Listener Feedback #180

Description: Steve and Leo discuss the week's major security events and discuss questions and comments from listeners of previous episodes. They tie up loose ends, explore a wide range of topics that are too small to fill their own episode, clarify any confusion from previous installments, and present real world 'application notes' for any of the security technologies and issues we have previously discussed.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-435.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-435-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Lots of news. We'll talk about the "60 Minutes" piece on the NSA. Steve will answer some questions about that, and we'll also answer some of your questions. It's a Q&A episode next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 435, recorded December 18th, 2013: Your questions, Steve's answers, #180.

It's time for Security Now!. Drink.

Steve Gibson: Gulp.

Leo: Gulp. But what we're drinking is coffee, so it's okay.

Steve: Yes.

Leo: Take a sip every time Steve takes a sip. Security Now! is the show all about your security and privacy online.

Steve: If you could stay in your seat.

Leo: Got it. Now, that looks like a nice little ceramic mug. Steve Gibson's here. He's

our...

Steve: That is. But I'm pouring, I'm refilling it with this.

Leo: I see.

Steve: My Contigo.

Leo: Contigo.

Steve: I really like my Contigo.

Leo: Mm-hmm. I ordered that Indiegogo or Kickstarter thing.

Steve: I did, too, yep.

Leo: Yeah. So I'll be interested to see - it's got some sort of special properties for keeping coffee warm.

Steve: Well, what I love is that the coffee that comes right out of the pot would scald your mouth.

Leo: It's too hot.

Steve: It would burn you. So you pour that coffee into this, and it immediately drops its temperature down to drinkable temperature by absorbing the heat. But then it gives that heat back over the next few hours, holding the coffee at that temperature.

Leo: Perfect.

Steve: I think so.

Leo: I hope. I hope.

Steve: We'll see if it's true or nonsense. That would be good, yeah.

Leo: We shall see. So this is our last show for, well, wait a minute. I don't know.

Steve: No, no, no.

Leo: Next week is Christmas. Going to be...

[Talking simultaneously]

Steve: Yes. We've got, well, we have our time capsule episode, or actually three episodes from our blast from the past, when you and I first were in physical proximity with each other, when we first actually met face to face.

Leo: "Meet space," we call it, yes.

Steve: Back in - meet space - back in '98, 15 years ago. And then - and I left in, I mean, I just basically, the whole - it's an hour long. So it's three pieces of three shows with some fun commercials from back then, and you being you back with The Screen Savers show...

Leo: Kate Botello.

Steve: ...and Kate Botello. And I was looking for, well, actually I'm sure I have some that's got Sarah, when she was in diapers still.

Leo: She was in the - in diapers. She was in what they called the babe - what I foolishly at the time called the "babe corral." It was tongue-in-cheek, I assure you. I don't think I made that up. But, yeah, she was. But she was a regular with tips and so forth. Yeah, that was later. That was the...

Steve: And Kevin, looking he didn't need to shave.

Leo: Kevin Rose, 12 year old, yeah, yeah. He didn't, actually, yeah.

Steve: Yup, exactly. So I'm sure I have those. I have everything we ever did. So over the next...

Leo: Wow, we could do this for years.

Steve: Over the next few years this will be our Christmas Special. So then...

Leo: So that's next week.

Steve: And, yeah, we did, we had such a long podcast of news last week that, even

though it was nominally a Q&A, we didn't get - we answered two questions. So I've removed those from the top and added a couple to make up. So we're going to do another Q&A since, I mean, we've not been doing Q&As. That as our first Q&A in a month when we did that last week because there's been so much news. Today we've got pretty much all things NSA. We had a week of crazy NSA stuff. The Constitution, "60 Minutes," Obama, Silicon Valley, and who actually is Edward Snowden. Then we have the just-occurred-today news of a new side channel attack on crypto keys. Listening to a laptop, you can decode a 4096-bit RSA key.

Leo: Oh, my goodness.

Steve: I know. And then a lot of people have asked me - because Microsoft made some news by announcing that they've joined the FIDO Alliance, which is the Internet authentication alliance that Google and Yubico and a bunch of companies are doing. So everyone wants to know, well, what's that relative to my work with SQRL. And a few sci-fi tidbits and 10 great questions. So a great show.

Leo: Holy cow. We'd better get going.

Steve: Yeah. So, okay.

Leo: So did you watch "60 Minutes"?

Steve: I did, and we'll talk about that in a second. I thought it'd be fun to play this YouTube video, Leo, into the podcast. It's not very long. It's only a couple minutes. And people, I think, will get a kick out of it. It's clever.

Leo: Is this from the NSA, or is this a...

Steve: This is "The NSA Is Coming to Town," rather than Santa Claus.

Leo: Unfortunately, there's a 15-second commercial in front. So we'll just pause for a moment and breathe deeply, enjoy the silence.

Steve: Take a sip from your coffee, everyone.

Leo: Here we go.

[Video clip]

MALE VOICE: Ah, the holiday season. A time for celebration, magic, and spending time with the people you love. But don't forget who's watching to make sure you're not being naughty.

SONG: You better watch out, you better not Skype, you better log out, yeah, you better not type, the NSA is coming to town.

Leo: Santa in sunglasses.

SONG: You're making a list, they're checking it twice, they're watching almost every electronic device...

Leo: Edward Snowden at the top of the naughty list.

SONG: The NSA is coming to town.

Leo: And he swaps the phones, takes a picture.

SONG: They see when you are sleeping, they hear while you're awake...

Leo: Only in New York.

SONG: They know who you call and who you write so encrypt for goodness sake.

Leo: He took his hat and stepped on it.

SONG: With Congress in the dark and a cloak-and-dagger court, we're looking for answers and we're coming up short...

Leo: He just ran off with her phone. That's cold.

SONG: The NSA is coming to town.

Leo: Uh-oh.

SONG: They're making a list, checking it twice, they're watching almost every electronic device, the NSA is coming to town.

Leo: Obviously these are real people who are getting a little miffed.

Steve: Uh-huh.

Leo: This is from the ACLU.

SONG: The NSA is coming to town. The NSA is coming to town.

Leo: I love it. The ACLU keeping up their record...

MALE VOICE: You wouldn't let government agents spy on your special holiday moments in person. Why are we letting them do it in the digital world? Help us end the NSA's unlawful spying program.

Leo: Wow.

MALE VOICE: Click here and take action now.

[End video clip]

Leo: ACLU. That's something. So if I get a takedown from the ACLU, I'm going to tell them to call you.

Steve: So I don't think - I think they'll be quite happy that we're spreading this around.

Leo: They should, but you never know.

Steve: This holiday cheer. Just so everyone knows, all of the show notes are now always at GRC.com on GRC.com/sn. So, for example, the link to this, which Leo just played, if anyone wants to, who just heard it, isn't looking at the video, wants to play this, it's on YouTube. You can get the link right here in the show notes, the same one that Leo just clicked on. So everyone is certainly welcome to do that [youtube.com/watch?v=8pcWlyUu8U4].

So, yeah. The NSA on "60 Minutes." That generated a lot of Twitter traffic with people saying, oh, come on. And in fact even the EFF said, they tweeted: "We planned to write a takedown of @60Minutes' NSA puff piece yesterday, but then the DC District Court did it for us eff.org/r.fudf." And so they gave a little link to it. So, I mean, for me what was most interesting was right off the bat the, quote, "investigative journalist" who did the piece was no one we've ever seen before.

Leo: Yeah, he happened to be - have a lot of ties to law enforcement over the years.

Steve: Yes, he used to be DNI. But I thought it was interesting that...

Leo: Defense...

Steve: That's, shoot...

Leo: He worked for the feds.

Steve: Yes, yeah.

Leo: He was intelligence. Defense and National Intelligence.

Steve: That's it.

Leo: Or Department of National Intelligence or something.

Steve: But what you normally have for, like, big stories is one of the core "60 Minutes" team. And it struck me as very odd that for a story of this clear import that they had chosen to do, none of their main people wanted to weigh in on this. And in fact, I mean, it was a puff piece. What annoyed me was that, on camera, to the "60 Minutes" investigator's face, they were just saying what we know to be lies. I mean, they were asked: "Is the NSA collecting data on millions of Americans?" "No, we're not doing that." Well, okay, wait a minute. The next day...

Leo: We're just collecting phone records, metadata. They even said that later in the piece.

Steve: Yeah, they did. And so there was - although it was a different person. So there was some - so the piece was also self-contradictory. So again there was, like, there were no follow-up questions. No one was pushed hard. They made a point of saying, "Oh, I've never been interviewed before. I've never been on camera." And it's like, okay. Anyway, I was, yeah, our listeners were just sort of rolling their eyes. So I didn't think very much of it.

And of course the next day, Monday, came the news that a federal judge ruled that in fact the mass collection of telephone metadata is unconstitutional. So there's a huge collection of lawsuits. Everybody is suing everyone. There are shareholders suing IBM because IBM has indicated, or IBM didn't tell them that there would be a consequence to IBM's international sales. Apparently IBM's China sales has just take a big hit because China's now worried that IBM is a branch of the NSA and is spying on them. And so the shareholders of IBM are suing IBM. Meanwhile, we've got all kinds of class action suits against AT&T and Verizon for sharing the data. I mean, it's just a disaster.

