Cryptocurrency Antics

Description: This week we discuss today's preempted Second Tuesday of the Month, slow progress on the Intel Spectre firmware update front, a worse-than-originally-thought Cisco firewall appliance vulnerability, the unsuspected threat of hovering hacking drones, hacking at the Winter Olympics, Kaspersky's continuing unhappiness, the historic leak of Apple's iOS boot source code, a critical WiFi update for some Lenovo laptop users, a glitch at WordPress, a bit of miscellany (including a passwords rap), some closing-the-loop feedback from our listeners, and then a look at a handful of cryptocurrency antics.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-650.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-650-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got lots to talk about, some fascinating facts about cryptocurrency, a little more about Spectre and Meltdown. And we'll continue to find a name - or look for, anyway, I don't know if we'll find it - for the Spectre and Meltdown flaws in Intel. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 650, recorded Tuesday, February 13th, 2018: Cryptocurrency Antics.

It's time for Security Now!, the show where we cover the latest security news, help you protect yourself, help you really deeply understand how computers and technology work, and it's all thanks to our Explainer in Chief, Mr. Steve Gibson.

Steve Gibson: Leo, great to be with you again for the - I don't know why this number hits me. I mean, like 666, that would be a good one, too. But this one is 650. I guess just the roundness of it.

Leo: I know why. Because Bill Gates famously said no one would ever need more than 640 Security Nows. Right?

Steve: That's 640K, that's right, 640. No, nobody needs more than that.

Leo: Nobody would ever need more than that.

Steve: As I was pulling things together this week, there were a surprising number of interesting stories that involved cryptocurrency in one way or another. It's been a topic for us because, as we know, dating from our first description of how the bitcoin block chain works, then sort of cryptocurrency went silent for a long time, and recently it's just, like, everybody's talking about it, and my mother is asking whether I should have her - or once asked. She's not asking any longer. But a friend of hers was wondering if she should invest in bitcoin, and Mom called up and said, "Honey, do you know what a bitcoin is?" I thought, yeah.

Leo: It's so funny, my dad sent me an email saying, "I'd like to find out more about this block chain thing."

Steve: Yeah. So there's, like, some bizarre fun stories that involve cryptocurrency. And so as I was putting everything together, I realized, okay, there wasn't any single big news event which we normally put at the end and acquires the title of the podcast. So this one is Cryptocurrency Antics. And we have five interesting sort of antics that we'll talk about as we wrap up the podcast, but lots of other interesting news, I think.

We've got what turns out to be sort of a preempted Second Tuesday of the Month because this is it. This is one of those months where - and we always sort of note them - where the first of the month is Wednesday, which means that the second Tuesday is as late as it can possibly fall - which is the 13th, interesting - in the month. Which is normally the big patch time. But as we mentioned last week when we talked about Woody's enumeration of the fact that 15 out of the 30 days or 31 days in January had updates in them. So anyway, so we've got the preempted Second Tuesday of the Month.

Tracking the slow progress on Intel's Spectre firmware update front. A worse-than-originally-thought Cisco firewall appliance vulnerability. I knew about this last week, but it looked like the conditions where it would be a problem were restricted, and it didn't really affect end users. It was an enterprise problem. Well, since then it's sort of exploded, so I thought, okay, we've got to talk about that. Also, MIT Technologies Journal introduced this notion of an unsuspected threat from hovering hacking drones. And I thought, well, okay, that sounds kind of fun. We also have hacking at the Winter Olympics to talk about, Kaspersky's continuing unhappiness, and the historic leak of Apple's iOS boot source code, which...

Leo: I was hoping you'd talk about that, yeah.

Steve: Yeah, got to talk about it. And what I love about this story is how it's the classic way that this sort of happens where it wasn't malicious, but it was inevitable. So it's sort of fun. Also there's a critical WiFi update for some Lenovo laptop users. A glitch at WordPress, that's a sponsor, and so it's sort of an interesting automatic update thing that I want to make our listeners aware of. Then we have a little bit of miscellany. Oh, and I have to have you play, because I think we're going to have time, an interesting rap on passwords. Don't know if you saw it. But if you want to scroll down while I'm doing some of this and take a look at it.

Leo: I'll prepare to play it, yes.

Steve: It is SFW, meaning it is safe for work. And it's really just - it's cute. It's a few minutes long. It's posted over on Facebook, and the link is here in the show notes under Miscellany. Then we'll close the loop with some of our listeners. Most of the feedback was people suggesting names in response to my musing last week that it was annoying to have to say Meltdown and Spectre and Spectre and Meltdown, and we really need a concatenation somehow. So we've got every possible variation that you could imagine that our listeners have provided. And then we're going to take a look at some cryptocurrency antics, including finishing up with one that just makes you do a face plant. It's just - it's too amazing. So I think a great podcast for us.

Leo: And I have an Image of the Week to join your Image of the Week.

Steve: Oh, yay. Cool.

Leo: Yeah. I sent it to you, but maybe you didn't see the email. I know you don't like to look at email.

Steve: Actually, I saw it, but I didn't - I don't know, I guess I didn't click on the link, or I didn't find the image. But I had seen what you were referring to, so I'm glad you'll be able to provide it.

Leo: Well, we had mentioned it last week. And the guy who actually created it, or knew the guy, it was created in his office, sent us more information about it. So I will pull that up, as well. All right.

Steve: So our Picture of the Week ties into the second thing we're going to talk about at the top of the show. And this is just another fun, I mean, people having a lot of fun with the Intel logos, and we've been having fun with them during the various, well, all year so far, since this whole Meltdown and Spectre nightmare. And so this is just - it's the Spectre Inside logo that we talked about last week. This time there's a little circumflex, and it says "still" inside, "Spectre Still Inside." And that is the case.

In fact, I'll just jump over, and we'll talk about the second Tuesday in a second. I went looking for the podcast for the latest news from Intel, and it's rather quiet. The only thing I could find was another press release written in the first person. I'm always a little jarred by this, when Navin Shenoy at Intel writes in the first person. But on February 7th he posted: "Security Issue Update: Progress Continues on Firmware Updates," which, oh, that's good, they haven't given up. We don't want them to give up.

Anyway, so Navin said: "Intel continues to work closely with industry partners to protect customers against the security exploits disclosed by Google Project Zero. As I shared" - which is always the part that jars me a little bit - "January 22nd, we identified the root cause of the reboot issue affecting the initial" - and remember this is the guy who was saying they were rebooting more often than usual, like what? Anyway, at all would be bad - "affecting the initial Broadwell and Haswell microcode updates. Since then, we've

been focused on developing and validating updated microcode solutions for those and other impacted platforms."

