Web portal? Yes, please!

Description: What U.S. 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!.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-963.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-963-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got a great show planned for you. Steve's going to talk about that web-based login that's supposed to be secure. Turns out it's not, not even close. I hope you're not using it. We'll talk about the state of Nevada. Their attorney general wants to ban encryption on Facebook Messenger, but just for kids. That'll make them safer; right? And Steve has a new app he just made just for you. Then a whole lot more coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 963, recorded Tuesday, February 27th, 2024: Web Portal? Yes, Please!

It's time for Security Now!, the show where we cover the latest security news with this guy right here, everybody's favorite geek, Mr. Steve Gibson. Hello, Steve.

Steve Gibson: Yo, Leo.

Leo: How are you today?

Steve: Good to be back with you for the - I just realized we've got a leap year. I was looking at the calendar.

Leo: Yes, we get an extra day.

Steve: I said, hey, February's got 29 days.

Leo: Aren't we excited. What are you doing to do on your free Thursday?

Steve: How often does that happen? Well, we know exactly how often.

Leo: We know exactly.

Steve: Because many of us have written, what was the linear date...

Leo: Oh, god, it's hard to do that code.

Steve: It is a mess, yes. It is, like, who came up with this calendar? Certainly not any programmer because no one would do that to themselves.

Leo: The 29th, I know, well, but they have to because the actual clock is a little bit off from the calendar. But what is it, it's the 29th day every year that ends in four unless it ends in zero zero.

Steve: Like 100 and 400 are also exceptions, I think.

Leo: 100 and 400, yeah.

Steve: And, boy, whenever there's any talk about, like, well, we're going to get rid of this daylight savings time because it's a real mess, we're just going to stay on one, I think all the technology now that knows when the time changes would get broken; right? I mean, like, there's a lot of different things like clocks that have that built into them now.

Leo: Oh, yeah.

Steve: That's like, okay, well, there's more problems.

Leo: For the longest...

Steve: Anyway, speaking...

Leo: Go ahead.

Steve: Speaking of problems, Leo, our listeners come here to find out about problems.

Leo: Yes.

Steve: And, boy, have we have got some problems for them today. I titled this "Web Portal? Yes, Please!" for a reason that we'll be getting to. But we've got a lot of interesting questions to answer. What U.S. state is now trying to ban encryption for minors? Which has, like, got a lot of people wound up, as you can imagine. What shocking truth did a recent survey of IT professionals reveal? Things are not good for them. What experimental feature from Edge is Chrome inheriting? Are online services really selling our private data? And just how big a problem is that? And what about browser add-ons? Them, too? Should we be paying extra to obtain cloud security logs, is the question. And now that the dust has settled somewhat, what happened with LockBit? What new features just appeared in Firefox 123? And what lesson have we just received about another horrible instance occurrence? We'll be getting to that. And I have some news on the GRC software front.

Leo: Ooh.

Steve: And we also have a bunch of interesting feedback from our terrific podcast listeners. So another jam-packed episode of Security Now. And there were a couple things I couldn't get to, but I'll tell everybody about those at the end because we'll be getting to a couple of them next week.

Leo: Oh, good.

Steve: So I think a lot of fun in store.

Leo: You're busy again, all week long, preparing for this show. We also have a Picture of the Week that's tied to the headlines. It's ripped from today's headlines.

Steve: That's right.

Leo: All right. Well, you know what else doesn't get old, your Picture of the Week. Tell us about this one.

Steve: Yes. Now, somebody repurposed this picture for the news, as you mentioned. First of all, it started off just as a great picture all by itself. We see a yellow painted cinderblock wall that has sort of an exterior because, you know, cinder block is you can't, you know, you're not going to install an outlet in the cinderblock. So it's a steel, two-plug, AC outlet box which is mounted on the outside of the cinderblock. And it's got a black cord and an orange cord plugged into it. And they're sort of running off the screen. But taped next to what we learn is a very important set of cords, it says, "Do not unplug!! Magic cord runs entire company!!"

So, you know, clearly the lesson here is, whatever you do, just don't, you know, like if you need to vacuum the floor, don't unplug one of these to plug the vacuum cleaner in temporarily. Just go plug your vacuum cleaner in somewhere else. Well, repurposing this picture for today's news, it says down below, "Live Look at AT&T Network Security."

Leo: Oh, boy.

Steve: Where of course somebody tripped over a cord somewhere. And you know, AT&T's been curiously unsatisfying in what little they've said. I mean, we heard that, you know, that DHS, our U.S. Department of Homeland Security, and CISA, and the FBI were going to roll up their sleeves and get to the bottom of this major 12-hour outage that happened last week. I'm sure anybody with AT&T probably knew, and even people trying to call people with AT&T knew. And then AT&T just sort of said, well, it's an update that we were doing. The software went sideways.

Leo: They didn't even say software. I mean, their statement was completely...

Steve: Opaque.

Leo: Opaque is a good word.

Steve: Yes, yes. It is the word of the day, actually. I'll be using "opaque" several more times by the time we're done. There's a lot of opacity in today's podcast. I'm not sure why that all landed today. But, yeah. It's like, what? You know? And it's not like they're some little nothing company, right, that doesn't matter. It's like, this is important. People want to know, you know, what happened? But AT&T's like not really telling us yet.

Leo: I hope we find out something.

Steve: I imagine they have to tell their shareholders or something, I would think.

Leo: You know, after you talked so many years about BGP routing mistakes, I thought could be that. Then some wag on Mastodon wrote, you know, probably a certificate expired on some server in a closet somewhere that the only guy who knows it's there was fired eight years ago, and so they can't - it took them, it literally took them 12 hours to get back up and running.

Steve: And it could be that embarrassing; right?

Leo: Yeah, it must be.

Steve: Where like a company like AT&T cannot say, well, we had an expired certificate.

Leo: And we fired Joe, and he was the only guy.

Steve: Yeah, it took us a while to figure out what - it's like, oh, no. Heads would roll.

Leo: They implied they were doing some sort of network upgrade. But I think that was self-serving, too, like we're expanding our network. And, well, things sometimes go wrong when you expand.

Steve: For our listeners' benefit, we've had a 12-hour outage.

Leo: Yeah.

Steve: Wait, what?

Leo: What? Yeah.

Steve: No.

Leo: We did it for you, kids.

Steve: So Kim Zetter's Zero Day blog had the best coverage I've seen of this surprisingly aggressive move. I edited what Kim wrote down for length and readability. But here's the gist of the news, and then two other outlets weight in. Nevada's attorney general filed a motion to prevent Meta from providing end-to-end encryption to users under 18 who reside in the state of Nevada. And it's like, what? The request explains that its intention, of course, is to combat predators who target minors for sexual exploitation and other criminal purposes - now, there's always that extra clause in there, right, like, oh, wouldn't it be nice if we got some terrorists while we were at it - and, they say, to allow law enforcement to retrieve communication between criminals and minors from Meta's servers during investigations.

