SECURITY NOW!

Transcript of Episode #738

A Foregone Conclusion

Description: This week we look at another collision created by third-party AV; a powerful new Windows Defender feature that's easy to have missed; a public database breach by someone who should know better; what's worse than having all your files encrypted?; a VERY nice-looking, fully encrypted and free email service engineered in privacy-respecting Germany; stats coming back from Firefox's newly enhanced tracking privacy protection; a new and very bad remote code execution vulnerability affecting Nginx web servers; and the planned introduction of RCS to replace SMS next year. We also have a piece of SQRL news and some miscellany. Then we look at the outcome of a recent appellate court decision which complicates the decision about whether using a password or a biometric is more "judgment proof."

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-738.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-738-lq.mp3

SHOW TEASE: It's time for Security Now! with Steve Gibson. I'm Jason Howell, filling in for Leo Laporte one week more, and actually a couple more weeks coming up. So I'm with you in the long haul. Steve's got a bunch of stuff to talk about this week: a new powerful feature in Windows Defender. Asks the question, what's worse than having all your files encrypted? Takes a look at the new replacement for SMS, that's RCS that's coming sometime next year. And the big story, taking a deeper look at this thing we take for granted, that our PIN or our password cannot be compelled by a court of law. It's a foregone conclusion that Steve details next on Security Now!.

JASON HOWELL: This is Security Now! with Steve Gibson, Episode 738, recorded Tuesday, October 29th, 2019: A Foregone Conclusion.

It's time for Security Now! with Steve Gibson. I'm Jason, filling in for Leo, who is still out enjoying himself. And I'm going to enjoy myself joining you, Steve. How are you doing today?

Steve Gibson: Great, Jason. Great to be with you again.

JASON: Great to be with you.

Steve: Happy to see that the lights are on in Petaluma.

JASON: Yeah.

Steve: There's been some concern. I guess our listeners away from California are aware that there have been, like, crazy major fires all over California.

JASON: Yeah.

Steve: And not even only in the north. Now there's something that's endangering the Getty, I guess, and a bunch of celebs had to get evacuated from their homes in Malibu over the night.

JASON: The state is on fire, essentially. California as a state is on fire. Yeah, you know, and it's really - it's just kind of like when - like Petaluma, thankfully, is not part of all the evacuations that have happened, at least in the North Bay area. But a lot of towns and cities north of us have had to evacuate over the past couple of days. And it's just, you know, it's sad. I go out traveling to another part of Petaluma, and I just see these areas that are normally empty, and they're full of RVs, people who are just kind of like camping out, waiting for the signal to be able to return to their homes. It's just got to be so stressful, so bad.

Steve: Well, that sort of sounds like Palo Alto, where there's a major RV problem because you can't get housing in Palo Alto. So they're just lining the streets with RVs.

JASON: Right.

Steve: And of course then all of their waste disposal is a problem. And, I mean, it's just created a mess. Anyway, we are Episode 738 for this last podcast of October. What is this, I guess it's two days before Halloween.

JASON: This is our Halloween edition. Lots of scary news today.

Steve: So, well, this one is called "A Foregone Conclusion," which is an interesting phrase that was used by the prosecutor in an Oregon case that wanted to compel a 20 - she was 27 or 29, I can't remember, but a young adult woman who got into a car accident while under the influence of methamphetamine - and it involved six other people in this accident, so it wasn't nothing - to compel her to unlock her iPhone.

And what's interesting is the logical reasoning behind the decision that the appellate court in Oregon upheld, which changes the game. We've talked often about, just sort of from a theoretical standpoint, not that any of us listening to this podcast are actually going to have to worry about the police or law enforcement compelling us to unlock any of our devices. But as a matter of curiosity, we've wondered in the past whether you're better off using your fingerprint or another biometric like your face, or that being something that you are versus something that you know. So anyway, we're going to talk about that. Thus the podcast is called "A Foregone Conclusion," which, again, is part of the legal reasoning behind this interesting decision.

We're also going to talk, however, first about another collision created by third-party antivirus software. A powerful new Windows Defender feature that many of our listeners may have missed, and I wasn't aware of it until I encountered it, so I wanted to put it on everybody's radar. We've got a new public database breach by an organization who should know better. We sort of rhetorically ask the question, "What's worse than having all of your files encrypted?" And this actually happened to Johannesburg, South Africa recently.

We've got a very nice-looking, fully encrypted, free email service engineered in the privacy-respecting Germany that I wanted to also make our listeners aware of. And that actually was a result of something that came out of the Johannesburg attack. We've got stats now coming back from Firefox's newly enhanced privacy, well, newly enhanced anti-tracking privacy protection. So we're getting a sense for how effective that is. A new and very bad remote code execution vulnerability affecting Nginx web servers. That's one of the more popular new web servers on the Internet. It's got 30-plus percent adoption, and new sites are often using it. Anyway, it's got a big problem with the way it's invoking PHP.

We also have the planned introduction of RCS, which is slated maybe finally next year, in 2020, to finally replace SMS as sort of the base cellular provider, interphone, non-speech protocol and communications standard. So we've got to talk about that. We also have a piece of SQRL news; a bit of miscellany; and then, as I said, we're going to take a look at the outcome of this interesting recent appellate court decision which sort of changes the game a little bit in the decision, if I'm wanting to lock my phone for maximum protection, which course of action, which phone-locking technology do I use? So I think another great podcast for our listeners.

JASON: Absolutely. And maybe the answer is it doesn't matter what you use to lock your phone. There's always a way, apparently. Even the things you thought were foolproof or whatever will not be. Who the heck knows? We're going to talk all about that. All right. Starting off with a little SpinRite, right at the top; right?

Steve: Yeah, but I first want to just talk about this. I'm 100% bullish on the idea of synthetic credit cards. We've talked about it a number of times on the podcast through the years. There are a couple of card issuers that make that available. None of mine do, unfortunately, or I would be an avid user of that service. I just, and I think maybe once upon a time, maybe it was in the early days of PayPal, I remember once having access to it. And I just, you know, I've had so many of my credit cards through the years compromised, where you get the security alert, and somebody in another state just bought luggage, and it's like, whoa, wait a minute, slow down.

And so as a consequence, one of the reasons I route so many of my purchases through Amazon, aside from the convenience, is that I'm not having to give random websites spread all over the Internet my credit card information. And where I can, I will use PayPal, if a site offers that. I mean, I will always use it because it's just been such a hassle. In fact, right now I'm using a card that is new relative to the one I had for years because I got fraudulent purchases or attempted charges on the card that I had had for years.

And I've told the story on the podcast a number of times that normally, back also sort of pre-Internet, or in the early days of the Internet, I worked with a travel agent who I would just use because it was more convenient, even after the Internet. And I would annually go north for the holidays to visit my family. And it became routine for my travel agent, who had gotten to know me, to start off our conversation with, "Okay, first of all, Steve, do you still have the same credit card that you did last year?" The point was that I was, you know, often I wouldn't because there would have been a fraudulent charge, and I would have had to change the number, blah blah blah.

Anyway, so this is unsolicited endorsement of the idea of Privacy.com. I'm not saying this. Our listeners know because I've always been bullish on the idea of creating a synthetic credit card for a specific purpose. And I am a little anti "sign up and get a subscription," like one more thing that I have to subscribe to. So I like the idea that they have a free tier, which is probably where I would be because it sounds like it would meet - it's not like I'm having to do this all the time. But there are occasions where I don't have a choice. There's something I want. They don't offer PayPal. I can only get it from some random site. And they're not using a major credit card clearing house. They're just like, oh, like, trust us. And it's like, I don't know.

So sounds like these guys are a perfect go-between, where you can say this is the vendor. I'm going to give them this much money. Let this one go through and then kill this number so it can't ever be used again. So anyway, yay. I'm delighted to have an outfit like this as a sponsor for the TWiT Network. I just think it sounds great. So there.

JASON: I completely agree. I love the idea, especially when you talk about kind of like comparing it to password protection online and having a different password for every site. That way on the other side, if something is compromised, that one password, if it's ever discovered, it can't be applied to anything else. It's totally kind of a direct kind of comparison to the idea of having a different card for every service that you sign up for. I mean, they're very similar.

Steve: Well, actually, I would argue that if - I think you said "single use." And so that's more like having a one-time password.

JASON: Yeah, true.