So one other thing that happened is that yesterday, on Tuesday, the White House invited 15 of the top Silicon Valley tech execs to come and sit around a table and have a photo event, basically. Tim Cook of Apple, Eric Schmidt of Google, execs from Twitter, Microsoft, Facebook, Salesforce, Netflix, Etsy, Dropbox, Yahoo!, Zynga, Sherpa Global,

Comcast, LinkedIn, and AT&T. So sort of like the - although I don't see any Verizon among those, but maybe they were there, 15 of them in total. And so this was to, I don't know what, air everyone's grievances. Barack apparently tried...

Leo: Ostensibly it was about the HealthCare.gov debacle.

Steve: Oh, was it, also? Because I knew that that was part of it.

Leo: That seemed to be, well, the stories before the event said "meeting with Silicon Valley to talk about HealthCare.gov." Of course these executives had a different agenda. And apparently the President deflected them as best he could.

Steve: Yeah, well, for example, the Washington Post had an article. And they quoted an industry official who was familiar with the companies' views as saying, for example: "What the hell are you doing? Are you really hacking into the infrastructure of American companies overseas? The same American companies that cooperate with your lawful orders and spend a lot of money to comply with them to facilitate your intelligence collection?" So that was - that characterized the sentiment of these companies.

And I did see an interesting counter-take on this that was a little bit of a reminder, and that is to say, well, let's not forget that these companies, to varying degrees, live by tracking and profiling their own users. It's literally their published business model. Eric Schmidt has been quoted saying: "We know where you are. We know where you've been. We can more or less know what you're thinking about. Your digital identity will live forever because there's no delete button." So he's got to be careful that he's complaining about the government spying when many consumers are concerned that Google is watching them too carefully. I think, of course, that there is a difference. One is disclosure.

Leo: You can also opt out.

Steve: Exactly. And power. Google doesn't have the ability to knock on...

Leo: They don't have [indiscernible].

Steve: Exactly, to knock on your door and disappear you under the Patriot Act and deny any opportunity for you to have legal representation. You just are gone. And our government does have that power. And I remember a lesson I learned way back, and you'll remember, Leo, in the spyware days, when the term was born, "spyware," because I discovered this stuff on my own machine, this Aureate spyware, which was later renamed Radiate. And this was - it was when I was beta testing the Zone Alarm firewall, which was the first firewall that did outbound blocking. And so I installed it on my machine, and a notice popped up telling me that something I'd had no knowledge about was trying to connect to the Internet. And it's like, what?

And it turns out that PKZIP had installed this. It was part of their business model. They brought this stuff in. And even though I had registered it, it was - you downloaded the

unregistered version, and then you would give them the license, and then it stops displaying ads. Well, this was a technology that displayed ads in a window in the application. And so this was the framework for that. But, and even though I think I'd even uninstalled PKZIP, I was using something else by then, the instructions were explicit from Aureate, do not uninstall us because we can be shared by many different freeware in order to monetize freeware.

So this wasn't malicious. But this was something I never was told about. It was not consensual. And so I wrote OptOut immediately in order to remove this stuff from my machine, which was - it turns out this was phenomenally widely spread through the Internet, and nobody knew about it. And so it generated a huge response. And my point is that I was privy to the letters and email people were writing to Aureate. I mean, beyond livid. And some of the language and the terminology, I mean, with reference to totalitarian regimes in the past. It was sobering to see how upset people were.

And what I realized was it's because this was a surprise. I mean, this was in their machine, and they had no knowledge of it. It wasn't that it was malicious back then. Stuff today is much more so. It's just that this happened without their knowledge. And I think also it was the beginning of this. Now everyone's kind of like, okay, well, I've got to get this stuff out of my machine. But this was the coining of the term "spyware." This was the first evidence of this kind of thing happening. So people were really upset. And so my point is that I'm sure that a lot of the heat generated over the whole Edward Snowden revelation series is we just really didn't know. On some level, yeah, okay, there was an assumption. But we've seen slides with graphic details. And so that's a different story.

Leo: That was another funny part of the "60 Minutes" thing is the only debate over Edward Snowden was should we give him amnesty so we can get the other files back from him? Or should we just, he's a terrorist, so throw the book at him. You don't want to negotiate.

Steve: Right. And the guy who understood the depth of the breach and the number, 1.3 million, I think, files was the number that was cited, and I've seen that elsewhere, he was of the opinion, well, it would be really nice to have those back because he knows now the extent of what, if they have a number like that, they have a sense for what has gone missing.

Leo: I would guess they would know exactly. I mean, there must be logs. I would guess they...

Steve: I don't know. Maybe. But certainly they have a count. And he knows actually how much on his good behavior Edward has been. I mean, what hasn't been disclosed is a lot relative to what they believe he got. And again, it's the case that Snowden had a specific goal, and that was of generating accountability. And he is achieving that, and not more. And so I really think he's acted responsibly so far.

Leo: I didn't ask you, and I don't want to sidetrack you because you've moved on. But about this BIOS virus that they were describing in this piece.

Steve: Okay. So the thing that I was put in mind of was the Chernobyl virus. That was the thing that I wrote the free recovery tool for. I called it FIX-CIH because it was also known as the CIH virus. And there are BIOS Flash ROM-erasing viruses around. This was a hard drive-erasing virus. But there have been some viruses that specifically wipe the BIOS and turn your board into a brick. And so that's what I'm thinking. I mean, I heard a lot of people pooh-poohing it. It's like, well, if somebody wanted to...

Leo: Generally the MO of people who spread malware these days, I mean, that's just malicious.

Steve: Well, right, because it kills the platform that the thing's on.

[Talking simultaneously]

Leo: ...what they want.

Steve: Right, exactly. And this has always been the argument against why viruses don't kill the machines they're living on is that there's always a chance they can spread more if they stay alive. But if, for example, a foreign power wanted to really, I mean, seriously hurt the U.S., I mean, this is like Daniel Suarez sort of, you know...

Leo: Yeah, this is cyberwarfare stuff.

Steve: Yeah, cyberwarfare.

Leo: But again, I think more successful bringing down the power plant in other ways. You know? It's, you know.

Steve: Well, think about it, Leo. If it were actually possible to spread something that zeroed the firmware on motherboards, it would, in fact...

Leo: That'd be pretty bad.

Steve: Oh, it would be truly devastating.

Leo: Seemed as if they were looking for something that would be as scary as possible to make people feel better about the work they're doing.

Steve: Although - okay. See, the other thing that has come out is it's very clear, I think, to anyone looking at all of the news, that the NSA lies to Congress. They exaggerate the benefit and the effect of this vacuuming of everything. And, for example, someone matter-of-factly said, oh, yeah, we've thwarted at least 50 terrorist plots using the mass metadata collection. And then upon further analysis it's like, okay, well, talk to us. Which

ones? Oh, well, we'll have to get back to you on that.

Leo: You know, that one.

Steve: You know, those bad ones.

Leo: Those ones. Well, they can't, I mean, in their defense, they can't - they're not really going to say...

[Talking simultaneously]

Steve: And as I've said, I really appreciate the tension that exists. I mean, in my reading I did see somebody, a third party saying - and this is one of the talking head shows - mentioning that - but one of the smart people - saying, you know, the NSA says they can't tell Congress because Congress will leak it.

Leo: Right. General Clapper said, "I said as much as I could. I was as honest as I could be."

Steve: And Leo, I've seen Congress. They would leak it.

Leo: Of course they would.

Steve: I mean, I wouldn't trust Congress as far as I could throw them.

Leo: But that raises the issue of, if you can't trust Congress, who do you trust to do oversight? Somebody has to do oversight. And you can't - and I think it's not unreasonable to say you can't reveal all this stuff publicly. I mean, you can't let the American people decide. Their representatives have to decide. This is not - I don't think it's all that clear.

Steve: Okay. So one thing I wanted to - I want to get off this topic. Maybe in the New Year we'll be able to.

Leo: Yes, yes, sorry, yeah, yeah.

Steve: No, no, no, no, no. I mean, like, sort of in terms of podcast future because we've beaten this thing to death. But one of the things that annoyed me was the characterization we have seen of Snowden, and we saw it in the "60 Minutes" piece. There was the one little anecdotal business about, well, he cheated on a test in order to get hired. And there is a - I understand that the NSA has to paint this person as a traitor and the worst thing that ever happened to national security and a high school dropout and, I mean, really paint him as badly as possible.

But there was a piece that I linked to for anyone - oh, in Forbes. So this is in Forbes. And this is going a little bit, I think, overboard on the other side, where "An NSA Coworker," the headline is, "Remembers the Real Edward Snowden as a Genius Among Geniuses." Now, okay, all of our geeks, all of we geeks have mothers who think we're geniuses. So that's sort of, okay, you have to understand who's doing the thinking here. I'm sure that the janitor in the NSA building thinks that all of the NSA employees are geniuses.

But, for example, before coming to - and so this is from the interview of an NSA coworker who knew Snowden well, said that: "Before coming to the NSA in Hawaii, Snowden had impressed NSA officials by developing a backup system that the NSA had widely implemented in its codebreaking operations. He also frequently reported security vulnerabilities that he discovered in the NSA's own software," and apparently "many of these bugs were never patched. Snowden had been brought to Hawaii as a cybersecurity expert working for Dell's services division, but due to a problem with the contract was reassigned to become an administrator for the Microsoft Intranet management system known as SharePoint.