Now, what's interesting about this language is that suggests that there's some retention on Meta's part, and we'll get in a minute here to what Matt Blaze says about that. But anyway, Kim said in his reporting: "Last Tuesday, AG (attorney general) lawyers filed a partially redacted brief in Las Vegas federal court seeking a temporary restraining order" - so they're asking for a TRO - "and a preliminary injunction against Meta to prevent it from offering" - well, okay, but they have been. But okay, we'll get to that, too - "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 Attorney General's office claims that Meta's decision to enable end-to-end encryption by default is 'irresponsible' and 'drastically impedes law enforcement's efforts to protect children from heinous online crimes, including human trafficking, predation, and other forms of dangerous exploitation.'"

The AG requests an immediate hearing on the matter two days from then. And this was, I think, at the beginning of the week last week, or maybe like Tuesday, and so - yeah. I think this was Tuesday, and they wanted a hearing on Thursday. And it's like, wait. Okay, again, what? And it would have been, yes, last Thursday, citing the "extreme urgency" - again, exact quotes - affecting "the safety and well-being" of children in Nevada who use Messenger. The court scheduled the hearing for February 26th, so they didn't quite have it as quickly as they wanted to. So that was yesterday.

In its response to the filing, Meta said that the - so Meta, of course, responded, saying wait a minute, saying that the request makes no sense since it and other messaging services have been offering end-to-end encryption to minors and other users, which is to say anyone who wants it, 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 or minors.

Meta wrote: "The State cannot properly assert that it requires emergency injunctive relief, on two days' notice, blocking Meta's use of end-to-end encryption, 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."

So a legal expert and a research scholar at the Stanford Internet Observatory calls Nevada's request "bizarrely aggressive" - that was her quote - 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?" The expert cited it as being the biggest attack on encryption in the U.S. since 2016. And of course we know what that date was, 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, of course, the FBI wound up gaining access through another means and so dropped its push on Apple.

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 by bypassing it using 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, and this was a couple months ago, 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 for all 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 Signal and other similar applications, you know, Telegram and so forth. End-to-end encryption has been considered essential for protecting communications for years, Meta notes. And they said: "Indeed, Nevada law recognizes the value of encryption, requiring data collectors to encrypt personal information."

The Stanford Observatory expert noted that, if the court were to grant the restraining order and injunction, it would actually be making 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 of course there's the danger that this could set a precedent with other states then following.

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 seems like a stretch, 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 Attorney General states. Therefore Meta is directly and indirectly aiding and abetting child predators - boy - by providing them with end-to-end encryption. The Attorney General also states that Meta further violates the Unfair and Deceptive Trade Practices Act by misstating the risks minors face - what? - in using Messenger.

Leo: It's ridiculous. It's just ridiculous. There's no way to logic it out. It makes no sense.

Steve: Oh, wow. 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 Attorney General'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."

Wow. Well, I sure hope that the Attorney General will be required to back that up with some clear evidence rather than just waving their arms around. The Attorney General 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, of course, disagrees, saying in its response that its ability to identify users based in Nevada is limited and is based on IP addresses and the users' self-disclosure about their location, both of which are not always accurate. And we talked about this before, that like IP addresses, Internet routing is not constrained by state borders. Maybe by national borders, but not within the United States. We don't route based on state.

So "To ensure compliance with the temporary restraining order, as a result, Meta may have to attempt to disable end-to-end encryption on Messenger for all users."

Leo: Oh, well, that will make us all safer.

Steve: That's right.

Leo: According to that logic.

Steve: And that's like, why should only the kids be made safer, Leo?

Leo: Right. We all need safety.

Steve: Let's have - exactly. That's good. They said: "Due to the truncated timeline here, Meta has not yet been able to assess the feasibility and burdens of doing so." Oddly, the Attorney General 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 - so, right, save the children - 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 Attorney General cites as the reason for the court to grant the restraining order. Wow.

Of course The Register picked up on this and had a field day with it, you can imagine. I just grabbed one little piece of it. They quoted Georgetown University's Professor of Computer Science and Law, Matt Blaze. And Matt said: "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 Stanford 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 end-to-end encryption means giving some of the state's most valuable residents" - I'm sorry, most vulnerable, well, yes, and valuable residents - "less digital privacy and cybersecurity than everyone else."

And, she said, "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. A strong encryption is the gold standard means of doing that. It's therefore puzzlingly backwards," she wrote, "for the Nevada attorney general to argue that Meta is violating Nevada consumer protection law here."

Okay. So, now, I went looking for the outcome of yesterday's hearing because that Thursday request got bumped to yesterday, the following Monday. I found a mention in the Las Vegas Review Journal which noted that a follow-on hearing was now scheduled for some time next month. So that would be March. So we can hope that, whatever happens, this establishes a stronger precedent for encryption rather than one against it. It is, as you said, Leo, it is just nonsense. You know, and 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 what? AI scanning them? Help us.

Leo: I guess the theory is, well, predators could be having encrypted conversations with children that we wouldn't be able to see. But it's already illegal for the predators to be using the encryption. So I don't know exactly...

Steve: That's exactly the point that occurred to me, too; right.

Leo: Yeah. What are they gaining here?

Steve: Nevada makes it clear that bad guys can't use encryption. So, okay. Let them do it and then convict them for that. Nuts.

Leo: I just - it makes no sense.

Steve: And who knows what is going on? I mean, it would be interesting to know why Nevada? Like, what? But, you know, the good news is this will probably get smashed, hopefully, and set a precedent so other states won't even bother.

Okay. So what's it like out in IT land? Cybereason conducted a survey of more than 1,000 enterprise IT professionals, asking them about ransomware. How's that going? The survey found that all respondents, all more than 1,000 IT professionals suffered at least one security breach over the past two years. 84% of the respondents admitted ended up paying a ransom to attackers, 84%. But only 47% - so just over half - said they got their data and services back and running uncorrupted. So that's interesting. And 82% of the respondents were hit again within a year.

Okay. So it's difficult for me to imagine being responsible for the security - I've said this before - 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 in order to get their job done. Although all of that is required for the business to function, it's also all a nightmare to secure. I can't imagine how you even do that. And the job of making all of that work securely, which these survey results suggest is mostly not possible, is also mostly thankless.

So I just wanted to take a moment, having seen these results, you know, to say to all of the IT professionals who, you know, are literally on the front lines of cyber defense, that I salute you, and I sincerely wish you the best of luck. I'm sure the job is both or all fascinating, frustrating, infuriating, and certainly challenging. So, you know, more power to you. And god bless because...

