Transcript of Episode #494

Listener Feedback #206

Description: Leo and I discuss the week's major security events and discuss questions and comments from listeners of previous episodes. We 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-494.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-494-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Boy, Microsoft pushed out a lot of patches this Patch Tuesday. He'll talk about that. And we'll answer your questions. Security Now! is next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 494, recorded February 10th, 2015: Your questions, Steve's answers, #206.

It's time for Security Now!, the show that protects you and your loved ones and your security and your privacy online with the man in charge of all of that, Steven "Tiberius" Gibson, the Gibson Research Corporation, GRC.com. Hello, Steven.

Steve Gibson: Yo, my friend. Great to be with you again, as always.

Leo: Hey, we've got a Q&A episode.

Steve: Yup, we do. Things have been relatively quiet. We've got a bunch of interesting news. We had a mega Patch Tuesday today, just landed, with 55 vulnerabilities. And I love one of the phrases that Microsoft posted, so I'll share that. I sort of - the Adobe patches went by, but I just thought I'd give them a nod. The U.S. government is today announcing, I don't know if they have yet, but I know that it was scheduled for today, I got the news beforehand, the Cyber Threat Integration Center, with an awkward acronym, unfortunately, because we'll probably be using it a lot. I thought we'd give an update on the Anthem breach. Some news about Chrome's move to HTTP/2. And then of course we have car hacking, we have TV eavesdropping, and the news of GPG getting a new infusion of support and more.

Leo: Oh, good. Oh, good.

Steve: So lots of fun news. And in the show notes is the image of the week, courtesy of our friend Simon Zerafa. I just got a big kick out of this. It was something like - there was a caption like "Passive aggressive WiFi station ID" or something. The two items at the bottom, I just - I got a hoot out of it. So anyone who...

Leo: So actually, people have started to do this with their SSIDs on their WiFi networks, putting, you know, started with "Don't steal my WiFi," and then we have somebody nearby who's an FBI surveillance van. But now it's like there's so many WiFi signals in apartment buildings, you know, it's so cluttered, that neighbors are actually communicating with one another. So the first access point is "You're music is annoying," but they spell your, instead of Y-O-U-R, which would be correct, Y-O-U-'-R-E, a common grammatical error. And so the response: "Your grammar is more annoying." Both of these WiFi access point names. That's hysterical. That's hysterical.

Steve: Yeah. Over by Starbucks there's someone who says something like, "The pierced chick upstairs," is like...

Leo: Oh, wow.

Steve: It's like, okay.

Leo: Hey, there's an image. Wow.

Steve: I don't know where she is, but she's pierced, apparently.

Leo: I'm sticking with my dead rock stars methodology.

Steve: My buddy has his named something like NORAD Southwest Listening Station. Where it's like, okay.

Leo: I love it. Thank you, Simon.

Steve: Oh, yeah, those are fun. So we do, we're on Patch Tuesday here. Microsoft released nine bundles, which contained at least, it's kind of hard to count them at this stage, 55 distinct security vulnerability fixes. Three of the patches are considered critical. And I love the way they described it, MS15-009. It says - you're just sort of reading along - "This security update resolves one publicly disclosed and 40 privately" - it's like, what? Forty privately reported vulnerabilities in Internet Explorer.

Leo: Wow. Holy cow.

Steve: So, baby. You know, and this does sort of - we've noted that there's a - it's not like it's a smooth flow. For whatever reason, there are dead months, and then there are catch-up months. And this is clearly one of those. There's a vulnerability in the Windows kernel mode driver that involves the rendering of embedded TrueType fonts. There's a TIFF vulnerability. There's problems in Office, in group policy, in the virtual machine manager. And something called the Microsoft Graphics Component that I haven't heard referred to before in any security update. So anyway, nothing, like, clearly crucial, but certainly worth doing, as everyone will, I'm sure. Oh, and I did see some notes online that in some cases this can require multiple reboots of a Windows machine. Mine, because I fired up my Windows 7 box, which is where I run Skype from just for this, got the updates, updated everything, and it was just a single restart. But apparently in some cases it can take more than one. So definitely good to do.

Back in the old days we were talking about Adobe all the time. They've had a rough couple weeks. They released an emergency out-of-cycle update to Flash to patch a zero-day flaw that was affecting Windows, OS X on Macs, and Linux, allowing remote code execution on all of those platforms, at least where they were being employed, and noted at the time of the release that there was another known zero-day exploit that that patch didn't fix, but they'd be getting to it soon. So less than a week later they released a second zero-day update to Flash. So we have both of those now behind us.

I've seen Firefox has - like I did the update, and then it said, no, you're still not there. It's like, oh, okay. And I did it again, and then I was able to run something I was using Flash for a couple days ago. So, and of course Chrome works to keep itself updated within its own internal update mechanism. And so this has been pushed out everywhere. So it's a little bit of old news, but we hadn't mentioned it before, so I wanted to. And I don't know how we're going to pronounce this: CTIIC.

Leo: Yeah. Mike Elgan had a little trouble with that.

Steve: It's like, what? Okay.

Leo: Not a great acronym. As Mike said, it's an abbreviation, not an acronym.

Steve: Right. So it's the Cyber Threat Intelligence Integration Center, whose name tells you what it does: cyber threat intelligence integration. So this is just coming. It's being rolled out today. And in sort of the preannouncement it was noted that the NSA, the DHS, the FBI, and the CIA, all of our three-letter initial organizations, each currently have their own cybersecurity groups. And they've not been communicating in any formal fashion with each other. I'm sure they share notes on an ad hoc basis. But similarly to the creation of the National Counterterrorism Center, which was established after the 9/11 attacks, this is intended to explicitly create a means for the individual cybersecurity groups to pool all their resources and pool their knowledge and hopefully make us as a whole, the U.S. as a whole, more responsive and aware of what's going on in cyberthreats. And this is clearly something that we need to take a look at in the wake of the Sony breach and then the Anthem problems and the problems with Target and Home Depot and so forth.

Leo: Is it going to replace the DHS CERT, the Computer Emergency Response Team and these other kind of...

Steve: I don't think so. I think my feeling is this is more secret. So, for example, DHS CERT is our public-facing announcing group where we have one place to notify everybody of threats publicly. I think this is going to be under wraps. This is, you know, so I don't think we'll see much of what's actually going on. The idea is that behind the scenes, much as they've had their own, the individual organizations have had their own systems, the idea is for them to coordinate and share information. I think, you know, stovepiping is the term that's often used for saying, okay, they're just not communicating. And they're recognizing that they'd be doing a better job overall if they were, if there was some means for them to share things.

Leo: The consensus in the chatroom is it'll be called C-TIIC.

Steve: Okay. I can go with that.

Leo: Cyber Threat Intelligence Integration Center, CTIIC, C-TIIC. I think it's fine. The truth is you probably won't hear it a lot because, as you say, this is more of an internal and private secretive - and you need to do both; right? You need a public-facing something like CERT, where people learn about security. NSA [crosstalk] right.

Steve: Like OpenSSL flaws can be published and discussed.

Leo: But you also need an internal security force, cybersecurity force. And I gather this is more of that.

Steve: Yeah, and for example, one presumes that ongoing attacks, you know, these groups may be individually monitoring ongoing attacks, but they have different resources that are giving them different visibility. And so we're not aware at the public level of, like, what attack is actually happening right now on some government servers somewhere. Presumably some group is. But by coordinating that information from their various sources, they can do a better job. So, yeah, I don't think much of this will come to the surface. But we can all sleep easier now.

Leo: Or not.

Steve: Or not.

Leo: Or not.

Steve: Speaking of not sleeping easy, we've not mentioned the Anthem breach, which is

chilling because of not only the size, but the scope of what was leaked. The good news is that Anthem, from all reports, did respond very well. I mean, maybe this is the - while it's bad that we've had a breach of this scope and magnitude, and I'll discuss that in a second, they themselves detected it, rather than, as normally happens, some third party sees, like, credentials appearing on the Internet and says, eh, by the way, I mean, normally Home Depot and Target found out about these things when it became clear that the common thread among fraud was that everybody had shopped recently at Home Depot.

Well, so this is way better than that. This is Anthem's own internal security monitoring found the problem. The bad news is that apparently the first malicious access to Anthem's internal insurance subscriber database was December 10th, and they first became aware of the suspicious activity on January 27th. So the bad guys were in there, apparently exfiltrating data, for some length of time. And the data that was exfiltrated is of course the real crux of this problem because...

Leo: Well, and the thing that bothers me is it wasn't encrypted at all. Right?

Steve: Right.

Leo: What?

Steve: Right.

Leo: That's terrible.

Steve: By their own admission, they said that it was not encrypted in its database. They also did a little sort of a mealy-mouthing, saying, well, additional encryption would not have thwarted the attack because an administrator's credentials were compromised, and security protocols were bypassed. So they're saying, well, okay, but it wouldn't have mattered. At the same time it's like, well, maybe in this case. But that's not an excuse for not encrypting your database because a simpler form of attack might have just been able to exfiltrate the database, in which case they would have gotten nothing. So clearly, clearly this demonstrates that they're still not behaving. And remember, this is not Anthem's first problem. They had problems a couple years ago that we talked about on the podcast, a much smaller security problem, but it came to light then they were not encrypting. And now we know they're still not. So that's certainly a problem.

