The Double Pulsar

Description: This week Steve and Leo discuss how one of the NSA's Vault 7 vulnerabilities has gotten loose. A clever hacker removes Microsoft's deliberate - and apparently unnecessary - block on Win7/8.1 updates for newer processors. Microsoft refactors multifactor authentication. Google is about to add native ad-blocking to Chrome - and what exactly are abusive ads? MasterCard's building a questionable fingerprint sensor into their cards. Are Bose headphones spying on their listeners? Ten worrisome security holes are discovered in Linksys routers. MIT cashes out half of its IPv4 space. We've got the return of two meaner Brickerbots, some errata, a bit of miscellany and, time permitting, some "closing the loop" feedback from our podcast's terrific listeners.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-609.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-609-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Oh, boy, do we have a lot to talk about. Steve will answer some of your questions. We will talk about the latest from the CIA code dump, Vault 7. It's actually spreading now to actual Windows PCs. In fact, surprisingly, to a lot of Windows PCs. We'll talk about what to do to mitigate for that. And Steve has some explanations of the fingerprint saga that we started last week. It's all coming up next. Why don't you watch? Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 609, recorded Tuesday, April 25th, 2017: The Double Pulsar.

It's time for Security Now!, the show where we cover you, your security and privacy online with this cat right here, Steve Gibson of GRC.com. He is our mentor, our leader in more cases than not. The question is, what would Steve do? Hello, Steve Gibson.

Steve Gibson: Hey, Leo. Great to be with you again for Episode 609.

Leo: Wow.

Steve: Following on the third dump of what is presumed to be the NSA documents in the so-called Vault 7 dump from these Shadow Broker guys, one of them has gone wild...

Leo: Uh-oh.

Steve: ...on the Internet. So the title for today's episode is "The Double Pulsar," which is believed to be an NSA-designed backdoor which is being dropped in by one of the other vulnerabilities, which Microsoft patched in March. Yet this is a great lesson for us about to what degree does it even help that Microsoft has patched these things. So we're going to talk about that; how a clever hacker removed Microsoft's deliberate and apparently unnecessary block on Win7 and 8.1 receiving updates for newer processors; how Microsoft has refactored multifactor authentication; how Google is apparently, the Wall Street Journal reports, planning to add native adblocking to Chrome.

Leo: I know. Hell froze over, yeah.

Steve: Counterintuitive. Which, well, but, boy, is it good news. And then we're going to look at exactly what abusive ads are because that's now been formally defined. MasterCard has announced they're going to be building what I consider a questionable, for reasons we'll explain, fingerprint sensor right into their next-generation cards. The question about Bose headphones spying on their listeners. Ten worrisome security holes were discovered in state-of-the-art recent Linksys routers, 25 different models. MIT cashing out half of its IPv4 space. The return of two even meaner Brickerbots. A little bit of errata, some miscellany, and time permitting - and today since we're getting a late start we may squeeze on the backend some "closing the loop" feedback from our listeners, which we may get to next week. So we'll just play it by ear.

Leo: Good. Good. We'll get it all in.

Steve: So our Picture of the Week reminded me how much I love curves. And I noticed - in the context of a graph, of course, orthogonal axes. And it's just at some point in my past I realized that that's the way I think. I think in terms of a relationship between two variables where a curve describes them. And I've mentioned this a long time ago, but one of my first jobs out of college was I was the third person in a small company, Minicomputer Technology, that made, not surprisingly, designed and manufactured hard disk controllers for the large, large 5MB, and in some cases you could get 10, or if you were really pushing it with something the size of a dishwasher, 20MB hard drive. And I was doing some of the sales and marketing in addition to the engineering.

And what I realized was, and this was really a consequence of one of the brilliant founders of the company, that the relationship between cost and complexity was not a straight line. That is, in terms of the cost of the controller. As the disk controller increased in complexity and functionality, its cost went up. And if it was a straight line, it didn't really matter where you were on the line because you would always get the same amount of return for your investment.

But it turned out that the way we designed ours, the curve had a real knee in it. And what I realized is, if you operated at that knee, at that inflexion point in the curve, you could - there was a place where the design could give you a lot of function and very low cost. All of our competitors took the wrong approach. They went high function, high cost. And so we were just selling these things like crazy.

Anyway, the point is that maybe it was then that I realized that's just the way I see things. Anyway, today's picture of the week is just a wonderful cartoon from cartoonist Zach Weinersmith which shows the relationship between, on the horizontal axis, our knowledge of physics, and on the vertical axis, how much the universe make sense. And it's wonderful because, if you imagine physics knowledge moving with time, moving forward with time, how the universe makes sense is going up and up and up and up and up and up and up. And so it's making more sense to us as we're better understanding physics. And then, as this curve shows, at some point, probably when we hit quantum physics...

Leo: Yeah, or string theory.

Steve: ...this thing, yeah, just crashes down to zero. And it's like, now, suddenly it doesn't make any sense whatsoever anymore. So anyway, I just love how much meaning you can put into a curve.

Leo: Good point.

Steve: And of course, yeah, they're really valuable that way. So great Picture of the Week. Thank you, whoever it was who shot me that note.

DoublePulsar. So this is from our "this didn't take long" department. Less than two weeks after the third dump of the Shadow Brokers documents - which we believe, and all evidence indicates, originated with the NSA - the numbers vary by researcher, but tens of thousands of Windows machines are today infected with the DoublePulsar backdoor that was disclosed in this third document dump. So 14 days. And it's a major outbreak. In fact, it's being considered the worst outbreak since Conficker, which we covered on Podcast No. 193, back on April 23rd of 2009. And the little tag on the podcast says "Steve analyzes Conficker, the sophisticated worm that has spread to more than 10 million PCs worldwide." Now, this has not spread to that number. On the other hand, that was, what, eight years ago? And here we are again, still with this kind of, with sort of this nature of problem present, which is surprising.

Okay. So DoublePulsar is a RAM-resident implant, that being the term that we first saw being used in the Snowden release of the CIA documents, an implant being something that is implanted into a system for some purpose. And it gets into these machines through the EternalBlue exploit, which we discussed last week because Microsoft patched it in March, in that delayed, well, in that big update which covered all of these various Eternal* exploits, one of which was EternalBlue. But across the industry malware researchers are comparing this, as I said, to Conficker because it is really serious. Conficker leveraged the Windows RPC, the Remote Procedure Call.

And, surprisingly, as we discussed last week, this EternalBlue exploit was an SMB, the Server Message Blocks exploit, that is, the port 445. And as we said, anybody behind a router is safe; although Woody, writing in his Woody for Windows column in InfoWorld, reminded us that, even if your machine doesn't have an exposed port 445, most Windows machines are gluing themselves together on Intranets through 445. I mean, it is the intermachine communications port that Microsoft has settled on. So if any other machine in your Intranet could get infected, then it could spread within to essentially all the machines in an Intranet. So, I mean, this thing is worrisome.

So there's no real consensus about how widespread this is. But more than five million Windows machines - so listen to this. Five million Windows machines currently have port 445 publicly exposed. Which is astonishing to me. And I would ask our listeners to send me a reason, if they know.

Leo: Well, especially since - the chatroom is saying that typically ISPs will block that port; right?

Steve: Correct, correct. ISPs are blocking it. And if you're behind a NAT router it's blocked unless you...

Leo: You open it for some reason.

Steve: Exactly. And so I can't, I mean, I can't imagine anyone crazy enough to deliberately make that port open. And any newer machine has a firewall that's running by default since Service Pack 2 of XP. It's going to be blocked even if you put the machine right on the network. So, I mean, and any of the Windows servers, they'll deliberately open the ports they need for, like, 80 and 443 for HTTP and HTTPS, and maybe FTP and so forth. But they're not going to open 445 unless you really want them to do that. So I'm stunned at the news that a mass scan of the Internet in the last two weeks has shown more than five million - actually it's 5,561,708 machines, IPs, answering TCP connections on port 445.

Leo: That is kind of amazing. Geez.

Steve: It's just shocking. Now...

Leo: There's got to be some explanation. I mean, there's something going on. I would think.

Steve: They've got to be old. They have to be someone who just stuck them on an IP without any concern. Or maybe just like, oh, look, our connectivity is hampered. Let's turn off the firewall.