Leo: I wouldn't do it.

Steve: I hope you can sleep at night.

Leo: We couldn't do it.

Steve: No.

Leo: We couldn't do it.

Steve: No, no.

Leo: It's the hardest work ever.

Steve: Yeah. And, you know, make sure you're getting paid enough money because, you know, you're going to need it for your health coverage later in life.

Leo: Yeah. Whenever I talk to these guys, though, mostly what they complain about is not lack of money for them - I'm sure they'd like more - but lack of budget to do the job they need to do.

Steve: Lack of resources.

Leo: Constant pressure to do it for less money, without the tools they need, et cetera, et cetera.

Steve: And the problem is it doesn't look like a profit center; right? It looks like a...

Leo: That's exactly right.

Steve: It looks like a profit sink.

Leo: Right.

Steve: And so it's, you know, it's like, well, but if we invest, you know, in a new Chromax 9 on the assembly line...

Leo: Exactly.

Steve: You know, we'll be able to spit out twice as many widgets. So, but, wow, you know. One, you know, look at the reputation damage that we're seeing sprinkling across the industry as major company after major company gets themselves zapped.

Leo: 82%, wow.

Steve: I know, Leo. Yeah.

Leo: It's really stunning.

Steve: Okay. So we talked about this little goodie three years ago, back in 2021. And I did, and who wouldn't, love the name. How could anyone not love something called "Super-Duper Secure Mode"?

Leo: Yes, I remember we talked about it.

Steve: It's wonderful. You know? And the surprise was that it came from stodgy old Microsoft, you know, 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-versus-security tradeoff had been noticed because more than half of all prior Chrome/Chromium engine zero-days exploited in the wild turned out to be issues directly related to the V8 Just In Time (JIT) compiler.

What he and Microsoft were proposing for Edge was that with computers having grown so much more powerful than they were in yesteryear, back when Just In Time compilation was added for its performance benefit, that extra edge in performance today had become much less important that having an extra edge in security, and that the most obvious way to increase security was just to turn off Just In Time code compilation. Super-Duper Secure Mode did 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's stodginess did win out, as it was bound too; right? There's no way Super Duper Secure Mode would actually end up in the Edge UI. No. It became Enhanced Security Mode, you know, much less fun.

But anyway, last week we saw the release of Chrome 122. The Chrome browser in the process inherited the result of Microsoft's pioneering. If you put the address into your Chrome URL, chrome://settings/content/v8, you'll be taken to a page titled "V8 Optimizer." And there you will find two radio buttons. The first one, which is on by default, sites can use the V8 Optimizer. The other one, which I would argue is worth

exploring, click it and you get "Don't allow sites to use the V8 Optimizer." Now, as for getting there, I did try searching from the top level of settings for "V8 Optimizer," but that didn't get me there. So again, chrome://settings/content/v8, you know, numeral 8. And this page - so as I said, this page allows you to flip the default from yes, everybody gets to use V8, to no, don't want V8 because it's dangerous.

So my advice to Chrome users would be to give it a try and see whether you notice any difference. I'm guessing that for most sites, maybe all, the probably minor difference in performance would end up being masked by the site's own performance and the network overhead of stuff getting between you and them. And if that's not the case, that is, if a site should actually like be noticeably slower, that page also allows for per-site overrides. So you could just disable the use - so you could globally disable the default use of the V8 Just In Time compiler. But then if you end up with a site that does benefit from having it, just whitelist it for that one site.

And I should also note that also with Chrome 122 they added some experimental AI features. And I'm not going to roll my eyes. We have a long way to go. This is the very beginning of the AI What Is It journey. So if you click the three dots in the upper right of Chrome, you know, the Chrome of Chrome, and choose Settings at the bottom of the dropdown menu, over on the left, in that list on the left, about a third of the way down, you'll find "Experimental AI." If you flip the switch, which is off by default, to "On," the box there expands to show you three items: "Help Me Write," a tab organizer, and "Create themes with AI."

I've not gone any further since I'm not using Chrome any longer as my default browser. I'm happily back using Firefox. But, for example, under "Help Me Write," it says: "Helps you write short-form content for things on the web, like reviews." Great. Oh, boy. "Suggested content is based on your prompts and the content of the web page. To use this feature, right-click on a text box." So anyway, they gave us an example, you know, where you, like, I don't know, I want to ask for a refund on my airline tickets, you know, and it wrote it for you. So okay. Anyway, it's built into Chrome now. Have fun. And we'll see where AI takes us.

On the topic of how much is apparently continuously going on behind our backs, without our knowledge or awareness, I noted in passing that the home delivery service DoorDash has agreed to pay, not a crushing fine, $375,000, still attention-getting, in 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, like, what was brought to your door that you were dashing to get, to a marketing cooperative.

Now, 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 for them to function. You know, we give them the information that we understand that app needs based on the service it's providing. But when we're not being told that that information, which could be significant about us, is going to be used to create further profit for this company, that seems wrong.

You know, along comes a marketing firm and offers these companies real money in return for sharing everything they know about us, in many cases never giving us any permission or any opportunity to say whether that's all right with us or not. So, you know, and then where does it go from some marketing cooperative? It's being resold to, you know, information brokers and who knows what else. So it's a hidden privacy cost of participating in today's connected economy.

And speaking of which, the United States Federal Trade Commission, the FTC, has just fined the cybersecurity firm Avast a somewhat larger sum, gulp, $16.5 million for selling its users', okay, its users' web browsing data.

Leo: Oh, I remember this. I'm glad. Good.

Steve: Yes. Yes, the finally got a number. And you're right. If that sounds familiar to people, it's because we talked about this when it first became news. Essentially, Avast was functioning as a spy in our browser. 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.

Leo: Brilliant.

Steve: Yeah, we're going to block that tracking for you because you don't want to be tracked.

Leo: Let us do it.

Steve: Yeah. We'll be one-stop shopping for tracking. So between 2014 and 2020, Avast - get this, Leo - sold browsing data to more than 100 third parties, everywhere their users went, through its Jumpshot subsidiary. The FTC has banned Avast from engaging in similar practices - I wish they would ban them from doing business on the planet.

Leo: No kidding.

Steve: And has ordered the company to notify - oh, that'll be fun - notify all users whose data was sold. That'll be...

Leo: [Crosstalk] sorry note.

Steve: That'll be an interesting written - okay, attorneys.

Leo: I'm sorry.

Steve: Earn your keep, you attorneys. Make this - or I guess the attorneys meet with the PR people.

Leo: Yeah. Then they fight.

Steve: And probably, yeah, and probably the - I forgot the department name that - Human Resources. Because we would all like to keep our jobs also.

