detection platform started lighting up with another series of strange publications on PyPI. After digging into it, we were able to link it to another smaller campaign from last month.

First, we can confirm that this is an ongoing attack. As we worked on this write up we saw the list of packages published go from a few dozen to over 500! The most recent packages appear to be getting published at around one every 4-8 seconds so we suspect this may continue for some time. You can look at the Package Publication section at the bottom of this post to see the packages we've seen (as of the publication of this post we've already seen 1,138 packages published.

They go on to explain in detail the nature of the malware. The short version is that the malicious packages connect to a Dropbox account to download and install a Rust-based malware strain. Phylum says the attacker appears to be the same group that was previously spotted by Fortinet and ReversingLabs the week before in a separate, smaller campaign.

So, as I was saying last week, our open source repositories are under more or less constant attack. And Leo, as you commented last week, the industry needs to come up with some solution to the poisoning of our open repositories. The only solution I can see is a future where Internet identity and reputation can be rigorously established and verified. People love the freedom of using synthetic online identities and monikers. And I think that's 100% fine, so long as they don't also want the benefits that accrue from being known and trusted. At least at the moment it's unclear how it's possible to have both. At the moment we have neither.

The QNAP bug bounty program

In the interest of giving credit where it's due, the often in the dog house Taiwanese hardware vendor QNAP on Friday announced the launch of its own bug bounty program. Vulnerabilities relating to QNAP operating systems, applications, and cloud services are in scope, and rewards can go up to US $20,000.

They still need to find some way of keeping their devices patched when problems are found, but as we've just seen with VMware, they're not alone in having that problem to solve. I think that tomorrow's systems are all going to have to phone home just as our consumer operating systems have for some time. And there will either have to be an autonomous upgrade and reboot facility, or some reliable notification path to the device's administrators.

But finding and fixing those problems comes first, and QNAP's bounty, while not huge, is a clear step in the right direction.

SpinRite

I haven't talked about SpinRite at all for several weeks because I didn't have any significant news to share. But as a result of the work and discoveries over the past week, I have news today which will more than make up for my previous weeks of silence...

A major mystery which had been stymieing the project for weeks has been solved. For the past many weeks I've been tracking down various sources of SpinRite crashes. I've discovered various sorts of misbehavior from DOS and motherboard BIOSes. They've been altering values in registers that they assume (wrongly) that others won't be using, which is against every rule of good citizenship. The only way I can explain the behavior is that they're using the upper-half of 32-bit registers figuring that in a 16-bit environment no 16-bit app will notice. But SpinRite is now largely 32-bit code and makes constant use of all of those extra 32-bit resources. And in other cases they're just not bothering to preserve any part of a register. So a lot of the work I've been doing recently has been defensive.

SpinRite already works without trouble for nearly everyone, but those for whom it does not work have my attention because there's no reason why it shouldn't be able to protect itself from anything that a system might throw at it. Though that was recently challenged.

One Canadian SpinRite tester, Andre, was able to get SpinRite to crash for him reliably. He had a system with a couple of internal drives and a 160 gigabyte USB drive. The USB drive was being marked RED by SpinRite, meaning that during its initial appraisal of the drive, SpinRite had found something that wasn't right. That's something we've been seeing with drives that are quite near death. When the user attempts to select the drive for use they receive a pop-up explanation of what's wrong and, when possible, will be given the option to proceed with that drive's selection.

But that 160GB drive being marked red wasn't the problem. The problem was that shortly after enumerating those three drives, SpinRite would intercept its own attempt to execute an illegal x86 processor instruction. That should never happen. I don't recall what made me suspicious, but I first asked Andre to try unplugging that damaged USB drive and, interestingly, no crash. Then, with that drive reattached, I provided Andre with an old school DOS utility called "eatmem" which simply consumes some amount of RAM and returns to DOS. Back in the day, this was used to stress-test programs by subjecting them to limited memory situations. But it also has the side effect of changing the location in RAM where DOS will load subsequent programs.

