Security Now! #963 - 02-27-24

Web portal? Yes please!

This week on Security Now!

What US state is now trying to ban encryption for minors? What shocking truth did a recent survey of IT professionals reveal? What experimental feature from Edge is Chrome inheriting? Are online services really selling our private data? And what about browser add-ons? Should we be paying extra to obtain cloud security logs? Now that the dust has settled, what happened with LockBit? What new features just appeared in Firefox v123? And what lesson have we just received another horrific example of? I have news on the GRC software front, and we have a bunch of interesting feedback from our terrific podcast listeners. So another jam-packed episode of Security Now.

Security News

Nevada attempts to block Meta’s end-to-end encryption for minors.

Kim Zetter’s Zero Day blog had the best coverage I’ve seen of this surprisingly aggressive move. I’ve edited what Kim wrote for length and readability. So the news is:

> *Nevada's attorney general filed a motion this week to prevent Meta from providing end-to-end encryption to users under eighteen who reside in the state.*
>
> *The request explains that its intention is to combat predators who target minors for sexual exploitation and other criminal purposes, and to allow law enforcement to retrieve communication between criminals and minors from Meta's servers during investigations.*
>
> *Last Tuesday, AG lawyers filed a partially redacted brief in Las Vegas federal court seeking a temporary restraining order and a preliminary injunction against Meta to prevent it from offering end-to-end encryption on Messenger for anyone residing in the state whom Meta believes may be a minor. In its request to the court, the Nevada AG's office claims that Meta's decision to enable end-to-end encryption by default is “irresponsible” and “drastically impedes law enforcement efforts to protect children from heinous online crimes, including human trafficking, predation, and other forms of dangerous exploitation.”*
>
> *The AG requested an immediate hearing on the matter two days later, which would have been last Thursday citing the “extreme urgency” affecting “the safety and well-being” of children in Nevada who use Messenger. But the court has scheduled it for yesterday, February 26.*
>
> *In its response to the filing, Meta said that the request makes no sense since it and other messaging services have been offering end-to-end encryption to minors and other users for years, and law enforcement, as acknowledged in the Nevada AG's own filing, can still obtain such messages from the devices used by criminals and minors.*
>
> *Meta wrote: “The State cannot properly assert that it requires emergency injunctive relief—on two days’ notice—blocking Meta’s use of E2EE, when that feature has been in use on Messenger for years and began to be rolled out for all messages more than two months ago," Meta writes in its response.*
>
> *A legal expert and research scholar at the Stanford Internet Observatory, calls Nevada's request “bizarrely aggressive” and says the timing of it is perplexing, writing: “It seems to come out of nowhere and what’s the motivation for this to happen now?” This expert cited it as being the biggest attack on encryption in the U.S. since 2016, which was a reference to the FBI's attempt to force Apple to undermine the encryption on its iPhones so the agency could access a phone used by the suspect in the San Bernardino terrorism case. As we recall, the FBI wound up gaining access through another means.*
>
> *Meta has made end-to-end encryption available to Messenger users since 2016, but last December, the company promoted it to the default setting for all Messenger communication, it being the application used for private messaging between users on Facebook and Instagram.*

As we know, law enforcement investigators can still read the messages, even if they were encrypted in flight, if they obtain the device used by either party to the communication and are able to access the device with the password or bypassing it with forensic tools. This has been true since 2016 when any user, including minors, opted to enable end-to-end encryption. The only thing that has changed recently, is that Meta is now encrypting all messages by default.

But Nevada's Attorney General appears to be asking the court not just to prevent Meta from enabling end-to-end encryption for minors by default, but also to prevent the company from providing the option to use end-to-end encryption at all for minors who reside in the state – even though they've been able to use end-to-end encryption for eight years.

In its response opposing the request for a restraining order and injunction, Meta points out that end-to-end encryption has been available by default for Apple iMessages since 2011, and is also available to users of the Signal communications app and other similar applications. End-to-end encryption has been considered essential for protecting communications for years, it notes. Meta wrote: "Indeed, Nevada law recognizes the value of encryption, requiring data collectors to encrypt personal information."

The Standard Observatory expert noted that if the court were to grant the restraining order and injunction it would actually make minors less secure than other users of Messenger, writing: "It's bizarre for the state to be saying that the AG wants to ensure that only children in Nevada receive less privacy and security protection than any other user of Messenger."

And there's the danger that this could set a precedent with other states following suit.

As the basis for its request to obtain a restraining order, the AG's office claims in its filing that in providing end-to-end encryption for minors, Meta is violating Nevada's Unfair and Deceptive Trade Practices Act, which prohibits the violation of laws in the course of selling or leasing goods or services. Nevada law prohibits the use of encryption to commit a criminal offense or conceal a criminal offense or obstruct law enforcement, the AG states, therefore Meta is directly and indirectly aiding and abetting child predators by providing them with end-to-end encryption. The AG also states that Meta further violates the Unfair and Deceptive Trade Practices Act by misstating the risks minors face in using Messenger.

The Attorney General states that Meta presents Messenger as a safe application for minors to use, but fails to inform them that in using Messenger with end-to-end encryption they are putting their safety at risk. Wow. The Ag's document actually states: "Meta represented that Messenger was safe and not harmful to Young Users' wellbeing when such representations were untrue, false, and misleading."

I sure hope that the Attorney General will be required to back that up with some clear evidence rather than just hand waving. The AG also says that there would be "minimal or no cost to Meta in complying with such an injunction, and therefore the burden on the company is light."

Meta disagrees, saying in its response that its ability to identify users based in Nevada is limited and is based on IP addresses and the user's self-disclosure about their location – both of which are not always accurate.