And then he said: "Earlier this week, we released production" - now, this is news - "production microcode updates for several Skylake-based platforms to our OEM customers and industry partners, and we expect to do the same for more platforms in coming days. We also continue to release beta microcode updates so that customers and partners have the opportunity to conduct extensive testing before we move them into production."

So last week Skylake mobile and desktop firmware was released. So as we remember from this debacle, the first microcode update attempted to add instruction set features to give operating systems control over branch prediction, which is this aspect of the speculative execution and branch prediction which was what Spectre was able to leverage. Those first firmware updates were releases for Broadwell, Haswell, Skylake, Kaby Lake, and Coffee Lake processors. But we quickly learned that Broadwell and Haswell systems were having trouble. And then we later determined that all of them were - Skylake, Kaby Lake, and Coffee Lake also.

So that shut down all of this firmware. It all got pulled back. In some cases, new BIOSes were made available to put the old firmware back. We know that Microsoft ended up producing an official patch which wasn't pushed that their customers could go and get to set the bits in the registry to disable their use of these new firmware instructions. And as I mentioned last week, it turns out that the same bits that GRC's little InSpectre app sets if you want to deliberately disable the mitigations for these.

So at this point the latest news I have been able to find is that last week Intel did release firmware for Skylake, but for no other major processor families. So the expectation is that those who have Skylake-based systems may shortly, after the OEMs verify probably more carefully than they did last time that the firmware is not causing trouble, may be able to get firmware updates.

So that's where we are. This is taking a while. And again, I'm impressed that Intel can do something this significant with firmware. I mean, as I explained a couple weeks ago, firmware in a microprocessor is not like software, application-level software where you can write word processors, you can write graphics programs, you can write image compressors, I mean, you could do anything because it's meant to be a general purpose instruction set that the software runs.

Firmware, that microcode firmware is not a general purpose instruction set. It's a way of simplifying an incredibly complex hardware design by moving some of it essentially into a masked ROM which allows some sequencing to manage the flow of data through pathways in the chip. And it's easier to do it with some data patterns. And what Intel provides is the ability to override that built-in ROM with a microcode patch.

But the idea of coming along after the fact and arranging firmware to have control over something you never intended it to control, I mean, this is a major design revision to the processors. And so I'm impressed that they can do it at all. I mean, they haven't done it yet, apparently. So I guess they have on Skylake. But it is an architecture, an intensely architecture-dependent thing.

It's possible, and I'm not predicting this, and I'm way on the outside, so I have no inside knowledge, but I wouldn't be surprised if they said we can't fix this on this or that processor. That is, we can't fix it in firmware. The firmware doesn't have the flexibility. That wouldn't surprise me. I would say, okay, well, that's entirely reasonable for that to

be true.

So it's going to be interesting to see how this progresses. The last thing Intel wants to say is a fundamental problem has been discovered in our chips that we can't fix. But the idea that these chips are even fixable is something we sort of take for granted. But they may not be. So it's going to be interesting to see how this plays out over time. At this point, Skylake, they've released the firmware for that, and we'll be waiting for firmware for the others.

And I mentioned that the Second Tuesday of the Month is today, and that as a consequence of the crazy startup that we've had this year, we got the February security rollup update for our systems last Tuesday, the 6th. And it wasn't clear to me, because we've had so many of them, I mean, it's been a week. Which is longer than we went all through January. But, okay. I fired up my machine and looked for updates, and nothing came.

So it looks to me like the only thing that we are getting today was what Microsoft did push, which was an update for the Adobe Flash zero-day which we discussed last week. And so that was made available today. But it looks like Microsoft is going to sit tight with the major rollup that we received for February last week. So a week ahead of schedule, ahead of their normal schedule.

I mentioned also Cisco's problem. Late January, so a couple weeks ago, the news hit that there had been responsible disclosure to Cisco from a Cedric Halbronn with the NCC Group that they had discovered a very worrisome vulnerability in a family of Cisco enterprise class, like kind of big iron firewalls, they're collectively known as "adaptive security appliances." ASA is the acronym.

And at the time it was initially believed that only the VPN component of those ASA devices - and there's a family of them, but the initial list was like maybe five or six of them of a much greater number were vulnerable. And that it only affected some configurations. And I remember thinking, okay, well, this doesn't really impact our listeners directly. It's not like consumer routers or something.

Well, since then, several things happened. The number of affected devices has more than doubled, and the list of subsystems, the subsystem components went from one, meaning the VPN, to 11. So all kinds of things in there are vulnerable. And it has now become attacked in public. We are now seeing hackers actively exploiting the vulnerability in these devices. And there is something on the order of 200,000 of them publicly exposed. These are inherently Internet-facing firewall appliances from Cisco, so their job is to be out there on the front line, keeping people from getting into your network.

Unfortunately, it itself is vulnerable. And this has got an exploitation score of 10 out of 10. So it's critical severity. And the 10 out of 10, in order to get 10 out of 10 it has to be, essentially, as bad as it can get. It's easy to exploit so it has a low level of attack code complexity. It can be exploited remotely and requires no authentication on the device. So the good news is this was disclosed responsibly. Cisco made the patches available.

On the other hand, as we know from all the machines, all the Windows machines, for example, servers that are still unpatched, as we were talking about last week, months and months after important vulnerabilities had been made available, anybody running in an enterprise environment that has one of these adaptive security appliances needs to make sure that they're current to their current patch level because it is now not just a known problem, but it is actively being exploited in the wild.

MIT's Technology Review had an interesting story about cyberwarfare taking to the skies aboard drones. And when I encountered that, I thought, what? I think I remember sometime ago we were having fun with the idea of a drone that had some sort of a projectile launcher on it, and we were joking that, if the projectile was too heavy, the drone would get pushed backwards from recoil when it tried to launch this. In this case, these are radio relay drones. And it turns out they exist in actuality. It is a new mode of attack, the idea being that there are some instances where people improperly believe that, if they've got unsecure radio, they are safe from attack, that is, just due to the fact that, oh, well, the range isn't that great.

A perfect example was a demo that was given at one of the hacking conferences recently where a drone, one of these hacking drones was flown outside of a building and was able to pick up the radio signal of a wireless mouse. And although you'd have to watch the signal and do a lot of reverse-engineering in order to figure out what was going on, you could certainly from that get position and clicks. And potentially that's an information leakage. It doesn't seem that worrisome to me. But this is, of course, as we know, this is always a way that these things start.

Leo: And a wireless keyboard would be more informational.

