SpectreNG Revealed

Description: This week we examine the recent flaws discovered in the secure Signal messaging app for desktops, the rise in DNS router hijacking, another seriously flawed consumer router family, Microsoft Spectre patches for Win10's April 2018 feature update, the threat of voice assistant spoofing attacks, the evolving security of HTTP, still more new trouble with GPON routers, Facebook's Android app mistake, BMW's 14 security flaws, and some fun miscellany. Then we examine the news of the next generation of Spectre processor speculation flaws and what they mean for us.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-664.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-664-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Lots to talk about including, yes, the new Spectre variants. We finally have the details. Steve will talk all about that and what you can do about it and what the mitigation's going to cost you. We also will talk about a new attack on your Amazon Echo device or your Google Home device. Steve and I may disagree about the seriousness of that one. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 664, recorded Tuesday, May 22nd, 2018: SpectreNG Revealed.

It's time for Security Now!, the show where we talk about your privacy, your security, your health and happiness and well-being on the Internet with Mr. Steve Gibson of the GRC Corporation. Hi, Steve.

Steve Gibson: Yay, Leo, great to be with you again, as always.

Leo: Always a pleasure.

Steve: Yeah, we scrambled here a little bit at the end because I was apparently watching a tape-delayed version of MacBreak Weekly and didn't - I thought, wow, they're kind of running late. It's like, uh, no, [crosstalk].

Leo: [Crosstalk] half an hour ago. No, but I also figured you had some work to do on this new Spectre flaw.

Steve: Well, yeah. Two weeks ago the news leaked through Heise.de, the German magazine, about this having been discovered. And they did their due diligence to verify it. And so two weeks ago the title of the podcast was "Spectre Next Generation." Last week I mentioned that, okay, we were still waiting, we're still waiting, we're still waiting; and all the news dropped yesterday. So today is SpectreNG Revealed.

Leo: They were presumably waiting to disclose until companies had patched and things like that; yeah?

Steve: Yes. And even so, I mean, this is - what we heard was eight new problems, one of which was a big concern.

Leo: Right.

Steve: I can't square that with the news of two because all we got is two new problems. But they are problems. And every company, essentially every processor, in this case even the IBM power chips are vulnerable. I mean, this is a fundamental - I think the thing that makes this so interesting and such a concern and not like oh, just do a patch is it's a consequence of the fundamental architecture which everybody in the industry has adopted in order to get more performance, in order to squeeze every last bit of speed out of the existing technology that we have for non-quantum single instruction at a time.

We've done all kinds of things like running multiple cores at the same time, sharing caches so that they share memory, and then going ahead. Like for example it's not - if you run across a branch instruction, you take both forks of the branch. It's like, what? Well, it's because you may not yet know if you're reading ahead what the outcome of earlier instructions are that would determine whether the branch is taken or not. But if the processor has extra resources that are not being used, why not let it run ahead and then discard the results of the branch that wasn't taken, thus speculating on its own future?

And it turns out that once - it was Jann Horn at Google's Project Zero who first sort of got this glimmer of, is this really secure? And this of course then - this was last summer this happened, that this all began, when he realized, ooh, boy, because this modifies the state of the processor, it breaks trust boundaries. It breaks security boundaries between processes and between the OS and its client applications. So this is big, and it continues to give us interesting things to talk about. Which is cool, too, because it's like fundamental technology, not, oh, some router had a default password that they shipped. Of course we have some of that this week, too.

We're going to examine the recent flaws discovered in the secure Signal messaging app for desktops, the rise in DNS router hijacking, another seriously flawed consumer router family, Microsoft's Spectre patches for the earlier Spectre problems, which appeared just the day after last week's podcast, last Wednesday, for the April 2018 feature update.

A really interesting thing I know you're going to find fascinating, if you haven't already

read into it, the threat of voice assistant spoofing attacks. An interesting paper was released where some researchers did a very convincing job of getting the wrong malicious app to run for the Amazon and Google devices. Also the evolving security of HTTP as being driven by Google and the strength of their Chrome browser.

Still, believe it or not, more new trouble with those GPON routers. A mistake that Facebook made with their Android app that upset a lot of people. BMW's 14 security flaws. Oh, yes. Some fun miscellany, and then we'll take a look at what we just learned yesterday about variant 3a and 4 - we already had 1, 2, and 3, now we have 3a and 4 - and what that means. And if you really want protection, get this: an 8% hit on processor performance. So ouch.

Leo: Yeah, yeah.

Steve: So I think lots of good stuff to talk about.

Leo: All right, Steve. On we go.

Steve: Our Picture of the Week is one that I've had in our log of pictures to get to, since there was nothing particularly that jumped out at me about this week's topics. But it's something that we've all experienced often. It's a three-frame cartoon with a laptop that is shown saying, "There's a new update, but there are a few unsaved projects, and you are gone for like five minutes now. May I update and restart anyway?" And then the center frame sort of shows the laptop kind of waiting. And then finally, the third frame, "I'll take that as a yes." And of course it restarts and loses the unsaved data and so forth. So yes, the nature of automated software updates in the current PC era.

Okay. So the Signal protocol is secure. As far as everyone knows, Moxie and company have a lot of experience with security and security protocols. It's been looked at closely. We've done a podcast about exactly how it works. And as I was getting into it, I remember thinking, wow, this is really kind of overdesigned. What? But then as I've understood the set of features which they wanted the solution to have, the cleverness of what they had done was clear. And of course I shared that on our podcast about the Signal protocol at the time.

So that's different than mistakes being made in the implementation of an application which implements that protocol. So anyone who has the Signal desktop app for Windows and Linux needs to make sure that you're running the latest, which is 1.10.1; and the prerelease, which is currently in beta, of 1.11.0. Those are safe. The earlier ones, not so much.

We remember Juliano Rizzo, who was on the beach when he and his security researcher buddy discovered or came up with or realized the Beast and Crime attacks on SSL and how it was possible to cause an information leakage there. Well, they're still doing security stuff. And a couple weeks ago they discovered a severe vulnerability in the Signal messaging app for those two desktops, for Windows and Linux, which allows remote attackers to execute malicious code on the recipients' systems simply by sending them a message which requires no user interaction.

What was interesting was that this vulnerability that these guys discovered occurred purely by accident. They were chatting on Signal using their desktop apps, probably for

Linux, knowing those guys, and one of them shared a link of a vulnerable site with a cross-site scripting payload in its URL. And the cross-site scripting payload got executed on the Signal desktop of that message recipient. And of course being a security researcher, I mean, if it happened to many people they would have gone, "Oh, what? That's odd." But these guys instantly realized, wait a second, this thing just ran on my desktop.