> *"To ensure compliance with the TRO, as a result, Meta may have to attempt to disable E2EE on Messenger for all users. Due to the truncated timeline here, Meta has not yet been able to assess the feasibility and burdens of doing so."*
>
> *Oddly, the AG asserts in its filing that the request for a restraining order is tied to a complaint that it sent Meta at the end of January. But, Meta notes, that complaint is based on claims that Meta's services are addictive to users and contribute to mental health issues in teenagers. The complaint barely mentions end-to-end encryption and doesn't reference at all the Nevada Unfair Practices law, which the AG cites as the legal reason for the court to grant the restraining order.*

The Register's coverage of this news included a quote from Georgetown University professor of computer science and law, Matt Blaze, saying: *"It's worth noting that it's not actually the encryption that they seem to object to, which would only hinder real-time interception. It's the failure to make a surreptitious, permanent third party record of otherwise ephemeral communications for the potential future convenience of a law enforcement investigation."*

Yikes.

And The Register also quoted the Standard Internet Observatory expert saying: *"Prohibiting Nevadan children, and only Nevadan children, from having end-to-end encryption for their online communications would not help children's safety, it would undermine it. Banning children in Nevada from having E2EE means giving some of the state's most vulnerable residents less digital privacy and cybersecurity than everyone else."*

–and–

"The FTC and other state attorneys general, such as California's, have long been clear that it is a consumer protection violation for companies ***not*** *to give users adequate digital privacy and security – and strong encryption is the gold standard means of doing that. It's therefore puzzlingly backwards for the Nevada attorney general to argue that Meta is violating Nevada consumer protection law here."*

Looking for the outcome of yesterday's hearing, I found a mention in the Las Vegas Review Journal which noted that a follow-on hearing was now scheduled for next month. So we can hope that whatever happens, this establishes a stronger precedent **for** encryption rather than one against it. Based on what Matt Blaze said, one has to wonder whether the ban on end-to-end encryption will then be followed by a mandatory requirement for the archiving of the communications of Nevada minors from some period of time and then AI scanning.

What's it like out there?

What's it like out in IT land? Get a load of this! Cybereason conducted a survey of more than 1,000 enterprise IT professionals asking about Ransomware. The survey found that **all** respondents suffered at least one security breach over the past two years. 84% percent of the respondents admitted ended up paying a ransom to attackers, but only 47% – so just over half – said they got their data and services back and running uncorrupted. And, 82% of respondents were hit again within a year. It's difficult for me to imagine being responsible for the security of a sprawling enterprise with complex networking requirements, people needing access everywhere all the time with employees receiving a stream of eMail and needing to click on links. Although all of that is required for the business to function, it's all also a nightmare to secure. And the job of making all of that work securely, which these survey results suggest is mostly not possible, is also mostly thankless. So to all of those IT professionals who are on what is literally the front line of cyberdefense, I salute you and I sincerely wish you the best of luck. I'm sure the job is fascinating, frustrating, infuriating, and challenging. So more power to you and god bless.

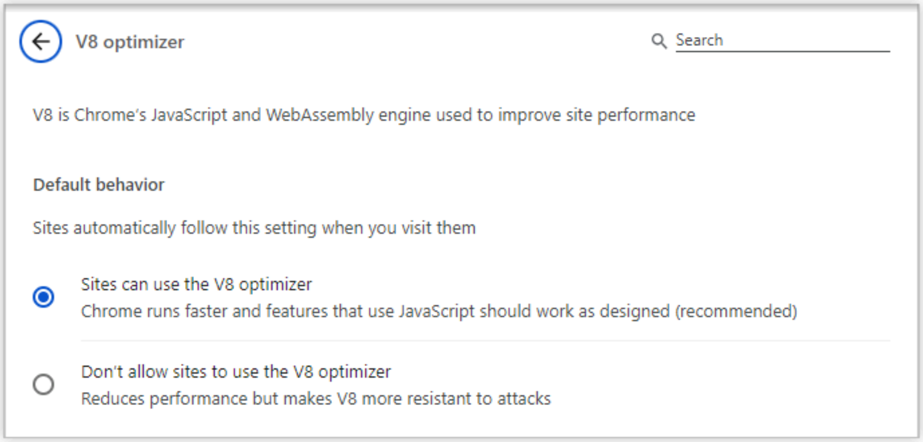
Edge's Super-Duper Secure Mode moves into Chrome

We talked about this three years ago in 2021. And I loved the name. How could anyone not love something called "Super-Duper Secure Mode"?? The surprise was that it came from stodgy old Microsoft – the IBM of the PC industry. Back then, Johnathan Norman, who was leading Edge's Vulnerability Research team at the time, explained that an important performance-vs-security trade-off existed because more than **half** of all past Chrome/Chromium engine 0-days exploited in the wild were issues related to the Just In Time (JIT) compiler.

What he and Microsoft were proposing for Edge was that with computers having grown so much more powerful than once upon a time with Just In Time compilation was added for performance, that extra edge in performance today was much less important that having an extra edge in security... and that the most obvious way to increase security was to turn off Just In Time code compilation. Super Duper Secure Mode does just that.

The idea proved to be a total success and it eventually went from being an experiment to being incorporated into Edge. Sadly, however, in the process, Microsoft stodginess won out, as it was bound to, so "Super Duper Secure Mode" became "Enhanced Security Mode" which is much less fun.

So last week, with the release of Chrome 122, the Chrome browser inherited the result of Microsoft's pioneering. If you put the address: **chrome://settings/content/v8** into Chrome's URL bar, you'll be taken to a page titled: V8 Optimizer:



V8 is Chrome's JavaScript and WebAssembly engine used to improve site performance