Steve: That's even stronger than a password per site. You're saying I want you to authorize - you ask Privacy.com for a single-use card number. That's, I think, where they're getting this 12-per-month deal. And so you then give that to flaky XYZ site.

JASON: Dot com.

Steve: Dot com, exactly. And they're able to, you know, process the transaction. And, now, I don't know if there's like a per-card use fee. I mean, they're doing something in order to - maybe they're just presuming you're going to want to go over the free limit of 12 per month. Anyway, I'm going to find out because I'm definitely signing up for this. This sounds like the kind of thing I would like to have available for those instances where it's a perfect match of my desire not to give out my cherished credit card, at the risk of losing it again, out to the wild.

JASON: Absolutely.

Steve: So anyway, yes, as you were kind enough to say, we have a Picture of the Week.

JASON: Yes.

Steve: And this came to me, this was tweeted by a listener who said, well, I'm finally going to honor my longstanding promise to you to purchase a copy of SpinRite. And, okay, I don't know exactly what he's referring to, but thank you. And what's cool is he immediately got his money's worth. I mean, this is like - this is the classic perfect case for the thing that SpinRite does that nothing else does, which is here is a drive which, I mean, it's a good thing he didn't wait any longer. It is covered with those - and this is something that SpinRite users will recognize, the green R's which says there was a spot where the initial attempt to simply, lazily, casually read the data failed. It came back no.

And so SpinRite said, huh, okay, and decided to, as it does, to give that spot on the hard drive its full laser-focus attention. And it arranged to change the drive's mind, essentially, to perform a full perfect recovery of the data that was in that sector that previously refused to be read. It got it all back. And then it went on to the next one. And as we can see, what is that [counting to 26]. Okay, so at least 27 sectors. However, as is often the case, and you can sort of see how they're clumped up. You often have regions which will be bad. So there may well have been multiple actual sector recoveries which all collapsed to within a single one of those characters. On one of the other SpinRite screens, you're able to get a complete detailed count. And of course the SpinRite log logs each and every sector that it recovers and where it was on the drive.

So this sort of is a summary screen that gives you the 10,000-foot view of how the whole drive looks. But anyway, this was just such a perfect example. And I thank, first of all, our listener for following through with the commitment that he felt that he had with me. It may have been someone that said he really desperately needed a copy some time ago, and he would buy it later once he had the money, and could he get one - I mean, I do that sort of thing, if someone is in real trouble or who knows what. But in any event, I just thought this was a really nice example of, like, it doesn't get any better than that. He definitely needed to run SpinRite.

JASON: No kidding.

Steve: And it's probably good he didn't wait any longer.

JASON: No kidding. Yeah, that's like the sign of success right there, all R's. Love it.

Steve: Love it.

JASON: And that's got to feel good from your perspective, to create something that is so capable of basically saving people's butts when it comes to their own data. That's awesome. Time and time again.

Steve: It does really feel good. And, you know, and we've shared the stories of someone's wife was a listener, I mean, a guy was a listener, his wife retired and finally decided to pick up her long-on-the-backburner dissertation for her Ph.D. She got it all done on a laptop, which then crashed. And it was like years of work that of course had not been backed up, blah blah blah blah. And so he whips out a copy of SpinRite and saves the day for his wife. So, you know, those are just really fun, fun stories to share.

JASON: That's great.

Steve: Not so fun is the growing problem that we are seeing with third-party antivirus, at least on Windows. And we've talked about the cause of this. The cause is that today's AV, in order to continue to demonstrate its value to its customers, really has to hook its claws deeply into Windows. Microsoft now is arguably competing. We're seeing instances where third-party AV is causing Microsoft trouble. It's always been sort of controversial because, in order to get the access that antivirus software needs, it needs essentially to kind of be a rootkit. It needs to go in and make modifications to the underlying kernel level code in order to essentially insert taps into Windows for the things it wants to do. Well, the problem is, you know, I said it was rootkit-like. Well, Windows is trying to prevent rootkits. So, I mean, there really is a fundamental schism here which we're now increasingly reporting. And it happened again last Tuesday, a week ago, in an interesting way.

Chrome, Google's Chrome, began updating itself to its most recent release, 78. And there were a bunch of features there in addition to its 37 welcome security fixes, its experimental support for the increasingly controversial DoH, you know, DNS over HTTPS, which may be our topic for next week because, I mean, it's worth continuing to keep our eye on this because it offers benefits, but it is upsetting people. Also they've added page tab mouse hover dropdowns and a bunch of other goodies that may require some manual enabling on the part of users.

In addition to all that, an increasingly growing subset of users immediately began receiving Chrome's equivalent of the Windows Blue Screen of Death. In the case of Chrome, what you get is the "Aw, Snap! Something went wrong while displaying this webpage" message. I have a picture of it in the show notes. I've encountered it. Not often, but occasionally. In this case, many people began reporting that they were experiencing this error whenever they started the browser. I mean, basically something about Chrome's update to 78 caused it to stop functioning under Windows. Well, the trouble was quickly tracked down to a collision between something that Chrome was doing and systems running an older version of Symantec's Endpoint Protection (SEP), which is the enterprise-oriented security suite that Symantec makes.

It turns out that an outdated version has an incompatibility with a newly added feature in this most recent release of Chrome. And because Microsoft's own Chromium version of Microsoft Edge uses the same core, Edge had the same problem. The collision was the result of Microsoft's new Code Integrity feature, which checks the drivers and system files for signs of unauthorized tampering. And as I said, that's what AV is increasingly needing to do in order to continue to prove its relevance.

This new feature was added in Windows 10 1803, and it since has been rebranded WDAC, which is Windows Defender Application Control. With that rebranding came an extension of its protection to include all applications running on Windows. So essentially it's a whitelisting system, and the default is pretty much nothing is allowed to run. I mean, it's breathtaking. I mean, even Microsoft's own programs will not run unless they are explicitly whitelisted. There is now you're able to back it off one level to accept all the things that ship with Windows plus some of Microsoft's optional things like Office that don't ship with Windows.

But, I mean, this thing is really squeezing down tight. And as a consequence, it's a pain in the butt for many people to use. I mean, like it's going off all the time because it turns out, gee, that people actually want to run code that Microsoft didn't write themselves and sign. So anyway, it is a whitelisting system where enterprises that want strict control over what they permit their users to run have been asking for. The flipside is be careful what you ask for because, boy, you know, it is really tough to use.

So it turns out that Chrome turned this on and began using it for their own protection, and Symantec's Endpoint Protection apparently ran afoul of it. Server 2016 and Windows 10 RS1 are unable to have this fixed. So those two Windows editions will need to wait until Tuesday, November 12th, when Symantec will be fixing this. Existing Symantec Endpoint Protection customers not using Windows Server 2016 or Win10 RS1 can fix this by updating their outdated versions of Symantec's Endpoint Protection to the current version, if that's feasible. One wonders why somebody with Symantec's Endpoint Protection hasn't been keeping current. But whatever. If you update, then you're okay.

The other workaround for people for whom it may be unfeasible to update Symantec's Endpoint Protection is to disable that feature in Chrome, which you can do. If you edit the shortcut that you use to launch Chrome, you can add - and I have it in the show notes, the details. I'm sure you could probably google at this point. Well, actually I tried it, and it didn't come up very easily. The switch is -disable-features=RendererCodeIntegrity. Again, it's in the show notes. But the problem is that would be, and that's been what's recommended on the 'Net. The problem is that edits the link or the shortcut that you use to launch Chrome. If you ever click a link to launch Chrome, then it's not going to be using that shortcut.

So what I would recommend, although do so at your own risk because it's a registry edit, I also have in the show notes a DWORD which can be added to the registry under HKLM\Software\Policies\Google\Chrome. And then you add a DWORD. You add a registry entry of DWORD, 32-bit DWORD type named RendererCodeIntegrityEnabled. You thus set that to zero in order to globally cause Chrome to disable that feature. Then it will be compatible with the outdated version of Symantec's Endpoint Protection, and you'll be able to continue to use Chrome.

Once again, none of this is a problem if you just use Windows Defender, which is now rated pretty highly and works well in Windows. I get it that, if an enterprise has a Symantec Endpoint Protection license, that's expensive, and it's bought and paid for, and they want to use it, okay. When Leo and I talk to people now we just say just use Windows Defender. It's pretty good.