So they dug into what was going on and tested a whole bunch of different payloads and figured out that the vulnerability resided in a function which was responsible for handling shared links which allowed attackers to inject user-defined and provided HTML and JavaScript via an iframe, an image, a video, or an audio tag. I mean, basically there were lots of ways in. And they were even able to inject a fillable form onto the recipient's chat window which could have been used in a social engineering attack to trick them into revealing sensitive information.

And it's interesting that Signal is another of those apps which is built upon the Electron application platform framework that we talked about as having some problems last week. Not clear, in fact it doesn't look like, now that we know what's going on, that Electron was to blame. It turns out that the Open Whisper Systems guys, the people who publish Signal, had already addressed this issue previously, but somehow there was a regression in their code. The April 10th release of a Signal update reintroduced a problem which they had previously added some functions to prevent. So I imagine they'll go back and figure out how this regression occurred.

However, this was responsibly disclosed. The Signal guys refixed what they had previously fixed immediately, and within hours the desktop update was made. But this was regarded as a pants-on-fire big security problem while it was there because, I mean, any bad guy simply had to send a Signal user a message, and it would have been executed. So you do want to make sure that you're updated to the latest version of the desktop app. Now, Signal does have an auto update facility, so I imagine unless you did something to deliberately deny updates, you were probably safe before this even was known publicly because this was also responsibly disclosed.

Oh, and also there was another problem - this is the second problem. There was another problem in the Signal desktop for macOS where disappearing messages could be recovered and be undisappeared. And that was also fixed at the same time. So as far as we know, the Signal desktop app is up and running and clean.

In general, I think probably, you know, there's always sort of like best practice advice. What's the number one thing you would ask someone to do or tell someone to do or make sure people do, very much like - I guess number one is probably related to passwords, where you don't want to be using the same password everywhere because of the vulnerability that creates, and you want to use a good password and so forth, not "monkey." Probably right up there, although arguably much less easy to deploy as a device, is the concern about router security.

Routers are really coming under attack. I don't know when we've had a week in this podcast when we haven't talked about one or more new or newly discovered or newly exploited or big problems with consumer routers. It seems that it's one of these problems where they're deployed en masse before a problem is found, and then it's very difficult to get the problem fixed. And it just sort of seems like it's another, you know, they're a classic case of an IoT device, but it's unique because they're very complex. They're typically running some flavor of Linux in order to offer all the features that people want in routers. And it's not even features people want, it's that manufacturers have a competitive checklist. So there's a ridiculous amount of power in these things that most

people never even know about or ask for, yet it's there. And of course we know that complexity is the enemy of security.

So the other thing that makes these unique is that, unlike most IoT devices, which are safely hiding behind a router, its role is to be there on the frontline. So it's the device more than anything else in your entire network where security is crucial because it's got a public-facing surface or attack surface that is potentially vulnerable. So what we're finding is that DNS hijacking is, if the router's not being used, as we've been discussing in recent weeks, for hosting botnets or for using Universal Plug and Play to bounce traffic off of its public surface in order to disguise the source of attacks, one of the other ways that routers are being abused that is being seen now in the wild is DNS hijacking.

This is a problem because most devices behind the router simply use DHCP, Dynamic Host Configuration Protocol, to configure the devices, whether it's your Android, your iOS, your light bulbs, or all your computers. It's the "obtain IP address automatically." Well, you're not only obtaining the IP address for the main machine, you're also through DHCP, unless you deliberately configure it otherwise, you're obtaining the DNS address for that machine, that is, the two IPs which all of the systems on your network will use to look up domain names' IP addresses.

Well, what this means is, if that's a malicious server, that is, if something has hijacked your router's DNS so that the IP address your computer has is to a malicious DNS server, then one of the crucial assumptions of the Internet for security, which is you are really going to the server you think you are going to, when everything else is correct, you know, you look at the URL in your browser, www.amazon.com, no misspellings, no umlauts, no unicode snuck in there, it absolutely is www.amazon.com. You're sure of it. Well, you need DNS to send you to the proper IP for Amazon.com.

Now, the problem that hijackers have is unless they're able to somehow get a security certificate for Amazon.com, which is increasingly difficult thanks to the CA system really working hard to improve its security, then it's probably unlikely that they can get you to an HTTPS connection. But it's often the case that users aren't paying attention. And so you might well be at http://www.amazon.com and think everything is fine, when you're in fact at a spoofed site. And it then pops up a page saying, oh, your login has timed out. It's been too long since you've been here. Please log in again. And so that's a way the bad guys have of obviously acquiring your username and password at a site which has been spoofed.

All of this to say that in the news this week is yet again widespread DNS hijacking malware which has recently been found to be targeting Android devices and has also recently updated its capabilities to target iOS and desktop users. It's called the "Roaming Mantis" malware and has been hijacking Internet routers in the last month to distribute various kinds of malware. Kaspersky Labs have been tracking this and have been posting the news that this thing is spreading strongly. There's a router that it's been compromising from DrayTek, which is a Taiwanese manufacturer. They make router switches, firewalls, and VPN devices. And there is a zero-day vulnerability in those particular routers which is being used to change its DNS settings.

So it's not clear where the geographic spread of these routers is, if they're only being sold to specific ISPs to provide them to their customers, or what the retail path is. But if you happen to know that you have a DrayTek router on the shelf, you want to absolutely make sure that you've got it updated. DrayTek is aware of the problem. They were informed by Kaspersky, who found this particular DNS hijacking malware roaming around the Internet. They've got a really long list of router model numbers of their manufacturer which are vulnerable. So if you happen to have a DrayTek router, it's the latest victim of

DNS hijacking.

But even if you don't, it's, I would say, probably worth just maybe logging into your router and just making sure that nothing has configured your router's settings. Or just take a look at your own DHCP settings for your machine and make sure that it's what you expect them to be. In some cases that may not be enough because your system might be using the router's gateway as its DNS, that's often the case, so that your queries go to the router, which it then forwards to public DNS servers. So it's probably necessary to check the router. As I said, it's not simple consumer-oriented advice to follow. But unfortunately the danger is real. This is something which is becoming increasingly widespread as routers are falling victim to one exploit after another.