Leo: Right.

Steve: It's like...

Leo: I wonder if there's some commercial service or gaming machine or, you know...

Steve: No, 445...

Leo: ...the thing is there are so many Windows machines that five million, you know, that's just kind of - it's crumbs; right? It's just crumbs.

Steve: Right, right. So there may be people who don't have a NAT router, who have their machine on. Maybe their machine's infected with something else that turned off their firewall.

Leo: Hmm, yeah.

Steve: You know, it's like, wow. And of course Universal Plug and Play is on by default in all routers. So if anything got into your machine and said, oh, open up that port, I mean, you could have a poorly designed light bulb which said, yeah, we'd like to have 445 exposed. And so now you're open.

Anyway, so that's Windows machines with 445 exposed. Of those, there is a publicly posted, widely used script on GitHub, and I've got the link in the show notes, of essentially a script that pings those machines for the presence of the DoublePulsar implant. So of those five million machines, there are reports of as many as 50,000 today vulnerable and infected. One malware hunter who goes by the Twitter handle @Below0Day, zero as in numeral "0," who I just got a tweet from David Redekop, who said that the account's just been shut down by Twitter. So I guess his postings pushed them over the line. He did a 24-hour Internet scan. And I have a screenshot of what came up on the terminal. It took a little over a day, about 25 hours. And he found - he's the person who found 5,561,708 machines. Of those, 30,626 instances of DoublePulsar implant were detected.

Leo: So a small fraction of the total machines with 445 open were infected.

Steve: Right, right,

Leo: That's interesting, too.

Steve: But, well, but like 10%. So it's like, well, not quite 10.

Leo: And presumably it will spread.

Steve: Correct.

Leo: To all the rest at some point.

Steve: Correct. So Binary Edge, that's a Swiss-based security firm, reported finding more than 107,000 infected machines in their recent multiday scan. Errata Security's Robert Graham, who as we know came to early fame - he was the original author of the

BlackICE personal firewall back in the early personal firewall days. He scanned and found 41,000 infected machines. Dan Tentler, who's the founder and CEO of the Phobos Group, did their own Internet-wide scan. They found somewhere between 62,000 and 65,000 and said that about 3.1% of vulnerable machines were already infected.

So the numbers are varying, but there's no question, I mean, certainly everybody who looks finds tens of thousands, many tens of thousands of existing infected machines. Now, it's not assumed that these were existing implants. They are very likely recent infections as a consequence of these documents going public because this is very well engineered. It's a drop-in, script kiddie-compatible exploit that you just - it's trivial to use.

The good news is, if there is any here, is that it's a RAM-resident implant. It doesn't write anything to the file system. Of course we know that that's one of the ways that these things hide because typical malware scans for known viruses in files, and we're not seeing much RAM scanning at this point. And if we are, it's typically not in the kernel. This thing uses the - since the TCP/IP stack that does all this Internet traffic lives in the kernel, the exploit is in the kernel. And DoublePulsar installs itself in RAM, hooks into the kernel, and rides along on top of port 445. It doesn't open up its own port. Instead it just monitors the port 445 traffic, which is how these various security scanners are looking for it is they know how to ask, to generate a query on port 445 that the DoublePulsar will grab and respond to.

So Matthew Hickey, who's a founder of the U.K. consultancy Hacker House, added that: "The fact that people are using these attack tools in the wild is unsurprising," he said. "It shows you these tools are very well developed, very weaponized, and don't require a lot of technical sophistication. So attackers are quick to adopt them into their repositories and toolkits, and they're using them as-is."

Then Kaspersky added a little bit of technical detail, saying that: "DoublePulsar works on older Windows Server versions with older versions of PatchGuard kernel protection. Modern versions of Windows such as those derived from Windows 10 have better kernel checks that could help block or prevent these hooks deep into the OS. Once DoublePulsar is on a compromised host, an attacker can drop additional malware or executables onto a machine, meaning that this bug will quickly move from the exclusive realm" - and I would argue it already has - "from the exclusive realm of nation-state hackers to cybercriminals, and it may be a matter of time before ransomware and other commodity malware and botnets take advantage of these exploits to spread. One drawback for the attacker is that, since the attack lives in memory, once a machine is rebooted, it's gone."

On the other hand, as we know, it comes back up, and it gets reinfected because nothing will have changed. "DoublePulsar also comes with a kill or burn command that won't remove the infection, but does prevent others from making use of the backdoor."

So anyway, I have a link to Woody's column in InfoWorld. He went further and has a really nice breakdown for anyone who's concerned about which versions of 7, 8.1, and 10, which build versions and knowledge base patch levels you need to have. And of course the short version is, just be current. Make sure that whatever you're using - 7, 8.1, or 10 - that you updated with the March patches because he makes the point that, even if you're not publicly exposed, you could still be attacked from other machines on your network if anything, either this or something else, got into them, since it wouldn't have to be a 445 port exploit that compromised a machine.

But, for example, if something - take Sony, for example. We don't even know today, we never had any details about how this exploit went so wide. But if something got into

Sony, and this, for example, was known before it was patched, which was only in March, then that would allow an Intranet, essentially a massive Intranet exploit within an organization that could then allow you to go from a receptionist's computer, for example, and then leverage that into getting onto a server, and then you install this thing. And so it's a way of going from machine to machine, breaking through the security that would otherwise exist.

So, wow. A bad problem. Microsoft is questioning these numbers, I think because they don't like them. But it's a little difficult to question eight different completely separate security organizations that have all run their own scans and all, while they're disagreeing about the exact quantity, they're all upwards of tens of thousands of machines that have this thing in them today. So we'll see how this goes in the future.

Leo: But was the attack published by WikiLeaks? How are people getting the code?

Steve: Yeah, it was. It was in this...

Leo: They published - it was irresponsible.

Steve: Yes. It was in that third dump by the Shadow Brokers.

Leo: Because at first they - oh, it was Shadow Brokers, not the WikiLeaks. It was the Shadow Brokers.

Steve: Yeah, the Vault 7 disclosure.

Leo: No, that is WikiLeaks.

Steve: Mm-hmm.

Leo: So I thought they weren't going to publish code.

Steve: Ah. They published enough.

Leo: Okay.

Steve: Yeah.

Leo: Okay.

Steve: So this is a little controversial. And I just thought it was interesting. A GitHub

user who goes by the handle "Zeffy" created a patch that removes a limitation that Microsoft deliberately imposed on users of seventh-generation Intel processors which prevents those users from receiving, for example, last month's or this month's Windows Updates, if they still use Windows 7 or 8.1 with Kaby Lake or Ryzen PCs. So it was controversial, of course, even though, as we know, Microsoft told everybody well in advance that this is what was going to happen. Still, people who were choosing to use Windows 7 or 8.1 on the latest hardware discovered that they could no longer receive updates.

What was interesting was that this Zeffy guy on GitHub, he was just sort of curious exactly what was done. So he took a look at the updates that were in Knowledge Base 4012218, which was the March 2017 Patch Tuesday, and discovered two new functions which Microsoft added: IsCPUSupported and IsDeviceServiceable. Those two functions return a Boolean result, true or false, yes or no. Making a one-byte change to "IsCPUSupported" so that it returns a "1" rather than a "0," and everything works. Meaning that it's not that the updates aren't compatible in some fashion with these later version processors. But Microsoft simply wanted to enforce their policy that they would not allow newer processors to operate on older versions of Windows and continue to receive updates.

Leo: So that solves the question because we thought maybe it was a technical issue.

Steve: Correct.

Leo: It's not.

Steve: Exactly. And so that's what's annoying is it's not that they had to do any, like, the engineering of the updates is not compatible, which always did, I mean, I get it that Microsoft doesn't want to have to not support older architectures, but why not wait until they actually don't support older architectures, rather than enforcing the policy because it's a policy? When in fact doing so is denying users of Windows 7 and 8.1, which are being kept updated. For example, I'm getting them on my Windows 7 because I bought Skylake on purpose so that this wouldn't happen to me.