Steve: Exactly. And we know that, as we had talked about, some of the early ones just used pathetic encryption. I mean, you can't even call it encryption.

Leo: ROT13 or something, yeah.

Steve: Yeah, well, yeah, in fact it was - they're putting out 8-bit ASCII, and they came up with a random 8-bit mask which just XORed the bits.

Leo: But they do use the same mask for every keyboard, right, yeah.

Steve: Yeah, yeah. As soon as you put the battery in, it sets this XOR mask. And it's like, okay, wait. I mean, in fact, that would be a nice final exam question for CS 101 or an Introduction to Computer Security is here's a stream of text which is from a wireless keyboard with a static 8-bit XOR. What was being typed? Because frequency analysis immediately collapses this encoding. And as soon as you - the interesting thing about this, I mean, how bad that is, is as soon as you identify like what the T or the S or the E is, as soon as you can figure out one character...

Leo: Right, you're done.

Steve: ...then you immediately know what the mask is, and the entire thing collapses. So Leo, that's...

Leo: It's a convenient thing about XOR. It goes both ways.

Steve: Yes, exactly. And so your example, Leo, is perfect. So what is being done is that there are now experiments being conducted with drones being used to move essentially a spy outpost into an unsuspected location. So it's easy for people to think, oh, well, we're out in the middle of nowhere. We wouldn't have to worry about securing our WiFi or using secure phone communications and so forth. Well, it's just worth planting the seed that lack of proximity to radio is no longer any true security, and that spy drones sucking in radio are becoming a thing. Yikes.

The Winter Olympics, as we know, are underway as we're doing the podcast, and so is Olympic hacking. There haven't been any clear stories yet to surface from this except that it is the case that tensions were expected to be lower with North Korea because they're part of the Olympics now, and so there's not the heightened problems with diplomatic relations. On the other hand, Russian athletes were banned from participation.

And so it was expected and has turned out that Russia's hacking efforts were running full bore, and that there were multiple nations' security teams who were in place for months beforehand and have been fending off and observing attacks against the Olympic network's infrastructure that have been going on. For months beforehand systems were being probed. And it's looking more like they're trying to create embarrassment, like of individuals or the actual operating of the system in real time. So no huge events from that. We'll have to see. Maybe a couple weeks from now we'll have more information. But it is an obvious target for attackers, and they're not missing the opportunity.

Also, yesterday, Kaspersky Lab filed another lawsuit against the U.S. We talked about Kaspersky's unhappiness with the U.S.'s decision late last year to back away from the use of and in fact moving forward to forbid the use of their antivirus tools over concern that there was some Russian state government influence over what was going on. We talked about the news that apparently it was as a consequence - which seemed inadvertent and may well have been inadvertent - of Kaspersky's AV picking up some code samples and phoning home with them, that that was a way that there had been some exfiltration of confidential data from systems that were being protected by Kaspersky's AV.

So this lawsuit filed just yesterday is related to this ban on the U.S. use of Kaspersky's products. That was part of the 2018 National Defense Authorization Act that the Trump administration signed into law at the beginning of October. And in the court documents filed yesterday in the U.S. District Court of D.C., the District of Columbia, Kaspersky is claiming that the ban is unconstitutional. Their lawyers are saying that under the U.S. Constitution's Bill of Attainder clause, Congress is forbidden from enacting laws which impose individualized deprivations of life, liberty, and property, and inflict punishment on individuals and corporations without judicial trial.

So I understand, and we've talked about, their unhappiness. But it's difficult to forgive our agencies from being, I think, reasonably cautious over this idea. As I have said for quite some time, I'm surprised that Microsoft Windows is as widely deployed globally as it is because it's a closed system. And apparently Microsoft does make source code selectively available in some instances to major states where they have, like, they have to show what they're doing in order to get the okay. But we've moved to a mode now where, as we know, Windows 10 systems no longer have any choice about accepting updates. So Microsoft literally has an institutionalized software implantation system in place today. We have no reason to believe that Microsoft would ever or has ever abused that; and it would be, I think, the last thing that they would do. But the facility is there.

So given that we've got increasingly mature alternatives, open source alternatives, it's just difficult for me to see how anything but inertia is responsible for continuing to use

these sorts of systems. So anyway, it'll be interesting to see. I doubt anything will come of this. I think that - I'm sure that our governments, our state and local and national governments have a right to choose which software they wish to use. It is creepy to have antivirus systems installed which maintain a connection back outside of the country, and specifically to Russia, that have the ability to send questionable software back for analysis. That seems like something you want to think twice about.

And the story, Leo, that you wanted to talk about, and I'm glad you were wanting to go into this because I think this is really interesting, Apple had the source of their iOS boot code posted on GitHub last week, and it's not surprisingly called iBoot, which is the code which first runs as the system powers up in order to form the anchor of the chain of trust which brings the whole system online. And we've talked about the way secure boot technologies in general work, and Apple's is no different. That is, you start with some highly trusted code which is probably in ROM or is signed, and its signature is verified in a way that cannot be interfered with, intercepted, contravened, you know, just there's no way to subvert that.

That code comes up, makes sure that it has not been tampered with, and then it goes and loads the next stage in the boot sequence and, similarly, after obtaining it from storage, but before ever executing it, absolutely verifies that it, too, has had its integrity maintained and is then safe to transfer control to. Control then transfers to it, and it repeats that process, essentially forming a chain of trust where each link is independently verified from the code that's already running, prior to turning control over to it. The point is that the entire integrity of the system depends upon the anchor, that original code. And so it is significant that the source for that was posted last week.

Now, Apple has sort of downplayed this a little bit. In their statement they said: "Old source code from three years ago appears to have been leaked, but by design the security of our products," writes Apple, "does not depend on the secrecy of our source code." Well, we know that that's good. "There are many layers of hardware and software protections built into our products," they write, "and we always encourage customers to update to the newest software releases to benefit from the latest protections."

At the same time we also know that this kind of core code doesn't change very much. I mean, this was part of iOS9 back in 2016 when this code leaked. And it's certainly the case that, now we're at 11, there may well have been some changes. There may be some, and probably are, some new hardware in the later, yeah, some new security hardware in the later iPhone hardware. So it's probable that the iBoot code was tweaked a bit.

At the same time, we also know that a lot of code doesn't change. Windows users are seeing dialog boxes, if you drill down underneath the UI very far, back from Windows 2000 that haven't changed. If you bring up the network adapter details dialog, that's the dialog from Windows 2000. That core code that is done tends not to change. So Apple's understandable sensitivity is not only to the fact of the leak, but to the fact that, while their statement suggests correctly that they're not depending upon obscurity, it is also the case that bad guys or phone hackers will be salivating over the idea of having access to this iBoot source because by attacking the source code from the standpoint of trying to find something that was done wrong, you can just get a lot of ideas, a lot of potential leads for where to attack.