"Impressed with his technical abilities, Snowden's managers decided that he was the most qualified candidate to build a new web front end for one of its projects" - maybe they could get him over on HealthCare.gov. Anyway, "despite his contractor status. As his coworker tells it, he was given full administrator privileges, with virtually unlimited access to NSA data. 'Big mistake in hindsight,' says Snowden's former colleague. 'But if you had a guy who could do the things nobody else could, and the only problem was that his badge was green instead of blue, what would you do?'

"As further evidence that Snowden didn't hijack his colleagues' accounts for his leak" - which is one of the bogus stories we heard in order to paint him in that light, "the NSA staffer points to an occasion when Snowden was given a manager's password so that he could cover for him while the manager was on vacation. Even then, investigators found no evidence Snowden had misused that staffer's privileges, and the source says nothing he could have uniquely accessed from the account has shown up in news reports." Of course, we know that not everything Snowden has has shown up in news reports, but still there's another bullet point.

"Snowden's superiors were so impressed with his skills that he was at one point offered a position on the elite team of NSA hackers known as Tailored Access Operations," or TAO. "He unexpectedly turned it down and instead joined Booz Allen to work at the NSA's Threat Operation Center. Another hint of his whistleblower conscience, aside from the telltale hoodie" that he always wore. He wore an EFF hoodie that showed that eagle with its talons holding all the fiber optic cables. So, I mean, he was saying to people, look, I'm with a spying organization here, folks. "Snowden kept a copy of the Constitution on his desk to cite when arguing against NSA activities he thought might violate it. The source tells [the reporter that] Snowden also once nearly lost his job standing up for a coworker who was being disciplined by a superior." And this goes on.

So my point is that there's ample evidence to say that Snowden had a conscience, that he was demonstrating it at work, and he was - characterizing him as a high school dropout who cheated on a test in order to get into the NSA is clearly not telling a fair recitation of the facts. So there.

Leo: So there.

Steve: Now...

Leo: Take a sip, everybody.

Steve: Maybe we're done with the NSA for 2013.

Leo: Yeah, let's move on. I think that's a good idea, yeah.

Steve: It's been a major topic for 2013. We will try not to have it dominate 2014. But speaking of dominating, I posted, I tweeted a neat graphic which popped onto the 'Net last week showing - I think I tweeted "The bots are winning," or something to that effect. This is the result of an analysis showing the bot versus human traffic distribution, which is to say the traffic on the Internet generated, not by human activity, but by automated activity. In 2012, 51% of the traffic was nonhuman agents. In 2013, that 51% has expanded to 61.5, so an additional 10%. Human use has dropped to 38.5.

Leo: But that's kind of to be expected. This is machines talk to machines and can do it much more rapidly than a human can.

Steve: Yes. And so, for example, search engines, of the 61.5 majority of the traffic, 31.1, so half of that 61.5%, 31% is search engines and other good bots.

Leo: And this isn't a surprise to anybody who has their own website because, if you look at the log, there's always a crawler from one of the search engines in our site. Always.

Steve: And in fact the other day I looked at my Perfect Passwords page, which, I mean, I'm using a password for my WiFi, to protect my various WiFi routers, that I got from GRC. I just can't get a more absolutely unbiased random blob designed for WiFi than from GRC. But on the day that I looked at it, there had been, like, 12,000 average pulls. And it normally runs, like, three or four. So I'm thinking, okay, it must now be that some automated thing is just going there and sucking a lot of pages in order to collect random noise from GRC. It's like, okay, well, I've got to go put a stop to that one of these days. It's not on my priority list, but...

Leo: What is Netflix? Is that a bot? I mean, if I go watch a stream, is that a human interaction?

Steve: I wouldn't think so.

Leo: How do they classify that?

Steve: Well, so 31% is search engines and other good bots. Then 5% is scrapers, and

what I just described was a scraper. Something is scraping my site. I designed this page for a person, one person to go and get one piece of very, very high-quality pseudorandom data for their own use. But something, I'm thinking, is scraping my site. And it's the same way, like Craig's List has complained about scraping. Other people set up bots to go and, like, scrape. And eBay has the same problem. So they're scrapers who are pretending to be humans that are collecting data from websites. Five percent of the bots are doing that. Four and a half percent are hacking tools. I don't know what that actually means, what they qualify. Half a percent are spammers. And then 20.5 are called "other impersonators."

Anyway, just some interesting stats that, over time, clearly human use of the Internet is increasing from 2012 to 2013. We know that's increasing. As all of us now have smartphones, there's additional points of entry to the Internet, allowing us to use more of our day hooked to the 'Net. So human traffic on the Internet is growing, but nonhuman traffic is growing faster than human traffic such that we're losing out as a percentage, which is interesting. As you said, not unexpected. But to me the reason it makes sense is there's never been anything more automatable than the Internet. It's a bunch of protocols that obey rules. And so, yes, a human running a browser can click a URL and pull up a page. But so can a machine. And so machines are. People have found all kinds of reasons to have machines doing that.

I got a bunch of tweets earlier in this week, maybe it was the end of last week, concerned with the news that Google was now displaying images in Gmail. Previously, Google, when you got email that had embedded images, you would have a "click to display this image" if you want to. So that did a couple things. It saved bandwidth, so that you weren't downloading stuff that you didn't care about. But there was also some protection there because, in the same way sort of that NoScript protects you by not loading a script by default on a website you visit, this wasn't loading an image by default on email you were viewing.

And we've spent a lot of time talking about how sad it is, but true, that actual images can be malicious because the renderers in our computers that convert that image into something we can see, essentially the code representation of the image - JPEG, GIF, whatever, TIFF - into an image, that rendering code can have buffer overflows. And it's possible to then, if you find a flaw in the rendering, you can craft, like, a fake image which will deliver malware.

So what's very cool about this - so a lot of people were saying, hey, Steve, I thought, like, this was a security problem. And what Google has done is extremely cool. They are proxying images. And here's the key: They are transcoding them. And there are a number of reasons for doing this. But what that means is essentially Google's own servers are following the link and obtaining the image and interpreting the image, turning it into a graphic. Then they're taking that and re-encoding it as whatever makes sense, a PNG or a JPEG, and then embedding that in your Gmail. So it's now safe for Gmail to display images by default.

And so this means a couple things. First of all, that act of transcoding is the best filter you could ever have. For example, I'm using it on GRC's packets. I don't talk about my security technology often. But from the beginning, before I began doing things, I implemented a packet transcoding technology which re-represents the packets in a meta language, and no packets from the outside ever come to the inside of GRC, only transcoded packets. And it's technology I've never done anything with. I just did it for myself.

But it provides protection because it essentially - it completely sanitizes, in Google's

case, an image by discarding what could be a malicious specific representation and turning it into an image and then re-encoding it. But the other thing Google can do is that many times somebody may send you an image which is like ridiculously high resolution, tens of megabytes in size. So Google, because they're proxying it, is able to, in the transcoding process, is able to essentially recompress it to a size that makes sense.

So it comes into Gmail. And you might be able to click on it to get the full size one, but it shows you a nice little thumbnail, much smaller, at a compression ratio that makes sense for the type of image and your application, which then makes your page load much faster than if your browser was going out and having to download a 20MB photo that somebody sent you in full hyper resolution without reducing it. So it's very cool. And it means a few other things, too, that they comment in their blog posting where they describe this, and that is that senders cannot use image loading to get information like your IP address, which they otherwise could.

If your email is collected by Google, then Google is the store-and-forward server. But when Gmail comes to your browser, and your browser opens it, its requests for images will come from you and not from Google, which means that anyone wanting to track you based on images added to email would be getting queries from your browser at your IP address, and then they would know your location. And they would be able - that would be a cookie transfer. So they'd be able to set and read cookies on your browser through images. Both of those things are thwarted by having Google proxy images in Gmail. So that's also very cool.

But the one glitch is, if images are tagged with unique filenames in their links, then they would still be able to track you that way. That is, if you opened the email, and then your browser asked Google for the image by name, then Google would query the image in order to transcode it and send it back to you. And so there would still be tracking by unique image name. But still, a nice step forward.

Leo: And you can disable it. You may ask, well, why would Google spend all the time and money to do this, because it's expensive. Because they want you to see ads, frankly. And they want - I don't really want to see those images. So I've immediately gone into my settings and turned this feature off. I mean, they're not - it's good they're sanitizing it. And I guess for most people seeing images in their emails is kind of better. But I don't want to see ads in my email. And so, fortunately, you can go into settings and turn it off, at least for now.

Steve: Right. So they flipped it back on, and you can go in and say no, no thanks, again.

Leo: And they do explain there why it's okay now to have images in emails and so forth. They don't have the - I'm curious what this is going to be. Ask before displaying external images. Used to be they wouldn't display them at all. And what I suspect is people like MailChimp and Constant Contact said, hey, you know, we do newsletters with images. You turned them off without asking. So my suspicion is we're going to get pestered every time there's an image. You want to see these images? Come on. I know you want to see these images. So, you know.