And recent comparisons of it versus other AVs, as I recall, the last time we talked about this and looked at it - it was a few months back - it tended to have a slightly higher false positive rate. On the other hand, I've never had it give me a false positive. And in fact, when I've let it look at a drive from the past, where I was deliberately having some viral samples off on a carefully labeled and sequestered directory, it's gone berserk. So I know that it sees things that are problems. I've never had it give me a problem when it shouldn't. So I like it, and it's what I use.

And so I will mention a Windows Defender feature that I just discovered as a consequence of digging around. It turns out Windows Defender has an Offline Scan feature which was one of these things where - it annoys me with Windows 10 that Microsoft is not leaving it alone because it's inherent in something this big that you're going to break things when you constantly mess with it. And so Windows 10 is now never having a chance to stabilize. It's never having a chance to settle because Microsoft's new idea is, oh, Windows as a Service. And so we're going to keep making it better. It's like, no, stop making it better. Leave it alone. Please, let it have a chance to calm down. But no, that's not the model we're in today. We're going to keep giving you new things, whether you want them or not, and in the meantime breaking everything that used to work. And so we can patch them, and that'll, I don't know, keep you updating.

Anyway, the point is Windows Defender has always had its traditional Quick Scan, the Full Scan, and then the Custom Scan. It recently received the Offline Scan as a fourth scanning mode. The reason this is significant is, and we've talked about the insidious nature of rootkits, is that it is possible that once Windows is running, and a sufficiently clever rootkit has sunk its hooks deeply into Windows, that even Defender, trying as it might, would not see something malicious on the machine.

The whole point of a rootkit is that you are using the operating system to look at the contents of your hard drives. Which is to say you are viewing your hard drives through the OS, through the API that the OS has given you. So what a rootkit does is hook those APIs in the OS kernel itself so that when you say "Give me a list of all the files on the directory," the malicious ones are edited out of the list that is returned. So then you go, great, I'm going to scan those. Well, great, but you didn't scan the malicious ones because you couldn't see them. You don't even know they're there.

So the point is Windows Defender now has an easily accessible Offline Scan. You're able to access it through the regular UI, the Windows Defender UI. There is now, I think it was, yeah, it's Scan Options which appears when you go through the Control Panel to Virus and Threat Detection. There's Windows Defender, and there's a Scan button. Below that it says Scan Options. A new one has just appeared, Offline Scan. And when you do that, I did it last night so that I could see for myself how it works, you are warned that the system is going to restart and shut down, and you need to save your work. And so yeah, yeah, yeah. And you do that.

And then the system reboots into the Windows Recovery environment, which lives on its own little partition. It's sort of a minified Windows which is used for performing maintenance tasks. And I have in the show notes a picture of the dialog that came up while it was running the offline scan. It's minimal, and it's a little text window, says: "Your PC is being scanned. This might take a while. When it's done, your PC will restart." And so then it shows when I started it in the show notes. It's at 5:35 p.m., and it had scanned 3420 items and was just about one quarter of the way through. It didn't find anything, happily, on the VM version of Windows 10 that I had running. I didn't expect it to.

But anyway, I just thought it was interesting. And here I am, complaining on one hand that Microsoft is changing things all the time, and at the same time saying, oh, look, we got a really great new feature. So, okay. I guess this is a great new feature. Certainly it can be. And so I wanted to just bring it to our listeners' attention that, you know, it's not the kind of thing you want to do all the time, but I would argue that an occasional full offline scan with Windows Defender, you know, you want to make sure that it's got the updated signatures, so update the signatures first. And then I would say, you know, maybe once a quarter? I mean, the point is you would always expect it to find nothing. If it ever did not find nothing, you would be really glad you ran that scan.

And the point is that the other thing that Microsoft talks about is they're less interested in talking about the fact that they may not be able to see something. I think that's the big advantage. But they also talk about how you may not be able to remove a rootkit while it's in place. That is, you just - there's no way to get rid of it because the nature of the rootkit defends itself. So if you are not booting Windows, if you are scanning a non-running Windows partition, then you find the rootkit before it's ever hooked into the OS because the OS isn't booted. And it makes it much easier, first, to find it; but second, to kill it and remove it from the system so that it never has a chance to sink its hooks in in the first place. So anyway, I do think it's a useful additional feature.

JASON: Is there any way to schedule that? I mean, I realize it doesn't have to happen very often. But I know at least for myself, if I'm not scheduling something like that, especially like once in a quarter or whatever...

Steve: You know, that's a really good question, Jason. I know that they're - and I don't know if I can bring up - let me see while we're talking about it because I probably have that on this version of Windows 10 that I'm talking to you on.

JASON: Yeah, LawnDog in chat says it's not available as a schedule right now.

Steve: Yeah. I guess I'd be surprised if it were.

JASON: I just know for me, if I don't schedule things like that, my chances of remembering three months later are slim to none. I suppose you can set yourself a reminder, but...

Steve: Yeah, and again, it's not the kind of thing that I think you need to worry about at that level. It's sort of like, when it occurs to you, like hey, you know, I haven't done an offline scan for a while, and then sort of like, yeah, it's probably worth doing.

JASON: Yeah, right, take the time.

Steve: So Scan Options. I think that the traditional Defender UI did have a scheduling feature, but I'm probably thinking of it under Windows. Check for Updates. Ransomware Protection. Allowed Threats. Scan Options. Yeah, there's Quick Scan, Full Scan, Custom Scan, and Windows Defender Offline Scan. So, yeah.

JASON: All right.

Steve: I'm not seeing - but again, it's not like it needs to be done on an absolute, oh, my god, don't forget to do this. It's just sort of like, hey, when it occurs to you, and you're going to go have dinner or something, then start an offline scan while you're off doing something else and just give it a clean look at your system. Seems like it's worth doing.

JASON: Like it.

Steve: So I titled this "All Your Database Are Belong to Us," which will be reminiscent to those listeners of ours who remember "All your base are belong to us" that was this weird Internet meme that happened years ago.

JASON: Oh, yes.

Steve: So even before I was skipping over the news, as I do now, of the ransomware attack du jour because I sort of like - we got a little crazy this summer with all of the pre-school startup ransomware attacks that were nuts pretty much through the Southeast of the country. I had already been skipping over the news of the publicly exposed database du jour because that's another thing which actually preceded the ransomware attacks is that this or that company would be found to have their database publicly exposed on the Internet. And it's like, what? How does this happen? I just - you've got to wonder.

And so, yes, we've talked about plenty of those in the past. But even so, when it happens to Adobe, such a major player, it still seems worthy of dishonorable mention. A data hunter whom we've spoken of in the past, by the name of Bob Diachenko, discovered a wide-open public and unsecured database belonging to Adobe, Saturday before last, on October 19th. The database contained the email addresses and other Adobe-specific information of nearly 7.5 million customers of Adobe's Creative Cloud. The database included the account creation date, the products that they're using, the status of their subscription, whether or not this is an Adobe employee, the member's ID, the country they are located in, how long it's been since their last login, and the payment status of their Creative Cloud account.

So it turns out that information affects approximately half of the believed size of Adobe's Creative Cloud customer database. However, what was not exposed this time was any obviously compromising information such as their passwords or their credit card numbers, payment information. On the other hand, what was exposed is exactly the sort of inside information - member IDs, products used, subscription status and so forth - that can be leveraged to make phishing emails far more believable and can therefore be used to induce users to download and run malware in a phishing campaign.

So once upon a time there was much less mischief that miscreants could get up to with this sort of data breach. Those times are long gone, and we see successful phishing campaigns leveraging exactly this kind of disclosure. And the bigger question we must ask here is how does this happen to an organization such as Adobe where security must be like an out-in-the-front requirement for any sort of work like this. How does a database containing 7.5 million of Adobe's customers get placed by mistake out onto the public Internet?

In its security updated dated last Friday, October 25th, Adobe posted this. They said of this breach: "At Adobe we believe transparency with our customers is important. As such, we wanted to share a security update," which is the way they couch this. They said: "Late last week, Adobe became aware of a vulnerability related to work on one of our prototype environments. We promptly shut down the misconfigured environment" - and by the way, they did, like immediately, within hours of being notified, to their credit. They said: "...addressing the vulnerability. The environment contained Creative Cloud customer information, including email addresses, but did not include any passwords or financial information. This issue was not connected to, nor did it affect, the operation of any Adobe core products or services." Okay, that's technically true.