Default behavior

Sites automatically follow this setting when you visit them

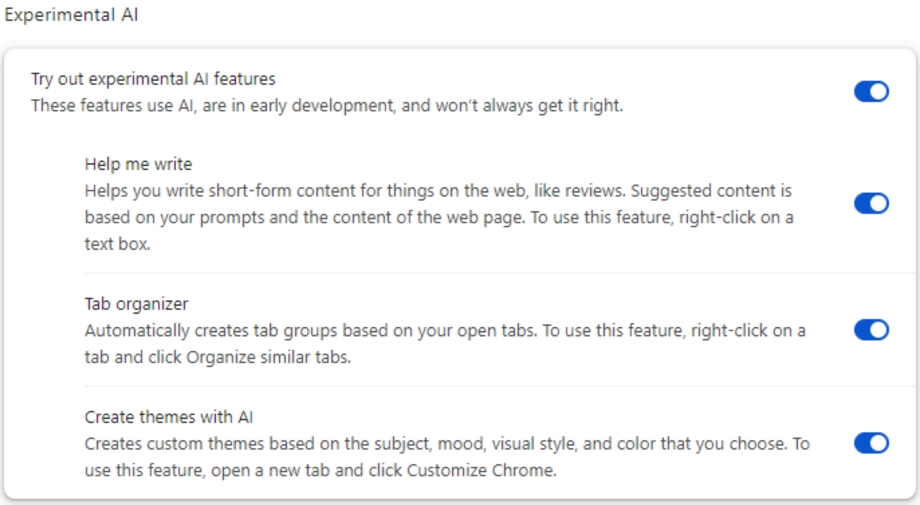
● Sites can use the V8 optimizer
Chrome runs faster and features that use JavaScript should work as designed (recommended)

○ Don't allow sites to use the V8 optimizer
Reduces performance but makes V8 more resistant to attacks

I tried searching at the top level of settings for "V8 Optimizer", but that didn't get me there.

This page allows you to flip the default from “Sites can use the V8 optimizer” to “Don’t allow sites to use the V8 optimizer”. My advice to Chrome users would be to give it a try and see whether you notice any difference. My guess is that for most sites, the probably minor difference in performance would be masked by the site’s own performance and network overhead. If that’s not the case, that page also allows for per-site overrides. So you could disable the use of the faster but more risky V8 JIT compiler unless a site really does benefit from having it, in which case that site could be white-listed to use V8.

I should note that Chrome 122 has also added some experimental AI features. If you click the three dots in the upper right, and choose “Settings” at the bottom of the drop-down menu. Over on the left about 1.3rd of the way down you’ll find “Experimental AI”. If you flip the switch, which is off by default, to “On”, the box expands to show “Help me write”, “Tab organizer” and “Create themes with AI”:



Experimental AI

Try out experimental AI features
These features use AI, are in early development, and won't always get it right.

Help me write
Helps you write short-form content for things on the web, like reviews. Suggested content is based on your prompts and the content of the web page. To use this feature, right-click on a text box.

Tab organizer
Automatically creates tab groups based on your open tabs. To use this feature, right-click on a tab and click Organize similar tabs.

Create themes with AI
Creates custom themes based on the subject, mood, visual style, and color that you choose. To use this feature, open a new tab and click Customize Chrome.

Each of those can be individually enabled. For me they were all initially enabled after I flipped the master switch. Since Firefox is my default browser I needed to fire up Chrome to check this out. So I cannot comment on the effectiveness of these new features for Chrome users.

DoorDash dashes our privacy.

On the topic of how much is apparently continuously going on behind our backs, I noted in passing that the home delivery service DoorDash has agreed to pay a $375,000 civil penalty for violating California's privacy laws. California’s Attorney General sued DoorDash for selling customer data without notifying its users or providing a way to opt out. The company sold customer data such as names, addresses, and transaction histories to a marketing cooperative.

More and more we’re all using these services. COVID drove a significant upswing in the use of home delivery services of all sorts. And many people use Uber, Lyft or something similar. And all of these services are being managed through online apps that need to know a lot about us in order to function.

So then along comes a marketing firm and offers these companies real money in return for sharing everything they know about their customers – who, in many cases never gave their permission to have their otherwise private two-party transactions **ever** shared with anyone, least of all some shadowy information broker who will be reselling it. But this is apparently exactly what's going on all the time. It's a hidden privacy cost of participating in today's connected economy.

Avast ye Matey!

And speak of the devil (and I do mean devil) the United States Federal Trade Commission has just fined the cybersecurity firm **Avast** $16.5 million for selling its users' browsing data. Yep, a spy in your browser. Wow. The FTC accused the security firm of using bait-and-switch tactics by offering browser extensions that blocked internet tracking but then selling browsing data behind its users' backs. If this sounds vaguely familiar it's because we did talk about it at the time. Between 2014 and 2020, Avast sold browsing data to more than 100 third parties through its Jumpshot subsidiary. The FTC has banned Avast from engaging in similar practices and has ordered the company to notify all users whose data was sold. Whoopsie.

No charge for extra logging!

We know how beneficial logging can be for monitoring a network environment's security. And to that end, Microsoft has taken some heat and come under the gun for charging their enterprise cloud customers extra money if they wanted logging that would serve to better protect them from security threats. So, in a move that CISA has greeted happily (after noting that Microsoft should do it) Microsoft has made many previously no-pay security logs free to use by its enterprise customers. 31 log categories have been moved from the premium tier of the Microsoft Purview Audit service into the standard offering. This was something Microsoft had promised last year in the aftermath of its Storm-0558 hack. So it's a welcome move in the right direction. Given the precipitating events and the pressure it was under, I wouldn't go so far as to suggest that this represents any actual change in philosophy within Microsoft, but this was definitely the right thing to do, regardless.

Who needs encryption, anyway?

Meanwhile, while politicians in the EU consider reducing browser security by forcing EU member country root certificates into our browsers, and consider the imposition of limits on the use of end-to-end encryption, the European Parliament's IT service has found traces of spyware on the smartphones of its security and defense subcommittee members. Yeah, who needs encryption anyway? The infections were discovered after members went in for a routine check up. The EU Parliament has sent a letter urging its members to have their devices scanned by its IT department.

LockBit gets bitten — *(Couldn't happen to a nicer bunch.)*

Law enforcement agencies from 11 countries have disrupted the LockBit RaaS – Ransomware as a Service – operation in the most thorough and coordinated takedown of a cybercrime portal to date. During the operation, which was codenamed **Operation Cronos**, officials seized LockBit server infrastructure, froze cryptocurrency wallets which were still holding past ransoms, released decryption tools, arrested members and affiliates, filed additional charges, and imposed international sanctions. Operation Cronos began months ago and was led by the UK's National Crime Agency (NCA). The agency infiltrated the gang's servers, mapped out their infrastructure, collected encryption keys, and accessed the LockBit backend, where admins and affiliates collected stats about attacks and negotiated with victims.