Steve: Yeah.

Leo: It's an ad. I love Google. I'm not going to complain about Google. I love Google. But it's an ad company. And this is about commerce.

Steve: Yes, and that's exactly it. We are Google's product, as has been said. They're monetizing us. And that's, I mean, television does. And so it's funny, Leo...

Leo: TWiT does. We do. I mean...

Steve: Yeah. I was commenting to Jen a couple days ago, I stumbled on a site, I don't remember now what it was, but it was clean. It was like GRC. No ads. And but obviously it wasn't mine. But I just looked at it, and it's like, wow. And I just - I realized then, look at the typical - look at what has happened to the typical website. It's just - it's all across the bottom. It's half of the - it's all down the right-hand side, stuff jumping around. And then now, as you scroll down, new things, like, slide out of the bottom in order to grab your attention. And I went to some other site that had no ads, and it's just like, oh, it was nice.

And so, yeah. It is incredible what's happened to the 'Net. And as you know, I'm sure, the file download sites have all just become unusable now, too. They're all forcing you to, like, download some intermediate download manager of their own and install junk on your machine other than the file that you're trying to download.

Leo: Yeah. There is a difference, you know, I really - there must be a rhetorical - you know how there's names for all sorts of rhetorical tricks. There must be a name for this rhetorical trick. "But Leo," says somebody in the chatroom, "you're always saying it's unethical to use adblockers. What's the difference?" Because it's in my email. That's different. I'm not using a free service like Facebook and saying, well, I don't want to see the ads. This is my email. And I don't want to see bazooms in my email, thank you very much. Off of soapbox.

Steve: Right. So, okay. We have to talk about this because everyone's just in a frenzy. What was shown was, at this point, a purely academic, theoretical attack. It is definitely interesting, and it's the kind of thing that I love to bring to our listeners because this is the kind of stuff that really makes a difference in fundamental understanding of security. And this is a group of researchers who published a paper, and in the show notes is the link to it [www.cs.tau.ac.il/~tromer/acoustic/#]. I'm just going to share their synopsis because they do a perfect job of characterizing this. And we'll talk about it in a little bit. The title of their paper is "RSA Key Extraction via Low-Bandwidth Acoustic Cryptanalysis." And so they said: "Many computers emit a high-pitched noise during operation, due to vibration in some of their electronic components." And this actually is the switching power supply, but we'll talk about more details in a second.

"These acoustic emanations are more than a nuisance. They can convey information about the software running on the computer, and in particular leak sensitive information about security-related computations. In a preliminary presentation, we have shown that different RSA keys induce different sound patterns, but it was not clear how to extract individual key bits. The main problem was that the acoustic side channel has a very low bandwidth" - naturally, because basically it's an audio carrier, so variations in the audio carrier are going to be limited to the carrier's frequency, which is on the order of 20 kHz,

which is like the switching power supply frequency. So that's why some of us can still hear those switching power supplies.

And then, so they say - and of course they're limited by microphone bandwidth. They say "...under 20 kHz using common microphones and a few hundred kHz using ultrasound microphones, many orders of magnitude below the GHz-scale clock rates of the attacked computers. Here, we describe a new acoustic cryptanalysis key extraction attack, applicable to GnuPG's current implementation of RSA." They were actually able to extract GPG RSA keys.

"The attack can extract full 4096-bit RSA decryption keys from laptop computers of various models within an hour, using the sound generated by the computer during the decryption of some chosen plaintexts. We experimentally demonstrate that such attacks can be carried out, using either a plain [microphone planted] next to the computer, or a more sensitive microphone placed four meters away. Beyond acoustics, we demonstrate that a similar low-bandwidth attack can be performed by measuring the electric potential of a computer chassis. A suitably equipped attacker need merely touch the target computer with his bare hand or get the required leakage information from the ground wires at the remote end of VGA, USB, or Ethernet cables." Oh, my lord.

So anyway, so we've talked often about side channel attacks. This is clearly a really cool low-bandwidth side channel attack, basically using things the machine does. Apparently vibration is another one. And sound is one. And subtle variations in the electric potential between the machine's ground. So essentially, when you have an algorithm, a cryptographic algorithm where secret information changes what the machine does, even in subtle ways, that is detectable. It varies the load that the processor puts on the power supply. Varying the load that the processor puts on the power supply varies the ground reference of the machine, just by microvolts, but enough that you can detect it. And it will vary the strain on the power supply that changes the sound that the power supply makes.

And so technically it's all leaking information. It's one of the reasons, and I've mentioned this before, that the elliptic curve technology that I chose was deliberately designed by Dan Bernstein to have none of its secret information involved in any branch decisions or changing the flow of the machine in any way. I mean, so it is side channel neutral. But it's very difficult to write code that way and takes an extra effort. What we're seeing is it is incredibly important to do this. We talked last week or the week before about another attack, which was using variations in the code flow which change the caching of code and data in the microprocessor. And so malware running in the same processor, but in a completely different virtual machine, but like in an Amazon shared hosting environment, could obtain cryptographic information from other processors running in the machine.

Now, okay. So the reason we don't have to all run around with our hair on fire is this was an academic demonstration. First of all, the machine could not be doing anything else at the same time. It was absolutely set up so that it was only doing this crypto, using chosen plaintext, over and over and over and over and over. So, I mean, that's all it was doing, with no contention. If you were in a typical laptop where you've got all kinds of annoying stuff connected to the Internet, downloading stuff and updating and other tasks happening at the same time, all of this goes out the window.

Now, that's not to say it's then impossible, but it's very much like if everyone in a coffee shop, like in Starbucks, was absolutely quiet, and one person was talking to someone else, you could hear their conversation across a room. Whereas, if everyone is talking to someone else, yes, that original conversation is still there. It's part of the mix of what you're hearing. But oh, my lord, distinguishing it from the cacophony is vastly more

difficult. So, similarly, in any real-world situation, it's going to be much more difficult to extract keys.

But this has been a focus, a recent focus of cryptographic work, the nature of side channel attacks. And so what we're going to be seeing in the future are going to be explicitly side channel safe crypto. I'm already using it. Other people are going to have to start using it just so that you know you're safe against these kind of attacks because they're inherently stealthful. It's machines and processors and boxes and hardware that you may have no control over whatsoever that are unintentionally changing what they do. I mean, it's like the fundamental nature of the way the processors function is leaking this information. So now we have to start designing crypto so that it is side channel-safe against these kinds of attacks. But this was, wow, really interesting.

Leo: You've heard of Van Eck phreaking; right? This has been since the '80s.

Steve: Well, and remember, Leo, being an old-timer like me, we used to put AM radios on the top of mini computers, and we would write code with different lengths of loops in order to generate different frequencies and then, like, play Christmas carols and things, back in the old days. And so this is the same sort of thing. There it was the core memory. Core memory, the nature of the way it operated was pulses of current through loops of wire, and that was inherently generating radio frequencies all over the AM band. I mean, basically you couldn't listen to an AM radio inside of a computer shop. And so we would tune the radio to someplace where we'd get a good intermodulation, and you could play music from a computer, just even though it had no sort of interface at all. It was just generating - you could get it to generate deliberately really strong RF emissions.

Leo: There's a scene in "Cryptonomicon" where somebody uses Van Eck phreaking to see what's on somebody's monitor through a hotel wall because the radiation from a monitor does travel through walls.

Steve: Oh, yes.

[Talking simultaneously]

Leo: That's why there's a TEMPEST specification for highly secure...

Steve: Yes. And back then when you had big CRTs that were scanning, they were sending out the contents of their screen moment to moment.

Leo: Right, right. Hey, we have a winner in the Mega Millions lottery. I'm surprised you don't have this in your notes because you're big on math. Apparently there are a lot of people confused about math. Mega Millions, if they hadn't had a winner, would have gone to over or close to a billion dollars in the lottery prize on Friday, which means everybody was buying a ticket. But one of the reasons they hadn't had winners in so long is because they made the odds literally astronomical. You're more likely to get hit by an asteroid, literally more likely to get hit by an asteroid than win the Mega Millions lottery.

Steve: And that's way more likely than you are to have a collision of a 256-bit number.

Leo: Right. Since you love these big numbers, they changed the - I forgot what it is. I think you have to pick 15 numbers between one and 75.

Steve: And get them all?

Leo: Yeah, and then pick - well, yeah. And then pick a 16th that's only between one and 15. Get them all to win. It's the odds are so huge, I'm surprised there was a winner. But apparently somebody did pick it in Georgia. One in 75, 15 times.

Steve: Yow.

Leo: That's a big number.

Steve: That's a big number.

Leo: Anywhere. We have a winner. I don't think it's - somebody does win sometimes, and that's why people do it. On with the show.

Steve: That's 1.6 times 10^88, just for those 15.

Leo: I don't know how they got a win. How did somebody win? I don't even understand how somebody won that.

Steve: Oh, no, wait. One to 75, 15 times. Sorry, 1.3...

Leo: I'm sorry. I have not ever purchased a lottery ticket or a lotto ticket in my life. Somebody has corrected me that there are five, not 15. So one to 75, five times. Is that really right? I thought it was harder than that.

Steve: Wow. That begins to seem reasonable because, Leo, that other number...

