Security Now! #763 - 04-21-20

The COVID Effect

This week on Security Now!

This week, as an interesting case study, we continue tracking the latest actions being taken by Zoom and another unfortunate consequence of their overnight success. We have two pieces of Chrome browser news, and security news including what happened with last Tuesday's Windows patch, rollbacks in authentication plans, Signal's reaction to the planned EARN IT Act, trouble at the Tor Project and an interesting CAPTCHA change at Cloudflare. I also want to share my recent change in preferred VM systems, two bits of listener's closing the loop feedback, and a SpinRite update -- since stuff's beginning to happen.

Wow! Everyone's favorite Polar Bear accepted a job at Microsoft!

https://twitter.com/SandboxBear/status/1210133985478791171

A couple more Zoom follow-ups

I think it's quite instructive from a security viewpoint to watch how Zoom deals with all of the various consequences of the overnight explosion in the popularity of their 9 year old platform. If their response was lame, it would not be interesting. But so far they have been anything but lame. Their response has been quite impressive. And the past week brings us even more impressive news.

Zoom's next steps for enhanced security

Last week, Zoom announced that they had hired Luta Security, which was founded by Katie Moussouris, a well-known cyber-security veteran. Katie created some of the most important vulnerability programs still running today. She started Microsoft Vulnerability Research and Symantec Vulnerability Research, and the bug bounty programs for Microsoft and the Pentagon.

Katie has been given a free hand to rebuild Zoom's existing security program which had been based upon HackerOne. So she's now taking input from across the entire cyber-security community, seeking ways to improve Zoom's vulnerability disclosure process.

In her own posting about the new assignment, Katie wrote:

> No company can bug bounty their way to being secure, and we at Luta Security emphasize building strong internal engineering to reduce the number and severity of vulnerabilities BEFORE software is released, as well as being capable of fixing bugs efficiently when they slip through secure development practices. When the pandemic hit we were wrapping up a full internal vulnerability coordination and management maturity assessment against ISO 30111 with Zoom.
>
> So that's where the real change has to happen – internally. One of the five capability areas Luta Security measures is Organizational. This is the executive will to change company culture, which we are all fortunate enough to witness with Zoom in real time. It's putting effort into investing properly in security and privacy, not just with words, not just by bringing in big names in security, or jacking up bug bounty prices in a frenzy to create the appearance of diligence.
>
> That being said, increased transparency from the many experts who are working together with Zoom to bring the very best security help gives the folks watching this effort a chance to see changes unfold. In cases like Luta Security's work with Zoom, we now get to ask the public for feedback on Zoom's bug bounties.

So who were these big names in security Katie was referring to? Her follow-on tweet last Thursday read:

> I'm excited to highlight my colleagues who are adding their expertise in the next few weeks. In addition to welcoming my former colleague @alexstamos to the extended Zoom security family I'd like to welcome @LeaKissner, @matthew_d_green, @bishopfox, @NCCGroupInfosec and @trailofbits.
>
> — Katie Moussouris (@k8em0) April 16, 2020

Lea Kissner was former Global Lead of Privacy Technology at Google. We all know cryptographer and Johns Hopkins professor Matthew Green. And then there are the three well-known security auditing firms: BishopFox, the NCC Group, and Trail of Bits. So perhaps some auditing is in the works as well.

To maintain a high degree of operating transparency Zoom's CEO, Eric Yuan, has started hosting a weekly "Ask Eric Anything" webinar. During last week's webinar we learned:

That account owners and admins can now configure minimum meeting password requirements to include numbers, letters, and special characters, or allow only numeric passwords. Free Basic account users will now use alphanumeric passwords by default instead of numeric passwords.

And since last Saturday, account admins now have the ability to choose to have their data routed through specific data center regions, giving them control of their interactions with Zoom's global network. This feature is intended to help with fears that Zoom chats and encryption keys might be sent to Chinese servers.

And Sunday the system added the ability to report abusive users so that Zoom can shut down accounts engaging in Zoom-bombing.