I also wanted to mention last podcast we talked about the firmware updates, that is, the latest Intel microcode firmware updates for all versions of Windows 10 up to but not including, which was sort of a head-scratcher that we talked about at the time, for this latest April 2018 feature update, which is v1803 to Windows 10. The day after that, last Wednesday, Microsoft added the microcode updates for 1803 and 1803 Windows Server. So those are available. I've got the link in the show notes. It's KB4100347. So if you are running 1803, and you haven't updated to this, KB4100347 will give you the latest, I mean, just across all of the chips that Intel is going to update for that latest version of Windows 10.

I did see that this was going to be pushed out both through the "go get it yourself" and the regular Windows Update mechanism. I didn't look, I could have - I shoulda woulda coulda - checked to see whether my Windows 10, which I did verify this morning that there are no updates for it available, whether it had already grabbed that and brought itself current. But in any event you may want to.

Okay, now, Leo, this is what I think is going to catch your fancy. And perhaps it was inevitable. As we know, we were just talking about making sure that Amazon.com was actually A-M-A-Z-O-N and not A-M-A-Z-0-N dotcom because for years lookalike phishing attacks have used lookalike domain names. Well, now we have voice assistants, and we have phishing of sound-alike trigger phrases, which when you think about it shouldn't surprise us at all. In a just-published research paper titled "Understanding and Mitigating the Security Risks of Voice-Controlled Third-Party Skills on Amazon" - and you know the name of that device, the Echo - "and Google Home," a group of Chinese and U.S. researchers - in that paper they describe their various methods of attacking smart assistants, those made by - and I'm working hard not to say the "A" word.

Leo: Say Echo. That's okay.

Steve: Oh, Echo, yes. But everywhere in the coverage they're talking about the Amazon A-word. So we've already talked about previously the problem of applications maliciously and deliberately leaving the microphone on, staying active in the background, and then streaming whatever the microphone receives off onto the Internet. These guys do describe that. But they also describe, and they convincingly demonstrate, a more worrisome - and I don't know what Amazon and Google are going to do about this - attack which they call "voice squatting."

The idea is to trick the user into opening a malicious application by using voice triggers which are acoustically similar to the ones of authentic apps, and then of course using the malicious apps to phish users for sensitive data or eavesdrop on their surroundings or do whatever. So, for example, an attacker might register an app that triggers on the phrase

"Open Capital Won," but spelled W-O-N, that is, as in winning and losing, won. Which obviously sounds similar to "Open Capital One," O-N-E, which is the real intended trigger for that phrase. How does the voice assistant know?

So they demonstrate in their paper, for example, there is an Amazon Echo skill named "rat game," and they registered what they call an attack skill, "rap [R-A-P] game." And the very fact that I had to spell it demonstrates the problem of the skill. It sounds almost identical. And sure enough, in their testing, having registered this "rap game," after it being registered, when someone wants to open "rat game," "rap game" with a "p," the attack skill wins preferentially. They did that over on the Amazon side. For Google Assistant they registered the attack skill "intraMatic opener," which sounds very similar to an existing target skill, "Entrematic Opener." And again, when users tried to invoke "Entrematic Opener," they got "intraMatic opener" instead.

And then in one additional and I thought very clever attack they experimented with making a more specific request. For example, they registered "rat [R-A-T] game please" so that, if the user says to the Amazon Echo device, "A" word, "open rat game please," well, that's a more specific match than just "rat game." And again, if the extra "please" is added on the end, it matches more strongly than without it, and again the attack skill is executed.

So these guys put together the paper, demonstrated their proof of concept. They've notified both Amazon and Google, who have both responded and said, "Thank you for your report. We're looking into it." But, boy, Leo, I don't know how you solve this problem. I mean, imagine...

Leo: Yeah, but so what? So how would you use this maliciously, besides you got the wrong game?

Steve: Well, or the wrong anything.

Leo: It then says tell me your bank password, and you're going to tell it your bank password? No.

Steve: You don't think so?

Leo: No. You never do that on an Echo. No one would fall for that. I mean, I don't think so. I think this is more an equivalent of getting somebody's computer to do something silly than anything seriously dangerous. I mean, I'm trying to think of an attack surface. With a banking app you can't transfer money to somebody. I mean, most of these apps are really very limited. So you could, I mean, it's more like a kid could get something funny happening to you.

Steve: Okay. So I guess...

Leo: I'm trying to think of how this would be maliciously abused.

Steve: If nothing else, it causes the device to misbehave, to do something that you don't want it to.

Leo: Right, right. That's annoying. But that happens all the time, by the way, anytime you're using an Echo. I mean, it's not unusual for - I mean, you use Siri sometimes; right? Doesn't she get you wrong about three quarters of the...

Steve: I don't use any voice things.

Leo: Well, try, I mean, Siri gets you wrong half the time. It's annoying. I mean, you'd have to be - I guess you could make, okay, I'm playing rap game now, and it's asking for my bank password, I don't know why, but I'm going to give it to you. I mean, that's a pretty long stretch.

Steve: Yeah. I guess the problem is that we're using a very soft match technology...

Leo: Oh, yeah. It's not good. I agree.

Steve: ...to cause things to happen. And so it's a fundamental problem with voice.

Leo: It's similar to the problem, though, of a web page, as you said, Amazom instead of Amazon. But I think it has less long-term utility in terms of maliciousness. If I get you to Amazom.com, I could ask for your Amazon password reasonably. But I don't know what you're going to do with a voice app to screw somebody up. You've entered the wrong address, I mean, we all do it. And you get some strange page; right?

Steve: Yeah, yeah.

Leo: It's equivalent to that.

Steve: Well, in this case, though, I mean, both Amazon and Google want their systems to work as well as they can. And this is a problem which will be - it's hard to see how they mitigate this. I mean, like...

Leo: Well, that's right. I mean, you'd have to have - I think what you do is you block duplicate accounts based on phonemes, not on spelling, something like that.

Steve: Exactly.

Leo: That would eliminate Capital W-O-N.

Steve: Exactly, yes. So you start to look for deliberate attacks for sound-alike registration phrases and disallow those. Which clearly they're not doing at this point.

Leo: No.

Steve: I just think it's, at this point, we're in the early stages. But I immediately think of Stacey, who's got like her entire world is controlled by, you know, her whole house and everything. And if you're running a mischievous voice-controlled app that is there in place of something legitimate, it could probably get up to some mischief with you. But again, it's not going to probably steal your bank account stuff, you're right.

Leo: No, it's a vandalism kind of attack. Oh, we're going to turn the thermostat up to 80 <evil chuckle>.