Leo: Yeah, if you make it too hard...

Steve: That other number was just wrong.

Leo: Can't make it too hard. Apparently they have two winners. They've announced two winners in the $636 million jackpot. Yeah, it's five numbers. So you have...

Steve: Okay, so now we're at...

Leo: Eight, 14, 17, 20, 39, and a mega ball of seven.

Steve: Okay. So we're at 2.373 billion combinations, one in 2.373 billion.

Leo: It's crazy. It's crazy. Your odds were not good. But, you know, people buy enough tickets, somebody's going to win. On with the show. Sorry.

Steve: So Microsoft made a bunch of news by their joining the FIDO Alliance.

Leo: What's that?

Steve: That's something that I'd never really looked at much. Stina at Yubico always talks about FIDO Alliance, FIDO Alliance, whenever we're together. And it is a big, slow-moving bunch of committees. To give you a sense for it, you have to pay $25,000 a year in dues in order to join. And so it's like, okay. And it's what Google - Google is now involved. And what they have is two different authentication proposals. And so these are big, lumbering committees. And Google has this thing, and I'm sure you're aware of it, that they're using YubiKeys internally, and they've come up with a second-factor authentication technology using YubiKeys. And I think they've got custom plugins for Chrome that work with it and so forth.

So people have asked, where is that relative to SQRL. And I had no idea. So I did a little poking around. I've got links in the show notes if anyone wants to get details. Google has posted - they had a project called, I guess you'd pronounce it "nubby," Gnubby, which was their working name, which is their OAuth technology. And they were calling it "Universal Second Factor," U2F. And there are some similarities to SQRL, that is, the system that I've designed. And, frankly, we solved many problems they haven't solved. There's, like, SQRL is superior. But that doesn't mean it's going to win.

I was initially a little saddened when I looked at what they were doing because it wasn't really good. For example, their system requires the thing that you have to have, like a YubiKey; and it has to have storage, which stores keys for every site. Well, that's one of the coolest things about SQRL is you don't have to have any - it doesn't have any state in it, like per site. It generates it on the fly. And then when they were trying to make these things cheaper - because they're saying, well, this will cost many tens of dollars. And so that's the other problem is that naturally you've got a lot of entrenched biases. Like Google wants to protect their ecosystem. Yubico wants to sell tokens. Everyone has their territory to protect. And this thing, the spec is really overdesigned. I mean, at one point there's like a certificate signing request being sent back and forth. And it's like the kind of thing you do when you register certs with a certificate authority, and you want them to sign your certificate. So one of the problems is it'll never be free because it's tied to a physical token.

Leo: So it couldn't use, like, Google Authenticator.

Steve: Well, for example, you can't use Apple iOS ecosystem because it's also all near field.

Leo: Well, no wonder Microsoft loves it. That explains a lot.

Steve: It's all near...

Leo: It's dead in the water, then. You've got 400 million devices it won't work with.

Steve: Right. And SQRL, of course, will work with everything. So when I realized, well, first I realized there's a whole bunch of things we did right. Also, we've solved the key lifetime, what we call "key lifetime management," where if worst happened, and someone did steal your identity, SQRL gives you control so you can get it back. You can actually get it back from a hacker who has stolen it. And there's nothing like that in FIDO, or in the FIDO proposal.

And then this other thought I had was, okay, not only is SQRL better, I mean, much easier to implement, much simpler, much more clearly, like much clearer about how it works, but it's free, and it'll work on all devices. And it's truly free. You don't have to buy anything. It doesn't need near-field technology, so you can use it on Apple mobile devices. But also it could, if FIDO ended up happening, there's nothing to prevent them from cohabitating. I mean, like if people wanted to use the free one, which is arguably better, we'll have that, too. And it's so simple for a website to implement SQRL, and everyone's writing libraries now for all the different major platforms, that it'll be easy to add it. So it's like, okay. Microsoft and Google and all these other people are happy, welcome to do that. And I imagine that SQRL will not be kept from existing just because there's something else.

Leo: Good. Although, as we know in the tech industry, best technology does not always win. In fact, it often does lose.

Steve: That's true. And if it doesn't take off, well...

Leo: You did your best.

Steve: I'll use it, and other people will use it. And I will have given my expression for here's how you solve the problem the right way. And anyone who looks at it will go, wow, this solves the problem. And, I mean, it'll just be there. It'll be free. So we'll see what happens. I am not discouraged. And lord knows, I mean, I'm writing code. I've got the - I assembled all of the crypto libraries and got them running on my toolchain, which is earlier than everyone else's. So I'm deep into it now and moving ahead. And as are a bunch of other people, who now - because the spec is finalized. I had sort of - had a pro forma finalization. I talked about it last Wednesday. We're done. It has been settled. The semantics and the syntax are finished, and everyone's writing code. So I think we'll have something here in probably a couple weeks.

Leo: So we had talked last week about HealthCare.gov.

Steve: Yes. And what was funny was, completely unsolicited, our wonderful podcast transcriber, Elaine, was listening to the podcast, as she is forced to do whether - actually I think she really enjoys it because she is a geek and, you know, self-ascribed. And so when she sent me back the transcripts, text files in email, like Thursday night late, she wrote: "Steve, I'm too sleepy to go into detail right now, but you might want to relax a bit." Now, okay, remember what I did last week, first question we answered was why haven't we ever talked about the security catastrophe of HealthCare.gov. And I gave everybody who thought that we were afraid to talk about it a nice rant about don't even talk to me about the security, it's just -it has to be the worst disaster anyone's ever imagined.

So Elaine says: "I'm too sleepy to go into detail right now, but you might want to relax a wee bit about the healthcare thing." And by the way, I should tell you I have her permission, as you'll see in a second, to repeat this. She says: "I got it. It's painless. They ask for less information than any utility company, and the website's just to allow you to compare plans available in your area. You pick one, they send your name and address to the insurance company, who sends you an invoice for the first month's premium, and that's it. You're done with the website, and you don't have to give a credit card number or any medical information."

Leo: Oh, that's interesting.

Steve: Yes. Now, we should also remember, though, she's in California.

Leo: Oh, she's not using the federal site.

Steve: Correct.

Leo: She's using the state site.

Steve: She says: "I'm tickled pink because I haven't had insurance for nearly 20 years."

Leo: What? What?

Steve: "And I'm about to need eye surgery."

Leo: Oh, boy. She's been very lucky.

Steve: She said: "Your major disastrous catastrophe" - she says to me, that's what I was calling it - she says: "is my godsend." Although we were, of course, again, we're talking about different implementations. We're talking about HealthCare.gov national,

and she's in California, so I'm presuming she's using whatever the California site is [CoveredCA.com].

So I asked Elaine whether she would mind my sharing her real-world experience with our podcast audience, and she replied: "I don't mind at all. I just tried to tell my farrier" - and I thought, what's a farrier? And I knew that Elaine would not have a typo. And that's somebody...

Leo: "Drill, Ye Tarriers, Drill." It's a horseshoer.

Steve: Yes. So Elaine has a horseshoer, but she has no healthcare.

Leo: Hey, if you have horses, you've got to buy them shoes. Baby needs new shoes.

Steve: Yes. So she tried - she just says: "I just tried to tell my farrier I got health insurance (he got dropped last year and has two people with heart conditions in his family)."

Leo: Oh, that's why he got dropped.

Steve: And he just went nuts on me a la Fox News. So any little bit of truth and balance is a good thing.

Leo: Wait a minute, he's upset because there's national healthcare, but he got canceled last year? I don't understand.

Steve: Yeah, well, people will vote against their interests, unfortunately. We see that all the time. So she says: "It's not spectacular news, like the woman who got a family policy for $3.16 or the person who got billed 13 cents, but it feels very solid and real to me. I'm getting a $580 per month policy for $170 per month, with reasonable deductibles."

Leo: Wow. That's not bad.

Steve: And then she said: "Actually, now I'm thinking it would be GREAT" - in all caps - "if you did mention it. I'm sure lots of geeks like me have small incomes and/or have suffered catastrophe. Remember the listener from New Orleans who wrote after Katrina? You've got listeners in New England who suffered through Sandy. There must have been listeners in Colorado who lost big-time in last summer's flood."

Leo: Yeah, absolutely.

Steve: "Not every working mother is the CEO of Yahoo!, and the recession still exists for lots of us. Healthcare is a definite bright spot." So, and anyway, she said: "Along those

same lines, you can buy refurbished computers at CedarPC.com for $200-300, desktops or laptops. You could do a little 'in case you've been struck by disaster' segment." And so we just did.

Leo: Very good.

Steve: A number of people have asked for GRC's SSL/TLS cipher suite ordering. That is, I talked about it, how when I decided that I no longer had to worry about BEAST, I could now use an ordering of cipher suites that would put Perfect Forward Secrecy at the top of the list. Many people said, ooh, I'd love to see what you chose. Because what I did was I went through, for Windows Server 2008/R2, which is what I'm using, I went through all the available cipher suites very carefully and very deliberately ordered them from most secure to least secure. It's a long - I have it in a directory on GRC. So rather than - I didn't bother making a bit.ly link or anything. But anyone who is interested, I did tweet it, I think.