So the interesting thing about this from a sort of a reality of keeping things private is that an intern, what was described as a low-level employee in some of the coverage, and I also saw the term "intern" used, a couple years ago had some friends in the phone hacking, in the jailbreaking community. And he was at Intel, he or she, don't know, I'm

sorry, I keep saying Intel, was at Apple. And the friends were pushing and pushing and pushing to have this person share this iBoot source and a bunch of other tools, as well, that this person had access to.

So this was a small group of trusted individuals. And I should mention that Motherboard had really good reporting and coverage of this. They interviewed, under promises of anonymity, a lot of these people who had firsthand knowledge of how this happened. So there was a very tight-knit small group, friends of this intern, who did receive this iBoot source code two years ago under the promise that it would not be shared outside their little circle.

Leo: I'll give it to you, man, but you've got to swear, swear you don't give this to anybody else.

Steve: Exactly.

Leo: You gotta swear, pinky swear, man.

Steve: And to their credit, I guess, that promise endured for two years. But the code existed nevertheless on thumb drives or on hard drives outside of Apple. And some secrets are just too good to keep. They're just too difficult. And so you can imagine that when it was new, when it was fresh, when it had just been received, it was, oh, nobody else has this. This is super cool.

And so the people outside of Apple had their own proprietary interest in keeping it safe. But those kinds of secrets are hard to keep. And so they wanted to share it. One of them - and within this group, they don't today know who leaked it outside of their little group. But presumably someone did. Somebody had a friend and of course said, okay, I'm going to share this super cool code with you, but you absolutely have to double pinky swear that it goes no further. Oh, of course, of course, of course. And of course, who knows, maybe it was that person.

The point is that some secrets are just too hard to keep. And so through essentially what amounts to social engineering and insider disclosure, this code got out. This wasn't a flaw in a highly academically scrutinized cryptographic cipher where, oh, my god, if you number crunch this thing for 12 CPU centuries, then there was a chance that you could find some information leakage. No, this was an intern who was at Apple who had access to this and did sign an NDA, of course. There was some mention of that in the Motherboard coverage that the person of course, I mean, he had to sign a confidentiality agreement in order to be there and to have this kind of access. But that didn't in fact prevent it from escaping.

So it's not clear that this, I mean, this is not probably devastating. It is true that the other side of this code being as mature and unchanging as it probably is, is that it is probably mature. But Apple did immediately use a DMCA takedown order to get this yanked from GitHub. It also was somewhere else, I can't remember, where it was removed from. So basically Apple immediately moved to clean this up. And I imagine Apple has a security team that is probably hot on the trail of how this happened, working to remove as much of this from public access as possible. But it was on GitHub for, like, the fact that it was there at all means it disappeared, I mean, it got loose. So it is now out in the jailbreaking, the greater jailbreaking community.

Someone whose handle was ZioShiba, Z-I-O-S-H-I-B-A, was the pseudonym of the person who posted this on GitHub. And among the people who knew of the original leak it was identified as a "copy," whatever that means. But what we understand it means is that this was a subset of the total leak package. So there was more that was leaked than ended up being posted. So maybe the disclosure from the original small group was less than all that was posted, or who knows what path the leak took to get to GitHub. But it was there, and it was up for some length of time. It got a lot of attention.

And so I don't have access to it. I have no interest in it. But I'm not inside that community. And you have to imagine now that there are lots of eyeballs scrutinizing the code, looking to see whether, oh, look, you know, here's an unsafe type conversion where, if we're able to get a negative something in here, blah blah blah, who knows. At the same time, this is not code that probably interacts with the outside world much. It's not accepting random parameters and things. The fact that it is as root as it is means that it is probably not subject to external manipulation. On the other hand, it is knowledge which Apple did not want made available, and it did get out.

Leo: Yeah, I mean, the interesting - there are a couple interesting conversations that are unknown. You raised one, which is how much is this code modified? Apple says it's old code. But as you point out, why modify iBoot?

Steve: Right.

Leo: I didn't look at the code, so I didn't know how complex it is or how hardware-dependent it is. This came from iOS, what did they say, it was 2014, so it would have been iOS9, I think.

Steve: Yup.

Leo: But the other one is - and obviously it's security through obscurity. And of course you don't want to give it away. But how much damage is done? Jonathan Levin, the guy who verified the code, said yes, well, I've disassembled the iBoot, and it matches my disassembly. So what would you get? You would get symbol names. You'd get a symbol table that you wouldn't have that with a disassembly. That's about it; right?

Steve: Yeah. When you disassemble, you're inherently making lots of guesses.

Leo: Okay.

Steve: And so you're doing a lot of reverse-engineering. Sometimes stuff happens like with specific hardware registers, where bits are being poked and stuff.

Leo: Right, don't know, yeah, right.

Steve: And so that's like an unknown. But if you've got the names of those, the actual nomenclature that goes along with that, there's a lot more that you're getting. So you can really - you can look at it both ways. I don't think it's the end of the world by any means. And frankly, I'm surprised this doesn't happen more often. I mean...

Leo: Well, it has happened once before, I think; right?

Steve: Right, right. But still, I mean, the idea that a company the size of Apple with as much intellectual property as they have, that this is a major event because it's considered an unprecedented leak. That alone is impressive, an impressive statement about Apple's security. I think they're doing a good job overall.

Leo: Yeah, yeah. Why that intern had access to the source code is another question.

Steve: That's, yeah. You can imagine there'll be some internal surveilling of their processes.

Leo: But it may be an indicator that they didn't care that much, you know, like saying, yeah, well, fine, if it gets out, it gets out.

Steve: Yeah, it could be. So last September we covered the news of a big concern in the Broadcom WiFi chipsets. This affected Apple devices and Android devices, all of which use these Broadcom chipsets. And our listeners will remember that you didn't have to be associated with an access point. You didn't have to be connected to WiFi. It just had to be a specially formed packet that your mobile device could receive that allowed it to be taken over.

So everyone ran around, and it was important to get patched, and we talked about again another example of the necessity of our devices over time being maintained somehow, that the lesson keeps being the nature of this is that these are computers, and they need to have some sort of an ongoing maintenance facility of some kind.