Alex Stamos also spoke during the same webinar and announced that "in a matter of weeks" they'll be moving from Zoom's current barely adequate call encryption to a more widely tested and trusted solution. Specifically, Alex said that Zoom will be moving away from the current AES-256 ECB encryption to a more secure AES-256 GCM encryption (which is what SQRL uses to encrypt its identities). But Alex also noted that "the long-term focus will involve a totally new cryptographic design that greatly reduces risk to Zoom's system."

So far, Zoom's response looks to me like a business management school case study in the proper way to manage the explosive growth of a highly used, highly targeted and abuse-prone online facility. So again, to Zoom I say "Bravo."

Meanwhile, >500,000 Zoom meeting IDs and passwords are for sale

And for just pennies (actually apparently 1/10th of a penny) each. To no one's surprise, a black market for Zoom meeting IDs and passwords has quickly sprung into existence. Since the price is so low, we can assume that the seller knows that there's not much value there. The credentials are gathered through so-called credential stuffing attacks where bad guys attempt to login to Zoom using lists of common passwords and passwords found in previous data breaches. The successful logins are then compiled into lists that are sold to other hackers. So mostly this is just another indication of Zoom's overnight celebrity.

But, on the other end of the pay scale...

But, on the other end of the pay scale...
Motherboard reports that people who trade in 0-day exploits say there are two Zoom 0-days, one for Windows and one for MacOS, currently on the market at an asking price of $500,000!

Talk about Coronavirus driven extortion! $15 for a roll of toilet paper and half a million dollars for a Zoom 0-day. A few months ago a hacker would have been happy to get five grand for those 0-days! But now?!?!

So here's what we know: Hackers are selling two critical vulnerabilities for Zoom which would allow someone to hack users and eavesdrop on their calls. According to three independent Motherboard sources who are knowledgeable about the market for these kinds of hacks one each reportedly exists for the Zoom client for Windows and another for the Zoom macOS client. The sources had not seen the code for these vulnerabilities, but have been contacted by brokers offering them for sale.

Motherboard had previously reported that there had been a sudden increase in interest in 0-days for Zoom after hundreds of millions of people, including employees and executives at big companies around the world, had moved to the platform for sensitive or confidential meetings.

Adriel Desautels, the founder of Netragard, a company that used to sell and trade 0-days, said: "From what I've heard, there are two 0-day exploits in circulation for Zoom. One affects macOS and the other Windows." He added: "I don't expect that these will have a particularly long shelf-life because when a zero-day gets used it gets discovered."

Two other independent sources, who asked to remain anonymous to discuss sensitive topics, confirmed the existence of these two exploits on the market. One of the sources, a veteran of the cybersecurity industry, said: "[The Windows zero-day] is nice, a clean RCE [Remote Code Execution]. Perfect for industrial espionage."

The zero-day for Zoom on Windows would allow hackers to access the app, but would need to be coupled with another bug to access the whole machine. The MacOS flaw is not an RCE, according to the two anonymous sources.

And, as I noted at the start, the asking price for the 0-day for the Zoom Windows app is $500,000, according to one of the sources, who deals with the procurement of exploits but has decided not to purchase this one. The source said the exploit requires the hacker to be in a call with the target, making it less valuable for a government spy agency that aims to be stealthy and doesn't want to get caught. This guy sounds like a vulnerability reseller, like Zerodium. He also told Motherboard that he estimated that the exploit was worth about half the asking price.

And the MacOS bug is not an RCE, making it less dangerous and harder to use in a real hack, according to the two anonymous sources.

And as for Zoom, especially in their current posture, you can imagine they are not taking this lying down. When asked they said: "Zoom takes user security extremely seriously. Since learning of these rumors, we have been working around the clock with a reputable, industry-leading security firm to investigate them. To date, we have not found any evidence substantiating these claims."

But then, of course, that's what makes a 0-day a 0-day.

Browser News

We have two bits of web browser news this week, both for Chrome:
Google updated their Windows, Mac, and Linux editions of Chrome to v81.0.4044.113 to squash a critical flaw that existed in earlier versions. That's both the short and the long version of the story, since Google is not yet saying anything more. They wrote: "Access to bug details and links may be kept restricted until a majority of users are updated with a fix. We will also retain restrictions if the bug exists in a third party library that other projects similarly depend on, but haven't yet fixed."