But it's in the show notes. So go to GRC.com/sn. The show notes are now always the third icon over, and it's there. So anyone who wants to is welcome to grab them: [GRC.com/miscfiles/SChannel_Cipher_Suites.txt]. And actually it's a text file. You just make it one long - I have line breaks in it to make it easier to read. But you take the line breaks out, and you can drop that directly into Windows Server, and it becomes your cipher suite order.

Also, I was preparing actually a list of eBooks for Bob, my friend, whom you remember, Leo, up in Vancouver. And I ran across my directory of Honor Harrington books. And Baen Books is the publisher, but these are all non-DRM, freely downloadable from Baen's site, although their site is a disaster to navigate. And they even have like an ISO image you can download with all of the books on it. But they're hard to find. So I thought, what the heck, I'll just make them available and number them because the sequence, the proper sequence is also not clear, and there's 13 of them in the main sequence of Honor Harrington novels.

So I have them on GRC. I tweeted it late last week. And the server really, I mean, they're large downloads, so I saw the effect at GRC. Many people were saying, oh, thank you, thank you, thank you. So again, the links are in the show notes. But I did create shortcuts. It's bit.ly/HHkindle if you want the mobi format that run on the Kindle machines. Or, again, bit.ly/HHitunes if you want the ePub versions. And so I continue to think that it's one of the best series of military, really beautiful military and sort of political sci-fi intrigue, which I really enjoy.

And last thing up is just a note for people who are interested, that an interesting-looking new series is premiering January 7th on CBS called "Intelligence." I have a YouTube link in the show notes, and I did tweet it [www.youtube.com/watch?v=wMayn0vdCpM]. And it's fabulous-looking. Again, beggars can't be choosers. I'm not suggesting that it's like the end of everything we could ever ask for from sci-fi. We're just not going to get that for free on broadcast TV. But this looks extremely good. The show is called "Intelligence." And apparently a chip is implanted in a guy that wires him into the Internet, so he's able to access basically...

Leo: I'll take that. Is that available now?

Steve: Oh, well, it's very much - this is the technology in Peter Hamilton's Commonwealth universe, where you just have access to your own data store. But also Marg Helgenberger - is that how you pronounce the name? She's in there. She used to be on "CSI Vegas" for a while. And anyway, extremely good. And my server is definitely showing the effect right now, Leo, of people grabbing the Honor Harrington novels. So I'm glad people like it. Anyway, definitely very cool.

Leo: Very good.

Steve: And I have a nice note from a Microsoft certified systems engineer that I ran across in my mailbag when I was pulling Q&A for the show. He said: "Steve, I'm a Microsoft certified systems engineer. Today I restarted an older laptop of mine, a Dell Latitude D820 that had been running Windows XP without problems for years. This morning it was locked up. So I thought to myself, okay, a reboot is required. But then the laptop gave me the dreaded Blue Screen of Death three times, even after the last known good configuration was selected at reboot.

"Oh, dear," he writes. "It was a half a terabyte hard drive. And because this laptop does a lot of stuff in the background on my home network, I dreaded the thought of having to replace the laptop or hard drive and getting everything reinstalled again. So I decided to try the SpinRite disk I bought from GRC a couple of months ago. After two hours it reported that Sector 125 were unrecoverable." And he says 1-2-5, so maybe that's sectors 1, 2, and 5, or I'm not sure because he says "were," as opposed to Sector 125. He says: "I was crushed. I thought, my life is chaos now. I crossed my fingers and prayed.

"When SpinRite finished, and I checked again, SpinRite reported that there were no unrecoverable errors. What? How can that be, I asked myself. But sure enough, on further inspection, SpinRite reported that there were no abnormal sectors on the hard drive. So there I sat, after a total of three hours, hoping that SpinRite would salvage more than my day. Once again I crossed my fingers and removed the SpinRite boot disk and restarted the laptop. Steve, thanks for such a great hard drive maintenance and recovery software. It took only three hours or so to recover this hard drive, and SpinRite saved me many more hours of reinstallation and possibly relicensing software. From this day forward, SpinRite will be a part of my normal backup routine." So just another happy SpinRite customer.

Leo: They're all happy-go-lucky SpinRite customers here in SpinRite world. So are you ready? We have questions.

Steve: We do, yes, absolutely.

Leo: Wow. We've got 15 minutes.

Steve: We'll get through some of them.

Leo: I am perfectly happy to do as many as you choose at the time of your

choosing. Question 1 from Richard, he asks about running SpinRite in a virtual machine. That's interesting. My TiVo recently stopped working. I put the TiVo's 320GB hard drive into my PC so SpinRite could fix it. It ran on a Level 2 scan twice. SpinRite found and recovered data from bad sectors on both passes. The number of bad sectors found on the second pass was fewer, but they still were present. So I decided to run a deeper Level 4 scan to really root out the problems.

After I started the scan, SpinRite reported it was estimating 30 hours to complete the scan. I let it go overnight, but since I needed my computer back for the work week, I had to stop the Level 4 scan at about 50%. I then got to wondering, can I run SpinRite in a virtual machine so that I could use my PC? Because SpinRite works in a DOS box, or a FreeDOS box. So he wants to know, could he do that and still get to use Windows? He says he found your tweet from January with a link to step-by-step instructions on how to use VirtualBox - the free VM - for just such a task.

I got VirtualBox installed and configured and started a Level 4 scan on the TiVo hard drive. To my amazement, the Level 4 scan was flying. Then, after the usual 60-second sampling, the estimated time populated. It said it would only need about four hours. I verified that Level 4 was in fact running. It was. So that long hoo-round is all to the point of saying: Does running SpinRite in a virtual machine cause it to run better than booting it from a CD? This is the same hardware. Is SpinRite really working while running in the VM? Maybe it was just making up those numbers. Seven-fold increase in speed, what's the story?

Steve: Okay. So I wanted to - this is a great case in point for what essentially is a tip for anyone who is wanting to run SpinRite at maximum speed on a motherboard BIOS that won't run at maximum speed. And that is SpinRite 6 today. What happened is that Richard has a BIOS that is not doing Ultra DMA by itself. And SpinRite 6 famously still runs through the BIOS. Many people have motherboards with BIOSes that natively support Ultra DMA transfers, in which case SpinRite gets the advantage of that and runs at full speed. The VirtualBox virtual machine that Richard was using, and VirtualBox, has a state-of-the-art virtual BIOS. So you can run SpinRite in a VirtualBox VM, and it will be guaranteed to run at full SpinRite 6 speed, which, as you can see, like in this case, was seven times faster on that particular motherboard.

So this was an interaction between SpinRite 6's use of the BIOS and the fact that that motherboard wasn't doing Ultra DMA. Many other motherboards do Ultra DMA, in which case SpinRite always runs at that speed without putting it in a VM. And of course the reason everyone's excited about SpinRite 6.1 is it will always get that speed because I will no longer be using the BIOS on any platforms. And in fact we'll get a lot more speed because I'll be talking natively to the drives and using a 32MB buffer. I don't remember now the benchmarks that we were making back when I was working on 6.1 before suspending that to get SQRL finished. But I remember that - what I remember was a 4TB drive we would then be able to do overnight. So you'd be able to run SpinRite on a full 4TB, like, overnight.

SpinRite 6 won't run that fast because it doesn't use a large enough buffer to do that. But it can run at maximum speed in a VirtualBox VM. So people sometimes say, hey, Steve, I'm creating a machine that I want to use just to run SpinRite on. And so absolutely setting it up with VirtualBox and seeing whether it runs faster in VirtualBox than it does natively is something worthwhile because VirtualBox has a very good virtual BIOS that it brings along.

Leo: That's cool

Steve: Isn't that neat? Yeah.

Leo: But so the BIOS will work on a hardware-connected drive. It's not working on the virtual machine hard drive. The VM has its own hard drive that's a pseudo hard drive.

Steve: Actually, one of the things that VirtualBox allows you to do is to get direct access to a drive.

Leo: Because you can't. Ah, that's cool.

Steve: Yes. So you could not get direct access to the system drive because that's the one that it's running on itself. But you can get physical hard - I mean, even VMware will allow you to do that if you're careful. So the various virtual technologies give you direct access.

Leo: Raw mode, basically.

Steve: Yes, exactly. And then so when SpinRite makes its calls to what it thinks is the BIOS, it's of course the virtual machine BIOS. And the virtual machine BIOS is a very nicely, recently written BIOS that then gives SpinRite Ultra DMA access to that physical hard drive. So you're really running SpinRite raw, and you're really running a state-of-the-art BIOS. So it's a really great solution.

Leo: Cool. Stephen Adams, Aurora, Illinois highlights probability versus possibility. This has something to do with the Mega Millions jackpot, I suspect.

Steve: Yes, it does.

Leo: I fully understand the concept of extremely low probability events and the fact that 256 bits provides extreme protection against collision - we're talking about BitTorrent Sync, of course, once again. But it does not prevent collision, as you have stated. And I agree.

Steve: Well, yes. So BitTorrent Sync, Bitcoin, and SQRL, for example...

Leo: All use the same thing.

Steve: ...all use 256 bits. Yes, I'm sorry, go ahead.