The takedown occurred last Monday the 19th and was announced the following day, one week ago on February 20th by the UK's NCA, Europol, and the US Department of Justice. In total, officials say they:

- Seized 34 LockBit servers;
- Identified and closed more than 14,000 online and web hosting accounts used in past LockBit attacks.
- Seized more than 200 cryptocurrency accounts holding past ransoms;
- Detained two affiliates in Poland and Ukraine;
- and indicted two other Russian nationals.

Lockbit affiliates who logged into their LockBit backend accounts on Monday were greeted by a special message from the NCA blaming the takedown on "LockbitSupp and their flawed infrastructure." The message urged affiliates to rat on their former boss, which tends to confirm the belief that law enforcement has yet to identify LockBit's creator. And you might imagine that he's gone into hiding, whoever he is.

And as was done in other recent cases of the Hive and AlphV disruptions, the cybercrime officials didn't just take down servers. They also collected the coveted Ransomware-as-a-Service backend — the encryption keys that were used to lock victim files.

Officials say the keys were handed over to a technical unit inside the Japanese national police, which created a decryption utility that can recover files from Windows systems. The utility is available through Europol's No More Ransom project.

The long term impact of this takedown is still unknown. As we've seen before, ransomware operations that met a similar fate later relaunched under new names. On the one hand, the Hive gang never returned after the FBI hacked its servers and released decryption tools last January. Whereas the operators of the AlphV RaaS service popped back online and started launching attacks from new infrastructure a month after the FBI took down servers and released their decryption keys this past December.

Firefox v123

Firefox advanced to release 123 last Tuesday and they wrote three things that might be of interest to our many Firefox users:

> - *We've integrated search into Firefox View. You can now search through all of the tabs on each of the section subpages - Recent Browsing, Open Tabs, Recently Closed Tabs, Tabs from other devices, or History.*

- *Having any issues with a website on Firefox, yet the site seems to be working as expected on another browser? You can now let us know via the Web Compatibility Reporting Tool! By filing a web compatibility issue, you’re directly helping us detect, target, and fix the most impacted sites to make your browsing experience on Firefox smoother.*

- *Address bar settings can now be found in the Firefox Settings' Search section.*

That web compatibility issue was something I recently encountered, but I don’t now recall where. And I’ve seen it more than once. The page attempted to load and it looked like it was going to, but then it just remained blank. The first thing I tried was to disable uBlock Origin for the site and reload it, but that didn’t help. So I turned uBlock Origin back on and tried it under Chrome, where the site worked perfectly.

In researching this further for this story I found that Firefox’s “Enhanced Tracking Protection”, which I **do** have enabled for all sites, is the most likely cause of trouble, but I didn’t think to try that and I should have. So next time this happens with Firefox I will. You click on the little shield icon to the left of the URL bar and assuming that “Enhanced Tracking Protection” is on, you turn it off. This will cause a page reload which may fix the problem. Now the shield will have a slash through it since “Enhanced Tracking Protection” will be disabled for the site. If you click on it then, you’ll see the question “Site fixed? Send report” and if you click that you’ll be able to add some optional comments and send a report to Mozilla about the site so that they can see about increasing Firefox’s “Enhanced Tracking Protection” compatibility. So the next time that happens that’s what I’ll remember to do!

But, if that’s not the problem, as of this release 123, there’s an explicit “Report Broken Site” option now always present under that Shield icon. For that to show, you need to have “Allow Firefox to send technical and interaction data to Mozilla” enabled on your main “Privacy & Security” page, but that’s now the default for new installs. And it’s definitely worth going to that “Privacy & Security” page anyway and scrolling down through its many friendly settings. You might well find something that you’d like to be doing differently.

My main point is that all of us who are using Firefox can contribute to its continued if somewhat threatened survival by providing that sort of broken site feedback whenever we encounter any trouble. It’ll be up there under the shield.

The last thing I’ve been wanting and intending to mention for a while is that I had become annoyed by Firefox’s apparently pointless division of the URL bar into two separate fields, with the URL on the left and a separate search box on the right. There are some instances where what I’m searching for looks like a domain name and might be confusing. So placing that into the right-hand search field would make that clear. But enclosing the term in quotation marks solves that easily enough. The single unified field is also the default now for new installations, but I’ve been using Firefox from before that was changed, so my top-of-screen still had two separate fields. So, if, like me, you still have separate fields, you might want to give it a try. Open Settings and search for “Address” and the option will immediately be at the top of the page.

ConnectWise? Not so much...

Last Monday the 19th the industry was informed of another horrific web authentication bypass in a widely used and popular product known as ConnectWise: Screen Connect. Unfortunately, this allowed bad guys to trivially connect to an enterprise's screens and network by completely sidestepping their need to identify themselves as an authorized party. And connect they did – in large numbers and almost immediately wasting no time. I'm not going to go into this very far, but to provide a bit of color, here's what Huntress Labs wrote about what they found. In their posting last Wednesday titled "A Catastrophe For Control: Understanding the ScreenConnect Authentication Bypass", Huntress wrote:

> *On February 19, 2024, ConnectWise published a security advisory for ScreenConnect version 23.9.8, referencing two vulnerabilities and software weaknesses. The same day, Huntress researchers worked to understand this threat and successfully recreated a proof-of-concept exploit demonstrating its impact.*
>
> *This write-up will discuss our analysis efforts and the technical details behind this attack, which we're coining as "SlashAndGrab." The ConnectWise advisory indicated that in all versions of ScreenConnect below 23.9.8 there were two vulnerabilities:*
>
> - *Authentication bypass using an alternate path or channel.*
> - *Improper limitation of a pathname to a restricted directory ("path traversal").*
>
> *The first vulnerability was disclosed with a critical base* ***CVSS scoring of 10, the highest possible severity.*** *The authentication bypass would ultimately open the door for the second vulnerability.*
>
> *ConnectWise made a patch available and expressed that all on-premise versions of ScreenConnect 23.9.7 and below must be updated immediately. At the time of release, the ConnectWise advisory was very sparse on technical details. There was not much information available as to what these vulnerabilities really consisted of, how they might be taken advantage of, or any other threat intelligence or indicators of compromise to hunt for.*
>
> *Once we recreated the exploit and attack chain, we came to the same conclusion:* ***there should not be public details about the vulnerability until there had been adequate time for the industry to patch. It would be too dangerous for this information to be readily available to threat actors.***
>
> *But, with other vendors now publicly sharing the proof-of-concept exploit* [and Huntress is posting this 48 hours after the ConnectWide disclosure]*, the cat is out of the bag. We now feel that sharing our analysis shares no more threat than what is already available. So, we're ready to spill the beans.*
>
> ***The "exploit" is trivial and embarrassingly easy.***