And, sure enough, by "eating" various amounts of memory before running SpinRite, SpinRite's crashing behavior did not occur, or it occurred differently. Around this same time, one of our SpinRite testing participants, Paul Farrer, who also knows how to write DOS programs, was experimenting on his own with a similar crash that he and another user were both seeing. That other user had attached a 1 terabyte USB drive to his machine and it was crashing when he ran SpinRite.

Paul hypothesized that the trouble might be caused by SpinRite attempting to read above the 137GB region of a drive. And that turned out to be correct. 137GB is one of those many size limitations that were constantly plaguing the PC industry during its early growth. Over and over

and over we kept outgrowing every upper limit that we assumed would never be exceeded. The classic story was that the 16-bit Apple II had a maximum of 64 kilobytes of main RAM memory, and when the IBM PC came out, with its initial maximum of 640 kilobytes – exactly 10 times as much as the Apple II – the story was that during a trade show in 1981, Bill Gates said "We'll never need more than 640K." Today, Bill doesn't recall ever having said that. But whether or not he did, although it may now seem ludicrous, I can easily imagine that seeming reasonable at the time. And that's the point, these were always reasonable-seeming limitations because none of us, who were in the middle of this back then, could have foreseen what has happened since.

The early IDE drives had sizes in the hundreds of megabytes or few gigabytes. So the designers of those drives repurposed the addressing bits which had been used by the original cylinders, heads, and sectors registers to scrounge up a grand total of 28 bits that they could use to linearly address the sectors on an IDE drive. This was called LBA for Linear Block Addressing.

The use of 28 bits to address sectors meant that a drive could have at most 2^28 total physical sectors. Since sectors were 512 bytes each, that meant that the maximum size of those 28-bit LBA drives would be 137 gigabytes. But, my goodness! 137 GIGABYTES! No one would EVER be able to create a drive that large, let alone have that much data that needed storage. I mean, come on!, that 137 **billion** with a "B". Right. Whoops.

So, Paul's intuition was that SpinRite's code was somehow being corrupted when it attempted to access sectors at the end of the drive. This is one of the things that SpinRite does when it's sizing up a drive before listing it for use. So Paul and I each independently wrote testing utilities to better understand what was going on. And what we discovered over this past weekend was a bit astonishing.

I ended up writing two new utilities. BIOSTEST first filled all of the system's main RAM memory that wasn't already in use by DOS and the BIOS and buffers and the program itself, with a deterministic pseudo-random pattern. Then it simply used the system's BIOS to read several sectors from the front of the drive and from the end of the drive. After each test read it would rescan all of that memory looking for the first mismatching data. It was looking to see whether reading any sector from a drive through the BIOS would cause the BIOS to alter main memory. And sure enough, after reading the last sectors of larger than 137 gigabyte USB drives on some motherboards, it found main memory mismatches. Reading sectors from the front of drives never caused any problems. But reading any sectors whose linear address had more than 28 bits, would actually damage the data stored in main memory. The USB support in those BIOSes was seriously buggy.

During what we call drive discovery and enumeration, SpinRite goes out to the very end of every drive it can locate to perform reading and writing confirmation and confidence testing. It's safest to do that out there at the end because the very ends of drives are usually empty and don't contain any user data. But when SpinRite was doing that on USB-connected drives on motherboards with these broken USB BIOSes, bugs in the BIOS were blasting SpinRite's code, which was then causing it to crash.

Whereas BIOSTEST checked a few sectors at the front and back of every BIOS drive, BIOSSCAN read every sector of a user-specified BIOS drive. It also filled memory with a pseudo-random

test pattern, then it would scan the entire drive from front to back, while re-scanning main memory once every second looking for any changes. The moment BIOSSCAN crossed the 137 gigabyte boundary and asked for any sector on the other side of that, these defective BIOSes reacted by blasting a region of memory – as much as 64K of RAM. BIOSSCAN went a bit further to locate and report the first and last altered regions. It varied somewhat from one tester to another, but we now had clear incontrovertible proof that Paul Farrer's initial hunch was correct.