Leo: That said, it is a fallacy to state that collisions cannot happen, as you and Leo did. I did not say that. I'm going to defend myself. I know the difference between probability and impossibility.

Steve: We were actually laughing.

Leo: I think we were choking.

Steve: We were, yes, we were saying - we were laughing, saying it will not, it cannot happen. But...

Leo: But it could.

Steve: Yes.

Leo: Extremely low probability, he points out, does not equate with impossibility, Steve. You should know that. No, I added that part. While we calculate the mean time to solve something by brute force, it is entirely possible that the very first attempt succeeds. Look, three people won the Mega Millions. The odds against this are long, assuming a large enough range to guess from, but they are non-zero. Because they are non-zero, it's entirely possible, though highly unlikely, that two people will have the same key.

And, frankly, all it takes is an error in a pseudorandom number generator to create a collision. One mistake by a programmer, and all of a sudden we have collisions and chaos. Since we can't review every single PRNG that's used by every single software that generates these random strings - he says "stings" - we are subject to the skills of the least capable or least careful coder. And we see where this has taken us in the past. It's obvious this guy has no sense of humor. It's pretty clear from Security Now! that such errors are all too common.

To summarize, a pure brute-force attack is unlikely to succeed, but it is - and I highlight this in bold - possible that it will. A collision for a properly created pseudorandom number generator is unlikely, but possible. An error by a programmer is very possible and thus could easily create such collisions. Fundamentally, the point is low probability does not equal impossibility. And when human actions are involved, counting on what amounts to security by obscurity is a bad idea. Signed, Guy With Little Sense of Humor, Stephen Adams in Aurora, Illinois.

Steve: Okay. So, yes, obviously we all know what Stephen said.

Leo: Yes.

Steve: The reason I selected this was that, first of all, he's very correct about the extreme dependence we have on the quality of pseudorandom numbers. We've talked,

for example, about the surprising discovery by the EFF's Observatory that completely unrelated servers were using the same private keys without knowing it because the technology that they have is that they generate the private key themselves, and then they send the public key off to the Certificate Authority to have that signed. Basically they generate a certificate which asserts their identity with their public key. The Certificate Authority signs it and returns it. So what this says is that multiple servers around the world generated the same, chose for themselves the same private key. So the lesson to we who have a stake in using 256-bit strings for our identity, for our bitcoin wallet, for our private BitTorrent network or bit sync, is we really, really do.

And so this is a good point Stephen makes. We really, really do need high-quality random numbers. Probably what happened in the cases of those surprising, I mean, these are probably, what, 128-bit, well, no, I'm sorry, they're RSA keys, so they're probably 1024-bit RSA keys. So these were probably servers that were powered up or booted and hadn't had a chance yet to acquire much entropy from the universe, from the random timings of packets, from noise that they had access to, whatever they were doing to mature their entropy pool.

We talked about that recently. They just didn't have much time. They probably got booted, and somebody said, "Make me a key." And so they did the best job they could with what low relative level of entropy they had. And sure enough, multiple servers with the same OSes, running the same key generator, running the same random number generator, started off having an entropy collision and gave the same key.

So it's definitely important. I mean, for example, we see this in TrueCrypt, where TrueCrypt makes you move your mouse around a lot. That's just to enhance the entropy so you're not all depending upon it from a single source. And listeners will remember how I was talking about, in SQRL, in a mobile phone setting, I'll have you wave your phone around in the air while we're streaming video from the lens into a hash to just, I mean, to create a fabulously random pool which we mix with what entropy we get from the platform we're running on in order to get a really, really random identity for users of SQRL. So there are certainly ways to do this. But turning a computer on and immediately having it generate a key with a limited entropy pool is obviously not - is going to be prone to collision. And so he's right that it is something that we need to look at.

And the second thing is I just want to mention that we're all - there's a thing that makes some people uncomfortable, when they say, yeah, but my BitTorrent Sync could collide with somebody else. In which case then I'd be, like, I'd have the key to their network. And my bitcoin wallet could collide, in which case I'd have somebody else's wallet. And so the reason that this is like the new model, this is the model we're heading towards, and SQRL is the same way, is that there is no central authority. There is no one you ask: Does anyone else have this key? You don't ask: Does somebody else have my bitcoin wallet key? You don't ask: Does somebody else have my BitTorrent Sync key? I mean, there is no one to ask. It's a decentralized model.

When you sign up for email, you put your email, and many times people have signed up, like for Yahoo! email, and that's why you see names like SteveGibson327. It's because Steve Gibson nothing through 326 were already taken. So when you have a central authority, you're able to ask it: Is this account name available? And then it says: "Oh, no, you've got to choose something else." I mean, same thing for when people sign up for Twitter. It's like, oh, darn, I can't use that, I can't use that, I can't use that, I can't use that. Oh, finally, here's one I can use.

But in this next-generation, no third-party, decentralized model, what we rely on is just the vastness of the keyspace and the quality of the entropy that we're able to generate.

And so everyone doing this really needs to pay attention to entropy quality. And really, operating systems should flatly refuse to generate entropy until they've accumulated enough to be confident that they have it. And clearly that was not done in the past. It's something that we're seeing now.

Leo: So just as an example, if you have a Schlage lock on your door, there are - it's only - it's one in a million that somebody has a similar key.

Steve: Oh, I think it's lower than that, Leo.

Leo: It might even be lower than that. There's some difference of opinion on how...

Steve: Yeah, you get the birthday scenario. If you have a bunch of people all trying a given lock, the chances are rather high that you'll just have a collision.

Leo: So you should just go around trying your key and...

Steve: Yeah, because, I mean, look. There's a tumbler with six pins, and they don't have that many positions each.

Leo: Right. So, yeah, exactly. Probably in the hundreds of thousands, at most.

Steve: Oh, I'm sorry, five pins. Five pins...

Leo: Some have five; some have six. Yeah. Kwikset has five; Schlages have six, I think. Actually both Kwikset and Schlage have five and six pin. So...

Steve: And they can't have that many positions because you need to be able to tolerate key wear over time and not have them constantly going out of alignment. And they don't. They're pretty tolerant.

Leo: That's why it's so easy to pick a lock.

Steve: Yeah.

Leo: Moving along to Question 3, Marcus in Corona, California discusses the power of the pointer. He's apparently been taking a course in computer programming. But, he says, I'm somewhat dyslexic and mostly an auditory learner. My instructor does almost no lecturing and expects us to read the book and come to class with that week's subject fully understood. Well, what does the instructor do, then?

Steve: Uh-huh.

Leo: But I was having a lot of trouble with pointers, a lot of trouble. Yeah, this is one of the first subjects you come up against in programming that could be a little confusing. I have been listening to Security Now! for over a year, and I always listen to the current week's episode and a few "old" ones in between. Well, almost right on cue for my exam I happened to listen to Episode 237 - like four years ago - which is where you explained the fundamentals of programming. Long story short, I passed the test, thanks to the podcast. I just wanted to let you know that no matter how old Security Now! gets, its content will always be relevant. And that goes for you and Leo, too! Just kidding. Hey, that's awesome. He figured out pointers because of you.

Steve: So I just - this was a perfect opportunity for me to bow at the altar of the pointer.

Leo: It's a magical thing; isn't it?

Steve: Oh, my goodness. It is. I once said - I was quoted somewhere saying that God is six levels of indirection.

Leo: I don't know what it means, but I like it. A pointer is one level of indirection; right?

Steve: Correct. The idea is, and the thing that's confusing to new programmers, is does a variable contain the thing or a pointer to the thing? And there's nothing to prevent it being a pointer to a pointer to the thing, or a pointer to a pointer to a pointer to the thing. And in my programming experience, whenever I have carefully set up structures that have meaning, and the structures contain pointers to other things, and then I have a pointer to that structure, it's like, I mean, I've only ever gone, like, three levels of indirection. That's why I say God is six. Every time I code this way, I find myself - I just get a chill. It's just like, wow. It is such a powerful way of working. And I think it's powerful because pointers are the way we humans think. We use nouns to represent something. That's a pointer to an object is a noun, a grammatical pointer.

Leo: A Zen Buddhist would say the finger pointing at the moon is not the moon.

Steve: Correct. Anyway, so I just - I loved his note that he understood pointers, or thanks to the podcast. And I probably maybe swooned over them back in Episode 237, I don't remember.

Leo: Oh, yeah. Oh, yeah.

Steve: But I just posted something this morning in the SQRL development newsgroup that the next thing I will do is I'm going to write a library, now that I've got all the crypto code assembled and linked, I'll write a library for SQRL's low-level plumbing, where I

pass a structure containing pointers to strings to a function. So I'll give the function a pointer to the structure, and that will contain pointers to strings. And, I mean, that's the way I'm going to set it up. And it's like, it's the way you do things now. And just it makes for such an elegant description of real-world stuff. So I bow to the altar of pointers.

Leo: First experience, those PEEKs and POKEs on the Atari 800. And then pointers. And then the next thing you've got to get is recursion. That's even a little bit harder.

Steve: Ooh, yes.

Leo: And then the thing I'm still stuck on is lambdas or closures, which are anonymous functions. And that I just don't understand at all.

Steve: And the one thing...

Leo: If I'd taken Calculus 3, maybe I would.