Anyway, further details are unimportant to further establishing the point; everyone gets the gist. We have yet another example of the truth that we do not yet understand as an industry how to do web authentication interfaces securely. Ohhhhh... we want to... since they're so friendly, colorful, attractive and appealing. Look at that, you just go there with any browser and you're logged into the enterprise's network. It's magic! And the bad guys love it just as much. They love how easy we've made it to log into enterprise networks. ***Web portal? Yes please!***

SpinRite

I'm very pleased to finally be able to announce that SpinRite 6.1's code is no longer a release candidate; it has graduated to official release. As we all know, since we've been watching, this significant update has been given a great deal of time to settle down and mature and there's nothing more than any of those who have been working on it know that needs to be fixed. It's ready. Since a great deal has changed from 6.0, I want to get its documentation created and online before we move purchasing from 6.0 to 6.1. But anyone who has SpinRite 6.0 can put their licensed serial number into GRC's "prerelease" page at www.grc.com/prerelease.htm to obtain SpinRite v6.1.

I also learned something very interesting yesterday morning: Sunday evening I had submitted SpinRite's final code to Microsoft's threat detection system, which was generating false positive detections and making downloading and running the program in Windows difficult. Here's the reply I received from them yesterday morning when I checked back in:

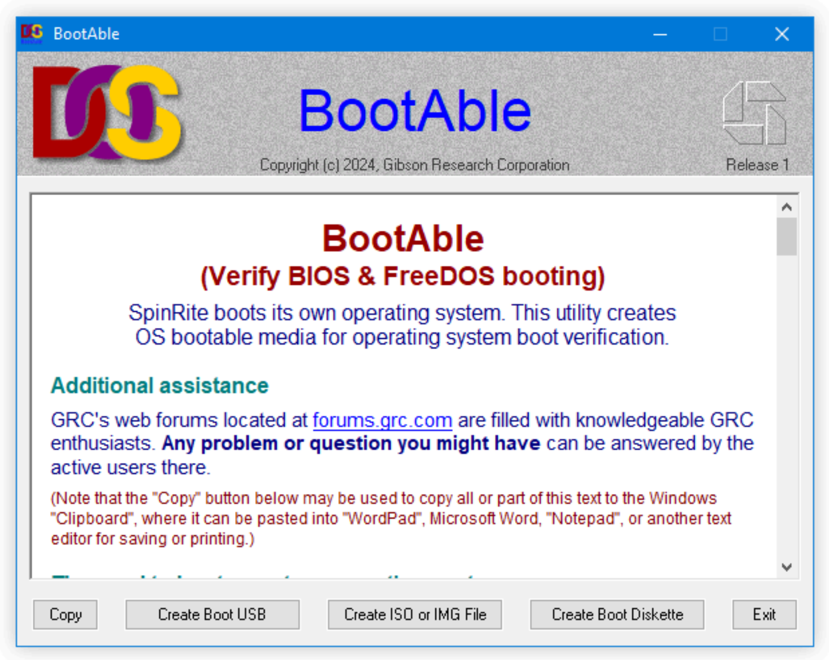
> *The warning you experienced indicates that neither the application nor the signing certificate had established reputation with Microsoft Defender SmartScreen services at the time. We can confirm that the application "sr61.exe" has since established reputation and attempting to download or run the application should no longer show any warnings.*
>
> *Please note, the signing certificate (56058A939D462D71A43A846A9145F35D9D1F2D92) thumbprint is still in the process of establishing reputation. Once completed, all applications that are signed with that certificate should have a warn-free experience from the start.*

That last bit is the best news I've received in a very long time. As I've mentioned before, I've been despairing over this, because there've been times in the past few months when Windows has been so aggressively protecting its users over its false belief that SpinRite was malicious that people have been unable to download and run its pre-release code. Every time they tried SpinRite would be immediately quarantined and removed from their system. And since every SpinRite download is customized with its owner's license information, and would therefore have a unique hash and signature, my great worry has been that no fresh download would ever be able to earn its own individual reputation.

Hoping that a signature ***might*** mean something, I spent a month figuring out how to get Microsoft's less well documented code signing APIs to work remotely on GRC's server with a hardware security module (HSM). That's finally been working without a single hiccup, and an HSM is the only way to perform EV code signing. But all I had was hope. It wasn't until I received Microsoft's notice yesterday that it finally became clear that it would actually be possible for GRC's EV certificate to eventually protect these individual downloads and SpinRite's users, from unwarranted harassment. I have no idea when that will come to pass, but it's clearly in the works.

What's interesting is that the reputation of that single SpinRite executable which I sent to Microsoft for analysis only took a few hours to obtain, but GRC's code signing certificate still hasn't. Since I wanted to obtain the longest run time possible for this new signing technology – and the certificate it would be using – right before I deployed it I asked DigiCert for an update. EV certs are good for three years, max. So on January 16th that new certificate was created and immediately placed into service. That's exactly 6 weeks ago today ... and over the course of those 6 weeks, thousands of copies of SpinRite's code, all signed by that new certificate, have been downloaded. And Microsoft's note exactly identified that certificate by its thumbprint. So Microsoft has been watching it for the past 6 weeks and still says: "*the signing certificate is still in the process of establishing reputation.*" What this suggests is that it takes quite a bit longer for a code signing certificate to establish a reputation – even an Extended Validation code signing certificate – than for any single piece of code that it signs. And that does make sense, since a fully trusted code signing certificate would be a very potent source of abuse if it were ever used to sign malicious code, since Microsoft has just confirmed what I have been hoping, which is that code gets a green light. At the same time, I'm quite certain that a reputation that was long and hard earned would be instantly stripped if Microsoft were to ever confirm that a piece of true malware was bearing that certificate's signature. So it's all good news on the SpinRite front.