Four days earlier, before I knew any of this, at 7:30pm last Wednesday the 22nd, I authored a posting titled *"The disappointing state of BIOS support for bad drives."* Here's what I wrote:

> *Well... It's a good thing that the first big improvement in SpinRite 7.0 will be native support for USB-connected drives, since I've spent a long and frustrating day (which began at 4:30am) learning how poorly our PC BIOSes handle troubled USB drives.*
>
> *I can empathize, since the early SpinRite alphas here had similar problems until it became clear what was going on. We've seen that drives which are in bad shape exhibit all sorts of rule-breaking and degenerative behavior. I haven't yet managed to duplicate Andre's crash, but I have watched many USB-connected drives upset their system's BIOS.*
>
> *BIOS designers can be forgiven, since they would naturally be testing on good drives that work, just as I was until recently. When asked to read an unreadable sector or write to a problem region, BIOSes will often turn off hardware interrupts, stop SpinRite's spinning, kill keyboard service... and then never come back. The system is dead.*
>
> *SpinRite's newest screen, which briefly flashes before displaying the drive discovery screen, was an attempt to hold the BIOS accountable, so that SpinRite's users wouldn't think that SpinRite had hung. But today I've seen that SpinRite v6.1 will be far less able to help users who have badly damaged drives that they connect over USB -- and the MOST frustrating part is that there is literally NOTHING I can do. Today's USB BIOSes just don't handle drive errors. So SpinRite really cannot work through them in the case of badly damaged drives.*

I should clarify that by "badly damaged drives" I mean really just short of dead drives. Not the sort of drive that anyone would actually be using. Or hopefully not. But since one of SpinRite's roles is damaged drive recovery, it needs to be able to deal with drives in any state, and at the moment USB BIOSEs, even those that are not also badly broken at the 28-bit boundary, are standing between SpinRite and USB-connected drives.

So we're now faced with a dilemma. USB BIOSes are buggy and they really cannot be trusted beyond 137 gig. Testing the safety of any given USB BIOS is fraught with problems because the testing code needs to be running in the same memory as what the BIOS might be blasting. And even if a USB BIOS doesn't blast main memory, we still don't know that it properly handles drives past their 137 gigabyte point. For all we know it might allow the 28-bits to overflow and wrap around so that it begins re-testing the front of the drive rather than moving further back.

So here's the upshot of all this: After a great deal of discussion, everyone who has been testing SpinRite agrees that the sooner we get SpinRite 6.1 out the door, the sooner we get to work on SpinRite 7 which will add the same native, non-BIOS support for USB that SpinRite 6.1 now has for SATA and IDE drives through AHCI and ATA interfaces. That will mean no drive size limits, no

trouble dealing with seriously damaged drives and running drives at their full speed, even through USB.

In the meantime, SpinRite v6.1 is going to continue to place the safety of its user's data first and foremost. Now that we know that there are USB BIOSes that misbehave past 137 gigabytes, and also that we cannot reliably detect them, nor what else they might do, SpinRite is going to place a temporary limit on its access to USB-connected drives of 137 gigabytes, because we cannot know that it's safe to go beyond that. It **will** analyze, maintain and repair the first 137 gigabytes of **any** USB drive. But until we get to SpinRite 7, which will have no limits, nothing beyond that is safe over USB.

The best answer is to do what is safe while making this limitation as temporary as possible by getting SpinRite 6.1 published and out the door, then immediately beginning work on SpinRite 7 which will add the same native hardware support for USB host controllers that 6.1 added for directly attached drives. I'm not planning to take any time off once 6.1 is published. I will immediately be starting to work on 7.

Humble Bundle: Think Like a Programmer

A great looking 18-book Humble Bundle to support the EFF:
https://www.humblebundle.com/books/think-like-programmer-no-starch-books

A bunch of programming books by Randall Hyde and a whole lot more.

https://grc.sc/bundle

.