So people who bought newer machines, choosing to stay with older versions of Windows, aren't getting the updates that people on older hardware are through 2020. So three more years of updates, just because. So that's annoying. If anyone is interested - and I'm not suggesting this is a good thing to do because this requires patching and essentially hacking a couple files. All the information is on GitHub. I have all the links in the show notes. If somebody is in this position, it's well vetted. It works. But it does mean that every time Microsoft, like every month Microsoft will probably refresh this, and I wouldn't be surprised if it doesn't work in a month or two. Microsoft will decide, okay, we're just not going to make it as easy to do.

But the cat's out of the bag. We now know that they just added a test - it tests the CPU ID. Is this CPU seventh-generation or not, or later? And if it is, it says "CPU not supported." Not for any good reason except because they said that's what they were going to do. And what's annoying is in the process they're denying people updates for security, which they're saying are important, which people could otherwise have.

A lot of our listeners were wondering about this Microsoft Authenticator change which

was announced last week. And I called it "Microsoft refactoring multifactor." As a result of this announcement, a Microsoft Windows account may now be registered with the Microsoft Authenticator app which is available for iOS and Android - and, interestingly, not for Windows Phone - after which the app will receive a Windows logon confirmation prompt. So you unlock your mobile device, acknowledge the request, and you're logged in.

So the question has been, is this multifactor? And Microsoft says yes because they think that phrase is the holy grail, like being multifactor is automatically more secure. I would say no, it's not multifactor, since "multifactor" means multiple secret factors. And since your username is not a secret, you have been previously relying on your password as a single-factor secret. So when you add, for example, your username - and remember, your username is often your email address, which we know is not secret. So when you add, for example, a time-based six-digit one-time token, that's another secret that's making it multifactor. What Microsoft has done is saying you don't need your password. If you register your Windows account, you log in with your name just to say this is who I am, and then your phone will ping and go, you know, are you logging in? And so you say yes, and then you're good to go.

Leo: But it is, well, but it is something you know and something you have.

Steve: No, it's one factor. And so here's...

Leo: Well, something you know is your username, admittedly not very secret.

Steve: That's not a secret. So that's not a factor.

Leo: You had to use, by the way, for that to work on your phone you had to use your password to activate it on your phone.

Steve: Or fingerprint, which your spouse might be able to unlock.

Leo: No, not just your fingerprint. It won't do it the first time unless you use your password. When you install the app, you have to log in fully to your Microsoft account.

Steve: Okay.

Leo: So you log into your Microsoft account on your phone; right? And then you can lock that with a fingerprint from now on. So you log on on the computer. I mean, you did it all on the phone originally.

Steve: So here's my point. We only need to resort to the added encumbrance of multiple factors. I'm not saying this is bad, Leo. Don't...

Leo: But if you don't have my phone, it's not going to work.

Steve: Correct. So it's one...

Leo: You have to have the phone. It's one factor.

Steve: ...strong factor.

Leo: Steal my phone; right.

Steve: That's my point. We have only needed to resort to the added encumbrance of multiple factors because the factors themselves have been individually weak.

Leo: Right, right.

Steve: So having more individually weak single factors, where they must all be correct in aggregate, provides us with stronger final security. And so what this is...

Leo: So Google and Duo and others do this. But you log on on your computer with your name and password, and then it fires up the acceptance on your phone. So you would call that true two-factor.

Steve: Correct.

Leo: Okay.

Steve: Because you the attacker would have to provide multiple things.

Leo: Right, secrets, yeah.

Steve: And of course I'm all for the idea of a single strong factor because that's the entire basis of SQRL. SQRL is a single factor, but extremely secure solution. So again, I'm not saying that this is like a bad thing for Microsoft to do. I think they should just say, instead of saying, "Oh, this is multifactor," they should say, no, but it's one - and I understand they can't explain this to everybody.

Leo: Right.

Steve: But we can to our audience. If you have one really strong factor, that's good

enough.

Leo: They could very easily just make you enter your password on your computer, like Google does.

Steve: Yes.

Leo: And that would make it true two-factor.

Steve: Yes. And then you would be multifactor.

Leo: And it's better, that's better than a text message, or arguably even an authenticator.

Steve: Oh, I agree. And I think having it tied to something like that, where your mobile device has authentication, I think that's, I mean, that's good, useful security because it means that, if someone grabbed your computer, they could not log on. And in fact your phone would get pinged when they tried. You'd go, oh, look, someone's trying to log on, and it's not me. So I think it's good. What Microsoft is, you know, and all the press coverage they're getting is specifically because they do not ask you for a password because everybody hates passwords. And so they're saying, yeah, we've eliminated the password. It's like, yeah, okay, fine. And as long as they don't have any other bad compromises in their system, I think one strong factor is arguably all you need.

So I know you picked up on this news because you've mentioned it before, Leo. Google, the Wall Street Journal reports, is planning to add native adblocking to Chrome. And I think this is fabulous because they're going to do, very likely, if they pursue this - Google has not commented on the Wall Street Journal's reporting, but the Wall Street Journal's probably, I mean, this makes sense if nothing else. But we don't have confirmation from Google. But I expect them to be able to do for advertising very much what they did for security. And it's been a mixed blessing, as we know.

For example, it was Google's leveraging the power of their Chrome platform that forced changes in the TLS and certificate infrastructure on the Internet because, if Chrome wasn't going to support some features, everybody had to run around and scramble in order to accommodate them. I mean, I went to great lengths to keep GRC able to be viewed right up until New Year's Eve of 2015 in order to make Chrome happy, yet still allow GRC visitors who could only use SHA-1 signed certs to get to GRC. Thus the power of what Google decides to do.

So the Wall Street Journal said: " Alphabet Inc.'s Google is planning to introduce an adblocking feature in the mobile and desktop versions of its popular Chrome web browser, reported by people familiar with the company's plans. The adblocking feature, which could be switched on by default within Chrome, would filter out certain online ad types deemed to provide bad experiences for users as they move around the web. Google could announce the feature within weeks, but it's still ironing out specific details and still could decide not to move ahead with the plan, the people said."

Leo: They were going to, remember, put encryption in Gmail, too, and didn't do that.

Steve: Yeah, yeah. Although this to me seems real clean. Encrypted Gmail, okay. "Unacceptable ad types would be those recently defined by the Coalition for Better Ads" - and I've got a link here in the show notes below, Leo, that breaks out what those are - "an industry group that released a list of ad standards in March. According to those standards, ad formats such as pop-ups, autoplaying video ads with sound, 'prestitial' ads" - which is a term I hadn't encountered before. Instead of "interstitial," these are "prestitial" - "with countdown timers are deemed to be 'beneath a threshold of consumer acceptability.' In one possible application Google is considering, it may choose" - and get this - "choose to block all advertising that appears on sites hosting offending ads, instead of the individual offending ads themselves. In other words, site owners may be required to ensure all of their ads meet the standards, or could see all advertising across their sites blocked in Chrome."

Leo: Woohoo. Of course no Google ads violate these standards.

Steve: Correct. And that's why I think this is a brilliant move because...

Leo: You know what, it was forced because the choice was let everybody use adblockers, and then you're really dead meat, or do something meeting them halfway. I think they had to do this. This is an example of us winning, in effect.

Steve: Yes, exactly. The Wall Street Journal in their reporting said that the "Uptake of online adblocking tools has grown rapidly in recent years, with 26% of U.S. users now employing the software on their desktop devices, according to some estimates." So again, Google, as we know, in 2016 they made $60 billion in revenue from online advertising. They're seeing that threatened because users are responding to obnoxious, I mean, I've listened to you so many times annoyed by self-starting videos playing, like when you're trying to do a podcast and something's there making noise.

Leo: All the time, yeah. It's really annoying.

Steve: Yeah. So I just say bravo to Google for this. So on the desktop they're saying that pop-up ads, autoplaying video ads with sound, prestitial ads with a countdown, and large sticky ads would be banned. All of those also on mobile. Plus mobile ads, if a site has an advertising density higher than 30%, if the animated ads are flashing to grab your attention, if they are positional ads with a countdown, or full-screen rollover ads, those additional four categories would be banned on mobile.