I also have a new piece of GRC Freeware to announce:



BootAble

(Verify BIOS & FreeDOS booting)

SpinRite boots its own operating system. This utility creates OS bootable media for operating system boot verification.

Additional assistance

GRC's web forums located at forums.grc.com are filled with knowledgeable GRC enthusiasts. **Any problem or question you might have** can be answered by the active users there.

(Note that the "Copy" button below may be used to copy all or part of this text to the Windows "Clipboard", where it can be pasted into "WordPad", Microsoft Word, "Notepad", or another text editor for saving or printing.)

It's a Windows app called "BootAble" because it creates any sort of boot media, USB, CD, ISO, IMG or diskette, for the purpose of allowing its user to confirm, and/or to figure out, how to get any given PC compatible machine to boot DOS. And, Leo, you know how I am with naming my programs. I still vividly remember you laughing out loud when I first told you about "Never10". If for no other reason, I was sorely tempted to name this one "DOS Boot". The reason I didn't is that SpinRite 7 will boot on either BIOS or UEFI machines, and it will no longer bring DOS along for the ride. So I thought "BootAble", which is more generic, would be the better choice for the long run.

Okay. . . Let's see what our listeners have been thinking about:

Closing the Loop

Astralcomputing / @astralcomputing

> *Cox is transitioning its email service to Yahoo Mail for its users. Customers will be moved to Yahoo email while still retaining their email address and password. However, the POP/IMAP/SMTP settings for Outlook will change. My main concern is the security hassles this is going to create for users due to the "Password Reset" issues you've been talking about lately. Thinking of moving my 86yr old mom off COX before this happens, but it's going to be a nightmare to change all those email addresses at every utility/bank etc... Keep up the good work! (past 999) - SN listener from day #1, and proud SpinRite Enterprise supporter - W*

A SpinRite Enterprise supporter. That's very nice and honorable, thank you. For those who don't know, we offer three levels of license. The standard SpinRite end-user license allows it to be run on any machines the user personally owns. And as I've often noted, I would never complain about someone coming to the rescue of a friend or family member. If a company wishes to use SpinRite on any or all of their machines at a single location, we ask them to maintain 4 licenses for the version of SpinRite they are using. And if a large multi-location Enterprise wishes to use SpinRite across their entire enterprise, maintaining 10 licenses officially allows for that. So, again, thank you.

I did a bit of poking around and I've confirmed that 86 year old moms everywhere will not be disturbed by this change. Although Yahoo!'s network and servers will be the ones handling everything for COX in the future, none of COX's eMail addresses, ending in "Cox.net", will be changing. In their announcement about this, Cox wrote:

> *To ensure the best email experience possible for our customers, we have decided to transition the email service and support of your cox.net email to Yahoo Mail. This transition lets you keep your email address, messages, folders, calendar, and contacts. After the move, Yahoo Mail will become your email provider and Cox will no longer manage or support your email services. We realize how important your cox.net email address is to you and have carefully selected Yahoo Mail because we believe they are a trusted provider that will continue to offer the advanced support and enhanced protection for your email account that you've had at Cox. We will work with Yahoo to provide a seamless transition for our cox.net email customers.*

So, no need to change anything related to the eMail address itself. Your eMail client login domain will apparently need to move to Yahoo!, but that change should be minimal and quick.

eric mann / @e777ann

> *Hey Steve, I was just at my local grocery store and had a thought. In this day and age why do credit cards have the number, exp date, and cvv code printed/embossed on them? Everything a thief needs is right on the card. Simply not necessary for in-person transactions. All the info can be stored somewhere else, say Bitlocker? still loving the show, -E*

That's an interesting question – especially the embossed part. It's all certainly a holdover from the manual card processing days where the card would be placed in a manual credit card machine, a multi-part carbon slip would be placed on top, and the roller would be rolled back and forth across the slip and the card underneath.

I cannot recall the last time I saw that being done, but it remains a possible fallback in the event of a power outage where credit cards still need to be processed. As with an increasing number of things, like phone books and going to a library or even a physical book store, I imagine there are young people who have never encountered that being done.

Matsumura Fishworks / @matsumurafish

> *Hello Steve - I've been a Security Now listener for many years and can't thank you enough for all the security and computer science education you've given out so freely! (Also my kids are on a daily vitamin D regimen because of you) I had a question about one of the items from SN #962 (the gold standard of client-side hashing for password creation).*
>
> *In a scenario where the client submits their own hashed password, and the adherence to the password requirements is governed only by client-side controls, would there be any way to prevent a malicious party (like a pentester for example) from swapping out the hash in-transit and supplying the server with a valid hash of a non-conforming password? This would be admittedly counter-productive for the user, but it would seem that the server would lose the ability to make strong assertions about the hashes that it was accepting. Am I thinking about this correctly? I would love to hear any thoughts you may have on this, and thanks again for all you do!*

The essence of this listener's question is whether the receiving server is able to determine anything about the quality of the user's password from its hash, and the answer is no. Assuming that the user's browser employs a strong local PBKDF (password based key derivation function) the result will be a completely opaque fixed-length blob of bits from which absolutely nothing about the original source password can be reverse engineered. Hopefully, that PBKDF will also be salted so that it's not even possible to compare the results of that PBKDF function with previously computed passwords. So it's due to the total opaqueness of the result that we now depend upon the user's browser to enforce password complexity requirements right up front before the PBKDF function is applied because that's the only time it can ever be done.