Leo: Yeah. Yeah.

Steve: So you guys figure out what we have to write in order to send this. Okay. One more, and then we'll take our next break. 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 services that would better protect them from security threats which were Microsoft's fault. Ouch. So in a move that CISA has greeted happily, after noting that Microsoft should do it, Microsoft has finally made previously extra pay security logs free to use for its enterprise customers. Thirty-one logging categories have just been moved from the premium tier of the Microsoft Purview Audit service into the standard offering.

Leo: Oh, good. Wow.

Steve: Yes.

Leo: Great.

Steve: Yes. 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. On the other hand, 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.

Leo: We'll just charge for something else, that's all.

Steve: That's right. We'll just, you know, we'll make up for the lost profit by increasing the price of, what, maybe security patches. How would that be?

Leo: All right. Let's take a little break. And then you and I shall return with more. And I'm looking forward to hearing this news that you mentioned, that you referred to about GRC. But all that's still to come.

Steve: I've got another laugh-out-loud app title for Leo.

Leo: Oh, good. I love those.

Steve: Okay. So in a little bit of, I don't know that this is exactly schadenfreude, but while the 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 for their citizens, the European Parliament's IT service has found traces of spyware on the smartphones of its security and defense subcommittee members.

Leo: Oh-ho.

Steve: Who needs that encryption? The infections were discovered after members went in for a routine checkup. The EU Parliament has sent a letter urging its members to have their devices scanned by its IT department.

Leo: Wow.

Steve: So, yeah, maybe it's good to be running with security set to max on your smartphones.

Law enforcement agencies, as there's been a lot of coverage of this and then some brief mention here, I titled this "LockBit gets bitten." Law enforcement agencies from 11 countries disrupted the LockBit RaaS, the Ransomware as a Service operation in which was the most thorough and coordinated takedown of a cybercrime portal service to date. During the operation, which was codenamed Operation Cronos (C-R-O-N-O-S), 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 several months ago and was led by the UK's National Crime Agency, their NCA. The agency infiltrated the gang's servers, mapped out their infrastructure, collected their truly secret master encryption keys, and accessed the LockBit backend, where admins and affiliates collected stats about attacks and negotiated with their 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 U.S. Department of Justice in a coordinated disclosure. 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" - who's the big cheese of Lockbit, the Kingpin - "LockBitSupp (S-U-P-P) 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 actually there's even some news since then. 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 the 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, who created a decryption, a master decryption utility that is able to recover all files from Windows systems that had previously been locked with LockBit. The utility is available now 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 might relaunch under a new name. On the other hand, for example, the Hive gang never did return after the FBI hacked its servers and released decryption tools a year ago January; whereas the operators of the AlphV Ransomware-as-a-Service did pop back online and start launching attacks from a new

infrastructure a month after the FBI took down their servers. And in even more recent news, just as we were getting ready to start the podcast, I saw that LockBit has reemerged already under new infrastructure and has posted the news about its first 12 new victims.

Leo: Didn't take long. Wow.

Steve: Did not take long, no.

Leo: Amazing.

Steve: Firefox v123. That happened last Tuesday. And they wrote three things that might be of interest to our Firefox users. They said: "We've integrated search into Firefox View. You can now search through all the tabs on each of the section subpages - Recent Browsing, Open Tabs, Recently Closed Tabs, Tabs from other devices, or History." And actually that's kind of cool, to be able to search like recently closed tabs. Sometimes when I am busy and closing things, I go, ooh, what was that thing that I had before? And so being able to search through that content would be very cool.

Also, well, okay, two other things, and they had a lot to say. They wrote: "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 helping us detect, target, and fix the most impacted sites to make your browsing experience on Firefox smoother." And finally they said: "Address bar settings can now be found in the Firefox Settings Search section."

Okay. So the web compatibility issue was something I recently encountered. But, and it bugs me, I don't now recall where because I would like to go back. 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 then reload it, but that didn't help. The same thing happened. So I turned uBlock Origin back on. And then I tried the site under Chrome, where the site did work correctly. And so I just did whatever it was I was doing and then came back to Firefox.

In researching this further for the story I found that Firefox's "Enhanced Tracking Protection," which I do have enabled for all sites, is the most likely cause of this kind 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 an automatic page reload which may fix the problem. Now the shield will have a slash through it, since "Enhanced Tracking Protection" has been 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, with a single click, about the site containing the information that they will need so they can see what's going on and work on fixing Firefox's "Enhanced Tracking Protection" compatibility so that it works better. So the next time that happens, that's what I'm going to remember to do.

But that's not what just changed here in Release 123. There's now an explicit "Report Broken Site" option always present now 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. I just

tried it to verify that, and it is on. And I would imagine all of the listeners of this podcast have that turned on.

And doing this, figuring this out, brought me back to the Privacy & Security page in Firefox, and I think it's definitely worth going just to that page and scrolling through it just from time to time; but, you know, but do it soon because it's got many friendly settings, and you might well find something that is off that you thought was on, or that you'd like now to be doing differently. But anyway, for all of our Firefox listeners, and I know we have many, if a site misbehaves, click on the little shield, and you'll be able to easily and quickly send news of that misbehavior to Mozilla so that they will be able to keep Firefox working well. And of course, as we know, unfortunately its continued existence in the world may be a little endangered, so it's worth doing that, I think, to keep it going.

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 that might be confusing to Firefox where it's trying to figure out should I search for it or go there? So placing that into the right-hand search field would make that clear. But just enclosing the term in quotation marks solves that easily enough. The single unified field is now the default for new installations of Firefox. But I've been using Firefox for so long, from before that was changed, so my top-of-screen still had two separate fields.

Leo: Wow. I turned that off, like, 30 years ago, I feel like.

Steve: Yeah. Yes. And I did it, like, two months ago.

Leo: Wow.

Steve: So anyway, I just wanted to say, if like me you still have separate fields, if you go to - just open Settings and search for Address, you know, A-D-D-R-E-S-S. The option will immediately be at the top of the page. Just flip it to a unified field and, yes.

Leo: You're okay with it now. You get used to it.

Steve: Oh, no, I mean, I was using Chrome for a while where it's a unified field. Or Edge, where it's a unified field - nobody else still does that except me on Windows 7. So I just wanted to let everybody know, Firefox lets you easily turn that off.

Leo: The unibrow, I think, is what we call it. No, no. That's not right.

Steve: The which?

Leo: The unibrow. But I don't think that's right.

Steve: Yes. It's the uni something.

Leo: They do have a name for it, yeah, yeah.

Steve: Yeah. Okay. So this one is the reason I titled the podcast "Web Portal? Yes, Please." Last Monday the 19th, the industry was informed of yet another horrific web authentication bypass in a widely used and popular product known as ConnectWise ScreenConnect. 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.