"We are reviewing our development processes to help prevent a similar issue from occurring in the future. Should you have any questions, we encourage you to contact us at," and then the generic helpx.adobe.com/contact.html. And of course note that their transparency was not really optional since the news of this breach was widely published the same day. So to their credit, Adobe did take the exposed data offline immediately upon being notified of its exposure on October 19th. But they say that they're reviewing their development processes to help prevent similar issues in the future.

So one would have hoped that this was done after their very similar October 2013 breach, which impacted at least 38 million of their users, three million of whom had their encrypted credit cards and login credentials exposed. So maybe the previous security review wore off, and it was time to have another one so that they can figure out, again, how to keep this from happening in the future. I don't know.

But anyway, the problem is, as I said, these days, even though the information was not massively instantly exploitable, like login information, financial information and so forth, the big threat now is phishing attacks. That's the way users are induced to clicking on email. And in that database was a flag, if you were an Adobe employee. So for a phisher, that says, ooh, let's send a very convincing-looking phishing email containing information that only we at Adobe would have to our own employee, inducing them to click something in order to, who knows what. You could easily see that that kind of email would be far more likely to induce someone in Adobe to click a link and then get malware introduced into Adobe's network. So, yeah, it's not at all clear that these are - that despite the fact that there isn't any instantly leverageable information, that it can't now be used for a modern attack because that's exactly the kind of thing we see.

And Jason, before we go on, let's take our second break. And then we're going to talk about what is worse than having all of your servers encrypted with ransomware? Turns out there is something.

JASON: Uh-oh. All right, yes. We've got that coming up next. All right. So there is something worse, then. We've got more to look forward to here. More bad.

Steve: Yes. When would you wish it was just ransomware? And the answer is...

JASON: I mean, this is the Halloween episode, so this makes sense.

Steve: Ah, yes. When the attackers who got into your network stole all of your data, rather than encrypting it, and are threatening to expose it to the world unless you pay four bitcoins. Now, okay. Four bitcoins seems...

JASON: How much is that?

Steve: ...kind of reasonable in today's extortion game.

JASON: That's only $37,000 U.S.

Steve: Yeah, exactly, yeah. For a city? Yeah. So that's what just happened to the city of Johannesburg, South Africa this past weekend. They spent the weekend struggling to recover from its second cyberattack this year as key affected and infected services and systems were taken offline. And of course that meant that services were not available to the citizens and people trying to use their city services. The city first alerted users of the attack via Twitter on Thursday, October 24th.

And since then the note left by attackers, who call themselves the Shadow Kill Hackers, has been seen. And it reads as follows. The note reads: "Your city has been hacked. Hello, Joburg city! Here are Shadow Kill Hackers speaking. All of your servers and data have been hacked. We have dozens of backdoors inside your city. We have control of everything in your city. We can shut off everything with a button. We also compromised all passwords and sensitive data such as finance and personal population information. Your city must pay us 4.0 bitcoins," and they say in parens...

JASON: I love this part.

Steve: I know, yeah, I do, too, "...(that's a very small amount of money)..."

JASON: Parentheses, yeah.

Steve: Just in case you weren't sure, "...to the following address," and then we have a bitcoin address, "until October 28, 17:00 p.m. your time." So of course that was yesterday. "If you don't pay on time, we will upload the whole data available to anyone in the Internet. If you pay on time, we will destroy all the data we have, and we'll send your IT a full report about how we hacked your systems and your security. Contact us for more information at shadowkill@tutanota.com. Have a nice weekend. Shadow Kill Hackers Group."

JASON: Ah, they really do care, don't they.

Steve: They want the best for Johannesburg.

JASON: This is tough love here, is what it is.

Steve: So I'll note that while the English used by this group suggests they're not native English speakers, they do have very good taste in secure encrypted email services. Tutanota, a GDPR-compliant, German-engineered, client-side encrypted email service, with a useful free starter tier, does look very nice. They are extremely privacy centric and ask for donation support. So let's talk about them because we're going to wish Johannesburg good luck and hope that, I mean, I don't know what they chose to do. When I was putting this together yesterday, it was noted in the security press that was following this that there had been no recent bitcoin payment activity to that bitcoin address, which of course, as we know, the bitcoin monitoring technology is now rather mature, so it's possible to monitor activity on a given address and determine whether anything has happened. So it doesn't look like, as of the reporting that I saw yesterday, payment had been made.

JASON: Yeah, I actually just found an article on CNN that says they are refusing to pay. The date has elapsed. They say, "The city will not concede to their demands for bitcoins. We're confident that we will be able to restore systems to full functionality."

Steve: Well, and they must also be confident that maybe the hackers were bluffing.

JASON: Right.

Steve: Maybe everything is encrypted enough that they're not worried about what could be posted publicly. It'll be interesting to see whether anything does get posted; and, if so, if it's in fact damaging or not.

JASON: Right.

Steve: But let's talk about Tutanota because I've been aware of them in the past. I don't think I ever took a close look at them. But the fact that the Shadow Kill Hackers were using Tutanota as their email service sort of put it back on their radar. And I have to say I am very impressed. So, I mean, to the point where I could recommend these people without reservation, based on everything I've seen, to our listeners. It's T-U-T-A-N-O - now, I had "nova" somewhere. It's "nota," T-U-T-A-N-O-T-A, Tutanota.com.

So they say on their page: "Join us in our fight for privacy. In the future, Tutanota will be the privacy-respecting alternative for Google with a calendar, notes, cloud storage, everything encrypted by default. This is our dream of the future Internet, and we are truly amazed by the feedback and the support we have received from you so far." They said: "We invite all of you to get in touch and to support our goal in bringing privacy and security to the world."

And I'm just going to quickly touch on some of the bullet points to share what it was that they said that so impressed me. So they say: "Secure email for everybody. Tutanota is the world's most secure email service." Okay, now I would argue that, as we know, there are several of these. But these guys do look like they've got their hearts in the right place. They said: "Easy to use and private by design. Sign up for free to take control of your mailbox. Safe from attackers. With end-to-end encryption and two-factor authentication, your emails have never been more secure. The built-in encryption guarantees that your mailbox belongs to you. Nobody can decrypt or read your data."

They say: "We love open source. Tutanota is open source so security experts can verify the code that protects your emails. Our Android app is Google-free, making Tutanota the best open source email service." They said: "On your favorite device. With apps for iOS and Android, your secure emails are available any time. Our fast web client and fully featured apps make secure encrypted emails a pleasant experience.

"Free secure email without ads. We take your email needs seriously by offering you a secure email service, free from advertisements. We work around-the-clock to bring you an email service that lets you focus on the important things. From our clear and minimalist design to white label customizations, from our free version for personal use to our fully featured secure business email for companies, we are committed to making sure Tutanota is the only email service you will ever need."

They say: "We encrypt everything. We guarantee that your data in Tutanota is always encrypted, whether you use our secure webmail client, our apps for Android and iOS, or our desktop clients for Windows, Linux, and macOS." Now, that's interesting, a desktop client for Windows. I'm going to take a look at that myself. "Easily send a secure email with the knowledge that Tutanota encrypts subject, body, and all attachments for you. Import all your contacts into Tutanota's encrypted address book and rest assured that nobody else can access your contacts' personal information.

"Anonymous email. Privacy," they say, "is a human right. Everybody has the right to privacy and security. That's why the basic secure email account in Tutanota will always be free, with all security features included. No personal information such as phone numbers is required to access your anonymous email account. We do not log IP addresses and strip IP addresses from emails sent and received. Your right to privacy is at the heart of Tutanota."

And then they conclude: "Proudly engineered in Germany. Our entire team is based in Hanover, Germany. All your encrypted emails are stored on our own servers in highly secured data centers in Germany. With its strict data protection laws and GDPR, Germany has some of the best laws in the world to protect your secure emails. The German Constitution stresses the importance of freedom of expression and of the human right to privacy."

So anyway, I am impressed. It's hard to argue with those principles, which I know many of our listeners here share. So I wanted to put, as I said, Tutanota.com on everyone's radar. And I guess we could thank Shadow Kill Hackers for bringing Tutanota back to our attention. Either some listeners or I, I mean, it's not the first time I've heard of them, and they have been around for several years. But they really are saying all the right things. And I'm absolutely going to check out the Windows desktop app to see what it looks like and how it works.