Leo: Didn't learn their lesson.

Steve: Names, dates of birth, member IDs, Social Security numbers, residential addresses, phone numbers, email addresses, and in some cases employment information, including members' incomes. So what we're looking at there is the mother lode. You couldn't ask for more for identity fraud. And that's of course the problem is identity threat, people creating new accounts under fraudulent credentials and Social Security numbers and residential addresses. This essentially is everything you need in order to impersonate somebody and get credit under their name. So and of course, yes,

they're - and 80 million subscribers, if I didn't mention that number before, 80 million. Anthem, of course, is big in California. They have 37 million subscribers in California alone. And this is across their various properties: Anthem Blue Cross, Blue Cross and Blue Shield, I guess Blue Cross and Blue Shield of Georgia, which is a separate group, Empire Blue Cross and Blue Shield, Amerigroup, Caremore, Unicare, Healthlink, and DeCare.

Leo: I'd be so furious if I were a customer.

Steve: Yeah.

Leo: Because the real problem is you cannot change your Social Security number.

Steve: Right.

Leo: It's not like a credit card number, where you could say, oh, federal government, give me a new one. So you have this, the rest of your life now you have this still hanging over your head. There is no remediation.

Steve: Yes. And with a credit card, that's all that's been compromised. And so while, yes, you're then subject to credit card fraud, of which you're indemnified from, we've all heard the horror stories of what happens with identity theft. I mean, people's lives are turned upside down. And it's very difficult to recover from. I mean, as you said, Leo, it is really bad. Now, Anthem...

Leo: We have to fix this Social Security issue because, I mean, first of all, the Social Security folks say don't use your Social Security number as an identification.

Steve: As identification.

Leo: And yet everybody has it. If you apply for credit, you have to give them your Social Security number.

Steve: That's what we're being asked for, yes.

Leo: Which means that you have no recourse. If you want healthcare, if you want a car loan, if you want a credit card, you're going to give your Social Security number to people who do a crap job of protecting it.

Steve: Right. And oftentimes it's - I know in some cases - I'm trying to think if it was something I did with Verizon a while ago. I have my cell phone account through GRC, so we use our corporate tax ID. But had it not been that, they wanted my Social Security number. I mean, that's my tax identification number.

Leo: So I think they need to - we need two numbers. It needs to be like private key crypto. You need a public number and a private number or something.

Steve: Sort of like a one-way function so you cannot - sort of like a hash of your Social Security number so that somebody could verify that you know your Social Security number. If they know it, you can prove yourself that way by just hashing it and then saying, okay, here's the hash, but not be able to go the other direction.

Leo: If you can prove - I'm reading the documentation on the SSA.gov website. If you want to change your Social Security number, you can, in some very specific cases. But one of them is, "If you've been a victim of identity theft, and you continue to be disadvantaged by using the original number." So but you have - you can't just change it because you want to. You have to prove that you've - it has to be all after the...

Steve: After the fact.

Leo: ...horse has left the barn.

Steve: It's sad, too, because I memorized mine when I was 16 or something, you know, when I got it. And I love my number. It's like, it's been in my head ever since. And it would be a shame to have to change it. Anthem did bring in an outside security firm, Mandiant, that is a well-known sort of inspector of these things. And they got thumbs-up from the guy who was interviewed by several third parties, saying, well, what do you think of the job they did? And this guy Dave Damato is the managing director of Mandiant, who said that immediately after Anthem noticed the incident, they reset some passwords and performed a series of actions to remove the attacker from the environment. Any passwords that were affected by the breach were reset, and they began blocking traffic associated with the attacker and removing any compromised systems from their network.

And it was a sophisticated attack. Apparently some custom, never-before-seen backdoors were inserted into the network somehow. And it does sound like an administrator's credentials were compromised. So this was a deliberate focused attack. We have to learn, as we're finding out, how to do better. There needs to be better protection. And, boy, you're right, Leo, I mean, this is devastating for customers of Anthem Blue Cross, Blue Shield, and of the other properties, about half of whom are here in California.

Leo: They're already seeing phishing attacks with the Anthem logo and email from somebody pretending to be Anthem saying, hey, we had this problem. You'd better log in, change stuff.

Steve: Click this.

Leo: Yeah. So if you're an Anthem customer, congratulations. You need to know a

lot about protecting your good credit. You need to know how to file a fraud alert. You need to know how to check your credit report regularly. I mean, welcome.

Steve: Yeah.

Leo: In fact, probably we should all be doing this.

Steve: They did set up something called AnthemFacts.com, A-N-T-H-E-M-F-A-C-T-S dotcom. And it was a little unimpressive. There's someone named Joe who runs Anthem, and he signed his name with a big happy "Joe" and noted in there that he, too, was a victim, meaning that he was one of the 80 million. It's like, well, okay, Joe, that doesn't really make...

Leo: Sure, all the Anthem employees were Anthem customers; right?

Steve: Yes, yes, exactly, yeah.

Leo: Oy oy oy oy oy, 80 million people.

Steve: Wow.

Leo: Boy, hackers are just going to have a field day.

Steve: Yeah, it's not good, it's not good. So Google announced yesterday that with Chrome 40, which is in the process of rolling out, they will be adding support for HTTP/2. Remember that HTTP/1 was the original standard. Then we've been running on 1.1 for the last 16 years, since 1999. So it is time for us to move forward. And we talked a couple weeks ago, there was that neat site, https v. http [httpvshttps.com], which demonstrated that SPDY was in fact well named. It was so speedy that it is faster to use SPDY with encryption than to use HTTP without. And to the extent that we move to this next-generation encryption, there's no longer an argument to be made that it is a speed impediment. It's just not.

So what Chrome is doing - and I'm putting it in the show notes to remind myself, this is truly Google at its best. There's nothing heavy-handed about this. All browsers now support SPDY, which started out as Chrome's test protocol. They put it in their browser. They supported it on their servers. And of course lots of people using Chrome also had the occasion to use Google's properties, increasingly so. And so Google was able to meter this and install metric surveys to tune it and tweak it and see how much benefit was gained from this part of SPDY and how much benefit was gained in the real world from that part. And ultimately Chrome, IE, Firefox, Opera, Safari, and even Amazon Silk server all support SPDY, just because it was a good thing. It then - essentially unchanged, just some little committee-ish tweaks here and there - it, essentially unchanged, became the core of the HTTP/2 protocol.

We'll give it a full podcast here as soon as it settles down and gets ready to be finalized, which is right in the process of happening, to give full coverage of what the change from 1.1 to 2 means. But they're now going to support this next generation in Chrome 40. And then they'll leave SPDY in place while HTTP/2 comes fully up to speed. And then their plan is next year, in 2016, to at some point remove support for SPDY itself from Chrome, presuming by that time that everybody, all of the - basically HTTP/2 will have had a chance to take hold. And we're still looking for more pervasive server-side support. That's sort of the next piece of this is for that to become widespread. Everything supports 1.1. And this puts good pressure, I mean, essentially SPDY itself puts good pressure on servers to begin supporting that.

Leo: If you wanted to support it, you would do it at the server level.

Steve: Correct.

Leo: You'd update Apache. You'd put a plugin or something in Apache.

Steve: Correct.

Leo: Yeah. It's not the web page itself, it's the server.

Steve: Right, exactly, because the web page is up at the application, at the content level. And this is down in the protocol level itself.

Leo: You're going to hate our new page. It's all JavaScript. It's all Node.js. You're going to just hate it.

Steve: I'll drop my drawers for TWiT.tv.

Leo: Well, but see, but you just say, "NoScript, please allow Leo to take over my computer and use it in any way he sees fit." The good news is we don't have your Social Security number, Steve.

Steve: Well, and I was...

Leo: Actually, we do. Wait a minute.

Steve: But probably not on your website.

Leo: Not on the website, I can promise you.

Steve: Not on the website.

Leo: Actually, you know what's funny, and you've probably gotten your 1099 at some point in the last few weeks...

Steve: And in fact I got a request from you to update my W9 information, and I sent that back to your W9 coordinator. I thought, oh, you have a W9 coordinator.

Leo: Oh, yes. Because, well, this is a sideline, but we use ADP, which is a payroll system. And you're actually not being paid by us. None of our employees work for us. They all work for ADP in Florida because of the way the system works. So, but we trust that they have good security because they do have your Social. We don't.

Steve: GRC always used an outside certified. That just makes so much more sense.

Leo: It's crazy to do that, yeah, yeah.

Steve: So did you see "60 Minutes"?

Leo: No, I heard they talked about DARPA. Yes?

Steve: They did. But they also gave a really disturbing demonstration, which I'm glad they did because the public needs to have seen this, and now has, of car hacking. And the most disturbing aspect of it was - it's funny, too, because the car that they were demonstrating, I'm sure aficionados of cars knew what this was, but they blanked out all of the identifying information about what make and model this car was for the sake of putting this on the air.

The disturbing thing was that not only did someone, as we've seen before, hack into the dashboard and put the "60 Minutes" reporter's name up in the display - this was Lesley Stahl who was doing this segment, and so it's "Hi, Lesley," which she thought was cute. But it overrode her braking. And in a non-braking, that is, where she slammed her foot on the pedal as the car was approaching and then drove through a set of cones, it disabled her brakes. And so first they did some little cutesy stuff, as you're showing now, where it squirted her windshield wipers and so forth.