Leo: Holy cow.

Steve: I know, Leo. It's just astonishing. And connect they did, in large numbers and almost immediately, wasting no time. I'm not going to go too far into this because, you know, it's like, really? Again? But Huntress Labs wrote about what they found, and it's worth giving this a little more color. The title of their posting two days later, last Wednesday, was "A Catastrophe for Control: Understanding the ScreenConnect Authentication Bypass."

They wrote: "On February 19th, 2024, ConnectWise published a security advisory for ScreenConnect v23.9.8, 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. In other words, it's always been there, folks: an authentication bypass using an alternate path or channel, and improper limitation of a pathname to a restricted directory." In other words, a path traversal mistake.

They wrote, Huntress wrote: "The first vulnerability was disclosed with a critical base CVSS score of 10 - that is, right, 10 out of 10, the highest possible severity." Which is basically where a system just says, please, come on in, whoever you are. Username, password, ah, don't bother. Just click, you know, Submit. The authentication bypass would ultimately open the door for the second vulnerability.

They wrote: "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." Uh-huh. "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." You know, basically ConnectWise just was saying, holy crap, please, please, please, everybody update to 23.9.8. Don't ask any questions. Just do it now.

Huntress said: "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," they wrote, "with other vendors now" - two days later - "publicly sharing the proof-of-concept exploit, the cat is out of the bag. We now feel that sharing our analysis shares no more threat than what is readily available. So we're ready to spill the beans." And they finished with their intro saying: "The 'exploit'" - and they had that in quotes because it's not - "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 fully understand as an industry how to do web authentication interfaces securely. Oh, yes, we want to, since they're so friendly, colorful, attractive, and appealing. Look at that, you just go there with any web browser, and you're logged into the enterprise's network. It's magic!

Leo: Oh. Yeah.

Steve: 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!

Leo: It wasn't as easy as just leaving the fields empty and pressing Submit; right?

Steve: No. I didn't even go any further.

Leo: Okay.

Steve: Because, you know, because it was immediately picked up by bad guys.

Leo: Yeah, of course.

Steve: And every enterprise that hadn't updated by the time it was reverse engineered, which took apparently minutes to reverse engineer, they were then being compromised.

Leo: I wonder how many people use ConnectWise.

Steve: It's a big deal, apparently.

Leo: Is it? It's for managed service providers.

Steve: It's, yes, because they're so easy and so powerful.

Leo: Yeah, wow. Oy. Well, I hope our MSP - I don't think Russell uses it. I think we'd know if he did.

Steve: Yeah. Well, I do have some good news. 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 its official release.

Leo: Yay.

Steve: So I announced that on Sunday.

Leo: What does that mean?

Steve: It means that, well, it means that I thought I was done. It turns out that something I did with probably conditional assembly, when I switched it around out of release candidate stage, caused the SpinRite executable, which is written to the diskette image, to have the attribute of the volume label. I have no idea, I mean, and I learned about it yesterday morning when I was already started in on the podcast production. So I haven't looked at it. But it makes USB, it makes bootable USBs just fine. And that's what almost everybody uses now. So it's not a big problem.

But the diskette image is used both for creating a bootable diskette and the ISO and the IMG images. So they don't boot right now. This evening I'll fix it. And it'll be SpinRite 6.1, Release 2, rather than Release 1, as it is right now. But so you might want to wait till tomorrow, if anybody's been waiting. But it's done, is what this means. And all the bugs, all the features, all the bell-and-whistles, blah blah blah, it's done.

Leo: It's more a typo than a bug is really what you're saying.

Steve: Yes. It is, exactly, it is something so dumb, I mean, intellectually I am so curious. It's like, what the heck?

Leo: Where did I put that?

Steve: How did that happen? Yeah.

Leo: Yeah.

Steve: So, okay. But I learned something very cool that I wanted to share with our listeners as a consequence of this. Sunday evening I had submitted SpinRite's final code to Microsoft's threat detection system, as everyone's been hearing me talk about. Prerelease users have been driven to grief by their Windows systems immediately deleting their copy of SpinRite, where they were unable to run it because it would immediately be quarantined and deleted, thinking that it's some random trojan which obviously it's not. But it's a false positive. And if you google that particular trojan, turns out like it misfires for a lot of people. A lot of people are doing something that, like, freaks out Windows. Anyway, so that's why I spent a month doing code signing, right, hoping that if I signed my code, then I would get the benefit of the doubt. But it didn't seem to be happening.

Okay. So Sunday evening I submitted this final code to Microsoft Threat Detective System because, yes, it was generating false positive detections and creating a problem. Yesterday morning I checked on that, and here's the reply that I received from Microsoft. They said: "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."

Then they said, and this was what warmed my heart: "Please note, the signing certificate thumbprint" - and then they gave the hex, which I checked is the signing certificate on the server - "thumbprint is still in the process of establishing a reputation. Once completed, all applications that are signed with that certificate should have a warn-free experience from the start."

Leo: Oh, nice. Interesting. So you somehow have to establish reputation, even for a signing certificate.

Steve: Yes. And that last bit is the best news I've received in a very long time. You know, as I've mentioned before, I've been despairing over this because there've been times in the past few months - there was some guy a couple days ago who wanted a refund from a recent purchase because he was unable to run SpinRite. He was all excited, but he couldn't run it because his Windows 11 kept deleting it out from under him. So, you know, every time they tried, SpinRite would just immediately quarantine it, I mean, Windows would immediately quarantine SpinRite and remove it from their system.

So hoping that a signature might mean something, and by the way, this was - all of these have been signed; right? The signing system's working perfectly now, and beautifully, never having a hiccup. But it wasn't helping. So that's why I had spent that 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 because I have it, you know, the EV code signing cert is in an HSM, and you have to use an HSM for EV code signing. So I did all that, and all I had was hope.

It wasn't until yesterday morning when I received Microsoft's note 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. And, I mean, and the reason is that every copy a SpinRite user downloads has their licensing information embedded in it. So it's brand new. It's never been seen by Windows, which is why it's always freaking Windows out. It's like, what? What's this? And because the signing certificate doesn't yet have reputation, Windows quarantines it.

So 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, that is, reputation. 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 that it would be using, right before I deployed it in the middle of January I asked DigiCert for an update. EV certs are good for three years, max. So on January 16th that new certificate was created, and I immediately placed it in the certificate. That is exactly six weeks ago today. And over the course of those six weeks, thousands of copies of SpinRite's code have all been signed by that new certificate and downloaded and run when users are able to. And Microsoft's note exactly identified that certificate by its thumbprint.