And I hope this happens because what this would do, I mean, this is, again, in the same way that Google leveraged their clout in order to force security to be improved, they're helping us. I mean, I only have adblocking on because, as I've often commented, I look at someone's machine that doesn't have it, and I'm thinking, how can you even see through the ads in order to get to the content? So if this gets fixed - and the point is it'll have to get fixed because, if Chrome won't display it, that's half of the market. Half of

the install base now of browsers are Chrome. And so if Chrome won't do it, the ads will have to back down. So, yay.

This is a bad idea.

Leo: Uh-oh.

Steve: Yes.

Leo: We haven't done a "they're doing it wrong" in a while, you know.

Steve: Well, this is they're doing it dumb, at least. The headline on the MasterCard press release reads "Thumbs Up: MasterCard Unveils Next-Generation Biometric Card." Now, it's clever, I'll give them that. Anybody who's received a credit card in the U.S. at least - and of course this is the EMV standard, standing for Europay, MasterCard, and Visa - you'll have that little contact area a little above the center line on the left-hand side of the card, above the account number and name.

Leo: Let's take a look at Lee M. Cardholder's card.

Steve: Yes, exactly. And he's got an expiration date that I don't think is possible.

Leo: 12/23.

Steve: Yeah, that's way out there. And so the point is, as we know, you stick your card into the terminal, and it only goes about, maybe, what, a little over a third of the way?

Leo: Yeah, yeah.

Steve: So the right-hand side of it is sticking out. Well, they very cleverly put a biometric thumb reader, thumbprint reader in the card. So the card itself, and it's probably capacitive as opposed to optical, so it's probably a capacitive reader. And you can do this because the card is powered by the contact strip. So it doesn't have...

Leo: Ahhhh.

Steve: Yes, that's why this is a...

Leo: It's clever.

Steve: It's a clever idea. You now have a card receiving power from the terminal, so it

doesn't have all of the problems of a battery and thickness and all that stuff. It does have some problems, though. The problem is, as we know, fingerprints are not exact. Which means the card has to know how to decrypt itself. That is, it has to contain the information in it to authorize the transfer. If, for example, it were a PIN pad, where you had to enter a lengthy PIN for security, then the PIN could be hashed, which would be - the exact PIN could have an exact hash that would generate an exact key, which could be used to decrypt the information about your identity and then authorize the payment.

But a thumbprint is not exact. And this is why this whole thing fails. I mean, it's better than nothing, but it's a gimmick. From a cryptographic standpoint, it means that a fuzzy match must be allowed. That means a fuzzy match doesn't produce an exact result. That means that a decision is being made somewhere in there, is this the thumb that I was trained on or not? And in the same way that the hacker changed one byte in Microsoft's March update in order to reenable updates that Microsoft has banned, somewhere there's a single jump command. There's a single decision being made, is this a matching fingerprint or not? And the point is you're not using information that the card doesn't have. You're just saying, yeah, that looks like a thumb I recognize. Well, that means a hacker can hack that in order to unlock the card.

So it's, yes, it's better than nothing. But if we look at the technology that had to be employed, it doesn't mean that this is cryptographically secure. And of course you have to wonder then also how you can hand this to a restaurant server and have him or her run your charge. Because unless you're going to follow them into the back...

Leo: But by itself that makes it more secure; right? He can't do anything without you.

Steve: Exactly.

Leo: You know, in Europe what happens is they don't bring it in the back. They bring a little reader to your table because you have to do the chip, and you insert it and then enter a PIN because they do chip-and-PIN, which we don't do.

Steve: Right.

Leo: So you're saying a PIN would be better than this fingerprint.

Steve: Well, we know that PINs have been bypassed through a different technology because you enter the PIN into the terminal, not into the card. So what I'm saying is, if you entered - and unfortunately the PIN is just compared with the PIN that's in the card, instead of the PIN being used to decrypt information in the card.

Leo: So a PIN is just as bad.

Steve: Yes. The PIN is just as bad. And this is no better, unfortunately.

Leo: All right. Look, it seems better.

Steve: Oh, I know. That's the point is security through obscurity. But it doesn't, you know, because in fact the fingerprint doesn't give you a precise, like, password equivalent. It's just a gimmick.

Leo: And this is why Apple Pay and Android Pay are still the best way.

Steve: Right.

Leo: Most secure way. You're not giving any information to the merchant. You're just giving them a token. You have to use the fingerprint reader on your device to verify that it's you, and those are much better. It's been solved, frankly.

Steve: Yup, yup. So the headlines all over the place again...

Leo: This is why - you've got to explain this because I didn't even read the article thinking, well, I don't understand how that would work.

Steve: Yeah. And it doesn't, Leo.

Leo: Oh, good, thank god.

Steve: Yeah. Even Consumer Reports, that is otherwise a trustworthy organization, I think, in general, but maybe the security and technology's a little tricky. Or again, we know that oftentimes people who write the articles don't put the headlines on them.

Leo: That's right. That's right.

Steve: I had the problem for the eight years I was writing the Tech Talk column. Sometimes I would just cringe when I looked at the headline. It's like, oh, no, that's not what I said. But they do it because they want to get readers.

Leo: Right.

Steve: So the headline is "Some Bose Wireless Headphones Track and Share What You Listen to, Lawsuit Says." So, no. It turns out that there is an optional Bose Connect app which users of headphones which support it, and there's, I don't know, I was going to put them in the show notes, but I thought, okay, I'm not going to read all of those for everybody. It's just, if you have them, you know it. They're smart Bose headphones that are connected, no doubt with Bluetooth, to a Bose Connect app which gives you

additional features which you're able to use. For example, you can change the amount of noise cancellation that the 'phones offer. So there's a little bit of a hook to it.

Well, apparently someone discovered, probably by looking at the traffic that this app was generating, that it was harvesting. In fact, I was thinking of this story when at the end of MacBreak Weekly you guys were talking about the Unroll Me or Unroll It or whatever it was.

Leo: Yeah, Unroll.me, yeah.

Steve: Yes. A serious privacy breach. These guys have been caught, Bose has been caught apparently doing the same thing. Without any explicit user permission, they are going way beyond - because it's not a media player app. It's just there to interface with your headphones. But they are sending back everything you listen to, all of the media that you play with these headphones, and essentially everything you do with them, continuously recording the contents of the electronic communications that users send to their Bose wireless products from their smartphones, including the names of the music and audio tracks they select to play, along with the corresponding artist and album information, together with the Bose wireless product's serial number.

So anyway, this has resulted in a class action suit about collecting all customer data without permission, which the plaintiffs allege is in violation of the Federal Wiretap Act to do this. The complaint says: "No party to the electronic communications alleged herein consented to Bose's collection, interception, use, or disclosure of the contents of the electronic communications." And the attorney representing the plaintiffs said: "This case shows the new world we are all living in. Consumers went to buy headphones and were transformed into profit centers for data miners."

I have, if anyone's interested, a link to the Bose complaint PDF. And there is a company called - I have it here in the notes somewhere. I'm looking for the domain name. It was - I'm not seeing it. Oh, Segment.io is the company whose - and they're one of the recipients of this - a company whose home page says "Collect all of your customer data and send it anywhere."

Leo: Yeah, baby. They're actually a sponsor of our network, so...

Steve: Ah, well.

Leo: But they're not about - they don't do the customer collection stuff. They just take - they integrate with whatever it is you're using for the data collection. So they're not actually, I mean, I don't know. I don't know. That's an interesting conundrum there.

Steve: That's a question.

Leo: They do plumbing. So you put in your app whatever...

Steve: The hooks.

Leo: Put in the hooks. So it's up to companies not to do stupid stuff. And then they plumb it over to whatever databases you want to keep track of. They don't in fact send it to marketers or anything.

Steve: So last Thursday Tao Sauvage, who's a security researcher with IOActive, published the results of his reverse engineering of one of the most recent models of Linksys routers. And of course we've been talking about them, unfortunately, a lot recently. This is completely separate from those previous discussions, again. So this adds to that. In his case, he purchased a recent EA3500 Series router which is part of their Smart Wi-Fi router series. And this made me shudder. Smart Wi-Fi is the latest family of Linksys routers, which includes 25 different models that use the latest 802.11N and 802.11AC standards. Okay. So that's the good news.