Well, it turns out that there are a subset of Lenovo's ThinkPads also using this Broadcom, it's the BCM4356 is the chip. The wireless LAN driver for Windows 10 is also and similarly vulnerable. The vulnerability is rated 10 out of 10 for exactly the reasons I just enumerated. It's remote, requires no user interaction, and can potentially be used to perform remote code execution attacks. But it does, as was the case with the previous mobile devices, require you to be within radio reception range proximity. So it's not like the North Koreans are going to get you when you're in South Dakota. You're safe.

But it does mean that, if this were to become prevalent, somebody within radio reception range, somebody in the coffee shop where you are, or on the train, or in the airport or wherever, could exploit this driver in a select set of laptops. And those laptops are the ThinkPad 10, the L460, the P50s, the T60, T60p, T460, T460p, T460s, the T560, the X260, and the ThinkPad Yoga 260. Those and only those devices have updated WiFi driver available, and owners of those should go to Lenovo, or most of our ThinkPads now have a Lenovo check to make sure you're updated and current and safe option from Lenovo. So either go get it, or if you have a ThinkPad that I just named off there, they've got many, many more. For example, our X1 Carbons are not among those. But I don't

know if your Yoga is a 260, Leo.

Leo: I have a second-generation Yoga, so I'll be checking. And I have a T470s. Was that in there?

Steve: No. Not in there.

Leo: Yeah, because I don't think anything when I run [crosstalk].

Steve: So they have so many models that this is just a small subset. But it is worth making sure that you've got the latest wireless LAN driver, just to be on the safe side, because it's regarded as a 10 out of 10. Again, more likely that you would be targeted for attack. So if your profile is such that you're known to be carrying that laptop, one of those laptops, and you might be a target for attack, definitely want to make sure you're...

Leo: Actually, when you run the Lenovo Vantage software, which is now what they use to do updating, it says right on the front, "Lenovo WiFi Security, protect yourself from malicious WiFi networks." Oh, that's not related. Or is it? Oh, that's something that you can turn on. That's something else, though, which I really like.

Steve: Ah, okay.

Leo: That lets you kind of - I don't what they do. It's not a VPN, but they have some sort of thing that they do.

Steve: Extra goodie.

Leo: That's not - the extra goodie. Yes, that's not [crosstalk].

Steve: Since WordPress is a sponsor, I wanted to let our listeners know that there was a small glitch last week in the release of WordPress v4.9.3 which last week was the latest. They patched a total of 34 known vulnerabilities. But unfortunately, in the process of installing the updates, the automatic update mechanism itself got broken. Which meant that WordPress would no longer be doing automatic updates.

So I'm not sure what mechanism they have for notifying their users. They probably - you probably get email if you are a registered WordPress user who's receiving this. But the problem is that since 4.9.3 broke the update mechanism, it's necessary for WordPress admins whose systems did grab 4.9.3, to go get 4.9.4 manually because the system is no longer able to update itself.

The lead developer Dion Hulse explained, he said that one of these changes aimed to reduce the number of API calls which get made when the auto-update cron task is run. He said: "Unfortunately, due to human error, the final commit didn't have the intended

effect and instead triggers a fatal error as not all of the dependencies of" - and then there's a function, find_core_auto_update() - "are met." He said: "For whatever reason, the fatal error was not discovered before 4.9.3's release. It was a few hours after the release when discovered."

So again, I salute WordPress for making this well known, and within a couple of hours they fixed the problem. But during that little window, those systems that updated to 4.9.3 won't be able to update to 4.9.4 and anything moving forward. So I would just say to our listeners, check to make sure that you don't have 4.9.3. If you do, you should now just go to - you can go to the dashboard under Updates and then click Update Now, and your system will then jump to 4.9.4.

Leo: And as long as we're mentioning that they're a sponsor, just for those of you listening, if you are a customer of WordPress.com, who's our sponsor, they do all of that automatically. That's been done. You don't have to worry about it. This is only for who are on self-hosted WordPress.

Steve: Right, right. Good, good. I got a note from a frequent tweeter and friend of the show, Simon Zerafa, who saw somebody else's tweet and sent it to me as "Murphy's Law as applied to software design and programming." Someone named Bruce Dawson tweeted sort of the software programmer's spin: "Any mistake that is not explicitly prevented will be made. Any error that can occur, will," which of course is - everything we've learned on this podcast about the way security happens follows that. And Leo, now is the time for the College Humor video about passwords.

Leo: Now, I'm glad you mentioned this is safe for work; right?

Steve: Yes, it is safe for work.

Leo: Okay.

Steve: One hundred percent. And it's just very clever. I just love the idea that this is being done sort of as a public service.

Leo: And by the way, I want to - it's on Facebook, and I want to mention that the College Humor had to fire people because they're relying on Facebook for their traffic, and they don't get a whole lot of traffic from Facebook. Or they don't get to monetize it, I guess. So here you go, on Facebook, College Humor: How to keep your passwords safe.

[Begin video clip]

MALE VOICE: You know what they say. Mo' money, mo' passwords, to protect that money from hackers. But there's more of them, too. Mo' money, mo' potential problems. But mo' passwords, no hackers. Yeah. MC Safe Search. Online I'm shopping and I'm feeling fine. But I've got to use some passwords to keep what's mine. Passwords are codes that protect your stuff. You've got to keep them all a

secret to say safe enough.

But then how do you remember is the question that I get. I say, why not use the first name of your favorite pet? Mine's my dog, Mr. Noodles. It don't matter if you know because I was tricky and replaced some vowels with [indiscernible]. I mean, not, you know what, never mind. That was a general example, not specifically mine. Oh, man, give me two quick seconds. I mean, I'm not worried, dog, because I'm password protected.

VOICES: Keep your words, keep your passwords tight. If you keep your words a secret it'll be all right. If you want to be safe [crosstalk].

Leo: That's hysterical. If you're not seeing the video, it's great.

VOICES: Gotta go with a phrase that no one knows, don't ever write it down, don't let it show. Don't you ever let it slip in front of a hacker, got to be extra careful with your secret password.

MALE VOICE: There we go. My account's all safe and sound. What do you think you're doing creeping, peeking around? Did that video just show what I just typed in? I just changed my login, I've got to do it again? Now it's telling me I can't change again that quick. When I was typing I really hope you did not film it.

Just got an alert that my account got hacked, but they don't have my social number, so I can just relax. Why would you show this, man? What a terrible choice. I said they do not have my number. Please, listen to my voice. Oh, cool. That was another message about my account from my bank saying I just bought a speed boat, so that's cool, thanks. And a car, and a fur coat, and yet another boat? These hackers are the worst. They really get my goat. Was trying to spread the word about keeping passwords safe, but this whole song has blown up in my face. Come on.

Leo: The piano players.