Leo: Where did they - did they get the hacker from central casting? Because heavyset guy with a beard and a black T-shirt...

Steve: He's like the perfect hacker; right.

Leo: Yeah, yeah.

Steve: Yeah. So first they squirted soap on the windows and turned on the windshield wipers. It's like, oh, okay, that's cute. Then right there they're showing that they somehow overloaded the car's systems in order to get access to the car's networked computers. And so she thinks this funny. But then they bring her around and have her driving to a set of orange cones. I think that's what you're showing right now.

Leo: Yeah.

Steve: And it's like, okay, now go up here and stop in front of these - drive up to these cones.

Leo: And Mr. Neckbeard has a different plan. "Oh, my god, my brakes, they don't work, agh."

Steve: Now, that's disturbing, Leo, because I have a hard time, I mean, we know that cars are now full of computers and that this is going to be a problem. But for the computer to be able to override the braking system to me seems irresponsible. Maybe you need it to stutter the wheels in order to do antiskid braking. But that ought to be the limit of what it can do. I mean, there's a point here where, if the car is, as they say, "fly by wire," it has to be responsible for the way it operates.

Leo: I think that what you could point to is cost savings on this.

Steve: Yeah.

Leo: And I've talked to Ford about this, and they say we have two computer systems, and the system that runs your multimedia center...

Steve: Your entertainment system.

Leo: ...which is the point of access for a lot of these hacks, is in no way connected to the computer system that runs your car, and never should be. I mean, I'm not an expert on this stuff. But I think if the same computer is running your car and your radio, I think that's a cost savings, not a sensible thing to do.

Steve: So Senator Markey asked all of the car manufacturers, sent them a questionnaire. And I have a bunch of tweets for the show that I tweeted - I'm sorry, a bunch of links for the show that I tweeted. One is bit.ly/carhacking, which links to a 12-page report that the Senator's office just published, I think it was yesterday, which details the responses that they received in response to the questionnaire. And it's a little disturbing because a lot of the companies don't apparently have the kind of security that we would like to see in place. So I don't know how this gets resolved. We have legislation to mandate seatbelts and mileage. So it looks like, unless they take responsibility for the security of their mobile computers, you know, it's one thing for your thermostat at home to be taken over. It's a little worrisome when that can happen to your car. And like you,

I've not dug into this deeply. I don't know the details of this. But we need our carmakers to be responsible.

And of course we've got all the news about the Samsung TV listening to people. Leo? So there's been a lot of tweeting about this. And Parker Higgins, who's the EFF's director of copyright activism, tweeted that Samsung Smart TV's privacy policy warns users not to discuss personal information in front of their television. So I dug into this a little bit because I was curious what the Samsung privacy policy was. And so quoting directly from what Samsung has, Samsung says: "You can control your Smart TV and use many of its features with voice commands."

Leo: Sorry, I was in the bathroom. I apologize.

Steve: No problem.

Leo: I peed all over my shoes when you talked to me.

Steve: Oh, sorry. You're able to listen?

Leo: Oh, yeah, I keep a - I monitor. Well, I don't want to miss it.

Steve: Ah, okay.

Leo: Go ahead. Mine does this, by the way.

Steve: Okay. So we'll talk about this. So of course Samsung said: "If you enable Voice Recognition, you can interact with your Smart TV using your voice. To provide you the Voice Recognition feature, some voice commands may be transmitted along with information about your device including device identifiers, to a third-party service that converts speech to text or to the extent necessary to provide the Voice Recognition features to you.

"In addition, Samsung may collect and your device may capture voice commands and associated texts so that we can provide you with Voice Recognition features and evaluate and improve the features. Please be aware that, if your spoken words include personal or other sensitive information, that information will be among the data captured and transmitted to a third party through your use of Voice Recognition.

"If you do not enable Voice Recognition, you will not be able to use interactive voice recognition features, although you may be able to control your TV using certain predefined voice commands. While Samsung will not collect your spoken word, Samsung may still collect associated texts and other usage data so that we can evaluate the performance of the feature and improve it." What do you think?

Leo: I think this is a tempest in a teapot. I commend Samsung for...

Steve: Full disclosure.

Leo: ...being explicit, as they should be. But most devices, including, by the way, your smartphone, aren't smart enough to understand what you're saying. They have to send it to a server. Siri does this. Google does this. Windows Cortana does this. They send it to a server which interprets it and then sends the information back. And I would not expect a TV to be smart enough to do that. In fact, they say that. They say, well, if there's certain canned responses, we can handle those. But everything else is going to go to a server.

I'm not sure why they're using a third party. I'm glad they revealed that. But it may just be they don't have the expertise to do this. Who knows what they're doing? They may be using a human like an Amazon Mechanical Turk-style system to interpret what you're saying. The good news is you just turn it off. And I have, by the way. I never found the speech recognition to be very valuable. But I am not concerned about - you know, if you have an Xbox One, it's always listening. Always. And always sending everything you say back to Microsoft. I mean, it bothers people, but it doesn't - it's like, so what?

Steve: Well, yeah. So I think - I agree with you. I think full disclosure makes sense. I think that what I like is what Amazon did with the Fire TV, where you have a button you press that presumably starts it listening. You then say what you want to say, and then you let go. Now, that's less magical than "Hello Google" or "Hello Watch" or, you know, where it's sort of like always listening, so you just are able to start it to listen by addressing it using some keyword.

Leo: Although in those cases, for instance the "Hello Google," that doesn't leave the phone because the phone is smart enough - in fact you have to train it - to be able to recognize it and then send the rest of it to space. So that's much like pushing a button. And I don't know how - I'm trying to remember on my Samsung. I turned it off almost immediately, not because I was worried about privacy, but just because it wasn't useful. You can also wave at it. And every time I stretched while I was watching TV, the TV would start to do stuff. "Yes, master." So I don't remember if you issue a command or not. I think you have to somehow signal. No, they don't want to send everything you say upstream.

Steve: No. They couldn't possibly be streaming everything that everybody who owns one...

Leo: Why would they do that?

Steve: Yeah.

Leo: They're not owned by the NSA, are they? No.

Steve: Well, I think that technical users will understand, for example in the case of Siri, that your phone is not itself figuring this out. I mean, for example, your phone doesn't

know everything that you're asking Siri about. So your phone is your conduit to something which is performing this speech recognition, which is really impressive. I mean, it's like, in the same way that no one can beat computers at chess anymore, or like IBM's Watson can actually play "Jeopardy" and win against the best humans that we have, it's clear that within constraints, computers are now getting good enough to do things they have not traditionally been able to do. So winning at "Jeopardy," playing chess, and now answering questions within certain, you know, a narrow range, they're able to do, and provide valuable features.

Leo: I can't wait. Mike Elgan has the new Amazon Echo, you know, that black tube.

Steve: Uh-huh, yeah.

Leo: And it's always listening. So you say things like "Alexa, what time is it?" And in a beautifully - by the way, best voice synthesis ever - voice, it'll say "It's 4:53." And in fact I don't know if we're reviewing it today on Before You Buy. I think we are.

Steve: So right after this podcast.

Leo: Yeah. I immediately ordered it. In fact, what I want, the real problem with it is it can only be in one place. You want one in every room because you pretty soon become dependent on it. And I would like Sonos, I want Amazon to buy Sonos because I have Sonos speakers in every room. Sonos already has beautiful sound. And I want them to build that capability into Sonos speakers so that I can, as I wander through my house, say things like, "I wonder where I left my socks." It's great. Who cares who's listening?

Steve: It would also have olfactory sensors built in, then.

Leo: Clean socks. Now, you know, actually I think the real point of this story is this is why companies don't disclose, because you see all the attention Samsung got for disclosing something that pretty much everybody else is already doing, and disclosure scares the hell out of people.

Steve: They were being responsible.

Leo: They were doing the right thing.

Steve: I'm interested, though, in your comment that you quickly become dependent upon it. Is that what Mike has found?

Leo: I don't know. Now we have devices that listen all the time. And I don't use them all that much. The reason I use the phone that much is that you kind of have

to get it out and speak to it. Alexa is there, it has a very nice, some sort of array mic technology because anywhere in the room, in a normal tone of voice, you can say, well, here's an example of a query that Mike said, "Try this." Alexa - and by the way, there's only two words you can use right now, "Amazon" or "Alexa." But in fact I asked Alexa, "Alexa, is there any other name I can use for you?" And she said, "Right now you can only use Amazon and Alexa." But the implication is soon you'll be able to use more. It is like interacting with a human. And it's very Hal 9000, very much - and so the query that Mike suggested, he said, "Ask her how old Michael Jackson is." So I say, "Alexa, how old is Michael Jackson?" And then Alexa says, "Well, Michael Jackson died in [whatever year], but at the time of his death he was 50 years, eight months, three weeks and two days old." It's very smart. And I'm just - I am very - you know what? I think, if we can get over this privacy...

Steve: This is transformative.

Leo: Yes. But we have to get over this hump. And what I'm afraid of is that people are so worried about privacy that Congress will pass a law against that stuff, and we won't have this technology.