The NSA @ Home

The NSA offers home security "Best Practices" advice
Last Wednesday, the US National Security Agency, our NSA, published an attractive and end-user accessible 9-page PDF loaded with tips for helping to secure a home network environment.

I want to share and comment on some of what the NSA suggested. Our long time listeners will feel right at home, so to speak, with everything that the NSA wrote. They led off with three major key points:

- *Upgrade and update all equipment and software regularly, including routing devices.*
- *Exercise secure habits by backing up your data and disconnecting devices when connections are not needed.*
- *Limit administration to the internal network only.*

And ...

IoT devices on a home network are often overlooked, but also require updates. Enable automatic update functionality when available. If automatic updates are not possible, download and install patches and updates from a trusted vendor on a monthly basis.

It's interesting that they, too, see the threat posed by of out-of-date or defective IoT devices. Of course, the question is, who you gonna call to update some random IoT light switch or wall plug? But moving forward it would be good to see future devices based upon open standards and platforms and for there to be some sort of certification system in place. We have a long way to go but such work is underway.

And this one was interesting:

Your Internet Service Provider (ISP) may provide a modem/router as part of your service contract. To maximize administrative control over the routing and wireless features of your home network, consider using a personally owned routing device that connects to the ISP-provided modem/router. In addition, use modern router features to create a separate wireless network for guests, for network separation from your more trusted and private devices.

That was surprising. Even if your Internet Service Provider offers a modem/router as part of the service package, get your own, that you control and manage, and use it to connect your network to the provider's bandwidth. Again, some sound advice.

And on router guidance, they write:

Your router is the gateway into your home network. Without proper security and patching, it is more likely to be compromised, which can lead to the compromise of other devices on the network as well. To minimize vulnerabilities and improve security, the routing devices on your home network should be updated to the latest patches, preferably through automatic updates. These devices should also be replaced when they reach end-of-life (EOL) for support. This ensures that all devices can continue to be updated and patched as vulnerabilities are discovered.

How many times have we seen companies explaining that they won't be offering updates to fix known critical remote code execution problems for older devices because they are EOL, end-of-life, so anyone still using those devices are SOL. So, yeah. When selecting a router, an important criteria that's easily overlooked is the active and supported service life that has historically been provided by various competing vendors. If this were to become a popular advertised selection criteria, it would put more pressure on vendors to keep older devices supported longer, even though it might mean reduced sales in the future due to the longevity of previous products which were still supported and going strong.

The NSA also talked about WPA3. We briefly touched on this next-generation WiFi 6 and WPA3 encryption standard, but we haven't yet given it a deep dive and it's probably time to do so. It's had a somewhat slow liftoff, since the WiFi Alliance's WPA3 certification process started back in 2018. But WiFi 6 and WPA3-capable devices are here now.

Here's what the NSA wrote:

To keep your wireless communications confidential, ensure your personal or ISP-provided WAP is capable of Wi-Fi Protected Access 3 (WPA3). If you have devices on your network that do not support WPA3, you can select WPA2/3 instead. This allows newer devices to use the more secure method while still allowing older devices to connect to the network over WPA2.

When configuring WPA3 or WPA2/3, use a strong passphrase with a minimum length of twenty characters. When available, protected management frames should also be enabled for added security. Most computers and mobile devices now support WPA3 or WPA2. If you are planning to purchase a new device, ensure it is WPA3-Personal certified. Change the default service set identifier (SSID) to something unique. Do not hide the SSID as this adds no additional security to the wireless network and may cause compatibility issues.

We'll do a WiFi 6 podcast soon. Seeing this next one raised my eyebrows since everyone knows that I worry about the day a widely used IoT device goes rogue. The NSA wrote:

Implement wireless network segmentation
Leverage network segmentation on your home network to keep your wireless communication secure. At a minimum, your wireless network should be segmented between your primary Wi-Fi, guest Wi-Fi, ***and IoT network****.* [Wow! Way to go, NSA!] *This segmentation keeps less secure devices from directly communicating with your more secure devices.*

Yep. As we know, I've been promoting multi-NIC routers which are able to do that for some time. And some more recent WiFi routers are offering stronger segmentation options as well.