What little we do know is that Google was made aware of the problem by researchers at Qihoo 360's Alpha Lab and that it addresses the Critical CVE-2020-6457 which is described as a use after free flaw in Chrome's speech recognizer. We don't know whether you needed to be spending a lot of time talking to your browser, or whether a malicious website might have been able to leverage the flaw on any age you visited.

When I looked, that's the version of Chrome I was already using… and something else turned out to be different than planned in Chrome 81...

Chrome "undeprecates" FTP!
For the past 6 years, ever since 2014, Google has been wanting to eliminate support for the FTP protocol in Chrome since it was only used by only by between 0.1-0.2% of the browser's users (slightly higher among Linux users) and because there are quite acceptable an far more feature-embellished FTP client's readily available.

By the end of 2018, Google had moved FTP to "WONTFIX" in Chrome for iOS and began the march to slowly deprecate FTP support in their desktop browsers as well.

We talked about this at the time, with Google writing: "FTP is a non-securable, legacy protocol. We've WONTFIXed FTP support on iOS, but its usage in Blink-based Chrome is high-enough that it seems difficult to remove it all at once. This seems like a reasonable way of reducing its viability as an attack surface as a stepping stone to more complete removal."

What they were referring to was the plan to continue displaying FTP directory listings, but no longer render files in the browser, only allowing users to download them.

Then, with Chrome 80, Google added an "Enable FTP" flag (chrome://flags/#enable-ftp) to control whether or not =ANY= FTP support would be present. It was still enabled by default, but Google used the flag to conduct a test where it is turned off for 1% of its user base to see whether anyone noticed and complained. The plan was finally to disable FTP support by default in Chrome 81, but still allow it to be enabled again using the #enable-ftp flag.

But then, COVID-19. So, on April 9th, Google software engineer Asanka Herath posted to the "Remove built-in support for FTP from Chrome" Chromium bug topic that "In light of the current crisis, we are going to "undeprecate" FTP on the Chrome stable channel. So, in other words, FTP will start working again."

As with support for TLS v1.0 and v1.1, FTP support is being restored by default to make sure there no problem with people accessing content on FTP sites during the pandemic. For instance, many government agencies still utilize FTP sites, including the National Institutes of Health (the NIH). Asanka Herath stated that this momentary reversal of FTP's deprecation would endure until things were back to normal and people were in a better position to deal with potential outages and migrations.

Security News

Microsoft Patch Tuesday

It's a good thing that all Windows 10 users will have, by now, installed this month's Patch Tuesday updates and rebooted, because everyone who did eliminated 113 notable flaws, 19 of which were rated critical, in Windows and Microsoft's related software. And that crop of 113 problems includes three serious 0-day flaws that Microsoft was aware of being actively exploited in the wild.

The uncomfortable news for those of us who have chosen, for whatever reason, to continue using Windows 7 is that these three 0-days and many other problems are also present in our beloved Windows 7, but, as we know, without extended security updates those flaws will remain unpatched.

Two of the 0-days found to be exploited in the wild were in Adobe's Type Library. We were already aware of one of these and we've talked about temporarily disabling, renaming or deleting the offending Windows DLL. The problem with those rather heavy-handed moves was that other somewhat important pieces of Windows, like Media Player, were dependent upon the offending DLL. An alternative for Windows 7 users was to consider subscribing to the http://0patch.com MicroPatch service to have their fixes deployed on the fly.

The 3rd 0-day was a Windows Kernel Elevation of Privilege Vulnerability that, being a 0-day, was found by Google's Project Zero being used in the wild. This one only rates as "Important", presumably since it doesn't allow for remote code execution. But as we know, privilege of elevation vulnerabilities can be combined with other flaws to create a much more powerful attack. So it ought not be ignored.

There's also some question about whether there might actually be a 4th 0-day vulnerability fixed this month, but it appears that the advisory for a critical Internet Explorer flaw (CVE-2020-0968) was revised to indicate that Microsoft has not yet received reports of it being exploited in the wild. Nevertheless, the advisory says this IE bug is likely to be exploited soon and that would not be good since it's a scripting engine memory corruption vulnerability in IE. Microsoft explains that a remote code execution vulnerability exists in the way that the scripting engine handles objects in memory in Internet Explorer. The vulnerability could corrupt memory in such a way that an attacker could execute arbitrary code in the context of the current user. On the other hand... IE.