I was just talking last week about - or maybe it was to Leo, so it must have been the week before - about the fact that I'm a Thunderbird user. Oh, yeah, because it was in the context of Thunderbird, a future version coming, I think it was next summer, will be integrating PGP into Thunderbird natively to dramatically simplify its use for people who want to use PGP-style encryption. And of course you do need to use an account at Tutanota if you want to use their encrypted email. But, yeah, from time to time it's clear that there must be some means for sending an encrypted email to someone who is then able to view it through their web browser without themselves needing to have a Tutanota app or account. So anyway, it looks like these guys have done everything right. And I just wanted to let our listeners know.

JASON: Tutanoted.

Steve: Duly Tutanoted.

JASON: Duly Tutanoted.

Steve: So Firefox's Privacy Protection. For just little old boring, "I never surf anywhere exciting" me, Firefox has blocked 3,080 web trackers since just September 4th of 2019, which is interesting because, again, really I'm not wandering around the 'Net very much. But Mozilla just released the broader figures to demonstrate Firefox's anti-tracking effectiveness. Their stats show that Firefox blocked 450 billion cross-site tracking requests since just the 2nd of July, shortly after enhanced tracking protection was first introduced for Firefox. And since that time the rate of tracking protection blocking has risen to 10 billion blockings per day. So what do you think, Jason? Time, perhaps, to give Firefox another look? You know Leo has switched back to Firefox; right?

JASON: So, I mean, I installed it, actually, on my desktop the other day.

Steve: Since we talked about it last week? Oh.

JASON: Yeah, we talked about it last week. Installed it again. You know, it wasn't very long ago that I still was actually using Firefox. But I wasn't using it isolated to itself. I was using Chrome on one screen and Firefox on the other. And I would log into one of my Google accounts on one browser, and the other one on the other, just because switching between the two accounts used to really suck.

Steve: Oh, interesting.

JASON: So having them siloed on their own was the way that I chose to do it. And so I was still kind of in the Firefox realm. But they've improved it, and so I've been using Chrome. So now I have it installed, at least. And so I'm going to challenge myself to kind of focus on it a little bit. I think one side of that is that this is a Chrome OS laptop that I use.

Steve: Ah, yes, yes, yes. So it does behoove you to stay with Chrome. I completely understand that.

JASON: There's a little bit of lock-in there, I think.

Steve: So I'll also note that Firefox is continuing to move forward with what they call "Lockwise," which is their built-in password manager. They were boasting that it can now generate a secure password. It's like, okay, well, good.

JASON: Welcome to the party.

Steve: When signing up for a new account. I would argue that that's a minimum requirement for a password manager is that it be able to do that for you.

JASON: Agreed.

Steve: And also that it can be used to replace a current weak password with a new, more secure one. They also note that access to local browser-contained Lockwise database can be protected using Apple's Face ID and Android's Touch ID biometric recognition systems. So that's nice. And it does look like they did everything correctly. They're using AES-256-GCM encryption, which is what you want to use. That's what I chose for SQRL, for example, which is a tamper-resistant block ciphering technology. They use the onepw protocol to sign into Firefox accounts to obtain encryption keys; PBKDF2 and HKDF with SHA-256 to create the encryption key from Firefox account's username and password.

So anyway, it looks like they've done everything right. And I guess my only problem is that I'm, like you, actually, Jason, I am not browser monogamous. I'm a bit fickle when it comes to browsers. Some things cannot be beat under Firefox, like managing a large number of open tabs with the tabs sidebar. I love having a bunch of tabs running down the side of my screen. There's kind of a wannabe sidecar add-on for Chrome to put tabs on the side, but I've never liked it. Firefox makes that, sort of builds it in. So I would argue that Chrome has its place. And I'm occasionally faced with IE or Edge.

So having a polygamous password manager such as LastPass allows me to keep all of those things that I'm using in a single archive, you know, all in one encrypted store, regardless of which browser I'm currently in the mood to use at the moment. So I don't think Lockwise is going to get me because I really do want to be browser agnostic for password management. But still, for those people who are only using Firefox, for what it's worth, Lockwise is continuing to move forward.

And I mentioned at the top of the show a very worrisome remote code execution affecting Nginx servers. That's N-G-I-N-X, for those who don't know. It appeared on the scene relatively recently, so it has the advantage of not dragging along Apache's long history, which means a growing, some would argue bloated, codebase, which is having to keep, you know, sort of slows down and makes it a little more accident prone. On the other hand, Nginx has had an accident. So that can happen to even new servers.

Saturday, this past Saturday, a new and dangerous PHP flaw surfaced. The very short version is, if your site or any site you are aware of and/or are responsible for, is Nginx web server based, and you're using PHP with the high-performance PHP-FPM feature in order to obtain optimal performance, you'll want to immediately, if not sooner, update to the latest versions of PHP. Under v7.3, that's 7.3.11. And under 7.2, that's 7.2.24. The reason is your site is very likely susceptible to a serious remote code execution vulnerability.

This PHP-FPM that I mentioned, FPM stands for Fast CGI Process Manager. It's an alternative means for sites to invoke the PHP script interpreter. It brings increased efficiency through reduced per-instantiation overhead, which helps to keep busier PHP-based websites from becoming bogged down by PHP invocation overhead, which otherwise can happen. If you're really busy with PHP, it can be the case that calling into PHP from the server has a substantial overhead relative to the amount of work that PHP's doing per call. So the first mitigation was the switch from traditional CGI to Fast CGI. And now we've gone an additional layer in to Fast CGI Process Manager, or FPM. I use it myself for PHP over on Windows IIS. So this vulnerability doesn't affect me. But if I was telling people about Windows IIS server, rather than Nginx, then I would be a lot more worried.

So the problem is here the Nginx version of this module, its interaction with PHP contains a newly disclosed vulnerability that could allow unauthorized attackers to remotely hack a site's server. The vulnerability, which is being tracked as CVE-2019-11043, affects websites with certain configurations of this PHP-FPM module, which is a common use in the wild because a proof of concept exploit for the flaw has been released publicly and has been shown to affect many sites in the public. The primary vulnerability is in environment path info (env_path_info) underflow memory corruption in that module. When chained together with several other issues, it allows attackers to remotely execute arbitrary code on vulnerable servers.

The vulnerability was spotted by Andrew Danau, a security researcher at Wallarm, while hunting for bugs in a Capture the Flag competition over the weekend. He stumbled on the problem, and then two of his fellow researchers, Omar Ganiev and Emil Lerner, developed this into a fully working remote code execution exploit. And of course we know that's the way these things tend to happen. First you crash the machine, then you figure out how to get it to execute your own code. Though the publicly released proof-of-concept exploit, which by the way is up on GitHub now, is designed to specifically target vulnerable servers running PHP 7 versions, the underflow bug also affects earlier PHP versions and could be weaponized in a different way.

So I've got the details in the show notes. I'm not going to go over it at length here. It involves the configuration of the invocation and the way the URL which is being fed to the FPM module has the script split off from the base URL. It turns out that you can play some games by putting a new line in the URL that can create a zero-length component after the URL is split, which a hacker is able to use in order to get their own code to execute. So even though this looks like it's very specific to this particular exploit, it turns out that it is the way PHP tends to be executed on Nginx machines.

For example, I had here in the show notes, and I'm quickly scanning - oh, yeah, here. One affected web hosting provider is Nextcloud, who we've been also talking about recently relative to personal cloud hosting services. They released an advisory yesterday warning their users that "the default Nextcloud Nginx configuration is vulnerable to this attack." So it recommends system administrators take immediate actions. A patch for the vulnerability was released just last Friday, and that was a month after researchers reported it to the PHP development team. So the PHP folks have known of it for a month. The proof of concept exploit, as I noted, is available, and the patch was just released.

So unfortunately we're seeing a window here where it's going to be very likely that lots of servers are not going to be updated in time. And this thing has been weaponized. It can execute the attacker's code remotely. And it's very easy to scan systems to see whether they're vulnerable. So once again, this feels like the kind of thing that we may unfortunately be talking about having been used to install cryptominers and malware, maybe yet even more ransomware, who knows.

But it's real, and if you are an Nginx-based PHP site, or you know of one that you have some contacts with, make sure that they've updated their version of PHP. This is a fix to PHP. It's not absolutely certain that every configuration would be vulnerable, but the most common ones are. And it's believed now that, I mean, that I saw figures like many, many tens of thousands of websites on the 'Net are currently vulnerable to this. So again, needs to get fixed fast.

So Jason, we have been living with SMS for quite a long time.

JASON: Indeed.