Efraim K / @efykay

> *Hi Steve! Thank you for the great show! I am a long-time listener and excited for the opportunity to continue listening for many more years! In regards to password-less login by way of a link sent to a user's email and the concern over email security (episodes 961 and 962), I was wondering if there would be a way to construct the magic link from a cookie or the like from the user's browser session? That way the link would only work from the same browser session where the login request originated? Looking forward to hearing your take!*

At one point the same thought occurred to me, but I was in the middle of assembling the podcast so I didn't pursue it. But the answer is absolutely and unequivocally yes. Now that I've thought about it, here's a far stronger solution:

Even without being logged in, the user's browser will have obtained at the very least a session cookie from the site they wish to login to. That cookie will be valid until the browser is completely closed. And a bunch of information can be encoded and encrypted into the link that's eMailed to the address the user provides. So the eMailed link could include the time of day, the user's IP address, and the value of the unique cookie that their browser has been given. When the user then clicks on the link, it will open a new page at the domain they are wishing to authenticate to. In opening that page and sending the URL to the site's server, the server will be obtaining all of that information which is totally opaque because it's encrypted. So it decrypts the information and verifies that a reasonable amount of time has passed since the link was created and sent. It verifies that the IP address encoded into the link matches the IP address of the browser's query, and that the 1st party cookie the browser just returned with its query also matches the cookie value that was encrypted into that link.

I don't see any way for that system to be compromised. The IP address provides strong verification about the location and connection, and the browser cookie verifies that it's the same browser at that same IP. That link would be totally useless to anyone else who might be able to intercept it as a result of eMail's less than super-strong security.

So thank you for posing the question, Efraim... I'm very glad that we were able to revisit this issue again. We've just made the eMail-only login system utterly bulletproof.

Mykel Koblenz / @znelbokm

> *Steve - Just listened to your commentary again on auto keys (and the banning of the flipper zero). What you and the Canadian government have missed* [And Mabel is 100% correct] *is that this is only the access to the inside of the car - all cars from about year 2000 have used a (lets call it an RFID chip to simplify it) in the key that needs to be present for the car to start. Typically, the remote function is a separate system to the RFID chip in the car - so fixing the remote feature is not going to prevent the car from being stolen.*
>
> *And don't think that a remote is the only way to get into a car. Getting physical access to the inside of a car is easy. Break a window, use any number of methods for unlocking a door when keys are locked inside etc. Banning the Flipper zero will have no impact on the number of cars being stolen, not unless it is able to replicate the RFID function of the key.*
>
> *If the car has a CAN bus, then that is another avenue for attack/theft. There are video's of a Lexus having its headlight popped out to access the CAN bus at the back of the headlight and then the car is opened and started using an injection technique that fools the ECU into thinking that the key is present and the start signal has been given. Cheers*

And of course Mykel is 100% correct. My entire conversation about this was effectively off topic last week, since I was only thinking about unlocking the car, not about starting it and thus stealing it. And you cannot steal a car merely by unlocking its doors. So, thank you! And you're right, having the Canadian government banning Flipper Zeros will obviously have no impact upon auto theft. I would imagine that it's How-To TikToc and YouTube videos that provide the greatest impetus and explanation for the rise in Canadian auto theft... but what is any politician going to do about that?

ViperXX / @Viper2X

> *Hi Steve, a hello from Germany, long time listener, spinrite license holder ... , the Router Topic, The company "AVM" a very popular German router brand, actually does that, they require you to confirm security sensitive changes by pressing a button on the router or via a connected phone and in the last release they added an OTP token which lets you add it to your authenticator app.*

That is very cool. Let's hope that spreads since it might help to put a crimp in the trouble we're seeing with routers... and something really needs to be done.

Raed Iskandar / @Raed_I_21

> *Hello Steve, I was just listening to your response on our new Canadian ban of the flipper zero. Your challenge system is a good method to strengthen the car to key communication. However, the current Canadian car thefts are not relying on the jamming method. The thefts have been recorded by victims' security cameras using a signal extender to allow the attacker to unlock and start the car from the owners driveway while their key is in the house.*
>
> *Once the car has started the attacker just drive off with it and as long as they don't turn it off before reaching their destination they got what they came for. This is not even a capability that the flipper zero can currently perform. In my opinion, this type of attack requires a redesign of how the key and car communicate. Perhaps a shorter communication field would be required, like nfc, in order to make the key's signal not audible by a radio location outside of a victims house. Or perhaps a physical kill switch on the car key itself so when an owner is inside their house and are not expecting their key to be used to actively unlock the car, they can disable its radio. I keep my car keys inside an rf sleeve, which creates one extra step to unlocking my car, but completely blocks all the current attacks that have been occurring in my neighborhood. Looking forward to hearing your thoughts on this. Raed Iskandar*

I'm glad for the additional information, and our long-time listeners will recall that we extensively covered exactly this attack some time ago – the use of signal extenders for car theft which serve to trick the car and the key into believing that they are much closer to each other than they are. Keys normally not working from a distance is a feature, not a bug, and signal boosters defeat that somewhat weak security. At the time, we talked about adding "time of flight" to the security, though that becomes tricky when an active agent must respond to a ping, since its own response time might be long compared with the speed of light... though there might be something that could be done using phase shifting or interferometry to determine distance separately from sign strength. Again, I presume that there's a lot of work being done along those lines. But, once again, targeting the Flipper Zero as the culprit is way off the mark.