Also of note is CVE-2020-0796. If that number rings a bell it's because it's the CVE for the "Windows SMBv3 Client/Server Remote Code Execution Vulnerability." Recall that this one really had Microsoft freaked out and worried recently, calling it a "wormable" pre-auth remote code execution vulnerability. Microsoft released emergency out-of-cycle patches in the middle of last month because they were unable to wait until this month's very late in the month patch Tuesday after a handful of public Proofs of Concept appeared. And although none of those had successfully performed the dreaded Remote Code Execution, yesterday, researchers at Ricerca Security demonstrated exactly that. So it's a good thing that it's been fixed.

In more "We changed our minds because of COVID-19" news...
Known as "Basic Authentication" and originally specified by RFC 2617, published in June of 1999, Basic Authentication is a dead simple method of providing HTTP authentication for a online web app or a web browser page by simply adding the HTTP query header "Authorization:" followed by the one-word token "Basic" and a Base64-encoded username and password, where the username and password are concatenated by a colon (:).

It is simple and convenient because, rather than any sort of back-and-forth handshaking or sticky state created by cookies, the query simply contains and asserts the application's credentials. In a single query it simply says: "This is who I am and this is what I want." Once upon a time, without HTTPS, doing this would have been horribly insecure since there's no cryptographic challenge/response, nor even any encryption of the credentials. Just a static assertion of the requester's identity.

Today, 21 years later, there are myriad ways of accomplishing the same goal with FAR more security, with Oauth 2.0 being the industry's current favorite. So the use of this seriously old school Basic Authentication has been falling by the wayside for some time.

Way back in July of 2018 Microsoft announced that it would be switching off Basic Authentication in its Exchange Web Services (EWS) API for Office 365, and it planned to turn off support for the feature entirely on 13 October 2021. Google also committed to turning off Basic Authentication. In December 2019, Google warned that it would deny what it called 'less secure apps' access to

its back end services, favouring OAuth 2.0 instead. That was meant to happen two months from now on June 15th, 2020.

But then COVID-19 happened and now **both** companies have reconsidered and changed their plans in the interest of giving their online users more leeway as they cope with the COVID-19 chaos. Earlier this month, on April 3rd, Microsoft said that it would postpone the deprecation of Basic Authentication for Exchange Online for those tenants using it, keeping it available until the second half of 2021. And Google also announced at the end of March that it will defer its Basic Authentication switch-off until further notice.

But anyone still using Basic Authentication really should be reading the handwriting on the wall and be moving to Oauth 2.0, which is what both companies now recommend for their preferred authentication flows.

We need to look at the EARN IT Act again.
I think I used the word "despicable" when we first introduced this horrific pending legislation to our podcast audience. It's despicable because it is employing an underhanded backdoor means for attempting to force encryption companies to alter their technologies to make them "subpoena compatible."

Joshua Lund is a developer and spokesman for Signal whom we've quoted in the past. Signal is, as we know, arguably the best and most securely designed end-to-end encrypted messaging solution. And, not surprisingly, it's not only Zoom that has recently experienced a massive pandemic-driven adoption rush. In this context, Josh has recently spoken out about this "EARN IT" act. I want to share Signal's position:

https://signal.org/blog/earn-it/

Over the past several weeks, Signal traffic has gone through the roof. New users are signing up at unprecedented rates, and we've expanded our server capacity faster than we ever anticipated.

It means a lot to us that so many people are relying on Signal during this difficult time. When users check in on their families, share moments of solace, smile with their friends, or discuss sensitive health issues with their doctors, Signal's end-to-end encryption and privacy-preserving technology helps keep this information secure.

At a time when more people than ever are benefiting from these protections, the EARN IT bill proposed by the Senate Judiciary Committee threatens to put them at risk. COVID-19 has us sheltering in place, but we cannot quarantine our concerns.

Broadly speaking, Section 230 of the Communications Decency Act protects online platforms in the United States from legal liability for the behavior of their users. In the absence of this protection, many of the apps and services that are critical to the way the internet functions today may have never been created in the first place – or they couldn't have been created in America.