The bad news is that they can be remotely managed from the Internet using the Linksys Smart Wi-Fi free service. So he didn't even look at that. I mean, again, remotely managing your router from the Internet? What could possible go wrong? And by the way, there are four WRT models among those 25. So there are 21 EA models and four WRT. I've got the list in the Linksys link to their own disclosure.

So this guy and a friend extracted and forensically examined the router's firmware, identifying simply by inspection and then verifying by sending some packets at them, 10 different security vulnerabilities ranging in risk from low to high. Six of those 10 are remotely exploitable by unauthenticated attackers. Two of the security issues they identified allow unauthenticated attackers, meaning anybody on the public Internet, to create a denial of service condition on the router. So you can crash it.

By sending a few requests or abusing a specific API which will respond without authentication, the router becomes unresponsive and reboots. The admin is unable to access the web admin interface, and users are unable to connect until the attacker stops the DDoS. So this you could imagine would be fun for kiddies to blast people who they want to keep off the Internet, if you have a Linksys router that has got this exposure. And it appears that this is exposed by default.

Attackers can also bypass the authentication protecting the CGI scripts to collect technical and sensitive information about the router. So there are CGI scripts whose authentication can by bypassed, which allows them to obtain the firmware version and Linux kernel version, the list of running processes, the list of connected USB devices, and the WPS PIN for the WiFi connection, of course which then allows you to get onto the router if you're within range. Unauthenticated remote attackers can harvest sensitive information using available APIs to list all connected devices and their respective operating systems, access the firewall configuration, create FTP configuration settings, or extract server message block, that is, SMB server settings. Furthermore, an authenticated hacker, meaning someone who can log in remotely, which raises the bar, but unfortunately only eliminating 88%, still allowing 11% of the vulnerable devices, of which there are about 7,000 at the moment.

So an authenticated attacker on 11% of the currently exposed 7,000 routers can inject and execute commands on the operating system of the router with root privilege. So one possible action for such an attacker would be to create backdoor accounts, gain persistent access to the router. Backdoor accounts would not be shown on the web admin interface and cannot be removed using the web admin account.

It should be noted that they did not find a way to bypass the authentication protecting that vulnerable API. And that authentication is different from the authentication protecting the CGI scripts which can be bypassed. And, however, this is where they discovered that 11% of the approximately 7,000 currently publicly exposed Linksys routers were using default credentials. That is, admin and password or admin and admin, whatever the default login is that's out there flapping in the breeze, 11% of the 7,000 routers. Which then allows somebody to log in and obtain root and put in persistent accounts.

So this is a nightmare. They responsibly disclosed the vulnerabilities back in January and have been sharing the technical details with Linksys. Since then they've been in constant communication with Linksys to validate the issues, evaluate the impact, and synchronize their respective disclosures, which were both made last Thursday. And these guys, the IOActive guys, noted in their report that Linksys has been exemplary in handling the disclosure.

So I think Linksys has a better owner now in Belkin, who purchased Linksys from Cisco some time ago, a better owner in Belkin than they did in Cisco because Belkin has apparently jumped right on this. They're being very proactive. They have published security advisories offering temporary solutions to prevent hackers from exploiting these vulnerabilities while they work on getting new firmware available. And as we know, it's often the case that you need to go get the firmware for your router. Shodan can be used to search for and has been used to search for these vulnerable devices. That's what turned up 70,000 of them. 69% are in the U.S. The remainder are spread around, with 10% in Canada, 1.8% in Hong Kong, 1.5% in Chile, the Netherlands has 1.4%, and then on to smaller percentages. So nearly 70% of them are in the U.S.

I've got links to the Linksys note with a list of all the vulnerable versions. If you happen to have an EA Series or a late-model WRT, it's not the older, very popular WRT54 or whatever they were. They're all more recent routers that have the Smart Wi-Fi stuff. And so Linksys says enable automatic updates, disable the guest WiFi network if you're not actively using it or when it's not in use, and by all means change the default admin password. And I would say, my god, turn off WAN side admin if you don't really, really need it. Or minimize the attack surface by only enabling it if you're, like, for whatever purpose you have for needing it, if you're going to be away. It's just always a bad idea to have that enabled.

MIT, I love this little piece of news, is selling off half - get this - of their 16 million IPv4 addresses. Back in the 1970s MIT's senior research scientist and a researcher with the MIT Computer Science and Artificial Intelligence Lab, which is CSAIL, saw the importance of IPv4 addresses and requested an early allocation of them, both to support research and to eventually support all of computing at MIT. They were given the entire 18-dot Class A IPv4 network, so all IPv4 addresses beginning with 18, 18 dot anything dot anything dot anything. And as we know, that's 24 bits. So that's 16 million IPs. 14 million of those 16 million were never used. And they recently concluded that at least 8 million, or half of their original allocation, are excess and could be sold without impacting their current or future needs. The funds raised from the sale will support MIT's migration to IPv6. And Amazon was the winning bidder, purchasing that IPv4 space from MIT.

Leo: Interesting.

Steve: Yes. And we've talked about the IPv4 space depletion in the past and how IPv4 is

a commodity which is pulling some serious money. And Leo, the link here in the show notes to IPv4auctions.com is really interesting. It'll give you and our listeners an update on what's going on. They are subject to discount in quantity, but they are currently selling for around 11 to $12 per IP.

Leo: Wow.

Steve: Yeah. And so what MIT said was that they're going to take this cash windfall from Amazon and use it to build out their IPv6 infrastructure. And they already have a bazillion IP - well, everybody can have a bazillion IPv6 IPs because there are - I think they had a nonillion number, that is, MIT's chunk. But, so, yeah, you're scrolling now through the recent auctions for various size networks of IPv4 space.

Leo: Why is there variation in price? Are there some numbers better than other numbers?

Steve: Well, it's the size of the network. Normally you get a quantity discount. So the larger the network, the larger the unbroken block, the lower the price per IP.

Leo: Yeah, but I see a - oh, I guess the /24s are all about 3,500. All right. Yeah, you're right. There's a consistency. There's some variation, but /22s are 12 grand. What is MIT selling off? Is that a /4? A /8?

Steve: Let's see. That would be a /9. And you never see those.

Leo: You don't see any of those in here.

Steve: No. So that's a huge - that's eight million. Now, we don't know what price Amazon paid. But eight million times $10, that's $80 million, which Amazon said, yeah, we'll buy it.

Leo: It's worth it to them.

Steve: Yeah.

Leo: Of course.

Steve: I mean, for all that stuff they're doing, absolutely.

Leo: Yeah, yeah.

Steve: So very, very cool.

Leo: Some people aren't rooting for IPv6.

Steve: Well, IPv4 is here. And so if you're Amazon, and you can drop $80 million in order to get eight million more IPv4 addresses, I could see where it makes sense. If you've got a serious, serious cloud need for them, for, like, the hosting that they're doing.

So also last Thursday - Thursday was a busy day last week - Brickerbot 3 and 4 both surfaced. Now, remember that Brickerbot was so named because it is a bot that bricks your devices. It goes beyond just inhabiting them. It uses a series of commands to try to erase your file system from your IoT device. It uses the same entry point, the Mirai exploit, which is essentially any busybox-based Linux device that has the Telnet port publicly exposed with the factory default credentials would be a potential victim. So these are security cameras, some DVRs, as we know, basically the things that got pwned by the Mirai botnet previously. Brickerbot is going after them and, when it can, just killing them. Just wiping them out.

Brickerbot 3 and 4 are clearly from the original author. The attacks are matured. They've eliminated some things that weren't effective. They've added at least four more different ways of bricking devices. And they're also attacking more ferociously and from geographically distributed IPs and different ones than before. And the industry has heard from the author. The author goes by the handle Janit0r, with a numeric "0," J-A-N-I-T-0-R. And he reached out to a Victor Gevers, following up from a comment that Victor had made in one of the first articles about Brickerbot.1 and .2, as opposed to .3 and .4, who confirmed that he's the author and had two things that he was quoted saying.

First he said: "Like so many others, I was dismayed by the indiscriminate DDoS attacks by IoT botnets in 2016. I thought for sure that the large attacks would force the industry to finally get its act together." Okay, well, we're talking light bulbs, people. "But after a few months of record-breaking attacks, it became obvious that, in spite of all the sincere efforts, the problem could not be solved quickly enough by conventional means."