Steve: Well, and again, though, it is an attack on the integrity of a system that more and more people are using and wanting to use. So I wouldn't be surprised if someone clever were to come up with something that'd cause some trouble. So we'll see.

Leo: It's conceivable. I mean, you could have some sort of sales app that would - I don't know. I mean, I'm really working to come up with something here.

Steve: So Chrome is Google, and their Chrome browser is continuing to move their intent to secure the web through using the power of the presence. Like they're now the majority web browser, so they're able to force change, as we have seen with them, for example, forcing the early wind-down of SHA-1 certificates prior to its already sort of more slowly planned withdrawal because their browser was going to stop honoring them earlier than the rest of the industry. Right now, when we go to a secure TLS-encrypted site using HTTPS, they show us the little lock icon and then a big comforting green "Secure" to say, yes, this is a secure connection.

So with Chrome 68, which is slated for release in a month and a half, in July, they will start showing "Not Secure" for HTTP. So right now they're silent on HTTP. They give you the green happy "Secure" for HTTPS. In 68 in July they'll start saying "Not Secure." Then for 69, Chrome 69 in September, they're going to remove the "Secure," the happy green "Secure," probably because they'd like to just have more room for the URL. It takes up a chunk of space up there in that space. So they'll only show the lock.

Google has stated that, since most traffic is now HTTPS anyway, they feel that it's no longer necessary to draw the user's attention to the "Secure" indicator, that being just sort of the de facto. And so instead Chrome will start focusing on highlighting situations when the user is accessing an insecure website. So again, in September - well, in July 68 will start saying "Not Secure" if you're on HTTP. And then in the next one after, in Chrome 70, which I don't have the date for that, probably either the very end of 2018 or maybe the beginning of 2019, they're going to further amplify that "Not Secure" indication by if you start entering data into a form on a non-HTTP site.

And what would really be interesting is to know whether it's - because we've talked about this at length - whether it's the submission URL which is not HTTPS or whether the page where you're filling the data in is not HTTPS. Be interesting to see which of those they

choose, or maybe both. Maybe they just want both to be an HTTPS page and an HTTPS submission. Anyway, they're going to animate that "Not Secure" by turning it red to, like, really call the users' attention to the face that they're entering data into a site which is not HTTPS. So again, they're determined to move the industry away from HTTP over the grumblings of people who argue that their site just has no need for it.

On the other hand, I do support the idea that a site that is wanting data from a user could be benign. It might be that it really doesn't need security. But chances are what a user is putting into a form they would like to not have eavesdropped on. So highlighting the fact that where they're putting the data in isn't secure is probably a good thing.

GPON routers are still and again in the news: 17,000 of these GPON routers that we've been talking about the last few weeks have been infected with the Satori botnet. They're now scanning port 3333, which is used by the Ethereum miner known as Claymore. It turns out that there's a zero-day flaw in the Claymore mining software which exposes it when it's publicly exposed on port 3333. This new botnet, this Satori botnet has added its own skill, looking for those Ethereum mining rigs. And, if found, they reconfigure the Ethereum miner to join the DwarfPool mining pool and use the attacker's Ethereum wallet. So yet another example of where routers are being put to a bad purpose.

And just to finish up on GPON routers, we already knew of two zero-day exploits which we talked about a couple weeks ago that the researchers, the Chinese researchers at 360 Netlab had discovered. Well, they found a botnet running a third, previously unknown flaw. So we also talked about how there were five different botnets fighting for dominion over those GPON routers: Hajime, Mettle, Mirai, Muhstik, and Satori. To that we now add a sixth one called TheMoon. And it's currently winning because it's exploiting a new, previously unknown zero-day for which there is no patch, allowing it to get into and take over these GPON routers.

And of course remember that, since GPON stands for Gigabit Passive Optical Network, this suggests that the routers in question are well connected to the Internet, which makes them extremely valuable as DDoS attack platforms, as scanners for other vulnerabilities, and maybe, if they're able to run gigabit fiber, they have beefier processors, so they could even be used usefully for some bitcoin mining or cryptocurrency mining.

So, wow. I mean, it's really looking like, as I said at the top, consumer routers are the target for attackers. And we had previously been talking about how the good news was they were just being used for attack reflection or purposes that didn't cause a subversion of the user's network. But if they start saying, well, while we're here, let's point DNS to our own malicious DNS server in order to perpetrate those effects, then that becomes certainly a lot less than benign.

Leo: Is GPON a brand name or a description of a type of router?

Steve: It's not clear to me. The company is Dasan, D-A-S-A-N, a South Korean manufacturer of these GPON home routers.

Leo: It's a technology; right? So there might be multiple routers using GPON?

Steve: Well, we know there are multiple ISPs using them.

Leo: Ah.

Steve: And so GPON stands for Gigabit Passive Optical Network. But it might be an acronym that they use on their own routers like as a brand name of what they also are. So sort of both.

Leo: All right, Steve. Back to you.

Steve: Yeah, thank you. So this was sort of a tempest in a teapot. There was a mistake made in the Facebook app for Android that probably could not have come at a worse time, given all of Facebook's recent privacy controversies. It turned out it was the result of an innocent coding error in a beta release of the Facebook app for Android. But what it did was to cause a permission pop-up, asking the user to give the app superuser root-level access. And of course a lot of people said, "What?" And there was a lot of Twitter traffic.

Security researchers looked into what was going on and determined that the package that appeared to be triggering the superuser popup was something known as "WhiteOps," which is an SDK which Facebook was in the process of deploying, used for detecting fraud and implementing domain white and black lists. So they didn't intend to have anything get superuser root-level access. A Facebook spokesperson confirmed that the popup dialogue was caused by a coding error.

Facebook said: "A coding error in one of our anti-fraud systems caused a number of people running the Facebook app and certain permission management apps on rooted Android phones to see a request for additional access permissions." They said: "We do not need or want these permissions. We have already fixed the issue. We apologize for any confusion." So it generated a bit of a firestorm on the Internet with people saying, wait a minute, what's going on here? And it was a mistake, and those things can happen.

We've talked in the past about the Tencent team at Keen Labs. They have a great, well-earned reputation for hacking autos. And they spent, it turns out, pretty much all, actually a little more than 12 months, all of 2017 starting from January 2017 through the first month or two of 2018, looking closely at the security of BMW automobiles. And the short version is, as a result, BMW is now working on firmware updates to fix 14 flaws which were found to affect the I series, the X series, and the 3, 5, and 7 series cars, dating back to 2012. And that's an estimate based on the components that they found to be insecure, and when those components first appeared in use in the BMWs.