VOICES: Keep your words, keep your passwords tight. If you slip up you'll probably get hacked tonight. Then someone will use your cash to buy a coat, or a boat or a car or another boat. You can't be too careful with your info, don't ever let it slip, don't let nobody know. You never know who's watching when you show your [indiscernible], stealing someone's stuff is the only way. [Indiscernible].

MALE VOICE: Hey, Rodney, where'd you get that coat? Did I just get my identity stolen by my backup singer?

MALE VOICE: My man got hacked, ha ha.

MALE VOICE: Dang, I've known you since the fourth grade, Bobby. I even shared my pizza bagels with you.

[End clip]

Leo: Aw. That's good. College Humor. I love it.

Steve: So the word is getting around.

Leo: Yeah, that's good.

Steve: How to make your passwords safe. Now, okay. This is bizarre, but I thought it was sort of fun. The tweet came in, and those of us who have been watching or have watched some of the series that we talked about a couple weeks ago, "Altered Carbon," will realize that someone was having fun mixing SpinRite with some of the terminology from "Altered Carbon."

This person tweeted: "Dear Steve, I have a question regarding SpinRite. I've come across an old stack of my great-great-great-great (possibly more) grandmother. When trying to spin her up in VR, she loads for a very long time, but eventually fails. There is also no luck when putting her in a new sleeve. What level would you recommend running SpinRite at to try and recover a failing stack? Thanks for everything. Mitch."

Leo: Love it, love it, love it.

Steve: So, yeah, I got a kick out of that.

Leo: Did you watch any more of "Altered Carbon" or just that first couple episodes?

Steve: I didn't.

Leo: So you missed - did you see the episode where she spins her grandma up, her abuela up into this tattooed Nazi skinhead?

Steve: No. And so now I understand.

Leo: Oh, that is hysterical.

Steve: Now I understand what he's talking about, an old, old, old stack of his grandmother.

Leo: Apparently ever year on Dia de los Muertos, she spins up Grandma into whatever - they call the body "sleeves" - lying around at the jail where she works. And so she spins it up. But the funny thing is this guy's got a role in the show. Later in the show he turns back into the Nazi skinhead. So it's very - it's actually quite amazing.

Steve: So you are moving through the series?

Leo: Yeah, you know, I like it.

Steve: Yeah, good.

Leo: Yeah, I enjoy it. It's very adult, we should mention, very violent, but also sexy. And I thought it was - and it's beautiful. I thought it was beautiful. Did you get your HDR TV yet?

Steve: No, I didn't do that yet.

Leo: Check it out there and do it, yeah.

Steve: Okay. I'll wait until I get amazing vivid color.

Leo: That's it. It's so much better.

Steve: So Tom Terrific asked: "If I change my local computer's DNS to 9.9.9.9, and my router has a different DNS lookup address, which has precedence?" Okay. So it's a great question. Normally the way all of our contemporary Internet-connected computers are designed, whether Windows or Mac or Linux, is they use DHCP, Dynamic Host Configuration Protocol, to obtain all of those IP-related things from the network. So at startup a query is sent out because the computer has no idea what network it's on. So it's a nonspecific broadcast saying, hey, is there a DHCP server here?

And then your DHCP server, typically your router, will speak up and say, yeah, we've got DHCP services. What do you need? And so then the client asks for IP address and DNS, and those things get sent. And also sometimes time of day and other stuff. So it's actually sort of a general purpose, DHCP can provide all kinds of information. But that's configurable. I'm sure that certainly Windows users have seen the "Obtain IP address automatically," and the same is true for DNS. You're able to obtain the DNS automatically from your DHCP server. So the point is you are able to override that on your computer to say, no, I don't want it automatic. I'm going to specify the one or two, typically two DNS servers myself.

So the answer is, if you deliberately override your computer's default setting, which is to obtain it automatically, then it will use what you tell it to, Tom. It will not use whatever the router's DNS is. It's offering it, but the computers essentially have to ask for it when they start up. And if you tell them, nope, I want to provide my own DNS, then that's what gets used.

A question from Skynet asks: "Does InSpectre" - GRC's little program for monitoring the Meltdown and Spectre vulnerabilities that we talked about a couple of weeks ago - "run in Boot Camp?" He says: "Can it see the chip? Or should I run it in WINE on the Mac side?" And the answer is, yes, Boot Camp is just an alternative boot for real Windows. I was making the distinction with a VM. Boot Camp is not a virtual machine environment.

Neither is WINE. So either booting Windows on a Mac with Boot Camp or running it under WINE will allow it to see the chip, and you'll be able to determine whether the actual microcode has been updated in your processor and what vulnerabilities it may have. But only not if you're actually in a VM.

And then we had one, two, three, four, five different people, with a little bit of overlap, suggesting various ways of combining Spectre and Meltdown. David Lemire, he was suggesting Specdrown since we have an endless stream of news about them, Specdrown. Kyle at @craigconsulting said he proposes Smectre, S-M-E-C-T-R-E, Smectre?

Leo: No, no. That's not - no, no.

Steve: Yeah, no, that didn't work. We have Joe McDaniel and Kevin both had - how about Smelter? So it's like, okay. Smelter's not bad. And then Marko of course said how about just combining Spectre and Meltdown as S&M. And it's like, ah, okay. That's not bad, S&M. I was thinking maybe SAM, S-A-M, Spectre and Meltdown, so just referring to it as SAM. But unfortunately those are just so different from each other that none of those has really grabbed us so far. But thank you for your submissions.

I was reminded of that famous quote from bank robber Willie Sutton, who was asked about his chosen career.

Leo: Willie, Willie, why do you rob banks, Willie?

Steve: Exactly. And he said: "I rob banks because that's where the money is." And what we're seeing is so many weird things happening around cryptocurrency because after it happened, and as soon as it goes from being cyber to there being exchanges, which allow you to turn this synthetic made from thin air, or actually made from a lot of power these days, turn them into actual dollars that you can buy food with and use to keep a roof over your head and so forth, suddenly it becomes a thing.

So one thing that we saw in the last week was another worrisome way of getting cryptocurrency mining into unwitting browsers. We've talked about how malicious ads are running mining software on people's machines. It turns out that over the weekend approximately 4,300 websites, many high profile, such as, get this, CUNY.edu...

Leo: Jeff Jarvis's site.

Steve: Yes. USCourts.gov. CookCountyTreasurer.gov. All visitors to those sites were mining the Monero cryptocurrency.

Leo: Uh-oh.