Second quote: "I consider my project a form of Internet chemotherapy." Actually, maybe that should have been the title of this podcast.

Leo: Yeah.

Steve: Anyway, he says: "I sometimes jokingly think of myself as 'the doctor.' Chemotherapy," he writes, "is a harsh treatment that nobody in their right mind would administer to a healthy patient; but the Internet has become seriously ill in Q3 and Q4 of 2016, and the moderate remedies are ineffective." So this vigilante is killing off devices that he is able to access and that have writeable file systems. Okay.

A couple bits of errata. Vasile noted from Episode 608, he said: "Just to be meticulous," and he said, "I know you treasure 100% accuracy, Unicode has space for up to" - and then he did this in hex format - "0x110000 code points, more than could fit into 16 bits. They can be encoded in multiple ways, ranging from variable-length UTF-8 to fixed-size UTF-32." And he's completely correct. Remember last week I talked about how ASCII uses 128 because essentially it only uses the lower seven bits. Extended ASCII is twice

that because it uses all eight bits. So it's got all of the seven with the high bit off, and then additionally all of another seven with the high bit on. So that gives us 256 code points.

Unicode is divided into planes, with 16 bits per plane. And last week I was only referring to what's known as the basic multilingual plane, which is 16 bits and, as I said, 64K code points. But, and Vasile is right, there are also up to 16 additional supplemental planes, each having an additional 64K code points, for a grand maximum total of 17 64K planes, totaling 1,114,112 code points. Which is arguably why you could accommodate pretty much every emoji that you ever needed to without worrying, I mean, and even with different skin colors, which we're now seeing.

So thank you, I'm glad to have the correction and to note that I was just talking about the basic multilingual plane, which is 16 bits. But as he notes, yes, and there are 16 more of those. So we're not going to run out of space in Unicode. And in fact I'm well versed in this because GRC's SQRL client is, as everyone will remember, explicitly multilingual. And I'm using UTF-8 encoding in order to be able to handle any character set that should come along.

Also Rick, who tweeted as @rpodric, said: "@SGgrc Just a note regarding the apparent fix in Chrome 59 for punycode" that we discussed last week. He said: "59 is the dev version. 57 is current stable, with 59 due by June 6." Now, this is a puzzle to me because everyone was saying that Chrome was broken. And my Chrome is fixed, and I'm back on 49. So I don't know if it got fixed earlier, if maybe Google pushed out a fix just for this. But I did put links for everyone to be able to verify specifically that the tweak, if they're using Firefox, where you turn off the punycode recognition, and it will show you the raw true domain name in the URL. I put that there so people could verify that that was working for them. You might want to check Chrome. I just assumed it was fixed for everyone. So I don't understand why everyone, like that morning of last Tuesday, apparently it was being fixed. So maybe Google just pushed out a fix across the board in order to fix that. I didn't track down the details to determine that.

Leo: There were patches pushed out.

Steve: Yes, good. Two bits of miscellany. @elheffe said: "Not sure if I should thank you or be mad. Frontier Saga is sucking my productivity away." And so I replied to him, I said: "Yeah, tell me about. I've finished all 19 books in print. And if anything, the second series starts off even better than the first. Book 3 of the second series," I wrote, "Is unbelievably good. Worth reading everything up to there just for the setup." I mean, I'd have to say Book 18, oh, my lord.

Leo: How many thousands of pages is this? I mean...

Steve: It would be thousands because they're all, like, 350 - 250 to 350 pages. They vary a little bit. And then he replied to my response, saying: "I'm halfway through Book 6. Had to tear myself away to work on cleaning the garage. You weren't kidding about it being nonstop action. Thanks for everything you do. Keep the recommendations coming." And I just - I wanted to cite this one. I've had a lot of people come back and say, okay, I'm not getting anything done any longer.

Okay. And finally, just a bit of fun. This is a less than two-minute-long YouTube video. I

was reminded of it by someone who tweeted this, saying, "This explains so much." And it comes off really well in audio. So Leo, if you could share this YouTube video, the link is in the show notes for anyone who wants to share it around. But it's just too fun.

[YouTube: Turbo Encabulator]

BUD HAGGERT: For a number of years now, work has been proceeding in order to bring perfection to the crudely conceived idea of a transmission that would not only supply inverse reactive current for use in unilateral phase detractors, but would also be capable of automatically synchronizing cardinal grammeters. Such an instrument is the Turboencabulator.

Leo: We should point out that this is a scientist - and we know that because he's wearing a lab coat and a pocket protector - standing in front of a blackboard with a sign on it that says "catalytic converter." And to his right is some sort of, it looks like, frankly, it looks like the space shuttle, some sort of spacecraft.

Steve: Sort of a schematic of a transmission.

Leo: Schematic, yes. We'll continue.

[YouTube: Turbo Encabulator]

BUD HAGGERT: Now basically the only new principle involved is that instead of power being generated by the relative motion of conductors and fluxes, it is produced by the modial interaction of magneto-reluctance and capacitive diractance.

The original machine had a base plate of pre-famulated amulite, surmounted by a malleable logarithmic casing in such a way that the two spurving bearings were in a direct line with the panametric fan. The latter consisted simply of six hydrocoptic marzlevanes, so fitted to the ambifacient lunar waneshaft that side fumbling was effectively prevented.

Leo: He's really serious about this.

Steve: It's so good.

[YouTube: Turbo Encabulator]

BUD HAGGERT: The main winding was of the normal lotus-o-delta type placed in panendermic semi-boloid slots of the stator, every seventh conductor being connected by a non-reversible tremie pipe to the differential girdle spring on the "up" end of the grammeters.

The turbo encabulator has now reached a high level of development, and it's being successfully used in the operation of novertrunnions. Moreover, whenever a forescent skor motion is required, it may also be employed in conjunction with a drawn reciprocation dingle arm, to reduce sinusoidal repleneration. It's not cheap, but I'm sure

the government will buy it.

Leo: $750 million. Oh, that is hysterical. That is great. What is this? What? What's the story?

Steve: Had you not encountered that before?

Leo: Never seen that before. That's classic.

Steve: Oh, it's very rare that I'm able to show you something that you haven't seen before.

Leo: Classic doubletalk; you know?

Steve: Oh, god, it's just - and in his white lab coat, and he's deadly serious. Oh, anyway, it's just [crosstalk].

Leo: And you know anybody in a lab coat is probably pretty sophisticated.

Steve: Oh, yeah. Don't try that at home. Anyway, I just wanted to share that with our listeners. Anyone who hasn't encountered it, it's just wonderful. And I don't know how you could find it on YouTube. Again, I have the link in the show notes. Maybe look up "encabulator." I think that's the - I bet if you google "encabulator" you could probably...

Leo: Probably go right to it, yeah.

Steve: You can probably go right to it.

Leo: Or "waneshaft" and "girdle spring." You know...

Steve: That would do it, yeah. And the double flamulated dipple guard.

Leo: I want to memorize that.

Steve: It's wonderful.

Leo: Just come in so handy. Oh, my god.

Steve: So I do have an apology from someone who didn't give me his real name. He

goes by "KeenDreams." And he said: "A slightly different SpinRite story with an apology." This was dated last Wednesday, the 19th. He said: "Dear Steve: First off, thanks for the informative podcast. I've been listening since I started grad school five years ago and have learned quite a bit thanks to you.

"I was recently feeling nostalgic and decided to buy an old Win95 laptop off eBay to play some of the DOS games from my youth. It was great fun at first, but my excursions into the world of Commander Keen" - and by the way, that's this guy's handle, KeenDreams - "excursions into the world of Commander Keen were interrupted a week later when the laptop stopped booting. The first thing I thought was my copy of SpinRite, which saved my butt back in undergrad once or twice. When I booted it up, however, I was surprised to see a name I didn't recognize at all in the license field. Confusion came over me as I starred at the screen. Then it dawned on me. I must've pirated it.

"I felt so bad that I couldn't start the scan until I sent a 'yabba dabba doo' your way. But needless to say, the old machine was back up and running after my now-legitimate copy of SpinRite worked its magic. My sincerest apologies, Steve." Hey, the guy has nothing to apologize for. "I would use the excuse of being a poor undergrad who desperately needed his research papers back, but that doesn't change a SpinWrong into a SpinRite."