Steve: Calling it the Simple Messaging Service...

JASON: Little too simple.

Steve: ...probably is understatement, yes. It is embarrassingly simple.

JASON: Yes.

Steve: And of course, as we know, it's also been a source of many headaches for people through the years. On the other hand, it's been better than nothing. Imagine that you could not send an SMS message. I mean, imagine if there was, like, nothing but voice. That would not be good either. So here's the story. Maybe. I mean, they're promising it. We'll see. Next year, in 2020, all four of the major U.S. mobile phone carriers - so that's AT&T, Verizon, T-Mobile, and Sprint - will, they are announcing, finally be joining forces to launch an initiative to finally replace SMS with the long-awaited RCS mobile messaging standard. We potentially and hopefully care about this because the nationwide switch over to RCS would serve to finally, at long last, raise the incredibly low lowest common denominator for cellular content exchange well above where it currently sits with SMS.

As I said, it couldn't be any worse. The bar couldn't be any lower than sending a limited size ASCII message that's not encrypted, not authenticated, can be hacked because the carriers, the global carriers are still using the SS7 switching system which is known to have flaws. Well, the biggest among them is that there's no authentication in the SS7 protocol. So, I mean, if you were designing something to be bad, you could not do a better job of designing something that was bad. And that's what everyone's been using down at the SMS level.

So this initiative is also working with its carrier ownership group and other companies to develop and deploy this new RCS standard in a new text messaging app for Android phones that is expected to be launched next year in 2020. However, because it got tired of waiting, earlier this year Google independently released RCS messaging for Android smartphones in two countries, the U.K. and France. So that effort will presumably be integrated into the formal RCS, if and when it actually does happen next year.

So the goal of this joint venture, which is called CCMI, Cross-Carrier Messaging Initiative, because wouldn't it be nice to have it be cross-carrier, is finally to get the four major cellular carriers off their butts to deliver GSMA's Rich Communication Service, thus RCS, which is an industry standard, that is to say, an unadopted, unused, not yet deployed standard, to consumers and businesses, both in the U.S. and globally. Although the RCS standard was developed more than a decade ago, it's never been adopted widely due to the quagmire of mobile carrier and phone maker politics.

And, you know, we have an example in front of us which is Apple, which has no particular interest in RCS because it's already offering more than that through iMessages. On the other hand, they're going to have to, in the same way that they do allow SMS to interact with iMessage, thus the green balloon in iMessage, they'll have to allow RCS to interact with their system, as well.

JASON: Certainly hope so.

Steve: Yeah. Well, yeah, I'm sure they will. And of course iMessage is, as we know, end-to-end encrypted, which RCS is not and cannot be. A few carriers and services have struck out on their own so far to offer non-universal versions of this new messaging standard. But those are, because they're non-universal, they're limited to the exchange of RCS-based messages to the subscribers of their own network. So that hasn't been very useful. I mean, okay. You have to be conscious of whether - I guess it would be useful for, like, if you knew your friends were on the same carrier, and among family members if you were in the family plan on the same carrier. But great. It's got to be universal.

So what is it? Unlike our now ancient SMS technology, RCS-based enhanced text messaging service supports high-resolution photo sharing, location sharing, group messaging, animated stickers, read receipts, and some of the other features which are all very reminiscent of Apple's iMessage. So in that sense it does significantly raise the base level, lowest common denominator messaging functionality that comes installed on phones by default. And we should remember that there is also a large lower tier feature phone market lying underneath the smartphone market. Those will likely be the greatest feature recipients of this change, since smartphones have obviously had all these features for quite some while using their own protocols.

Speaking of which, unlike iMessage, Signal, WhatsApp, Threema, et cetera, as I noted, RCS messages are not end-to-end encrypted. They do, however, contain message verification and whatever this is, they called it "brand certification mechanisms," to ensure that users are interacting with legitimate brands to protect them from fraudulent accounts, impersonators, or phishing. Also in RCS's favor is that, whereas SMS communicates over the SMPP protocol, using as I mentioned the insecure SS7 switching system, all RCS traffic between the device and the network can be protected using SIP over TLS encryption. So we will get endpoint authentication and communications privacy through encryption.

The news reporting on this notes that the CCMI project has not yet fully developed its RCS-based messaging standard. Whereupon, reading that, I thought, really? After more than a decade? Okay. So it's not yet clear at this moment if the major U.S. carriers will implement protections for their users' privacy from government surveillance. But I'm sure that many of us would be willing to take the bet that that won't happen at this late stage, where the government's campaign against the absolute privacy within its borders is so strong at this point.

There's just no way that cellular carriers are going to be able to stand up to the U.S. federal government and themselves incorporate secure end-to-end encryption, if we even believe they have the ability really to do so. I mean, we know that Google can. We know that systems like iMessage and Signal and WhatsApp can because they've been developed by serious security research and cryptography work. There's no sign that AT&T and their ilk have any similar interest in doing that.

So I don't think we're looking at secure end-to-end encryption for the feature phone market from this. However, the announcement does say that it will "enable an enhanced experience to privately send individual or group chats across carriers with high-quality pictures and videos." So it's not clear what "enable an enhanced experience" means. Sort of sounds like corporate speak written by a PR flack who is also unsure of exactly what the system would do. But at least it sounds good. So who knows. We'll see how this rolls out. We'll see next year what it brings.

What I think it means is that for the non-smartphone market the lowest common denominator which is already being used in feature phones, which is text, that's going to get a big boost. So that's probably a good thing. I don't think it'll have any impact on the higher end smartphone market, where users who can have encryption and want encryption already have it. And we already have all of those other features - group chat and high-resolution photos and videos and FaceTime and everything. So it's like, okay.

Still, it's useful. At some point we're still using SMS for delivering one-time passwords; right? I mean, we're trying to move away from that. It's still being done. So switching that to encrypted authenticated SMS messaging or SMS-equivalent messaging, well, that would be a good thing. It would marginally increase the security. Still not a good solution because you still have all the SIM swapping problems and all of that. But bringing up the lowest common denominator certainly makes some sense. So if it happens, we'll get that.

One bit of SQRL news and two bits of miscellany. I learned this past week that there is a site called SQRLFor.net, S-Q-R-L-F-O-R dot net, as in SQRL for .NET. And the site says: "SQRLForNet is middleware for implementing SQRL in your ASP.NET Core solution. It offers a fully protocol-compliant server-side solution with a high degree of customization. Our goal is to enable any and all ASP.NET Core sites to enable SQRL login for their visitors." So in other words, yet another server-side package for SQRL.

And this actually, first of all, I was delighted to learn of this work. And it does further demonstrate one of the aspects of SQRL that I've argued would really help its adoption, which is it is just not very difficult to bring up SQRL support on the server side. Arguably, the large body of complexity exists in SQRL clients, but we only need a few of those. We need a good one for iOS, and we have one. We need a good one for Android, and we have one. We need a good one as a browser extension, and we have one. There is work on a native Linux client underway.

The point is that, once we've got the few clients that we need covered, then that handles the client side. And the fact that it is relatively low effort, comparatively, to implement SQRL on the server suggests that adoption's going to be much easier. And it's toolkits like this that reduce it to being a slam dunk. For example, there's a plugin for WordPress, which means that more than half the sites on the 'Net that are WordPress-based can add SQRL with a single click over at WordPress.org.

So anyway, there's lots of ongoing news, and so I just wanted to keep our listeners in the loop with what's happening over there. And Jason, we could take this opportunity to remind our listeners that I will be up in Petaluma in the TWiT studios on Saturday, the afternoon of Saturday, November 30th, to have the SQRL story formally recorded by all of the great TWiT producers over at your end.

JASON: That's right. Anyone, I think that there's still room for this, but...

Steve: Oh, how could there be. Good, good.

JASON: Regardless, either way, tickets@twit.tv. There might not be. Honestly, I haven't heard in the past couple of days where it's at. But tickets@twit.tv is where you should email if you want to inquire and see if there's room in the studio. If you happen to be in the area, and you want to be here for that, you can. That's 2:15 in the afternoon Pacific time on November 30th. It's a Saturday, like you said. So tickets@twit.tv.

Steve: I think we're going to have some sort of hangout afterwards, I think, also.

JASON: Right, like a meet-up of some sort.

Steve: Yeah, yeah. And again, please don't sign up if you think you cannot make it.

JASON: Right.

Steve: Only do so if you're absolutely committed to being there because we'll know who you are.