Steve: Uh-huh, on behalf of someone. How did this happen? Well, all of those sites support a plugin or a browser enhancement, actually it's not a plugin, it's JavaScript called BrowseAloud which is a web screen reading service. And naturally many governments have a requirement for assistive technology in order to help, for example,

visually impaired visitors so that there is a screen reading thing they have built in. So those sites had on their pages the invocation of this BrowseAloud.com script, ba.js.

Well, what happened was the BrowseAloud server was compromised such that a document.write() was inserted into the BrowseAloud JavaScript, which wrote an invocation for the Coinhive Monero cryptocurrency mining script into the pages. As a consequence, sort of through this second-level indirection, all of the sites, and there are more than 4,300 of them - in fact, Leo, in the show notes here under publicwww.com it's a search engine that allows you to find instances of scripts. And it shows, it's how I saw that CUNY.edu was on the list. It shows 4,300 sites which are known to be hosting this BrowseAloud ba.js script.

And so all of those sites were inadvertently invoking the Coinhive Monero cryptocurrency miner. And as we know, and as I've said, it's not technically a massive security problem. It pulls some processing power from your system. People have objected to it being done behind their backs. But what it really does is the fact that this could happen highlights a problem that we're sort of not paying enough attention to, and that is - and we've talked about it here before - that blindly importing the contents of third-party scripting libraries is inherently dangerous. That is, those 4,300 sites intended to only be offering the BrowseAloud service. Instead, for this period of time until the BrowseAloud folks were notified, and they got their servers cleaned up, they were the inadvertent hosts of cryptocurrency mining.

So of course our web browsers have dealt with the inherent danger representing by third-party scripting libraries by enforcing containment. And we've talked about how the browsers strongly enforce the so-called same-origin policy, which creates script isolation to limit what a script can do and what it can touch. But this happens to be sort of an interesting example of a script whose purpose is not to breach that containment, but it's quite happy to purr along running within its own container because its intent is to soak up as much of its hosting machine's spare CPU cycles as possible for annoyingly low yield cryptocurrency mining.

In other words, the thing that's sort of annoying is that JavaScript is just not a useful language for implementing cryptocurrency mining. But they're hoping to operate on the principle of, well, you know, if it's spread far and wide enough, you end up with enough aggregating hashing power to have it amount to something. And in fact, in some of the stories that we've been covering recently, we have seen - we talked about one last week where in that case they were actually using EternalBlue to infect Windows servers and mining on them. Well, that's much more high-yield mining that using JavaScript. But they were generating thousands of dollars for something. Was it a day or a week? I don't remember. I think it was a week. So it does add up.

In a somewhat different twist, another interesting bit is a website - and this is sort of what we were talking about probably going to happen in the future. Salon.com is now formally offering visitors who are using adblockers the option of mining Monero while they're on Salon's website. And Leo, I went there myself, and sure enough, because I'm using uBlock Origin, I got the blocker. I got the notice. If you just go to Salon.com it'll come right up.

Leo: So they're doing it without telling you?

Steve: Well, unfortunately, yes. You get a blocker. But if you click on "Tell me more," then while they're telling you - yup, so there you're seeing the page that I saw this

morning. If you click on "Tell me more," then they try to be mining cryptocurrency while you're reading that message.

Leo: Oh, yeah. It says "Block ads by allowing Salon to use your unused computing power."

Steve: Yeah.

Leo: Which of course means - and then they have a fairly long thing.

Steve: They do.

Leo: I don't know if they talk really very much about the Monero part of it.

Steve: Yeah, they don't go into any detail. And it's funny, too, they spin it as, you know, the advertising-based model is beginning to collapse, and we're excited about the block chain, and we want to be participating in the future and blah blah blah. So it's sort of a way of saying you can either bring your adblocker down, or we're going to run cryptocurrency mining in the background. Now, it didn't work for me. I thought, oh, cool, because I wanted to verify...

Leo: It's darkened my screen. I don't know if that's uBlock Origin doing it or...

Steve: No, that wouldn't be uBlock Origin. That's probably them saying "We're still not happy with you."

Leo: Yeah.

Steve: Because what happened was the reporting said that they start mining the moment you say "Tell me more," rather than giving them opt-in permission. So I thought, okay, I want to verify that by seeing my CPU get pinned. Unfortunately, uBlock Origin also blocks Coinhive, and they're just using Coinhive. So one of the things, I mean, and as I have said here on the podcast, I don't think this is such a bad idea. That is, I mean, users need to be informed. But, for example, I am constantly annoyed by - is it The New York Times? I think it's The New York Times. It has a very strong paywall. Sometimes I'm interested in stories that are there, but I don't want to subscribe. But on the other hand, I wouldn't mind if they pinned my CPU for the duration of time that I'm looking at a story in The New York Times. If they want to do that with permission, saying, okay, you're not a subscriber, can we use your processor while you're here, I'd say, yeah, fine.

And so, I mean, it's an interesting sort of micropayment scheme. It's annoying that JavaScript is so inefficient because it means that the economics won't really work until we come up with browser-integrated cryptocurrency mining. But I think it's a valid model. I don't want ads that are annoying. But if I could pay by using a little bit of

processing power while I'm somewhere, and if it's efficient - because JavaScript isn't. But if our web browsers had true GPU-accelerated mining built in, the economic model might work. And I think it's kind of an interesting model. Anyway, Salon is giving it a try. And so it'll be interesting to see how that works for them.

And again, sort of following the notion that the lure of actual exchangeable-for-real-currency cryptocurrency creates pressure. We've talked about how security is probably impossible to make perfect. Certainly Apple security surrounding their source code is going to be incredibly good. Yet despite that, where there was pressure from this intern's friends who kept bugging him or her to please, please, please, please, please give us the iBoot source, we promise, we promise, we promise we'll never let it go, we'll never blah blah blah, finally the person was convinced to do so. So it was pressure which created the problem.

Well, cryptocurrency creates pressure. And so we now have the first known instance of a water purification plant's network, that is, its SCADA - S-C-A-D-A is the acronym for Supervisory Control And Data Acquisition. It was the SCADA network running the centrifuges in Iran which was breached and allowed them to get messed up. So these SCADA networks are notoriously insecure. They're the kinds of things which you really don't want to have on the Internet. You need them on the Intranet because their nature is to allow you to do process control monitoring and so forth, much as you would have in a big water purification plant.

Unfortunately, it's convenient for them to be on the Internet. So of course that's where they end up being. And in this case a company, a cybersecurity company who specializes in this sort of critical infrastructure, a company named Radiflow, announced that they had found for them, and first time it's been in the news, an instance of cryptocurrency mining - again, Monero cryptocurrency - being done on the servers within the network of this company as a consequence of their SCADA network being compromised. So yes, cryptocurrency also being mined in the water purification plants.