What about the presence of personal assistant technologies and worries over eavesdropping? Not surprisingly, the NSA is not a big fan of things with microphones. They wrote:

> *Be aware that home assistants and smart devices have microphones and are listening to conversations, even when you are not actively engaging with the device. If compromised, the adversary can eavesdrop on conversations. Limit sensitive conversations when you are near baby monitors, audio recording toys, home assistants, and smart devices. Consider muting their microphones when not in use. For devices with cameras (e.g., laptops, monitoring devices and toys) cover cameras when you are not using them. Disconnect Internet access if a device is not commonly used, but be sure to update it when you do use it.*

All of that advice falls nicely under the umbrella of generally sound, if maybe a little paranoid, security advice. And following that, under the topic of general security hygiene, they add:

> *To minimize ransomware risks, back up data on external drives or portable media. Disconnect and securely store external storage when not in use. Minimize charging mobile devices with computers; use the power adapter instead. Avoid connecting devices to public charging stations. Leave computers in sleep mode to enable downloading and installing updates automatically. Regularly reboot computers to apply the updates. Turn off devices or disconnect their Internet connections when they will not be used for an extended time, such as when going on vacation.*

In other words... "Think Security" at all times and try to never take it for granted. It's sort of the broader equivalent of what has happened to eMail, where it's no longer ever safe to assume that all eMail is legitimate and that links can be clicked on without careful scrutiny. It's a sad state.

And everyone knows that I'll love this one:

> ***Limit administration to the internal network only***
> *Disable the ability to perform remote administration on the routing device. Only make network configuration changes from within your internal network. Disable Universal Plug-n-Play (UPnP). These measures help close holes that may enable an attacker to compromise your network.*

And there was one piece of advice that makes sense... but that I have never recommended:

> ***Schedule frequent device reboots***
> *To minimize the threat of non-persistent malicious code on your personally owned device, reboot the device periodically. Malicious implants have been reported to infect home routers without persistence. At a minimum, you should schedule weekly reboots of your routing device, smartphones, and computers. Regular reboots help to remove implants and ensure security.*

What's interesting about this, is that many forms of malware are ram resident only. They never write anything to non-volatile media. Since routers are almost never rebooted, malware authors probably figure that there's no reason to write to non-volatile memory. And we know that in well-protected environments writing to disk can trip all sorts of monitoring alarms. And some malware might **want** to disappear after a reboot so that it's larger network of devices can remain hidden. So, if it's not necessary to survive a reboot, malware might well choose not to.

Consequently, indeed, a reboot will permanently flush ram-based malware from the system. Now, if the way such malware got into the system in the first place to obtain its foothold in RAM is not also resolved, it might not be long before it returns. But, yeah, reboots are inherently cleansing. That's a great point.

Okay.

So those were just **some** of the highlights that I thought were the more interesting, surprising or insightful. But there's **much** more in that 9-page document than what I've shared, and the entire document is so good that I think everyone listening would benefit from at least scanning it and probably also by recommending it to others. It has the additional pedigree of bearing the official seal of the National Security Agency which might help everyone's non-Security Now! Listening friends to sit up and take it seriously. And, as we've seen, it's far from being the typical useless piece of say-nothing bureaucratic crap.

As friendly and useful as the document is, its line and a half wrap-around URL is not nearly as friendly. So it's this week's GRC shortcut of the week. So you can find it at https://grc.sc/912

https://media.defense.gov/2023/Feb/22/2003165170/-1/-1/0/CSI_BEST_PRACTICES_FOR_SECURING_YOUR_HOME_NETWORK.PDF

And big props to the NSA for assembling something so useful and largely actionable.

If nothing else, the nature of the recommendations would help someone who doesn't live in the security realm to realize the way security conscious professionals think. And that would probably be surprising to many.