The good news is the exploitation of the flaws is difficult. It can be done remotely, but it apparently requires access to a GSM cell tower, which we know that there are spoofed GSM cell towers, so maybe you could do it creating a spoofed connection, a cellular connection to the car. And it can also be done with a USB connection. But both the researchers and BMW agreed that the dangers are high end, that is to say, requires a high level of skill in order to perpetrate the attack, and do not represent imminent danger to the car's owners. Yet they do need to be fixed.

While you were on vacation most recently, Leo, Father Robert and I talked about the topic of the CAN bus in cars and the fact that there are high-speed, high-performance CAN buses and low-speed CAN buses. And in fact there's not just one CAN bus. There are now in contemporary vehicles multiple CAN buses with firewalls that are deliberately

there to decide what can cross from one CAN bus to the other. So it's often the case that the point of entry is through the so-called "infotainment" systems in the cars; but while, yes, you may be able to change the radio or play with the dashboard, you're not able to affect steering or braking or driver-critical vehicle safety-related systems because those are deliberately on an isolated bus. It wasn't clear because in the paper they talked about being able to penetrate the car's CAN bus security. But the reason it's not clear is they're giving BMW all of this year to get this fixed.

Leo: Yeah, we talked about it before, too.

Steve: Yes, yes.

Leo: I mean, part of the problem is sometimes car manufacturers only have one CAN bus network, and then they depend on some sort of barrier that is not always perfect to prevent the telematics systems from getting to the drive systems. I mean, that's the real attack; right? The attack is hard enough to get on the bus; but once you get on the bus, then to get through the security that keeps you from accessing the brakes and stuff.

Steve: Yes, unless you want to...

Leo: We've talked to guys who do this. I mean, you saw the "60 Minutes" story. I mean, people can do that, yeah.

Steve: Yeah, yeah. I mean, and it is absolutely the case that you don't want something malicious in your car, regardless. Because, I mean, even spoofing things like how much gas you have left could be an annoyance if it led you to believe that your tank was full and you suddenly ran out of gas. It's like, wait a minute; you know? Because that kind of stuff, the dashboard stuff, we have seen multiple compromises of that. So it looks like all of the passenger UI is on one side, often; and then, like, engine and collision and braking and acceleration and even steering, there have been instances of compromises there, but I think more for demo purposes.

Leo: Well, Charlie Miller, remember he had that hack where he wirelessly could get into a car and steer it into a ditch.

Steve: Yes.

Leo: That was pretty scary.

Steve: Yeah. So it's certainly the case that car manufacturers are taking the security of these systems very seriously. And I saw that they're not going to disclose details. They have a nice paper talking about, with charts - I have the link to their PDF in the show notes - showing a chart of things found, like 14 different flaws and where they are and so forth, but no explicit discussion of them because they want to give BMW, first of all, time

to fix them. But then it also requires that the car, maybe it could be an over-the-air fix, but maybe take the car in to have it serviced; and then, while the car is in the shop having its oil changed, it gets itself updated. It's not clear what the path would be to get it fixed.

But they are giving BMW plenty of time, which I think is due. And especially given the fact that none of this appears to be - it requires a high-end attack. These guys had to work to make it happen. And even then it's not clear that it's more than just a nuisance. But as we know, many attacks start as a nuisance and can escalate from there.

I have some fun miscellany. We've talked about often Star Trek. Of course ShieldsUP! is the name of my rather famous port scanner at GRC. I just wanted to mention that "Discovery" on CBS, even though it was all-access, and it upset a lot of people, including myself, it's been renewed for a second season; and that there are a whole collection of additional Trek movies in development. So Star Trek continues to live.

Also, many of our listeners were huge fans, as I was, of "The Expanse" series on Syfy. Really, the books were great. I read them deliberately before the series began. I have not yet caught up and watched the second and third seasons. We'd just finished with Season 3, and Syfy announced that they were going to cancel the series. However, Jeff Bezos is a huge fan, and he's now in talks to pick up the production and plans to produce, given that everything proceeds as expected, to produce a fourth season. So for fans of "The Expanse" who have watched all three - I'll be catching up here at some point because the first season was excellent - there will be a fourth.

Leo: Yay.

Steve: And Leo, you'll get a kick out of this. You were right to be skeptical about the EM-Drive which we talked about. I couldn't sell you on the idea that maybe there was a reactionless space drive that was somehow possible, that apparently we were told NASA researchers had built a chamber and had somehow this reactionless drive that used microwaves in a specially formed cavity to resonate, and it was apparently generating some thrust. And at the time you were like, uh-huh, yeah, I'm not buying it.

Leo: Well, it just violates Newtonian physics. But, you know, not that that couldn't happen. It's a high standard, high bar to cross.

Steve: So it turns out that Germans got involved.

Leo: Oh, they know what they're doing, now. You've got to trust the Germans, yeah.

Steve: They know what they're doing. They were not - they left no stone unturned. They used a super high vacuum so that heating of the air in the chamber could be ruled out. They used mu-metal in order to mute and absorb magnetic lines of force. And they really, really - they said, okay. We need to determine for sure whether these fringe drive space drives work or don't. So they built up a system. They implemented their own version, very carefully, German-ly crafted, of an EM-Drive, and it produced thrust. Then they said, what? And they looked more closely.

It turns out that the magnetic shielding was incomplete just for convenience sake. Turned out it was very difficult to shield it from the Earth's magnetic field completely, and that the DC power delivery to the EM-Drive was interacting with the Earth's magnetic field and electromagnetically pushing this thing. So that kind of rules it out because, by definition, we cannot take the Earth with us into space to get acceleration since leaving the Earth behind is the whole point. And a space drive which reduces to a simple magnet and coil is not going to get us anywhere, literally.

Leo: Oh, too bad.

Steve: Yeah, too bad.

Leo: And it doesn't have enough impulse to use it as a subsonic transporter around the world. Or does it?

Steve: No, no. And even in space, Leo...

Leo: You'd be going, like, two miles an hour.

Steve: Yes. And even in space it would require a great deal of patience to, like, okay, we're accelerating. Wait, are we going any faster than we were last year?

Leo: Little bit, little bit.

Steve: A little bit.

Leo: Little bit.

Steve: Not much. Not enough to make a difference.

Leo: Oh, well.