The EARN IT act turns Section 230 protection into a hypocritical bargaining chip. At a high level, what the bill proposes is a system where companies have to earn Section 230 protection by following a set of designed-by-committee "best practices" that are extraordinarily unlikely to allow end-to-end encryption. Anyone who doesn't comply with these recommendations will lose their Section 230 protection.

Some large tech behemoths could hypothetically shoulder the enormous financial burden of handling hundreds of new lawsuits if they suddenly became responsible for the random things their users say, but it would not be possible for a small nonprofit like Signal to continue to operate within the United States. Tech companies and organizations may be forced to relocate, and new startups may choose to begin in other countries instead.

For a political body that devotes a lot of attention to national security, the implicit threat of revoking Section 230 protection from organizations that implement end-to-end encryption is both troubling and confusing. Signal is recommended by the United States military. It is routinely used by senators and their staff. American allies in the EU Commission are Signal users too. End-to-end encryption is fundamental to the safety, security, and privacy of conversations worldwide.

Proponents of this bill are quick to claim that end-to-end encryption isn't the target. These arguments are disingenuous both because of the way that the bill is structured and the people who are involved.

Riana Pfefferkorn, Associate Director of Surveillance and Cybersecurity at the Stanford Center for Internet and Society, wrote a detailed breakdown of some of the myriad problems with this bill. She also astutely points out that the bill would give unprecedented power to Attorney General William Barr, a vocal critic of end-to-end encryption, who would become the arbiter of any recommendations from the "best practices" commission that the EARN IT bill would create.

It is as though the Big Bad Wolf, after years of unsuccessfully trying to blow the brick house down, has instead introduced a legal framework that allows him to hold the three little pigs criminally responsible for being delicious and destroy the house anyway. When he is asked about this behavior, the Big Bad Wolf can credibly claim that nothing in the bill mentions "huffing" or "puffing" or "the application of forceful breath to a brick-based domicile" at all, but the end goal is still pretty clear to any outside observer.

[And Josh concludes with an "It's not too late" message...]

As billions of conversations transition online over the coming weeks and months, the widespread adoption of end-to-end encryption has never been more vital to national security and to the privacy of citizens in countries around the world.

Bad people will always be motivated to go the extra mile to do bad things. If easy-to-use software like Signal somehow became inaccessible, the security of millions of Americans (including elected officials and members of the armed forces) would be negatively affected. Meanwhile, criminals would just continue to use widely available (but less convenient) software to jump through hoops and keep having encrypted conversations.

There is still time to make your voice heard. We encourage US citizens to reach out to their elected officials and express their opposition to the EARN IT bill. You can find contact information for your representatives using The Electronic Frontier Foundation's Action Center.

Stay safe. Stay inside. Stay encrypted.

https://act.eff.org/action/protect-our-speech-and-security-online-reject-the-graham-blumenthal-bill

https://grc.sc/earnit

COVID-19 hits the Tor Project hard
The Tor Project is 100% donation supported and, understandably, with so many past supporters worried about the shape of the future, the flow of donations has dwindled since the start of the year. Consequently, their recent staff of 35 people has just been reduced to 22 after a 13-person layoff. They were sorry to lose anyone, since those lost were valuable contributors. But there was just no money to pay them.

The project states that with their reduced overhead they should be okay now. But if you've been a user of Tor, and/or want to help keep the lights on and the servers spun up, it might be worthwhile to drop them some monetary support if you're in a position to do so.

Cloudflare switches CAPTCHA providers
https://blog.cloudflare.com/moving-from-recaptcha-to-hcaptcha/

Matthew Prince:

Since Cloudflare's earliest days, we have used Google's reCAPTCHA service. ReCAPTCHA started as a research project out of Carnegie Mellon University in 2007. Google acquired the project in 2009, around the same time that Cloudflare was first getting started. Google provided reCAPTCHA for free in exchange for data from the service being used to train its visual identification systems. When we were looking for a CAPTCHA for Cloudflare, we chose reCAPTCHA because it was effective, could scale, and was offered for free — which was important since so many of Cloudflare's customers use our free service.
Privacy and Blocking Concerns

Since those early days, some customers have expressed concerns about using a Google service to serve CAPTCHAs. Google's business is targeting users with advertising. Cloudflare's is not. We have strict privacy commitments. We were able to get comfortable with the Privacy Policy around reCAPTCHA, but understood why some of our customers were concerned about feeding more data to Google.