JASON: Mm-hmm.

Steve: If you signed up and didn't show.

JASON: That's right. We got you.

Steve: Okay. Two pieces of miscellany. Tim Kington tweeted. I got a kick out of this. He says: "You made a serendipitous mistake during the podcast this week" - meaning last week's podcast - "and said 'securious.'" He said, "I loved it. I think it should be the byline for the show: 'Security Now!, Podcast for the Securious.'"

JASON: I like that.

Steve: So, yeah, I think that's pretty good. Thank you for catching that.

JASON: Has a nice ring to it. That should be on a T-shirt.

Steve: I kind of remember that slip of the tongue. Security Now!, the podcast for the securious. And Dr. Flay's posting in GRC's Security Now! newsgroup. He posted: "The official Checkra1n domain" - remember we talked last week about the forthcoming CheckM8 publicly available exploit that we're expecting, which will allow anybody with a phone, from an iPhone 4S with the A5 chip up through and including the iPhone 10 with the A11 chip, it'll mean that it will be possible always to jailbreak any of those with a USB connection once that's developed. Remember that I mentioned that it was weird that "Checkrain" spelled correctly, R-A-I-N dot com, had been immediately commandeered by an annoying download-and-install-for-pay site, sort of a scam site; whereas the Check R-A-1-N dot com was the authentic one.

Anyway, Dr. Flay noted that now Checkra1n.com have wisely registered - oh, first of all, it is live with a valid cert, and it gets an SSL Labs grade A certificate, and they have wisely registered other possible squatting options - checkra.in, checkra1n.io, checkra1n.dev, and checkra1n.net. So anyway, he says - oh, and then he also said: "I gave the scam site an appropriate review in WOT, though it will need more reviews before triggering any warnings in the reputation systems." And then he put a link there to the mywot.com page, where our listeners who are interested could also go in and down-rate that site in order to hopefully begin, hurt its reputation so that browsers will begin warning users. But anyway, thank you, Dr. Flay, for the update.

JASON: Yeah. Super important. A foregone conclusion. What in the heck? What have we got here?

Steve: So the subtitle for this discussion should be "Supreme Court, where art thou?" because the way our court system works, as we have seen and talked about a number of times in the past, lower court judges are judges, and they get to make their decisions sort of like based on what law they're able to find. But when there isn't any clear law, they've got to rule based on what there is. And of course that gets appealed. It gets challenged. Ultimately the Supreme Court is - obviously we call it the Supreme Court because that's the final arbiter of these things.

So I wanted to give a recently announced Oregon state appellate court decision, which addresses the question of whether forcing the disclosure of a memorized device unlock password is in fact tantamount to compelling self-incriminating testimony against one's interest, which thanks to the Fifth Amendment to the U.S. Constitution is a protected right. The privilege against compelled self-incrimination is defined as "a constitutional right of a person to refuse to answer questions or otherwise give testimony against himself."

We've many times discussed this issue, since at this point in the development of the consumer product industry we broadly have two common ways of unlocking our devices: as we discussed last week, biometrics, something you are, whether your face or your thumbprint or your knuckle print or whatever; and the more traditional password, a passcode or a swipe sequence, all of which can be lumped under a secret, you know, considered to be a secret, which is something you know.

In the U.S., as I mentioned, we have this problem where, when there is no ultra-clear, constitutional, black-letter law, nor any more recent ruling by the ultimate Supreme Court of our land, the lesser courts and judges are left with the freedom and the burden of attempting to interpret what has come before in order to create a foundation and a context for new cases which are brought before them. So this has all boiled down to are we safer against forced device unlock using a password or a biometric?

And the upshot, as we've discussed in the past so far, has been that people who have memorized their passwords are safer because they're protected by the Fifth Amendment against being forced or coerced in any way to testify against themselves by divulging their devices' unlock password. In other words, telling someone your password has been considered to be testimonial in nature. And I would argue that, if it's your password that is your secret, then ultimately, whether or not the court rules for or against you withholding it, you still have the right to withhold it. I mean, you could be held in contempt of court, right, if you don't give it up. But you may decide that, well, that's better than letting them see what's in my phone. So it seems very clear that using a password is the better solution.

And of course we should also note that there is sort of a hybrid here, right, because if you do the panic act on your iPhone, I think you click the power switch rapidly, like five times in a row or something, that immediately puts it into - it disables the biometrics, requiring that you use your password. So you sort of can get the best of both worlds. On the other hand, if something happens like you're in a car accident, and you're unable to get to your phone, and the authorities are able to get to it, then you don't have the ability to do that.

So anyway, the problem is this is not something, this whole issue of is your password something that can be compelled is not something that the U.S. Supreme Court has yet ruled on. So the lower courts are flip-flopping around. And another flip-flop happened last week. This Oregon appeals court last Wednesday decided that a woman whose judgment was impaired by methamphetamine when she crashed into a tree, seriously injuring one adult and five children passengers - and actually a different adult, so I guess she injured herself, a different adult, and five children - they ruled that she can be forced to unlock her iPhone with something she has previously committed to memory.

The appellate court's decision states that in their opinion, which is their ruling on the matter, it is not a violation of her Fifth Amendment rights against self-incrimination because the fact that she knows her phone's passcode is a foregone conclusion.

Okay. So that's interesting and new. This was a 27-year-old woman who crashed her car into a tree in Salem, Oregon, injuring herself, her friend, and five children in the back seat. She did not want to give police any help in building a criminal case against her. They wanted to search the contents of an iPhone they found in - her name is Catrice Pittman, so found in Catrice Pittman's purse. But she never confirmed whether it was hers and wasn't offering her passcode. Her defense attorney argued that forcing her to do so would violate her rights against self-incrimination under the Fifth Amendment of the U.S. Constitution and Article I Section 12 of the Oregon state Constitution.

But a Marion County judge sided with police and prosecutors - and this was a clever prosecutor who came up with this - and ordered Pittman to enter her passcode. The ruling was appealed. Then Wednesday the Oregon Court of Appeals agreed with that ruling in a first-of-its-kind opinion for an appeals court in Oregon, and apparently anywhere. The ruling will likely make it easier moving forward for Oregon police to compel access to the contents of suspects' cell phones and all of the massive amount of personal information they contain, you know, photos, videos, past Internet searches they've conducted, text messages, phone contact lists of everything they have been in contact with and when and so forth.

So the deputy public defender who represented Pittman on the appeal, an attorney named Sarah Laidlaw, she said: "The upshot of this case is when police have a warrant to get information off a cell phone, and the government knows the phone is yours and you have the passcode, then the court can compel you to enter the passcode."

Ryan Scott, a criminal defense attorney in Portland who's not associated with this case, but closely follows appeals cases, said the ruling is an example of a continuing erosion of rights. Ryan said: "Our rights are a little less than they were yesterday. But for those of us following this area of law, it's not a surprise." He said that federal law has been leaning in this direction for the past few years.

Scott said the ruling won't affect many Oregon defendants whose phones are seized by police because police already have technology that allows them to crack into most of those phones. But sometimes, as apparently in Pittman's phone, police cannot get in. Scott said that the latest iPhones, more than other phones, have proven the most difficult. Scott volunteered that, from his experience, for people who want their information private, he would recommend getting an iPhone. And he added that: "Apple is not paying me to say that." And of course we know for many reasons that Apple puts a premium on trying to keep that stuff private.

JASON: For sure.

Steve: Pittman had argued through her attorneys that punching in her passcode would amount to testifying against herself because doing so would effectively admit that the iPhone was hers, and that she knew the passcode. But the Court of Appeals ruled that because police already had good reason to believe the phone was hers, given its location in her purse, the fact that she knew - and here's sort of the crux of this. The fact that she knew its passcode was already a foregone conclusion. In other words, she could be compelled to cooperate as an exception to her constitutional rights.

In doing some digging around additionally, I saw that this odd-seeming foregone conclusion standard has been coming up a lot recently in these compelled unlocking cases. It allows prosecutors to bypass Fifth Amendment protections if the government can show that it knows that the - I've got too many "that's" - that it knows that the defendant knows the passcode to unlock a device. And if I ever needed another reason to be a coder, rather than an attorney, this is it, since this makes my head hurt.