Steve: I found a nice note from a Jeff Karpinski in Elizabeth, Colorado. His subject was "SpinRite does you know what." On May 18th he sent it to me. He said: "Hey, Steve. Happy SpinRite owner for over a decade. Used it many times on conventional drives, but today was a first for me and SSD." He said: "My shop PC, which runs CNC for my laser cutter and mill, became sluggish all of a sudden." Oops. And we know what that can often mean. He said: "A peek at the Windows system logs showed an explosion of drive write failure warnings. Promptly rebooted into SpinRite, and I did a Level 1 for a quick assessment." He said: "It's only a 250GB drive, so the scan took just a few minutes. Sure enough, some issues in the early sectors." He says: "Reran SpinRite at Level 2, and it took an hour or so to beat through the problem spots. After SpinRite finished 'blessing the bits,'" as he put it, in quotes, "everything was back to normal. PC performance was

back, and Windows logs were quiet."

He says: "Now my question. If SSD writes are truly randomized across the media as marketing claims, why were the bad sectors on my drive largely contiguous and localized at the beginning of the drive?" He says: "Also, the SMART window showed no issues being reported whatsoever. Not very 'smart' in my book." He says: "I'm not trusting this drive anymore, so I'll pick up a second drive this weekend and let Windows mirror it. Thanks as always for a fabulous product and keep up the great work with Security Now!." He says: "Now I'm retired, it's the only podcast that I listen to."

So Jeff, thank you. And to answer your question, the SSD controller performs the mapping from logical sectors on the drive to physical sectors spread out across the actual physical surface. So the sectors at the beginning are where the OS typically resides, and they're the ones that get the most use. But they're logically positioned at the beginning of the drive, not physically positioned at the beginning of the silicon. So they are in fact spread out. But normally they're not being recycled unless you're writing to them; and probably there was a lot of reading being done, but not a lot of writing.

So what happens, remember that SSDs are just little tiny capacitors. All they have is they have charge stranded on a floating gate which the technology is able to sense that charge. Over time, the charge can and does leak off. And so this is one reason why running SpinRite on an SSD is so useful is, whereas hard drives' magnetism tends to be a fundamentally permanent facet of a magnetized surface, the SSD has just got some electrons kind of suspended in limbo, and you're hoping they all stay there. But there's a tendency for them to wander off. So what happened was, when you ran SpinRite on Level 1, which is only a read pass, it immediately detected, whoa, we've got some problems here reading, just passively reading these sectors.

And so what it then did when you ran it on Level 2 was, when it found problems reading, it then did recovery and rewrites which strengthened, essentially recharged those little capacitors on the SSD. And maybe then the controller relocated physically, again, used its drive leveling which is trying to level where the writes occur. It may then, when SpinRite rewrote those troubled sectors, it would have physically moved them. But when you're only reading, that's not considered a destructive process. But it is unfortunately one where after time you can start having significant problems. And as we know, something slowing down is a good early sign of reading requiring more time, which is a clue that it's probably a good time to run SpinRite.

And as for SMART not being very smart, it is the case that not all drives choose to publish what they're doing publicly. This is one of the biggest annoyances of hard drives is that there's no way to force a manufacturer's behavior. Back in the day, Compaq was able to, and did, they had so much marketing strength that they were able to force hard drive manufacturers to publish what was going on behind the scenes. Otherwise, Compaq would not have purchased their drives. So they forced this to happen. And over time people are somewhat belatedly still doing it, or in this case just saying, "Trust us, it's an SSD." It's like, eh, no.

Leo: So you shouldn't use an SSD for backup because it'll eventually just leak out all the information.

Steve: Correct. Long-term, you know what is still the most reliable is magneto-optical storage or just standard optical storage. The archivists are still using really good optical storage. But you're right, Leo.

Leo: Like a CD or a DVD kind of?

Steve: Yes, yes, yeah.

Leo: Okay.

Steve: Because that's a state change which is reliable. There can be an oxidation problem, so gold is good to use as a substrate. And then of course keep them in a dark place. But yeah, an SSD, if you did do it, you'd want to put it - oh, and by the way, I heard a comment the other day on one of your podcasts about whether Steve still had his Treos in the refrigerator.

Leo: Yes. And do you?

Steve: Yes, I do.

Leo: Why? Now, Steve...

Steve: My tungsten - well, because nothing made me - I don't need the room.

Leo: Well, that's what I'm wondering. Did you buy extra refrigerator space for this?

Steve: No, no.

Leo: And do you keep it in the freezer or the fridge?

Steve: I have no food in my fridge, so there's lots of room. I have lots of room for unused PDAs.

Leo: So you open Steve's fridge, there's no mustard, there's no ketchup, there's...

Steve: There's milk for lattes.

Leo: There's milk, and there's hard drives.

Steve: And batteries.

Leo: And gadgets and batteries. That is hysterical. And not surprising at all.

Steve: Chemical things that you want to keep cool, yeah. So, okay. As I mentioned at the top of the podcast, it was pretty clear when we first learned at the top of this year, the beginning of the year, that speculative execution and also caching, because that's what Meltdown was about, all which were in place for performance, were in trouble. The problem was that they fundamentally used resources that were shared among processes. And in order to function, a cache is inherently a history of what's been done using the fact that what is about to happen is often related to what has just happened in computer code. So if you remember what you've done, maybe you have it on hand if you need to do it again in the short term.

So any kind of a history means that what has happened recently alters the speed of what happens next. And if you are very careful at noticing your own execution speed, it tells you about the past. Which, incredibly enough, attackers can use in order to leak information from areas of a system they should not have access to.

Okay. So we initially had three variations of this problem. Collectively they were known as Meltdown and Spectre. We had Variant 1 and 2 were Spectre, and Variant 3 was Meltdown. One and three were relatively easily solved with some software changes. That is, like OS-level changes. That second one, that Spectre Variant 2, that was the pesky one. That was the branch prediction speculation problem which required turning branch prediction off. And that's why it had a big performance impact if you chose to do it because being able to predict which way a branch would go in the ability of the processor to fetch instructions ahead allowed it to keep the memory bandwidth channel saturated.

Memory is still the slowest thing. DRAM is slow, which is why we have typically three levels of caching in order to just try to feed the hungry multicore chips we have with DRAM, which stubbornly refuses to go any faster. We've talked about promising and hopeful future technologies. There are some. HP's working on that cross-point memory, which would potentially give us DRAM density at static RAM, that is, at cache RAM speed, which would completely change everything. I mean, it just would revolutionize. But it hasn't happened yet. And they seem to be stalled because they were supposed to already have test devices out by now. So who knows what's happening there. But the problem with turning off one of these features is that it hurts us. That is, it's there because it helps our performance so much. So if you make it not there anymore for the sake of security, you get a performance hit.