We also had issues in some regions, such as China, where Google's services are intermittently blocked. China alone accounts for 25 percent of all Internet users. Given that some subset of those could not access Cloudflare's customers if they triggered a CAPTCHA was always concerning to us.

Over the years, the privacy and blocking concerns were enough to cause us to think about switching from reCAPTCHA. But, like most technology companies, it was difficult to prioritize removing something that was largely working instead of brand new features and functionality for our customers.

Google's Changing Business Model

Earlier this year, Google informed us that they were going to begin charging for reCAPTCHA. That is entirely within their right. Cloudflare, given our volume, no doubt imposed significant costs on the reCAPTCHA service, even for Google.

Again, this is entirely rational for Google. If the value of the image classification training did not exceed those costs, it makes perfect sense for Google to ask for payment for the service they provide. In our case, that would have added millions of dollars in annual costs just to continue to use reCAPTCHA for our free users. That was finally enough of an impetus for us to look for a better alternative.

A Better CAPTCHA

We evaluated a number of CAPTCHA vendors as well as building a system ourselves. In the end, hCaptcha emerged as the best alternative to reCAPTCHA. We liked a number of things about the hCaptcha solutions: 1) they don't sell personal data; they collect only minimum necessary personal data, they are transparent in describing the info they collect and how they use and/or disclose it, and they agreed to only use such data to provide the hCaptcha service to Cloudflare; 2) performance (both in speed and solve rates) was as good as or better than expected during our A/B testing; 3) it has a robust solution for visually impaired and other users with accessibility challenges; 4) it supported Privacy Pass to reduce the frequency of CAPTCHAs; 5) it worked in regions where Google was blocked; and 6) the hCaptcha team was nimble and responsive in a way that was refreshing.

The standard hCaptcha business model was similar to how reCAPTCHA started. They planned to charge customers that needed image classification data and pay publishers to install their CAPTCHA on their sites. Sounded great to us, but, unfortunately, while that may work well for most publishers, it doesn't at Cloudflare's scale.

We worked with hCaptcha in two ways. First, we are in the process of leveraging our Workers platform to bear much of the technical load of the CAPTCHAs and, in doing so, reduce their costs. And, second, we proposed that rather than them paying us we pay them. This ensured they had the resources to scale their service to meet our needs. While that has imposed some additional costs, those costs were a fraction of what reCAPTCHA would have. And, in exchange, we have a much more flexible CAPTCHA platform and a much more responsive team.

Miscellany

"STUNNED by VirtualBox"

The more I use this incredible FREE tool the more amazed I am.

For my work on SpinRite I had built and fine-tuned an optimal FreeDOS-based DOS system with Windows file sharing so that the DOS machine could see the SpinRite development directory on my workstation. It also had a font replacement for improved 50-line text fonts, my favorite non-protected mode debugger and various other utilities, and a boot menu so that I could boot into different configurations. This perfect DOS machine was operating in a VirtualBox VM. So now I wanted to create an image of the drive so that I could clone it into several physical systems for additional testing on real hardware.

To do that I wanted to use my favorite bootable imaging tool (Terabyte's Drive Image). So I plugged a DriveImage bootable USB thumb drive into the Win7 host machine. I opened Windows Disk Manager and looked up the physical drive number that had been assigned to the USB drive. In this case it was #5.

I then opened a command prompt and switched to the "VirtualBox" directory so that I could conveniently use the "VBoxManage" command line utility. This invocation:

> VBoxManage internalcommands createrawvmdk -filename "%USERPROFILE%"\.VirtualBox\usb.vmdk -rawdisk \\.\PhysicalDrive5

... created a mountable "usb.vmdk" virtual virtual disc from the physical USB drive plugged into the Windows box. So I added that VMDK to the Virtual Machine I wanted to image and moved it into the Primary Master position so that it would have boot precedence. I also added another virtual disk to receive the image. When I rebooted the VM the "Drive Image" main menu popped up and it all worked perfectly.