But the reason we need the Supreme Court is that, as we've covered in the past, not all courts, nor even all appellate courts, have ruled in the same direction on this decision. So we need someone to make this final. Of course we talked about this before. An example of the decision that came out the other direction was the Florida Court of Appeal just recently, in November of 2018, regarding a case very similar to that of Pittman, it involved an intoxicated minor who crashed his car, leading to the injury or the death of some passengers, then refused to unlock his phone for police.

In Florida, the court refused a request from police that they be allowed to compel this underage driver to provide the passcode for his iPhone because of the "contents of his mind" argument that invoked the Fifth Amendment. But the Florida court even went beyond that, saying that whereas the government in the past has only had to show that the defendant knows their passcode, with the evolution of encryption, the government needed to show that it also knew that specific evidence needed to prosecute the case was on the device - and in fact we talked about this aspect of that at the time - not just that there was a reasonable certainty the device could be unlocked by the person targeted by the other, which is to say they had to demonstrate they had some cause for knowing that there was evidence there.

If prosecutors already knew what was on the phone, and that it was the evidence needed to prosecute the case, they didn't prove it, the Florida court said at the time. From the order to quash the passcode request, the court wrote: "Because the state did not show, with any particularity, knowledge of the evidence within the phone, the trial court could not find that the contents of the phone were already known to the state and thus within the 'foregone conclusion' exception."

So here again, I mean, this invoked the "foregone conclusion" and said, okay, we know about that, but you can't show us that you have reason to know what it is that you expect to find, so no. Like I said, this is a mess. Oregon found differently. Even looking at this prior Florida case law, the appellate court and the lower court both ruled that "foregone conclusion" applied. So this is going to go back and forth until eventually one of these things makes its way to the Supreme Court and we get someone to decide for us one way or the other.

As I said, it looks to me like, if somebody were really concerned about this, then using a passcode is the right way to be safe in situations like this, rather than the convenience of a biometric. And as we also talked about last week, it's not clear to me that a biometric provides and affords the degree of protection that we hope it does. But it certainly is easier to use.

JASON: How, in that situation, you've got a phone, you've got a PIN on it, and say you are in the position to say I don't remember my PIN, I mean, most of us would probably say that we know the PIN on our phone. But it's not impossible for someone to not remember their PIN.

Steve: That's true.

JASON: Maybe it's super long, and in that moment they don't. So then in the case of this foregone conclusion, like, what then? Because you legitimately can't remember, you're prosecuted against?

Steve: Yes. So that's when they hold you in contempt, and they put you in a cell and give you some time to try to remember it.

JASON: Go ahead. Think about it real hard.

Steve: So, you know, we will - you're not going to like the food.

JASON: Right.

Steve: You're probably not going to want to stay here for long. So just do try to wrack your brain.

JASON: Yeah. Isn't it convenient that after all this you finally remembered.

Steve: Yes. And so, let's see. With a PIN, if you - I forgot how this works. If you don't guess correctly, then you have to power cycle the phone to try again? I mean, it can't lock you out forever; right?

JASON: I think it depends on the phone. If I remember, the iPhone - and I've never been, like, a long-term iPhone user. But if I remember on the iPhone, if you enter your

PIN enough wrong times, every time you do it, it adds more time. And, I mean, that could get so long that you just - it could be months.

Steve: Yeah. I think something bad happens if you exceed some number of guesses. I know it does.

JASON: Yeah. At a certain point it could erase all data, right now they're saying in the chat room.

Steve: Oh, that's what it is, that's what it is. Yes, yes, yes. It doesn't evaporate or teleport from your hand, but that's what it does. It does the full erase of the phone to protect itself from - and actually, these days, since there's an on-the-fly cipher, an AES-256 cipher always in line between the iPhone's drive and the phone, all it has to do is wipe the key. So it just - it instantly zeroes that key, its RAM-based storage of that key, and at that point you need to go back to Apple in order to you, you know, like if you do an iCloud backup and go through all those rigmaroles, or restore from home and so forth.

JASON: Right, right.

Steve: Very cool.

JASON: Well, this show is a foregone conclusion. I don't know. I don't know if that makes any sense, but it is a conclusion because we've reached the end.

Steve: I just thought that was a real interesting legal argument.

JASON: Yeah, it is.

Steve: You know, it's your phone. It's in your possession. We know you use it, so you know how to unlock it. So it's a foregone conclusion that you have this information, and that means it's not testimonial.

JASON: Right.

Steve: So it's like, okay, I guess Denise Howell can explain this to us in more detail. But like I said, I'm glad I write code and not interpret the law because, ouch.

JASON: Indeed. I completely agree.

Steve: Computers are much more logical.

JASON: To you, for sure. This was excellent stuff. I'm really happy also that you got to talk a little bit, shed some light on the RCS stuff. That's been, yeah, I feel like that's been jumping back and forth between Google and the carriers for the last couple of years now. And at least there's some sort of light at the end of tunnel as far as that's concerned. I'm not sure if it's Google's preference as far the way it went down. I think Google probably wanted the carriers to jump onto their solution.

But we recorded All About Android early last night, and Ron Richards on the show made the point that, like, no matter what, if it's implemented, and it improves things - kind of like you said. Even if the baseline gets improved as a result, and they're all supporting it, it doesn't matter which direction it comes from. Anything is probably better than SMS at this point. And so I'm looking forward to it.

Steve: Does Google have a native end-to-end encrypted messaging? Or is it always an add-on? [Crosstalk] Signal and WhatsApp and, you know.

JASON: Yeah.

Steve: But, like, iMessage is built into iOS. Is there a built-in to Android? I guess really it would have to be Android-centric; right? So you wouldn't be able to send an iMessage. No, you wouldn't be able to send an SMS to somebody over on Apple world.

JASON: Yeah. I mean, Google Hangouts was their messaging app for quite a long time. And it does have encrypted conversations. It doesn't use - I don't know enough about how it handles the encryption on Hangouts to know.

Steve: So, for example, it's not wrapped in Signal or one of the mainstream strong encryption...

JASON: No, no, no, no. It's not.

Steve: End-to-end encryption service.

JASON: And, you know, Google's been moving to push Hangouts out the door, basically replacing it with Android Messages with RCS. So I would say the answer is no. Like if you really want truly encrypted messaging, you're not looking at Google for that right now.

Steve: Or you add another, you know, your own layer.

JASON: Sure, sure. But Google isn't really putting out a solution right now that appeals to that, to that market.

Steve: Right, right.

JASON: Honestly, Google has just dropped the ball repeatedly as far as messaging is concerned, and it's exhausting. As an Android user, it's really exhausting. You want something that works. And I've just gotten used to using, like, 10 different messaging apps for different reasons with different people, and, yeah.

Steve: Oh, goodness. So it's like, oh, wait a minute, she has this one, and he has that one.

JASON: Right.

Steve: So you...

JASON: Totally.

Steve: Whoa, yeah.

JASON: And strangely, it makes sense, like I've done it long enough now that I know who's where and all that stuff. But it shouldn't have to be that complicated. And on iOS it's not. And so that's what's really frustrating. But anyways.

All right. We've reached the end of this episode of Security Now!. And Steve, awesome stuff, as always. Anyone who wants to check in on what Steve is up to and inform yourself on everything Steve Gibson, go to GRC.com, find everything you expect to find there. You're going to find SpinRite. You're going to find information about SQRL. Audio and video of this show is also there, as well as that's the only place that you'll find transcripts of the show. So if you want to find a transcript, just go to GRC.com.

And our website that you're seeing right now is TWiT.tv/sn. There you can also find all the audio and video, all the subscription links. If you want to watch on YouTube, or you want to subscribe through Apple Podcasts, Google Podcasts, all the major ones are there, and you can just link right to it. Or just watch them in the page. That works, as well. But we always recommend that you subscribe so that you don't have to think about it. It just delivers to you like the podcast magic that it is. If you want to watch us record this live each and every week, it's Tuesdays, 1:30 p.m. Pacific, 4:30 p.m. Eastern, 20:30 UTC. I'm sure that's changing anytime now. I know the times are about to change here. And that's TWiT.tv/live. And when you do that, you can be a part of the chatroom, and chat about all these topics while Steve is talking about them.

So Steve, great job today. Thank you so much. And we will see you next week on another episode of...

Steve: And we will see you next week. And the week after. So thank you.

JASON: That's right. That's right, two more weeks.

Steve: Thank you, Jason. It was a pleasure.

JASON: Me, too. Right on. We'll talk to you next week on Security Now!. Bye, Steve.

Steve: Bye.