So now we know what happened two weeks ago, that is, the news hit that there were some more problems found. And it shouldn't surprise any of us that that was the case because the things we've done in our chips, and by "we" I mean everybody - AMD, ARM, IBM with several of their processors, and of course Intel. Everybody has done this because it's the way you squeeze more performance out of your architecture. So consequently now everybody is in trouble and is watching all of these things very carefully.

What we knew two weeks ago is that new problems had been found. We heard, you know, Heise.de reported eight new problems. I don't know where the other six are, but we now have good information, complete information about two new problems which are being called 3a and 4. Of the two, Variant 4, which is a Spectre problem, is the big problem. What's interesting is that both Google and Microsoft security researchers independently discovered these problems just because everybody is now looking deeply into other possible ways that our contemporary speedup architectures can be leveraged.

And remember that for decades we have all been sort of merrily going along with processing architectures that could be exploited in this fashion. Yet it wasn't until last summer that - I'm blanking on his name. I have it here. The guy at Google's Project Zero.

Leo: Travis Kalanick?

Steve: No, not Travis in this case. Oh, Jann, J-A-N-N, Horn at Google. He was the original discoverer of the Meltdown and Spectre flaws. And he was the guy at Google, along with people at Microsoft, who figured out that there was more trouble that still had to be resolved. So Variant 3a is known as the "Rogue System Register Read," which is a variation of the Meltdown flaw. Variant 4 is "Speculative Store Bypass." And that's the one that's most worrisome and the one that requires yet another microcode update. So we've got more microcode updates coming.

Red Hat is involved because of course they want to secure Linux against this. So they explained that Variant 4 relies upon the presence of a precisely defined instruction sequence in the privileged code. Now, of course, that's not hard to achieve because everyone can disassemble other people's code. So open source makes it super easy. But even closed source, the idea is that, as we know, contemporary security technology does not rely on unknown code, that is, unknown algorithms. It relies on unknown keys. So there's no bar being set by requiring precisely defined instruction sequences to be known or unknown. Everybody has that.

And they continue, saying: "...as well as the fact that memory read from an address to which a recent memory write has occurred may see the previous value and subsequently cause an update to the microprocessor's data cache, even for speculatively executed instructions that never actually commit." That is to say, remember that, if the processor executes speculatively, assuming that a certain branch will be taken, then it can cause some change, even though that it ends up throwing away all of that work it did if a different branch got taken.

Okay. So Microsoft just yesterday, I mean, this has just all come to light and happened. Yesterday Microsoft said: "In January 2018, Microsoft published an advisory and security updates for a new class of hardware vulnerabilities involving speculative execution side channels known as Spectre and Meltdown. In this blog post [yesterday], we will provide a technical analysis of an additional subclass of speculative execution side channel vulnerabilities known as Speculative Store Bypass (SSB) which has been assigned [and they gave it] CVE-2018-3639. SSB was independently discovered by Ken Johnson of the Microsoft Security Response Center (MSRC) and Jann Horn of Google Project Zero."

Microsoft says: "What is affected?" and answers, "AMD, ARM, and Intel CPUs." And it turns out in other reporting IBM is there, too, with their chips. "What is the risk?" Microsoft says. "Microsoft currently assesses the risk posed by [this CVE, this SSB] to our customers as low. We are not aware of any exploitable instances of this vulnerability class in our software at this time; but we are continuing to investigate, and we encourage researchers to find and report any exploitable instances as part of our Speculative Execution Side Channel Bounty program." Remember, that's the one, what was it, Leo? I think it was a quarter million dollars?

Leo: Yeah, it was a big one, yeah.

Steve: Yeah. It times out at the end of this year. So you've got through 2018. But they're offering a lot of money to anybody who can actually find an exploit. They say: "We will adapt our mitigation strategy for [this SSB] as our understanding of the risk evolves." So they say: "What is the mitigation? Microsoft has already released mitigations as part of our response to Spectre and Meltdown that are applicable to SSB in certain scenarios, such as reducing timer precision in Edge and IE." So because this is, as we've talked about before, it is subtle variations in the speed at which instructions execute, which is the root of this information leakage, if you just don't let the software know what time it is with sufficient resolution, you just shut down that exploit.

Leo: That's funny. That's such a funny fix.

Steve: Isn't it? Yeah. And so this is a different thing, a very different thing. But it's still ultimately, because the way the information leaks is in timing of individual instructions. So they said: "Software developers can address individual instances of [this exploit] if they are discovered by introducing a speculation barrier instruction as described in [blah blah blah]." And that's that LFENCE that we've talked about. There's a longstanding instruction where if the processor hits it, it forces everything to complete before it goes on. And so what can be done and what has been done is this instruction can be salted through areas which are known to be sensitive in order to specifically block information leakage from that little bit of code.

So the beauty of that, I mean, that's like the most labor-intensive, most error-prone solution because it can be hard to find every single instance where this might be necessary. But the beauty of it is that it's like virtually undetectable performance hit because it's only in particular places that you're saying, okay, everything needs to catch up before we move forward. So that LFENCE instruction will slow down that tiny bit of code, but let everything else run at full speed.

The hurtful mitigation is the next one. So I'll just continue. They said: "Microsoft is working with CPU manufacturers to assess the availability and readiness of new hardware features that can be used to resolve [this]. In some cases, these features will require a microcode or firmware update to be installed." And I should say we're beyond "may." Intel has already announced they're working on it. "Microsoft plans to provide a mitigation that leverages the new hardware features in a future Windows update." So in other words, when Intel has yet again new firmware across the family of chips, Microsoft will support that in Windows. Probably not old Windows, but newer Windows.

So then under "Preventing speculation techniques involving SSB," they said: "As we've noted in the past, one of the best ways to mitigate a vulnerability is by addressing the issue as close to the root cause as possible. In the case of SSB, there are a few techniques that can be used to prevent speculation techniques that rely on SSB as the speculation primitive." And then they talk about what I just was, this notion of the LFENCE to force a catch-up. Then they say, as another mitigation: "Speculative store bypass disable," so that's SSBD.

They say: "In some cases, CPUs can provide facilities" - and Intel doesn't have it yet, but it's coming, thus the microcode update - "for inhibiting a speculative store bypass from occurring and can therefore offer a categorical mitigation for SSB." They said: "AMD, ARM, and Intel have documented new hardware features that can be used by software to accomplish this. Microsoft is working with AMD, ARM, and Intel to assess the availability and readiness of these features. In some cases, these features will require a microcode or firmware update to be installed," and blah blah blah.