I was able to create a drive image of the working DOS VM, which was stored onto another virtual disc. I removed the temporary drives and dismounted everything. Then I copied the newly created drive image onto the USB to create a self-contained bootable "setup a DOS machine" thumb drive. And using that I cloned my working SpinDEV image to multiple laptops.

So far... ANYTHING I have imagined that I wanted to do with VirtualBox has been possible.

Upon seeing that, a long time contributor in the forums, Greg Bell, replied:

15 years ago, then again 10 years ago, I tried all the various VM technologies and settled on VMWare Workstation as being head and shoulders above the rest for pretty much everything. And I've used it all this time.

Recently I had a need to revisit VirtualBox so I could collaborate with someone else. And *man* have they come a long way. Like, *everything* is possible if you know the right VboxManage spell. Things I have to wrestle VMware to the ground over, vbox is like "Sure! Want a hundred other options too? Command lines for everyone!"

Wait'll you start getting into USB device attaching.

So... I wanted to take a moment to share my utter amazement over Oracle's VirtualBox. I, too, have always been using VMware, assuming that it was the best. But now, I think that the best is free and it's name is "VirtualBox": https://www.virtualbox.org/

Closing the Loop

Geoff Clow (@geoff_clow)
@SGgrc Steve - FF with TST - do you disable the FF built-in horizontal tabs? If so, how?

Lee Hadassin (@leehadassin)
@SGgrc Hey Steve, Listened to SN#762 - I think I see Moxies P.O.V. - privacy is there until you test positive. If this is a framework for 3rd party apps and they get the diagnosis keys - what is stopping them from correlating the time/location to possibly identify who it was?

SpinRite

I'm pleased to report that SpinRite's technology for talking to drives through AHCI controllers is starting to work. I have successfully performed an "Identity Device" command on both a VirtualBox VM and on a Dell laptop. I chose the very simple 512-byte "Identify Device" query specifically because it required the least possible from the drive and because it was probably the easiest thing to get working. The logic here is that getting ANYTHING at all to work with an AHCI controller requires so much hardware to be exactly correctly set up that pretty much everything has to be working correctly for ANYTHING to work at all. So when I saw that I had captured the drive's identity information into RAM, I knew that now it was just a matter of refinement.

Until now, with SpinRite using the BIOS for its bulk data transfer, it has had the advantage of the BIOS to provide a compatibility interface to the motherboard's hardware. That's the big thing that's changing now that SpinRite will be bypassing the BIOS for all mass storage access.

So, my plan is to refine this first utility, which I call "SpinTest", since it's intended only to be a development and proving ground for SpinRite's new technologies. SpinTest will wind up being a complete, robust mass storage device enumeration utility. As it develops I will run it on every piece of hardware I own--and I own plenty since I never throw anything away. Then, once it's working on everything I own and I've tested it as much as I'm able, I'll turn it loose over in the grc.spinrite.dev newsgroup for everyone there to use on all of their many pieces of old and new hardware.

I have no idea what will happen, because I've seen reports of some very odd, finicky and non-standard hardware out in the field. But wherever SpinTest initially fails, and I'm sure it will, I will get it working on everything it encounters. And this is a useful investment since this new technology will be the basis for "Beyond Recall" and SpinRite 7.

Back in 2013, before I put this SpinRite work on hold, I had already written drivers for the AHCI controller when it was in Legacy or Compatibility mode, and also for older IDE motherboard

controllers. That's how I was able to perform those benchmarks which showed, back then, SpinRite reading 65,536 sectors at a time into a single 32 megabyte buffer and running at half a terabyte per hour. I currently have that code turned off to focus upon AHCI. But once I've got native AHCI working with SATA drives I'll turn that other code back on and we'll have ALL system drives, old and new, accessible to SpinTest without any BIOS involvement.

And next week, unless something happens to derail my plan, I want to take our listeners on a deep dive into a very cool new service recently spun up by Cloudflare. It allows us all to test our own ISPs BGP routing security in an effort to promote the adoption of an effective security framework known as RPKI - Resource Public Key Infrastructure. Stay tuned. :)