Steve: Well, I love the idea of a key phrase to enable. I think, as I said, that's why I like the Fire TV. I press the button, and then I say something, and then it goes and finds it. And it's, like, it's shockingly good. It's like, okay, this stuff has arrived. And we should understand that you verbally asking Alexa how old Michael Jackson is, is indistinguishable from you typing the question into Google.

Leo: It's that same kind of thing, yeah.

Steve: It's identical in every privacy aspect.

Leo: Right.

Steve: It's just you're doing it verbally rather than textually.

Leo: Right.

Steve: And it does become magical to be able to do it verbally.

Leo: It does. And I do think that, by the way, Kinect and these other devices, all of them, and I'm sure the Samsung, as well, have trigger phrases because they don't want to be sending every bit of data upstream all the time.

Steve: They can't.

Leo: Just the cost of it is ridiculous.

Steve: They can't, yeah. I mean, it's just almost impossible.

Leo: Too much space, yeah. So they all have a trigger. If it's not a button push, as with Siri, it's a verbal command, as it is with Google.

Steve: And there was also facial recognition I encountered in Samsung's privacy policy. What's that about? You're, like, able to log into Samsung just by staring at your screen?

Leo: Well, I haven't tried that. I don't know if my TV is smart enough to do that. But Xbox One does. When I walk in the room it says, "Hello, Leo."

Steve: Except you also said that it misrecognizes...

Leo: It thinks Lisa is her son because Lisa doesn't have an account, but apparently she looks enough like her son that when she walks in the room it says, "Hi, Michael." I, by the way, again, I guess I'm a sucker, love this. I don't have to log in. It just knows I'm there. And when you're playing a game it's great because, if another person sits down, it'll say, "Oh, hi, Michael," and Michael can join in.

Steve: And it's like my thumb is now my access to my phone. And Apple has nailed that technology, and it is never a problem any longer. It's just thank you very much.

Leo: You know, I started using the iPhone, I mentioned before the show that I started going back to the iPhone 6 because I'm going to want to get ready for the Apple Watch. And I have to admit, for that alone, the iPhone is a superior device. The fingerprint is incredible.

Steve: And the SQRL client for iOS, when you are facing a QR code, and you want to identify yourself to the website, you just put your thumb on the button, and you're logged in because it's like, okay, the person doesn't just have Steve's phone, he also has Steve's finger. And hopefully it's still attached.

Leo: I just think that there should be - Larry Page asked for this a couple years ago at Google I/O - a place for those of us who just want all the new technology and don't give a damn if somebody's listening, so that we can go there, and the rest of you can sit in your little cave and have privacy. Good luck. I don't care. I want the stuff.

Steve: I think the right solution is for Samsung not to be hurt because everybody else also needs to disclose, in which case Samsung isn't being penalized. I think disclosure is important. But just make it an option. Allow it to be turned off.

Leo: Exactly.

Steve: As Samsung has done.

Leo: Exactly. That's the right answer.

Steve: And I think it's neat that Amazon gave their thing the name "Alexa" because you're not probably going to - it's not "coffeepot" or something that you might say by mistake. It's a word, it's a name for that. And I guess maybe if it's not sensitive enough, you might be talking about a Lexus car and it might go boing. Does it make a sound when it acknowledges you?

Leo: No. There's a little glowing ring on the top that wakes up and starts going [making sound].

Steve: I've seen the picture, or I've seen the commercial. It's cool.

Leo: Some idiot in the chatroom said, oh, well, it's just the same as Anthem giving up your Social Security number. No, it's not. Sorry. It's not the same. You can want security of your Social Security number and still want something you can talk to in your house. They're not mutual inconsistent. And I agree with you. Give us the choice. Let us turn - but what I really am concerned about is these privacy advocates shutting down technologies because they don't want that to happen. And so just give us the choice and disclose. That's it. That's all you need to do. Give us the choice and disclose. And I agree with you, if everybody did that, then there'd be no harm to Samsung for telling the truth.

Steve: Right. And the fact is I think certainly for some people they will be concerned enough that they'll choose not to use it. For many others, it's like, hey, this is convenient. I want to be able to just ask the air what time of day it is.

Leo: So cool. Choice, that's all.

Steve: Yeah. Yeah.

Leo: [Janie] says, well, that's what privacy is. No, no. See, that's the problem. There are people out there who would very much like to prohibit this kind of stuff. And if you lobby loud and long enough about it, Congress is going to say, well, I guess we should prevent this. And I really hate to see technology held back by Luddites. That's all.

Steve: Yeah, I think it's too late to have it held back. I think we just - and I have no problem with requiring the people who offer these services to be responsible with the

information that is collected. We need that, too. I mean, and here's Anthem. Anthem is offering a service called health insurance, but we could argue the fact that in 2015 their database of 80 million subscribers was not encrypted was irresponsible of them. They can say, oh, well, the nature of the attack was such that they would have been able to decrypt it anyway. Okay, this attack, but not all attacks. And having that much data, that crucial, crucially personal data not encrypted, there is just no excuse for that in 2015. So thank you for offering health insurance, but not for doing an inadequate job of protecting the Social Security number that you got along with it. And why do they have to have that? Is that clearly important for them to have my...

Leo: No. In fact, Dr. Mom says she does not give it to health insurance companies. So, but I just don't...

Steve: Ah, interesting.

Leo: But I just don't know what the policy - I don't know what hoops you'd have to jump through. You obviously have to give it to an employer. The whole purpose of the Social Security number is to identify income.

Steve: For tax reporting purposes.

Leo: Well, for taxes and for retirement, for Social Security.

Steve: And Social Security, right, accrual, yeah.

Leo: But I don't see any reason to be using it as an identification number for any other purpose.

Steve: No. So we have a nice story. Eighteen years ago Werner, is it Koch, K-O-C-H, decided to create GPG, in 1997. He wrote the first instance of Gnu Privacy Guard.

Leo: Yay.

Steve: Yes.

Leo: I use it. Love it.

Steve: And it was never a huge money winner for him. He notes that since 2001, meaning since it was four years old, 14 years ago, it had been generating about $25,000 a year in GPG donations, so below the income that a similarly skilled programmer would have been able to generate by hiring himself out to a company, yet he kept it going. In 2013 he was right on the verge of calling it quits. He has a young daughter. He's the sole breadwinner for his household. And it just - it was straining him too much to be the sole

developer and support for this thing and not be generating any greater amount of revenue.

But the Edward Snowden revelations occurred right at the point where he was thinking of bailing; and he thought, okay, I can't. This is clearly too important. And of course GPG was what Snowden was using in order to communicate. So he decided to stick it out. He then in December started trying to crowd fund and generate revenue, which was more successful than he had been so far, but it wasn't generating the amount of money that he was hoping to. So on February 5th a Julia Angwin did a posting, a story about him in ProPublica, and this went public, her story did, her posting, at 10:24 a.m. on the 5th. By 8:10 p.m. that day, he had reached his funding goal, which was $137K. Facebook and the online payment processor Stripe both independently pledged to donate $50,000 a year.

Leo: Yay. Yay.

Steve: And her going public with the story allowed him to then disclose that the Linux Foundation's Core Infrastructure Initiative had just the week before given him a one-time grant of $60,000. So this was a big win. He's now rolling in dough. He can afford to hire a full-time programmer, which he has wanted to do, in order to really get behind this and give it what it needs. And thanks to Facebook and Stripe - and I want to say to both Facebook and Stripe, thank you for this. They have, combined, created a 100K a year ongoing pledge to support GPG. So as we've seen, it's often just the case that people need to make it clear that something that is essentially core infrastructure is in need of some financial help. And there are big public companies and well-funded organizations that are able to say, oh, glad you told us. We're going to help you.

Leo: Yay.

Steve: Yeah. So this is a nifty story and a nice turnaround.

Leo: And we've recommended Gnu Privacy Guard, GPG, for a long time. It's basically the open source version of PGP, which makes it better than PGP because it's open source.

Steve: Right.

Leo: And it's available - he maintains Gpg4win, which is the Windows version - Gpg, the number "4," win. And then there's GPGTools, which is the Mac version, which I recommend. And I've given them money. I didn't realize it didn't go back, flow back to him. So I'm glad to support him. It's such important work.

Steve: Yeah. I also ran across - and I'm trying to think where this was. I think it was someone - oh, no. It was in the mailbag for today's Q&A, but sent anonymously, so I don't know who it was from. But it's an interesting slide presentation. I created a bit.ly link, bit.ly/sslslides, all lower case, S-S-L-S-L-I-D-E-S. And what I liked about this - so this was a Google presentation showing the work of the Chromium team, trying to come

up with the way to communicate the difficult concept of the various ways a communication over the Internet, through your browser, could be insecure. And really I was impressed by this. It demonstrates the depth of their intent. And it was just a good demonstration of, I don't know, it's like 30-some slides, at least, where they show the way Chrome has evolved over time to show different types of problems. And it illustrates - and they're also, like, giving themselves a grade, A through F. And they're demonstrating from their own metrics the number of people that have responded in which fashion to different types of warnings that Chrome has offered over time.