And, finally, and this one pretty much - I wanted to wrap up with this one because it's one for the ages. Russian nuclear scientists have been arrested for logging into one of Russia's most secure supercomputers to mine bitcoin. Several scientists working at a nuclear weapons facility in Russia were arrested on a suspicion that they used one of the country's, one of Russia's most powerful supercomputers to which they had access for nuclear science research to mine bitcoins.

This apparently took place at the Federal Nuclear Center in Sarov, which is a top-secret area surrounded in Russia by barbed-wire fences where the Soviet Union's first nuclear bomb was produced during the Cold War. According to reporting by the BBC and the Interfax news agency, they wrote: "There has been an unsanctioned attempt to use computer facilities for private purposes including so-called mining. As far as we're aware, a criminal case has been launched against the believed perpetrators."

So, yes, once again, as Willie Sutton said: "I rob banks because that's where the money is." If there's a supercomputer that is not being used completely, well, why not run a little bitcoin mining operation on it and generate some bitcoins while you're at it. Except that that was unsanctioned.

Leo: Did you see that China is now shutting down its bitcoin miners? So they're all moving to Iceland, where because of geothermal and hydroelectric power, it's very cheap. So Iceland may become the new miner central.

Steve: Yeah. And in fact there was another story last week. I guess the state of Washington, some county or region in the state of Washington is being turned upside down because their power is $0.03 to $0.04 per something, megawatt or kilowatt or I don't remember what the units were. But it was, like, way inexpensive. And so major mining operations are coming in and asking for 200 megawatt connections.

The problem is that requires them to build out a much larger infrastructure, which they would like to do. I mean, they would like to satisfy demand. But if mining then collapses, and all of this business dries up, they're stuck with infrastructure that had a 30, 40-year expected amortization life, and no one who needs it and is willing to continue paying for it. So it's causing all kinds of problems. It's like, oh, well, cryptocurrency mining.

Leo: Oh, boy. Apparently a bunch of warehouses on the waterfront, the requested amount of power I think was 100, maybe it was 1,000 megawatts, exceeds the entire usage of all of Iceland.

Steve: Wow.

Leo: I think Iceland uses 340 megawatt hours, and they're asking for a thousand.

Steve: Yeah, you've got to be careful about melting ice up there, too, because that generates a lot of heat.

Leo: Wow. That's a lot of heat. That's amazing. Steve, as always, so much fun. So interesting and informative. Always a pleasure. I know we were talking before the show, you've got three more things to do on SQRL.

Steve: SQRL is - I plan to have an announcement next week. We're very close. I thought I had two things left to do, and someone said, hey, I thought you were going to deal with high DPI awareness? It was like, oh, that's right, I forgot about that because with screens getting higher resolution, my UI normally just is like the standard useful size for standard resolution monitors. But...

Leo: It would be this size on a DPI [indiscernible].

Steve: It is very small.

Leo: A postage stamp.

Steve: Yeah. But remember that from the very beginning I built SQRL to be language independent. Every single - there's no text, no English in the app at all. It all exists in an external file which - and remember there was a crowd source, no, it was not, it was crowd - I can't remember what it's called. I think if you go to sqrl.grc.com, I think that was an alias to the service that provides translation. We were going to crowd source various translations of it.

Anyway, the point is I built that in from the beginning. And because different languages have different requirements for linear length of their text, I built in a scalable UI facility from the start. So in theory I should be able, if I see that somebody's got their fonts scaled to 150%, which is the way Windows handles high DPI screens, I ought to be able to just say, oh, and drop that single number in at one point in the code, and it will scale the entire UI. I've never done that before, but it's built in. So even if it doesn't work at first, it ought to work pretty quickly.

But anyway, we're down to very few things remaining to be done. I will then spend a little bit of time on the demo website. I do need to bring up web forums because all of the people who want me to get back to work on SpinRite don't want me to be spending all of my time redundantly answering questions about SQRL. So my plan is to bring up some web forums where all of the people who have been working with me over in GRC's kind of semi-private, old-school, text-only NNTP newsgroups, they'll be able to - and are interested in helping people to understand what SQRL is, we've got Android developers, native Mac developers. It's running for iOS. Mine runs under Windows and WINE so it can run under Mac and Linux.

And so things are happening. And so of course my intention is to get it launched. And then I will turn my attention to GRC's web pages to bring its description up to speed because I haven't looked at it for years, and I need to make that current. And then back to SpinRite 6, happily.

Leo: Good.

Steve: And not a minute too soon, and happily so.

Leo: Yay.

Steve: And I'm going to come up, and we will do a SQRL announce and presentation in the TWiT studios.

Leo: Nice. I look forward to that.

Steve: Thanks to your willingness to host. That'll be great.

Leo: Yeah, that'll be great. Wow. Well, if you listen to the show, and you enjoy it, there's a couple of things I should tell you about. First of all, you can always watch it live or listen to it live or join us live in studio, which is always nice. We have our friend from Des Moines visiting. It's so nice to have - let me get your name right here. Adam, visiting from Des Moines. All you have to do is email us and let us know you're coming: tickets@twit.tv.

You can also listen and watch live on the stream. We have audio and video streams at TWiT.tv/live. If you do that, by all means, check out the chatroom: irc.twit.tv. People really have a good time during the show in there trying to parse everything

we talk about. It's kind of fun. You can also get on-demand versions. Steve has audio at GRC.com. He also has transcripts. It's the only place you can get nicely written transcripts. Steve commissions them from Elaine Farris. It's very nice. So you can search. You can use that to search. You can also use it to just read along as you listen: GRC.com.

While you're there, check out SpinRite. This is Steve's only source of income. It's his bread and butter. Make some yabba dabba doos happen in the studio during the show. Get a copy of the world's best hard drive maintenance and recovery utility. You can even work it on your sleeve, you know, get your - what do they call that token that's got your personality in it, get that all...

Steve: Bring Granny back. It's called a "stack."

Leo: Get your stack all - bring Granny back. And your new slogan...

Steve: Get your stack back.

Leo: Get your stack back.

Steve: Get your stack back, yup.

Leo: GRC.com. If you want to see video, or you have other interests, you want to see more, you can always go to our website, TWiT.tv. This show is at TWiT.tv/sn. We also have links to all of the different podcatchers, so you can subscribe and make sure you get every episode. Audio or video, you want to keep them all for future reference. Next week, Steve. Thanks for being here. We'll see you next time on Security Now!.

Steve: Thanks, buddy.

Leo: Bye-bye.