And under "Generally applicable mitigations" they talk about, as we were saying, the things that they've already done that can work. However, Intel acknowledges that it can be up to an 8% decrease in system performance if you do this global mitigation; that is, if with updated firmware, which offers a new bit in a feature register, that is, the SSBD bit, the Speculative Store Bypass Disable, that will produce a performance hit. And Intel has announced new patches. Intel said: "We've already delivered the microcode update for Variant 4 in beta form to OEM system manufacturers and system software vendors, and we expect it will be released into production BIOS and software updates after the coming weeks. This mitigation will be set" - and this is important, and Microsoft concurs with this. "This mitigation will be set to 'off' by default, providing customers the choice of whether to enable it." So they're saying, and of course the reason is it's a performance hit.

They said: "We expect most industry software partners will likewise use the default off option. In this configuration, we have observed no performance impact." Well, yeah, because you didn't do anything. They said: "If enabled, we've observed a performance impact of approximately 2 to 8% based on overall scores for benchmarks like SYSmark 2014 SE and the SPEC integer rate on client and server test systems." They said: "The same update also includes microcode that addresses Variant 3a" - which is that Rogue System Register Read - "which was previously documented publicly by ARM in January. We have not observed any meaningful performance impact on client or server benchmarks with the Variant 3a mitigation."

So they said: "We've bundled these two microcode updates together to streamline the process for our industry partners and customers. This is something you will see us continue, as we recognize that a more predictable and consolidated update process will be helpful to the entire ecosystem." In other words, they decided to group these together because why not.

So what will happen is it looks like Microsoft plans to also leave this off unless we learn at some point that it's important to turn it on, probably because it represents, I mean, a very significant performance impact on our systems simply to turn this on. Maybe they will be able to go through and, for example, sequester. One of the recommendations has been to recognize that it's no longer safe to mix the physical location of secrets with just less sensitive data, and that secret information needs to be physically placed somewhere in memory where it can be safe.

And so if you worry about, if you understand the nature of this kind of speculation and the attacks on it, it's possible to sequester your secrets so that they are not susceptible to this, even if you leave speculation on. And this is what we were talking about in the future where the immediate worry has been to turn these things off that creates a performance impact, but probably over time we will be able to incrementally protect our secrets and then get back some performance that we initially lost.

In this case it looks like the hit of shutting down this SSB is so big that no one wants to do that, and they're expecting that it'll be possible to go in and, both by softening timing resolution in browsers and eventually by protecting those sensitive areas with speculation prevention instructions like this LFENCE instruction, that it'll be possible to just now take responsibility for the potential security problems that speculation introduces, which as of the beginning of the year we are now soberly aware of.

So bottom line, it's not the end of the world for users. Microsoft will be giving us Intel's updated firmware, which mostly will allow them to respond fast if a problem is found. That is, we'll already have in our chips the ability to shut this down if a catastrophe comes to light, although no one expects that. So more updates from Intel, more from

Microsoft, and more awareness of problems which the developers will have to keep in mind as they move forward and continue to massage the code to keep it secure.

Leo: Yeah. I think that's the other question is what happened to the other six?

Steve: Yes, exactly.

Leo: Where are those?

Steve: Because we heard there were six.

Leo: Do you think we've heard the worst ones? Do we know about the worst ones? Or there could be worse?

Steve: What kind of happened was that this leaked.

Leo: Oh, okay.

Steve: Yeah. So both Microsoft and Google found this. It has that 90-day...

Leo: Disclosure requirement, yeah.

Steve: ...disclosure. And in Jann's posting it does show that they're in - I can't remember what he called it. Oh, "grace period." They exceeded the 90, and they're being given an extension. So it could be that there are still six more problems that we'll be talking about as they come to light in the future. Wow.

Leo: Yeah.

Steve: It's what happens when you have a fundamental hugely widespread, I mean, fundamental technology, this whole concept of caching and speculative execution, which suddenly everyone goes, "Oh, shoot. We really can't do that safely."

Leo: Yeah.

Steve: Yet everybody is.

Leo: Right. Whoops.

Steve: Wow. Oops.

Leo: And now everybody's banging on it, so I expect that we haven't heard the end of this anyway, even if there may be other ones, as well. Well, there you go. Spectre - I guess NG is Next Generation?

Steve: Yup. Spectre NextGen.

Leo: Next bad thing?

Steve: That keeps on giving.

Leo: Keeps on giving. You'll find Steve is the guy who keeps on giving. You'll find his stuff at GRC.com. Of course the podcast is there. Just go to GRC.com and you can download the audio versions plus read the transcriptions, the human-created transcriptions of each show, a few days after the show is done. You'll also find all the freebies that he gives away like ShieldsUP! and Perfect Paper Passwords, SQRL, all the information about SQRL. And his one sole piece of software that he sells, which is SpinRite, the world's best hard drive recovery and maintenance utility. You can help support Steve and the show by buying a copy. Buy two. They're cheap.

Steve: They last a long time, that's for sure.

Leo: The site license is how many copies?

Steve: So the idea is an individual can use it in all the machines they own. We ask corporations that want to use them on all the machines they own to hold four licenses.

Leo: Four is all you need. After that it's unlimited.

Steve: Four is all you need. Then a corporation can use it throughout their organization with my blessing.

Leo: It's very fair.

Steve: And again, they do last a long time. So you get a lot of use out of those.

Leo: A fair way to do it. Thank you, Steve. GRC.com is where you'll find Steve, or @SGgrc on Twitter. He takes DMs, so you can always contact him there. Or leave a feedback form at GRC.com/feedback. You can find this show, audio and video, on our website, TWiT.tv/sn, or wherever you subscribe to podcasts. If you want to

watch us do it live - we had a big studio audience here today, Steve, from Louisiana, Connecticut, Pennsylvania. All big fans. They're a little sleepy now, but that's okay.

Steve: Hi, everybody. Wake up, wake up.

Leo: You've got the comfy chair. That's why he's very relaxed. If you want to be here, just email tickets@twit.tv. We'll make sure we do get a comfy chair out for you. Of course you can watch it on the stream, too. You don't have to come all the way to Petaluma. Just go to TWiT.tv/live and watch the live stream. If you do that, make sure you're in the chatroom. Nice bunch of people there at irc.twit.tv. That's about it on this side of the microphone. Steve, thanks so much.

Steve: Okay, my friend. Talk to you next week.