Anyway, it's really interesting: sslslides is the resource on bit.ly. And so I wanted to aim our listeners at it because I think we'll find it interesting. And it highlights the real problem, sort of a fundamental problem with this whole experience, that is, we want using the web to be simple and easy and safe. Yet it really does rely on an interlocking family of complex technologies. You were talking on The Tech Guy show this weekend, Leo, about how, for example, I think somebody was - you were giving a complete answer to a complex question. And somebody was...

Leo: Somebody said it was a Steve Gibson answer.

Steve: Right. They were complaining that it was too complicated.

Leo: Well, it probably was for the radio audience.

Steve: And you said...

Leo: But there's no way to do it simpler.

Steve: ...look, sorry, right. If you gave a simple answer, it would be wrong. And unfortunately, only a full answer is correct.

Leo: We were talking about bandwidth and why, you know, DSL versus cable and, oh, my god, it was - yeah.

Steve: Right, right, right. So...

Leo: So this is good. This slide show is good.

Steve: Yeah. It really is good because it demonstrates the problem of what can you show the user that they will understand because, you know, you show, like, the fields of a security certificate, and even podcast listeners, I mean, even listeners to our podcast are like, wait a minute, is that good or bad? I mean, it's complex stuff. And so they break it down in terms of the visual semantics, how can we convey what we want to convey - which is unfortunately complex, there's just no way around the fact that it's complicated - so that people will get the message that we're trying to give them, rather than just go, uh, what does that mean? Yeah. So anyway, I commend that slide deck to our listeners.

I think everyone will find it interesting. And you've been stepping through it while I've been talking.

Leo: Yeah. She grades various messages as to intelligibility and accuracy.

Steve: Right.

Leo: It's not an easy thing to do.

Steve: No. I mean, what I liked about this was it really does highlight that this is a hard problem, that is, we have complex technology which users who don't want to know anything about that have to understand, just like your caller over the weekend said, well, why are my pages loading slow? And it's like, oh. Where do I begin?

Leo: That was the question, wasn't it, yeah.

Steve: Yeah.

Leo: Um, well...

Steve: Okay. Let's see.

Leo: Blame Canada. There's the easy answer. But I don't know how useful that is.

Steve: Yeah.

Leo: This is good. And I like this: Opinionated design works where text fails.

Steve: Yeah, exactly. Where you sort of have to have Fred Flintstone with a mallet hitting them over the head.

Leo: Yeah, say it out loud, yeah.

Steve: In order to, really, to get their attention.

Leo: It's good. And she's - I don't know if she's a Google employee or she's with the Chromium effort, which is a Google effort, so...

Steve: Right. Adrienne Porter Felt is her name, with the Chrome security team, where

they're really - they're working to figure this...

Leo: Thinking about what do we tell people.

Steve: Yeah. How? How do we explain this? Yeah. Which is why I'll be interested to see what they do where they're trying to tell people that, I mean, look at these - the issues that addressed were, like, fraudulent certificates, something clearly fraudulent. So how, then, are they going to say the certificate of the website you're visiting is using a hash function that we're trying to get websites to stop using, even though there's really no problem with it today? Good luck communicating that subtlety, you know, in an icon. I don't know how you do that.

Oh, speaking of slides, many people have wished that my DigiCert presentation video slides were available. And so while I was in bit.ly land today I created bit.ly/sqrlslides, which gives you a PDF of my slide presentation for the SQRL presentation, so you could sort of flip through it while I was jumping around onstage there, or just browse through it yourself. So a bunch of people have asked.

I did want to follow up on last week. We had Audible as a sponsor, and we mentioned in that context the Expanse series, which you had listened to, Leo, in the past because when you went there they were available on Audible. I've finished the first book now, which was wonderful, really fun. So I was liking it last week, though I think I was like a quarter of the way in. It's behind me now, and I'm into No. 2, which is every bit as good. So I can now without reservation recommend at least the first book and as far as I am in the second book of the Expanse series of sci-fi. It's just enjoyable. It's sort of classic space opera. It's not super complex. It's not nearly as involved as Peter Hamilton's stuff, but just good old-fashioned space opera. And remember that the reason I got into this was that Mark Thompson noted that the Syfy Channel was turning this into a series that we'll be seeing later this year, so I wanted to read it first. And some interesting, fun stuff. So, yay.

And for my weekly SpinRite mention, I thought I would just - in the last 12 hours I saw two tweets that came by because they happened to have the @SGgrc in them. And they weren't actually directed to me. James Bliss tweeted: "Again, thanks to @SGgrc, it appears to others that I have a superpower. I do, all thanks to him, and it's called SpinRite." And then he linked to GRC.com, the "What It Does" page on GRC.com. That was the first tweet.

And the other one was from James Munn, who apparently I saw part of a conversation, a dialogue he was having, because again he had @SGgrc in it. So he tweeted to a Patrick Klepek, he said: "Get and run SpinRite 6 on it from the great @SGgrc! It has rescued many a HD [obviously hard drive] for me." So thanks, James. Oh, they're both named James. James and James, thanks for your tweets and letting me share them, yeah.

Leo: Question No. 1 from Matt at an undisclosed location. He worries about fuzzy string matching. Who doesn't?

Steve: I know. Those fuzzy strings.

Leo: You guys are such geeks, man. Matt writes - oh, wait a minute. Now somehow I blew it. Here we go: Steve, I thought I recalled you saying that you'd developed a novel method - oh, I remember this - for fuzzy string matching. I believe it was for processing/recognizing SpinRite testimonials in your records. I realize your plate's awfully full and you've no shortage of topics for Security Now!, but I wouldn't mind hearing about your solution. Thanks for all you do for your community.

Steve: Okay. So I see people asking this from time to time. And Matt's memory is correct. I spent some time dipping into sort of an unsolved question in computer science. It was technically called LRS, Longest Repeating String. So it wasn't a fuzzy string match as much as a longest repeating string. The idea was that I was sometimes editing the testimonials to fix grammar a little bit, just because I'm posting it in someone's name, and I want it to look right for them. And that meant that, if I wanted to find duplicates in order not to duplicate post, I needed to find a match where most of it matched, but not all of it matched. And that meant that, if there was a long run of text that was exactly identical, then that was probably the same testimonial in two different locations.

So I and a bunch of neat followers in - we have a newsgroup called "thinktank," where we go to sort of do brainstorming stuff - worked out what was actually probably a new algorithm in computer science which I was able to describe in text well enough to the people who had been following along, because I iterated the solution three or four times, every time I came up with something substantially better. They were able to independently write it in a higher level language because of course I was implementing mine in assembler. Anyway, the point is that it is on GRC at GRC.com/dev/lrs. And the source code is there from - all I can think of is his handle, Sparky.

Leo: MASM?

Steve: Paul...

Leo: Oh, a person wrote it.

Steve: Yeah. And all of mine are there.

Leo: You could say just Sparky's source code is there.

Steve: Sparky, yeah. I can't believe I'm blanking on his first name, but...

Leo: Paul Byford?

Steve: Yeah, it's Byford, Paul Byford. And all of my EXEs are there, and a list of all the iterations we went through. So for anyone who's interested, I never had a chance to put it on the website because it wasn't crucial. I solved the problem. And I need to get it documented because it is a new solution, I think, that no one has come up with before. But it's there, /dev/lrs, as in longest repeating string, for anybody who's curious.

Leo: Good. Good, good, good. I thought we did a show, but maybe we just talked about it. But we did talk about it.

Steve: We just talked about it.

Leo: Didn't we? Yeah.

Steve: Yeah, yeah, yeah. We did at the time. And what I want to do is I think in order to convey it - it's a cool solution. And I wanted to animate it so that you could see it working in order to have that, like, that a-ha moment where it's like, I understand that, because it's working. But, so, one of these days. After SpinRite 6.1. It's another thing that I'm pushing to the other side of SpinRite 6.1.

Leo: But to be clear, it's not fuzzy matching, it's direct explicit matching.

Steve: Correct. And there were lots of applications, like genetics, like gene sequencing, to find the longest run of gene sequences. I mean, what it is, is it's incredibly fast for finding the longest repeating string in a large corpus, which turns out to be a rather hard problem in computer science.

Leo: I would bet. I've never looked at grep source code. But I would bet, I would hope that the people who wrote the various regular expression parsing tools in various languages and various libraries would have done something that's efficient like that. That's one of the features, one of the things that grep can do.

Steve: I don't think it can find a longest duplicate string. See, the idea is you could say "find this string" in something else. This goes the next step. It says "find the longest repetition, not knowing what it is." So you're not asking for it to find a certain thing. You're saying what's the longest string that occurs twice in this entire text?

Leo: Ah. Well, I'm sure you can make an expression that does that, but I'm not sure I'd want to try. I'd be surprised if you couldn't do that. There's a great book, by the way, on grep that I love: "Mastering Regular Expressions," I think it's called.

Steve: Yes. And I think I mentioned, oh, no, it was on Coding 101. By the way, I never mentioned this, but I've been on the last three...

Leo: I heard. I haven't seen it yet.

Steve: ...episodes of Padre's Coding 101.

Leo: What, are you teaching assembly language?

Steve: The Padre just asked me to come and talk about stuff.

Leo: That's great.