Steve: The other thing that object-oriented programming introduces - because basically what I was just describing was technically object-oriented programming.

Leo: That's right. That's right, yeah.

Steve: But you're also - it gets kind of freaky when your structures can contain pointers to functions, not just pointers to, like, variables or strings. But you can actually have a pointer to a function, and that can change depending upon, like, the type of value that you're...

Leo: Now you're getting into lambdas.

Steve: It gets pretty heavy.

Leo: I don't want to go to lambdas, thank you.

Steve: You can get lost.

Leo: That's a nice - thank you for that. That's a nice email. Here's a gripe, though, a complaint from GeekWrench in Chino Hills, California: I use NoScript religiously on my laptop. I guess he prays every time he uses it. I don't know. This means that when I watch Security Now! on my laptop, most of the JavaScript is blocked. It takes time for my old machine to load the show, so I do other things. If I so happen to make the mistake of allowing scripts on another page, it dumps the whole show, and

I have to wait for it to reload all over again. Last week I got frustrated and never bothered to finish. This is our fault? Is there a way to allow NoScript in a single tab? Oh, that's an interesting idea. Or do I just need to continue mentally flogging Leo? Now, you see why I have a crappy job? Because some guy has a crappy browser in an old machine, and it dumps the - and he blames me.

Steve: Okay. So, okay. Newsflash, GeekWrench.

Leo: He's blaming me.

Steve: You right-click on the NoScript icon. And where it says "Temporarily trust TWiT.tv," right below it it says "Always trust TWiT.tv."

Leo: Always. Always trust us. We're safe.

Steve: And that's the key.

Leo: He doesn't want to do that, apparently.

Steve: No, I just think he hasn't seen that, or hasn't noticed it. And so what happens is NoScript then remembers that you trust TWiT.tv. And so, for example, my NoScript never bothers me because it knows I trust Amazon. I trust Google. I trust the various sites that require scripting that I'm a repeat visitor to. I don't trust anything by default, but I've trained NoScript over time. And sometimes I'll be at a site like GitHub, for example, and I'll think, eh, do I - am I - is this worth trusting? Because I'm adding a permanent entry in a database somewhere inside of NoScript to say, oh, yeah, let's always trust GitHub. And I can go, eh, no, I don't - I'm happy to trust it on the fly rather than permanently. But that's a huge feature of NoScript that apparently just missed your attention. And so I wanted to call your attention to it. Trust TWiT.tv, and you're good to go. No more problems.

Leo: Somebody said TNOEL, Trust No One Except Leo. Hey, this is going to be a holiday gift I just got from SCOTTEVEST. You might be interested in these blackout pockets. You can put - it has RFID, or actually it's basically...

Steve: Ooh, drop your cell phone in.

Leo: ...a Faraday cage in a bag.

Steve: Very cool.

Leo: Yeah. So it blocks cell phone signals. It blocks RFID. So if you have a passport or a credit card, you can't be skimmed or hacked. And so you just put this - it has Velcro on the back, so you just stick it into your SCOTTEVEST. And they have three levels of protection.

Steve: There was something I saw recently, gosh, I think it was a video, or maybe it was a TV show or something, where everyone entering a conference room had to drop their phones in a box...

Leo: Yeah, in government they do that.

Steve: And then the lid was closed. And you were offline.

Leo: Snowden did that, too. When Greenwald went to talk to Snowden, he said, "Put your cell phone in the fridge." Same thing.

Steve: Okay. You're right. We did talk about this, yes. Oh, and people were saying, okay, put it in the microwave, but do not turn the microwave on.

Leo: Yeah, because a microwave is a Faraday cage. Otherwise the high-frequency waves would fry your brain when you peered in to see if your Hot Pocket was done.

Ari in South Africa asks a fundamental NAT question. Network attached, no, network, no, network - never mind. Addressable something thingamajig. Hey, Steve. Wait a minute. Now I'm going to have to think of it. Network Address Translation. Thank you. Please could you clarify some basic Network Address Translation concepts pertaining to what a NAT router is expected to do with IP addresses of packets it lets through outbound. See "device bypasses NAT router" post in grc.security. Thanks, and best regards, Ari.

Steve: Okay. So we haven't talked about NAT routing technology for a long time. And I don't want to bore our old-timers, who are, like, rolling their eyes because it's like, oh, come on, talk about something we don't know. But I know that we're constantly getting new people on the podcast also. So I won't spend too much time on this.

But what NAT routers do with the IP addresses of packets they let through outbound is remember them. And this is the whole beauty of why a NAT router is such good security. Normally there's just junk arriving at everyone's IP address all the time. I've coined the term IBR, Internet Background Radiation. That's what it is. It's Code Red bots, Nimda bots, and nonsense that are living on forever and will never go away, scanning the Internet. There are spambots and just every kind of thing you could think of, just checking ports on your IP to see if maybe you've got that port open.

The idea is that they're all coming from random IP addresses that your router knows nothing about. So when they arrive at the router, it looks in a table to see whether it's expecting anything from that IP address on that port. Or, like, from that IP address, from that port, into your specific port that it's addressed to. And if not, if there isn't a match, it

just drops the packet. The packet just dies. I mean, it's just data arriving. It's not like they're pointy, and they're going to, like, pierce the router or somehow work their way in. It just ignores it.

But if you send a packet from inside your network out through the router, the router takes note of the destination IP and port and the source port that it's emitting the packet from, from its IP. And off the packet goes out onto the Internet. Then, when that destination sends the packet back, it'll match up with an entry. It's left an entry in this table. So this packet comes back amid all of the other noise that's trying to get in. But that particular packet came from an IP address and a port, destined for a port on the router that the router expected. So that one gets let back through. And that's how NAT works. It's just a fabulous fundamental firewalling technology.

Leo: Basically it will continue a conversation started from within the network and ignore any attempts to start a conversation from without the network unless otherwise...

Steve: You can kind of think of it like a one-way valve that allows data and transactions outwards, but blocks anything unexpected, unsolicited, as I use the phrase all over the place, coming back in.

Leo: Do you have any one in particular we want to do? Because we're kind of out of time here.

Steve: I think we've got a good podcast behind us.

Leo: We covered some good stuff, yeah.

Steve: Yes, very much.

Leo: And we'll leave some more questions for later. You can always ask Steve questions at his website, GRC.com/feedback. Do not email him. He doesn't even - you don't...

Steve: I don't have email.

Leo: He has no email.

Steve: I don't have it.

Leo: He's a black hole on the Internet. His pointer points to nothing.

Steve: So next week we've got a fabulous, I think everyone is going to get a big kick out

of it, holiday special. One of the first of many time capsules. Let us know what you think because I think people are going to enjoy them a lot. That's what'll be airing on the 25th, on Christmas, will be a blast from the past.

Leo: Yay. Can't wait. That's going to be fun. You can tweet at him. He's @SGgrc on the Twitter. And he may tweet back at you.

Steve: Yes. And everyone knows I read them and I answer them, so...

Leo: You're going to be sorry. He also puts 16Kb audio versions of this show, along with those great transcriptions written by Elaine Farris, on his website, GRC.com. Stop by there to get those and, of course, SpinRite, the world's best hard drive maintenance and recovery utility. If you have hard drives, you ought to have SpinRite. You can also get a lot of free stuff there, like Perfect Paper Passwords and more. GRC.com. We do this show, well, this is - we'll be doing it one more time at 11:00 a.m. Pacific, 2:00 p.m. Eastern time.

Steve: Yes.

Leo: 18:00, I'm sorry, 19:00 UTC on Wednesdays. We're moving to Tuesday at 1:00 p.m. Pacific, 4:00 p.m. Eastern time, 21:00 UTC.

Steve: Yes, starting January 7th, I think it is.

Leo: Yeah, January 7th is the first Tuesday.

Steve: And that's the premiere date for...

Leo: "Intelligence."

Steve: ..."Intelligence" on CBS.

Leo: Intelligence will appear here and on CBS in an amazing coincidental dual sighting.

[Talking simultaneously]

Leo: A collision. What else? If you can't watch live, though, you know we make on-demand audio and video available. He has it in lower quality audio, 16Kb audio for the bandwidth-impaired. We have high-quality audio, even video to watch our smiling faces at TWiT.tv/sn for Security Now!. You can also subscribe in your favorite

podcatcher. And that way you'll get every week. And you probably want to watch the whole set. Collect all the episodes.

Steve: And, yeah. I will just tell everyone once again, because I'm getting a surprising number of questions about this, the show notes are at GRC.com/sn, and it's the third icon over. So click on the third icon. You get the PDF that Leo has been reading and following along with me, and that I've been reading, of the top of the show stuff, with all the links in it that I refer to. So that's where they are.

Leo: Thank you for doing that. And thank you for watching.

Steve: Okay, my friend. And everybody will see me in-studio with Leo all day Christmas Eve day on December 31st.

Leo: New Year's Eve day, yes.

Steve: I will be up - what am I saying. Yes, New Year's Eve day.

Leo: Yes. You scared me for a moment.

Steve: Puttering around.

Leo: Yeah, that'll be fun. Looking forward to it. Thank you, Steve. It's always great to see you. Have a great Christmas, and we'll see you in the new year...

Steve: Thanks, Leo.

Leo: ...on Security Now!.