So, you know, we know that Microsoft has been watching this certificate for six weeks, and it 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, it was more difficult to obtain, and it can only be used from a hardware security module. And really, when you think about it, that makes sense, since a fully trusted code signing certificate would be a very potent source of abuse if it were ever used to actually sign malicious code, since Microsoft just confirmed what I had been hoping, which is that code that gets signed by it gets a green light by default.

So, you know, at the same time I'm quite certain that a reputation that was long and hard earned would be instantly stripped, obviously, if Microsoft were to ever confirm that a piece of true malware was bearing that certificate's signature. So anyway, this is all good news on the SpinRite front. I'm done with the product. I'm working on the documentation. I'll fix the little bug in the executable being flagged with a volume label attribute so it won't run, I'll fix that, and Release 2 will be available later this evening, I'm sure. And it looks like, you know, this certificate is on its way to establishing reputation. I have no idea who to ask or how long it's going to take. I think what I'm going to do, though, is the GRC Benchmark is now being downloaded about 1,600 times per day. And interestingly, ValiDrive has turned out to have some legs. It's now - it's been steadily increasing...

Leo: Oh, I'm not surprised, yeah.

Steve: ...in popularity. It's now at more than 1,200 downloads a day. So I think I'm going to cosign both of those with this same cert so that it gets way more downloads, and Microsoft sees it a lot more. And then, I mean, I don't know if it's time or it's number of downloads or I just don't know what their metric is for what it takes to establish reputation.

Leo: If anybody from Microsoft's listening, can you do a solid for our man here and run down the hall and say, can you push that through? There must be somebody who can help you listening to the show.

Steve: Leo, if they're listening to the show, they're upset with me more than they're wanting to help me.

Leo: Oh, no, they know. They work for Microsoft. They're used to it.

Steve: I guess that's true. I guess that's true. So I also have a new piece of GRC freeware to announce. 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 freely confirm, and/or figure out, how to get any given PC-compatible machine to boot DOS. And, Leo, you know how I am with my naming programs. I still vividly remember you laughing out loud when I first told you about "Never10."

Leo: Yes.

Steve: You thought, what a name.

Leo: Great name.

Steve: Anyway, so I was sorely tempted to name this "DOS Boot."

Leo: I wish you had. Oh, I wish you had.

Steve: I know. I know. It would be so good. The reason I didn't is that SpinRite 7 will boot on either BIOS or UEFI machines, and it will no longer be bringing DOS along for the ride. So "BootAble," which is more generic, would be the better choice for the long run. So anyway.

Leo: So, does this - I wonder, maybe this will be useful for people who want to install Linux, too, because Linux has trouble with secure boot and in some cases UEFI.

Steve: Yes. Yes. I mean, so the idea would be you would have to, well, actually Linux will install on UEFI, and this won't test that yet.

Leo: Yeah, most Linuxes will, yeah.

Steve: Yeah. This won't test that yet. So you need BIOS or a CSM, you know, the compatibility, the software module on UEFI. But this allows people - so I wanted something so that people weren't buying SpinRite and then getting upset that they wanted to run it on a laptop that's UEFI-only. Or that, you know, or they were concerned about whether or not they would be able to boot SpinRite on any given machine. This is freeware, and it has all the same boot technology that SpinRite 6.1 has, and it's free. So you're able to just easily create a USB thumb drive and play around with, like, you know, do you hit F12 or F2 or delete, you know, because you have to intercept the normal boot, right, in order to get it to, like, boot [crosstalk].

Leo: And I can never remember those keys.

Steve: Yeah. And there's no standard. Every machine is different. You know, I think they randomize it at the factory. So anyway, just another little simple piece of freeware for everybody.

Leo: Very nice, thank you.

Steve: And Leo, let's take our last break, and then we're going to do a bunch of feedback from our listeners.

Leo: Great, let's do it.

Steve: Astralcomputing tweeted: "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," he writes, "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 86-year-old mom off Cox before this happens, but it's going to be a nightmare to change all those email addresses for every utility, bank, et cetera. Keep up the good work past 999. SN listener from day one and proud SpinRite Enterprise supporter. Signed, W."

Which brought me to note, just for anyone who's interested, a SpinRite Enterprise supporter is rare, but I always like to see it. It's nice. 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 that the user personally owns. And as I've often noted, I would never complain about someone coming to the rescue of a friend or a family member in need. If a company wishes to use SpinRite on any or all of their machines at a single location, we ask them to maintain four licenses for the version of SpinRite they're using. And if a large multilocation enterprise wishes - and we call that a "site license" if you have four SpinRite licenses. If a large multilocation enterprise wishes to use SpinRite across their entire enterprise, you know, wherever, then maintaining 10 licenses officially allows for that. So, again, Astralcomputing, 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.

Leo: Oh, good.

Steve: Oh, yes. Although Yahoo!'s network and servers will be the ones handling everything for Cox in the future, none of Cox's email addresses, which all end 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'll work with Yahoo to provide a seamless transition for our Cox.net email customers."

So anyway, no need to change anything related to the email addresses themselves. Your email client login domain will apparently need to move to Yahoo!, but that change should be minimal; right? You just change a couple settings for POP or IMAP or SMTP, and you're good to go. But Mom will not need to change any of her email addresses.

Eric Mann asked, he said: "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, expiration 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. Eric."

So that's an interesting question, actually. You know, especially, Leo, the embossed part. You know, it's obviously all a holdover from the manual credit card processing days.

Leo: Remember that? You had to go shump-shump.

Steve: Yup, where a card would be placed in a manual credit card machine, a multipart carbon slip would be placed on top, and then the roller would be rolled back and forth across the slip and over the card underneath, you know, to basically transfer the card's data, the credit card number and the expiration date, onto the carbon.

Now, I can't recall the last time I saw that being done, but it does remain a possible fallback in the event, for example, of a power outage, where credit cards still need to be processed, or if there was some Internet connectivity outage where you were not able to do it, like your credit card processing terminal wouldn't work even though you had power. And, you know, as with an increasing number of things, like phone books and even going to a library or, sadly, a physical book store, I imagine there are young people who have never encountered a manual processing of a credit card. But anyway, just sort of interesting that it's still, you know, they are still embossed, just like in the old days.

Leo: Yeah. Although my latest American Express card does not have that on.

Steve: Oh, really. So they finally have given that up.

Leo: Yeah. I think that a lot of cards are giving that up.

Steve: I mean, it does make sense.

Leo: Yeah.

Steve: You know, because you could just manually transcribe the number onto the same carbon, if that was actually necessary.

Leo: Yeah. If you needed one, yeah.

Steve: Yeah.

Leo: I don't know who's - you're right. It's if the power goes out or some station in Nevada or somewhere, some gas station somewhere doesn't have - yeah.

Steve: Right. So Matsumura Fishworks, I guess that's the guy's company name...