Steve: And so for three - and because I'm able to do that off the cuff, I don't have to do any prep. It wasn't stealing any time from my work on SQRL. So I said, yeah, I'll show up. And so I did that three times. But I was mentioning that - because the week before last I was talking about compilers versus assembly language. And some people took umbrage at the idea that I was saying, eh, you know, assembly language is more efficient than compilers. But I drew the example. I used regex as an example of something I would never want to have to write in assembler.

Leo: No.

Steve: Because it's just a huge complex problem. And there are some things that make sense in assembly language; other things, eh, they just really don't so much.

Leo: This is the book I was referring to by Jeffrey Friedl. It's a computer science classic.

Steve: I own it.

Leo: And it's fun, actually.

Steve: Yup. Yup.

Leo: It's actually - maybe that's how geeky you and I are. But actually it's a fun book to read. Question 2, Sean Robinson. He wonders whatever became of the Portable Sound Blaster project?

Steve: Another blast from the past.

Leo: Mr. G., Mr. G., give us an update, what's going on?

Steve: Okay. So that, similarly, I didn't update the pages for it at GRC just because it sort of didn't seem necessary. But enough people ask continuously that I will get around to fixing that, too. Everything that happened, happened in plain sight over on the Google Groups. And so if you google "portable sound blaster," you will find groups.google.com, and I think it's just, like, /portablesoundblaster. Everything that you want to know is

there, including, and I have at the top of the first page that comes up, a zip file of the finished design.

What we ended up with was a very elegant, very small and simple-to-build circuit, perfect for like a little father-and-son project, a bill of materials that you can all source from DigiKey.com, which is a great source, an online source for electronics components. And it produces an extremely loud, high-frequency sound. Now, the problem was that what we learned was that, from my recounting my experiences with mine, people were wanting too much from it. They were saying, well, you know, two houses down there's a barking dog, and I want this to shut it up. And it's like, I'm sorry, that's just - it won't work. Or 25 yards away is a dog. It's like, well, no, it doesn't do that. Or I'm on the ninth floor, and there's a dog barking on the third floor. It's like, sorry, no, it won't do that.

So what it will do is, if you were being attacked, this would just stop a dog cold at three feet away from you. Because remember, that's what I did. That was my use for it was basically, at pointblank range, to blast this ferocious German shepherd in the face. And after a couple of those, it no longer lunged at the gate. And then it did knock seagulls out of the sky, so there was that. But it's not going to keep a barking dog from barking a long way away. That just isn't - we don't have - I don't think that technology exists. So if you want a handheld defensive device - for example, a mailman. This would be perfect for a mailman who's being nipped at or for dogs chasing him. And there were some people in the group who - and I should say a bunch of people successfully built it, and it works for them.

Leo: Wow, that's neat.

Steve: As a nice little handheld, high-frequency blaster. It was funny, too, because some guys, while they were building it and testing it, they couldn't hear it, but their teenage daughters upstairs were complaining.

Leo: Dad.

Steve: Stop that, yeah. So that's where we are. Anybody who wants it, I haven't moved, I haven't ever had time to transport that all back over to GRC. But it is "portablesoundblaster" in a Google Group, and everything is there - the design plans and schematics. And everybody posted their results and pictures of their projects and everything.

Leo: So we're going to stamp that one completed.

Steve: Done.

Leo: Done. Brian Tannahill in Overland Park, Kansas, has come up with maybe another use for SQRL: When I log into the system at work from home - it's a VPN - I have to enter my logon and password information three times. First I connect to the VPN. Then I connect to the network. And finally I log into my own machine at the office. Would it be practical for an employer to use SQRL to simplify remote logins?

Steve: So naturally my increasing proximity to having SQRL up and running and demonstratable and finished and the fact that over the holidays we aired the DigiCert presentation of it has stirred up a lot more interest. I don't want this podcast to become the SQRL podcast, just because there's a lot more going on. And so I've been responding to a lot of people just by mail who had questions. For example, someone was asking, hey, if we had a static QR code, could people use that? And I explained, no, because one person's signature could then be captured, and it could be replayed in a classic replay attack in order to impersonate them, which is why the SQRL code needs to change every time, and you need not to be able to use it more than once.

And so Brian - but Brian's question was a little different. And so I just sort of wanted to capture everybody's questions and say that there are indeed other things, many other things, where this protocol could be used. And in Brian's instance, absolutely. He's looking at various types of online logon, first to a VPN, then to his network, then to his machine. And I don't - it's not clear how his machine could log him in.

But if, I mean, we know that Windows, if it was a Windows system, you are able to replace the logon technology. I could easily imagine that Windows could present you with a QR code which you would then see, having remote connected to your machine on your VPN, then on your network. So you could very easily just present, essentially, three different QR codes at each stage, either tap it and the client on your machine would log you in, or snap it with your smartphone in order to proceed through each stage. So, yeah, there are many different things. Basically it needs to be an online login where you're challenged with a unique QR code to which only your SQRL client is able to respond. And it's that simple, really.

Leo: Question 4 comes from Andy Marks in Louisville, Kentucky, with some thoughts about improving Tor: As you said last week, vulnerabilities of Tor include analysis of timing, the size of the data. It seems a logical improvement of Tor is to mask the actual time by increasing it and setting a fixed or random data size, even if it gets bulky. That could help mitigate the problems with Tor. Tor is a program that runs on a server. Changing Tor to do these two tasks and rolling out the changes shouldn't be unrealistic. I know it would need to get tested. I don't see the big deal with Tor being changed to improve security. In addition, the number of Tor nodes and the amount of traffic, even if it is artificial, can improve Tor security. Does it make sense? A lot of programs are broken and need to be fixed. Tor can be fixed. Andy Marks, Certified Ethical Hacker.

Steve: So, okay. This question stands in for all of the similar questions that I encountered. And the reason I chose this one was Andy's conclusion was perfect for making my point: "A lot of programs are broken and need to be fixed; Tor can be fixed." No, it can't.

Leo: Oh.

Steve: And so I love that Andy finished that way because what I hoped to convey, and I will underscore it, is that this isn't a problem with Tor. This is asking Tor to do something it really can't do. It's not a bug. It's not adding something more. It's that the Internet resists anonymity. The nature of it actively resists providing that. It wasn't designed to provide anonymity. It isn't good at it. And so we're trying to - and Tor needs to be considered an experiment. Usable, yes. Worthwhile, yes. Better than nothing, yes.

Perfect, no. And nothing can make it perfect. It can't be fixed. Yes, you could have more nodes. But then the nation-state organization just needs to monitor more of it. And in fact some of the studies demonstrated that just being centrally located gave a single entity enough information in order to deanonymize people. And then there's that problem of you being able to use Tor to confirm an identity, which is an even more powerful attack against it.

So the point I wanted to make is not that it should be scrapped, but that it was originally created as an experiment; that it certainly has its purpose. Communication does go in, and you know not where it's going to come out. It provides anonymity services. I just wanted to sort of say that researchers have broken that guarantee, and there's no fix. You could continue to do things to sort of water down or make it more difficult to break that anonymity guarantee. But we're in that fuzzy world where we're trying to do something which is fundamentally, I don't want to say impossible, but kind of impossible. Like denial of service attacks, where they just keep getting bigger, and so we keep making the pipes bigger to absorb them, and so they get bigger. It's like that. The Internet wasn't designed to provide this. It just won't.

Leo: End of story.

Steve: Yeah.

Leo: Yeah, I got a lot of tweets, people say, oh, you guys don't understand Tor, blah blah blah.

Steve: Right.

Leo: And I think you do. I want to just say I think Steve knows what he's talking about.

Steve: Well, and the people who were most upset with me were Tor...

Leo: Tor users.

Steve: Were Tor node, well, or Tor node...

Leo: Tor node operators, yeah.

Steve: ...operators who were like, well, he doesn't know what he's talking about. It's like, well, and they didn't listen to the podcast. They just saw some of the Twitter traffic and jumped in for the ride. So it's like, oh, okay.

Leo: Justin Aborn in Boston. He wants to know how to be sure about emailed links.

He wants to know how whether to click on them: My bank just emailed me a clickable link. I'm 99.9% sure it's truly them, but I navigate to their site by hand, rather than click on the emailed link. To check the fit of my tinfoil hat, what do you recommend as the minimum procedure to confidently click an emailed URL? It would be a lot more convenient if we could just click on them.

Steve: Yeah. And I liked this also because in the context of Anthem, as you said, Leo, we're seeing now a big phishing wave of fake email coming out. The only way, I mean, the old-school way is to look at the email headers, which are generally available. But, boy, that's confusing. And headers are highly prone to being spoofed. I think the only thing I could suggest, first of all, is don't. They're just, you know, it's really not worth it. But if you have to, what you need to do is look at the source. That is, you need to be able to examine the source of the email.

The problem is that email today is HTML. And there's what you see is the result of the HTML markup which has created a presentation. And so you can see text that is underlined that says "Click here to email Anthem." Or I don't think he gave an example. Or his bank. And in fact it can even show you http://bankofamerica.com, like with no typos, exactly that URL, except that that's the presentation of the HTML. The markup is in brackets on either side of that, and it's hidden from you, by design, by the browser. By the browser or now, you know, email has become HTML, so your email client is hiding that on purpose to give you a nice, visible, simplified link to click on.