Leo: Oh, I like "SpinWrong." I like that.

Steve: Well, he bought a copy. And as our listeners know, I understand reality, and I appreciate the note sharing his success. And I replied. I said to him, "Look, thank you. I appreciate your support and sharing this." And I also told him, if he's got Windows 95 or, for example, 98, you have got to try ChromaZone. 95 and 98 and machines back then were able to use eight-bit color mode, which is the 256-color mode. ChromaZone was a product that I wrote...

Leo: Using the Turbo Encanabulator, I might add.

Steve: Yeah, exactly. It was how I taught myself Windows. As we know, I always say, if you want to learn a language, find a problem to solve in that language. And so I wanted to learn how to program Windows. So, I mean, ChromaZone is in many ways my masterpiece of Windows programming. All kinds of custom controls, doors that slide open, slide switches that have multiple positions, just a 3D sphere that you rotate with the mouse cursor. The problem is it's all 16-bit assembly language. And it is a palette-editing tool. That is, on machines that could barely run DOS, this thing animated the entire screen.

And what was unique about it is it is a screen saver construction set. More than 500 - back then we had a BBS. And so we were publishing the screen saver creations of ChromaZone customers at the time. If you google - I think it's GRC.com/chroma.htm. It's not linked to the website. It's not in the menu. But you can see some sample pictures of what ChromaZone does.

Anyway, the point is I sent him all the ChromaZone files in my reply email so that he could bring it up and play with it. I think I provided 400 screen savers, and our customers provided an additional 500. So you were able to create, you were able to design, it was like a screen saver construction set. And back on machines, as I said, that just couldn't do anything, this thing animated the entire screen. So it was very fun.

Leo: That's hysterical. It's an orphan page now. Aw. Aw. Oh, and I have just been handed, I didn't realize, but I've just been handed the GE Manual for the Turboencabulator because this is HBK-8359, in case anybody wants to get it, from December 31, 1962. Function, operation, technical features, ratings, the whole thing is here. I guess this was owned by Roger L. Pommerenke. And you could see, in fact, all of the - to measure inverse reactive current in unilateral phase detractors with display of percent realization is the purpose, of course.

Steve: And why does that sound familiar?

Leo: You can get some accessories: 8 ounces 5% tetraethyliodohexamine with 0.01 halogen tracer solution, or the interelectrode diffusion integrator. All of these can be added to your device, if you haven't run out of money after the $750 million. And even reference texts, too. You know, "Zeitschrift fr Physik der Zerfall von Dunge" and other esteemed journals. So this - I'll pass this along to you because...

Steve: Very nice.

Leo: This is an historic document for the Turboencanabulator. And, man, I just think this is - it's great that we have this. I didn't realize. Burke brought it. Or, no, Alex Gumpel brought it over because, in case, we have the manual. In case we need to use...

Steve: Well, if your Turboencabulator ever goes wonky on you...

Leo: The worst. The worst. And it's vital to the operation, of course.

Steve: Absolutely.

Leo: I did buy the Kindle version of - because it was 5.68 - of that Frontier series. It was cheap. It was like five bucks.

Steve: That's for the first three books, I think.

Leo: Yeah. I know. I didn't get all 18.

Steve: One through three, yeah.

Leo: I'm going to give him three volumes, see [crosstalk].

Steve: Well, see what you think.

Leo: I've read only a few pages because I had other stuff to read for shows and stuff. But I really liked - I was kind of pleasantly surprised. I really enjoyed it. So I think I will like it, yeah.

Steve: Good.

Leo: Can't wait.

Steve: And for what it's worth, the author clearly has this planned out. I read his whole back story because I'm just curious where he came from. And he, like, wrote something a while ago. He used to run a PC repair shop. So I was wondering, I wonder if he knows about SpinRite?

Leo: Oh, of course he does.

Steve: Yeah.

Leo: You should write to him. He's in - where is he? He's in the Bay Area, I think, for some reason.

Steve: Yeah, Ryk Brown. I think you're right, I think he's in the area somewhere. So closing the loop with our listeners, a couple fun, or a little collection of fun things. Andy Patuszak said: "I really like your idea of capturing and keeping two-factor authentication setup QR codes. But the problem is, I have had two-factor authentication in a few places for years. Is there any way to get those QR codes again?" And the answer is I've not found any. No one will normally give them to you. Most authentication apps don't export them, which I think is good. You want them not to volunteer them because that subjects them to being captured.

But so I responded to him, and I'll share my response. I said: "Andy, I had the same problem, since I had already established several accounts with Google Auth, which won't export," and I said, "for which I'm glad for security's sake. Fortunately, every service I have encountered so far will allow you to change your TOTP, your time-based one-time password secret. So I just logged in one last time with the original code, asked the site for a new one, and then printed that new one out on paper for all future needs on all devices."

So I had a lot of feedback from people who liked that idea, who are beginning to build up their collection of printed-out two-factor QR codes. And so if anyone else has had this problem, you're normally able just to tell a site where you already have an account established, I want to change mine. And like, okay, here's a new one. And so that's the one you then capture.

Leo: It makes the old one no good, so obviously...

Steve: Correct.

Leo: Yeah. So you have to change - see, the problem is, if you've done it already in a few things, and then you want a new one for a new phone, you've got to go back and change them all. Your authenticator's broken everywhere else because you need - but that's okay. What I do is I get a screen cap and put them, as I mentioned before, put them in LastPass. So they're secure, and they're there; right? And then I just take a picture of it into the phone.

Steve: Right. Okay. So...

Leo: LastPass, by the way, will give you your QR code again.

Steve: Oh, will it?

Leo: Oddly enough, yes.

Steve: Oh, interesting.

Leo: I'd show you mine, but...

Steve: No, no, no, no, no. We have some early results from our listeners' fingerprint tests. Paul Dawson said: "Hi, Steve. After listening to SN-608" - that's last week - "and your article about smartphone fingerprint sensors, I decided to put mine to the test. I have an iPhone 6 with iOS 10.3.1," the latest. "I've taught my phone two fingerprints - well, both thumbs, actually. I have asked" - get this - "84 people to try and unlock my iPhone with their fingerprint. I am glad to report that not one of them managed to achieve this."

Leo: Well, that's good.

Steve: Yeah. "Since my iPhone locks out the fingerprint detector after a few failed attempts, I even used my passcode to allow people several additional attempts at gaining access. From my findings, I think I am happy that the fingerprint system is secure enough for me. Best regards, Paul Dawson, Lincolnshire, U.K. P.S.: Love the show."

Andy Norman said: "Surely our own prints from other fingers are likely a closer match than a random finger. Have not managed to unlock iPhone with my other fingers." And I don't have an opinion one way or another about whether one's other fingers tend to track the other ones. I don't know anything about that.

Terry E. Snyder, Jr. said: "I have an iPad Air 2 and secure it with my thumbprint. I discovered to my chagrin that my three-year-old son is able to unlock my iPad with his thumb. The first time I saw it happen, I was using the feature that locks an app to be the only app he is allowed to use. Next thing I knew he was out of the app. I just thought the

app crashed. The next thing I knew he was able to unlock my iPad without my fingerprint. And I watched him never try to type in my passkey. After hearing the latest Security Now! episode, it finally all makes sense. Thanks for a great show, and long-time SpinRite user. Looking forward to its next release and SQRL."

And finally Phil said: "Wow. After listening to @SGgrc, I let my wife try to unlock my phone with her fingerprint. Works about one in 10 tries. Scary." And then he said in a separate tweet: "After removing a bunch of saved fingers, it stopped working. Wonder what I got in there?" So of course we know the more fingers you put in, the softer the matching will be because it's going to be not looking for one match, it's going to be looking for any of those. So now you're taking an inherently fuzzy match and making it way more fuzzy so that it's way more tolerant because it wants to match any of the things it's got registered. So if somebody was concerned, at the cost of the inconvenience of registering fewer fingers, that's clearly going to increase the security of your system.

But then NeoRenfield tweeted something that I thought was interesting. He sent me a screenshot from his Android phone where Google says that the Pixel fingerprint reader "may be less secure than a strong PIN, pattern, or password." So they acknowledge right upfront that there's fuzzy matching going on.