Leo: I hope so.

Steve: He said: "Hello, Steve."

Leo: Be a terrible personal name. But go ahead, okay.

Steve: He said: "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." That's great. He said: "I had a question about one of the items from SN-962" - that's last week - "the gold standard of client-side hashing for password creation."

He said: "In a scenario where the client submits their own hashed password, and the adherence to password requirements is governed only by client-side controls, would there be any way to prevent a malicious party, like a pen tester 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 counterproductive 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'd love to hear any thoughts you have on this, and thanks again for all you do."

Okay. So 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 of course is no. Assuming that the user's browser employs a strong local PBKDF, you know, a 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 upfront before the PBKDF function is applied because that's the only time it can ever be done.

And Efraim, he said: "Hi, Steve. Thank you for the great show. I'm a long-time listener and excited for the opportunity to continue listening for many more years. In regards to passwordless login by way of a link sent to a user's email and the concern over email security," he said, "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."

Okay. At one point the same thought had 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. In fact, it's absolutely strong. 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 log into. That cookie will be valid until the browser is completely closed.

And in fact the cookie's probably persistent and long-living, but it wouldn't have to be. And a bunch of information can be encoded and encrypted into the link beyond a one-time token, 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 just received from the site. When the user then clicks on the link, it will open a new page at the domain they're 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 been encrypted before it was added to the link and sent to the user.

So it first decrypts the information and verifies that a reasonable amount of time has passed since the link was created using the link's embedded timestamp. It verifies that the IP address encoded into the link matches the IP address of the browser's query. So the user hasn't moved. And that the first-party cookie the browser just returned with its query also matches the cookie value that was encrypted into the link. So it's the same browser. I don't see any way for that system to be compromised. You need no email security. You know, you could hand the link around to people, and it wouldn't matter. The IP address provides strong verification about the location and connection. 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 totally perfect security.

So thank you for posing the question, Efraim. I am very glad that we were able to revisit this once again. That makes it three weeks in a row since it's an intriguing idea. We've just made the email-only login system utterly bulletproof.

Leo: I like it.

Steve: Yup. It's a win.

Leo: So there is a rule with cookies that only the site that created the cookie can read the cookie. Right? So that's what...

Steve: Correct. I mean, that is rigorous.

Leo: Yeah. That's what's protecting you. So you get the email, or rather you get the link. Click in the email, click the link, it would open your browser. Now you're in that session, and you are theoretically with that first party; right? So the cookie then could work.

Steve: Right. You're back at that site domain where...

Leo: Right, it's first-party.

Steve: Which is where you want to log in. So it's first-party. So, and that link also could have encoded your IP address, which would not change from, like, minute to minute, right, because, you know, you have a connection to the site, and you say I want to log in here. Send me a link.

Leo: Right.

Steve: So it sends you a link. You open your email. You click on the link. And clicking on the link opens your browser back to that site. Well, your IP address hasn't changed. It's like, you know, 15 seconds went by.

Leo: Do we know that, I mean, people must be - I would think people are using that, in fact. Do you know if they are?

Steve: I don't know. But they certainly, I mean, so you...

Leo: That would be good.

Steve: Yes, you encode the timestamp, the user's IP, and their browser cookie.

Leo: Right.

Steve: And that locks that link to them.

Leo: It would only work with them.

Steve: Right now, where they are. Yeah.

Leo: I should ask, you know who uses that is Microblog. Let me ask him in Microblog if he's doing that because that's the only way to login, as far as I can tell, is you click a link, and it sends you an email, and you click the link in the email to reopen the site. I bet, he's smart, he's doing that.

Steve: It can be made - it can really be locked down.

Leo: Yeah.

Steve: And be made super secure.

Leo: Yeah. Oh, I'm going to ask. That's a good idea.

Steve: So Mykel, spelled M-Y-K-E-L, Mykel Koblenz, he said: "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 Mykel 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, let's call it an RFID chip to simplify it, in the key that needs to be physically 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" - like the slim jim, you know - "of unlocking a door when keys are locked inside, et cetera. 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 and theft. There are videos 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, as he points out. So thank you. And you're right, having the Canadian government as a consequence banning Flipper Zeros will obviously have no impact whatsoever upon auto theft. I would imagine that it's the how-to TikToks and the YouTube videos that provide the greatest impetus and explanation for the rise in Canadian auto theft. But, you know, what is any politician going to do about that?

Leo: Right.

Steve: ViperXX said: "Hi, Steve. Hello from Germany. Long-time listener, SpinRite license holder. The router topic." He said: "The company AVM, a very popular German router brand, actually does what you say. 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 a one-time password, an OTP token which lets you add it to your authenticator app."

So I just wanted to share that with the world. The company is AVM, a German router brand. And that is very cool. Let's hope that this heightened level of configuration security spreads since it might help to crimp the trouble that we are seeing with routers. And as we know, something really needs to be done.

Raed Iskandar, he said: "Hello, Steve. I was just listening to your response on our new Canadian ban" - he must be Canadian - "of the Flipper Zero." He said: "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 owner's driveway while their key is in the house." And of course we've covered this, too.

He says: "Once the car has started, the attacker just drives 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 victim's house. Or perhaps a physical kill switch on the car key itself so that 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." He says: "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."

Leo: Yeah.

Steve: "Looking forward to hearing your thoughts on this." Okay. So I'm very 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 actually are. Keys normally not working from a distance is a feature, not a bug. Right. 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 signal strength, which is what you would want. 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.

Leo: So I was going to show you this. This is the card key for my car. And newer BMWs use Apple's Car Key, they call it. So this is an RFID card. And you can see

there's even instructions, you know, tap it on the NFC. It is not, I'm sorry, RFID. It's NFC. So it doesn't work at longer distance; right? You have to tap it on the door. And then but the phone also has an unlock. My card key is in my Apple Wallet. And it's using UWB, the ultra-wideband.

Steve: Wideband.

Leo: Right, which is basically directional radar. And so it is, I think, immune to those kinds of replay attacks; right?

Steve: Very nice. Very...

Leo: Because it's UWB, yeah.

Steve: Oh, definitely, yes.

Leo: So I think this is - and by the way, it works so much better than the old Bluetooth car key in my Ford Mustang, which would fail all the time. This is infallible. In fact, it also works on my watch, which is nice. If I don't have my phone, my watch will get me in.

Steve: Very cool.

Leo: And I can drive. Although with the card key you have, because it's NFC, you actually have to put it in the location in the car because it has to be proximate to the NFC reader.

Steve: Ah, right, right, right.

Leo: So you put it in the phone charging tray, and it works.

Steve: So they provide a solution for people who have no additional Apple technology.

Leo: That's why the key.

Steve: Is there also something for Android?