So it's only by looking at the source that you can verify the actual URL that you're going to click on, if you did. And you might very well see that http://www.bankofamerica.com inside of brackets where on the left-hand side there is http://fakebankofamericawebsite.com, which is the actual URL that you will visit if you click that link. It's only by looking at the source that you can know. And the other problem is scripting gets involved, too, because there could be an on-click phrase, even if the href, as it's called, is correct. If there was an on-click, it turns out the JavaScript gets invoked before the href in the link is visited.

I just was dealing with all of this, as it happens, because I've added automation to the SQRL demo so that when you authenticate with a client the web page, the SQRL demo web page immediately, instantly updates itself to show you that you're now logged in. So I was visiting all this. And the JavaScript is invoked first. And that could be in an included library that you don't even see. So, boy. Unfortunately, the bad guys have a real advantage here. And I hope maybe I've made the case for my first recommendation, which is don't, because...

Leo: So a number of people in the chatroom are saying in some email clients, if, for instance, you hover your mouse over the - the real problem is that the presentation layer is HTML. That hides what the actual link is, even if it looks like it's a link, as you said. But if you hover your mouse over that, can you capture on hover in JavaScript and prevent the status line from showing the actual URL? Or am I going to see the actual URL in the status line at that point?

Steve: Whether it was in the status line, or sometimes it comes up as a little toolkit right there.

Leo: As a toolkit, right.

Steve: Right. Unfortunately, that will show you the static href, not the on-click call.

Leo: On-click.

Steve: Yes.

Leo: And that's what you're going to go to is what happens when you click.

Steve: Yes.

Leo: So it can actually be so obfuscated that it's in JavaScript.

Steve: Even that, right.

Leo: And you can, you know, you say, well, view source. But even then the JavaScript could be further obfuscated.

Steve: Oh, yeah.

Leo: You wouldn't see anything that says HTTP. You might see just nonsense.

Steve: Yeah.

Leo: Wow. So hovering is not going to do it.

Steve: Not even that.

Leo: You really just shouldn't. You should, I guess, what you should do is manually go to the website, by hand enter the URL.

Steve: I really think, yes, in fact, I think that that's...

Leo: So right-click, copy, and paste isn't going to do it, either.

Steve: Nope. It won't because you're actually, you're going to execute code as a result

of clicking on that.

Leo: Wow.

Steve: Yeah.

Leo: Isn't that amazing. That's great. That's - I think, frankly, turn off HTML email, period. It shouldn't be allowed. It's a bad idea.

Steve: And scripting in email. I mean, how much malware has crawled into people's machines from email scripting?

Leo: Yeah, yeah, yeah. A good email client will not do HTML. Unfortunately, most of us now use web browsers to do email.

Steve: Right.

Leo: Which means you're screwed.

Steve: Right. Basically you are...

Leo: Don't click that link.

Steve: You're receiving a web page from someone you have no control over, you don't know who they are, they're claiming to be somebody who is working in your benefit. The best advice, and I don't remember where it originated, if it was from Brian Krebs or someone else, but I loved it, and we've discussed it here through the years, and that is never do something that you didn't go in search of. If a popup says, oh, you need to update your Flash Player, no. If you weren't going, if you didn't have some reason to go looking to update your Flash Player, don't accept an offer to do so. You just can't do that safely.

Leo: Back it up. All right. Moving on to Michael Horowitz, who, by the way, wrote Computerworld's, or writes Computerworld's "Defensive Computing" column, so there. He says: Just an FYI, Steve. For those of us with separate routers and modems, it turns out you can communicate with the modem via a private IP address, even through a router. I tested this with multiple modems and routers. And it's in his Computerworld article, "Talk to Your Modem." This can be a useful feature for learning about an ISP connection. My next blog post will detail how it's a bad thing. Some modems have clickable buttons and no passwords, letting malicious JavaScript click the buttons. Oy oy oy.

Steve: Yup. It's funny because, by coincidence, I was just working with a buddy of mine

who's having a problem with his cable modem. And he had a Motorola SURFboard. Is it a SURFboard? Yeah. And it turns out that the Motorola's LAN-facing interface responds to 192.168 - and there is a picture of it, you've got it right up there - 192.168.100.1. So that's a private network on the LAN side. So then your router, which is behind it, probably becomes 192.168.100.2, that is, another on its WAN side. It's on that network to the cable modem. And then of course on the LAN side of the router is the IP, the set of IPs, things like 192.168.0.something or .1.something.

And the point is that - and this is what Michael was noting was that the router will send anything not in its LAN network outside. So that inside you can do 192.168.100.1. The router will see that that's not .0.1 or .1.1 or something in its network. So the 192.168.100.1 goes out the WAN interface, reaches the cable modem where you can now bring that up. And apparently he's gone further, and there are nasty things that could be done, as he said, by JavaScript running in a browser, to penetrate your router and get to your cable modem, which he'll be talking about in the future. I just wanted to raise the flag to our listeners, who may have a separate router and modem, that that's possible. And I can vouch for the fact that, in fact, it is. Never a dull moment.

Leo: Yeah. But I do have to say the information that you're getting from the modem, it's kind of interesting to see what the cable modem's up to.

Steve: In fact, that's why we did this over the weekend was you're able to get the signal strength of upstream and downstream and, like, really useful diagnostic information to see how many channels you've got bonded, what kind of connection it's got upstream and downstream to the ISP.

Leo: How do you find the private IP address for the modem? You just google it? The model and google it?

Steve: You could look at the WAN IP of your router.

Leo: Oh, see what's attached, yeah, yeah.

Steve: Yes, exactly.

Leo: Okay. There you go.

Steve: But it's interesting because he gave the example of 100.1, and that's what we found this weekend. That's what apparently they generally use.

Leo: Interesting. I'm going to try it tonight. I have a Comcast business account, so I have business-class router. I don't know who makes it, but that'll be interesting to play with. Glenn Musser, in Phoenixville, Pennsylvania and Fort Myers, Florida - a little snowbird action there - wonders about CloudFlare's SSL option: Steve, I have a few websites with non-sensitive information. I have used CloudFlare.com's free

services for some time, and I recently turned on their "Flexible SSL" feature. They explain that visitors have SSL between the visitor and CloudFlare. The visitor sees HTTPS on the site, but of course no SSL between CloudFlare and my web server.

For example, SeniorTechGroup.com supports my free tech group gatherings. I don't get paid in any way, so I don't want the extra cost of paying for an SSL certificate. My site, when you go there, SeniorTechGroup.com, shows the green lock in the web browser. But does that have any value? Any downsides? Thanks for your great podcast. I listen to every one. You inspire many security discussions at our tech group gatherings. And he gives a link to the CloudFlare SSL Help page [support.cloudflare.com/hc/en-us/articles/200170416].

Steve: So this is sort of a mixed blessing. I was a little disturbed to see CloudFlare doing this. They really are responsible about reminding and explaining that this is not actually providing any security. The problem is I'm not sure what value it offers to show people - yes, you're showing it there on the screen.

Leo: It looks good. I see a green padlock and everything.

Steve: Yeah. And the problem is, in fact, it's not a secure connection to the destination web server. Essentially CloudFlare is acting like a proxy that is stripping SSL security from the connection. And again, they make it very clear, and they say, you know, this is not safe. Yet they're offering it. And I don't get why they're offering it because it's really, I mean, okay. So what benefit is there? There would be the benefit in an open WiFi setting of the traffic from the person surfing to that site getting to CloudFlare servers, that is, out of the open, unencrypted environment. Similarly in a hotel setting, for example, where you either have open WiFi, or maybe a shared Ethernet. We've discussed that at length in years past.

So you would have encryption to prevent eavesdropping to CloudFlare's servers. They're then going to terminate the SSL connection at them. Essentially they're synthesizing a certificate on the fly in order to terminate that connection. Then they set up a non-SSL connection back to the actual target server. So, mmm, no security exists there. And anybody sniffing the Internet would be able to see the traffic. And so the only, I guess the only concern is that there could be some tendency to trust what should not really be trusted to the degree we would be used to trusting it. But I can't argue that, for example, encrypting in an open WiFi setting is a useful thing to offer. That's better than not having any at all. So there's that value.

Leo: You can examine the certificate, and you'll see that it belongs to CloudFlare.

Steve: Yeah.

Leo: But I don't know if the people who use this site would do that.

Steve: Yeah. Given that it is now - that annual expiring certificates can be had for free, Glenn, maybe it makes more sense just to get a free certificate from RapidSSL or Start, I

think it was StartSSL.com. Use their free one-year certificate. It does require an annual annoyance of updating the certificate. But it never costs you anything. And then you've got your own HTTPS and real security. And you can still run that through CloudFlare. They're able to then create a secure connection to your server, as well.

Leo: Question 8 comes to us from Justin Malone in Lacey, Washington. He thinks he's found a HSTS vulnerability: What would happen if an attacker were able to intercept and modify regular HTTP traffic and add the HSTS header as the traffic is passing by? Would this force devices to attempt to make a secure connection to a server that doesn't support it, or does the implementation of HSTS require the client device be able to see a certificate prior to caching the STS header?