Emma Sax / @emmahsax
(provided some useful thoughts about meeting the need for throwaway eMail…)

> *I have a few comments regarding the email signups for tons of different throw-away websites. I started moving to an email alias service about a year ago. It's been a game changer for me. Due to Bitwarden's integration with my choice service (Bitwarden currently integrates with*

> *SimpleLogin, AnonAddy, Firefox Relay, Fastmail, Duck Duck Go, and Forward Email it makes it super easy to generate email aliases on the go. So now, I no longer mind if I need to provide an email to access a random website.*
>
> *As Leo said, even if you use a single throw-away email address, it's still a fingerprint, and it's still trackable across different websites. And if you use a personal domain with multiple email addresses, all emails with that domain are a fingerprint. With these alias services, there is no fingerprint. There's no tie between the different email addresses. I'm not saying whether these email alias services are the best, or whether Bitwarden is the best password manager, but I chose a provider I trust for both my email alias service and my password manager, and I haven't been disappointed with them yet. And their integration with each other is invaluable.*
>
> *Thanks for all you do; I'm so happy to hear you're going past 999 on Security Now. When I started listening to your podcasts a year ago (LastPass breach is what led me to find Security Now), I was sad to hear you didn't have much longer on the show. Now, I'm thrilled to have become a weekly listener!*

Thank you, Emma. We're glad to have you, too, and everyone else who finds this podcast to be worth their time. I really do understand how valuable everyone's time is. We've talked about Bitwarden's integration before so I thought it was worth sharing Emma's experience to perhaps give our listeners a nudge in that direction. Since more and more listeners are reporting encountering the "Join our website to access our valuable content" notices, I have the feeling that throwaway eMail is going to become increasingly necessary for anyone who would prefer not to be providing explicit tracking data.

DH⚡ / @schilling2k

> *Hi Steve! One remark about the "click link in email to login to your account without password" feature mentioned in EP 962: as mentioned during the episode, one could see it as a password sharing prevention mechanism because no one in their right mind would give access to their main email account. Nevertheless, you still could use a shared, separate email account specifically created for the login to specific services you **intend** to share. Daniel*

That's a great point. Instead of not wanting to share your eMail address, create a deliberately shared eMail account which is shared with those who you wish to have shared access. Then the eMail loop actually makes that easier... and it could be reused for multiple accounts, all being shared. Nice.

Christopher Ursich / @chrisursich

> *Steve, Chris from Cleveland here -- a listener since the days of The Onion Router, SecurAble, Jungle Disk and the Astaro Security Gateway. :) In SN962 you gave a recommendation for client-side password quality enforcement. We need to deprecate website passwords entirely, but in the meantime I think I have a better idea that is even easier for sites to implement:*
>
> *It shouldn't be difficult to define a declarative microformat (ala https://microformats.org) that sites can use to machine-readably inform browsers and password managers what password constraints the site requires. Bitwarden or Mozilla could even write the standard. This would*

> *allow sites that don't actually handle passwords properly to at least avoid burdening the end-user with cumbersome rules. Regards, Chris Ursich*

So, I agree with part of what Christopher has suggested and I think it's brilliant.

I doubt that the microformats.org that Christopher refers to would be adopted in a world that's pretty much settled upon JSON (JavaScript Object Notation) as its textual representation for structured data. Microformats date from 2004, so it's now 20 years old, and it worries about counting and minimizing character counts. That doesn't pack the same punch today as it would have when the 90's were only a few years removed. But the representation format of the data is really beside the point and doesn't matter. The brilliance is the idea that there could be a very simple means for our password managers to obtain a website's more or less arbitrary password rules without any human intervention.

When you're using a password manager and you know you're never going to need to remember any site's password, the longer the password the better, right? So 32 characters with all possible character classes mixed together would be perfect. But then you hit some annoying site that says "Your password is too long. 20 characters maximum." So, okay, you dial the length down to 20. Then it says "You must have some upper-case characters" and, sure enough, by the luck of the draw, that shorter 20-character password happened to be all lowercase, numbers and special characters. So you need to make your password manager generate another password. So you do that and now you're told that it must also have at least four non-consecutive numeric characters. Okay, so perhaps I've created a worst-case example, but everyone gets the idea. As I'm sure we've all needed to adjust at least the length of our password manager's automatically generated passwords.

We already have the well established /.well-known/ directory for locating website information in specific directories. So the industry could define a /.well-known/ directory named "password-rules" and that directory could contain a JSON file which succinctly describes the site's acceptable password policy. A configuration option in our password manager would be to poll any site's acceptable password policy whenever our password manager is about to present a password recommendation... and design the password it offers to match the most secure password allowed under that site's policies.

Anyway, I know it would be a heavy lift to get this adopted industry wide. But not all sites need to do it and those that did would be encouraging the use of the strongest possible passwords for their account holders. And it would also make automatic password rotations, when necessary, much more reliable. We know that even with the adoption of Passkeys, passwords will not be disappearing; they'll be with us for the foreseeable future. So automating the selection of the strongest possible passwords for a site seems like a useful feature.

Now What?

We're at page 19 in the show notes, which almost certainly means that I've been trying everyone's patience long enough for the week.

Even so, there were three additional stories that I ran out of time to cover:

1. The story I thought was going to be most exciting, generated some quite frightening headlines about a new side-channel attack on fingerprint biometrics. For example, Tom's Hardware coverage was headlined: *"Your fingerprints can be recreated from the sounds made when you swipe on a touchscreen — Chinese and US researchers show new side channel can reproduce fingerprints to enable attacks"* The only problem with that is that it's not even remotely true. It turns out that within the fingerprint biometrics research community there is are two generic fingerprint templates one called "MasterPrint" and the other "DeepMasterPrint". By themselves, these templates have a 1.88% and 1.11% chance of fooling any fingerprint sensor that's been trained on some specific individual's actual fingerprint. That alone is sort of interesting; that there's a generic template for fingerprints. But what these researchers found was that they were able to slightly better inform those very low performance generic MasterPrint templates by listening to the sound of a finger moving across a touchscreen. I suppose it should not be surprising that something might be learned from that, but it should also not be surprising that it's not very much, and that it's certainly not, as the breathless headlines claimed: "*Your fingerprints can be recreated from the sounds made when you swipe on a touchscreen*". It barely helped at all, so enough said about that.

2. I also wanted to have time to check back in on the state of our intrepid Voyager 1 spacecraft since it appears that it may have finally lost it's battle with time and entropy.
I'll make some time for more detail about that next week.

3. But the story that's probably going to be next week's main topic, so I definitely didn't have time to fit it in today, is Apple's announcement last week of PQ3, where "PQ" stands for Post-Quantum. The blog posting from Apple's Security Engineering and Architecture group contains sufficient detail to make for a terrific main topic. So... stay subscribed and we'll be back next week with all of the interesting details.