So the screenshot says, under the topic of About Fingerprint Security: "We strongly recommend locking your screen to help protect your device. Your device's fingerprint sensor gives you a convenient unlocking option. But there are a few things to keep in mind: One, a fingerprint may be less secure than a strong PIN, pattern, or password. Two, a copy of your fingerprint could be used to unlock your phone. You leave fingerprints on many things you touch, including your phone. And, three, you'll be asked to add a backup PIN, pattern, or password. Remember your backup because you'll need to use it sometimes, like after restarting your device, or if your fingerprint isn't recognized."

So props to Google for right upfront saying, yeah, it's convenient, but it's a fuzzy match. So it's not like a long, strong password. On the other hand, you always have it with you, and you don't have to memorize your fingerprint because it is you.

BlueLED sent me a link to Gorhill's documentation on uBlock, which I was unable to find. And as we already know, Gorhill is a cantankerous developer. He does things his way, for his own reasons. And when I was putting the show together last week, talking about uBlock Origin, and reminding myself how his undocumented UI functions, I couldn't find the documentation. So anyway, this BlueLED person sent it me. Thank you. I have the link in the show notes, which is an explanation of the various columns in that expanded UI and what they all do.

Steven Doyle asked: "If ISPs were to start requiring certificate installation, would your HTTPS fingerprinting still indicate a man in the middle?" And the answer is yes. As our listeners know, I created that HTTPS Fingerprinting page specifically because the only danger or concern at the time was corporate middleboxes, as we're calling them now, which intercept HTTPS TLS connections and break open the encryption for the purpose of checking them for malware and content. I never foresaw until recently the concern that maybe ISPs were going to end up becoming essentially like the corporate middleboxes for ISP end users, much as corporations are looking at all the traffic of their customers, or their employees, rather.

So this certificate inspection may end up being far more useful and have widespread purpose than I recognized at the time. And so, yes, it will - the idea is that GRC has no

overlord. I'm getting an unfiltered connection direct from Level 3, a Tier 1 provider on the Internet backbone. So I'm seeing the certificate from the website. And so the question is, is the certificate that an ISP's customer's receiving, does it have the same fingerprint, the same hash as the one that GRC sees? There's a potential for false, what would that be, false negative, that is, for GRC getting a different certificate and the user getting a different certificate from the user in some instances where that's the design of a large, huge website, like maybe an Amazon or a Google, where they're minting their own certificates from their own intermediate CA, which allows them the ability to do that. In that case, there could be a difference. But your typical website has one certificate that everybody receives. And in that case there should be a match.

But in any event, if you get a match, you know it's definitely - that your connection from your ISP is definitely not being filtered. And so, yeah, because of the way it is, it's detecting any interception of your connection, whether your employer, or perhaps in the future your ISP.

Chris Sullivan asks, he was experimenting with the "puny" Apple site that we discussed in SN-608. And he says he found that LastPass did not get tricked and would not give his creds to a phony site. And that is, of course, that's one of the nice things about these integrated password managers is they match on the actual ASCII, that is, the seven-bit domain name which is in the DNS in order to decide if you're at a site that they know about.

For what it's worth, SQRL gives you the same kind of protection. It also would not be fooled. What's being fooled is the user visually when the punycode is converted into Unicode and displayed as it's meant to be by browsers that do that. And I think we're seeing the rapid end of any browser doing that. Firefox, you know, I imagine they will get around to flipping that switch by default soon because this is just such a problem otherwise.

And Thomas Smailus asks: "How was punycode ever anything but a bad idea if the DNS system doesn't also support it cleanly?" And I agree. This is, you know, here we have a problem of the original DNS, which has not changed, I mean, even from day one. We're extending the records that DNS can serve. We're trying to secure it with DNSSEC, but that migration is coming along slowly, as does all core fundamental changes to the Internet. And the problem is it just - it was designed in the U.S. by people on Unix systems on minicomputers, PDP-11s with seven-bit code. So the original RFCs are seven-bit ASCII. So the only way to extend it was by creating some ad hoc, after-the-fact hack which would allow an expression of higher code point alphabets. But it was going to be incompatible unless we did something like this.

So this was the best we could do, trying to layer something on top of a system that was never designed to have it. And I don't disagree that it was ever not a bad idea. It's unfortunate that this can be abused and that hackers are going to take advantage of it. I think that just means that browsers are going to lose the ability to show people large character set domain names. Maybe they'll show them both, you know, side by side or something. I don't know. Or maybe if the visual display doesn't match the domain, do it in a different color or something. Who knows? We'll see how this evolves.

Two last ones: Martin Badke says of the keypad, which was our Picture of the Week last week, where the 1, 2, 3, and 4, the print, the ink was completely rubbed off. He said: "The garage entry keypad COULD have repeated digits. Number of codes possible then would be 4^4. Still sad." He said: "I've seen similar for cash safes." And I would counter that, if the keypad had one digit repeated, which would be 4^4, then only one button would have the ink rubbed off of it. So the picture showed four buttons rubbed off, and

we know from those typical keypads they tend to be a PIN of three or four digits. So it does look to me more like it was 24 combinations, rather than 4^4. And what would that - that would be 256, it was 4^4. Nah, I think it's probably 24.

And then, finally, P. Hoffman said: "Possible mitigation for ISP snooping: OpenVPN server in Amazon's EC2." That's their Elastic Cloud, or Elastic Computing Cloud service. He says: "Less than a dollar a day. Easy to change the IP address at will. Your thoughts? Thanks." And that's not the first time I've heard that suggested. And I think that makes a lot of sense. In fact, I imagine that's really worth exploring. You had the advantage then of not being tied to a VPN provider whose business is VPN, where you inherently have a high-density concentration of interesting, potentially interesting to authorities, traffic emerging from a single location, the VPN endpoint. Instead, it's just Amazon. And all kinds of traffic is coming in and out of Amazon. And now they've got eight million fresh new IPv4 IPs to assign to their customers.

So the idea of running an OpenVPN server, spinning one up when you need one, I think makes a lot of sense. You have to take a look at the economics in terms of cost and traffic and how the pricing works. But I think it makes a ton of sense. So, yes, if it can be done economically, I love the whole profile of that. And that's our podcast.

Leo: Perfect timing. You're an amazing fellow. You planted an arrow in your Encontabulator, figured out exactly - Turboencontabulator, figured out exactly where to fire it.

Steve: Yes. The Entabulator by itself was the first version, and they decided they needed to beef it up a little bit.

Leo: Turboentabulator. You bet, yeah.

Steve: So it is the Turboentabulator, yeah.

Leo: Got to watch that video again.

Steve: It's wonderful.

Leo: You'll find Steve at GRC.com. That's his home. He has lots of great stuff there, including SpinRite, his bread and butter. Got to have bread and butter if you've got a home. All you've got to do is go to GRC.com and go to SpinRite and buy it. And that way, if you've got a hard drive, you can maintain it. You can recover it, if you need to. Awesome product. You'll also find the podcast there, audio and transcripts. And lots of other wonderful freebies. You can ask him questions at GRC.com/feedback, for the show. You can also tweet him. @SGgrc is his Twitter. He even takes direct messages, if you've got a tip. He likes tips.

You can go to our website, TWiT.tv/sn, for video as well as audio versions of the show. And of course everywhere, you know, you get your podcasts you could subscribe. And we would love it if you would subscribe because that kind of evens

out the download numbers for us, makes it easy for us to keep track of who's listening, how many, and all of that.

We do the show, if you'd like to watch live, pretty much always, well, it's always on Wednesday, pretty much always at 1:30, although that may vary. Sometimes we're held up by previous shows. So that's a rough estimate. It's not a train station. 1:30 Pacific, 4:30 Eastern, 20:30 UTC. Stop by on Tuesdays, not Wednesdays - see, I said it wrong already - Tuesdays, and join us for the conversation. The chatroom is irc.twit.tv, and you're always welcome in there, too.

Meanwhile, we're going to let Steve go and get ready for Tech News Today. Thank you, Steve.

Steve: Thank you, my friend. See you next week.

Leo: Bye-bye.

Steve: Bye.