Leo: Yes, it works in Android. I don't know if Android is as secure. I presume it is. I don't know how it works.

Steve: I don't think Android has ultra-wideband.

Leo: Has UWB? So it may not. May not be as secure. They also offer a fob for people, like old people like me, who don't understand how all this stuff works.

Steve: And who know what a fob is.

Leo: Yeah, I have two of them. Amazing.

Steve: So Emma Sax said, well, she provided some useful thoughts about meeting the need for throwaway email. Emma wrote: "I have a few comments regarding the email signups for tons of different throwaway 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," and she says, "Bitwarden currently integrates with SimpleLogin, AnonAddy, Firefox Relay, Fastmail, DuckDuckGo, 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 address to a random website.

"As Leo said," she said, "even if you use a single throwaway 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 address services are the best, or whether Bitwarden is the best password manager" - we think it is here - "but I chose a provider I trust for both my email alias service and my password manager, and I have not 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!." And so forth.

Anyway, thank you, Emma. We're glad to have you, too, as a listener - she said, you know, that she was happy that we were continuing - 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 little bit of a nudge in the direction of considering email integration. 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 said: "Hey, Steve. One remark about the 'click link in email to login to your account without password' feature mentioned in Episode 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 login to specific services you intended to share. Signed, Daniel."

And that's a great point. I love that. Instead of not wanting to share your email address, create a deliberately shared email account which you share with those who you wish to share login access with. Then the email loop actually makes all of that easier. You don't have to, like, keep a password synchronized among yourselves because you're just - you've already got email which is serving as your one-way of logging in. And of course it could be used for multiple accounts which all being shared among that group of users. Very nice.

Christopher Ursich, he wrote: "Steve," and he says: "Chris from Cleveland here, a listener since the days of The Onion Router, SecurAble, Jungle Disk, and the Astaro

Security Gateway." He said: "In SN-962 you gave a recommendation for client-side password quality enforcement. We need to deprecate website passwords entirely, but in the meantime" - and of course we know that's never going to happen - "in the meantime I think I have a better idea that is even easier for sites to implement." He said: "It should not be difficult to define a declarative microformat," and he says, "a.k.a. 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 user with cumbersome rules. Regards, Chris."

Okay. So I agree strongly with part of what Chris has suggested, and I think it's brilliant. Okay. So first, I doubt that the microformats.org that Christopher refers to as an example would be adopted in a world that's pretty much settled on JSON, J-S-O-N, JavaScript Object Notation, as its textual representation for structured data. Microformats date from 2004, so that's 20 years now, and it worries about counting and minimizing character counts because that was sort of its deal back then. That doesn't pack the same punch today as it would have back when the '90s 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 and constraints without any human intervention. When you're using a password manager, as I'm sure now everyone listening to this is, you know, 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 upon 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." What? 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 regenerate another password. So you do that, and now you're told that it must also have at least four non-consecutive numeric characters. Okay, 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 in the past.

We already have the well established, you know, it's in the root of a server, it's /.well-known/ directory which is used for locating website information in specific directories off the root. So we've got that in place. 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 - we could turn it on - 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. Gone, then, permanently, would be the need to constantly change the details of the password our password manager creates. It could always be set to maximum, and it would drop down to what a site said it was willing to accept, if necessary.

Anyway, I know it would be a heavy lift to get this adopted industry-wide. Remember, I'm the guy who spent seven years on SQRL. 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. So it would be beneficial to the site. And it would also make automatic password rotations, which are sometimes necessary, you know, where you want to change all your passwords, much more automatic because your new password wouldn't be violating that site's rules. 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 password for a site seems like a useful feature.

Okay. We're at page 19 in the show notes, which typically 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 the way I wanted to. The first one was a story that I thought was going to be the most exciting, generating actually some quite frightening, well, the story itself 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." It continued, "Chinese and U.S. researchers show new side channel can reproduce fingerprints to enable attacks."

Okay, now, what? The only problem with that is that it's not even remotely true. It turns out that within the fingerprint biometrics research community there are two generic fingerprint templates, one called "MasterPrint," and the other is "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. Okay? 1.88%, like this freaky MasterPrint that the industry has created turns out to sort of like be a generic fingerprint, and it'll work 1.88% of the time.

Leo: This may be more of a flaw in fingerprints. They're not unique. I mean, that's a...

Steve: They're not unique enough, exactly.

Leo: Yeah, yeah.

Steve: Exactly. Our fingerprint - well, and we know. You look at it, and you can decide, like oh, yeah, that looks like a fingerprint.

Leo: [Crosstalk], right.

Steve: That doesn't look like, you know, dust or entropy or something.

Leo: Or even a QR code; right.

Steve: Exactly. Okay. So that itself alone is interesting, the idea that there is this thing called a MasterPrint, which is known. Okay. And that it'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."

You know, it turns out it barely helped at all, although it took me a long time to figure that out because I had to read the research paper. But anyway, so much for that. If you saw that, and you wonder why I didn't talk about it, it's because it's nonsense.

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 its battle with time and entropy. I will make some time for a more detailed look at that next week.

And finally, 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 about Apple's PQ.

Leo: Yeah.

Steve: Adding post-quantum crypto to messages.

Leo: I was very curious what you thought about that. So, good. I suppose, I mean, PQ3 is not one of the NIST protocols. So I suppose they're using one of the NIST protocols. But we'll find all that out.

Steve: Right. Yeah, PQ3 is their own...

Leo: Their name for it.

Steve: ...encapsulation of, you know, how to do - and you've got to do key distribution and key rotation, I mean, basically what NIST is giving us are some ciphers. But as we know, there's a long distance between a cipher and an entire working protocol.

Leo: Yeah, yeah.

Steve: That has all the bells and whistles that Apple will need.

Leo: Pretty cool, actually. That's great.

Steve: So I think that next week's topic may just have three letters. Or characters.

Leo: I hope so. I hope so. Steve Gibson, GRC.com. By the way, our Discord, our wonderful Club TWiT members tell me that at least some Android phones do have ultra-wideband, including those new Samsung Galaxy S24s. Or, sorry, no, he has a Note 20. Wojo has a Note 20. So it's been around for some time.

Steve: Oh.

Leo: Maybe even predates Apple.

Steve: Maybe Apple's catching up, yes.

Leo: Yeah, yeah. I haven't tried the car key feature in Android. I have a Pixel whatever, the latest. I should try it. I'll get back to you. It sure is nice when your car walks up and recognizes you, and you just hope in, drive off. I love that. I love that feature. Steve is at GRC.com. Hop in. Drive off with a brand new copy of SpinRite 6.1. You'll be glad you did. It's official, and you can get your copy at GRC.com, the world's best mass storage maintenance and recovery utility.