Steve: Well, I loved this question. I take my hat off to Justin for thinking, for recognizing the problem. The good news is the designers of the HTTP Strict Transport Security, which is what HSTS stands for, thought of it, too. The problem that Justin notes is that, once you have given a - once a server has supplied the HSTS header to the client, the client will cache it. And until that HSTS header expires, and the expiration is part of what the client receives, the client will refuse nonsecure connections. So what if a server that did not offer HSTS support, that is, just a regular HTTP server, what if somebody maliciously intercepted that interceptable connection - it's interceptable because it's not secure - and tacked one of those HSTS headers into the server's response going back to the client? Then the client would go, oh, this site is saying from now on always use HSTS. And it would then refuse not to. And if the server didn't support it, it would break that client's ability to connect.

The HSTS guys foresaw that problem. And so the rule is no client will accept an HSTS header unless it is receiving it over an SSL connection which is completely verified with an up-to-date cert and in every way correct from the server. So that prevents exactly this malicious attack which Justin foresaw.

Leo: Brilliant.

Steve: Very cool.

Leo: No. 9, Paul Dove in Hampton, United Kingdom wonders whether there's a way to block Flash completely in every browser. Oh, I like that: Hi, Steve. With yet another Flash vulnerability in the news this week, I think it's time we got rid of it completely. But with Flash built into the Chrome browser, it's not that easy. Even if you disable the plugin for yourself, I've still seen it enabled for other users of the PC. I've googled for a registry hack or something like that to permanently disable Flash for all users, but couldn't find anything. Then I wondered, is there a way to configure a firewall to block all Flash content? Do you think that's possible?

Steve: Well, once upon a time, before all of our communications was encrypted, it would have been possible. Do you remember Proxomitron, Leo?

Leo: Yeah, yeah.

Steve: Yeah. Proxomitron was a really very cool local proxy, the idea being that your browser connected to it, and then it connected to the Internet on your browser's behalf, just like a proxy. And it was a - you could create rules that did all kinds of cool things. People used it to change their hosts headers, their user-agent, basically to tune and tweak their web browser interaction. And one of the things you could do, for example, would be to - and Proxomitron did this - would be to examine a page coming back from a server and, like, strip out things that you didn't want to have. And, for example, Flash tags were often stripped out, or ads were stripped out, all kinds of things.

The problem is now we're increasingly going to security, to SSL/TLS connections that nothing can see into between your browser and the remote server. I've poked around at Chrome, and I don't see anything other than going to chrome://plugins. That's not something that is normally available at the settings page. There you can see extensions but not plugins. But if in Chrome, you go to - up in the URL you put chrome://plugins, no hyphens or anything, just P-L-U-G-I-N-S. That'll take you to a page most people don't normally see, which lists rather comprehensively all kinds of things that are typically enabled unless there's some problem that Chrome has found, all the things that Chrome can bring to bear in order to display pages. And Adobe Flash Player is right there among them.

Now, as Leo is pointing to right here in the podcast, you are able to disable it. When you click that, it turns, like it grays out the whole region, showing you that it is no longer enabled. And then you can click it again to enable it. But as Paul noted, that's a per-user setting. Nothing that I could find in Chrome is global. So I don't see a solution other than going to each user's session and manually disabling Flash, the Adobe Flash plugin for that user and that login session. That's the only thing I can see to do it.

Leo: Can we at least say that Flash in Chrome is safer? Because it is sandboxed; isn't it? And they keep it up to date for you and all that?

Steve: Yeah. I agree with you. I think that using it in Flash is a better place. When I opened Chrome, because I'm not in there a lot, mine was auto-grayed out with a warning that it was no longer current. And so I updated Flash because we just had those two zero-days. In fact, that's what triggered Paul's email is all of these zero-day vulnerabilities. And then it came back to life, and then I manually disabled it because I'd just as soon fly without it if I can.

Leo: You can. So, but if - okay. So updating Chrome does not automatically update the Flash in Chrome?

Steve: I think it does. But since I...

Leo: You just don't update Chrome enough.

Steve: Exactly. I'm not running Chrome often enough to give it a chance to auto-update. But it does, it absolutely - in fact, the notes from Adobe talk about Chrome and IE. IE is also now keeping Chrome up to date itself.

Leo: Yeah. Flash, yeah.

Steve: Both of them - yeah, I'm sorry, right. Both of them Adobe was working with in order to push these zero-days out to both Google and to Microsoft so that they in turn could get them out to their browser users.

Leo: Well, I am now disabling Flash. I'll see what happens. Unfortunately, you need it to watch, for instance, TWiT on Ustream and some of our other - although, you know, now that Google's YouTube has eliminated Flash entirely, it's all HTML5.

Steve: I know, it's 100% HTML5.

Leo: I have to think we're getting to that point where everything will be HTML5, HTML5 with HLS.

Steve: Didn't I just - was it on one of your podcasts, there's a compiler now that's compiling JavaScript into - oh, no, it's compiling Flash SWF into HTML5 and JavaScript. So we're seeing, really, I think we're approaching the end of that, which can't come too soon because here they are now, still, two zero-day exploits, people actively infecting people's computers, sometimes with Flash-based ads [crosstalk].

Leo: That's the real problem because a lot of these ad networks are automatic. Yahoo! and Google don't really check the contents of the ads. And so it's easy to buy an ad that has malware in it. Somebody's saying, I don't know if this works, but you can, if you modify the - let's say you have other users on your machine. You've disabled it. I guess you could just go into their browser and disable it. But you could also create a new shortcut bookmark or icon that says "--noflash chrome.exe."

Steve: Ah.

Leo: So maybe that's another...

Steve: So that at launch time...

Leo: At launch it's disabled, yeah.

Steve: It's disabled, yes.

Leo: Yeah. I haven't tried that, but somebody said to try that. And that's Windows, of course.

Steve: That's a great hint.

Leo: Brian Williams - not that Brian Williams - in Kentucky, although maybe he is, I don't know - wonders about...

Steve: I think he's hiding somewhere.

Leo: Hiding in Kentucky, wonders about personal certificates. Steve, we hear a lot about server certificates, but what about personal certificates? I mean, each side of a TLS key exchange requires a cert; right? What else can we do to beef up the crypto on our end for our client apps that do SSL/TLS? Can we periodically generate new certs with higher bit-lengths than default? Thanks for all you do.

Steve: So I think there's a bit of a misunderstanding that I wanted to address in Brian's question. It is not the case that each end of an SSL connection uses a certificate. Each end can, and it's possible for the server side, for some connections that are sort of based on the server's configuration, to query the client and require a client certificate. And it is good security to do so, the idea being that you would install a user certificate or a client certificate in the browser, and then that would assert your identity to the server in exactly the same way that the server's identity is being asserted to the client, that is, when we go to GRC.com, for example, or Amazon, or Google, over an HTTPS connection, that remote server's identity is being asserted and guaranteed by the certificate that it has sent. Similarly, clients can offer certificates, not just username and password or other stuff, but an actual SSL certificate can be installed on the client.

But it never really caught on. It's not the way people typically operate. Apps can have client certs, which they use in order to assert their app identity to a remote, like to the company that publishes them, when they want to establish security where each end is authenticated. But again, that's not the typical browsing experience. Most browsing is single-side authentication where that side is the server, not the user.

Leo: Steve, you did it again. Ten questions. We've come to the end of our Q&A Episode #206. Almost as many question-and-answers as Microsoft has updates today.

Steve: It was close.

Leo: It was close. I thank you for doing such a bang-up job to help keep us all safe and secure on the Internet and remind people that they should go to Steve's website, GRC.com, to get a copy of SpinRite, yes, the world's best hard drive maintenance and recovery utility, and also all the freebies he offers, like SQRL, and you find out more about all the stuff he's up to. You should have a science fiction page, too, by the way, with all the recommendations in there and stuff.

You can also find this show. He actually has two forms of the show. He has the 16Kb audio, very low-bandwidth audio, for people who don't want to download a giant audio file. But he also does transcriptions, which are the smallest version of all, great

transcriptions by Elaine Farris. You can get all of that at GRC.com/securitynow. We will do questions again in a couple of episodes. If you want to leave him a question, don't bother to email, just GRC.com/feedback; or you can tweet him, @SGgrc, and that's a good way to get a hold of him. Steve actually is quite active on Twitter now.

Steve: It's a great social medium.

Leo: Yeah, it is. I agree. Let's see, what else? Oh, we have full-quality audio and video on our site, TWiT.tv/sn, so you can watch Steve as he answers those questions and waves his hands. You can also subscribe, which is probably the best thing to do. That way you'll get every episode automatically. That's kind of the idea behind podcasts. iTunes has it, very single podcatcher in the world, including all your apps on your smartphone and everything, they all have Security Now! because it is one of the longest running shows on podcasting. Ten years soon, yes?

Steve: And it seems to be having the reverse of podfade.

Leo: The reverse of podfade. That's what we specialize in, reverse podfade. So, yeah, please do subscribe. That way you'll get it. Or get one of the great TWiT apps on every platform, including iOS, Android, Windows Phone, and Roku. And you can watch, like on the big-screen TV if you want, on your Samsung. You can talk to Steve, and the Samsung will respond. That's fun. Or the Alexa, one or the other. Sorry. Oh, man.

Steve: Uh-oh.

Leo: Sorry, Lisa.

Steve: The "A" word.

Leo: Alexa, thank Lisa for listening for us, will you? Thank you all for listening. We'll see you next time on Security Now!.

Steve: Thanks, Leo.